



SDI Review Form 1.6

Journal Name:	<u>British Journal of Mathematics & Computer Science</u>
Manuscript Number:	2014_BJMCS_14714
Title of the Manuscript:	Software Quality Improvement Using Force-field Analysis
Type of the Article	

General guideline for Peer Review process:

This journal's peer review policy states that **NO** manuscript should be rejected only on the basis of '**lack of Novelty**' provided the manuscript is scientifically robust and technically sound.

To know the complete guideline for Peer Review process, reviewers are requested to visit this link:

(<http://www.sciencedomain.org/page.php?id=sdi-general-editorial-policy#Peer-Review-Guideline>)



[SDI Review Form 1.6](#)

PART 1: Review Comments

	Reviewer's comment	Author's comment <i>(if agreed with reviewer, correct the manuscript and highlight that part in the manuscript. It is mandatory that authors should write his/her feedback here)</i>



SDI Review Form 1.6

Compulsory REVISION comments

- The author mentioned that the goal was to design a software quality improvement model. Nevertheless, the authors only identify *what* needs to be improved and don't explain *how* to improve. The root of the problem is the implementation of the best practices related to supportive factors and to avoid worst practices related to demoting factors.
- A quality improvement model must focus on the implementation and assessment of performed practices in order to describe the reasons of failure and success.
- The authors present data obtained directly from the questionnaires and interviews. They don't analyse or justify those remarks.
- In the proposed model, the same supportive factors and demoting factors appear associated with several and different quality attributes. In my opinion, many of the factors associated with software quality attributes aren't directly related.
- The 14 quality attributes and respective tactics (supportive factors and demoting factors) are well known from the literature (Software Architecture in Practice by L. Bass, P. Clements, R. Kazman).
- In my opinion, the paper describes a case study using force-field analysis to identify the forces that will drive or resist in software improvement. This kind of approach has already been proposed in Juran's Quality Handbook (J. Juran, A. Godfrey).
- The authors should include references concerning all

My modified version of Force-field analysis depicts the procedural steps which entails the expected systematic approach of indulging in Software development projects (supportive forces) as well as the unethical practices that should be avoided (constraints). I stated in my conclusion that if my suggested procedural approach as it relates to each software quality is religiously followed; the subsequent software systems would demonstrate optimal performance of each software quality.

Sir, the content of my paper reveals the details extracted from both the power users; those who make use of CASE tools or downloaded available applications to develop a software system as well as the common users those who make use of the developed software system. Two different structured questionnaires were used to enumerate the opinion of the common users as it relates to software failures and whether or not the software systems are working in conformance to the specification given to their respective developers. The information extracted from the common users was traced back to their respective developers to confirm the unprofessional practices that led to the software failures. Based on these facts, I highlighted fourteen (14) expected software qualities needed for optimal performance. Sir, I can safely conclude that my software development model is based on the implementation and assessment of performed practices in the Nigerian software industries.

Sir, the analysis of the data extracted from my two different questionnaires was clearly stated in the paper (line 212 – 228). Also, line (232 – 237) clearly stated the justification.



SDI Review Form 1.6

	topics mentioned in the article. The title of the paper suggests a software quality improvement process. It isn't the most appropriated concerning the main goal of the paper, i.e., identify software qualities and procedures. On line 469, the authors propose a framework "suggesting the procedures to follow before, during and after the software development process". The model and the procedures described are already known from the state of the art.	With due respect sir, L. Bass et al. never stated the software quality in terms supporting factors and restraining factors. Software qualities are numerically more than 14. I only focused on the most needed qualities expected in an average software system in a multi-user environment. More so, the 14 software quality I picked on was not exactly what L. Bass et.al talked about. <i>The mere fact that L.Bass has worked on software quality does not deprive any author who is given to the same school of thought from contributing to knowledge in that line of interest. However, thanks for your comment.</i> Juran defined the terms quality as "fitness for use" and further gave what he called quality trilogy.
Minor REVISION comments	- In the development of questionnaires, the number of options in the response scale should be even to avoid a mid-point, which acts as a neutral option. - Tables 1-5 show the data clearly, the author doesn't need to describe textually the table information. Authors should analyse and explain how these data have influence on final results.	The data harnessed from the common user shows a high percentage of common users acknowledging average performance of their software systems. This is a justification (a proof) that there is a need for such research to be carried out.
Optional/General comments	The work is not innovative but describes a case in a developing country. This is the main contribution of the paper.	<i>I appreciate your comments. Thank you very much</i>