Analyzing the Impact of Code Smells On Software Maintainability

Naseef M.N.M¹, Nifthas A.R.M¹, and * KapilanK¹

¹Sri Lanka Institute of Information Technology Department of Information Technology

May 31, 2023

Abstract

The ease with which a software system can be modified or adapted to meet new requirements, fix defects, or improve its performance is referred to as software maintainability. Software maintainability is an essential quality that distinguishes high-quality software from low-quality software. Poor design or implementation choices are examples of code smells. These choices can have a negative impact on the software’s maintainability because they make the code more difficult to understand, modify, and test. In this article, we present a comprehensive analysis of the impact that code smells have on the maintainability of software. Our findings are based on a comprehensive review of the relevant literature as well as a meta-analysis of empirical studies. According to the findings of our study, code smells can significantly lower the maintainability of software, but the degree to which they do so varies depending on the nature and severity of the smell, as well as the programming language used, the method of software development, and the specifics of the software project. Refactoring, testing, documentation, and code reviews are some of the factors that we identify as being able to mitigate the negative effects of code smells. Our findings have important ramifications for software developers, managers, and researchers because they highlight the need for effective strategies to prevent, detect, and remediate code smells in software projects. These strategies can be found in software development environments.

Hosted file