MEDINFO 2010
C. Safran et al. (Eds.)
IOS Press, 2010

doi:10.3233/978-1-60750-588-4-927

Applying a User Centered Design Methodology in a Clinical Context

Hajar Kashfi

Division of Interaction Design, Department of Computer Science and Engineering,
Chalmers University of Technology, Gothenburg, Sweden.

Abstract

A clinical decision support system (CDSS) is an interactive application that is used to facilitate the process of decision-making in a clinical context. Developing a usable CDSS is a challenging process; mostly because of the complex nature of domain knowledge and the context of use of those systems. This paper describes how a user centered design (UCD) approach can be used in a clinical context for developing a CDSS. In our effort, a design-based research methodology has been used. The outcomes of this work are as follow; a customized UCD approach is suggested that combines UCD and openEHR. Moreover, the GUI developed in the design phase and the result of the GUI evaluation is briefly presented.

Keywords:

Clinical decision support system, User centered design, Usability, UCD, Prototype, Design and development process, Iterative design, *open*EHR.

Introduction

Errors that occur in a clinical process are mostly due to cognitive limitations of humans, the potential to forget knowledge in the health care flow. Information systems have the ability to decrease such errors by supporting clinicians in this process e.g. by reminding them of important factors to be considered for the current case or to alert them of adverse drug-drug interactions [1]. A Decision Support System (DSS) is an interactive application that is supposed to facilitate the process of decision making for decision makers. This support is done by mapping or compiling existing data to useful information that can be used as a clue for making the best decision [2]. Clinical Decision Support Systems are those DSS:s that are used in the clinical domain. CDSS:s are intended to help clinicians in the process of decision making. Services supported by CDSS:s include diagnosis, alerting, reminding, treatment suggestions, and patient education. Based on a thorough literature review done on around 140 papers about CDSS, it is clear that CDSS:s have the potential to improve care [3].

Usability of CDSS

ISO 9421 [4] defines usability as the "Extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use." Poor usability is the one of the reasons for why CDSS:s have not yet gained a broad acceptance. While there have been many efforts in developing CDSS:s, very few of those systems have been accepted in real clinical environments. Studies show that user interfaces have an impact on acceptability of CDSS:s in a clinical context. The success of a CDSS has a direct relation to the way its graphic user interface (GUI) has been designed [5].

CDSS:s are meant to reduce clinical errors, nevertheless, because of improper design of those systems, other kinds of errors may occur by using them [1,6]. Studies reveal that clinical information systems with low usability not only do not improve patient care and reduce clinical errors, but also may have the opposite effect [7, 8].

Involving Clinicians in the Design Process of CDSS:s

Not just in the clinical domain, but in every other domain experiences show that by involving users in the design and development process of a system, the system will be more usable for the intended users [9-12].The design approach which emphasizes on involving users in the design is called User Centered Design process(UCD) [9, 10]. Accordingly, one can not develop a CDSS which addresses clinicians' needs in a clinical context without a design process in which end users, clinicians, are involved actively [13]. To make a CDSS a usable product, we should consider not only user needs that reveal functional requirements of the system, but also non-functional or usability requirements as well as characteristics of the clinical environment in which the system will finally be applied.

In this paper, we present issues related to user-centered design of a CDSS for Dry Mouth, an oral disease. The main reason for selecting Dry Mouth is that our end users expressed a need for a CDSS for this disease.

Methods

The research method we applied in our work is a design-based research method [14]. For this purpose, our collaborators in Sahlgrenska Academy[1] suggested the design and development of a CDSS for an oral disease named Dry Mouth. "Dry mouth or Xerostomia is the abnormal reduction of saliva and can be a

[1] http://www.sahlgrenska.gu.se/english

symptom of certain diseases or be an adverse effect of certain medications"[15]. Treatment of Xerostomia is related to finding its cause(s). There are five main categories for Xerostomia: Drug-induced, Disease-induced, Radiation-induced, Chemotherapy-Induced, and cGVHD-induced [15].

The reason for suggesting Dry Mouth was that the dentists and dental hygienists are commonly the first clinicians to face the complaints by patients regarding this disease; hence, they should be aware of it and its problems to prevent the deleterious consequences of this disorder. However, according to our expert panel, finding cause(s) of Dry Mouth is a challenge for dentist and dental hygienists, and needs to be supported by a clinical application. The decision support process we aim for includes these two main steps (1) finding the cause(s) of disease based on the patient's medical records (2) suggesting related materials and treatment options, based on results from the first step. Since this system is intended to be used integrated with an existing Clinical Data Entry application [16], data entry forms are not part of the Graphical User Interface (GUI), however we have to provide users with options to edit existing data. Finally, users need to be able to enter their own comments; including diagnosis or treatments to the system.

The Design Process

The approach we use in this design process is UCD. UCD focuses on the end users, their needs and the context in which the system will be used. The main goal in this method is user satisfaction. UCD has an iterative nature. It means that during the design and development process, at several points, prototypes are delivered to users for evaluation and improvement.

As depicted in Figure 1, the idea of UCD is a circular design process including analysis, design, prototyping and getting user feedback. End users are in the heart of this design process and should be involved in all steps. Users are asked about what they expect the application to do for them and what priorities they have in doing their tasks using the intended application. Users have the chance to specify their needs as detailed as possible e.g. which colors do they prefer or what are their time limits running a specific task using the application.

On the other hand, informaticians can communicate with users to extract vital information about their current situation and their future needs e.g. what users like about the way they are currently doing their tasks or what would they like to be changed [11]. Finally, *task analysis* and *evaluation* [9-10] should be done based on the gathered information.

The Importance of Involving Clinicians in the design

Domain knowledge plays the main role in complexity of clinical applications. Clinical tasks may not be complex by themselves but what makes the clinical application development so complicated is that most of the clinical processes are unstructured. They are done in clinicians' mind and based on their expertise. Moreover, clinical knowledge is ever-changing.

Hundreds of data items are involved in a clinical decision making process. After all, concepts in clinical domain are not easy to understand for informaticians and they face difficulties communicating with clinicians or studying literature to get enough domain knowledge to be able to model it and to develop an application.

Extracting domain knowledge in the clinical domain has always been a bottleneck in the development process of such systems and a challenge for informaticians. Therefore, in the clinical domain, we need to involve clinicians in designing the Information model or more precisely domain concept models to be used for information modeling. One of the recent approaches that focuses on involving users in domain concept modeling is *open*EHR [17].

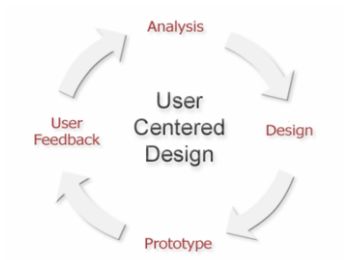

Figure 1- User Centered Design Process[2]

The openEHR Approach

*open*EHR is an open standard specification that emphasizes on the role of clinicians in organizing domain knowledge in form of different clinical concepts such as observation, evaluation, instruction and action [17].

In the *open*EHR approach, clinicians are in charge of defining the specification of clinical knowledge to be used in information modeling. This approach suggests a two level architecture for clinical applications to separate knowledge and information levels in order to overcome the problems caused by the ever-changing nature of clinical knowledge.

While the main emphasis of *open*EHR is on semantic interoperability of medical records, we found the approach highly compatible with UCD. Therefore, we applied *open*EHR to facilitate involving clinicians in the design and to ease domain concept modeling and communicating with our end users.

From an UCD point of view, *open*EHR is very helpful. This approach recommends the utilization of expert knowledge not only just by consulting clinicians but also by letting them design concept models based on what they have in mind. By applying *open*EHR, we can communicate better with our end users since clinical concepts recommended by *open*EHR are understandable for clinicians.

UCD Principles Applied in The Project

There are number of principles that are recommended in UCD [18]: *Multidisciplinary design team, Understanding users and context, Active user participation, Early prototyping, Continuous evaluation,* and *Holistic design*. Besides Holistic de-

[2] Copyright 2009, Kevin Bury Design

sign, we have been concerned about the other principles, as explained below.

Project Team

Our development team is a multidisciplinary team consisting of an interaction design expert, two computer scientists with different backgrounds (AI, Software Engineering), a domain expert (specialist in dentistry), a programmer, and a nurse. However, more end users and experts are involved in different steps in various time periods.

Intended Users, Tasks and Context of Use

One of the main principles in UCD is to define users and the context of use [10].

Users: In this project, direct users are dentists who work in an oral medicine clinic. We used narrative explanation of some typical end users, *personas,* [7] to find more about our end users' characteristics. Since the output of the CDSS will be a treatment decision, patients are our indirect users. Nonetheless, patients will not use the system directly.

Tasks: Based on literature review and interviews with end users and domain experts, we defined the tasks listed below as the main tasks that dentists carry out with regard to Dry Mouth: (1) Information Overview (2) Information manipulation, (3) Requesting related actions like laboratory tests, (4) Diagnosis, (5) Referring to guidelines and other clinical evidence, (6) Recording the results.

Context of use: Dentists will use the application while they are visiting a patient in the clinic. They will use it in presence of the patient, at the same time they are communicating with the patient and in a setting with a limited amount of time.

Users' priorities/Usability goals: The goal is to develop a system, which fits to the dentists' workflow as much as possible; experiences show that clinicians should not need to change their clinical workflow while using a CDSS [18]. It is also important to consider that not all clinicians are experienced in using information systems. On the other hand, because of their occupations, they do not manage to spend much time on learning a new application. Based on this information, we set up our usability goals such as: Effectiveness, Efficiency, Safety, Learnability, etc.

Iterative GUI Design and Evaluation

During the design, we have been using both low fidelity and high fidelity prototypes. From those, we can name sketches designed and improved during brain storming sessions for collecting functional requirements and usability requirements together with our expert panel. In this step, conceptual design of the application was done. These sketches were later translated to some power point prototypes. Afterwards, low fidelity prototyping tools were used to visualize the design solutions. Finally, a Java based GUI has been developed to make the final usability tests more realistic and reliable.

Iterative Domain Concept Model Design

The domain concept modeling started with brain storming sessions in which our expert panel (experts in Dry Mouth) were asked to think about Dry Mouth and its related concepts based on this question: *What do you want to know about a patient who visits you because he/she suffers from Dry Mouth?;* and to put as much information as possible on a paper. Later, our expert panel has been asked to prepare a questionnaire based on this question: *What do you ask from a patient who visits you because he/she suffers from Dry Mouth?*

Questions on the questionnaire were then categorized based on *open*EHR concepts; in other words, their logical relation e.g. is the question related to patient history or is it a lab result? In the next step, simple diagrams were created based on the questionnaire. For this purpose, a mind-map application[3] was used to make it possible for our expert panel to simply understand and edit the created diagrams.

GUI and Domain Concept Model Evaluation

For the GUI evaluation, we used the evaluation methods applicable in early stages of the project. Two main methods we have been utilizing so far are *Heuristic Evaluation* and *Usability Tests* [9]. Based on the results from the evaluations we improved the GUI in several stages. One of the resulting GUI screens is showed in Figure 2.

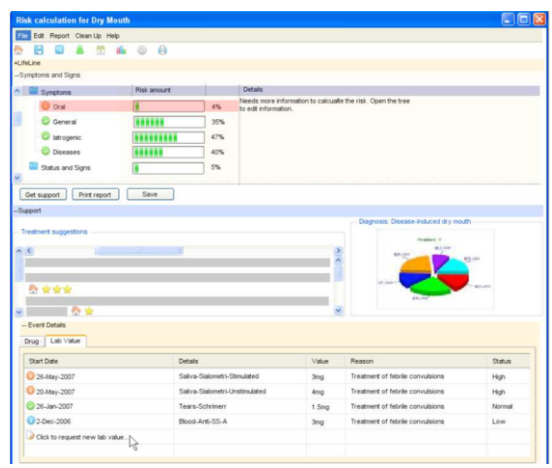

Figure 2- GUI Prototype

Iterative design of the domain concept model includes evaluations of the current model based on the literature and experts' opinions, and story-based assessment. Information modeling diagrams were improved several times based on the experts' opinions. Several experts were involved in this process to minimize the subjectivity of the design and to be as broad as possible in collecting knowledge. A sample mind map is depicted in Figure 3.

[3] http://www.xmind.net/

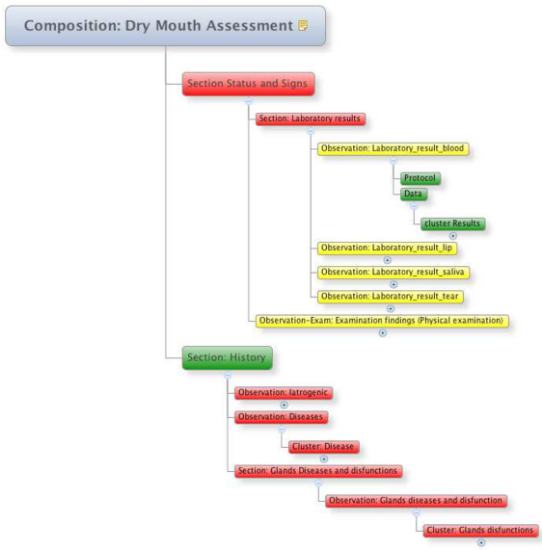

Figure 3- Information modeling

Results and Discussion

UCD emphasizes that users' needs should be reflected in the GUI design and that the GUI design should influence the design of the rest of the system [12]. On the other hand, *open*EHR emphasizes the domain concept modeling as the starting point. But how much does the domain model reflect the end users' needs? By using the *open*EHR approach without considering other aspects like usability issues, we may end up developing highly adaptable systems with comprehensive information models, which are not usable.

In this project, we tried to benefit from the strengths of the two of approaches and to introduce an adaptation of UCD in a clinical concept keeping an eye on the *open*EHR approach.

The Customized UCD Approach for *open*EHR Based CDSS Development

As references suggest "The actual contents of the UCD process, the methods used, the order of activities, etc, must be customized and adapted to the particular organization and project based on their particular needs" [19]. So it was not a surprise to see that we need to apply a customized version of UCD in this project.

As shown in Figure 4, the main idea of UCD is used in the process but in three different cycles. One is a general cycle to develop the whole application. This main cycle contains a cycle to develop the Domain Concept Model; and a cycle to develop the GUI. So the process includes two main steps in parallel (1) Iterative development of the domain concept model (2) Iterative development of the GUI. For the first step, several specialists in dentistry (expert panel) and for the second step, both domain experts and general dentists (end user panel) were involved.

GUI vs. Domain Concept Model

During the iterative design process we noticed that the impact of the domain concept model on the GUI is inevitable. Decision about the components to be shown on the GUI is directly related to the output from the domain concept modeling. Any changes in the domain concept model should be checked from the GUI point of view. Therefore, as depicted in Figure 3, in each iteration, there should be an input from the left hand side process (domain concept model) to the right hand one (GUI). In other words, after each domain concept modeling iteration, the necessity of a new iteration for GUI design should be checked.

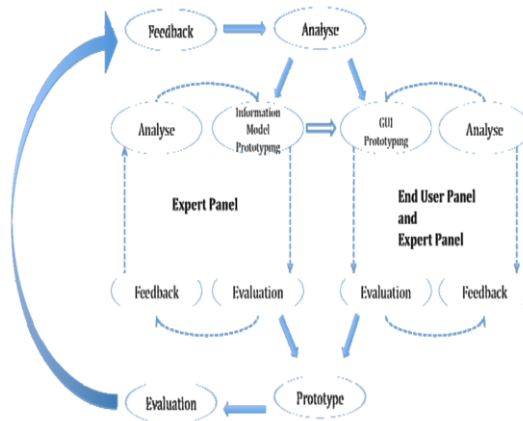

Figure 4- Customized User Centered Design Process

Characteristics of the Customized Approach and prblems

The recommended approach has several characteristics:

- This approach considers active involvement of the end users and domain experts in designing and evaluating the domain concept model and the GUI
- In the suggested approach new GUI and Domain Concept modeling iterations will be performed until the end users are satisfied with the results.
- In this approach, the effect of the domain concept model on the GUI has been considered as explained before.
- The approach helps overcoming the knowledge extraction bottleneck by applying clinical concepts suggested by *open*EHR for communicating with clinicians and providing an opportunity for them to model domain knowledge based on their expertise.
- This approach inherits the idea of the knowledge and the information level separation suggested by *open*EHR in order to make developed applications highly adaptable.
- Finally, the approach is applicable for developing not only *open*EHR-based applications but also all kinds of

clinical applications e.g. OWL based ones. Important issues to be considered while applying the approach are (I) the parallel iterative UCD of the domain concept model and the GUI, and (II) the effect of the domain concept models on the GUI.

We also faced some problems during the design phase. In design and implementation of CDSS:s a big challenge is choosing a knowledge representation and reasoning. Our experience showed that, selecting the representation and reasoning method have to be done in parallel with the information modeling and the GUI design, otherwise, the changes forced by this selection causes modifications in the GUI design which is more cost effective to be known in the early stages of the GUI design. Secondary, the classical bottleneck of knowledge acquisition in clinical domain still exists. While applying the suggested methodology decreases difficulties in mutual understanding of clinicians and designers, it cannot eliminate the bottleneck problem totally, especially for the cases that reasoning should be done by applying knowledge intensive methods.

Acknowledgements

Scincere thanks go to Ian McNicoll, Soren Lauesen, and Dowen Birkheld for evaluating domain concept models and the GUI. Many thanks also go to Mats Jontell, Marie Lindgren and Göran Falkman, our team members; and to my love Mohsen Nosratinia, and to Anna Gryszkiewicz for proofreading this paper. The author would also like to express her gratitude to Olof Torgersson under whose supervision this work has been done as a part of the author's PhD study. The project was funded by the Swedish Governmental Agency for Innovation Systems (VINNOVA), grant 2006-02792, as a joint project between Chalmers University of Technology and Sahlgrenska Academy, and is in progress at the time of writing this paper.

References

[1] Graham Ta, Kushniruk AW, Bullard MJ, Holroyd BR, Meurer DP, Rowe BH. How usability of a web-based clinical decision support system has the potential to contribute to adverse medical events. AMIA Symposium, 2008 ;257-61.

[2] Vikram K, Karjodkar FR. Decision support systems in dental decision making: an introduction. The journal of evidence-based dental practice, 2009 ;9(2):73-6.

[3] Kaplan B. Evaluating informatics applications-clinical decision support systems literature review. International journal of medical informatics, 2001 ;64(1):15-37.

[4] Organization IS. ISO 9421-11. Standard on Display Screen (VDU) Regulations, Use of Ergonomics for Procurement and Design. Geneva, Swiss, 1998.

[5] Ying-Jui C, Chirh-Yun AT, Min-Li BY, Yu-Chuan BA. Assessing the Impact of User Interfaces to the Usability of a Clinical decision support system. AIMIA Symposium, 2003;808.

[6] Saleem JJ, Patterson ES, Militello L, Render ML, Orshansky G, Asch SM. Exploring barriers and facilitators to the use of computerized clinical reminders. JAMIA, 2005 ;438-47.

[7] Han YY, Carcillo JA, Venkataraman ST, Clark RS, Watson RS, Nguyen TC, Bayir H, Orr RA. Unexpected Increased Mortality After Implementation of a Commercially Sold Computerized Physician Order Entry System. Pediatrics, 2005 ;116(6):1506-12.

[8] Koppel R, Metlay JP, Cohen A, Abaluck B, Localio AR, Kimmel SE, Strom BL. Role of computerized physician order entry systems in facilitating medication errors. JAMA, 2005 ;293(10):1197-1203.

[9] Vredenburg K, Isensee S, Righi C. User-Centered Design: An Integrated Approach. Prentice Hall PTR, Upper Saddle River, NJ, 2002.

[10]Organization IS. ISO 13407. Human Centered Design Process for Interactive Systems. Geneva, Swiss, 1999.

[11]User-Centered Design. IBM. Available from: https://www-01.ibm.com/software/ucd/ucd.html

[12]Norman DA, Draper SW. User Centered System Design: New perspectives in Human-computer Interaction. L. Erlbaum Associates Inc. Hillsdale, NJ, USA, 1986.

[13]Marcy TW, Kaplan B, Connolly SW, Michel G, Shiffman RN, Flynn BS. Developing a decision support system for tobacco use counselling using primary care physicians. Informatics in primary care, 2008 ;16(2):101-9.

[14]Wang F, Hannafin MJ. Design-based research and technology-enhanced learning environments. Educational Technology Research and Development, 2005;53(4): 5-23.

[15]Porter SR, Rcs FD, Rcse FD, Scully C, Rcps FD. Etiology and management of Xerostomia. *AJP*, 2004;97(1):28-46.

[16]Jontell M, Mattsson U, Torgersson O. MedView: an instrument for clinical research and education in oral medicine. Oral surgery, oral medicine, oral pathology, oral radiology, and endodonlogy, 2005 ;99(1):55-63.

[17]Beale T, Heard S. *open*EHR Architecture Overview. Available from: http://www.openehr.org

[18]Gulliksen J, Göransson B, Boivie I, Blomkvist S, Persson J, Cajander A. Key principles for user-centred systems design. Behaviour & Information Technology, 2003;22(6):397-409.

[19]Horasani RA, Anasijevic MI, Iddleton BL, C MS. Ten Commandments for Effective Clinical Decision Support: Making the Practice of Evidence-based Medicine a Reality. JAMIA, 2003 ;10(6):523-30.

Address for correspondence

Department of Computer Science and Engineering, Chalmers University of Technology, SE-412 96 Gothenburg, Sweden. Email: hajar.kashfi@chalmers.se.