A Survey on Method Naming Standards: Questions and Responses Artifact

Reem S. Alsuhaibani
Computer Science
Kent State University
Prince Sultan University
Riyadh, Saudi Arabia
ralsuhai@kent.edu

Christian D. Newman
Software Engineering
Rochester Institute of Technology
New York, USA
cnewman@se.rit.edu

Michael J. Decker
Computer Science
Bowling Green State University
Ohio, USA
mdecke@bgsu.edu

Michael L. Collard
Computer Science
The University of Akron
Ohio, USA
collard@uakron.edu

Jonathan I. Maletic
Computer Science
Kent State University
Ohio, USA
jmaletic@kent.edu

***Abstract*— The artifacts of a large (+1100 responses) survey of professional software developers concerning standards for naming source code methods is presented. The artifact consists of the survey questions along with all the responses from participants. The artifact allows other researchers to examine and study the responses to the survey.**

Keywords—identifier naming standards, identifier naming conventions, method naming

I. Introduction

What characteristics of an identifier determine whether it is high or low quality? This question lies at the core of the data we present as part of an artifact. There are a large number of naming practices that the research community has derived and suggested to developers through research publications. This advice, put together, forms the method naming standards that are based on the current available literature. To determine if the standards documented in software engineering literature reflect actual practice and align with developer opinion, we developed a survey study [1]. The goal of the survey is to collect professional developers' feedback to assess the standard's quality and practicality. This data helps support research that seeks to recommend and create stronger identifier names that fit developer expectations in terms of both semantics and structure.

Overall, the survey results found that developer opinions are very much in agreement with the set of standards. From the written responses we gleaned that developers are supportive of clearly articulating method naming standards and feel it has a positive impact to code comprehension.

The artifact we provide consists of the survey questions used in the study. Along with this all the 1604 responses (complete and incomplete responses) are provided (as a CSV file). Responses were collected from March 2020 to June 2020. The artifact is licensed under Creative Commons, making it readily available for researchers to use. Each part of the artifact is discussed in more detail in the next sections. Additionally, the location of the artifact is given.

II. Survey Questions

The survey was conducted using Qualtrics. The artifact includes a pdf file containing the survey questions used in Qualtrics for surveying professional developers on ten method naming standards. The pdf file contains three sections:

- The survey introduction describing the aim of the survey.
- The Likert scale questions including the source code examples used in the survey.
- The demographic questions asking the participants about their adherence to coding standards at their workplaces, programming languages they are familiar with, and their years of programming experience.

The survey questions concerning the method naming standards addressed 10 aspects (see Table 1).

TABLE I. Method naming standard. Each part of the standard and the associated survey question.

#	Standard Name	Survey Question
1	**Naming Style**	The method name should use a standard naming style such as camelCase or underscore_case; Camel case uses upper case letters for each word. Underscore case uses "_" to separate words.
2	**Grammatical Structure**	The method names with multiple words should be in a grammatically correct sentence structure.
3	**Verb Phrase**	The method name should always contain a verb(s) or verb phrase that refers to the behavior of the method.
4	**Dictionary Terms**	Developers should use only natural language dictionary words and/or familiar/domain-relevant terms.
5	**Full Words**	The method name should use full words rather than a single letter to clearly indicate the task of the method.
6	**Idioms and Slang**	The method name should not contain personal expressions, idioms, or unknown slang.
7	**Abbreviations**	The method name should contain only known or standard (i.e., recognized by others within the company) abbreviated terms. A poor abbreviation is one that has multiple possible expansions, interpretations, or is not typically used within the system domain.
8	**Acronyms**	The method name should contain only known or standard (i.e., recognized by others in the company) acronyms. A poor acronym is one that has multiple possible expansions, interpretations, or is not typically used within the system domain.
9	**Prefix/Suffix**	The method name should not contain a prefix/suffix that is a term from the system. This standard does not apply to languages such as C that do not have namespaces.
10	**Length**	The maximum number of words in a name should not be greater than (slider provided from 0-15)

III. Survey Responses

The artifact also includes a Comma Separated Values (CSV) file containing the responses. The responses are anonymous and do not contain any information that can identify an individual.

The vast majority of responses are from professional developers. However, there are approximately 30 responses from individuals with less than 3 years of experience. The vast majority of responses (~1050) are from individuals with 5 or more years of experience (see figure 1).

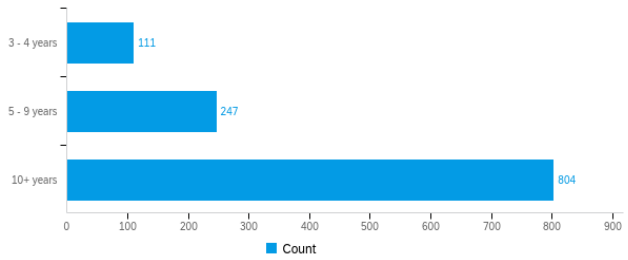

Fig. 1. Total participants according to years of experience

The responses to the questions use a Likert scale except for the question concerning the number of words in a name. This uses a scale between 1 and 15. Additionally, each question has an open feedback response option. These responses are also provided about each method naming standard. The file also includes developers' responses to the different demographic questions.

IV. Using the Artifact

The responses artifact contains the questions in columns, each of the standards question is followed by additional comment column (Text). This is followed by the demographic questions. For the question that ask developers about their programming language background, we note that participants were allowed to choose multiple languages using checkboxes. Answers to this question are separated by commas. All Likert scale answers contains one of the following four Likert scales: *strongly agree, agree, disagree, and strongly disagree*. The only numerical answers are the one provided to the question that asks about the preferred size for a method name.

The artifact can be used in multiple ways, for example it can be used to validate statistical correlations between different developers' demographic answers and a particular standard. It can also be used to find the preferred naming style by a particular group of developers and that is through their provided feedback. Also, one can find out more about what developers think about unit testing method names, and how this can be used to further support research on naming unit testing methods.

V. Obtaining Artifact

The artifact is available through GitHub via the following link:

https://github.com/KSU-SDML/Method-Naming-Standards-Artifact

The dataset is also available on Mendeley Data Archive Repository at [2] for DOI citation reference.

References

[1] R. S. Alsuhaibani, C. D. Newman, M. J. Decker, M. L. Collard, and J. I. Maletic, "On the Naming of Methods: A Survey of Professional Developers," in *2021 43th International Conference on Software Engineering (ICSE)*, Madrid, 2021, 13 pages.

[2] R. S. Alsuhaibani, C. D. Newman, M. J. Decker, M. L. Collard, and J. I. Maletic, "Method-Naming-Standards-Survey-Dataset," Mendeley Data, 2021, vol. 1. doi:10.17632/5d7vx88sph.1