Available online at www.sciencedirect.com

SCIENCE DIRECT®

Journal of Biomedical Informatics 36 (2003) 23–30

Journal of Biomedical Informatics

www.elsevier.com/locate/yjbin

Using usability heuristics to evaluate patient safety of medical devices

Jiajie Zhang,[a,*] Todd R. Johnson,[a] Vimla L. Patel,[b] Danielle L. Paige,[c] and Tate Kubose[b]

[a] *School of Health Information Sciences, University of Texas Health Science Center at Houston, 7000 Fannin, Suite 600, Houston, TX 77030, USA*
[b] *Department of Biomedical Informatics, Columbia University, Vanderbilt Clinic, 5th Floor 622 West 168th Street, New York, NY 10032, USA*
[c] *Department of Psychology, Rice University, Texas, USA*

Received 1 July 2003

Abstract

Objective. To modify the traditional heuristic evaluation method of assessing software usability so that it can be applied to medical devices and used to evaluate the patient safety of those devices through the identification and assessment of usability problems.

Design. Heuristic evaluation, a usability inspection method commonly used for software usability evaluation, was modified and extended for medical devices. The modified method was used to evaluate and compare the patient safety of two 1-channel volumetric infusion pumps.

Results. The modified heuristic evaluation method was successfully applied to medical devices. One hundred and ninety-two heuristic violations were categorized for 89 usability problems identified for Pump 1, and 121 heuristic violations were categorized for the 52 usability problems identified for Pump 2. Pump 1 had more usability problems with high severity ratings than Pump 2. In general, Pump 1 was found to have more usability issues that are likely to induce more medical errors.

Conclusions. Heuristic evaluation, when modified for medical devices, is a useful, efficient, and low cost method for evaluating patient safety features of medical devices through the identification of usability problems and their severities.

Keywords: Medical error; Patient safety; Heuristic evaluation; Usability; Medical device; Infusion pump

1. Introduction

The medical error report from the Institute of Medicine [1] has greatly increased people's awareness of the frequency, magnitude, complexity, and seriousness of medical errors. As the eighth leading cause of death in the US, ahead of motor vehicle accidents, breast cancer, or AIDS, medical errors occur in many medical situations. One such situation is the use of medical devices. Medical device use errors are a common source of patient injury and death. In many cases, medical devices have user interfaces that are so poorly designed and difficult to use that they invite a variety of human errors. FDA data collected between 1985 and 1989 demonstrated that 45–50% of all device recalls stemmed from poor product design (including problems with software) [2,3]. Furthermore, the FDA recognizes that a poorly designed user interface can induce errors and operating inefficiencies even when operated by a well-trained, competent user. In response, the FDA has revised its Good Manufacturing Practice regulations to include specific requirements for product usability [2]. They have also published guidelines for interface design and usability testing [3] and produced a continuing education article that specifically covers usability issues [4]. Other research suggests that injuries resulting from medical device use errors far exceeds injuries arising from device failures [5].

In this paper, we modify a usability engineering technique called heuristic evaluation for the evaluation of usability problems in medical devices. Through the identification of usability problems, we can indirectly identify medical devices' potential trouble spots that are

* Corresponding author. Fax: +713-500-3929.
E-mail addresses: jiajie.zhang@uth.tmc.edu (J. Zhang), todd.r.johnson@uth.tmc.edu (T.R. Johnson), patel@dmi.columbia.edu (V.L. Patel).

1532-0464/$ - see front matter © 2003 Elsevier Inc. All rights reserved.
doi:10.1016/S1532-0464(03)00060-1

likely to cause medical errors. In Section 2 we describe the usability heuristics modified for the evaluation of medical devices, the scale for severity rating of usability problems, and the procedure of carrying out a heuristic evaluation. In Section 3 we select two 1-channel volumetric infusion pumps and perform a heuristic evaluation on them. In Section 4 we discuss the implications of heuristic evaluation in the evaluation of patient safety in medical device use.

2. Background

Numerous research reports, medical error reports, and other documents show a clear link between usability problems and user error [6,7]. The FDA's report, Do it By Design [3], describes a variety of errors resulting from medical device interface design. For example, a physician treating an infant with oxygen set the flow knob between 1 and 2 L/min and then later noticed that the infant was not receiving any oxygen. Even though the knob rotated smoothly, the device was designed to deliver oxygen only when the knob was set on a number, not between numbers. Adding detents to the knob, so that it would click onto a number, and providing visible feedback of the rate of flow could have greatly decreased the chance of this type of error. In another example, found in the FDA's manufacturer and user facility device experience database (MAUDE), a nurse tried to program an infusion pump to deliver 130.1 ml/h of a drug, but inadvertently programmed the pump to deliver 1301 ml/h, because the decimal point on the pump was designed to operate for numbers no greater than 99.9. When the nurse pressed "1 3 0 . 1" the device ignored the decimal point key-press. Since simply ignoring the decimal point clearly results in a number that is 10 times larger than intended, this error could be prevented by designing the device to alert the user whenever the decimal point is pressed after more than two digits have been entered. The alert could inform the user of the problem and then force the user to reenter the number.

Human factors engineering is a discipline that seeks to design devices, software, and systems to meet the needs, capabilities, and limitations of the users, rather than expecting the users to adapt to the design. A complete human factors engineering analysis for medical devices or software systems includes four major components: user, functional, task, and representational analyses [8]. User analysis is the process of identifying the characteristics of existing and potential users, such as their expertise and skills, knowledge base, educational background, cognitive capacities and limitations, perceptual variations, age related skills, cultural background, personality, time available for learning and training, frequency of system use, and so on. User analysis can help us design systems that have the right knowledge and information structure that match that of the users. Functional analysis is the process of identifying critical top-level domain structures and goals that are largely independent of implementations. It is more abstract than task and representational analysis because it does not involve details of task processes and representations. Task analysis is the process of identifying system functions that have to be performed, procedures and actions to be carried out to achieve task goals, information to be processed, input and output formats that are required, constraints that must be considered, communication needs that have to be satisfied, and the organization and structure as well as the information categories and information flow of the task. One important function of task analysis is to ensure that only the necessary and sufficient task features that match users' capacities and are required by the task will be included in system implementations. Task analysis can be conducted at different levels of detail. A keystroke-level model lists the sequence of keystrokes and other physical actions required to complete a specific type of task [9,10]. For instance, the task may be programming an infusion pump to deliver 500 ml at 100 ml/h and the keystroke-level model lists the keys or buttons the user must press to complete this task. A cognitive task analysis includes cognitive operations, such as the goal of entering the rate (which may be accomplished physically in different ways, such as by using up–down arrows or typing in the rate), or determining the volume to be infused based on the physician's order and the drug concentration. Representational analysis is the process of identifying an appropriate information display format for a given task performed by a specific type of user such that the interaction between the users and the system is as direct and transparent as possible. With direct interaction interfaces, users can directly, completely, and efficiently engage in the primary tasks they intend to perform, not the housekeeping interface tasks that are barriers between users and systems. The file browser in Microsoft Windows uses a direct interaction interface to move, delete, and rename files, whereas command line systems (e.g., MS DOS) do not.

These four types of analyses, when combined and applied to a single product, can reveal the full range of usability issues, which are essential for an understanding of patient safety implications of the product. Heuristic evaluation, the method we modified and used in the current study, is primarily at the level of representational analysis and is only one of the major techniques at this level. We focus on heuristic evaluation in this paper, because it has been shown to be one of the most cost-effective methods of finding usability problems. We discuss the details of heuristic evaluation in the next section, along with a description of the technique's advantages, limitations, areas of application, and alternative techniques that my be used to augment heuristic evaluation.

3. Heuristic evaluation

Heuristic evaluation is an easy to use, easy to learn, discount usability evaluation technique used to identify major usability problems of a product in a timely manner with reasonable cost [11–14]. This technique requires three or more evaluators to independently apply a set of usability heuristics to a product, identify violations of the heuristics, and assess the severity of each violation.

Heuristic evaluation is a type of usability inspection method, which refers to a class of techniques in which evaluators examine an interface for usability issues. Inspection methods are considered an informal usability evaluation method, because they rely on heuristics and the experience and knowledge of the evaluators. In contrast, empirical techniques assess usability by testing an interface with real users, and formal techniques, such as task analysis, use models and formula to measure usability [12]. Formal methods are often difficult to use, so the most common usability evaluations are inspection and empirical methods.

During a heuristic evaluation, experts walk through the interface and identify elements that violate usability heuristics. This method has become extremely popular in the realm of usability evaluation due to its low cost, low time commitment, and ease of application [13]. Evaluators can conduct the evaluation in a few hours with minimal training. This method has been traditionally used to evaluate websites as well as desktop software applications, and it is typically used to point out software interface difficulties to be addressed in the design process. It can be applied to paper or electronic mock-ups or prototypes as well as completely implemented designs.

In this paper, we modify the heuristic evaluation method to address three issues in the evaluation of medical devices. First, we use it to discover usability problems that are likely to cause medical errors. Second, the "discount" nature of heuristic evaluation may also prove useful for the comparison of patient safety features of alternative medical devices, as is the case of the purchasing process of medical devices. Third, heuristic evaluation may also be a good tool for medical device manufacturers to improve the patient safety features of their products during the design and redesign processes.

3.1. Fourteen usability heuristics

Nielsen [13], as the major researcher who developed the technique of heuristic evaluation, described 10 major heuristics that should be followed by good user interface design. Shneiderman [15] also described eight golden rules that all good user interface designs should follow. Based on the ten heuristics by Nielsen, the eight golden rules by Shneiderman, and our own considerations, we state the following 14 heuristics with semantic tags (words in the brackets), names, general descriptions, and specific information about the heuristics. We call these 14 heuristics the Nielsen–Shneiderman Heuristics because these heuristics are mostly based on their work.

1. [*Consistency*] *Consistency and standards.* Users should not have to wonder whether different words, situations, or actions mean the same thing. Standards and conventions in product design should be followed.
   a. Sequences of actions (skill acquisition).
   b. Color (categorization).
   c. Layout and position (spatial consistency).
   d. Font, capitalization (levels of organization).
   e. Terminology (delete, del, remove, rm) and language (words, phrases).
   f. Standards (e.g., <u>blue underlined text</u> for unvisited hyperlinks).
2. [*Visibility*] *Visibility of system state.* Users should be informed about what is going on with the system through appropriate feedback and display of information.
   a. What is the current state of the system?
   b. What can be done at current state?
   c. Where can users go?
   d. What change is made after an action?
3. [*Match*] *Match between system and world.* The image of the system perceived by users should match the model the users have about the system.
   a. User model matches system image.
   b. Actions provided by the system should match actions performed by users.
   c. Objects on the system should match objects of the task.
4. [*Minimalist*] *Minimalist.* Any extraneous information is a distraction and a slow-down.
   a. Less is more.
   b. Simple is not equivalent to abstract and general.
   c. Simple is efficient.
   d. Progressive levels of detail.
5. [*Memory*] *Minimize memory load.* Users should not be required to memorize a lot of information to carry out tasks. Memory load reduces users' capacity to carry out the main tasks.
   a. Recognition vs. recall (e.g., menu vs. commands).
   b. Externalize information through visualization.
   c. Perceptual procedures.
   d. Hierarchical structure.
   e. Default values.
   f. Concrete examples (DD/MM/YY, e.g., 10/20/99).
   g. Generic rules and actions (e.g., drag objects).
6. [*Feedback*] *Informative feedback.* Users should be given prompt and informative feedback about their actions.

  a. Information that can be directly perceived, interpreted, and evaluated.
  b. Levels of feedback (novice and expert).
  c. Concrete and specific, not abstract and general.
  d. Response time.
    - 0.1 s for instantaneously reacting;
    - 1.0 s for uninterrupted flow of thought;
    - 10 s for the limit of attention.

7. [*Flexibility*] *Flexibility and efficiency*. Users always learn and users are always different. Give users the flexibility of creating customization and shortcuts to accelerate their performance.
  a. Shortcuts for experienced users.
  b. Shortcuts or macros for frequently used operations.
  c. Skill acquisition through chunking.
  d. Examples:
    - Abbreviations, function keys, hot keys, command keys, macros, aliases, templates, type-ahead, bookmarks, hot links, history, default values, etc.
8. [*Message*] *Good error messages*. The messages should be informative enough such that users can understand the nature of errors, learn from errors, and recover from errors.
  a. Phrased in clear language, avoid obscure codes. Example of obscure code: "system crashed, error code 147."
  b. Precise, not vague or general. Example of general comment: "Cannot open document."
  c. Constructive.
  d. Polite. Examples of impolite message: "illegal user action," "job aborted," "system was crashed," "fatal error," etc.
9. [*Error*] *Prevent errors.* It is always better to design interfaces that prevent errors from happening in the first place.
  a. Interfaces that make errors impossible.
  b. Avoid modes (e.g., vi, text wrap). Or use informative feedback, e.g., different sounds.
  c. Execution error vs. evaluation error.
  d. Various types of slips and mistakes.
10. [*Closure*] *Clear closure*. Every task has a beginning and an end. Users should be clearly notified about the completion of a task.
  a. Clear beginning, middle, and end.
  b. Complete 7-stages of actions.
  c. Clear feedback to indicate goals are achieved and current stacks of goals can be released. Examples of good closures include many dialogues.
11. [*Undo*] *Reversible actions*. Users should be allowed to recover from errors. Reversible actions also encourage exploratory learning.
  a. At different levels: a single action, a subtask, or a complete task.
  b. Multiple steps.
  c. Encourage exploratory learning.
  d. Prevent serious errors.
12. [*Language*] *Use users' language*. The language should be always presented in a form understandable by the intended users.
  a. Use standard meanings of words.
  b. Specialized language for specialized group.
  c. User defined aliases.
  d. Users' perspective. Example: "we have bought four tickets for you" (bad) vs. "you bought four tickets" (good).
13. [*Control*] *Users in control*. Do not give users that impression that they are controlled by the systems.
  a. Users are initiators of actors, not responders to actions.
  b. Avoid surprising actions, unexpected outcomes, tedious sequences of actions, etc.
14. [*Document*] *Help and documentation*. Always provide help when needed.
  a. Context-sensitive help.
  b. Four types of help.
    - task-oriented;
    - alphabetically ordered;
    - semantically organized;
    - search.
  c. Help embedded in contents.

3.2. Severity rating scale

The heuristics are used to check the interface of the device design. If a heuristic is violated, it is given a severity rating based on the following scales [12]:

0, not a usability problem at all;

1, cosmetic problem only. Need not be fixed unless extra time is available;

2, minor usability problem. Fixing this should be given low priority;

3, major usability problem. Important to fix. Should be given high priority;

4, usability catastrophe. Imperative to fix this before product can be released.

As a guideline for rating the problems, we consider the proportion of users who will experience it, the impact it will have on their experience with the product, and whether the usability problem will be a problem only the first time they encounter it, or whether it will persistently bother them. A persistent problem with a major impact that most users will encounter will get the highest severity rating.

3.3. Procedure

Typically 3–5 usability experts independently evaluate the user interface of a product and each of them generates a separate list of heuristic violations according to the 14 heuristics described above. We find it convenient for each evaluator to begin with a tabular form, shown in Table 1, where each row contains where the problem occurs (place of occurrence), a description of the problem, the heuristics violated, and the severity rating (though the ratings are filled in at a later stage). It is best to agree ahead of time on what to call each place of occurrence. For instance, "physical interface" may refer to the physical buttons, startup screen, to the screen that appears when the device is powered-on, and so on. A single usability problem identified by an evaluator can be a violation of multiple heuristics, which means that the number of heuristic violations is typically more than the number of usability problems identified. For example, the oxygen flow control knob problem described earlier violates consistency and standards (most devices with smoothly rotating knobs work at any position), visibility of system status (there was no indication of oxygen flow), match between the system and the world (users expect smoothly rotating knobs to work at all positions), and prevent errors (the design of the knob and lack of feedback increase the chance of error). Once the evaluators have identified potential usability problems, the separate lists

Table 1
Ten usability problems (out of 89) and heuristic violations for Pump 1

Places of occurrence	Usability problem description	Heuristics violated	Mean severity rating
Physical design	Tubing is difficult to install. Specifically, the clamp mechanism does not indicate which end goes into pump.	Error, Flexibility	3.25
	The start button may be confused with the on button; and the stop button may be confused with the off button. The meaning of open is not obvious.	Error	3
	Start button too close to power button	Error	3.25
	Contrast adjustment is hidden on the rear of pump handle. It may be inadvertently adjusted when handling pump, and it is hard to find it to adjust it back to normal.	Memory, Visibility	3.75
Opening screen	Displayed for only 15 s. During the first few seconds, no actions are available. After self-test, actions become available, user may not notice this change.	Visibility, Error, Consistency	2.5
	Once the next screen is selected (or appears) users cannot return to internal check/opening screen. To go back the pump must be turned off.	Undo	3.25
Select a Pump Personality	If a personality is highlighted and the select hotkey is pressed, the system immediately enters the Main Display. There is no way to undo selection without turning pump off.	Consistency, Feedback	2.75
Infusion Modes and Features	Erratic use of bolding. Some items in bold can be scrolled to and selected, others indicate states and cannot be selected.	Consistency, Minimalist	1.75
Main display	When "Downstream Occlusion" Alarm sounds, if the occlusion is cleared, the infusion begins automatically. If any button is pressed, including Silence Alarm, user must reenter programming screen (by either pressing primary hotkey or Rate or Vol buttons) and press start.	Error, Closure, Consistency, Visibility, Match	3
	Black bar near the bottom of the screen with the message to "press primary or piggyback" in small font. Visibility may be a problem.	Consistency, Visibility, Feedback	2
Primary	No decimals accepted after 99.9 for Rate or Volume – and there is no alert.	Consistency, Match, Message	2.75
	There is no immediate feedback after values are entered for infusion. Potential for user to get lost unless the correct action is remembered.	Feedback, Memory, Control, Error, Closure	2.75

are compiled into a single master list. This is one of the most tedious steps to the analysis, because someone must read through all the problems in order to eliminate duplicates. It is possible to eliminate this step by having evaluators sequentially evaluate the interface, with each evaluator passing the list on to the next. The master list is then given back to the evaluators who independently assess the severity of each violation. The ratings from the individual evaluators are then averaged.

Before conducting a heuristic evaluation, it is important to be aware of the technique's benefits and limitations. Nielsen has shown that any individual evaluator will only catch 35% of the usability problems, while 3–5 evaluators can detect 60–75%. The best cost benefit ratio is achieved with 3–5 evaluators. The technique is also relatively easy to do, even for those not trained in usability. We have found that 2–3 h of training, combined with clear examples, and a practice evaluation with feedback, is often sufficient to begin using the technique.

Heuristic evaluation also has a number of limitations. It does not indicate the elements of the interface that correctly follow usability guidelines. Nor does it reveal major missing functionality. Other usability engineering techniques at the levels of user, functional, and task analyses can indicate what is right with the system and identify the most appropriate functionality.

Although it requires minimal training in human factors engineering, some understanding of the heuristics is required. In addition, knowledge of the domain in which the interface is used and training in human factors engineering increases the number of problems identified. Nielsen [13] studied three different groups of evaluators; novice evaluators who had general computer knowledge, but no usability knowledge, "single experts" who were usability engineers, but had no specific domain knowledge of the interface being studied, and "double experts" who had both domain knowledge and usability expertise. He found that individual novices identified approximately 22% of the usability problems. The "single experts" found approximately 41% of the usability problems in the interface, and the "double experts" found 60% of the usability problems.

Heuristic evaluation focuses on a single device or application and therefore may not identify problems that arise because of the device's use environment. For example, when an infusion pump is used in the clinical setting, lighting, noise, and other devices in the room may affect the usability of the device. Alarms that are easily heard in a quiet room may not be heard above the noise of other devices or may be confused with other alarms. Leads and tubing for one device may be confused or inadvertently used on the wrong device. Lighting and device positioning may affect display readability. Contextual problems, such as these, are best identified using observational analysis during a short clinical trial. Observing 3–5 users interacting with the device in the real clinical setting can provide information on the device's interface as well as how the device works in context.

Several other techniques may be used in addition to, or as an alternative to heuristic evaluation. In a pluralistic walkthrough, another inspection method, one or more usability experts along with users (and developers for products under development) walk through an interface as a group noting potential problems. This technique is useful when the users (and/or developers) are not well versed in usability engineering and the usability experts are not also domain experts. Typically, the group walks through specific scenarios, such as common tasks.

Heuristic evaluation is often combined with small-scale users tests, in which 3–5 users are given a set of common tasks to perform with an interface while a neutral observer notes any problems that they have without providing assistance. User tests tend to identify a slightly different set of problems than heuristic evaluation, so combining the two techniques can increase the percentage of problems found.

4. Evaluation of two 1-channel infusion pumps

We selected two 1-channel volumetric infusion pumps from two different vendors for heuristic evaluation. There are three objectives for this evaluation. First, we wanted to evaluate whether the heuristics we modified from Nielsen and Shneiderman's heuristics and golden rules can be successfully applied to medical devices. Second, we wanted to use heuristic evaluation to identify usability problems that might be potential triggers for medical errors. Third, we wanted to use the results of heuristic evaluation to compare different infusion pumps in their usability and patient safety features.

Four individuals applied the 14 heuristics in Section 3.1 to the user interfaces of two 1-channel volumetric infusion pumps, identified usability problems in various areas/sections of the pumps, and identified one or more heuristic violations for each usability problem. Two of the four evaluators were graduate students in the School of Health Information Sciences at UT Houston, and the other two were graduate students in the Department of Psychology at Rice University. They had taken at least one graduate level course on human factors or human–computer interaction. Before the evaluation, they were given a copy of the report of a heuristic evaluation conducted for a different product using the same set of 14 heuristics [16]. They were then given instructions on how to conduct the evaluation by the first author of this paper, who is an expert in heuristic evaluation and has performed heuristic evaluations on several products. After the list of usability problems were discovered and heuristic violations were identified for each usability problem, the four evaluators independently assessed the severity of each usability problem. Their severity ratings were then averaged.

Table 1 shows examples of usability problems identified for Pump 1, their corresponding heuristic violations and average severity ratings, and the places where the usability problems were discovered.

Fig. 1 shows the numbers of heuristic violations for the two pumps across the 14 heuristics. For Pump 1, heuristics were violated a total of 192 times. Consistency and Visibility were the two most frequently violated heuristics (53 and 28, respectively). Feedback and Match were the next most common violations (22 and 21). These four heuristics account for 64% of the violations. For Pump 2, heuristics were violated a total of 121 times. Visibility was the most frequently violated heuristic (29 violations). Memory and Consistency were the next most common violations (19 and 17, respectively). These three heuristics comprised 54% of the violations. An example of a violation of the visibility heuristic would be: "When the 'enter' button is not pressed, after entering part or all of the value for 'Rate' and 'VTBI' (volume to be infused), a message appears that reads 'complete entry.' It is not clear what this means. A better phrasing would be Press 'enter' to confirm value." In this case, users are apt to become confused by no clear delineation as to what action would come next. In terms of the quantity of heuristic violations, the results in Fig. 1 indicate that Pump 1 has more usability problems and thus may accordingly have a higher chance of generating medical errors.

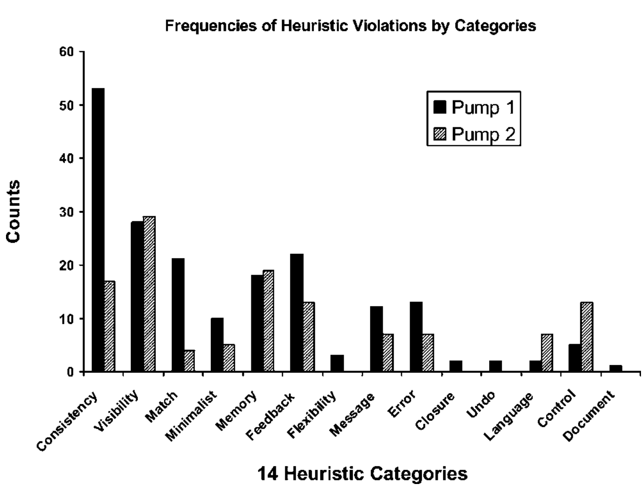

Fig. 1. Heuristic violations in two 1-channel infusion pumps. For Pump 1, heuristics were violated a total of 192 times. Consistency and Visibility were the two most frequently violated heuristics (53 and 28, respectively). Feedback and Match were the next most common (22 and 21). These four heuristics account for 64% of the violations. For Pump 2, heuristics were violated a total of 121 times. Visibility was the most frequently violated heuristic (29 violations). Memory and Consistency were the next most commonly violated (19 and 17, respectively). These three heuristics comprised 54% of the violations.

Fig. 2 summarizes the severity of the problems found in Pump 1 and Pump 2. The severity ratings were divided into four regions. A severity rating equal or above 3.5 is Catastrophic; a severity rating equal or above 2.5 but below 3.5 is Major; a severity rating equal or above 1.5 but below 1.5 is Minor; and a severity rating below 1.5 is Cosmetic. For Pump 1, there were two catastrophic, 38 major, 49 minor, and zero cosmetic usability problems. For Pump 2, there was one catastrophic, 26 major, 26 minor, and zero cosmetic usability problems. In terms of the severity of usability problems, the results in Fig. 2 indicate that Pump 1 has a larger number of more severe usability problems and thus it is likely to cause more medical errors than Pump 2. This conclusion is consistent with the measure in terms of the quantity of heuristic violations: Pump 1 has more heuristic violations than Pump 2. In short, Pump 1 has not only a larger number of heuristic violations in total but also a larger number of more severe heuristic violations.

Figs. 3 and 4 show the places of occurrence of usability problems for Pump 1 and Pump 2 and the average severity rating of usability problems in each area. For Pump 1, usability problems were identified in ten different areas. The areas with the most usability problems are

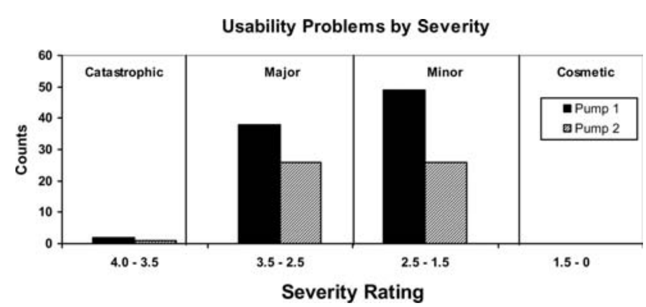

Fig. 2. Severity ratings of usability problems for the two 1-channel infusion pumps. For Pump 1, there was two catastrophic (severity rating $\geqslant 3.5$), 38 major ($3.5 >$ severity ratings $\geqslant 2.5$), 49 violations ($2.5 >$ severity ratings $\geqslant 1.5$), and zero cosmetic usability problems ($1.5 >$ severity ratings). For Pump 2, there was one catastrophic, 26 major, 26 minor, and zero cosmetic usability problems.

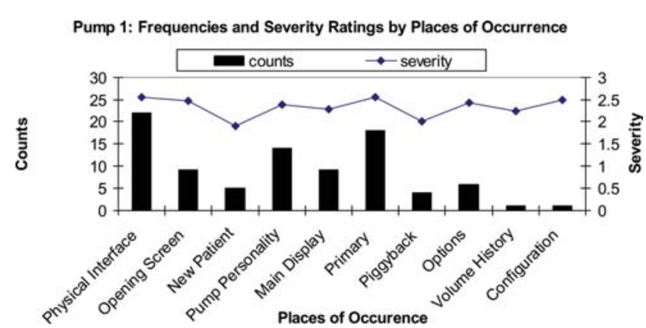

Fig. 3. The areas in Pump 1 with the most usability problems are the physical interface, the primary screen, and the pump personality screen. The average severity ratings by place of occurrence for the 10 areas were between 1.9 and 2.6.

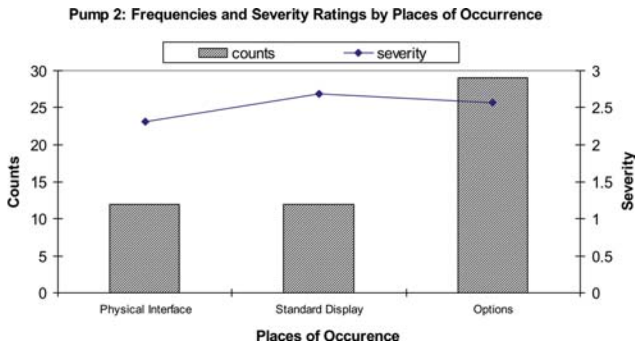

Fig. 4. The area in Pump 2 with the most usability problems was the Options Menu. The average severity ratings by place of occurrence for the three areas were between 2.3 and 2.7.

the physical interface, the primary screen, and the pump personality screen. The average severity ratings by place of occurrence for the ten areas were between 1.9 and 2.6. For Pump 2, usability problems were identified in three different areas. The area with the most usability problems was the Options Menu. The average severity ratings by place of occurrence for the three areas were between 2.3 and 2.7. Comparing Figs. 3 and 4, it is not surprising to see why Pump 1 was inferior to Pump 2 as shown in Figs. 1 and 2. Pump 1 has many different areas that users have to interact with, whereas Pump 2 only has three primary areas for user interaction. The larger number of areas in Pump 1 increased the complexity of its user interface, and accordingly it was found to have more usability problems in general and more usability problems with higher severity ratings.

5. Conclusion

As a discount usability technique, heuristic evaluation is easy to use, easy to master, efficient, effective, and useful. It can be used to identify a great proportion of major usability problems in a product in a timely manner with reasonable cost. As we discussed at the beginning of this paper, human errors in medical device use are largely due to interface design problems that can be potentially addressed through user-centered design, interventions, and other considerations. Since heuristic evaluation can identify usability problems, whose quantity and severity are strongly linked to the frequency of medical errors, heuristic evaluation is a method for indirectly assessing patient safety features in medical devices. Although it is limited in its scope of coverage of the full range of patient safety related features in medical devices, it is a practical tool that should be adopted by medical device manufacturers for the design and modification of medical devices, and by healthcare institutions for the evaluation of medical devices.

Acknowledgments

This project was sponsored by US Department of Army under Co-operative Agreement No. DAMD17-97-2-7016. The content of the information does not necessarily reflect the position/policy of the Government or NMTB, and no official endorsement should be inferred. It was also supported in part by Grant No. P01 HS11544 from the Agency for Healthcare Research and Quality. We thank Asma Ali, Julie Brixey, Philip Chung, and Xiaozhou Tang for their contributions to the heuristic evaluation.

References

[1] Kohn LT, Corrigan JM, Donaldson MS. To err is human. Washington, DC: National Academy Press; 1999.

[2] Food and Drug Administration. Human Factors Implications of the New GMP Rule. Overall Requirements of the New Quality System Regulations. Available from: http://www.fda.gov/cdrh/humfac/hufacimp.html [2000, June 27]; 1998.

[3] Sawyer D, Aziz KJ, Backinger CL, Beers ET, Lowery A, Sykes SM, et al. Do it by Design: An Introduction to Human Factors in Medical Devices. US Department of Health and Human Services, Public Health Service, Food and Drug Administration, Center for Devices and Radiological Health; 1996.

[4] Food and Drug Administration. Improving Patient Care by Reporting Problems with Medical Devices: A MedWatch Continuing Education Article. Available from: http://www.fda.gov/medwatch/articles/mdr/mdr.pdf; 1997.

[5] Cooper JB, Newbower RS, Long CD, McPeek B. Preventable anesthesia mishaps: a study of human factors. Anesthesiology 1978;46(1):399–406.

[6] Obradovich JH, Woods DD. Users as designers: how people cope with poor HCI design in computer-based medical devices. Hum Factors 1996;38(4):574–92.

[7] Lin L, Isla R, Doniz K, Harkness H, Vicente KJ, Doyle DJ. Applying human factors to the design of medical equipment: patient-controlled analgesia. J Clin Monit Comput 1998;14(4): 253–63.

[8] Zhang J, Patel VL, Johnson KA, Malin J, Smith JW. Designing human-centered distributed information systems. IEEE Intell Syst 2002;17(5):42–7.

[9] Card SK, Moran TP, Newell A. The psychology of human–computer interaction. Hillsdale, NJ: Erlbaum; 1983.

[10] Kieras D. Using the keystroke-level model to estimate execution times. Unpublished report. Ann Arbor: Department of Psychology, University of Michigan; 2001.

[11] Nielsen J, Molich R. Heuristic evaluation of user interfaces. In: Proceedings of ACM CHI'90; 1990. p. 249–56.

[12] Nielsen J, Mack R, editors. Usability inspection methods. New York: Wiley; 1994.

[13] Nielsen J. Usability engineering. Boston: AP Professional; 1994.

[14] Nielsen J. Finding usability problems through heuristic evaluation. In: Proceedings of ACM CHI'92; 1992. p. 372–80.

[15] Shneiderman B. Designing the user interface. 3rd ed. Reading, MA: Addison-Wesley; 1998.

[16] Zhang J, Johnson TR, Tang Z. Usability engineering project for Digital EMS: Part 1 – Heuristic evaluation of Version 1. Unpublished Report. Houston: UT Houston School of Health Information Sciences; 2002.