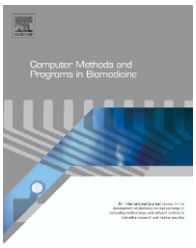
journal homepage: www.intl.elsevierhealth.com/journals/cmpb



User-centered requirements engineering in health information systems: A study in the hemophilia field

Leonor Teixeira[a,b,e,*], *Carlos Ferreira*[a,c], *Beatriz Sousa Santos*[d,e]

[a] *Department of Economics, Management and Industrial Engineering, University of Aveiro, Portugal*
[b] *Governance, Competitiveness and Public Politics (GOVCOPP), University of Aveiro, Portugal*
[c] *Operational Research Centre (CIO), University of Lisbon, Lisbon, Portugal*
[d] *Department of Electronics, Telecommunications and Informatics, University of Aveiro, Aveiro, Portugal*
[e] *Institute of Electronics and Telematics Engineering of Aveiro (IEETA), Aveiro, Portugal*

ARTICLE INFO

Article history:
Received 16 May 2009
Received in revised form
5 October 2010
Accepted 14 October 2010

Keywords:
Health information systems
Requirements engineering
User-centered design
Human-factor engineering system
Triangulation matrix
Hemophilia

ABSTRACT

The use of sophisticated information and communication technologies (ICTs) in the health care domain is a way to improve the quality of services. However, there are also hazards associated with the introduction of ICTs in this domain and a great number of projects have failed due to the lack of systematic consideration of human and other non-technology issues throughout the design or implementation process, particularly in the requirements engineering process.

This paper presents the methodological approach followed in the design process of a web-based information system (WbIS) for managing the clinical information in hemophilia care, which integrates the values and practices of user-centered design (UCD) activities into the principles of software engineering, particularly in the phase of requirements engineering (RE). This process followed a paradigm that combines a grounded theory for data collection with an evolutionary design based on constant development and refinement of the generic domain model using three well-known methodological approaches: (a) object-oriented system analysis; (b) task analysis; and, (c) prototyping, in a triangulation work. This approach seems to be a good solution for the requirements engineering process in this particular case of the health care domain, since the inherent weaknesses of individual methods are reduced, and emergent requirements are easier to elicit. Moreover, the requirements triangulation matrix gives the opportunity to look across the results of all used methods and decide what requirements are critical for the system success.

1. Introduction

The use of sophisticated information and communication technologies (ICTs) in the health care domain is a way to improve the quality of services. The literature points to a general consensus that health information systems (HISs) are thought to have the potential to improve patient care. However, there are also hazards associated with the introduction of ICTs in health care, and some works report how difficult it is the successful introduction of ICTs in this domain [1,2]. In fact, health care is a unique and complex domain and HISs have human safety implications and profound effects on individual patient care [3].

* *Corresponding author at*: University of Aveiro, Campus Universitário de Santiago, 3810-193 Aveiro, Portugal. Tel.: +351 234 370 361; fax: +351 234 370 215.
E-mail address: lteixeira@ua.pt (L. Teixeira).
0169-2607/$ – see front matter © 2010 Elsevier Ireland Ltd. All rights reserved.
doi:10.1016/j.cmpb.2010.10.007

Successful development of HISs can increase efficiency and productivity, ease of use and learning, adoption, retention and satisfaction of the users [2], and simultaneously, can help to decrease medical errors as well as to reduce support and training cost [4]. On the other hand, HISs are usually complex systems and their failures may cause negative effects on patients [5–7] and possibly, when insufficiently designed, they may result in spending more time with the computer than with the patient [8]. According to [9] ICTs have been hailed as a solution to reduce errors in health care, but there is also evidence that they can be part of the problem.

There are a large number of HISs projects that have failed, and most of these failures are not due to flawed technology, but rather due to the lack of systematic consideration of human and other non-technology issues throughout the design or implementation process [10–12]. Also, several studies have shown that 80% of total maintenance costs with information systems (ISs) are related to users' problems and not technical bugs, and among them 64% are related with usability problems [13]. A survey of over 8000 projects made by *The Standish Group* and undertaken by 350 US different companies revealed that one third of the projects were never completed and one half succeeded with partial functionalities. The major source of such failures resided on poor requirements, specifically: the lack of user involvement (13%), incomplete requirements (12%), changing requirements (11%), unrealistic expectations (6%), and unclear objectives (5%) [14]. These problems are mainly due to the fact that, in developing interactive software, most software engineering methodologies do not propose any mechanisms to: (i) explicitly and empirically identify and specify user needs and usability requirements; and (ii) test and validate requirements with end-users before and during the development process [15]. The health care domain has been particularly prone to such problems in recent years, and there are numerous examples of potentially useful systems that have failed or have been abandoned due to unanticipated human or organizational issues [16–18].

Since the design of systems that are used by people is a complex endeavor, the systems that cannot be used intuitively often lead to an increase in error rate and a decrease in user acceptance [19]. Involving end-users in the design process has been suggested to be a key marginal investment for being able to transform the cost related to the implementation of HISs into future benefits [20–22]. Moreover, the participation of end-users and the involvement of relevant stakeholders in early steps of the design process, not only prevents post-implementation problems, but also gives them the chance to address and resolve potential conflicts concerning the future system [23]. However, involving end-users in development of HISs is often complicated, especially when they have limited computer skills or due to the fact that users' knowledge is tacit and consequently task description is very difficult.

Particularly in healthcare, effective research of user requirements is often discussed in terms of which method to use or "which method is better", since the dominant culture in this industry is still train people to adapt to poorly designed technology, rather than designing technology to fit to people's characteristics [11].

The principles of user-centered design (UCD) [4,24] combined with ethnographic practices [25] can improve synergies among technology, people and work environment (tasks).

In order to perform a user needs analysis and to write requirements specification for integrated care in the hemophilia field, we followed a user-centered requirement engineering process involving the end-users through different techniques of requirements elicitation and validation. This work describes the application of UCD principles in the requirements engineering process to generate the requirements document of a web-based information system (WbIS) for managing the clinical information in hemophilia care, as well as some results. The research was based on three well-known methodological approaches: the first one consisted of a classical object-oriented systems analysis (OOSA) method based on *Unified Modeling Language* (UML) notation; the second consisted in a task analysis (TA) method based on hierarchical task analysis (HTA) notation; and the third consisted in prototyping based on the construction of an executable system model. These three methods were performed through an iterative development process combined with grounded theory (GT) in a triangulation work.

2. Background

2.1. *Requirements engineering process*

The success of any software system depends on how well it fits the needs of its users and its environment [26]. Software requirements comprise these needs, and requirements engineering (RE) is the process by which the requirements are determined [27]. A requirement is a property that a system must exhibit in order to meet the system's motivation need; and software requirements are a property which must be exhibited by software developed to solve a particular problem within one organizational context. Therefore the software requirements are a complex combination of requirements from different people at different levels of an organization and from the environment in which the system must execute [28]. They express the needs and constraints placed on a software product that contribute to the solution of some real-world problem and normally result of an arrangement between "user requirements" and "system requirements", Fig. 1.

User requirements denote the requisites of the people who will be the system client or end-users. System requirements add requirements of other stakeholders (such as regulatory authorities) and requirements that do not have an identifiable human source and that normally result from the intersection among technical, cultural and social environments [28,29]. The information system (IS) is never used on its own but always as part of some broader system including hardware, people and, often, organizations [30].

Successful RE involves understanding the needs of users, clients and other stakeholders, as well as understanding the context in which the software will be used [27]. Thus, identifying all the users and other stakeholders who may be impacted by the system is very important and this will help to ensure that the needs of all those involved are taken into account and, if required, the system is tested by them.