A Component-Based Evaluation Protocol for Clinical Decision Support Interfaces

Alessandro Febretti[1], Karen D. Lopez[2], Janet Stifter[2], Andrew E. Johnson[1], Gail M. Keenan[2], Diana J. Wilkie[2]

[1]Department of Computer Science,
College of Engineering, University of Illinois at Chicago (UIC)

[2]Department of Health Systems Science,
College of Nursing, UIC

Abstract. In this paper we present our experience in designing and applying an evaluation protocol for assessing usability of a clinical decision support (CDS) system. The protocol is based on component-based usability testing, cognitive interviewing, and a rigorous coding scheme cross-referenced to a component library. We applied this protocol to evaluate alternate designs of a CDS interface for a nursing plan of care tool. The protocol allowed us to aggregate and analyze usability data at various granularity levels, supporting both validation of existing components and providing guidance for targeted redesign.

Keywords: component-based testing, cognitive interviewing, user-centric design, healthcare interfaces

1 Introduction

Clinical Decision Support systems (CDSs) are software tools designed to support decision making in the clinical setting and facilitate the practice of evidence-based healthcare. CDSs have traditionally consisted of alerts and guidelines based on randomized clinical trials, systematic reviews and other sources of evidence. More recently developed CDSs are based on the characteristics of an individual patient that are matched to an electronic knowledge base and health record, to provide healthcare personnel with just-in-time, patient-specific recommendations.

The use of electronic health records in general and clinical decision support systems in particular has the potential of greatly improving care quality, but the adoption rate of these tools in the United States has been lower than expected. One of the main reasons for this delay is the lack of efficiency and usability of available systems [1].

Most CDS research and systems focus on identifying what information to show to users, but little has been done to find how to present complex patient data to support efficient decision making. Performing usability testing in the context of CDS design is therefore fundamental. CDS systems should drive healthcare personnel towards effec-

tive and targeted actions to improve patient outcomes. Poorly designed CDS features may confuse the user and lead to longer response times. Nursing staff often have strict time constraints and may also choose to ignore CDS features that are not easily accessible, or that do not provide clear information. Worse yet, inconsistent CDS features may drive healthcare personnel into making wrong decision about the patient's care.

Given the variety of forms in which clinical information can be transformed and presented, the overall organization of user testing is highly complex. For example, a single user may be exposed to multiple prototypes of the overall system, each one showing variants and compositions of CDS features in order to determine what is the best (i.e. fastest and clearest) interface. As the interface evolves and new evidence arises from practice or literature, features may be added, removed or redesigned and then evaluated in a new testing cycle.

In this paper we present a protocol that applies the principles of component-specific usability testing, quantitative content analysis and cognitive interviewing to the evaluation of a prototype CDS interface. The protocol has been applied to support the design of the next generation Hands-on Automated Nursing Data System (HANDS). In particular, we wanted to assess the accessibility, interpretability, satisfaction and value-to-practice of distinct CDS artifacts embedded in the interface. We wanted to compare variants of those artifacts across all those metrics. And we wanted to evaluate different compositions of those artifacts in the prototype.

2 Related Work

Usability testing of electronic health record (EHR) interfaces is not new and has been applied both for personal and clinical interfaces [2], [3]. Beyond usability capturing practice-based and literature-based evidence for CDS interfaces, it is also critically important to evaluate how the integration of this evidence into EHRs affects professional and organization practices [2].

Similar work has also been done in the context of CDS [4], but most of the work evaluates interfaces as a whole, have *a priori* defined tasks or do not consider compositional variations of multiple interface features. For these reasons, they typically lack a quantitative analysis of user response to specific features within the interface.

In [5], the authors underscore how traditional usability tests that capture usability for the application as a whole are less effective at capturing the inherent interaction between application components: evaluating the overall usability of an application also cannot inform the selection of right components, their composition into a system and the analysis of their value which includes human-factor issues.

A component-based testing methodology can drive the development of modular, usable interface artifacts for future use and helps in determining whether, for instance, the user interface provided by the various components do not rely on conflicting mental models. Our work represents a practical example of iterative, component-based testing applied in the context of CDS systems.

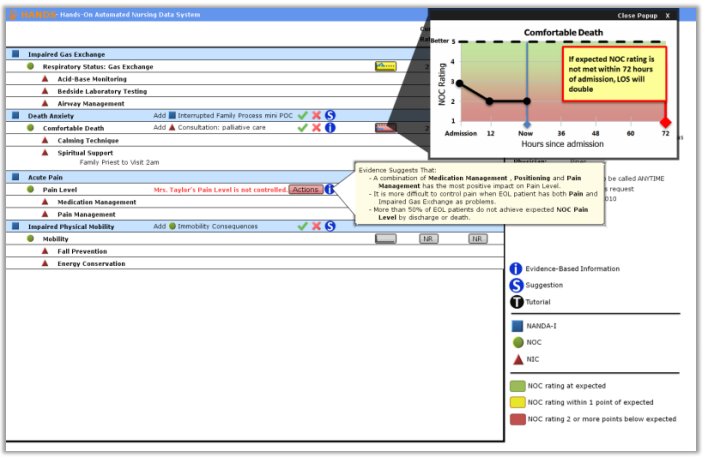

Fig. 1. An example of HANDS interface enriched with clinical decision support features. Shown here are quick actions, outcome trend charts with annotations, and evidence-based information tooltips.

3 Context: the HANDS System

The need for a component based evaluation protocol was driven by the need to integrate CDS into HANDS [6]. HANDS is an electronic tool that nurses use across time to enter data and track the patient's clinical history within a care setting, such as a hospital. A hospitalization includes all plans of care that nurses document at every formal handoff (admission, shift-change update, or discharge). HANDS uses a standardized nomenclature to describe diagnoses, outcomes and interventions.

Nursing diagnoses are coded with NANDA-I terms[7], outcomes are coded using terms and rating scales from the Nursing Outcomes Classification (NOC)[8], and interventions are coded with terms from the Nursing Intervention Classification (NIC)[9].

3.1 End-of-Life CDS

Of the 60 billion of Medicare dollars spent each year on care of the dying, $300 million are spent during the last month of life, including many millions for inappropriate treatments provided to hospitalized patients [10].

Until now, not enough standardized nursing care data was available, making it impossible to develop a set of CDS benchmarks that could be used to guide nursing actions for end-of-life patients. Recently, the HANDS system has been successfully used over a two-year period on 8 acute care units in 4 Midwestern hospitals, accounting for more than 40,000 patient care episodes. Data mining and statistical analysis on

those episodes of care identified a set of benchmarks that related to end-of-life pain management and death anxiety. For instance, specific interventions, like patient positioning, were statistically more likely to achieve desired pain outcomes; pain control achieved at 24 hours predicted pain levels for the entire stay; and dealing with family coping in younger patients helped reduce death anxiety. These findings allowed us to prepare 6 distinct evidence-based-information (EBI) components that we wanted to add to the HANDS interface.

We therefore wanted to develop an evaluation protocol that would allow us to:

- Assess the interpretation, accessibility and value-to-practice characteristics of single CDS features
- Evaluate the effectiveness of feature compositions into full prototypes
- Track the evolution of features at different stages of the design

4 Methodology

Our proposed evaluation protocol is defined by three major elements: a component library, a user interview protocol and a data coding scheme.

4.1 Component Library

To effectively analyze usability data it is fundamental to keep track of CDS and non-CDS component versioning and composition. To address this challenge we developed a component library that associates unique names to components, lets the researchers visualize their different versions and track which version is used in which prototype. Fig. **2** shows an example of a component library.

4.2 User Interview Protocol

User interviews are divided in two parts. During the first part of the interview, the user is introduced to the patient care scenario (in our context, an end-of-life patient history and current status). The user is then presented with the first of a series of prototypes exposing a set of pre-selected components and instructed to "think-aloud" as they interact with them.

It is important to underscore here that the user is not assigned to complete a specific task. Our users are health care professionals: we want them to take reasonable actions on the interface, depending on the patient status, history and on the presented CDS information. Defining tasks *a priori* is challenging and less meaningful in this context. We therefore take task completion to correspond to users verbally 'committing' their actions. For instance, after reading the patient information and CDS, and modifying the plan of care, the user could say he has done what was needed and is ready to move to the next patient. During this part of the interview, the user may go through multiple prototype versions.

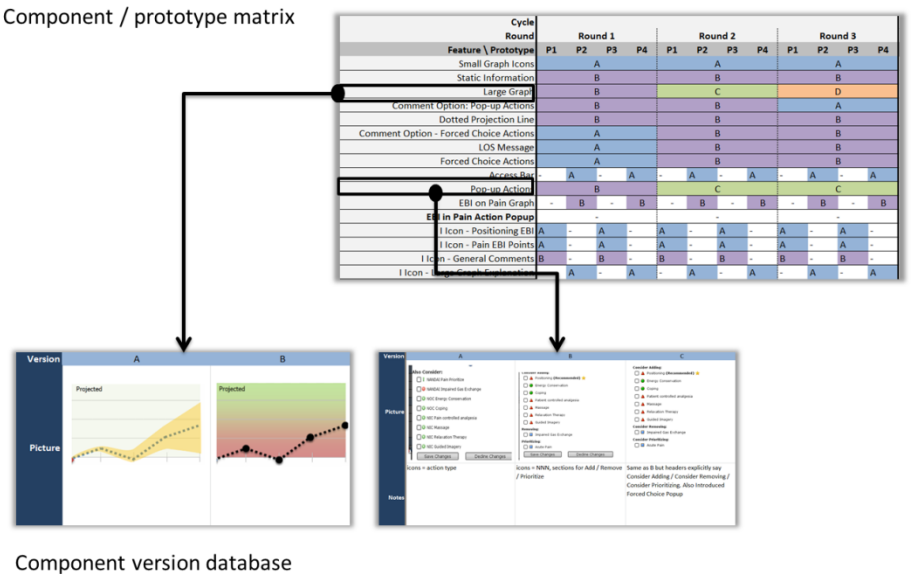

Fig. 2. An example of a component library. The summary matrix identifies components used for each prototype iteration. Component names are linked to a database of images and notes, to simplify recalling the evolution of each feature.

Before each version, the prototype is reset to its initial state, and a few components are switched to different versions or turned on/off completely. The composition of features into tested prototypes depends on which research questions we wish to address for a group of subjects.

During the second part of the interview, the interviewer performs a cognitive interview [11], reviewing their actions, investigating why specific paths in the interface were taken (or not) and eliciting additional responses about interpretability and value to practice of CDS features.

4.3 Data Coding Scheme

Meaningful information is extracted from the interview using a qualitative approach. Qualitative analysis is advantageous in our setting, since it allows an expert reviewer to analyze the full context of user actions and utterances [12], [13]. In particular we are interested in evaluating the meaningfulness of user interaction (i.e., users correctly interpret presented evidence and take appropriate actions), levels of user confusion over specific components, or their considerations over the significance or display of information in the interface. A disadvantage of qualitative analysis is its subjectivity. Different reviewers may code user behavior in different ways, or the same reviewer may be inconsistent in interpreting it. The first problem can be miti-

gated by implementing inter-rater reliability practices as part of the protocol [14]. The second is addressed by the rigorous definition of a coding scheme.

For component-based CDS interface evaluation, we propose coding based on linking codes to components: Each code in the scheme identifies a unique component, as described in the component library. Therefore, each appearance of a code marks some meaningful user activity associated to a specific component. Codes are enriched with additional information: a category that identifies the type of user activity (accessibility, interpretation, or comments on component value or pleasantness); a three-category score (positive, negative, unclear); a component version and prototype identifier; and an optional comment by the reviewer that further defines the activity.

As an example, consider the following user activity with a 'chart' component in a nursing plan of care interface. A nurse tries to click on the chart and is surprised when nothing happens. After realizing the chart is static, she goes on and correctly interprets the information presented by the chart. Before moving on, she mentions that although she likes the chart she does not think her colleagues would use that in practice, since they are more used to simple tables. In our protocol a reviewer would code the previous activity using the four markers shown in Table 1.

Component type	Category	Score	Prototype Id
Chart	Accessibility	Unclear	1
Chart	Interpretation	Positive	1
Chart	Likability	Positive	1
Chart	Value	Negative	1

Table 1. Markers used in the charting activity example

A valuable feature of this coding scheme is that, while extracting data in a qualitative (but rigorous) manner, it supports quantitative analysis on interface components. The coding scheme is generic enough to allow for a great amount of flexibility in possible research questions. It also supports exploratory analysis of recording data, when researchers have no *a priori* theory to validate. Moreover, it allows the aggregation of multiple component scores into bigger modules to change the granularity of the analysis. The presence of reviewer comments allows for qualitative analysis of specific findings when needed.

5 Experiment

As mentioned in section 0, this protocol was implemented to test the introduction of CDS features into a prototype of the HANDS system. This is an ongoing research project. The total number of CDS features developed at the time of publication is 6. Together with ancillary user interface elements that we wanted to evaluate, we had a total of 16 distinct components, possibly with multiple versions each (up to 4).

We recruited a total of 25 nurses in different age groups, years of experience and education levels. We ran 4 interview rounds. Each pair of rounds was considered part

of a design cycle: in each cycle we tested the introduction of EBI features relative to a specific end-of-life issue. The first cycle addressed EBIs related to pain, the second addressed pain and death anxiety. Minor prototype redesign were carried out between rounds in the same cycle. New components and major redesign of existing ones happened between the cycles based on component-specific usability data analysis.

Users were introduced to a fictional end-of-life patient that was assigned to their shift. The patient history, demographics and current plan of care were designed to elicit the activation of the CDS features that we wanted to test. Users were presented with a prototype of the plan of care interface. Once they considered their actions on the plan of care satisfactory, they would be presented with a new prototype: the initial patient plan of care would stay the same but some of the interface components would be switched to different versions.

The users were instructed to 'rewind' and observe this patient again through the interface, as if it was a new patient. We tested four prototype variations for each user. The order in which the prototypes were presented to the users was randomized. Cognitive interviewing would then be performed, and users were asked to choose their most and least favorite prototype versions before ending the interview.

6 Results

For the purpose of this paper we will present an example of analysis from our second design cycle. During this cycle we interviewed 15 users, collecting a total of ~1600 markers. Qualitative data analysis was performed by 4 separate reviewers. Inter-rater reliability was established through a tutorial coding run, and then by separately coding and comparing about 25 minutes of interview data. Coding agreement was measured at 80%. Most disagreement was represented by differing use of the negative and unclear scores.

The coded data were extracted from the coding software (Morae [15]) and preprocessed to extract data fields. The data were then pivoted / aggregated along several dimensions to perform analysis.

For instance, aggregating data by interview section along the user-id dimension, allowed us to perform a quick assessment of the data quality. Most subjects were coded consistently, except for two for which we collected a below average number of codes (3.5% compared to 7% average). Aggregating by component id along the prototype-id dimension was used to generate a component 'heat-map' (Fig. **3**) that could be used to quickly identify areas of interest for analysis.

Aggregating data by component along the score dimension provided an overview of score distributions for each component. This process allows us to quickly identify issues with specific components with high percentages of negative or unclear scores. For instance, one of our CDS features (a popup message related to pain management) had a low positive score of 25% (over 63 total component activations). Through the pivot table, we easily 'zoomed into' this specific component, to split percentages by category (Figure 5). We then assessed that the problem was not related to the compo-

Count of Event	Column Labels					
Row Labels	Cognitive Interview	Prototype 1	Prototype 2	Prototype 3	Prototype 4	Grand Total
A (Pop-up Actions)	5.08%	11.38%	10.33%	11.35%	11.85%	9.21%
B (Dotted Projection Line)	2.26%	1.72%	0.47%	1.77%	0.74%	1.58%
C (Forced Choice Actions)	13.56%	13.79%	15.02%	15.96%	7.41%	13.18%
E (EBI - Under Red Flashing Alert)	8.10%	0.00%	3.29%	0.35%	0.37%	3.28%
F (Flashing Alert)	3.01%	6.90%	13.15%	14.54%	15.19%	9.21%
G (Gold Star)	3.01%	1.03%	0.00%	0.00%	1.11%	1.39%
H (I Icon - EBI Pain)	7.72%	2.76%	0.00%	2.84%	2.59%	4.04%
I (I icon - General Comments)	1.88%	3.79%	1.41%	4.96%	1.48%	2.65%
J (Mini POC)	1.69%	0.69%	1.41%	0.71%	0.37%	1.07%
K (Palliative Consult)	1.32%	0.00%	0.47%	0.71%	0.00%	0.63%
L (LOS Message)	5.27%	2.07%	0.47%	1.77%	3.70%	3.15%
M (i Icon - Large Graph Explanation)	2.07%	0.00%	1.88%	0.00%	1.48%	1.20%
N (I icon - EBI Positioning)	2.45%	1.03%	0.00%	2.13%	0.00%	1.39%
O (Field Notes/Observation)	6.59%	6.55%	2.82%	6.03%	6.30%	5.93%
P (Preliminary Design Ideas)	5.27%	3.79%	2.82%	1.42%	3.70%	3.72%
Q (Static Information)	0.94%	6.21%	3.29%	3.55%	7.41%	3.78%
R (Large Graph)	5.65%	8.62%	8.92%	6.38%	11.11%	7.69%
S (Small Graph Icons)	0.75%	4.83%	9.86%	7.45%	7.41%	5.04%
T (NNN Icons)	1.51%	14.14%	7.98%	3.55%	4.44%	5.55%
U (Comments Option - Forced Choice Actions)	0.94%	0.34%	0.47%	1.06%	0.37%	0.69%
V (Change Recognition)	4.71%	4.48%	7.98%	7.80%	5.93%	5.86%
W (Long Access Bar)	1.88%	0.00%	1.88%	0.00%	1.85%	1.20%
X (Bug)	0.75%	2.41%	2.82%	2.13%	1.85%	1.77%
Y (Comments Option - Pop-up Actions)	2.64%	2.07%	1.88%	1.06%	1.85%	2.02%
Z (General Prototype Comments)	10.92%	1.38%	1.41%	2.48%	1.48%	4.79%

Fig. 3. A 'heat map' view of the marker data using conditional coloring on marker percentages. Through this view it is possible to quickly identify particularly active components. In this interview cycle we identified five main active components.

nent usability (i.e., finding and opening the popup), but interpretation scores were very low (16% positive).

This means that the pain evidence we presented was formulated in an inconsistent or unclear way. We then further zoomed the view for negative scores, to access available reviewer comments and identify problem patterns across multiple users. Incrementally zooming into the data in this fashion was a very effective analysis tool. It allowed us to identify issues at a high level, hiding unnecessary information until needed.

Another fundamental tool for data analysis was the ability to quickly filter data along any dimension. It was used to identify hard-to-find or unused CDS components. One CDS feature in particular was not noticed by most users until the interviewer guided them to it during cognitive interview. We excluded the cognitive interview codes form the aggregate data: when users accessed this CDS component unassisted, they valued it positively (67%). For the next design cycle we then kept the content of this component, and moved to increase the likelihood of users accessing it.

7 Concluding Remarks

One issue we observed with this methodology is common to other component-based testing approaches. A usability assessment of single components does not automatically translate into an assessment of full interfaces. For instance, averaging likability scores for all the CDS features expose in a prototype did not necessarily lead to an estimate of overall prototype likability.

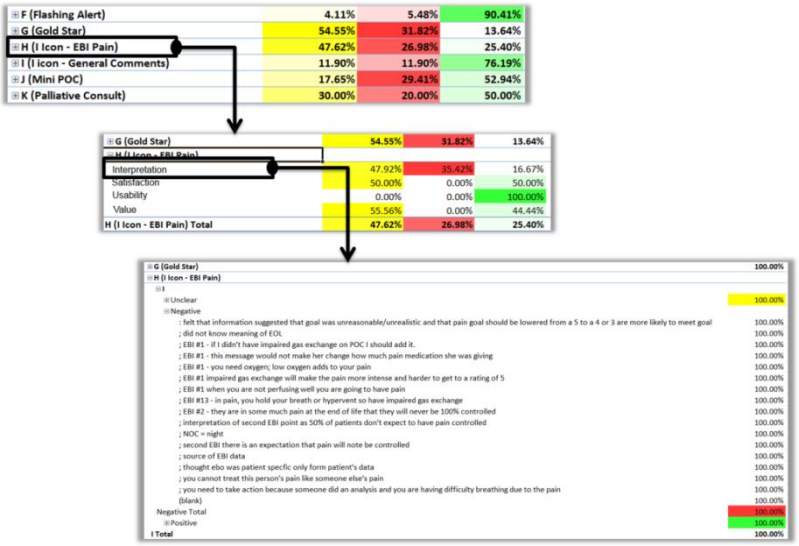

Fig. 4. Incremental zoom levels into the data

At the end of an interview we asked the user to choose most and least favorite versions of the HANDS CDS prototype. We observed that preference was usually tied to only one CDS component, and it influenced the choice of the favorite prototype.

Moreover, components that rely on different interpretation or interaction models may be considered clear or valuable as a stand-alone component but could result in an inconsistent user experience when assembled into a prototype. These issues can be mitigated by collecting separate overall usability or preference data, or by introducing markers in the coding scheme that better capture inter-component factors.

Regarding prototype preference, we also observed significant variability between users. During the design and testing cycle presented in this paper, no prototype out of the 4 tested was a clear winner: prototype choice polarized on two variations, and as mentioned, was mostly driven by preference of one CDS component version over another. We plan to further investigate this, as we suspect this preference is linked with user demographics (clinical experience, age, familiarity with electronic health record tools).

In conclusion, in its current version, the presented evaluation protocol performed well in assessing the usability and value of components of a CDS prototype. A well-defined coding scheme cross-referenced with a component library allowed us to effectively keep track of the evolution of prototypes and component versions.

The quantitative data gathered at the end of the current design cycle helped inform design decision for the next iteration of the HANDS prototype, and captured a few issues that did not emerge by inspection of prototypes, or by informal assessment of user performance. We plan to further use and validate this protocol in several future design and evaluation cycles of the HANDS CDS interface.

8 References

1. J. Belden, R. Grayson, and J. Barnes, “Defining and testing EMR usability: Principles and proposed methods of EMR usability evaluation and rating,” Healthcare Information and Management Systems Society (HIMSS), 2009.
2. D. a Haggstrom, J. J. Saleem, A. L. Russ, J. Jones, S. a Russell, and N. R. Chumbler, “Lessons learned from usability testing of the VA’s personal health record.,” Journal of the American Medical Informatics Association : JAMIA, vol. 18 Suppl 1, pp. i13–7, Dec. 2011.
3. M. Hori, Y. Kihara, and T. Kato, “Investigation of indirect oral operation method for think aloud usability testing,” Human Centered Design, pp. 38–46, 2011.
4. I. CHO, N. STAGGERS, and I. PARK, “Nurses’ responses to differing amounts and information content in a diagnostic computer-based decision support application,” Computers Informatics Nursing, vol. 28, no. 2, pp. 95–102, 2010.
5. W.-P. Brinkman, R. Haakma, and D. G. Bouwhuis, “Component-Specific Usability Testing,” IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, vol. 38, no. 5, pp. 1143–1155, Sep. 2008.
6. G. Keenan, E. Yakel, and Y. Yao, “Maintaining a Consistent Big Picture: Meaningful Use of a Web-based POC EHR System,” International Journal of Nursing Knowledge, 2012.
7. NANDA International., Nursing diagnoses: Definition and classification. .
8. M. M. I. Moorhead S, Johnson M, “Outcomes Project. Nursing outcomes classification (NOC),” Mosby, 2004.
9. B. G. Dochterman JM, Nursing interventions classification (NIC), Mosby. 2004.
10. B. Zhang and A. Wright, “Health care costs in the last week of life: associations with end-of-life conversations,” Archives of Internal Medicine, 2009.
11. P. Beatty and G. Willis, “Research synthesis: The practice of cognitive interviewing,” Public Opinion Quarterly, 2007.
12. M. Patton, Qualitative research & evaluation methods. 2001.
13. H. Hsieh and S. Shannon, “Three approaches to qualitative content analysis,” Qualitative health research, 2005.
14. D. Armstrong, A. Gosling, J. Weinman, and T. Marteau, “The place of inter-rater reliability in qualitative research: an empirical study,” Sociology, 1997.
15. “Morae usability testing software from TechSmith.", www.techsmith.com/morae.html.