

Estimating Time Requirements for Web Input Elements

Ludger Martin

Institute of Computer Science, Johannes Gutenberg-University Mainz, 55128 Mainz, Germany

Abstract. How do you know whether a website or Web application is usable? What you need is statistics about how the website is used. Such statistics usually include information about how much time a web user spends on a website. But to know whether those dwell times are acceptable you will need reference values. This paper provides such reference values based on a research study. After a detailed discussion of the study's results, a table is presented, which provides reference values for any kind of HTML/HTML5 input elements. This table can be used to estimate the time it will take a user to fill in a form.

1 Introduction

Nowadays, statistics are maintained for almost every website on the Internet. Two of the criteria that are monitored are the time a web user stays on a particular web page and which web pages the user visits. Then the results are analyzed to see whether these times are good or bad, and why.

This paper aims at providing criteria for more precision in assessing dwell times by examining HTML and HTML5 input elements. A survey shall help to estimate the time it will take a user to fill in a form on a website. To this end, statistics about the various types of input elements will be created. It is our aim to enable an estimation of the average time it will take users to fill in any Web form.

Using this data will help to assess the usability of websites and web applications. The time required for filling in forms can then be supplemented with further values obtained about a website. For example, the quantity, the readability and the main subject of a text, as presented in [6, 5]. All of this information as a whole will help to understand the statistics about a website.

The rest of the paper is structured as follows: In the chapter 2 the study and the questionnaire used will be presented. Then, in chapter 3 the results of the study are discussed. After presenting related work in 4 we conclude the paper with an outlook of future work.

2 Survey Questionnaire

An online questionnaire has been compiled for this study. Test persons are asked to input their answers to the various questions into the questionnaire. While they are filling in the form, a JavaScript records the time the test persons spent on each question. To make the time measurement more informative, the questions that are not currently dealt with

are grayed out. This method ensures that the test persons can read and answer only one question at a time. It is the test person's decision whether they want to use a desktop browser or a mobile browser.

The questions do not require much contemplating, e.g. the first question asks the users to describe what they did after getting up in the morning. This is followed by a choice of seven radio buttons about how the test persons came to work that day. Then they are asked to enter their favorite food in a single-line text field. Additionally, statistical data is maintained about the person's gender using three radio buttons. The next question asks for another piece of text. This time, a word from the title of a book is to be entered. As a rule, the type of input element is never the same as the one directly before. The next four questions use check boxes. A further statistical question has the form of a select field. Here, age ranges are chosen. Six lines are displayed at the same time. If users want to enter an age older than 34, they must scroll. The effect of scrolling will be described in greater detail later on. In the next field, the date of the day before is required. If the browser supports HTML5, the appropriate HTML5 date field will be displayed, if not, a simple text field. In our analysis we will look at these differences in greater detail. If the browser supports them, the color field and the range field will also be displayed. The form is completed with a simple submit button.

The other new HTML5-elements such as number, url or email are just short text fields and are therefore not considered separately in this study.

3 Analysis

In January and February, 2013, a total of 80 test persons participated in the study. The diagrams below show in round brackets the number of test persons who have filled in a particular input field. The differing numbers are due to the fact that not all test persons filled in all of the input fields and, depending on the browser, not all of the HTML5 input elements were displayed. In the following examinations only the values of the test persons who used a desktop browser were analyzed.

Figure 1 shows the input elements which can be selected using the mouse or the keyboard. The four check boxes together required an average time of 15.4s. Compared with the three radio buttons about gender, we note that these could be answered much faster (5.8s). Even the seven radio buttons took less time. The reason for this is that each check box requires an individual answer while the radio buttons belong together. But the time required to deal with a group of radio buttons depends on the number of group elements. The select field for the person's age is similar to the radio buttons, as it required a similar amount of time. Entirely different is the range field. Here, a user is to choose a predefined value. This field seems to pose most of the problems as it requires an average time of 13.3s. To facilitate applying the figures to a form you want to analyze, we have recalculated the numbers for the individual elements also. For example, it takes 3.0s to deal with a check box. The figure also shows that the standard deviation varies considerably. We will look at this in greater detail later in this section.

Now we will take a closer look at the select field. Figure 2 relates the time with the options chosen. At the same time, the options represent the test persons' ages. The first options show similar times and standard deviations. Only starting with option '35-39'

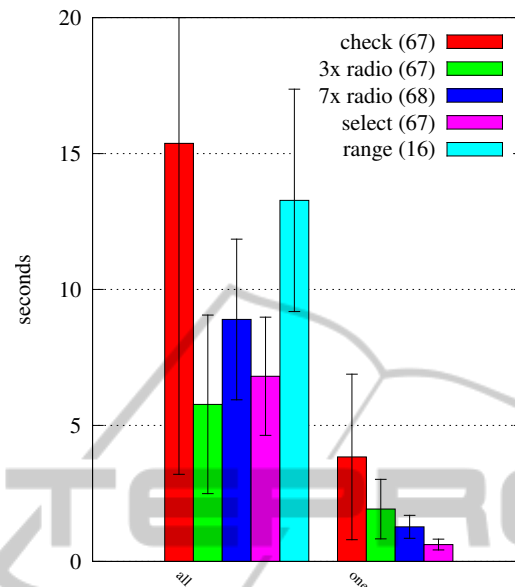


Fig. 1. Selection Input Elements.

does it become necessary to scroll down. This shows that if more scrolling was necessary the required time increased considerably. The increase in time might be explained by the higher age of the test person. But admittedly not many senior persons participated in the study and therefore these values must be used with caution.

In the second part of figure 2 we have tried to create a formula for select field usage. For choosing an option without scrolling we assumed 5.6s. For each line that must be scrolled we have added 2.3s. The figure shows this using the age list as an example, where quite a good approximation can be seen. We have allowed extra time for the standard deviations as well.

Next we will look at the three elements which require text entry using the keyboard. The date field is left out as no normal text is entered there. The angle brackets in the figure 3 specify the average numbers of letters that were input. The times refer to one character each. Interestingly, the question about a person's favorite food was answered more quickly than the question about a word from the title of a book. The test persons obviously thought longer about the second question. Another observation was that longer pieces of text can be entered significantly quicker.

Before that, larger standard deviations had been noted. These are now compared to the test persons' ages. Figure 4 shows minor standard deviations within the age ranges. The larger values were created by the senior test persons (60 or 70 and older) as also observed in the figure 2. If you know the age of a website's target group, then only the average values for this age group are to be used. This paper continues to look at all age groups.

After examining the age, now we want to look at the gender. In addition to the check boxes and the text area, the figure 5 lists three age groups with the majority of test

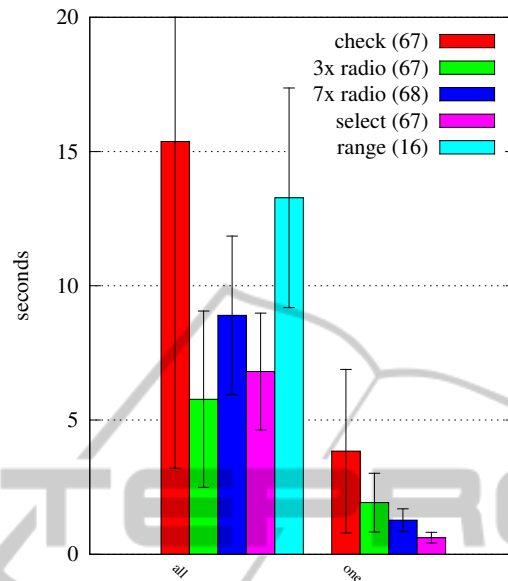


Fig. 2. Selection and Scrolling.

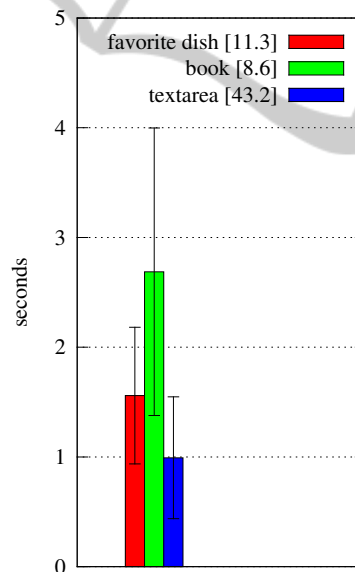


Fig. 3. Text Input Elements.

persons. Notably, with the female test persons there were fewer standard deviations. This may be due to the fact that fewer women participated in the survey. Hardly any difference can be seen between the female and male test persons at least in the age groups '20-24' and '30-34'. The big difference in the text area of the age group '25-29' may be explained by the small number of test persons.

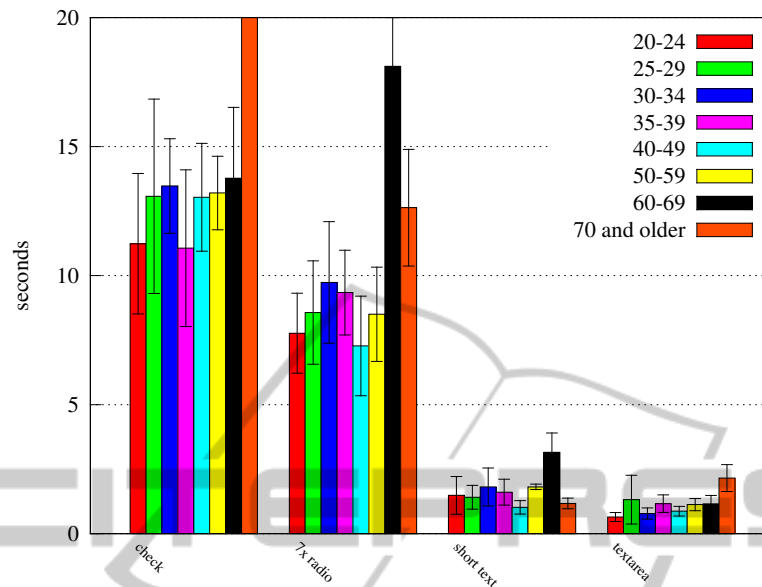


Fig. 4. Effect of Age.

One novelty in HTML5 standard is, among others, an input field for a date. In figure 6, this date field is compared to a field where the date must be typed in. The date field includes a calendar from which the date can be chosen. With two clicks you can select a date. In a text field, however, 10 characters must be entered. The figure 6 shows this extra effort. Date fields are about double as efficient as a text field where the date must be entered. Another new HTML5 is the color field. An average time of 13.5s could be determined for that field. However, only three people filled in this field.

So far we have only evaluated data for desktop browsers. The figure 7 compares the test persons who used a mobile browser to those who used a desktop browser. For date fields or color fields no data was available. Apart from the range field there were no big differences. The range field had already been different in desktop browsers and this proved even more true using mobile browsers. This makes us question the usability of the range field. Text fields hardly showed any differences. With longer texts the time spent per character was slightly more. This can be explained by the fact that typing on mobile devices is less convenient.

The form is sent off using the submit button. We did not examine these buttons as they are only pressed once to submit the form. Measuring time is hardly possible, especially as every form is submitted using this button and users don't spend much time on this kind of button.

The table 1 shows a summary of all times we ascertained. We calculated only one value for all of the check boxes and one for all of the radio buttons. For mobile browsers we could not determine values from scrolling a select field, as too few participants were available. For the texts, as before, we only specified the input times for one character.

The times shown in the table 1 shall be used to estimate the time it will probably take to fill in the entire form. Finally, as an example we will use these times for the

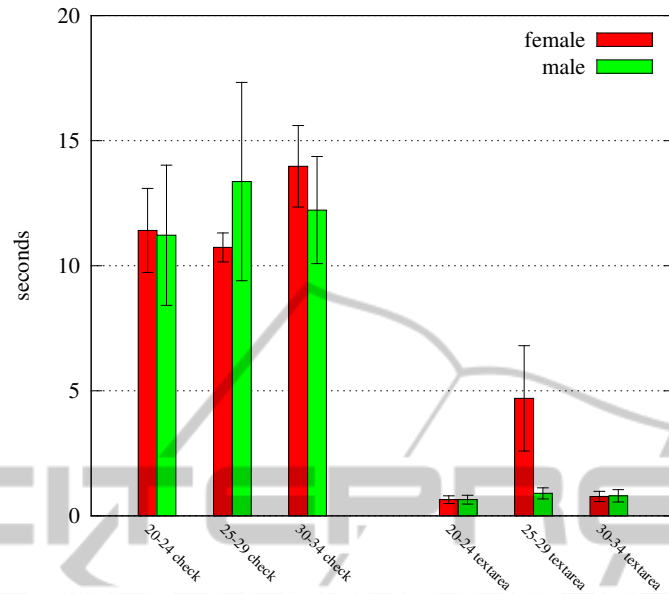


Fig. 5. Effect of Gender.

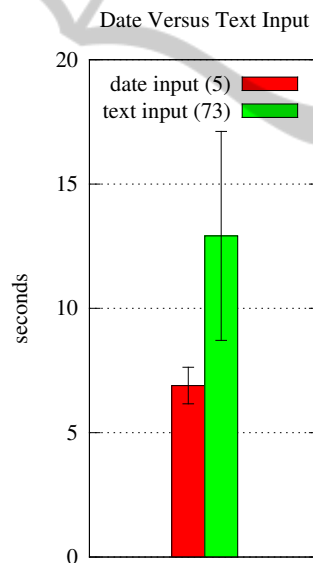


Fig. 6. Comparison of Date and Text Input.

form in the study. In table 2 you will find a calculation based on a desktop browser. If there are any optional fields in a form, we need a maximum (if all optional fields were filled in) and a minimum value (if no option was chosen). For the maximum value the estimated times must be included in the calculation, while for the minimum value they should be left out. The calculation of time to be spent on our form is 144.22s (see fig.

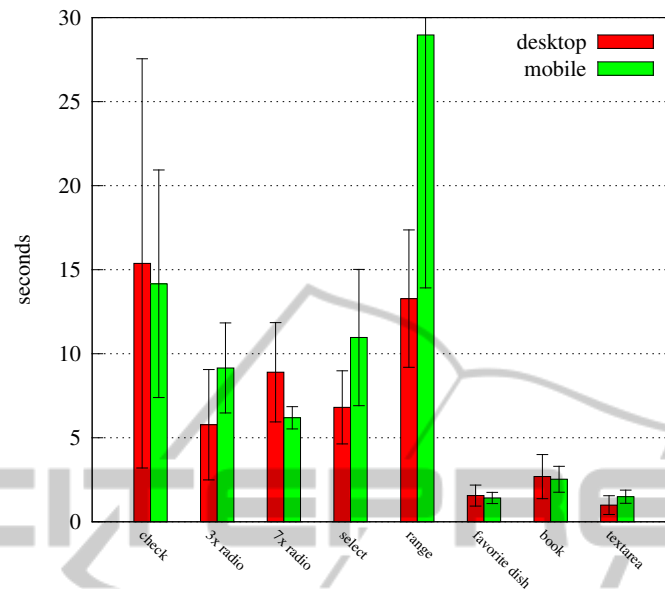


Fig. 7. Comparison Desktop/Mobile.

Table 1. Selection and Scrolling.

	desktop	mobile
check	3.84s	3.54s
radio	1.60s	1.97s
select	5.76s	10.60s
select scroll	1.30s	-
range	13.28s	28.96s
short text	2.12s	1.97s
text area	0.99s	1.49s
date	6.90s	-
color	13.50s	-

2. If you calculate the real value from looking at the first element until submitting the form, you will get a value of 142.30s. Thus, the estimation is very close to the time that is really required.

4 Related Work

We could not find any studies that dealt with similar estimations about form elements. This is why we here present related studies which deal with web usability and web statistics in general.

Atterer *et al.* [2] built a system to track all user interactions with a browser page including mouse pointer, text input, and page scrolling. Similar to our approach, the interaction is tracked by a JavaScript but the data is transmitted using an image object. The web pages are modified automatically by a proxy server to enable the tracking. The

Table 2. Time Estimation.

text area ca. 40 characters	$40 * 0.99s =$	39.60s
radio group with 7 buttons	$7 * 1.60s =$	8.60s
text with approx. 10 characters	$10 * 2.12s =$	21.20s
radio group with 3 buttons	$3 * 1.60s =$	4.60s
text with approx. 10 characters	$10 * 2.12s =$	21.20s
4 check boxes	$4 * 3.84s =$	15.36s
select field with 5 scroll options	$5.76s + 5 * 1.30s =$	12.26s
date field (as text)	$10 * 2.12s =$	21.20s
sum		= 144.22s

proxy server also protocols the usage data. The mouse-over event is recorded for every HTML element in the same way as in our study.

The tool Wusab is presented in [1]. Atter claims that automated usability validation is one of Wusab's features. During the website development the validation runs at regular intervals. In our tool the usability is checked on the request of the usability engineer. The Wusab tool downloads and analyzes HTML code. The analysis is done using guidelines such as *images should have alt text*. Our tool also analyzes the web page but offers more complex criteria such as the input elements used, readability [5], or subject [6].

Another tool is Google Analytics [3] which is very powerful analysis tool. It works with its own database, whereby statistics can be provided independently of the web server used. The integration into a web application is done using JavaScript. The tool collects various items of information and sends them to the database by loading an invisible picture. Whereas Google Analytics provides many different statistics concerning users, sources, contents and goals, no information, however, is available on the visiting paths of the users. Furthermore, Google Analytics does not analyze the website itself.

5 Summary and Future Work

In this paper, using a study we have examined how long it takes test persons to fill in single input elements of a form on a website. The times spent by 80 persons to fill in the input fields were discussed in detail. As a result, the table 1 has been created, which enables the estimation of the average time needed by one user to fill in any web form.

In [4] we presented a tool which can be used to analyze how people use a website. This tool is able to recognize paths that a single user or groups of users have taken. Additionally, the tool determines the corresponding dwell times. These dwell times can be validated by calculating the times described above for the forms offered. However, websites do not consist only of forms but they also have other content. In [5] we have collected further information on websites.

Our next aim is a case study which is to demonstrate that the average times calculated here will be useful for estimating the usability of forms on websites. To do so, we will examine websites and web applications with different forms.

References

1. Richard Atterer. Model-Based Automatic Usability Validation - a Tool Concept for Improving Web-Based UIs. In Proceedings of the 5th Nordic conference on Human-computer interaction: building bridges, pages 13–22, New York, NY, USA, 2008. ACM.
2. Richard Atterer, Monika Wnuk, and Albrecht Schmidt. Knowing the User's Every Move - User Activity Tracking for Website Usability Evaluation and Implicit Interaction. In WWW '06: Proceedings of the 15th international conference on World Wide Web, pages 203–212, New York, NY, USA, 2006. ACM.
3. Google. Google Analytics. <http://www.google.com/analytics/>, accessed March 2013.
4. Ludger Martin. A Tool to Estimate Usability of Web 2.0 Applications. In 11th IEEE International Symposium on Web Site Evolution, September 2009.
5. Ludger Martin. Gathering Information About Web 2.0 Applications for Usability Engineering. In Seventh International Conference on the Quality of Information and Communications Technology, pages 482 – 486, September 2010.
6. Ludger Martin. Subject classification of web pages. In IADIS International Conference WWW/Internet 2012, pages 298 – 306, October 2012.

