Research Journal of Applied Sciences, Engineering and Technology 9(3): 165-171, 2015
DOI:10.19026/rjaset.9.1391
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: March 19, 2014 Accepted: May 04, 2014 Published: January 25, 2015

Research Article

Exploratory Study to Assess and Evaluate Requirement Specification Techniques Using Analysis Determination Requirements Framework

Souhaib Besrour, Lukman Bin Ab Rahim and P.D.D. Dominic
Department of Computer and Information Sciences, Universiti Teknologi PETRONAS,
31750 Tronoh, Perak, Malaysia

Abstract: Various requirements engineering techniques have been proposed to enhance the quality of requirement as well as the implemented software and to growth customer satisfaction with the product. However the large number of requirement techniques makes the choosing between them complicated and confusing. Consequently, this study purposes to present an exploratory study to evaluate requirement techniques and illuminate their key features, limitations and strength.

Keywords: Requirements engineering, requirement specification, specification technique, technique evaluation

INTRODUCTION

It is well known that decent requirement process has a positive impact on final system. Requirement engineering involves different important processes, for instance requirements elicitation, requirements analysis, requirements specification and requirements verification (Schamai *et al*., 2011), (Loinig *et al*., 2011; Fanmuy *et al*., 2012). The objective of requirement engineering is to acquire correct requirements based on customer needs (Machado *et al*., 2005), (Lavazza and Valetto, 2000). There are several requirement engineering techniques purposes to create better requirement quality (Walia and Carver, 2013). This study intentions to evaluate a total of 4 diverse requirement engineering techniques to eliminate their main features, limitations and strength (Hughes, 2003). Additionally, this study presents an exploratory study to evaluate requirement engineering techniques. Numerousauthors and researchers have given diverse definitions of requirements engineering. According to Sommerville and Sawyer (2009), "*Requirements engineering is key process which covers all the activities of discovering*, *maintaining and documenting a group of requirements for a computer-based system*." Requirement engineering comprises four main processes: elicitation, analysis, specification and validation.

The following section offers more details about RE processes mechanisms.

RE PROCESSES

Requirement elicitation is the first process in requirement engineering; it is the process of collecting and acquiring requirement for the developed system. It purposes to acquire customer needs, system functions and constrains. Requirement elicitation is a sophisticated process where customers' desires need to be correctly understood to obtain the exact requirement (Cheng *et al*., 2009; Sherrell, 2013). Consequently, it requires adequate skills in working with social concerns and system constrains. Numerous techniques are available for acquiring requirements for instance: interview (Hansen *et al*., 2010; Hands *et al*., 2004), Card sorting (Goldstein and Naglieri, 2011; MacPherson and Della Sala, 2000), brainstorming (Kunifuji *et al*., 2007; Herrmann and Nolte, 2010) and Joint Application Development (JAD) (Duggan and Thachenkary, 2003). The second process in requirement engineering is analysis; it purposes to simplify requirement structures and meanings. Requirement analysis reflects the process of discovering "what" to build rather than "how" to build. Additionally, themain techniques for analyzing requirement are: Activity Diagram (André *et al*., 2014), Scenario based analysis (Use-case) (Schlee, 2013), Entity Relationship Diagram (ERD) and (Sumathi and Esakkirajan, 2007) Kano model Analysis (Yadav *et al*., 2013; Chuang and Chen, 2013). The third process in requirement engineering is specification; it purposes to document, record and note system requirement and to specify customer needs effectively and simply.

Therefore, even after project completion, requirement specification can be used as a part of project contract and as a base of additional system improvement. There are various techniques for requirement specification for instance: Structured Natural Language Specification (Johannisson, 2007),

Corresponding Author: Souhaib Besrour, Department of Computer and Information Sciences, Universiti Teknologi PETRONAS, 31750 Tronoh, Perak, Malaysia

Decision table based-specification (Tsang and Suyun, 2010; Alkharboush and Li, 2012), IEEE software requirement specification (SRS) (Rashwan, 2012), DIRT (DIRT, 2008). Lastly, the fourth process in requirement engineering is verification: it is the process of conforming system requirements and that client' needs are appropriately understood. Founded errors and mistakes can easily be repaired during the early stages of system construction. Therefore requirement faults are highly costly after the system is shaped (Sherrell, 2013). Thus, requirement verification has a critical function in reducing project cost significantly (Sommerville and Sawyer, 2009). There are various techniques for requirement verification for instance: Checklist-based validation (Brace and Ekman, 2012; Gupta and Goel, 2012), Ad-hoc based validation (Bjoner, 2006), Misuse-case (Soomro and Ahmed, 2013; El-Attar, 2012) and peer review validation (Moed, 2005). Next section discusses briefly details about RE methodologies.

ADRF

Analysis Determination Requirement Framework (ADRF) is a comparison frameworks designed in University of Texas by, (Sur. B. and Ralph Bravoco). Moreover, the framework fundamental purpose is to analyze, assess requirement engineering techniques and to evaluate technique complexity. ADRF encompasses four chief evaluation dimensions. First dimension is syntactic: to assess requirement engineering techniques regarding their contents and syntactic structure, as presented in Table 1 this dimension encompasses three chief criteria: Consistent Level of Abstraction and Detail, Consistent Viewpoint and Purpose and Complexity of the Syntax. This dimension result contains two chief consequences: Syntactical correctness of the model and Syntactic Completeness of the Model. Additionally, ADRF second dimensions is semantic: this dimension purposes to assess requirement engineering techniques based on how it works and decomposing degree, this dimension encompasses four chief criteria: Appropriate level of abstraction, Complexity/Understandability, Type of decomposing, Proper view point and purpose (Yadav *et al*., 1988).

Furthermore this dimension encompasses two fundamental consequences: Semantic correctness of the model and Semantic Completeness of the Model. Additionally (Table 1) outline the first two dimension including chief criteria and dimension consequences.

ADRF third dimension is communication ability: it assesses understandability and readability of document formed by the use of requirement engineering technique. Subjects who read of the document are requested to assess the communicability of the requirement engineering techniques. ADRF fourth dimension is usability: it refers to measurement of how usable the technique is. Noteworthy, the measuring of these criteria is rated directly by users. Furthermore, ADRF is able to be applied in the four requirement engineering pSrocesses: elicitation, analysis, validation and management. Lastly ADRF outcomes and conclusions have to be properly collected and analyzed.

As shown in Fig. 1 ADRF study composed of three chief components: First, case study design. Second conduct case study. Third process evaluation: in order to assess students' works and correctly counting their scores. The experiment can be performed in normal undergraduate class. Additionally, Expert Review Committee (ERC) chief function is to evaluate and estimate final collected results. Furthermore, ERC should apply ADRF structure in detail to assess requirement engineering techniques. Furthermore, ERC are responsible for creating research hypotheses. In the

Table 1: The first two dimension of ADRF

	Modeling process	Final outcome
Syntactic	1. Consistent level of abstraction and detail	1. Correctness and completeness of the model • Inputs • Outputs • Function
	2. Consistent view-point and purpose	1. Correctness and completeness of the model in term of inputs, outputs and functions
	3. Complexity of the syntax	
Semantic	1. Appropriate level of abstract	
	2. Proper viewpoint and purpose	
	3. Type of decomposition	
	4. Complexity/under-standability of the problem	

Table 2: Scoring weights for various criteria

	Modeling process	Final outcome
Syntactic	1. Consistent level of abstraction and detail-2 Points	1. Correctness and completeness of the model- 7 points
	2. Consistent view-point and purpose-1 Point	
Semantic	1. Appropriate level of abstract- 1 Point	1. Correctness and completeness of the model- 7 points
	2. Proper viewpoint and purpose- 1 Point	
	3. Type of decomposition- 1Point	

Table 3: Scoring weights for ADRF four dimensions

I syntactic (0 = <> = 10 MARKS)	II semantic (0 = <> = 10 MARKS)
III communicating ability (0 = <> = 10 MARKS)	IV usability (0 = <> = 10 MARKS)
Utmost score for all dimensions (0 = <> = 40)	

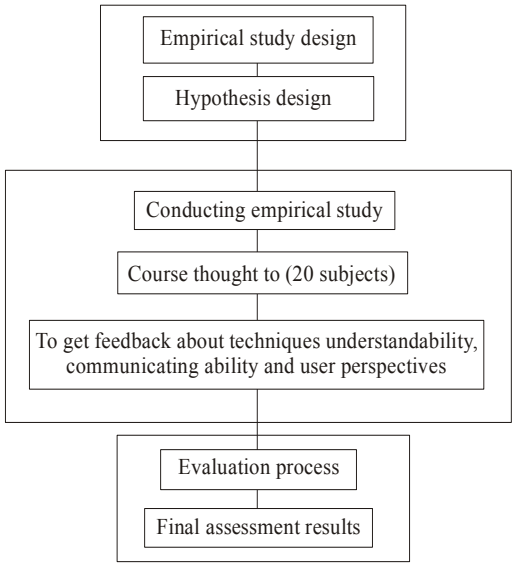

Fig. 1: ADRF flowchart

other hand students should be taught about requirement engineering techniques. Additionally, quizzes should be given to students to conform about their understanding. The total period of study may take 4 class's period (four weeks). Moreover, students have to be be divided based upon the Group Embedded Figures Test (GEFT) scores and on their final scores. Subjects score ranges from 10 to 18, afterward collect all the scores; they separated to (low, high, medium). Furthermore, two subjects are randomly picked from each cognitive style to construct a group. Consequently, a total of 10 groups were formed and assigned to use requirement engineering techniques and to specify the hypothesis. Lastly, surveys have to be spread across subjects to assess each requirement engineering technique and to assign a value for each criteria of requirement engineering technique. Afterward collecting survey documents ERC members assess the study and assign a value of each dimension. Noteworthy that ERC stay blind to participant to avert experimenter bias.

Afterward all groups submit their questionnaire they have to be treated by ERC. Marks will be given for each criterion (Table 2). Concluding score might range theoretically between 0 to 40 (Table 3). Consequently ERC members accumulate the concluding score of each groupbyan iterative method very alike to Delphi approach.

EXPLORATORY STUDY

Exploratory study performed in undergraduate class at Universiti Teknologi Petronas, the study encompasses of three principal steps:

- Analyzing exploratory study methodology
- Exploratory study preparation
- Conducting exploratory study

First, analyzing exploratory study methodology this step occupy 30 days to be wholly accomplished, it encompasses extended reading for exploratory study methodology, investigating exploratory study milestones, reviewing methodology dimensions their arrangements and characteristics, evaluating process, mark management, exploratory study pre-requisite, expected consequences, exploratory study requirements, analyzing exploratory study, analyzing research scope, exploratory study capacity, exploratory study recommendations.

Second, exploratory study preparation, this stepoccupy 60 days to be entirely performed, it encompasses establishing exploratory study milestones, quizzes, formulating training course syllabus, tests, course presentation, designing exploratory study headlines, exploratory study announcements, contacting with expected subjects, answering subjects questions regarding exploratory study, exploratory study time plan, location booking and arrangements. Third, conducting exploratory study, this milestone takes four training classes to be wholly accomplished. Exploratory study performaed with 25 undergraduate students at Universiti Teknologi PETRONAS Malaysia in faculty of computer and information system. ADRF was the principal methodology in the exploratory study. Training course in "software requirement techniques" thought to students during 4 class period each class duration 30 min. After concluding the four classes requirement engineering techniques are entirely thought to subjects. A (Test1) given to the subjects in order to distribute the subjects into groups based on them concluding results (low, high, average) each group should encompasses three different subject level to avert research bias and to make subjects grouping more transparent and clear. Subjects gave quizzes to ensure about their understanding. Midst the exploratory study subject presents great learning excitement and willing to apply different requirement engineering techniques. During third class subjects gave (Test2), the test encompasses of scenario based questions, to assess subjects' abilities and information. Underneath is a sample questions derived from (Test2):

> *Draw a use-case to analyses online university system*.
> *For online banking, write down system specification using structured natural language*.
> *Make a brain storming about auto-driving train*; *security issues*, *features*, *threatens*.

Write system specification for online games using SRS.
Draw an activity diagram. In order to analyses email security issues.
Perform Ad-hoc based validation for the given requirement.
Validate your work based on Checklist-based validation.

Finally, in the concluding class subjects gave Analyst Response Questionnaire, so as to assess communication capability and usability of requirement engineering techniques based on subjects' feedback. After completion of exploratory study, analysis process begins in order to assess subjects' works and answers. Final evaluation process done by, ERC which combines of three participant who they have a decent knowledge about software engineering. The following section presents exploratory study statistics and findings.

RESULTS AND DISCUSSION

Final scores are categories into four principal demotions which they are discussed independently in order to focus on the findings of each dimension:

- Syntactic
- Semantic
- Communicating ability
- Usability

Syntactic dimension: It presents syntactic rules or limitations in the sense that they mirror logically what is presented graphically. Additionally, this dimension encompasses numerous criteria for instance: Syntactical correctness, this criterion reflects the level to which a model is correctly labeled, constructed and identified; it is the evaluation to which syntactic processes have been clarified suitably in diverse designs. Complexity of the syntax; this criterion discusses about understandability of basic syntax of a technique. The final criteria of this dimension is syntactic completeness it refers to the level to which all labels, boxes, notations and arrows are existing in a model. As shown in Fig. 2 SRS displays high accuracy regarding syntactic correctness it obtains 8.71 points witch's the dominant score in this dimension. In the other hand, DIRT obtains 5.92 points which is considered to be the lowest score in syntactic dimension. Furthermore, Structured NL obtains 6.71 point, which is a decent score that reveal a good precision in syntactic dimension.

Semantic dimension: Semantic rules are evaluated by the modeling process and subjects works. This dimension encompasses of a number of criteria's for instance; Appropriate Level of Abstraction it discusses about the degree to which an decomposition is suitable for a particular scale. Appropriate viewpoint and purpose: it refers to analyzing process if the viewpoint

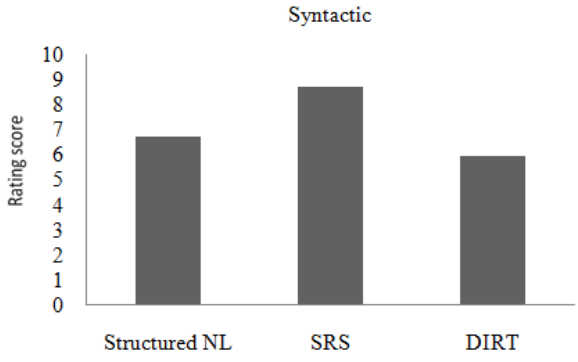

Fig. 2: Syntactic dimension score summary

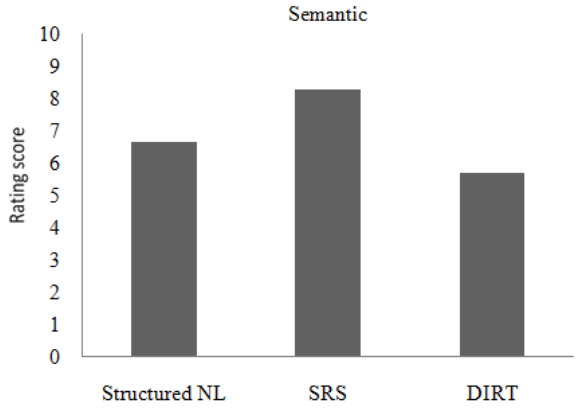

Fig. 3: Semantic dimension score summary

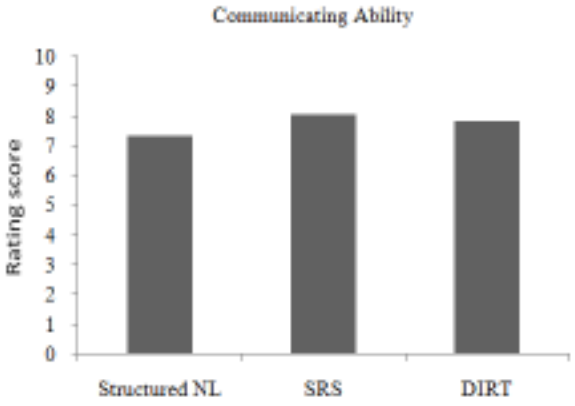

Fig. 4: Communicating ability dimension score summary

and purpose of the model is decent and proper. Moreover, Complexity/understandability this criterion evaluated by analyzing the following:

- Level to which the purpose of a model is obvious to the assessor.
- Extend to which "what the model reveal".

The latest two criteria in this dimension are semantic completeness and correctness of the model. As shown in Fig. 3 Structured NL obtains 6.64 points

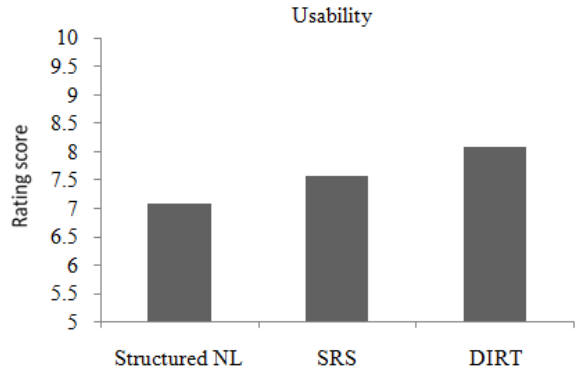

Fig. 5: Usability dimension score summary

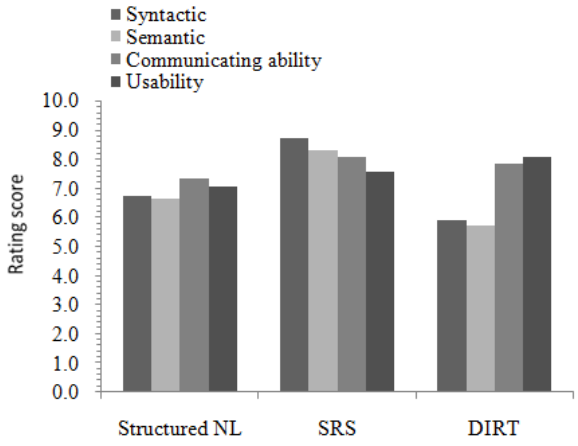

Fig. 6: Four dimension total score summary

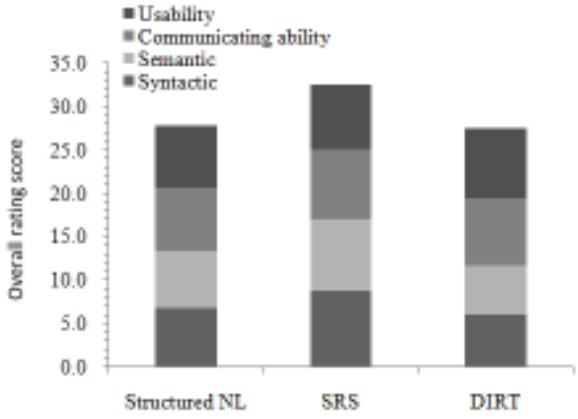

Fig. 7: Four dimension overview summary

which is an average score in this dimension. In the other hand DIRT obtain 5.71 point which is considered as the lowest score in Semantic Dimension. Lastly, SRS gains 8.28 point which is statitically signifigant and it's the uppermost score in this dimension.

Communicating ability: This dimension refers to the readability and understandability of the model produced by the use of the technique. Students are requested to assess this dimension. Subsequently collecting data from all students, the final scores have been calculated and summarized as revealed in Fig. 4. Structured NL gain 7.35 point in this dimension which is statistically the lowermost score in this dimension. In the other side SRS obtains 8.07 point which is the highest score in this dimension. Based on subjects' feedback DIRT gain 7.85 point which is decent score in this dimension.

Usability: This dimension assesses the level of how usable a technique is. Particularly it evaluate, the degree of complication and challenges faced by participant dealing with the technique. The usability of a technique depends on its approach and syntax. Noteworthy, usability of a technique can be objected by very vague or a rigorous syntax. As shown in Fig. 5 final scores gains by techniques are close to each other's, which reflects high contest between techniques in this dimension. In this dimension Structured NL and SRS gain 7.07 and 7.75 points, respectively which reflects a close scores and great competitiveness between afore mentioned techniques. In the other hand DIRT obtains 8.07 points which is the utmost score in this dimension.

FOUR DIMENSIONS SUMMARY

In order to have a border view, all dimensions scores are presented in Fig. 6 and 7. Theoretically the utmost assessment score may ranges between zero to forty ($0 \leq$ *Total score* ≤ 40). Structured NL, SRS and DIRT present a minor score variation in diverse dimensions. SRS and Structured NL gains 8.71 point and 6.71 point in syntactic dimension which is statistically significant (Fig. 7). Additionally, by having a look at usability dimension there is no large score differences between techniques. Structured NL, SRS and DIRT gains 7.07, 7.57, 8.07 points, respectively. That shows great competitiveness in this dimension. Furthermore, in communication ability dimension, SRS obtains 8.07 point which is statistically the highest gained mark in the dimension, trailed by 7.85 point for DIRT. Lastly, Structured NL is the lowest evaluated technique by students in this dimension thus it acquires 7.07 point. Generally speaking and Based on the total rated scores RE techniques reflect a great competitiveness between them. Nevertheless, statistics reflects (Fig. 6) that SRS acquires 32.63 point which is the highest gained score among all techniques, followed by Structured NL which acquires 27.77 point and finally DIRT acquiring 27.55 point which is statistically the minimal gained score.

CONCLUSION

This exploratory study purposes to assess three requirement spesification techniques: Structured natural

language, DIRT and SRS, exploratory study performed with 25 undergraduate students in order to assess different requirement engineering techniques. Students tough about requirement specification techniques their capacity, features and how they work. Moreover, quizzes have given to students to conform about their knowledge and understanding. Additionally, exploratory study performed in four classes, each class 30 min. Test 1 given to students to categorize students into diverse groups based on them results. However, in class three, Test 2 given to student to assess their abilities in using different requirement engineering techniques. Finally in last class a research survey filled by students to get same feedback from students' perspective about RE techniques. Noteworthy that ADRF framework was the principal methodology in this exploratory study. As a future work experimental quantitative study with larger users' quantities may presents further detailed findings. Moreover the utilizing of different methodologies might offer new insights about assessing requirement engineering techniques for both for industry and academia.

ACKNOWLEDGMENT

I would like to acknowledge Universiti Teknologi PETRONAS, which had given me the chance to study and perform my exploratory study in CIS department. Moreover, I would like acknowledge computer information science department members for their dedication and guidance. Lastly I would like to acknowledge all 25 participants in this exploratory study for their feedback, hardworking and willing to learn.

REFERENCES

Alkharboush, N. and Y. Li, 2012. A decision table method for randomness measurement. In: Watada, J., T. Watanabe and G. Philips-Wren (Eds.), Intelligent Decision Technologies: Proceeding of the 4th International Conference on Intelligent Decision Technologies (IDT'2012). Springer-Verlag, Japan, 1: 13-23.

André, É., C. Choppy and G. Reggio, 2014. Activity diagrams patterns for modeling business processes. In: Lee, R. (Ed.), SERA, SCI 496, pp: 197-213.

Bjoner, D., 2006. Requirements Verification and Validation. Software Engineering 3, Texts in Theoretical Computer Science an EATC Series, pp: 503-509.

Brace, W. and K. Ekman, 2012. CORAMOD: A checklist oriented model-based requirements analysis approach. Requir. Eng. J., 19: 1-26.

Cheng, B.H.C., R. de Lemos, H. Giese, P. Inverardi, J. Magee *et al*., 2009. Software Engineering for Self-adaptive Systems: A Research Roadmap. In: Cheng, B.H.C. *et al*. (Eds.), Self-Adaptive Systems. LNCS 5525, Springer-Verlag, Berlin, Heidberg, pp: 1-26.

Chuang, P.T. and Y.P. Chen, 2013. Applying KANO model to exploit service quality for the real estate brokering industry. Proceeding of the Institute of Industrial Engineers Asian Conference, pp: 783-791.

DIRT, 2008, Department of information resources. Software Requirements Specification Template, Version 1.2.

Duggan, E.W. and C.S. Thachenkary, 2003. Higher quality requirements: Supporting joint application development with the nominal group technique. Inform. Technol. Manag., 4(4): 391-408.

El-Attar, M., 2012. From misuse cases to mal-activity diagrams: Bridging the gap between functional security analysis and design. Softw. Syst. Model., 13: 173-190.

Fanmuy, G., A. Fraga and J. Llorens, 2012. Requirements verification in the industry. In: Hammami, O., D. Krob and J.L. Voirin (Eds.), Complex Systems Design & Management (CSD&M). Paris, pp: 145-160.

Goldstein, S. and J.A. Naglieri, 2011. Card-sorting test. Encyclopedia of Child Behavior and Development, pp: 313.

Gupta, K. and A. Goel, 2012. Requirement checklist for blog in web application. Int. J. Syst. Assur. Eng. Manage., 3(2): 100-110.

Hands, K., R. Peiris and P. Gregor, 2004. Development of a computer-based interviewing tool to enhance the requirements gathering process. Requir. Eng., 9(3): 204-216.

Hansen, S., K. Lyytinen and M. Jarke, 2010. Interview with fred brooks on "building effective large-scale requirements". Bus. Inform. Syst. Eng., 2(3): 191-200.

Herrmann, T. and A. Nolte, 2010. The integration of collaborative process modeling and electronic brainstorming in co-located meetings. In: Kolfschoten, G., T. Herrmann and S. Lukosch (Eds.), CRIWG 2010. LNCS 6257, Springer-Verlag, Berlin, Heidberg, pp: 145-160.

Hughes, J., 2003. The requirements problem, designing collaborative systems. Computer Supported Cooperative Work, pp: 3-43.

Johannisson, K., 2007. Natural language specifications. In: Beckert, B. *et al*. (Eds.), Verification of Object-oriented Software. LNAI 4334, Springer-Verlag, Berlin, Heidberg, pp: 317-333.

Kunifuji, S., N. Kato and A.P. Wierzbicki, 2007. Creativity support in brainstorming. In: Wierzbicki, A. and Y. Nakamori (Eds.), Creative Environments. SCI, Springer-Heidberg, 59: 93-126.

Lavazza, L. and G. Valetto, 2000. Requirements-based estimation of change costs. Empir. Softw. Eng., 5(3): 229-243.

Loinig, J., C. Steger, R. Weiss and E. Haselsteiner, 2011. Simulation based security requirement verification for transaction level models. In: Erlingsson, U., R. Wieringa and N. Zannone (Eds.), ESSoS 2011. LNCS 6542, Springer-Verlag, Berlin, Heidberg, pp: 264-271.

Machado, R.J., I. Ramos and J.M. Fernandes, 2005. Specification of requirements models. In: Aurum, A. and C. Wohlin (Eds.), Engineering and Managing Software Requirements. Springer, Berlin, Heidelberg, pp: 47-68.

MacPherson, S. and S. Della Sala, 2000. Welcoming normative data for wisconsin card sorting test. Neurol. Sci., 21(5): 258-260.

Moed, H.F., 2005. Peer review and the use and validity of citation analysis. Inform. Sci. Knowl. Manage., 9: 229-237.

Rashwan, A., 2012. Semantic analysis of functional and non-functional requirements in software requirements specifications. In: Kosseim, L. and D. Inkpen (Eds.), Canadian A1 2012. LNAI 7310, Springer-Verlag, Berlin, Heidberg, pp: 388-391.

Schamai, W., P. Helle and P. Fritzson, 2011. Virtual verification of system designs against system requirements. Proceeding of ACES-MB Workshop (MoDELS 2010), 6627: 75-89.

Schlee, C., 2013. Use Case Identification and Description. Targeted Advertising Technologies in the ICT Space, pp: 61-93.

Sherrell, L., 2013. Software Engineering. Encyclopedia of Sciences and Religions, pp: 2180-2189.

Sommerville, I. and P. Sawyer, 2009. Requirements Engineering: A Good Practice Guide. Wiley India Pvt. Ltd., New York.

Soomro, I. and N. Ahmed, 2013. Towards security risk-oriented misuse cases. In: La Rosa, M. and P. Soffer (Eds.), BPM 2012 Workshops. LNBIP 132, Springer-Verlag, Berlin, Heidberg, pp: 689-700.

Sumathi, S. and S. Esakkirajan, 2007. Entity-relationship model, fundamentals of relational database management systems. Stud. Comp. Intell., 47: 31-63.

Tsang, E. and Z. Suyun, 2010. Decision table reduction in KDD: Fuzzy rough based approach. In: Peters, J.F. and Skowron (Eds.), Transactions on Rough Sets XI, LNCS 5946, Springer-Verlag, Berlin, Heidelberg, pp: 177-188.

Walia, G.S. and J.C. Carver, 2013. Using error abstraction and classification to improve requirement quality: Conclusions from a family of four empirical studies. Empir. Softw. Eng., 18(4): 625-658.

Yadav, H.C., R. Jain, A.R. Singh and P.K. Mishra, 2013. Aesthetical design of a car profile: A Kano model-based hybrid approach. Int. J. Adv. Manuf. Tech., 67(9-12): 2137-2155.

Yadav, S.B., R.R. Bravoco, A.T. Chatfield and T.M. Rajkumar, 1988. Comparison of analysis techniques for information requirement determination. Commun. ACM, 31: 1090-1097.