

An Empirical Evaluation of Clean Architecture: Impact on Software Quality and Developer Productivity

Senadheera M.A.K.R.^{1*}, Somaweera W.T.S.² and Sandaruwan R.M.T.²

¹Department of Software Engineering, Faculty of Computing,
Sabaragamuwa University of Sri Lanka, Sri Lanka

²Department of Computing and Information Systems, Faculty of Computing,
Sabaragamuwa University of Sri Lanka, Sri Lanka

*makrsenadheera@std.appsc.sab.ac.lk

Clean architecture(CA) is one of the most reliable and most productive architectures in the modern world. By organizing code into distinct layers, each with a clear responsibility, allows developers to build robust, flexible applications and enhance it quality. However, there no enough empirical evidence to prove that CA has impact on enhance software quality and maintainability. To fill this gap, this study aims to conduct a systematic review of the effects on software quality CA and developer productivity. In order to achieve it, 106 CA users participated in a survey. Likertscale and open-ended questions were used to improve the outcomes.Participants selected through LinkedIn ,GitHub platforms and snowball sampling. After the data collection and cleaning correlation test was conducted and its analysis outlines CA adoption links with major ISO/IEC 25010 standards quality attributes. Significant positive correlations were found between the implementation of CA and overall software quality ($r = 0.428$, $p < .001$). In this respect, managing dependencies results in the most substantial impact on quality ($r = 0.707$) and productivity ($r = 0.691$), followed by Separation of Concerns ($r = 0.485$ for quality; $r = 0.443$ for productivity). These findings demonstrate that developers are accustomed to utilizing CA in their applications. Results from open-ended questions demonstrate the CA's adaptability. The overall advantages of CA can be greatly increased under appropriate mentorship, to help reduce technical debt, and to provide an appropriate architectural template. There are certain issues, such as steep learning curves, particularly for younger engineers during onboarding; tools and framework constraints cause developers to deal with greater complexity. All of this demonstrates that CA may improve software quality and developer efficiency, offering insightful information on its successful adoption and sustainability to industry teams, academics, and researchers.

Keywords: *Clean Architecture, Developer Productivity, Empirical Study, Maintainability, Software Quality*