

## Optimizing API and framework design for improved usability, maintainability, and developer productivity

Hashan G.T. <sup>1\*</sup>, Somaweera W.T.S. <sup>2</sup> and Sandaruwan R.M.T. <sup>2</sup>

<sup>1</sup>Department of Software Engineering, Faculty of Computing,  
Sabaragamuwa University of Sri Lanka, Sri Lanka

<sup>2</sup>Department of Computing and Information Systems, Faculty of Computing,  
Sabaragamuwa University of Sri Lanka, Sri Lanka

\*gthashan@std.appsc.sab.ac.lk

The use of APIs/ Frameworks in software development today cannot be done without and yet numerous of them still have design defects that hinder usability, maintainability and developer productivity due to lack of systematic and actionable guidelines to respond to bad design, documentation and tool support. This paper defines the most frequent API errors, documentation efficacy, IDE tool influence which was empirically checked based on survey data representing 100 software developers in Sri Lanka. The analysis of error patterns shows that the three most frequent error categories are caused by inconsistent naming conventions (40.2%), bad or vague error messages (41.2%), and absence of real-life examples (35.1%), which proves that a lot of mistakes are indeed caused by the flaws in the design, but not the misunderstanding of the developers. Regarding the quality of the API/Framework documentation 47.4% of the respondents rated it at only 3 out of 5 and only 4.1% rated it excellent, which shows that the gap between what developers require and what is available is very wide. Additionally, this paper has also determined that over 70 percent of the respondents think that intelligent autocompletion, inline documentation, and real-time detection of errors in their IDE are some features that have a major impact on their productivity. This piece incorporates the information concerning different design principles that offer a concise API toward addressing the typical software design issues. It serves as a contribution to the field, bringing together fragmented thoughts, suggesting what is missing in current practices. Altogether, this paper indicates that the improved API and framework design may help to decrease the intellectual load on the developers, decrease technical debt, and enhance the efficiency and sustainability of the software development process.

**Keywords:** *API Design, Software Frameworks, Usability, Maintainability, Developer Productivity*