

Why API documentation is insufficient for developers: an empirical study

Qiang FAN*, Yue YU*, Tao WANG, Gang YIN & Huaimin WANG

*Key Laboratory of Parallel and Distributed Computing, College of Computer,
National University of Defense Technology, Changsha 410073, China*

Received 3 January 2019/Revised 27 February 2019/Accepted 21 March 2019/Published online 30 October 2020

Citation Fan Q, Yu Y, Wang T, et al. Why API documentation is insufficient for developers: an empirical study. *Sci China Inf Sci*, 2021, 64(1): 119102, <https://doi.org/10.1007/s11432-019-9880-8>

Dear editor,

Application programming interface (API) documentation plays an important role in software development and reuse [1] for both API maintainers and API users. Well-written documentation helps developers understand and reuse codes effectively [2] and focus their time on desired interfaces and functions instead of the entire system [3]. Most high-quality open source projects maintain complete and informative official documentation. API documentation typically conveys detailed specifications, such as class/interface hierarchies and method descriptions, which can be of great benefit to developers [4]. However, despite its authoritative-ness and thoroughness, single-sourced official documentation does not always meet the developers' requirements [5]. As of January 2018, over 1 million posts about Android had been posted to StackOverflow and approximately 16% of these posts were related to its API (statistics found by searching for API and Android in StackOverflow). The lack of important information in API documentation makes developers spend large amounts of time searching for information and seeking help. During this time, developers obtain large amounts of unconfirmed information, which can affect the quality of projects and the developers' understanding of the API.

This research presents an empirical study concerning insufficient API documentation based on a hybrid approach, including a manual inspection and an online survey. Official documentation sources tend to be more authoritative and rigorous, which can effectively reduce bugs, standardize programming, and spread the APIs to crowd. API documentation plays an important role in the process of software development. We aim to determine what causes developers to seek help from other sources and abandon official API documentation. We believe that our results can be used to improve the quality of API documentation.

Herein, we explore crowd discussion concerning the Android API on StackOverflow, compare them with the official documentation, and identify what questions have troubled the third-party developers and why they are confused de-

spite the official documentation. Additionally, we conduct an extensive online survey of the developers who are involved in the discussion to understand their difficulties, and to verify our findings. Finally, an empirical study of 1000 posts of StackOverflow and 319 questionnaire responses is conducted.

API documentation overview. API documentation is deliverable technical content, containing details about the functions, classes, return types and arguments. Most of the time, the official API documentation is maintained by the core team or active members of the project. Easy-to-follow documentation is always well edited with a good structure. Under ideal conditions, good API documentation provides concise and basic information about the API, which can communicate how to effectively use and integrate an API with API users. However, the different perspectives of API developers and API users make this a challenge. Differences in shared keywords and technical backgrounds make it difficult for API users to use the API, which increases its cost. This also makes it difficult for API developers to determine the requirements of the API users.

Facing insufficient API documentation, API users often access Q&A websites such as StackOverflow to seek help from other developers. They post their question or problem on Q&A websites, and other developers who have solved the problem provide them with help and answer the post. As time goes by, these websites accumulate massive amounts of discussion concerning the API; this is seen as a type of crowd documentation [5, 6]. In comparison with official API documentation, the crowd documentation is more dynamic and interactive. API users can search archived discussion to find solutions, or post a new question to solicit a solution from other developers. The advantages of crowd documentation are that API users can find new solutions using other developers' experiences, and the question-oriented approach can allow API users to locate information quickly.

The flexibility and question-oriented approach of crowd documentation result in more and more API users accessing to find solutions; however, there are still some limitations to

* Corresponding author (email: fanqiang09@nudt.edu.cn, yuyue@nudt.edu.cn)

this approach. For example, the answers from crowd developers are uneven and may not be well edited or proofread. Accordingly, it is difficult to guarantee the correctness of the information provided by crowds. The process of filtering out the right answer can also be time consuming and upsetting. In addition, crowd developers need time to access the question and organize the language necessary to answer it. Some questions may not get attention, with few developers answering or responding to it. Factors such as these may result in API users not receiving help in a timely manner. Therefore, well-edited API documentation plays an irreplaceable role in providing reliable information for swift inquiries. It is necessary to identify insufficient API documentation and try to improve it.

Approach. In this study, we propose a hybrid approach to explore the requirements of third-party developers concerning API documentation. This approach includes a manual inspection and an online survey. An overview of our approach is shown in Figure 1.

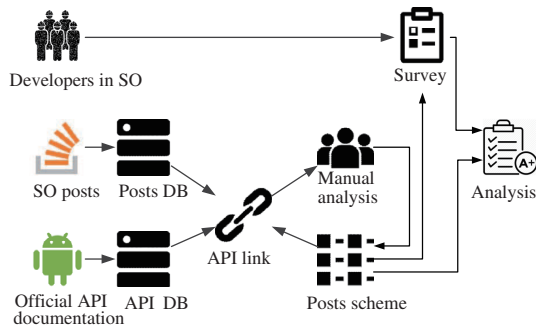


Figure 1 (Color online) The hybrid approach process.

API link. In this process, we link Android API and Android-related posts to filter out posts discussing Android API. To extract posts discussing the Android API, we use a process adopted from [7]. In our case, a link is a connection between a StackOverflow question and an Android class (or method). Using our method, we filter out 54927 StackOverflow posts that link to at least one Android API class or method.

Manual inspection. We use manual inspection to categorize the topics and causes of questions based on the posts we select in the process of linking the API. Additionally, we identify the information provided in the official documentation, which is used to compare different perspectives of API developers and users. We randomly select 100 posts associated with Android API and inspect the questions and answers in the posts to determine what is lacking in the API documentation that result in the question. Via a rigorous analysis of the existing literature and our own experience and analysis, we identify ten causes of questions. Next, ten master's students repeat the previous process from new selected questions to examine the stability of the taxonomic definition. Each selected post is assigned to two students. We use our stabilized taxonomy to identify 1000 Android-related questions.

Online survey. We use the results of the manual inspection to design a research survey and deliver it to StackOverflow users. We release the questionnaires via SurveyMonkey, one of the most famous web services currently available for online surveys. The questionnaires contain questions such as "What causes you to seek help in StackOverflow instead of

using the official documents?" We send the survey questionnaires to (1) users who have asked questions about Android API usage and (2) users who have answered questions about Android API usage. Within a period of 7 days, we receive a total of 319 responses.

Result. Our survey result find that, approximately 10% of API users never use the official documentation. The official documentation is insufficient to answer their questions. Our hybrid approach find that both the manual inspection and the online surveys present similar results. Below, we show the results of our approach.

Hierarchical vs. flat. Searching for API is one of the main requirements of API users. Approximately 20% of posts according to the manual inspection and 30% of the third-party developers in the survey indicate that searching for API is important. Therefore well-organized information is important for developers to quickly locate relevant material of API. Official API documentation is generally edited into a hierarchical structure. Related APIs are categorized into one Class or Package, and API documentation often uses the same structure to organize information. Q&A websites such as StackOverflow have a more flat organization. Users use posts to seek help and share information. Websites use tags and search engine to manage and locate information. In our results, over 30% of posts and nearly 30% of API users indicated that locating the information they require in the documentation is difficult. The crowd activities in StackOverflow provide more information, which plays an important role in helping locate key information. Therefore, more developers prefer searching for information in StackOverflow.

Functional explanation vs. usage example. For most API documentation, the API description primarily consists of a functional interpretation. Such information tells developers what the API can do; however, such documentation rarely tells developers how to use the API. From our result, approximately 50% of posts according to the manual inspection and 50% of third-party developers from the survey indicated a lack of examples in the API documentation. In addition, the official documentation often fails to fully meet the developers' requirements. Sometimes, developers cannot find what they want in the documentation and they need to seek help from other developers to find the information they want and to better understand the API. In our result, over 30% of posts according to the manual inspection and over 30% of third-party developers from the survey indicated the incompleteness of the documentation. Over 30% of posts and nearly 20% of third-party developers indicated that the lack of precautions causes incorrect API usage. The lack of precautions costs API users time because they need to fix the incorrect usages in their programs.

Official statements vs. free discussion. The official documentation is edited by the API maintainers using accurate and concise expressions. The discussion on StackOverflow has more freedom, and the content in the posts is more social and interactive. Results show that nearly 20% of posts and 30% of third-party developers indicate that sometimes the statements in the official documentation are difficult to understand. In comparison with StackOverflow, the official documentation may sometimes be incomprehensible and of poor readability.

Acknowledgements This work was supported by National Grand R&D Plan (Grant No. 2018-YFB1004202) and National Natural Science Foundation of China (Grant No. 61702534).

References

- 1 Gentleman R C, Carey V J, Bates D M, et al. Bioconductor: open software development for computational biology and bioinformatics. *Genome Biol*, 2004, 5: R80
- 2 Forward A, Lethbridge T C. The relevance of software documentation, tools and technologies: a survey. In: Proceedings of the 2002 ACM Symposium on Document Engineering. New York: ACM, 2002. 26–33
- 3 Roehm T, Tiarks R, Koschke R, et al. How do professional developers comprehend software? In: Proceedings of the 34th International Conference on Software Engineering. New York: IEEE Press, 2012. 255–265
- 4 Dekel U, Herbsleb J D. Improving api documentation usability with knowledge pushing. In: Proceedings of the 31st International Conference on Software Engineering. Washington: IEEE Computer Society, 2009. 320–330
- 5 Chen C, Zhang K. Who asked what: integrating crowd-sourced faqs into api documentation. In: Proceedings of the 36th International Conference on Software Engineering. New York: ACM, 2014. 456–459
- 6 Hartmann B, Dhillon M, Chan M K. Hypersource: bridging the gap between source and code-related web sites. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. New York: ACM, 2011. 2207–2210
- 7 Linares-Vasquez M, Bavota G, Di Penta M, et al. How do api changes trigger stack overflow discussions? a study on the android sdk. In: Proceedings of the 22nd International Conference on Program Comprehension. New York: ACM, 2014. 83–94