IESE

Fraunhofer Institut Experimentelles Software Engineering

Conducting On-line Surveys in Software Engineering

Authors:
Teade Punter
Marcus Ciolkowski
Bernd Freimut
Isabel John

Accepted for publication in
Proceedings of ACM-IEEE
Int. Symposium on Empirical
Studies in Software Engineering
(ISESE 2003)

IESE-Report No. 058.03/E
Version 1.0
July 2003

A publication by Fraunhofer IESE

Fraunhofer IESE is an institute of the Fraunhofer Gesellschaft.
The institute transfers innovative software development techniques, methods and tools into industrial practice, assists companies in building software competencies customized to their needs, and helps them to establish a competitive market position.

Fraunhofer IESE is directed by
Prof. Dr. Dieter Rombach
Sauerwiesen 6
67661 Kaiserslautern

Abstract

One purpose of empirical software engineering is to enable an understanding of factors that influence software development. Surveys are an appropriate empirical strategy to gather data from a large population (e.g., about methods, tools, developers, companies) and to achieve an understanding of that population. Although surveys are quite often performed, for example, in social sciences and marketing research, they are underrepresented in empirical software engineering research, which most often uses controlled experiments and case studies. Consequently, also the methodological support how to perform such studies in software engineering, is rather low. However, with the increasing pervasion of the Internet it is possible to perform surveys easily and cost-effectively over Internet pages (i.e., on-line), while at the same time the interest in performing surveys is growing. The purpose of this paper is twofold. First we want to arise the awareness of on-line surveys and discuss methods how to perform these in the context of software engineering. Second, we report our experience in performing on-line surveys in the form of lessons learned and guidelines.

Table of Contents

1 Introduction

Performing surveys is a well-known strategy for doing empirical studies. It is especially known from its use in disciplines like management science and social sciences. Asking people about their perceptions and opinions on a certain topic is a central issue when doing research in those disciplines. Although software engineering is impacted by human science, too [1],[2], we notice less attention for performing surveys compared to the attention paid to case studies and experiments in software engineering (SE). For example, [3], [4] and [5] provide extensive information on how to perform experiments, but do not provide an integrated approach on how to conduct surveys, although they stress the importance of surveys. We faced ourselves this lack of interest for surveys in SE during the development of a tutorial on empirical studiesin software engineering for a national research project [6]. We found over one hundred literature references on experiments (see, e.g., [7]), about twenty references on doing case studies in Software Engineering, and only about ten references for surveys.

Although the attention in the literature on software engineering methodology for performing surveys is low, we observe a growing interest for doing surveys in software engineering research practice. Especially, their capability to determine a trend or relationship for a particular SE technique or method makes the surveys attractive. A study amongst partners in the software engineering network ESERNET[1] [8] showed that 50% of the respondents (ten out of twenty) have conducted a survey, sixteen responding partners had experience with case studies and thirteen partners had experience with (controlled) experiments. Also Kitchenham's and Pfleeger's recent series on surveys in SIGSOFT-notes [9] shows the growing interest for surveys in software engineering. Another illustration for the interest in surveys is the distribution of empirical studies done within the German ViSEK-project[2]. Out of the twenty studies that are conducted at the moment in this project on transfer of software engineering knowledge, nine are surveys, which is equivalent to 45%.

There are several ways to perform surveys. Therefore they are often categorized according to the medium that is applied during data collection. Typical survey types are mail-surveys, street-surveys, or telephone-surveys [10]. With the arrival of the Internet it is also possible to conduct surveys electronically. The questionnaire of the survey is then made accessible via a website or sent via e-mail. The ability to collect large amount of data without large cost for interviewing or postage as well as the ability to process answers without a manual data entry

[1] www.esernet.org

[2] www.visek.de

from paper-based questionnaires into a computer makes electronic surveys attractive.

In this paper we discuss the role of on-line surveys – which are a subset of electronic surveys – in the empirical. software engineering community and share our experience in conducting on-line surveys. This is done by characterizing survey in software engineering in Section 2. Section 3 continues on the advantages and disadvantages of electronic surveys as reported in survey literature as well as observed in our own studies. To guide the reader in performing on-line surveys we describe the process for conducting on-line surveys in Section 4; hereby, we give guidelines for several phases of online surveys, namely for the design, implementation, execution and analysis phases.

2 Characterizing surveys in SE research

A survey is a strategy or design for an empirical study to "provide a quantitative or numeric description of some fraction of the population – the sample – through the data collection process of asking questions of people" [11, 12]. The facts, opinions or beliefs of the people are polled to determine how the population reacts on the particular topic.

According to [3], surveys may have a descriptive or retrospective character. This type of survey aims often at providing a state-of-the-art overview on a particular method, tool or technique. For example, a state-of-the-art overview on the use of integrated development tools would consider questions like: which tools (of which suppliers) are applied, for which reasons, with what satisfaction. Data are collected by asking people with a specific function; for example, quality assurance, developers, and/or representatives of specific departments within different organizations.

In [5] it is stated that surveys are also used to make explorative claims about the population. The purpose of explorative surveys is to discover opinion relationships in new areas. An example of this survey type is trend surveys, in which the demand for a particular product or service is evaluated or predicted. Such surveys are often conducted by investigators like Ovum and Gartner/IDC and are based on people's expectations on market for specific technologies or methods, such as a survey on the prospects for Sun One and .Net.

When comparing surveys to the other empirical strategies – experiment and case study –, surveys have the least control on variables. Thus, it is hardly possible to manipulate variables in order to investigate cause and effect relationships. Consequently surveys have lower internal validity compared to case studies and experiments. On the other hand, typically a large number of people from different real-world contexts are involved in a survey. Therefore, the results are better to generalize than data from a controlled experiment or a case study. Therefore, surveys have in general a higher external validity than experiments and case studies. With respect to the scope of the study, surveys can be regarded as"research in the large", compared to controlled experiments, which are often regarded as "research in the small" due to their laboratory setting [13].

Having set the context for surveys in software engineering, we will have a further look on the impact of doing such surveys electronically. We distinguish two types of electronic surveys. The *e-mail survey* is the first type. It is conducted by sending e-mails with a questionnaire as attachment. These surveys are easy to be sent and they require little technological skill to be developed. Usually, the questionnaires are in a text format like ASCII-Text or MS Word. This requires people to open the attachment, possibly to print it, to fill it in, and finally to send it back. Unfortunately, all these activities increase the probability that people refrain from filling in the questionnaire, as it is additional work to them. Additionally [14] report that people might have a reduced feeling of anonymity – because the respondent's e-mail address is generally included with his/her responses – which negatively influence response rate (i.e., the rate of returned completed questionnaires) of email-surveys.

The second type of electronic surveys is the *www- or on-line survey* in which people fill in a questionnaire presented on a web page. Typically, the respondents are invited by e-mail, newsgroups or advertisements on other websites to visit the website. The advantages and disadvantages of such on-line surveys are discussed in more detail in the next section.

3 Advantages and Disadvantages of on-line surveys

Online surveys offer several advantages in comparison to mail surveys (electronic and paper-based). First of all, the effort to handle the questionnaire is reduced for the participants. They have simply to follow the link to the web page and fill in the questionnaire. Thus, it is no longer necessary, to open a questionnaire from a mail, print it, and, most important, take the time to send it back. Second, also the task of filling in the questionnaire is simplified for the respondent, as answer options can be conveniently presented in the form of pull-down menus or check-boxes. Moreover, in on-line surveys it is possible to navigate easily through the questionnaire; that is, to present or skip questions based on the respondents' answers to previous questions. Thus, the respondent is unaware of complicated skip-patterns in the questionnaire, which are typical sources of confusion in paper-based questionnaires.

But also for the researcher on-line surveys offer appealing advantages. First, since the data are entered on the web page by the respondents they are already available in electronic format, they can be directly stored in a format suited for analysis (e.g., as spreadsheet table). This makes the manual data entry of paper-based forms, typically a high cost factor as well as error prone in paper-based surveys, unnecessary. Second, since information on the number of invited people, the number of people having completed, and questions producing the most drop-outs are available, it is possible to manage and control the response rate of the survey (e.g., by follow-up mails).

These advantages that we consider as beneficial assets of an on-line survey are corroborated by a survey amongst survey professionals on the advantages and disadvantages of electronic surveys [14]. This survey showed that the professionals were most positive about the reduction of costs (i.e., postage, phone charges) associated with electronic surveys, the use of electronic mail for pre-notification or follow-up purposes as a complement to other survey delivery methods, and the compatibility of data with existing software programs.

However, these survey professionals found also problems with doing electronic surveys concerning respondents' knowledge and their experience with technology. It is believed that individuals who are not comfortable with technology will not respond [14]. Web pages are indeed a new technology, at least compared to paper and pencil, and may therefore be uncomfortable for some people. In our opinion, this is less a problem for software engineering surveys, because those require mainly the participation of software engineering professionals, who are expected to be well skilled in web page technology. Therefore, we consider the technology not as a problem, although the risk should not be underestimated.

An additional task in performing surveys on-line is to bring the questionnaire online in the form of web-pages and to store the entered data. This task, if performed manually, requires an advanced technological skill and effort.

Another disadvantage is that technological skills are required to develop such surveys. We were faced with the technological complexity of designing online surveys on our own and saw it as severe problem. In order to circumvent these problems we decided to use an existing web survey tool management system, which allowed us to conveniently and quickly bring the survey on-line. At the moment several tools are available, such as Globalpark[3], iSurvey[4], or eSurvey[5.]

Considering all these advantages and disadvantages we conclude that on-line surveys are an effective tool for surveys in empirical software engineering. Using appropriate tool support, a questionnaire can be made quickly on-line and respondents can be invited cost- effectively via e-mail. Respondents can easily answer the questionnaire using the familiar web browser. Finally, the survey data are available in real-time in electronic format, speeding up data analysis.

[3] www.globalpark.de

[4] www.iwebsoftware.com

[5] www.logisoft.com

4 How to perform on-line surveys: process and guidelines

In this section we present a process how to perform surveys in general. In the following subsections we refine this process by discussing those aspects that are particular to on-line surveys.

Our process to perform an on-line survey is an instantiation of a general process for conducting empirical studies in software engineering. Various instances of this process have been described meanwhile, see e.g., [6], [5], which is also applicable for the empirical strategies experiment and case study. For surveys the process is sketched in the following table.

Activity	Purpose	Survey specific issues
Study definition	Determine the goal of the study	No specific issues
Study Design	Operationalize the study goals into a set of questions and select the respondents.	Questionnaire design, Define target population and sampling procedure Address validity issues.
Implementation	Operationalize the design so that the survey will be executable	Check completeness and understandability of questionnaire, Determine how to distribute the questionnaire (by post, mail), Define interview guidelines, Brief interviewers
Execution	Collect data and process data	Send out questionnaire, Collect filled-in questionnaires, Conduct interviews
Analysis	Interpret the data	Check on correct entering of data
Packaging	Report about the survey results	No specific issues

Table 1 Activities in the process to perform on-line surveys.

We applied this process in five on-line surveys that we performed since 2002 as shown in table 2. The survey process phase that is particularly affected by the choice of performing a survey on-line is the implementation task, because the infrastructure for the survey has to be prepared. Preparing the infrastructure means transferring the questionnaire into a web application, which is a technological challenge. However, other process steps such as design, execution, and analysis are affected as well. In the following subsections we discuss important aspects of the process activities, point out problems, and present guidelines for conducting on-line surveys, which we have derived from our surveys.

Survey Name	Description
Inspections 1	The survey goal was to describe the state of the practice in reviews and inspections. Doing that, the focus was on the process of reviews to find out how organizations that use reviews actually apply them. The survey was conducted in July 2002 in the context of the German ViSEK project. It used personalized invitations; more than 800 people from German companies were invited, and 120 responses were received.
Inspections 2	This survey had the same goal as the Inspections1 survey. However, it was targeted to developers worldwide, and it was conducted within ISERN[6]. People were self-recruited; that is, they were invited through a snowball system using the ISERN members, and additionally through mailing lists and newsgroups. In total, 105 completed questionnaires were received.
Portal evaluation	The survey aimed at evaluating the ViSEK portal and asked for the expectations of its potential users, the attractiveness and applicability of the information provided by the portal, the usability of the portal and the business model of the portal. All (285) Beta-testers of the portal were personally invited.
Reuse 1	This survey aimed at giving an overview on the systematic software development, reuse potential in development and factors relevant for product line engineering Respondents were self-recruited software developers in Germany.
Reuse 2	This survey had the same goal as the Reuse 1 survey but it used personalized invitations.

Table 2 On-line surveys of the authors

4.1 Designing on-line surveys

Designing an on-line survey requires the selection of the respondents, for which sampling is an important aspect. These respondents have to fill in a questionnaire which makes the questionnaire design an important aspect. In the following we discuss each of these issues.

4.1.1 Sampling

Sampling is the process of selecting a set of respondents (the sample) as a subset from the entire population under study. The reason of using only a subset is to save time and money (compared to asking all potential respondents). There are two general ways to obtain such a sample [15], [9]. The first one is probability sampling, in which a systematic approach is used to selectthe subset of the total population. The second one is non-probability sampling, in which the researcher selects the sample because it is convenient.

[6] International Software Engineering Research Network; see www.iese.fhg.de/ISERN

Systematic sampling requires that a list of the entire population is available and the sample is drawn so that that every member of the population has statistically seen an equal chance of being selected. The advantage of such a procedure is that it is possible to derive results from the sample that also hold for the entire population with only a low (sampling) error. The size of the sample depends then on several factors, namely: the sampling error (the degree of precision of the sample taken) that can be tolerated, the population size, and the variety of the population. Although survey populations in software engineering are smaller than in other fields, we are still dealing with larger populations than case studies or experiments.

Non-systematic sampling is applied when systematic sampling is not possible (e.g., when the individuals of the entire population are not known) or in small-scale surveys. There are several techniques for that. One is convenience sampling, in which the researcher chooses convenient persons to act as respondents. This strategy can be refined by characterizing various groups of a population and perform convenience sampling in each group. The non-systematic approach of drawing a sample implies that information on the accuracy (on the error between population and sample) is not available. Instead the characteristics of the respondents can be determined afterwards to characterize the sample. For example during our survey Inspections2, data on the coverage and non-response error were got by checking afterwards whether the respondents came from different contexts, and whether all relevant types of companies were covered. Although convenience sampling has the risk of being biased, while people are asked that might not be representative for the population, we applied convenience sampling mainly during our surveys because our target populations were hard to identify. Either those company representatives were asked, for which we had addresses or those which were interested in joining our survey (i.e., recruited themselves).

The choice of a sampling method is also related to the type of on-line survey that is to be performed. The first type of on-line surveys is personalized surveys, in which each member of the sample is known and is personally invited to participate in the survey. Typically some sort of access control is performed to assure that only people in the sample are able to access the questionnaire. For example, passwords or keys that enable linking to the web-questionnaire are enclosed in the survey invitation. This type of survey can be performed with both systematic and unsystematic sampling.

The second type of on-line surveys are self-recruited surveys, in which the respondents get to know somehow of the survey and decide to participate. For example it is possible to notify visitors of the web site with a pop-up window of the survey or it is possible to e-mail invitations over newsgroups, mail groups, etc. that inform about the survey and its location. Naturally, this type of on-line survey will result in a non-systematic sampling approach. An advantage of self-recruited invitation is that those people are addressed that were attracted already by the topic of the survey (because they are already visiting the web site). Those people may be eager to respond. The problem self recruited surveys is the convenience sampling that makes that conclusions from the survey are hard to generalize for a larger population other than the sample group, because the sample might be atypical for the total population. Therefore on-line surveys do not specifically address the appropriate definition of population and sampling that is regarded as a major problem for software engineering surveys by [9].

In order to give a feeling of sample sizes we dealt with in our surveys, table 3 shows an overview of the characteristics of five surveys that we conducted. Column 2 denotes the type of on-line survey performed and Column 3 denotes for personalized surveys the number of invited persons. Columns 4 and 5 show how many respondents completely filled in the questionnaires and how many respondents started the questionnaire but dropped out.

1	**2**	**3**	**4**	**5**	**6**	**7**
Survey Title	**Type**	**Sample**	**No. of completed questionnaires**	**No. of incomplete questionnaires**	**No. of Questions**	**Average Time to respond; minutes**
Inspections 1	Personalized	865	121	234	~ 90	52
Inspections 2	Self-recruited	-	220	105	~ 90	75
Portal evaluation	Personalized	281	55	22	~34	15
Reuse 1	Personalized	1020	84	51	~40	21
Reuse 2	Self-recruited	-	45	172	~40	18

Table 3 Key figures on the on-line surveys.

4.1.2 Questionnaire design

The second important aspect in the design step of a survey is questionnaire design. It is an important aspect of every survey: because 'no matter how big and representative the sample, no matter how much money is spent on data collection and what the response rate is, the quality of the resulting data from a survey will be no better than the questions that are asked' [16].

In addition of formulating relevant and good questions also the order of questions and skip-patterns are important. For example, an answer to a question might influence which subsequent questions are asked or left out. Additionally, answers to one question might influence the answer options of a subsequent question. On-line surveys allow to use these complicated dependencies between questions without the respondent noticing; this makes the task of filling in the questionnaire much easier. While this clearly reduces the complexity of the questionnaire for the user, it increases the complexity for the questionnaire designer. Here the designer needs to have a clear definition of all the paths that are possible in the course of the questionnaire. Branching conditions that describe when questions are to be presented or skipped have to be carefully defined and tested.

In [17] it is remarked that coverage error or bias will occur because people without Internet access are not addressed. We think that coverage error might be very low in our case, because all software engineering professionals have Internet access nowadays. To make sure that the coverage error is low, it has to be checked afterwards whether the respondents come from different contexts, and whether all relevant types of companies are covered.

4.2 Implementing on-line surveys

As stated before, the implementation step in the survey process is particularly affected by the decision to perform a survey on-line. Here the entire task to bring the questionnaire onto the web has to be tackled. In the following we discuss how to build or select an appropriate development environment to implement the questionnaire into a web-based questionnaire. Then, we give practical guidelines and lessons learned that we distilled from our experience with five on-line surveys.

4.2.1 Defining a development environment

The development environment has to support the researcher in bringing the web pages and related scripts on-line. Two approaches are possible here. The first option is to program the HTML-pages and scripts yourself. The second option is to use a commercially available tool for on-line surveys.

Self-development - For the Inspection1 and Inspection2 survey, we decided at first to implement a tool of our own. We decided this for several reasons; mainly, because, at that time, we were not able to raise funding for using a commercially available tool. Instead we had a student available who implemented an online questionnaire using HTML and PHP. In the beginning the implementation proceeded well, and we had working prototypes of our tool, which had a graphical layout of the questionnaire in a sidebar so that respondents could easily jump back to earlier questions. In addition, we implemented a feature that allowed respondents to temporarily stop answering the questionnaire and continue later with the questions where they had stopped before (e.g., because they wanted to take a break or their browser crashed). However, it turned out in the course of the implementation that the questionnaire for the surveys was too complex to be implemented in that way within a reasonable time. Additionally, we followed an incremental process. That is, there were several changes made to the questionnaire itself while it was being implemented, in addition to bugs that had to be removed from the implementation. These changes turned out to be unmanageable with the student's implementation. In the end, we decided to give up the own tool and instead to use a commercial product for implementing the questionnaire, because funds were raised to buy such a product.

Commercial Tools - Commercially available tools for on-line surveys provide the opportunity to enter the questionnaire into the tool and then to generate the necessary web pages and scripts. The GlobalPark tool that we used offers a wide range of functionality. When entering questions, the user can select from a large range of available answer categories such as measurement scales (e.g., Likert-scales, semantic differentials) or free text. For each question it can be defined whether it is mandatory. In this case the tool will prompt the respondents if they do not answer this question. For each question conditions can be defined when to present or skip the question, so that skip patterns can be implemented. In addition to defining the questionnaire, the tool had components for data management and respondent management. The data management component allowed the respondents to stop entering data and resume the entry at a later time. We had the possibility to view the results in real-time and also to download them in a spreadsheet format. The user management allowed us to enter the e-mail addresses of the persons in the sample, allowed to send invitation mails to all or single persons, and allowed to send reminder mails to persons having not started the survey. For self-recruited surveys the tool provides pop-up window facilities to catch visitor's attention for the survey when visiting the web site.

4.2.2 Development guidelines

Since there is only little methodological support on how to perform on-line surveys in software engineering, we present here guidelines and lessons learned that should be taken into account when defining an optimal web-based questionnaire.

An important fact to have in mind while designing and implementing an online survey is that the respondents are free to decide whether they want to participate or not. Therefore it is of utmost importance to **motivate** them to spend their time on the survey. One of the strongest motivators is when the respondents feel that they get something back for their invested time. The return can come in many forms: in our survey Inspections1 (see Section 4.2.2) the respondents could obtain a detailed analysis report afterwards so that they could compare their state of the practice against the population state of the practice. In our Portal-survey, books were raffled in order to acknowledge the respondents time. This, however has to be done carefully, in order to prevent that people are doing the survey because of the reward [18]. However, we also had many participants in studies when there was no tangible reward. Thus, also intangible rewards such as the feeling to contribute to an interesting study or to broaden one's mind by learning the questions asked.

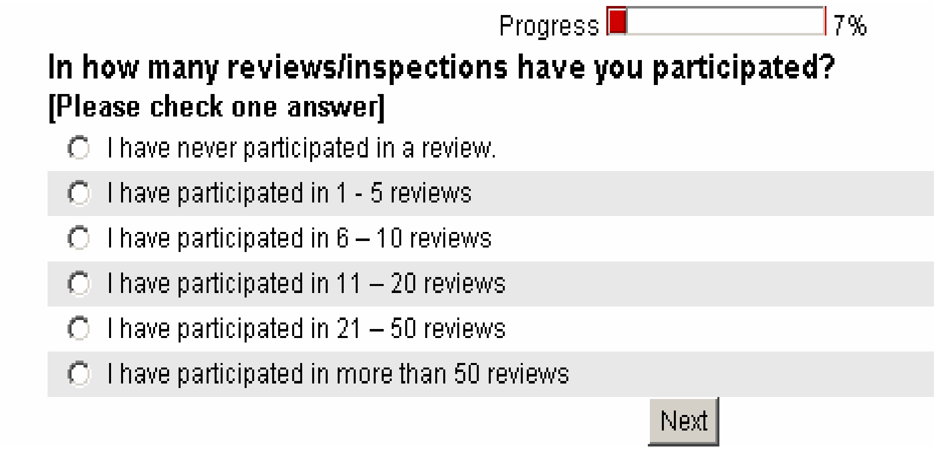
The **amount of questions** should be carefully considered. The more questions are asked the more time does it take to complete it. If the respondents consider this time as too high they will not start answering. Thus, as a basic rule, the questionnaire should not get too long. One way to achieve this is to restrict questions to the specific topic that has to be addressed according to specified survey goal. However, this is not as easy as it sounds. It is always tempting to ask more questions, as the information gained through them may be useful. Nevertheless, the risk is that these questions cannot be analyzed, as they broaden the scope of the analysis too much. Additionally some additional personal questions should be added to characterize the sample.

Unfortunately, it is difficult to define an appropriate number of questions for an on-line survey. For example, in one of our surveys we had a questionnaire with more than 90 questions (survey Inspections 1 and 2) that required a large amount of time of the respondents to fill in. People still answered, because they felt they were learning interesting things when answering the questionnaire. Table 3 shows in columns 6 and 7 the number of questions and the resulting effort needed to answer all questions. These data form a baseline that will help to plan the time required by the respondents for future surveys.

The **type of questions** should also be selected with respect to the average web page. Although it is tempting to apply mainly closed questions – to facilitate easy analysis – we nevertheless intentionally tried to integrate open questions (with empty text fields) anyway because they can give important background information and support the interpretation of the quantitative results of closed questions.

The **layout** of the web pages is also important, because respondents might stop when finding it hard to read the questionnaire or when they experience technical difficulties. One rule is that the information displayed has to fit on the screen without scrolling. Thus, all questions have to be fully visible on the screen. This is also true when displaying the answer alternatives. If the number of answer alternatives is too large to be displayed on one screen, pull-down menus or double-banking (i.e., using more than one column) should be considered. Pull-down menus are particularly attractive when a large number of answer alternatives are to be presented.

Another aspect of layout is to convey a sense of where the respondent is in the completion process – e.g., a progress indicator like the bar depicted in figure 1 – so that the respondent gets a feeling of progressing in the questionnaire and does not stop because the questions are apparently not ending. However, as progress indicator mechanisms require computer sources on the web server, they should not slow down the time to generate a page.



In how many reviews/inspections have you participated?
[Please check one answer]

○ I have never participated in a review.
○ I have participated in 1 - 5 reviews
○ I have participated in 6 – 10 reviews
○ I have participated in 11 – 20 reviews
○ I have participated in 21 – 50 reviews
○ I have participated in more than 50 reviews

Figure 1 Example of the lay-out of an on-line survey

In order to support the **readability,** questions and answer alternatives should be presented in a format similar to those normally used on paper self-administered questionnaires, (e.g., check-boxes or Likert scales), because the respondents are used to these formats. In addition, color usage will influence the readability significantly. In [19] it is suggested to restrain the use of color so that figure and ground consistency are maintained. Finally, the readability of questions should be ensured by avoiding differences in the visual appearance that result from different screen configurations, operating systems, browsers, partial screens and wrap-around text. This is of special importance for software professionals as they often tend to work with off-mainstream computer configurations like different browsers or high security settings. For example, we tested our surveys with different browsers, from Netscape and Mozilla to Internet Explorer and Opera.

The **order of questions** should maintain the respondent’s motivation. In our surveys we paid attention to define interesting first questions that should capture the respondents interest and attention. In particular, it is crucial not to start by asking any questions that may appear non-anonymous, such as the participant’s e-mail or name. Questions with personal characteristics were put at the end of the questionnaire in order not to startle people.

When using commercial tools it is tempting to have one question per page. However, in order to maintain a smooth flow of questions and topics, questions should be grouped according to topics and each topic should be introduced with a small introduction. This is especially important as the of web pages follow one by one does not provide the participant a feeling of what is coming next.

In order to support the respondent in answering the questionnaire, **specific instructions** should be provided on how to take each necessary computer action to respond to the questionnaire. In our surveys we introduced the web questionnaire with a welcome screen that emphasized the ease of responding and instructed respondents on the action needed for proceeding to the next page. We also provided skip directions in a way that it encouraged respondents to mark answers and be able to click to the next applicable question or even automatic integration.

Do **not require answering** of respondents to each question before being allowed to answer any subsequent ones. Although it is possible in on-line surveys to make answering (specific) questions mandatory before proceeding further, this will likely impact the motivation of respondents negatively. Only for questions where the answers are necessary – important for research goal, or needed for navigation – it can be considered to make these mandatory to answer.

For the **recovery of the questionnaire,** there should be support for several "exceptional" situations, e.g., when the browser crashes. That is, participants of online surveys should have the possibility to quit and restart the survey at the last answered question. In addition, it would be helpful if it was possible to easily jump to previous questions to review or change the answer given there. Such situations are hard to test, because they usually occur non-deterministically. Thus, it is also an issue of testing the web-questionnaire, not only of designing it. As testing such a tool is hard, this is another reason for using commercial tools instead of building a tool on your own. However, one basic rule that has to be considered during design is that *if* the browser crashes, the respondents will not start answering the questionnaire from the beginning. Therefore, the tool should provide a possibility for the respondents to resume the questionnaire at the question where he or she last stopped; for example, through using cookies or personalized login access.

Ensure anonymity of the participants who give answers. As you might ask questions that can be personal or concerning information on the company needed to characterize the respondent, make sure that the results collected in the survey are not connected with the name and the email addresses of the participants.

4.3 Executing on-line surveys

Executing on-line surveys consists of inviting all persons in the sample and waiting for the respondents to fill the questionnaire in. One of the advantages of an on-line survey is that an automatic tracking of the responses can be performed in real-time. Thus it is possible to monitor and control the participation in the survey. Two aspects are particularly beneficial, namely the control of non-respondents and control of drop-out questions.

4.3.1 Control of non-respondents

As it its possible to identify in personalized on-line surveys those people that have not started with the questionnaire (so called non-respondents) it is possible to specifically invite these people with a second invitation mail. We made very good experiences with this feature as this significantly increased the overall response rate. Figure 2 shows the number of respondents per day for one of our surveys. It can be seen that many respondents answer within a short time after the invitation is sent. After that the number of participants decreases. A reminder raises the awareness of the survey and thus increased in our surveys the response rate significantly.

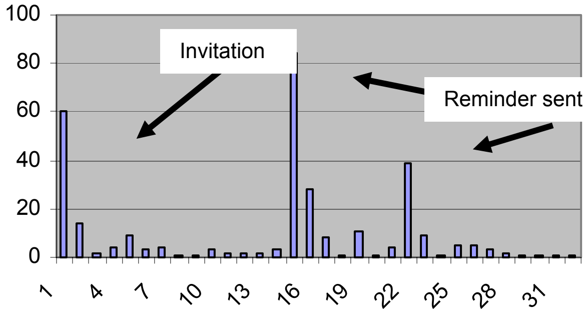


Figure 2 Number of respondents (Y-axis) per day (X-axis) for survey Inspections1

4.3.2 Drop-out questions

As for drop out questions it is also possible to track which question causes the most drop outs (i.e., questions where people stop filling in the questionnaire). Once such a question is identified it is possible to analyze the question and improve it in order to encourage respondents to continue beyond this question and raise the response rate. In one of our surveys, we analyzed the dropout rate after the survey was operational for a few days.

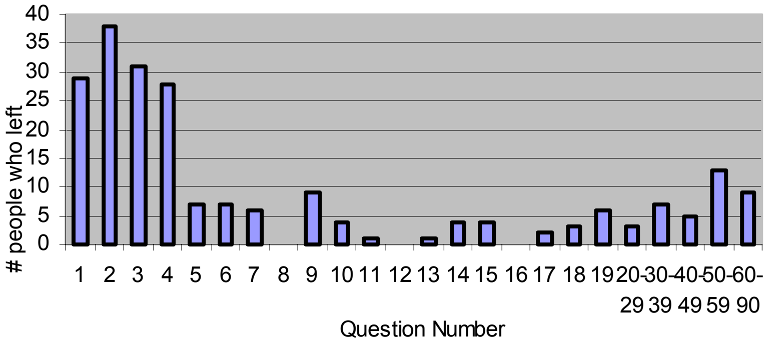


Figure 3 Dropout statistics with dropouts per question number (for survey Inspections1).

The dropout statistics as shown in Figure 3 taught us, that there was a high dropout on the second page of the survey. This was a page with two questions and a rather long introductory text. We split the page into two pages with one question and shortened the introductory text. With the tool we used, this was possible without compromising the results of the survey. Changes, however, must be done carefully because the questionnaire is in field already, and changes may bias the result.

4.4 Analyzing on-line surveys

One feature of on-line survey tools is that data are stored directly after they have been entered. Thus it is possible to get a real-time impression of how respondents answer. These preliminary data allow to devise ideas for presenting the data in tables and figures. Another usage of this feature is to forecast the survey outcome and to draw preliminary conclusions. The advantage of this feature is that the researcher can get an early impression of how the end result will look like (at least earlier than in paper-based surveys). This, however, should be done carefully as the results might change significantly in the remainder of the survey.

5 Conclusions

The purpose of this paper is to raise the awareness for on-line survey amongst software engineering researchers. Surveys play an important role as a research strategy as they allow us to learn about the state of the practice, identify improvement potentials, or investigate the acceptance of a technology. With the increasing pervasion of the Internet it is now possible to perform surveys easily and cost-effectively over Internet pages (i.e., on-line). Our paper contributes by presenting a process of how to perform on-line surveys and by sharing our experience we gained in five on-line surveys in the form of guidelines.

The challenge for the empirical research community is now twofold. The first challenge is to perform such surveys more frequently. For example, if we manage to motivate the persons that were involved in our first surveys to take part in future surveys we would have a direct link to the state of the practice. This knowledge would be beneficial in terms of finding fruitful research areas. A second challenge is to make the results of surveys available so that many people can benefit from this. We try to contribute to this challenge by performing our surveys in the context of the ViSEK project and publishing our results in that portal, in addition to scientific publications.

Acknowledgements

We acknowledge the German BMBF-funded ViSEK-project for the possibility of conducting the on-line surveys. Furthermore, we want to thank all respondents of the surveys that contributed to our research.

References

[1] I. Sommerville, Software engineering, 5th edition, Wokingham, Addison-Wesley Publishing company, 1996.

[2] C. Seaman, Qualitative Methods in Empirical Studies of Software Engineering, IEEE Transactions on Software Engineering, pp. 557—572, 1999.

[3] N. Fenton S. Pfleeger, Software metrics – a rigorous approach, Chapman and Hall, 1996.

[4] N. Juristo, A. Moreno, Basics of Software Engineering Experimentation, Kluwer Academic Publishers, Boston, 2001.

[5] C. Wohlin, P. Runeson, M. Höst, M.C. Ohlsson, B. Regnell, A. Wesslen, Experimentation in Software Engineering - An Introduction. The Kluwer International Series in Software Engineering, Kluwer Academic Publishers, 2000.

[6] B. Freimut, T. Punter, S. Biffl, M. Ciolkowski, State of the art in Empirical Studies, ViSEK Report 007/02 / IESE Report , Kaiserslautern, Fraunhofer IESE, March 2002.

[7] A. Zendler, A Preliminary Software Engineering Theory as Investigated by Published Experiments. Empirical Software Engineering: An International Journal 6: 161-180, 2001.

[8] C. Wohlin, T. Punter, Survey of Experimental Methods in Software Engineering, D2.1 of IST 2000-28754 Project ESERNET, December 2001.

[9] B. Kitchenham, S. Pfleeger, Principles of Survey Research – series of 6 articles, in: ACM Sigsoft Software Engineering Notes, Vol 26 (No.6), Vol 27 (No. 1,3,5), Vol 28 (No.2), 2001-2003.

[10] P. Salant, D. Dillman, How to conduct your own survey?, New York, John Wiley and Sons, 1994.

[11] F. Fowler, Survey research methods, Newbure Park, Sage, 1988.

[12] J. Creswell, Research Design – Qualitative and quantitative approaches, Thousand Oaks, Sage Publications, 1993.

[13] The Experimental Paradigm in Software Engineering, in: Collection of Position Papers of International Workshop on Experimental Software Engineering Issues, Dagstuhl Castle, September 14-18, 1992.

[14] D. Shannon, T. Johnson, S. Searcy, A. Lott, Using electronic surveys: advice from survey professionals, in: Practical Assessment, Research & Evaluation, 8 (1), http://ericae.net/pare.

[15] A. Oppenheim, Questionnaire design, interviewing and attitude measurement, London, Pinter, 1992.

[16] F. Fowler, Improving survey questions - design and evaluation, Sage Publications, Thousand Oaks, 1995.

[17] D. Solomon, Conducting web-based surveys, In Practical Assessment Research and Evaluation, 7(19), ISSN-1531-7714, http://ericae.net/pare.

[18] L. Prechelt, Kontrollierte Experimente in der Softwaretechnik: Potenzial und Methodik. Springer Verlag, 2001.

[19] D. Dillman, D. Bowker, The Web Questionnaire Challenge to Survey Methodologists, in: U-D Reips & M. Bosnjak (eds), Dimensions of Internet Science, Lengerich, Pabst Science Publishers, 2001.

Document Information

Title:	Conducting On-line Surveys in Software Engineering
Date:	July 2003
Report:	IESE-058.03/E
Status:	Final
Distribution:	Public