How to Conduct a Heuristic Evaluation

by Jakob Nielsen

Heuristic evaluation (Nielsen and Molich, 1990; Nielsen 1994) is a usability engineering method for finding the usability problems in a user interface design so that they can be attended to as part of an iterative design process. Heuristic evaluation involves having a small set of evaluators examine the interface and judge its compliance with recognized usability principles (the "heuristics").

In general, heuristic evaluation is difficult for a single individual to do because one person will never be able to find all the usability problems in an interface. Luckily, experience from many different projects has shown that different people find different usability problems. Therefore, it is possible to improve the effectiveness of the method significantly by involving multiple evaluators. Figure 1 shows an example from a case study of heuristic evaluation where 19 evaluators were used to find 16 usability problems in a voice response system allowing customers access to their bank accounts (Nielsen 1992). Each of the black squares in Figure 1 indicates the finding of one of the usability problems by one of the evaluators. The figure clearly shows that there is a substantial amount of nonoverlap between the sets of usability problems found by different evaluators. It is certainly true that some usability problems are so easy to find that they are found by almost everybody, but there are also some problems that are found by very few evaluators. Furthermore, one cannot just identify the best evaluator and rely solely on that person's findings. First, it is not necessarily true that the same person will be the best evaluator every time. Second, some of the hardest-to-find usability problems (represented by the leftmost columns in Figure 1) are found by evaluators who do not otherwise find many usability problems. Therefore, it is necessary to involve multiple evaluators in any heuristic evaluation (see below for a discussion of the best number of evaluators). My recommendation is normally to use three to five evaluators since one does not gain that much additional information by using larger numbers.

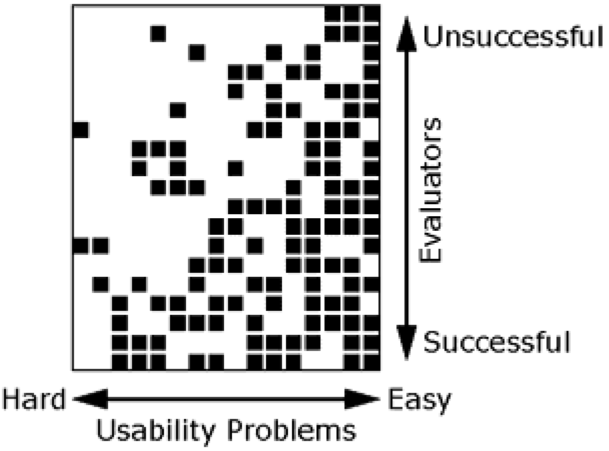

Figure 1
Illustration showing which evaluators found which usability problems in a heuristic evaluation of a banking system. Each row represents one of the 19 evaluators and each column represents one of the 16 usability problems. Each square shows whether the evaluator represented by the row found the usability problem represented by the column: The square is black if this is the case and white if the evaluator did not find the problem. The rows have been sorted in such a way that the most successful evaluators are at the bottom and the least successful are at the top. The columns have been sorted in such a way that the usability problems that are the easiest to find are to the right and the usability problems that are the most difficult to find are to the left.

Heuristic evaluation is performed by having each individual evaluator inspect the interface alone. Only after all evaluations have been completed are the evaluators allowed to communicate and have their findings aggregated. This procedure is important in order to ensure independent and unbiased evaluations from each evaluator. The results of the evaluation can be recorded either as written reports from each evaluator or by having the evaluators verbalize their comments to an observer as they go through the interface. Written reports have the advantage of presenting a formal record of the evaluation, but require an additional effort by the evaluators and the need to be read and aggregated by an evaluation manager. Using an observer adds to the overhead of each evaluation session, but reduces the workload on the evaluators. Also, the results of the evaluation are available fairly soon after the last evaluation session since the observer only needs to understand and organize one set of personal notes, not a set of reports written by others. Furthermore, the observer can assist the evaluators in operating the interface in case of problems, such as an unstable prototype, and help if the evaluators have limited domain expertise and need to have certain aspects of the interface explained.

In a user test situation, the observer (normally called the "experimenter") has the responsibility of interpreting the user's actions in order to infer how these actions are related to the usability issues in the design of the interface. This makes it possible to conduct user testing even if the users do not know anything about user interface design. In contrast, the responsibility for analyzing the user interface is placed with the evaluator in a heuristic evaluation session, so a possible observer only needs to record the evaluator's comments about the interface, but does not need to interpret the evaluator's actions.

Two further differences between heuristic evaluation sessions and traditional user testing are the willingness of the observer to answer questions from the evaluators during the session and the extent to which the evaluators can be provided with hints on using the interface. For traditional user testing, one normally wants to discover the mistakes users make when using the interface; the experimenters are therefore reluctant to provide more help than absolutely necessary. Also, users are requested to discover the answers to their questions by using the system rather than by having them answered by the experimenter. For the heuristic evaluation of a domain-specific application, it would be unreasonable to refuse to answer the evaluators' questions about the domain, especially if nondomain experts are serving as the evaluators. On the contrary, answering the evaluators' questions will enable them to better assess the usability of the user interface with respect to the characteristics of the domain. Similarly, when evaluators have problems using the interface, they can be given hints on how to proceed in order not to waste precious evaluation time struggling with the mechanics of the interface. It is important to note, however, that the evaluators should not be given help until they are clearly in trouble and have commented on the usability problem in question.

Typically, a heuristic evaluation session for an individual evaluator lasts one or two hours. Longer evaluation sessions might be necessary for larger or very complicated interfaces with a substantial number of dialogue elements, but it would be better to split up the evaluation into several smaller sessions, each concentrating on a part of the interface.

During the evaluation session, the evaluator goes through the interface several times and inspects the various dialogue elements and compares them with a list of recognized usability principles (the heuristics). These heuristics are general rules that seem to describe common properties of usable interfaces. In addition to the checklist of general heuristics to be considered for all dialogue elements, the evaluator obviously is also allowed to consider any additional usability principles or results that come to mind that may be relevant for any specific dialogue element. Furthermore, it is possible to develop category-specific heuristics that apply to a specific class of products as a supplement to the general heuristics. One way of building a supplementary list of category-specific heuristics is to perform competitive analysis and user testing of existing products in the given category and try to abstract principles to explain the usability problems that are found (Dykstra 1993).

In principle, the evaluators decide on their own how they want to proceed with evaluating the interface. A general recommendation would be that they go through the interface at least twice, however. The first pass would be intended to get a feel for the flow of the interaction and the general scope of the system. The second pass then allows the evaluator to focus on specific interface elements while knowing how they fit into the larger whole.

Since the evaluators are not *using* the system as such (to perform a real task), it is possible to perform heuristic evaluation of user interfaces that exist on paper only and have not yet been implemented (Nielsen 1990). This makes heuristic evaluation suited for use early in the usability engineering lifecycle.

If the system is intended as a walk-up-and-use interface for the general population or if the evaluators are domain experts, it will be possible to let the evaluators use the system without further assistance. If the system is domain-dependent and the evaluators are fairly naive with respect to the domain of the system, it will be necessary to assist the evaluators to enable them to use the interface. One approach that has been applied successfully is to supply the evaluators with a typical usage scenario, listing the various steps a user would take to perform a sample set of realistic tasks. Such a scenario should be constructed on the basis of a task analysis of the actual users and their work in order to be as representative as possible of the eventual use of the system.

The output from using the heuristic evaluation method is a list of usability problems in the interface with references to those usability principles that were violated by the design in each case in the opinion of the evaluator. It is not sufficient for evaluators to simply say that they do not like something; they should explain why they do not like it with reference to the heuristics or to other usability results. The evaluators should try to be as specific as possible and should list each usability problem separately. For example, if there are three things wrong with a certain dialogue element, all three should be listed with reference to the various usability principles that explain why each particular aspect of the interface element is a usability problem. There are two main reasons to note each problem separately: First, there is a risk of repeating some problematic aspect of a dialogue element, even if it were to be completely replaced with a new design, unless one is aware of all its problems. Second, it may not be possible to fix all usability problems in an interface element or to replace it with a new design, but it could still be possible to fix some of the problems if they are all known.

Heuristic evaluation does not provide a systematic way to generate fixes to the usability problems or a way to assess the probable quality of any redesigns. However, because heuristic evaluation aims at explaining each observed usability problem with reference to established usability principles, it will often be fairly easy to generate a revised design according to the guidelines provided by the violated principle for good interactive systems. Also, many usability problems have fairly obvious fixes as soon as they have been identified.

For example, if the problem is that the user cannot copy information from one window to another, then the solution is obviously to include such a copy feature. Similarly, if the problem is the use of inconsistent typography in the form of upper/lower case formats and fonts, the solution is obviously to pick a single typographical format for the entire interface. Even for these simple examples, however, the designer has no information to help design the exact changes to the interface (e.g., how to enable the user to make the copies or on which of the two font formats to standardize).

One possibility for extending the heuristic evaluation method to provide some design advice is to conduct a debriefing session after the last evaluation session. The participants in the debriefing should include the evaluators, any observer used during the evaluation sessions, and representatives of the design team. The debriefing session would be conducted primarily in a brainstorming mode and would focus on discussions of possible redesigns to address the major usability problems and general problematic aspects of the design. A debriefing is also a good opportunity for discussing the positive aspects of the design, since heuristic evaluation does not otherwise address this important issue.

Heuristic evaluation is explicitly intended as a "discount usability engineering" method. Independent research (Jeffries et al. 1991) has indeed confirmed that heuristic evaluation is a very efficient usability engineering method. One of my case studies found a benefit-cost ratio for a heuristic evaluation project of 48: The cost of using the method was about $10,500 and the expected benefits were about $500,000 (Nielsen 1994). As a discount usability engineering method, heuristic evaluation is not guaranteed to provide "perfect" results or to find every last usability problem in an interface.

Determining the Number of Evaluators

In principle, individual evaluators can perform a heuristic evaluation of a user interface on their own, but the experience from several projects indicates that fairly poor results are achieved when relying on single evaluators. Averaged over six of my projects, single evaluators found only 35 percent of the usability problems in the interfaces. However, since different evaluators tend to find different problems, it is possible to achieve substantially better performance by aggregating the evaluations from several evaluators. Figure 2 shows the proportion of usability problems found as more and more evaluators are added. The figure clearly shows that there is a nice payoff from using more than one evaluator. It would seem reasonable to recommend the use of about five evaluators, but certainly at least three. The exact number of evaluators to use would depend on a cost-benefit analysis. More evaluators should obviously be used in cases where usability is critical or when large payoffs can be expected due to extensive or mission-critical use of a system.

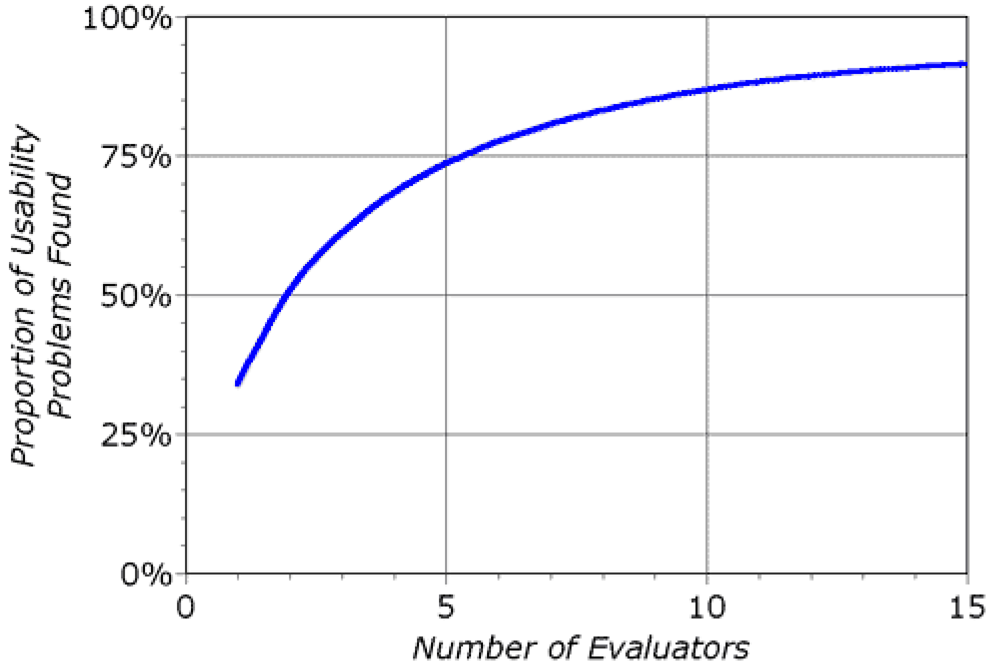

Figure 2

Curve showing the proportion of usability problems in an interface found by heuristic evaluation using various numbers of evaluators. The curve represents the average of six case studies of heuristic evaluation.

Nielsen and Landauer (1993) present such a model based on the following prediction formula for the number of usability problems found in a heuristic evaluation:

$$\texttt{ProblemsFound}(i) = N(1 - (1-l)^{i})$$

where $\texttt{ProblemsFound}(i)$ indicates the number of different usability problems found by aggregating reports from i independent evaluators, N indicates the total number of usability problems in the interface, and l indicates the proportion of all usability problems found by a single evaluator. In six case studies (Nielsen and Landauer 1993), the values of l ranged from 19 percent to 51 percent with a mean of 34 percent. The values of N ranged from 16 to 50 with a mean of 33. Using this formula results in curves very much like that shown in Figure 2, though the exact shape of the curve will vary with the values of the parameters N and l, which again will vary with the characteristics of the project.

In order to determine the optimal number of evaluators, one needs a cost-benefit model of heuristic evaluation. The first element in such a model is an accounting for the cost of using the method, considering both fixed and variable costs. Fixed costs are those that need to be paid no matter how many evaluators are used; these include time to plan the evaluation, get the materials ready, and write up the report or otherwise communicate the results. Variable costs are those additional costs that accrue each time one additional evaluator is used; they include the loaded salary of that evaluator as well as the cost of analyzing the evaluator's report and the cost of any computer or other resources used during the evaluation session. Based on published values from several projects the fixed cost of a heuristic evaluation is estimated to be between $3,700 and $4,800 and the variable cost of each evaluator is estimated to be between $410 and $900.

The actual fixed and variable costs will obviously vary from project to project and will depend on each company's cost structure and on the complexity of the interface being evaluated. For illustration, consider a sample project with fixed costs for heuristic evaluation of $4,000 and variable costs of $600 per evaluator. In this project, the cost of using heuristic evaluation with i evaluators is thus $\$(4{,}000 + 600i)$.

The benefits from heuristic evaluation are mainly due to the finding of usability problems, though some continuing education benefits may be realized to the extent that the evaluators increase their understanding of usability by comparing their own evaluation reports with those of other evaluators. For this sample project, assume that it is worth $15,000 to find each usability problem, using a value derived by Nielsen and Landauer (1993) from several published studies. For real projects, one would obviously need to estimate the value of finding usability problems based on the expected user population. For software to be used in-house, this value can be estimated based on the expected increase in user productivity; for software to be sold on the open market, it can be estimated based on the expected increase in sales due to higher user satisfaction or better review ratings. Note that real value only derives from those usability problems that are in fact fixed before the software ships. Since it is impossible to fix all usability problems, the value of each problem found is only some proportion of the value of a fixed problem.

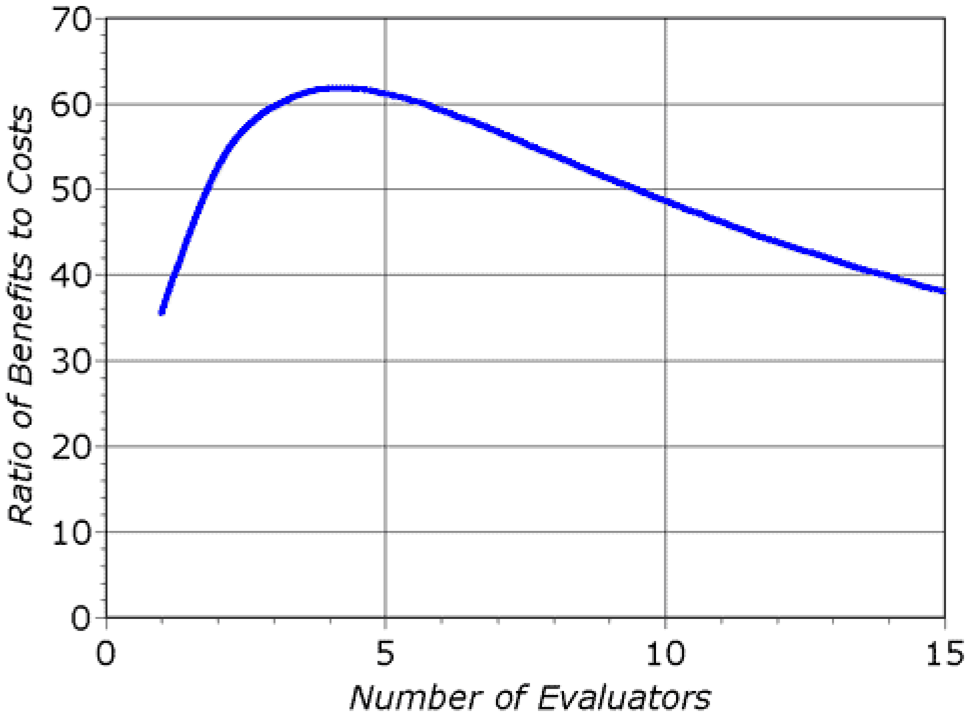

Figure 3
Curve showing how many times the benefits are greater than the costs for heuristic evaluation of a sample project using the assumptions discussed in the text. The optimal number of evaluators in this example is four, with benefits that are 62 times greater than the costs.

Figure 3 shows the varying ratio of the benefits to the costs for various numbers of evaluators in the sample project. The curve shows that the optimal number of evaluators in this example is four, confirming the general observation that heuristic evaluation seems to work best with three to five evaluators. In the example, a heuristic evaluation with four evaluators would cost $6,400 and would find usability problems worth $395,000.

References

Dykstra, D. J. 1993. *A Comparison of Heuristic Evaluation and Usability Testing: The Efficacy of a Domain-Specific Heuristic Checklist*. Ph.D. diss., Department of Industrial Engineering, Texas A&M University, College Station, TX.

Jeffries, R., Miller, J. R., Wharton, C., and Uyeda, K. M. 1991. User interface evaluation in the real world: A comparison of four techniques. *Proceedings ACM CHI'91 Conference* (New Orleans, LA, April 28-May 2), 119-124.

Molich, R., and Nielsen, J. (1990). Improving a human-computer dialogue, *Communications of the ACM* **33**, 3 (March), 338-348.

Nielsen, J. 1990. Paper versus computer implementations as mockup scenarios for heuristic evaluation. *Proc. IFIP INTERACT90 Third Intl. Conf. Human-Computer Interaction* (Cambridge, U.K., August 27-31), 315-320.

Nielsen, J., and Landauer, T. K. 1993. A mathematical model of the finding of usability problems. *Proceedings ACM/IFIP INTERCHI'93 Conference* (Amsterdam, The Netherlands, April 24-29), 206-213.

Nielsen, J., and Molich, R. (1990). Heuristic evaluation of user interfaces, *Proc. ACM CHI'90 Conf.* (Seattle, WA, 1-5 April), 249-256.

Nielsen, J. 1992. Finding usability problems through heuristic evaluation. *Proceedings ACM CHI'92 Conference* (Monterey, CA, May 3-7), 373-380.

Nielsen, J. (1994). Heuristic evaluation. In Nielsen, J., and Mack, R.L. (Eds.), *Usability Inspection Methods*. John Wiley & Sons, New York, NY.