Toward a Systematic Evaluation of Usability in RESTful APIs: A Metric-Based Approach

Ariel Machini[1]

[1]Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)

amachini@conicet.gov.ar

Abstract. *In recent years, web APIs have become a crucial part of modern software development, as they allow easy access to a wide variety of resources and services. This has led to a competitive market, "the API economy", and as a consequence, usability now plays a key role in the success of web APIs. Despite that, usability is often overlooked because there is no standard way to measure the usability of a web API. This work aims to address this problem by proposing a comprehensive model that enables systematic assessment of web API usability from various perspectives.*

1. Introduction

Application Programming Interfaces (APIs) are software-to-software interfaces that enable communication between applications without user interaction. Depending on the context, the term "API" might have different interpretations, as there exist APIs for operating systems, applications, and the web. However, in recent years, this term has been most often associated with web APIs built using Representational State Transfer (REST) technologies. Hence, the terms "API", "web API", "REST API" and "RESTful API" can be considered synonymous [De 2017].

Over the past decade, web APIs have increased in popularity as a result of the ease with which they allow access to resources and services from (and among) different organizations, becoming one of the pillars of modern software development [Raemaekers et al. 2012]. This has generated a highly competitive market, "the API economy", where APIs enable monetization, collaboration and integration across different platforms and industries [Tan et al. 2016]. In such a competitive ecosystem, usability becomes a key factor in determining the value of APIs [Stylos and Myers 2007], yet poorly designed APIs abound [Myers and Stylos 2016]. An API with low usability can lead to developer inefficiency, adoption failure and, in some cases, to unwanted monetary costs.

Even though there is research proposing methods and models for analyzing, improving and assessing API usability, it was found that (1) it is mostly centered around offline or "traditional" APIs, and metrics that apply to offline APIs do not always translate well to web APIs [Machini and Casas 2024c], since they are different in architecture, as most modern APIs use the REST architecture [De 2017]; (2) research geared towards web API usability has a predominant focus on documentation; and (3) proposed metrics are not explicitly linked to any usability attributes, or only to a few [Mosqueira-Rey et al. 2018]. These limitations evidence that, although usability is important for the success of web APIs, it is a topic that needs to be researched further. The aim of the research detailed in this paper is to build a comprehensive model for assessing web API usability, based

on an extended Goal-Question-Metric approach, and a complementary tool to automate assessment where possible, leveraging the structure of the OpenAPI specification.

This paper is structured as follows: Section 2 lists the proposed research questions and goals; Section 3 outlines the methods and strategies applied throughout the investigative process; Section 4 briefly describes related work; Section 5 presents all results obtained to date; and Section 6 provides some final remarks and explores future work.

2. Research questions and goals

The main goal of this research is to build a comprehensive model that allows developers to easily assess the usability of their APIs. In the following section, the list of research questions that motivate this study is presented, as well as the set of goals derived from these questions.

- **RQ1:** How can the usability of a web API be analyzed and improved in a comprehensive and systematic manner? **Justification:** A preliminary review of the literature revealed that (1) there is neither a consensus nor a standard method for measuring web API usability, (2) proposed metrics are weakly related to usability attributes, and (3) there are no models that integrate usability metrics for web APIs.
- **RQ2:** What usability aspects or dimensions of a web API can be measured, and how? **Justification:** In order to correctly measure the usability of a web API, a thorough analysis of the distinct elements that comprise this kind of APIs is first needed to be able to understand their individual influence over usability.
- **RQ3:** Which usability metrics can be derived from the aforementioned aspects to analyze and evaluate the usability of a web API? **Justification:** In order to comprehensively assess the level of usability of a web API, it is necessary to have metrics that help quantify usability from multiple perspectives.

Based on the main goal and these questions, a set of more specific goals was defined: (1) To identify, study and integrate methods and heuristics for API usability analysis and evaluation; (2) To formulate a theoretical and conceptual framework, using the GQM approach, to analyze and assess web API usability; and (3) To develop a tool that provides automation support for the proposed framework.

3. Methods

As a general methodology, Design Science Research (DSR) [Peffers et al. 2007] was used. This approach, commonly employed in the fields of engineering and computer science, is used to build artifacts that deliver useful and effective solutions to a problem within a specific domain. Artifacts designed using DSR include the usability model and a complementary catalog of metrics. The following list presents methods used during the different research activities performed thus far:

- The systematic mapping study was performed following the guidelines by [Petersen et al. 2015] for conducting this type of studies in the field of software engineering. Additionally, a search strategy based upon Population, Intervention, Comparison and Outcomes (PICO) [Kitchenham and Charters 2007] was implemented, which allowed developing useful keywords and search strings

from the research questions; categorization of empirical validation proposed by Shaw [Shaw 2003] was utilized to determine if papers met quality criteria; and a backward and forward snowball search was performed to expand the number of results.

- As previously mentioned, the usability model was designed following the DSR approach, but the multi-level Goal-Question-Metric (GQM) approach was also leveraged to structurally define the model[1]. To develop the preliminary version of the model, the following steps were performed: (a) Collection of an initial set of API usability metrics through a systematic mapping study; (b) Mapping of usability metrics for offline APIs to metrics for web APIs (when possible); (c) Search of additional metrics in grey literature published by recognized organizations in the web API industry; (d) Grouping of metrics by usability attribute(s); (e) Definition of Goals and Questions; and (f) Distribution of metrics among the defined Questions.
- The first, exploratory [Wohlin et al. 2012] survey was performed with a non-probabilistic sampling scheme (convenience sampling) [Kasunic 2005, Fowler 2013], mainly due to time constraints. To find interested individuals with the required profile (web API consumers and/or developers), the survey was shared on websites and forums related to software development or web APIs, on social media, and to email addresses extracted from the *Cámara de la Industria Argentina de Software* (CESSI)[2]. All questions related to the influence of factors on web API usability were close-ended and used a 1 to 5 Likert rating scale. Descriptive statistics such as mean, median, and standard deviation were used to obtain basic information from the collected responses, and Chi-Square tests of independence (or Fisher's exact tests where Chi-Square tests were not applicable) were conducted to study possible links between variables of interest. This survey was open from the end of September 2023 to December 1 of the same year.
- After updating the usability model using the feedback from the first survey, a second, more complex survey, in which the usability model as a whole was the object of analysis, was carried out. In this survey, all of the respondents were experts in the field of web APIs, given that the expert judgment method was used. This approach consists of collecting opinions from individuals with a well-known trajectory in the topic being investigated [Escobar-Pérez and Martínez 2008]), and the selection process was performed following the suggestions of Skjong and Wentworth [Skjong and Wentworth 2000]. For the analysis of the obtained responses, Gwet's AC1 coefficient [Gwet 2008] was applied to calculate inter-rater reliability, and common statistics were used to summarize the information.
- In addition to updating the usability model, with the obtained expert feedback a catalog of web API usability metrics categorized by key REST API components was developed. The aforementioned artifact was designed following the DSR approach, and the "key" REST API components were extracted from reliable sources.

[1]For more details, please refer to the appendix available at https://github.com/arielmachini/CIbSE-26-Appendix

[2]https://cessi.org.ar/socios

4. Related work

Referencing the findings of the systematic mapping study (described in more detail in Section 3 and 5.1) performed as part of this research, from the 18 articles of interest identified, 86 out of the 96 metrics encountered (90 %) were related to usability, being the top 3 usability attributes *learnability* (40 % of the usability-related metrics), *knowability* (31 %), and *efficiency* (17 %). This indicates a clear interest in this particular software quality attribute; however, only 10 of the 18 articles identified propose metrics related to APIs, and only two [Koçi et al. 2020, Ma et al. 2016] of such articles propose metrics related to web APIs.

Starting from the premise that the success of an API depends on the available documentation and its ease of use, [Koçi et al. 2020] present a data-driven approach to measure web API usability. Through a review of the state of the art, they identified a set of usability attributes, defined indicators—which were quantified using API logs—for web API developers to follow, and metrics for the aforementioned indicators. These metrics were used as features to build a classifier model to predict the error rate of API endpoints. The proposed approach was validated with a case study. On the other hand, in their article, [Ma et al. 2016] propose a methodology based on genetic algorithms (ε-Pareto) for the selection and combination of different web APIs based on various QoS indicators, which include response time, availability, price, and rating. To validate the efficacy of their approach, the authors performed quantitative experimental evaluations. Although [Koçi et al. 2020] link the metrics they introduce to usability attributes (to *knowability* to be exact), neither article situates their metrics within a model. There are some studies [Janes et al. 2014, Mosqueira-Rey et al. 2018, Yamamoto et al. 2018] that propose models or frameworks for APIs; nevertheless, these models do not apply to web APIs, do not focus on usability, or address a limited set of usability attributes.

[Janes et al. 2014] present an initial version of a measurement framework for web APIs. This framework, based on API call responses, was designed to contemplate, among other attributes, structural quality, learning difficulty, design consistency, and backward compatibility. It analyzes the tree-based hierarchical structure of JSON and XML data returned from API calls. A set of easy-to-compute metrics—with sample usage scenarios—is also proposed as a starting point. These metrics were evaluated in a case study with popular open web APIs, such as the ones exposed by PayPal, Twitter, and Wikipedia. [Mosqueira-Rey et al. 2018] introduce a set of guidelines and heuristics for API usability in the context of a usability model. The metrics proposed were extracted from the literature available at the time, including 27 articles and textbooks from 1976 to 2016. The authors concluded that the literature on API usability was very technically-minded and tended to neglect the subjective component of usability. [Yamamoto et al. 2018] propose a quality model that "reflects [the] unique characteristics" of web APIs and focuses on their learnability and stability to change, from the perspective of web API users. Based on their model, they also propose a set of measures and a quantitative evaluation method. To validate these artifacts, the authors applied the proposed quality model and evaluation model to real web APIs, and also performed an empirical study of the usability of web APIs.

5. Results

In this section, all major results obtained to date are presented. Methodological details will be skipped, as they were outlined in Section 3.

5.1. Systematic mapping study

Initially, a systematic mapping study was carried out [Machini and Casas 2024c] with the aim of identifying and categorizing metrics related to web API usability. To do so, four research questions were defined, which are related to (a) quality metrics and properties they measure, (b) usability metrics and attributes they contemplate, (c) usability models, and (d) automation level; From these, a set of search terms was derived to find papers that respond to the aforementioned questions. Additionally, two Python scripts were developed[1] to facilitate the search process in the Springer database. A total of 18 articles in the period of 2000 to 2021 were found, using both well-known academic databases and the snowball search method[1], and a set of 96 metrics was identified, from which only 14 % were explicitly designed for web APIs.

5.2. Survey #1 - With API consumers and developers

After the development of a preliminary version of the usability model, a survey which targeted API consumers and developers was carried out [Machini and Casas 2024b] with the purpose of validating the relevance of the collected metrics for web API usability. The survey was designed to answer three research questions related to the influence level of metrics on web API usability, the utility of specifications such as OpenAPI, and the effect of differences in respondents' expertise on their opinions. A total of 36 developers, most of whom indicated having a high level of expertise developing software (63.9 % of respondents) and with web APIs (58.4 % of respondents)[3], participated in the survey, and results suggest that (a) the documentation of web API components has the greatest influence on usability, (b) the number of consecutive parameters of the same type is the factor with least influence on usability, and (c) opinions from experts do not differ significantly from those of other survey participants. The obtained feedback was used to modify the usability model to expand the quantitative level of GQM, adding weights to the metrics. This was done so metrics perceived to have a stronger influence on usability would have a greater impact on the usability score of a web API obtained through the application of the model.

5.3. Survey #2 - With web API experts

After updating the quantitative level of the usability model with additional metrics and weights, a second survey was performed, this time with a group of recognized experts from the web API industry. In this survey, experts had to review all levels of the usability model, mainly the quantitative level (i.e., metrics), and provide their opinions and (optionally) suggestions. Statistical analysis of the responses showed substantial[1] agreement (AC1 = 0.67) among the respondents regarding the relevance of the model for web API usability, with their agreement leaning towards approval of the model. Additionally, metric weights were updated, and seven metrics that were not considered meaningful (by the experts) for web API usability were removed. At the time of writing this article, the paper that explores this survey has not been published yet and is currently under peer review.

[3]The complete set of responses can be accessed at http://hdl.handle.net/11336/266545

5.4. Usability model

The Goal-Question-Metric usability model is the main result of this research, and it has been revised several times throughout the process to adapt to suggestions received in conferences where the model was presented, from web API users, and from experts in the web API industry. Initially, the quantitative level of the model had 32 metrics without weights, and over time, weights were added and the number of usability metrics increased to 52. The latest version of the model, released after the survey with experts (Section 5.3), has six goals (conceptual level), eight questions (operational level), and 45 metrics with updated weights[1]. It is worth noting that the number of goals and questions (which respond to different usability attributes) remained constant in all versions of the model.

5.5. Catalog of metrics

To take full advantage of the feedback provided by the experts who participated in the second survey (Section 5.3), in addition to updating the usability model, a catalog of metrics for web APIs was developed. This catalog leverages the quantitative level of the GQM model, and instead of grouping metrics by goals and questions based on usability attributes, groups them by key REST API components. In order to define what the "key" components of a REST API are, an article [The Postman Team 2023] written by the Postman Team—a leading organization in the web API industry—was used as a reference. Furthermore, documentation was also defined as a key component of a REST API, in line with the findings of the systematic mapping study (Section 5.1). Thus, usability metrics in the catalog were grouped by: API Documentation, API Request, API Server, API Response, and General (for metrics that are important but did not fit in any of the previous categories). At the time of writing this article, the paper that highlights this artifact has not been published yet and is currently under peer review.

5.6. Key aspects of the proposed solution

The quality of web APIs is an issue that has gained relevance in recent years due to the widespread use of REST APIs and their integration in enterprise information systems [Yamamoto et al. 2018]. Even though there have been quality models proposed for web APIs, there is a lack of usability models: Usability is one of the most important software quality attributes [Nielsen 1992], and for web APIs, it can be a critical factor for adoption [Stylos and Myers 2007]. The solution proposed in this work provides several artifacts to help web API developers assess the usability of their APIs from different perspectives (namely[4] *learnability*, *understandability*, *knowability*, *efficiency*, *memorability*, *errors*, *robustness*, and *satisfaction*), thus enabling analysis in a comprehensive and organized manner. The usefulness of these artifacts, mainly the usability model and the catalog of metrics, has been proven by the positive feedback obtained in the surveys (Sections 5.2 and 5.3).

6. Conclusions and future work

As a result of the ease with which web APIs allow access to digital resources, their widespread adoption across different industries, and the prevalence of poorly designed APIs [Myers and Stylos 2016], the need for a comprehensive usability model has become critical. Even though web API usability as a research topic has gained more relevance in

[4]These attributes and their corresponding sources are detailed in [Machini and Casas 2024a].

academia over the last few years, it still needs more exploration, especially in the area of assessment models, as they allow for an organized measurement workflow.

The study outlined in this paper proposes a hierarchical model based on the GQM approach, comprising six goals, eight questions, and 45 metrics, as well as a catalog of usability metrics organized by REST API components. Both of these artifacts can be used to assess web API usability from different points of view: The model, which groups metrics within several questions and goals that respond to multiple usability attributes, allows for a deeper understanding of the different usability goals that an API should seek to satisfy and, on the other hand, the catalog provides a straightforward way for developers to access usability metrics, which can be especially valuable for those who think of APIs in modular terms. Validation results have shown that the developed artifacts are perceived as relevant and have the potential to positively contribute to web API usability.

As for the future work, the intention is to (1) finish building the tool that will help developers apply the model to their APIs, automating assessment when possible[1] (this is one of the goals mentioned in Section 2); (2) develop an evaluation approach to facilitate interpretation of the results from applying the model; (3) further validate the proposed model through its application to a set of real-world APIs; and (4) incorporate a new dimension to the usability model, following the Goal-Question-Indicator-Metric (GQIM) approach, so it is easier to understand and, therefore, easier to apply.

Referências

De, B. (2017). *API Management*. Apress Berkeley, CA.

Escobar-Pérez, J. and Martínez, A. (2008). Validez de contenido y juicio de expertos: Una aproximación a su utilización. *Avances en Medición*, 6:27–36.

Fowler, F. (2013). *Survey Research Methods*. SAGE Publications.

Gwet, K. L. (2008). Computing inter-rater reliability and its variance in the presence of high agreement. *British Journal of Mathematical and Statistical Psychology*, 61(1):29–48.

Janes, A., Remencius, T., Sillitti, A., and Succi, G. (2014). Towards Understanding of Structural Attributes of Web APIs Using Metrics Based on API Call Responses. In *Open Source Software: Mobile Open Source Technologies*, pages 83–92. Springer Berlin Heidelberg.

Kasunic, M. (2005). Designing an Effective Survey. Technical report, Carnegie Mellon University.

Kitchenham, B. and Charters, S. (2007). Guidelines for performing Systematic Literature Reviews in Software Engineering. Technical report, Keele University.

Koçi, R., Franch, X., Jovanovic, P., and Abelló, A. (2020). A Data-Driven Approach to Measure the Usability of Web APIs. In *2020 46th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, pages 64–71.

Ma, S.-P., Lan, C.-W., Ho, C.-T., and Ye, J.-H. (2016). QoS-Aware Selection of Web APIs Based on ε-Pareto Genetic Algorithm. In *2016 International Computer Symposium (ICS)*, pages 595–600.

Machini, A. and Casas, S. (2024a). A preliminary GQM model to evaluate web API usability. *Memorias de las JAIIO*, 10(2):1–13.

Machini, A. and Casas, S. (2024b). An empirical study on web API usability: The consumer-developer perspective. *Brazilian Journal of Technology*, 7(4).

Machini, A. and Casas, S. (2024c). Web API Usability: A Systematic Mapping Study. *Interacting with Computers*.

Mosqueira-Rey, E., Alonso-Ríos, D., Moret-Bonillo, V., Fernández-Varela, I., and Álvarez Estévez, D. (2018). A systematic approach to API usability: Taxonomy-derived criteria and a case study. *Information and Software Technology*, 97:46–63.

Myers, B. A. and Stylos, J. (2016). Improving API usability. *Communications of the ACM*, 59(6):62–69.

Nielsen, J. (1992). The usability engineering life cycle. *Computer*, 25(3):12–22.

Peffers, K., Tuunanen, T., Rothenberger, M., and Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems*, 24:45–77.

Petersen, K., Vakkalanka, S., and Kuzniarz, L. (2015). Guidelines for conducting systematic mapping studies in software engineering: An update. *Information and Software Technology*, 64:1–18.

Raemaekers, S., van Deursen, A., and Visser, J. (2012). Measuring software library stability through historical version analysis. In *2012 28th IEEE International Conference on Software Maintenance (ICSM)*, pages 378–387.

Shaw, M. (2003). Writing good software engineering research papers. In *25th International Conference on Software Engineering, 2003. Proceedings.*, pages 726–736.

Skjong, R. and Wentworth, B. (2000). Expert Judgement and risk perception. In *Offshore and Polar Engineering Conference (ISOPE)*, pages 537–544.

Stylos, J. and Myers, B. (2007). Mapping the Space of API Design Decisions. In *IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC 2007)*, pages 50–60.

Tan, W., Fan, Y., Ghoneim, A., Hossain, M. A., and Dustdar, S. (2016). From the Service-Oriented Architecture to the Web API Economy. *IEEE Internet Computing*, 20(4):64–68.

The Postman Team (2023). What are the components of an API? Available at: https://blog.postman.com/what-are-the-components-of-an-api.

Wohlin, C., Runeson, P., Höst, M., Ohlsson, M., Regnell, B., and Wesslén, A. (2012). *Experimentation in Software Engineering*. Springer Berlin, Heidelberg.

Yamamoto, R., Ohashi, K., Fukuyori, M., Kimura, K., Sekiguchi, A., Umekawa, R., Uehara, T., and Aoyama, M. (2018). A Quality Model and Its Quantitative Evaluation Method for Web APIs. In *2018 25th Asia-Pacific Software Engineering Conference (APSEC)*, pages 598–607.