# Can Scoring Rubrics be used in Assessing the Performance of Students in Software Requirements Engineering Education?

Emmanuel O.C. Mkpojiogu, Azham Hussain
*Human-Centered Computing Research Lab, School of Computing, Universiti Utara Malaysia, Malaysia*
*emelnuel@hotmail.com*

***Abstract*—In this paper, the authors investigated how helpful scoring rubrics are in the assessment of the performance of students in software requirements engineering education and how the use of the tool can lead to the enhancement of students' performance in the design of software requirements artifacts and work products. In the study, two instructors employed scoring rubrics to assess the cognitive performance of a learner (student) in the design of software requirements work documents and artifacts. The outcome of the study shows that the utilization of scoring rubrics is very supportive in assessing software engineering students' cognitive performance without the usual bias associated with other assessment tools. In addition, the findings indicated that the employment of scoring rubrics also helps in pointing to whether a student is either improving or not improving in an iterative or a repeated assessment. In sum, it can potentially lead to the enhancement of the learning capacity and cognitive performance achievements of requirements/software engineering students. The study's findings offer some insights for future investigations and will provide benefits to researchers and professionals in requirements engineering, software engineering and particularly, software engineering and software requirements engineering education.**

***Index Terms*—Engineering Education; Assessing Students' Performance; Scoring Rubrics; Software Requirements.**

## I. BACKGROUND

The outcome of learning squarely depends on the learning process and the tools used in the process. The success of software engineering education and software requirements engineering education efforts can be undermined by a given learning process. Therefore, software requirements engineering education and software engineering education requires a state of the art mechanisms and learning apparatuses for positive learning outcomes. Software requirements engineering and software engineering students are generally obligated as a component of their curriculum to learn, design and develop a number of software artifacts and work documents. These artifacts are inter alia: list of requirements, software requirements specification, use case documents, test plan documents, test cases. In addition are software requirements models such as activity diagrams, use case diagrams, sequence diagrams, collaboration diagrams, state diagrams, class diagrams, etc. Achieving these learning outcomes poses a challenge to both the instructor and the individual learners. The teaching and learning of the design and development of software requirements artifacts and work products require the realization of both formative and summative learning goals and outcomes, which will eventually lead to the enhancement of the performance of students in software requirements engineering education especially in the learning, design, and development of requirements artifacts, work products, and models. To keep an eye on the performance enhancement progress of students, it is needful for assessments (whether teacher-based or student-based assessment) to be carried out with simple but comprehensive tools/methods that enable such students' performance mentoring and monitoring. Several methods exist for assessing the cognitive performance of students (for instance, the traditional marking process of achievements tests). The drawbacks of some of these tools are that they are subjective, could be biased, non-transparent with arbitrary grading, are not sufficiently student-focused, are not adequately interactive, do not facilitate students' self-grading, and do not offer sufficient feedback machinery for both learners and their tutors for the enhancement of the learning process and outcomes [16][18].

The utilization of scoring rubrics in engineering education and in higher education, in general, seems to have commenced as a result of the deficiencies observed with the use of the traditional marking process which has been criticized for its rater bias, and uncommon standards/criteria [15] [16]. There appears to have been a glide towards the employment of scoring rubrics lately (the tool has the potential of countering arbitrary grading, subjectivity, and offering some transparency in the marking process) [16] [18]. In addition, the movement towards the utilization of scoring rubrics is also motivated by the desire of students to contribute to or be part of the planning and monitoring of their progress in line with their teacher expectation [1]. This makes the learning process a more student-oriented process and activity and also makes the individual learners to be personally involved and to fully participate in their learning efforts. The use of rubrics offers a self-learning environment for students and gives them the opportunity to self-grade themselves and personally monitor their individual learning progress and achievements. This, if done, gives them self-esteem, confidence, motivates them to learn and improve their positive attitude and commitments towards their teachers and the given course of learning [19]. The learner-centric approach lays emphasis on students' activity engagement hinged towards the achievement of the learners' learning outcomes [22]. The utilization of scoring rubrics gingers students' learning and assessment altogether [3]. Furthermore, scoring rubrics enhances the reliability of assessments, they also have the capacity of promoting learning and enhancing instruction and teaching [7] [8]. The

utilization of scoring rubrics offers a good opportunity for formative feedback to improve the learning of students and also makes needed information available to instructors for the purpose of course improvement and enrichment [9]. Prior research also shows that rubrics can be employed in longitudinally assessing students' development over time. Its adoption encourages and enables instructors and teachers to establish alignment between curriculum, course outcomes and program outcomes [9]. Rubrics facilitate cross and longitudinal comparisons [4] and assess students' cognitive and psycho-motor abilities with their effective capabilities. It is also used to judge the sufficiency of learners' responses to performance, achievement and/or aptitude tests [20]. A performance assessment requires students to use their knowledge and produce something (in the case of requirements engineering: artifacts, designs, work products and models) [2]. The employment and appropriate deployment of scoring rubrics is advised to address assessment validity, as the instrument and mechanism contribute to the quality of assessments and also facilitates valid judgments of complex competencies and abilities [7] [8]. In a nutshell, the payback and profits coming from the taking up and utilization of scoring rubrics are as follows: improved scoring and grading consistency, promotion of learning, encouraging and increasing interest in self-learning, and the likelihood and prospect of facilitating valid judgment of complex competencies devoid of bias [8].

One of the major aims and goals of learning for engineering students (captured in the engineering curriculum) is the learning of engineering design [2]. In requirements engineering specifically, the design and development of requirements artifacts, designs, work products, and models are an integral part of the main learning objectives and curriculum. The design and development of these artifacts, work products, and models, however subjective, implying that there are no standard mathematical proofs or any conclusive experiment to validate the design processes [2]. Scoring rubrics nonetheless offer an objective validation of the assessments of these requirements artifacts and models. It also offers a common answer and platform for enhancing rater reliability [2]. Rubrics can be used in assessing the quality of students' work products and documents (like the software requirements specifications, use case descriptions, and other software artifacts) [3]. Furthermore, rubrics offer flexibility for critical and creative solutions among students and also improve grading and scoring consistencies among instructors. It assists instructors to evaluate the understanding of students as well as their creativity in the production of artifacts and other products [14]. Besides, prior research has shown that early engagement of students with design aspects of their engineering curriculum improve on their memorability, retention, and overall success [5] [6] [17]. Moreover, assessment is a major issue in any student-centric and learner-oriented education. It is, however, a continuous process that improves the achievement of students and enhances the curriculum [22]. In addition, scoring rubrics assessment is not simply learner-centric, it also aims at the performance achievement of students [2] [21]. Studies have shown that in requirements engineering and software engineering, scoring rubrics can be employed in reading (i.e., detecting defects) and enhance the quality of requirements work documents, designs, models, and software artifacts [10] [11], and that the utilization of these scoring rubrics is both effective [12] and efficient [13] in finding out or detecting defects/errors in requirements artifacts. This notwithstanding, there is, however, a lack of sufficient research on the use of scoring rubrics in assessing the performance of students in requirements engineering education. This research as an initial pilot study intended to fill this gap and aims to investigate the potential of scoring rubrics in assessing the performance of individual learner(s).

## II. METHODS

This paper focuses on the assessment of an individual student, whose requirements artifacts were assessed by two assessors (instructors) who assessed the student's work products and artifacts twice (i.e., in two iterations/rounds) using scoring rubrics tools. The scoring rubric is an individual-centred and learner-oriented tool and especially focuses on individual learning and performance [2] [21] [33]. The focus is not on a team of students, but rather on the performance of individual students. It can be used on or with a group of students, but with a particular eye on or attention is given to individual usage and performance. Two types of scoring rubrics exist, they include holistic and analytic scoring rubrics. Holistic rubrics are more concerned with the overall performance rather than the individual steps to get to or arrive at the end result. They give attention to the quality of wholeness of the final performance of students [32]. Holistic rubrics are more appropriate for providing a global evaluation of the realization of a benchmark standard at a program level [16]. On the other hand, analytic rubrics structure the assessment by allowing for the consideration of grading of each criterion separately in the construction or development of the rubric [16]. The development of the criteria for each grade level in the rubric is constructed and framed based on a singular property that the assessor/instructor can make a separate qualitative evaluation about the demonstration of achievement of the criterion specified for that property. Analytic rubric works better for assessments that are task-specific [16]. In this paper, the type of scoring rubrics used for assessment was analytic rubrics.

### *A. Construction of Scoring Rubrics*

Scoring rubrics are a two-dimensional Likert-like tool. The columns of the scoring rubrics used in this research represent a 4-point Likert-type rating scale (for example, (1) Not acceptable, (2) Below expectations, (3) Meet expectations, (4) Exceeds expectations; A fifth column is added for "not applicable". The rows comprise of the attributes of the given artifacts or work product. In some cases, the attributes are further defined by criteria. Furthermore, the cells created by the intersection of the rows and columns represent a clear description of the artifacts' attributes with respect to the corresponding rating scale. Each attribute is scored and the scores of all attributes are totalled to form a total score for the given artifact. The scores of the scoring rubrics are in percentages and they indicate the student's performance level. These scores can also be used to compare students' performances in engineering related tasks and designs.

Table 1
Scoring Rubric Framework

| Attributes | Criteria (optional) | Scale of Score | | | | | Score |
|---|---|---|---|---|---|---|---|
| | | 1.(Not acceptable) | 2.(Below expectations) | 3.(Meets expectations) | 4. (Exceeds expectations) | N/A(Not available) | |
| $Attribute_1$ | … | $Cell_{11}$ | $Cell_{12}$ | $Cell_{13}$ | $Cell_{14}$ | … | … |
| … | … | $Cell_{21}$ | $Cell_{22}$ | $Cell_{23}$ | $Cell_{24}$ | … | … |
| … | … | $Cell_{31}$ | $Cell_{32}$ | $Cell_{33}$ | $Cell_{34}$ | … | … |
| $Attribute_n$ | … | $Cell_{n1}$ | $Cell_{n2}$ | $Cell_{n3}$ | $Cell_{n4}$ | … | … |
| Total Score | | | | | | | (%) |

Table 2
An Example of a Simple Scoring Rubric for Activity Diagram Report

| Attributes | Scale of Score | | | | | Score |
|---|---|---|---|---|---|---|
| | 1. (Not acceptable) | 2.(Below expectations) | 3.(Meets expectations) | 4. (Exceeds expectations) | N/A (Not available) | |
| Appropriate symbol of representation | Incorrect notation | Some notations incorrect or misused | Correct notations | So clear and complete | … | … |
| Flow of process presented | Unclear or poorly designed | Incomplete or not well designed | Clear and complete | So simple and clear. The design is understandable to all intended readers | … | … |
| Total Score | | | | | | (%) |

This study was carried out in the School of Computing, Universiti Utara Malaysia. The Software Engineering sub-department has over the years been using scoring rubrics in assessing the quality of students' software/requirements artifacts and models. But the tool has not been empirically validated. This study is part of attempts by the authors to offer an empirical validation of the tool. In addition, this study was part of the study carried out in developing an e-health awareness system [23-30]. After the requirements products and artifacts were developed, they were reviewed in two rounds by two assessors who pointed to pending issues in the artifacts. Each assessor reviewed the requirements artifacts and work products separately. After each round of review, the requirements artifacts and work products were refined. The following research questions guided the study: 1) Does scoring rubrics help in assessing the performance of software requirements engineering students? 2) Does scoring rubrics improve the performance of software requirements students in the development of requirements artifacts? 3) Can scoring rubrics be used in assessing the performance of software requirements/software engineering students? Descriptive statistics were used in answering the research questions.

## III. Results and Discussion

The following section presents the findings of the study and discusses the results obtained. The first assessor's assessment was based on some selected artifacts, however, as can be seen in the two figures (Figures 1 and 2), the instructors'/assessors' assessments produced similar and consistent results. Figures 1 and 2 provide answers to the three study research questions: 1) Does scoring rubrics help in assessing the performance of software requirements engineering students? 2) Does scoring rubrics improve the performance of software requirements students in the development of requirements artifacts? 3) Can scoring rubrics be used in assessing the performance of software requirements/software engineering students? In Figure 1, there is a performance improvement of 4.34% (Figure 1) in the assessment of the student's vision and scope document as assessed by the first assessor. However, there was a performance improvement of 33.33% (Figure 2) of the same artifact when examined by the second assessor (both assessments showed improvements in the student's performance). For the software requirements specification, there was a percentage improvement of 26.31% (Figure 1) and 18.51% (Figure 2) as assessed by the first and second assessors respectively. The percentage improvements in the assessment of the student's performance in the other artifacts are as follows: test cases (0% (Figure 1), 49.99% (Figure 2)), use case specification (9.9% (Figure 2)), list of requirements (100% (Figure 2)), test case document (73.91% (Figure 2)), use case diagrams (46.67% (Figure 2)), activity diagrams (25% (Figure 2)), collaboration diagrams (22.21% (Figure 2)), and class diagrams (38.33% (Figure 2)). From the results presented, it can be observed that the use of scoring rubrics produced improvement as assessed by the different assessors. The use of scoring rubrics has assisted in assessing the performance of the software requirements student in the design and development of requirements artifacts. The assessments were objectively made and devoid of any bias or subjectivity. In addition, there is a consistent student percentage performance improvement for all artifacts assessed except for the sequence diagrams that had a negative performance score (-12.01%, (Figure 2)). However, it appears that the examined student performed better in the development of textual artifacts more than design oriented artifacts (models). Nonetheless, with repeated feedback and guide from the rubrics and with repeated assessment, it is expected that even in the model design and development, the student will ultimately perform exceptionally well. Though this case study is an individual scenario, the findings show some insight into what may likely be the result if a group or team of students is simultaneously assessed using the scoring rubric instrument.

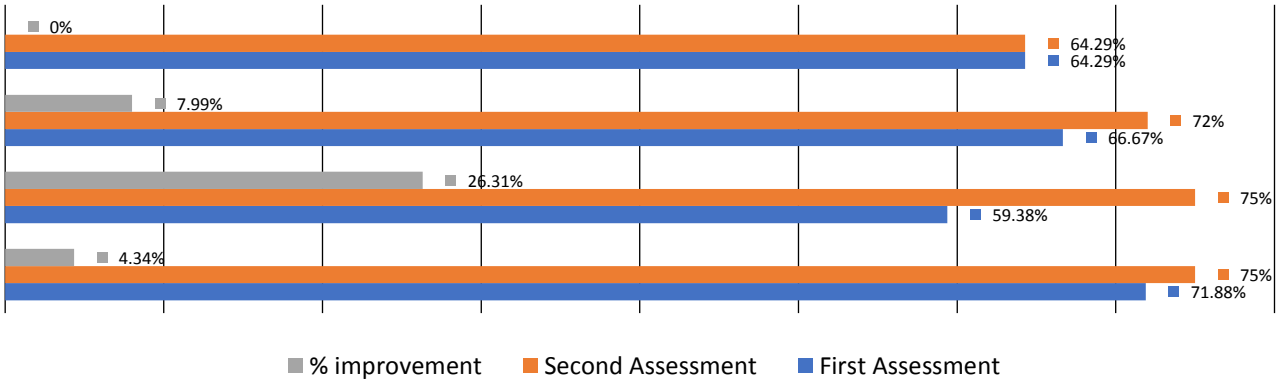


Figure 1: First assessor's rubrics assessment

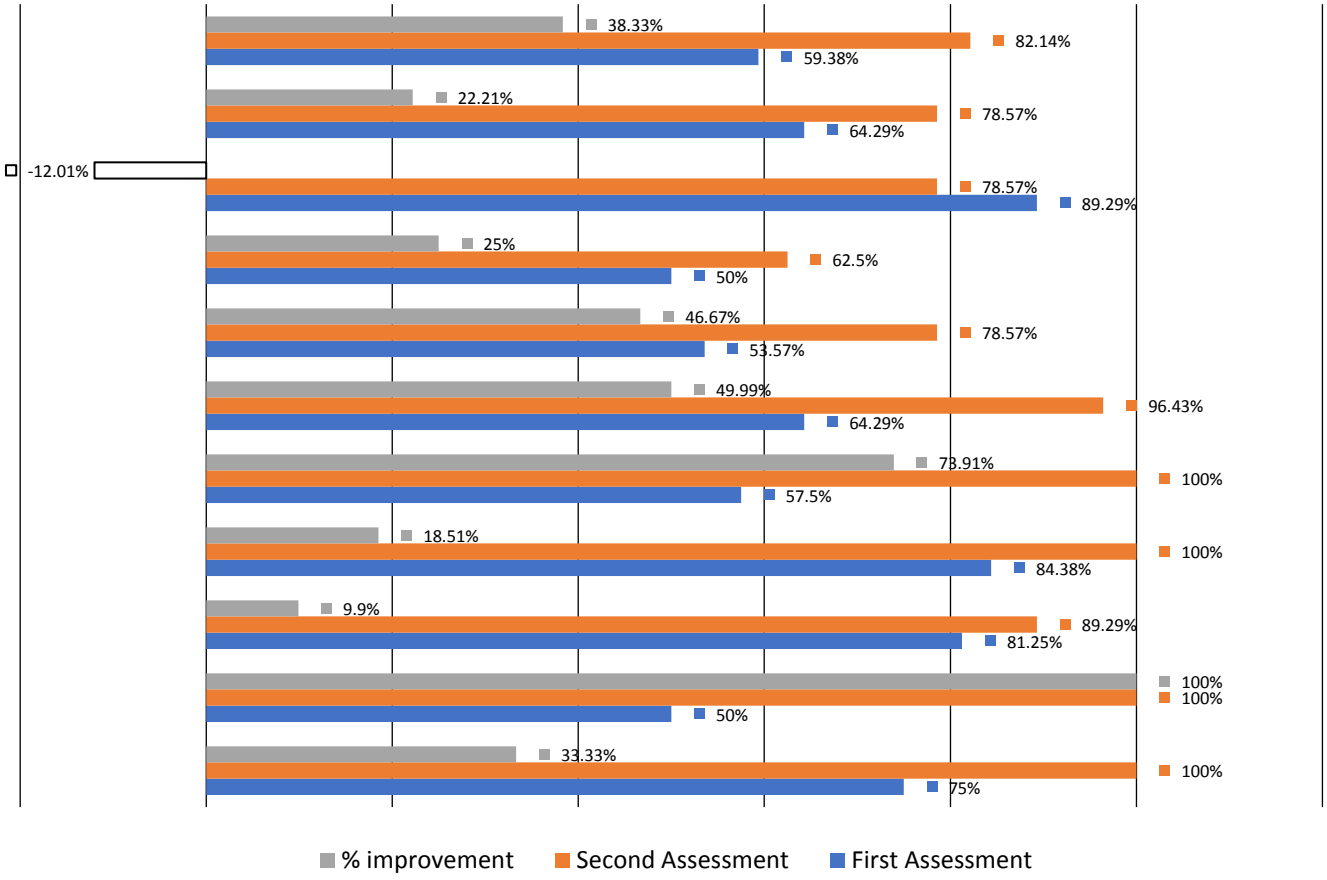


Figure 2: Second assessor's rubrics assessment

## IV. Conclusion and Future Work

The study investigates how helpful the use of scoring rubrics is, in the performance assessment of software requirements engineering students and whether its use can lead to students' performance improvement in the development of software requirements artifacts and models. In the study, scoring rubrics were used by two instructors to assess the cognitive performance of a student in the design and development of software requirements artifacts. The study results indicate that the use of scoring rubrics is very helpful in objectively assessing the performance of software requirements or software engineering students. Furthermore, the results revealed that the use of scoring rubrics can also produce a good achievement assessments direction, showing whether a student is either improving or not improving in a repeat or iterated assessment. In a nutshell, its use leads to the performance improvement of students. The results provided some insights for further investigation and will be beneficial to researchers, requirements engineers, software engineering and requirements engineering educators, designers, developers, project managers and the entire software engineering community.

Scoring rubrics, as a self-learning and self-grading tool, can lead to the performance enhancement of the individual student(s). The main thrust of the study was the work of a particular student (since scoring rubrics is allowed to be used on an individual learner); this nonetheless, is a constraint to the study, as its applicability is limited to an individual learner as opposed to a group or team of learners. In spite of this, however, the study's outcomes indicated and offered a number of useful insights that provoke further investigation

and future study. The authors intend as a future work to evaluate a set of students in a repeat/replicate investigation. The outcomes of this paper will be valuable software professionals and the research community. The study recommends that students (along with their teachers) should be motivated to utilize this instrument as this will assist in improving their skills and performance outcomes (and for instructors, their teaching outcomes). Therefore, employing the use of this tool in the teaching of software engineering and software requirements engineering (especially in the design and development of artifacts, models, and designs) will enable students to not only learn but to critically think and reflect, as they solve engineering/ information related problems [31].

## REFERENCES


[1] H. Andrade and Y. Du, Students use of rubrics. Practical Assessment, Research & Evaluation, vol. 10, no.3, p.1-11, 2005.

[2] R. Bailey and Z. Szabo, Assessing engineering design process knowledge, Int. J. Engineering Education, vol. 22, no. 3, p. 508-518, 2006.

[3] E. Barrella and M.K. Watson, "Developing a cross-disciplinary sustainable design rubric for engineering projects", Engineering Education for Sustainable Development, Bruges, Belgium, 4-7 September, 2016.

[4] M. Besterfield-Sacre, J. Gerchak, M. Lyons, L.J. Shuman, and H. Wolf, Scoring concept maps: An integrated rubrics for assessing engineering education, Journal of Engineering Education, 2004. doi: 10.1002/j.2168-9830.2004.tb00795.x

[5] L. Bodnar, M. Lagoudas, J. Hodge, T. Smith, J. Oronzco, J. Corso, C. Sanchez, J. Freise, H. Ringler, and I. Cortes, "Engaging freshman in team based engineering projects". Proc. American Society for Engineering Education Annual Conference, 2012.

[6] R. Caverly, H. Fulmer, S. Santhanam, P. Singh, J. O'Brien, G. Jones, E. Char, F. Mercede, R. Weinsten, and J. Yost, "Project-based freshman engineering experience: The core course". Proc. American Society for Engineering Education Annual Conference, 2010.

[7] B. Dirjak, D. Grabar, and M. Maretic, Assessing analytics for peer-assessment: a model and implementation, 2016.

[8] A. Jonsson, and G. Svingby, The use of scoring rubrics: reliability, validity and educational consequences. Educational Research Review, vol. 2, no. 2, p. 130-144, 2007.

[9] J. Kaupp, N. Simper, and B. Frank, "Triangulated authentic assessment in the HEQCO learning outcomes assessment consortium". Proc. 2014 Canadian Engineering Education Association Conference, Canmore, AB, June 8-11, 2014.

[10] E.O.C Mkpojiogu and N.L. Hashim, "Improving the quality of requirements work products using scoring rubrics-assisted reading". 6th International Conference on Computing and Informatics 2017 (ICOCI'17), Kuala Lumpur, Malaysia, 25-27 April, 2017.

[11] E.O.C. Mkpojiogu and N.L. Hashim, Does scoring rubrics-assisted reading (SRAR) technique enhance the quality of software requirements artifacts? Journal of Information and Communication Technology (JICT), 2017.

[12] E.O.C. Mkpojiogu and N.L. Hashim, Evaluating the effectiveness of scoring rubric-assisted reading (SRAR) technique in the reading of software requirements work documents. Journal of Engineering and Applied Sciences (JEAS), 2017.

[13] E.O.C. Mkpojiogu and N.L. Hashim, Assessing the efficiency of scoring rubric-assisted reading (SRAR) approach in the review of software requirements work products. Journal of Engineering and Applied Sciences (JEAS) 2017.

[14] A. Mustapha, N.A. Samsudin, N. Arbaiy, R. Mohammed and I.R. Hamid, Generic assessment rubrics for computer programming courses; TOJET: The Turkish Online Journal of Educational Technology, vol. 15, no. 1, January, 2016.

[15] A.R. Razaei and M. Lovorn, Reliability and validity of rubrics for assessment through writing, Assessing Writing, vol. 15, p. 18-39, 2010.

[16] L. Riebe and D. Jackson, The use of rubrics in benchmarking and assessing employability skills, Journal of Management Education, vol. 38, no. 3, p. 319-344, 2014.

[17] M. Roth, "Introduction to geotechnical engineering using a project-based module in a first-year engineering course". ASEE 123rd Annual Conference & Exposition, New Orleans, LA, 26-29 June, 2016.

[18] D.R. Sadler, Indeterminacy in the use of preset criteria for assessment and grading, Assessment & Evaluation in Higher Education, vol. 34, p. 159-179, 2009.

[19] M.G. Simkin, Should you allow your students to grade their own homework? Journal of Information Systems Education, vol. 26, no.2, Spring, 2015.

[20] M. Sindelar, L. Shuman, M. Besterfield-Sacre, R. Miller, C. Mitcham, B. Olds, R. Pinkus and H. Wolfe, "Assessing engineering students' abilities to resolve ethical dilemmas". ASEE/IEEE Frontiers in Education Conference, Boulder, CO, 5-8 November, 2003.

[21] D. van Breukelen, M. Smeets, and M. de Vries, Explicit teaching and scaffolding to enhance concept learning by design challenges, Journal of Research in STEM Education, vol. 1, no. 2, 87-105 December, 2015.

[22] T.G. Wang, D. Schwartz and R. Lingard, Assessing student learning in software engineering, Consortium for Computing Sciences in Colleges (CCSC): Southwestern Conference, JCSC, vol. 23, no. 6, June, 2008.

[23] A. Hussain, E.O.C. Mkpojiogu, and I. Abdullah, Requirements engineering practices in UUMIT centre: an assessment based on the perceptions of in-house software developers. *Journal of Telecommunication, Electronic & Computer Engineering (JTEC)*. Vol. 8, no. 8, p. 27-32, 2016.

[24] A. Hussain, E.O.C. Mkpojiogu and F.M. Kamal, The role of requirements in the success or failure of software projects. *International Review of Management and Marketing*, vol. 6, no. S7, p. 306-311, 2016; ISSN: 2146-4405.

[25] E.O.C. Mkpojiogu and N.L. Hashim, Understanding the relationship between Kano model's customer satisfaction scores and self-stated requirements importance. *SpringerPlus*. vol. 5, no. 1, p. 1-22, 2016; doi: 10.1186/s40064-016-1860-y

[26] A. Hussain, E.O.C. Mkpojiogu and I. Abdullah, Investigation of Current Requirements Engineering Practices Among Software Developers at the Universiti Utara Malaysia Information Technology (UUMIT) Centre. Proceedings of the International Conference on Applied Science and Technology (ICAST'16), Kedah, Malaysia. AIP Conf. Proc. vol. 1761, no. 1, p. 020045, 2016.

[27] A. Hussain and E.O.C. Mkpojiogu, Requirements: Towards an understanding on why software projects fail. Proceedings of the International Conference on Applied Science and Technology (ICAST'16), Kedah, Malaysia. AIP Conf. Proc. vol. 1761, no. 1, p. 020046, 2016.

[28] A. Hussain, E.O.C. Mkpojiogu and M.N.M. Nawi, Requirements model for an e-health awareness portal. Proceedings of the International Conference on Applied Science and Technology (ICAST'16), Kedah, Malaysia. AIP Conf. Proc., vol. 1761, no. 1, p. 020048, 2016.

[29] A. Hussain, and E.O.C. Mkpojiogu, An application of Kano method in the elicitation of stakeholder satisfying requirements for an e-Ebola awareness system. *International Journal of Systems Applications, Engineering and Development*. vol. **10**, p. 169-178, 2016; ISSN: 2074-1308;

[30] A. Hussain, E.O.C. Mkpojiogu, and F.M. Kamal, "Eliciting user satisfying requirements for an e-health awareness system using kano model". In Xiaodong Zhuang (Ed.) Recent Advances in Computer Science-*Proceedings of the 14th WSEAS International Conference on Computer and Computational Science (ACACOS'15)*, Kuala Lumpur, Malaysia, 23-26 April, 2015, p.156-165. WSEAS press. ISSN: 1790-5109; ISBN: 978-1-61804-297-2.

[31] P.A. Asunda and R.B. Hill, Critical features of engineering design in technology education. Journal of Industrial Teacher Education, vol. 44, no. 1, 2007.

[32] R.J. Howell, Exploring the impact of grading rubrics on academic performance: Findings from a quasi-experimental, pre-post evaluation. Journal of Excellence in College Teaching, vol. 22, no. 2, p. 31-49, 2011.

[33] E.O.C. Mkpojiogu, and A. Hussain, Assessing students' performance in software requirements engineering education using scoring rubrics. International Conference on Applied Science and Technology (ICAST'17), Langkawi Island, Malaysia, 3-5 April, 2017.