Journal of Universal Computer Science, vol. 25, no. 7 (2019), 733-739
submitted: 20/5/19, accepted: 26/7/19, appeared: 28/7/19 © *J.UCS*

A Note Concerning Feedback and Queries for Web Pages

Namik Delilovic
(Graz University of Technology, Austria
delilovic@tugraz.at)

Hermann Maurer
(Graz University of Technology, Austria
hmaurer@iicm.edu)

Abstract: When studying a web page, the common problem is that users notice a mistake, or feel that some vital information is missing, or do not understand some explanation or wording. The possibly supplied list of frequently asked questions and answers is often not helpful, and trying to contact the web administration can be quite frustrating. In this note, we discuss methods that make it easier for users to send and receive meaningful feedback from a web server when needed. The problem is that the system has to be easy to use, yet must avoid misuse by a possible attacker and that the server administration gets valuable feedback, yet is not overwhelmed by the requests. After studying and experimenting with various techniques, we report on a solution that allows users to send feedback or ask questions, yet remain anonymous if desired, and also protects the web server by filtering feedback and questions using language recognition techniques. We propose and report on an implementation that makes such feedback mechanisms available to users whenever they feel the need for it.
Key Words: usability, anonymous feedback, user interface
Category: M.3, M.6

1 Introduction

When studying a web page, it is a common phenomenon that one notices an error, has the feeling that some important fact is missing, does not understand some formulation or terminology, or is unable to follow the instruction on how to continue, like filling out which part of the form in what way, etc. In such cases, it would be desirable to be able to communicate with the persons responsible for the website, yet this is often quite frustrating. If an error is discovered or some essential fact is missing the tendency of users is to shrug shoulders and continue (although the information provider would profit a lot from the feedback). If information is needed to understand some issue one may consult a possibly existing long list of FAQs, often quite irritating, since the specific issue one is interested in is not discussed. If an email is also missing, but one is lucky to find a phone number, ringing it can increase frustration further since one often is ending up in a long waiting loop. And if finally someone answers, the person might not even be able to help.

Recognizing this situation, we decided to find a proper solution when building up a substantial non-commercial server with information mainly of interest to

Austrians. Indeed, the server at issue [Austria-Forum 2019] is holding some 1.1 million media objects by the time of writing.

It is important to realize that the general question of feedback is of importance in many ways. However, not so much in our sense, where we try to improve the user interface and content, but mainly in the sense of what kind of navigation, layout and tools users prefer, or which features receive praise or criticism, or how user attitudes change over time or are dependent on other parameters.

[Quora: Feedback-Management 2019] provides a good survey of Enterprise-Feedback-Management, and [Quora: Customer Feedback 2019] presents over 60 tools for analyzing customer feedback. Some, like [Keatext: Feedback 2019] use language analysis tools to master massive floods of feedback by instant categorization of customers into Praise, Problems, Suggestions and Questions. Sometimes users are not even aware that feedback on their actions is collected. Typically, to find out which of two navigation alternatives A or B on a server are more preferred, customers are randomly presented with one of the versions, and their behaviour is recorded possibly without their knowledge. This is described at length in, e.g. [Wikipedia: Feedback 2019].

The aim of our efforts was different: We wanted to offer users a very simple way to suggest improvements of the user interface or content or to ask a question with the expectance of a fast concrete answer.

2 Early Attempts

Our first idea was to provide the option to write an arbitrary comment at the bottom of any page and send a note to the server administration. The server administrator is then able to react to a suggestion, leaving the comment if useful for others, or erase it with or without further action.

Initially, we were quite satisfied with the comments obtained this way. However, one night, a user wrote a script that added to ten thousands of pages an advertisement with explicit content. It took us much time to clean all the advertisements. Thus, anonymous feedback visible for all was not an acceptable solution.

In the next step, we restricted comments to users who had registered with a valid email, accepted certain “policies“ concerning comments, and were logged in (i.e. identified) when writing a comment: Indeed, their username would show up with a time-stamp below their comment. Also, the server administration was notified of every comment, and he was able to change it (e.g. correct some spelling, delete it if the desired information was provided, etc.). This method restricted misuse immediately, but also significantly reduced the flow of comments. It is interesting to see the numbers: Of some 2.5 million different users only a roughly 1.6 percent are willing to register: Clearly, the benefits available for registered users being able to write comments and having their own arbitrarily large space to use for any legal purpose were considered only by few worth the trouble to register. Even worse, of those who registered many would not log in when just looking briefly for some information. If in the process they would find an error or thought something should be added, almost nobody seemed to take the effort to log-in (with the chosen username and the possibly by now forgotten password). Only those who needed some advice did log in, or at that stage even registered to be able to log in. However, we did receive complaints that such questions for help should not be public because it might indicate that the user asking the question was just not very knowledgeable. We accepted the situation for quite a long time but discussed in the background how to improve it.

3 The Final Solution

Eventually, we decided to add to every page a feedback button that also allows anonymous users to send some arbitrary text (suggestion, criticism, question, etc.) associated with a particular webpage to the administrator of the server, invisible for everyone else. This is indeed working well: mistakes are pointed out, additional information is provided, criticism of the user interface can be taken care of; even feedback to the feedback process has been sent. A typical example was the feedback message concerning the biography of Mozart, where a book on Mozart that can be read online is mentioned:

URL: austria-forum.org/af/Biographien/Mozart,_Wolfgang_ Amadeus

Feedback: Could you not provide a table of contents of the biography after the picture of the book?

Figure 1: Above we show a typical short feedback as obtained from an austria-forum user.

Of course, the administration acted. Figure 2 shows what that part of the biography now looks like.

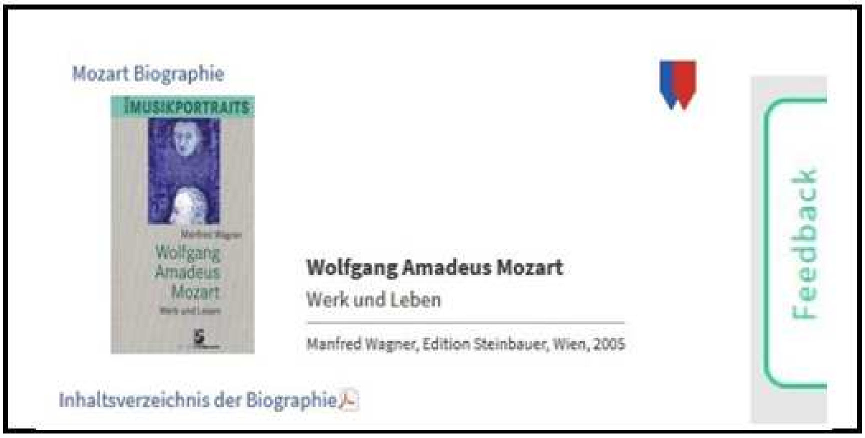

Figure 2: Below the picture of the book on Mozart is now a clickable PDF-File with the table of contents.

To be specific, we also show in Figure 3 what a click on the feedback button triggers: A very simple template in which users can write.

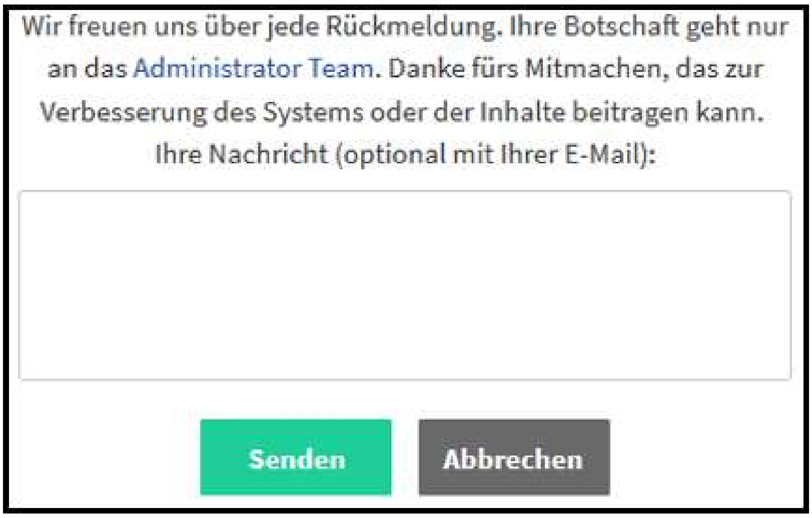

Figure 3: A simple area for textual input is provided and can be sent or cancelled.

Specific questions as feedback can of course only be answered if an email address is provided by the person sending the question (we guarantee that the email address is discarded after answering). However, specific questions have also taught us to expand individual sections of our help system.

The system is not quite as simple to implement, as it may sound. Since it can be used anonymously many users have tried it out with nonsense messages or just a short random sequence of characters; others have tried to overload the system by letting a script send thousands of messages. Hence quite elaborate filtering mechanisms are needed, to which we will return a bit later. There are two other points worth mentioning; When we presented the idea to a group of server administrators, they were appalled that they would be drowned with suggestions to change their user interfaces. However, our first experience shows that the feedback obtained concerning the user interface allowed to improve the interface, eliminating frustration and stabilizing the situation to the extent that fewer and fewer suggestions were obtained.

However, a kind of situation occurred that we did not expect. We would get a feedback like: “I do not understand the explanation concerning Point 3. Please elaborate more carefully“. Soon after we had done so, we got the feedback: “The explanation of Point 3 is too long, please shorten it“. Any attempt to have one explanation satisfying all seemed to fail, whether an explanation is sufficient or not depends on the level of expertise. A simple approach would be to provide a short and a long explanation, but in order to provide the correct kind and amount of feedback, a detailed profile of the user would be necessary. This is impossible in our case since we want to preserve the anonymity of users. Note that other approaches like “5 Steps for Giving Productive Feedback“ [Halford 2014] insist on giving up anonymity and compile a detailed profile.

4 Points to Notice

Introducing a feedback button and thus allowing all users to contact administration leads, at least initially, to a large number of emails. The system should not send each email separately to the administrator but collect them in a file and only send the file once it gets very large or in some user-defined intervals.

It is also much easier to look at a file of messages, often only test messages, than to open dozens or hundreds of emails where only some are useful.

The system should also exclude empty emails, or emails consisting of just a few characters since such short feedback do not carry important information in most cases. Note, however, that some short messages may be meaningful: like if on a page information is misspelled as “imformation“ maybe the user wrote “imformation“ to indicate a wrong spelling. To notice such helpful suggestions, an obvious way is to search for the word at issue in the page at issue, and if a match is found the message is not ignored.

Note that a feedback button also allows attacks on the server, using a script that sends thousands of spurious messages. The program can limit the number of messages per hour coming from one IP address, or can, like a SPAM filter, eliminate similar emails or emails similar to specific patterns. If used extensively, a learning algorithm can be employed to make sure that only meaningful feedback is obtained.

We do not want to discuss details of such methods, since knowing the algorithms that eliminate superfluous feedback may give hints on how to send a flood of feedback evading the recognition of meaningless messages.

However, we want to conclude this note with a few numbers indicative of what happens on fairly large servers. We introduced two counters: one counting all feedback received, and one counting those ignored (because of repetition or being short and providing no new input). In the first four weeks, the figures were 503 and 261, in the next four weeks 249 and 128, in the third four weeks cycle already dropping to 165 and 92. By now the four-week number seems to be stable around 100 and 50.

5 Conclusion

It is our firm belief that every website should provide an easy way to send feedback as described. It will certainly reduce the frustration that all of us have experienced with some sites, apps or such.

This must be a feature of all systems designed with good usability, as is pointed out to some extent in the literature on usability, like [Nielsen 1999] or [Law, Hvannberg and Cockton 2008] or the reference on feedback mentioned in the references.

Typically, we expect that all government agencies and big organizations make sure that some convenient feedback mechanisms are available. To drive our point home: We are certainly aware of cases when it was challenging to contact a suitable person in a large company.

In summary, it appears that the solution developed, works well on non-commercial servers of up to a few million pages and users if the only aim is to allow user feedback for content or user interface improvements, or for asking specific questions. It also has become clear that the mechanism could be extended in many ways, even for the purposes mentioned. Like questions asked could be put into a FAQ list, and a match of new questions with existing ones could be attempted, sometimes providing immediate answers, yet not burdening the user to dig manually through a vast set of FAQs. There are of course many other instances where feedback is required or useful that are not addressed by the presented simple solution, as hinted in the last four literature references.

Acknowledgements

We want to thank the four reviewers for their valuable recommendations and criticism that have helped us to improve this final version.

References

[Austria-Forum 2019] Austria-Forum. `https://austria-forum.org` Access: July 2019.

[Halford 2014] Halford, S.: "5 Steps for Giving Productive Feedback"; Entrepreneur (November 10, 2014). `https://www.entrepreneur.com/article/219437` Access: July 2019.

[Keatext: Feedback 2019] `http://www.keatext.ai/en` Access: July 2019.

[Law, Hvannberg and Cockton 2008] Law, L., Hvannberg, E. and Cockton, G.: "Maturing usability"; Springer, London.

[Nielsen 1999] Nielsen, J.: "User interface directions for the Web"; Comm. ACM 42, 1 (January 1999), 65-72.

[Quora: Customer Feedback 2019] `https://www.quora.com/What-are-the-best-customer-feedback-analysis-tools` Access: July 2019.

[Quora: Feedback-Management 2019] `https://www.quora.com/topic/Enterprise-Feedback-Management` Access: July 2019.

[Wikipedia: Feedback 2019] `https://en.wikipedia.org/wiki/A/B_testing` Access: July 2019.