Identifying Frequently Arising Problems in Software Engineering

Hugo Claus*

Department of Sociology, UC Louvain, Belgium

hugoclaus653@gmail.com

Received: 03 October 2022, Manuscript No. tosocial-22-81300; **Editor assigned:** 05 October 2022, PreQC No. tosocial-22-81300 (PQ); **Reviewed:** 19 October 2022, QC No tosocial-22-81300; **Revised:** 24 October 2022, Manuscript No. tosocial-22-81300 (R); **Published:** 31 October 2022

Description

Reuse of research software requires good documentation, but documentation is often criticized. Lack of documentation, especially in non-IT specific areas, can be attributed to lack of training, lack of time, or lack of reward. This article addresses the hypothesis that while scientist's document, they do not know exactly what, why, or for whom they are documenting. To evaluate practical documentation practices for research software, we examine existing recommendations, use specific engineering examples to evaluate their implementation in day-to-day practice, and translate the results into best-practice examples. Compared to get a rough idea of what research software documentation is needed, we defined categories and used these to conduct our research. Our results show that we lack a complete picture of what documentation means for research software. The recommendations do not take into account the critical role of researchers creating research software. Research software documentation is primarily done in research papers. Furthermore, it shows that research software always has a history that affects documentation. Outsourcing software development is becoming more and more popular due to benefits such as cost reduction, process improvement, and addressing the shortage of necessary resources. Unfortunately, research shows that the majority of software development outsourcing projects does not realize the expected benefits. A study of such project failures revealed several cases of software development outsourcing projects failing due to issues related to the requirements engineering process. The purpose of this study is to identify and categorize problems that frequently arise in the requirements engineering process in software development outsourcing. To this end, the contemporary literature was rigorously reviewed, the problems faced by practitioners were identified, and his three survey surveys involving experienced software development outsourcing practitioners were organized. Categories of commonly encountered problems are also ranked. Identifying and ranking the issues will help you draft a proactive software project management plan to address software development outsourcing failures and achieve the estimated software development outsourcing benefits. We conclude by recommending ways to improve access to structured systematic review results. Major opportunities for improving systematic reviews will come from new tools and changes in policy focused on doing the right systematic reviews rather than just doing more of them faster. Due to certain advantages, the amount of software development outsourcing (SDO) is growing rapidly. Due to challenges arising from the requirements engineering (RE) process, some projects do not get the expected benefits of SDO. The purpose of this research work is to recommend RE practices to address common RE process issues in the case of SDO. For this reason, a thorough literature review was conducted and two of his questionnaire surveys were conducted among experienced practitioners in the SDO industry. The survey was conducted using a simplified sampling method in a semi-supervised style. A 50% rule and a 4-point Likert scale were also used to determine the benefits of RE practices for handling problems. A comprehensive list of 147 RE practices was extracted by conducting focus group sessions. Additionally, 147 RE practices were ranked by applying the numerical assignment and $100 technique in two focus group sessions. Identifying and adapting RE practices can help improve the SDO RE process, avoid SDO failures, and achieve associated SDO benefits.

Acknowledgement

None.

Conflict of Interest

The author has declared no conflict of interest.