

Analyzing the Importance of JabRef Features from the User Perspective

Martin K. Simon¹, Linus W. Dietz¹, Tobias Diez², and Oliver Kopp³

¹Technical University of Munich,

²Max Planck Institute for Mathematics in the Sciences Leipzig,

³University of Stuttgart

`martin.simon@tum.de`, `linus.dietz@tum.de`,

`Tobias.Diez@mis.mpg.de`, `kopp@informatik.uni-stuttgart.de`

Abstract. JabRef is a reference management system mostly used by \LaTeX users to organize their bibliographic references. As it is often the case in free and open source software, the usability of this program has not yet been analyzed systematically. In this paper, we report on the first application of user-centered design methods in this project. To identify the important features of JabRef, we first analyzed the telemetry data and then conducted an online survey with 124 participants. The analysis of the responses shows that the features naturally fall into groups based on the feature's perceived importance and the user's awareness of it. Finally, we derive guidelines on how to improve JabRef's user interface.

1 Introduction

Reference management software is an essential element of the scientific tool set. It helps researchers manage their bibliography, organize their knowledge, and interact with external services such as digital libraries or recommender systems for related articles [2]. JabRef is a free and open source reference management software (RMS) written in Java, first released in 2002. It provides a graphical user interface to view and edit bibliographic references and stores them in the BibTeX or BibLaTeX format, both of which are natively supported by LaTeX. JabRef puts an emphasis on the quality of the references and the integration with external data sources to get an accurate and complete bibliography. Having a long history of being a free and open-source software (FOSS) under active development by a community of volunteers, it has stood the test of time and accumulated a multitude of features [11]. Since 2017, the development efforts mainly go into the development of a new user interface based on JavaFX¹ as the interface technology to replace the originally used Swing framework. This transition in technologies is seen as a chance to improve the graphical interface, which was one of the main criticisms raised in an online survey conducted in 2015 by the JabRef team [5]. It was chosen to introduce the user perspective to the development process through user-centered design [7]. This term describes

¹ <https://wiki.openjdk.java.net/display/OpenJFX>

not only a family of methods that can be used to incorporate user needs into the design of a software but also encompasses a process model for directly involving the user in software development [9].

In this paper, we apply user-centered design methods to improve the usability of JabRef based on user data. To identify the important features of this product, we first analyzed the telemetry data and then conducted an online survey. Based on the analysis of the responses, we derive guidelines on how to improve JabRef's user interface.

2 Foundations

One problem with open-source software (OSS) is the knowledge gap between the developers of the software and its end users [8]. While in its origin, OSS was mostly developed *by hackers for hackers*, its target group now also encompasses a large group of casual end users, who speak an entirely different language from the IT terminology and, thus, do not recognize technical terms routinely used by developers. Users often lack the technical knowledge to cope with the complexity of some programs, which is rarely addressed adequately in the development process [8].

Moreover, the special atmosphere in OSS projects often leads to a prioritization of functionality over usability. Adding functional features to a piece of software proves to be more rewarding for OSS contributors [3] and is therefore prioritized over improving the usability of existing software, as developers themselves confirm [1]. Furthermore, the decentralized manner of development requires much modular coding, which can threaten the consistency of a single program causing confusion for the user [12]. This is further fueled by the high frequency of potentially unstable releases of new versions, which are often preferred by software veterans due to their enhanced functionality but cause distress for novice users, who are not able to cope with possible system crashes.

The main reason named by maintainers of OSS projects for insufficient usability efforts made in their projects is the lack of financial resources [1]. As the work is mainly done on a voluntary basis, project leaders do not have the budget to hire outside professionals for support [3, 8]. One study showed that only 29% of examined projects hired experts from the outside to evaluate their usability, 21% used remote usability evaluations, and 8% hired a usability laboratory [1].

3 Telemetry Data of JabRef

According to JabRef's documentation, its features mainly fall into four groups: collection and import of new references, organizing these references, export of references to work collaboratively, and support the researcher in the process of writing articles by integration with text editors [6]. For adopting measures to improve the usability, we first need to identify which of these features are used most frequently by the users. In this way, we establish an understanding of the user behavior and needs.

Table 1. List of the ten most frequently called dialogs

Description	Frequency
create new entry	11618
import external file	4059
merge duplicates	1316
merge entry from web search	1282
add/edit groups	522
import entry from PDF metadata	500
merge edits of an externally modified entry	412
choose preferences	408
show web search results	126
define string constants	100

We used anonymous, telemetry data from users working with the latest versions of JabRef. The analyzed data was gathered from users that explicitly opted into the collection between June 2018 and September 2018. The data consisted of logged dialog calls in the JabRef code indicating an interaction of the user with the program by opening dialog boxes. In addition to the dialog name, the version of JabRef as well as the country and a timestamp of the call were logged. The latter two, however, were not used in the analysis of the data.

The sample consisted of 22,149 calls of 2,659 different users opening dialog boxes. We reduced it to 21,720 by removing calls triggered by unreleased JabRef development versions from further analysis. For the remaining sample, the frequencies of the different dialogs were calculated to serve as the first hint towards important JabRef features. The number of different dialogs found was 36, the ten most frequently called dialogs are described in Table 1. By far the most opened dialog was the one for creating a new bibliographic entry in the database, followed by the import dialog for new references. We should note that there is not a clear one-to-one relation between the dialogs and the complete feature set. For example, the dialog for adding a new empty entry also contains functionality to create an entry based on an article identifier such as a DOI. Moreover, the dialog calls unfortunately give only an incomplete image of the users' interaction with JabRef. At the time of the collection, no other clicks, such as in the menu or buttons in the toolbars, were recorded. Despite these shortcomings in the telemetry data, our analysis gives a good overview of the frequency of usage of most features.

4 Online Survey

Using the telemetry data and an expert assessment of JabRef's functionality, we constructed a list of 24 features, on which we based an online survey conducted by Simon [10]. The main part of the survey was a five-point Likert scale for each feature, where we asked the respondent to rate the importance of a feature for their work with JabRef. A sixth option was added to cover the possibility that

the respondent had not known the feature before. Furthermore, we asked the respondents about their general usage of JabRef for research and the level of their academic career. The survey was constructed with Google Forms and was advertised in university mailing lists and on the social media accounts of JabRef.

The survey was conducted over a span of three months, from October 2018 to December 2018. The final number of respondents to the survey was $N = 124$. Most respondents were PhD students or postdoctoral researchers, which accounted for 40% of the answers. Moreover, permanent researchers at a university accounted for 37% of responses. With 69%, the majority of the sample used LaTeX or LyX as their text processing software. 26% used a program other than those, the remaining 4% used a combination of multiple text processors.

The survey questions and the raw data of the answers have been published in anonymized form on Github².

4.1 Importance vs. Awareness of Features

The ratings of importance were transcoded into a numerical scale reaching from 0 (not important at all) to 4 (absolutely crucial) to make calculating mean values for the ratings possible. Additionally, we computed the relative frequency of the *do-not-know* answers for every feature. We will refer to the latter as the users' awareness of a feature.

The relationship between the mean importance and the awareness of a feature is presented in Fig. 1, which indicates a positive dependency between the two. By taking the means of the values in the two axis dimensions as reference points, more information can be gained from the figure. For instance, the number of Core Features and Underdeveloped Features are higher than of those in the other two quadrants, which supports the relationship between the two variables. Furthermore, splitting the data points into four groups as shown allows us to characterize the features based on their group.

Core Features The group in the upper left quadrant contains mostly features of the entry editor, a centerpiece of JabRef. Adding and editing entry details and attaching a PDF to an entry are the three features with the highest perceived importance, and they are integrated in the entry editor. Apart from the entry editor, the *Group* feature as well as features such as the import and export of references reside in that quadrant. As one of four features that were known to all respondents, *Find Duplicates* also was a member of this group. All of these features have already been found to be frequently used, based on the telemetry data (see Table 1), and their importance ratings support those findings.

Underdeveloped Features This quadrant consists of features with low awareness, which at the same time are not important to the users who actually know them. These are features that are not developed enough in JabRef to fulfill a

² <https://github.com/JabRef/2018-UserStudy>

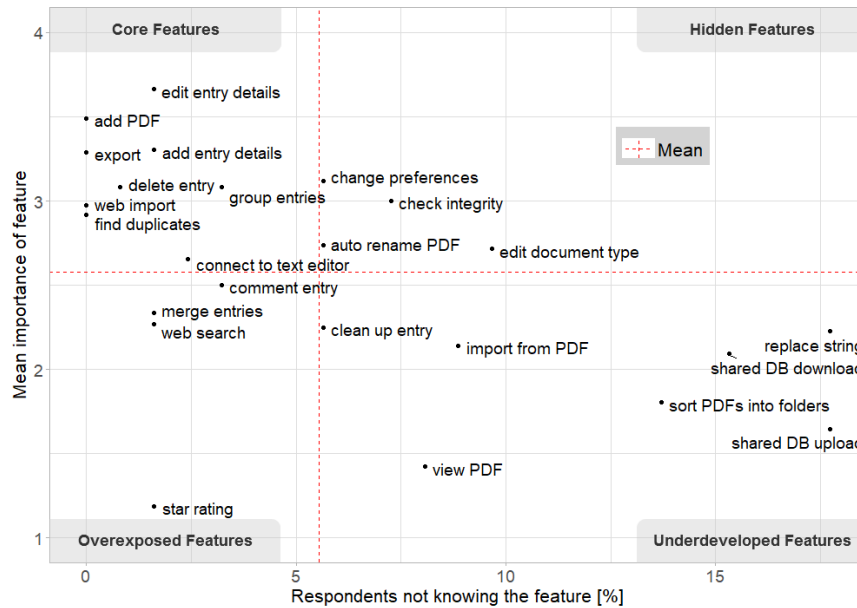


Fig. 1. Mean of perceived importance of a JabRef feature in relation to the percentage of users that were unaware of the feature

user need or are too cumbersome to use. Examples include the data transfer with shared databases and features for interacting with PDFs. The latter includes automatically sorting them into folders, importing entries through them, and viewing them in JabRef’s internal PDF viewer. While the import through PDF is actually used frequently by the users, the other features are not tracked in the telemetry data. Additionally, the tools *Clean Up Entries* and *Replace String* were in this group. Although these two did not differ a lot in importance, the latter was the least known feature of the survey.

Hidden Features While the features in the previous two groups are quite homogeneous, this is not the case in the other two quadrants. The Hidden Features consist of only four features, including the *Preferences* options, which scored fifth highest on importance out of all features and were found to be frequently used as well. Two other features in this quadrant were the automatic renaming of PDFs and checking the formal integrity of entries. The remaining feature was *Edit Document Type*, which was also the one with the least awareness out of the four. Differing from the other features in the entry editor, this one was, despite of scoring high on importance, not known to 10% of the respondents.

Overexposed Features This group contains features that many users are aware of but rate their importance as low. The Overexposed Features contained the star rating, which had the lowest importance rating of all features. A feature in this group that exhibited almost average importance was the *Comment* feature, which is the only feature of the entry editor not found among the highly important features. Additionally, the feature to merge entries and the *Web Search* are in this quadrant. The latter conflicts with the telemetry data, as it was among the top ten most used dialogs.

4.2 Interpretation and Discussion

Now that we have split features into groups according to our data, the question of how to proceed with these features in future versions of JabRef arises. In the following, we separate features that are important and, thus, should be included in future versions from those that should be removed, less exposed or transformed in other ways due to their low importance. Furthermore, we identify controversial features, which prove to be hard to assign to one of these categories because their importance varies depending on user characteristics.

Important Features The group of Core Features comprises exposed features integral to most users. Therefore, the exposure and availability of the features in this group should not be reduced in the future development of JabRef. The UI of these features should rather be improved to be more intuitive and easier to use, e.g. the layout of the group sidebar (see Fig. 2). Additionally, some of the Hidden Features are unanimously rated as important but lack the exposure to be found by all users. The *Preferences* and editing the document type are key features that should be available to all users. Therefore, the prominence of these features should be improved, e.g. by including *tip of the day* dialogs, which are common in modern IDEs.

Less Important Features Possibly obsolete features are indicated by low importance. If these features are also well known, it shows that they are made salient to the user, although they have little need for these features. They might distract the user from the actual work and can increase usage times [7]. Thus, we find that the *Star Rating* can be removed, as it is known to nearly all users, but has been rated as the least important feature of JabRef. What about the Underdeveloped Features? A considerable number of users was not aware of them and even the users that knew them did not rate them as important. Features like the PDF viewer and the interaction with shared databases represent useful features of RMS, but are not well developed in JabRef. The lacking maturity of these features would explain why users are hesitant to use them. At the same time, it gives a reason for their low exposure in the user interface, which in turn causes them to be unknown to numerous users. These features should be improved first and then integrated at appropriate places in the user interface.

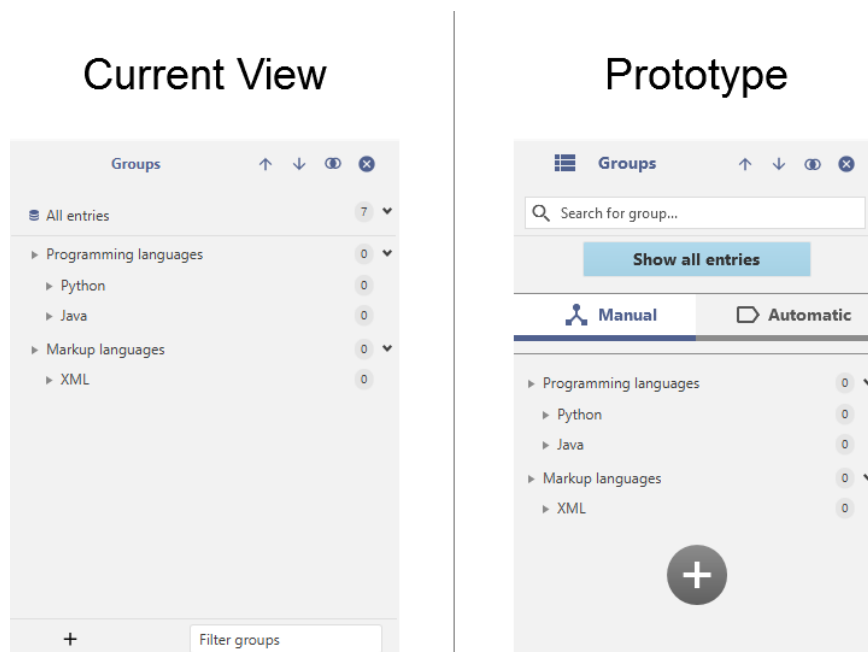


Fig. 2. High fidelity prototype of JabRef's Group feature compared to the current UI.

Controversial Features We found that, regarding some features of JabRef, there are two groups of users rating the importance of the feature differently. For example, permanent researchers rated the importance of the *Connect to Text Editor* feature as high, but PhD students regarded the feature as not so important. Therefore, features whose importance differs between the users need to be treated differently from the other features. Simply increasing the visibility of the features in the user interface would help the users that regard them as important, however, at the expense of distracting the others. Thus, it becomes necessary to find a way for these features to be accessible for the first group while keeping the user interface simple, e.g. through keyboard shortcuts.

5 Conclusions

This paper presented an approach within the framework of user-centered design to make use of telemetry data to design an online survey to investigate the importance of JabRef features. The answers to the survey were analyzed by contrasting the perceived importance and the knowledge about the existence of JabRef's features. Thus, the main contribution of this paper is an analytic grouping of the features using quantitative data and expert knowledge. We use this to derive guidelines and to make informed decisions in the current rework of

JabRef's user interface. The high-fidelity prototypes have been made available to the developer community on Github³, where it already sparked the interest of new contributors and enabled constructive discussions with the users of JabRef. The side-by-side, before and after approach [4] of the prototypes proved to be a very effective facilitator for discussions. The presented approach can be generalized and, thus, be of use to improve the usability of other software projects as well.

Bibliography

1. Andreasen, M.S., Nielsen, H.V., Schröder, S.O., Stage, J.: Usability in open source software development: opinions and practice. *Information Technology and Control* **35**(3) (2006)
2. Beel, J., Aizawa, A., Breiting, C., Gipp, B.: Mr. DLib: Recommendations-as-a-service (RaaS) for academia. In: *Joint Conference on Digital Libraries*. pp. 1–2. IEEE (Jun 2017)
3. Despalatović, L.: The usability of free/libre/open source projects. *International Journal of Computer and Information Technology* **2**(05), 958–963 (2013)
4. Harrer, S., Lenhard, J., Dietz, L.: *Java by Comparison: Become a Java Craftsman in 70 Examples*. Pragmatic Bookshelf, Raleigh, NC, USA (Mar 2018)
5. JabRef Developers: JabRef user survey (2015), <http://www.jabref.org/surveys/2015>, [Online; accessed March 11, 2019]
6. JabRef Developers: JabRef README (2019), <https://github.com/JabRef/jabref/blob/master/README.md>, [Online; accessed March 11, 2019]
7. Lowdermilk, T.: *User-Centered Design*. O'Reilly (2013)
8. Nichols, D., Twidale, M.: The usability of open source software. *First Monday* **8**(1) (2003)
9. Rudlof, C.: *Handbuch Software-Ergonomie* (2006)
10. Simon, M.K.: *Software Ergonomics of Free and Open Source Software: Augmenting the Usability of JabRef Through User-Centered Design*. Master thesis, Technical University of Munich (Jan 2019)
11. Ultimate Comparisons: *Ultimate comparison of reference management software* (2019), <https://ultimate-comparisons.github.io/ultimate-reference-management-software-comparison>, [Online; accessed March 11, 2019]
12. Viorres, N., Xenofon, P., Stavarakis, M., Vlachogiannis, E., Koutsabasis, P., Darzentas, J.: Major HCI challenges for open source software adoption and development. In: Schuler, D. (ed.) *Online Communities and Social Computing*, pp. 455–464. OCSC, Springer (2007)

³ <https://github.com/JabRef/jabref/projects/6>