Placeholder title

Master thesis Computer Science

Dennis Cardinaels

Promotor: Professor Mieke Haesen
Assistant: Jens Brulmans

2017 - 2018

Preface

TODO

Abstract

TODO

Contents

1 Introduction

The use of software solutions in health care has been more prevalent than ever before and has become an integral part of the daily workflow of clinicians. These solutions range from monolithic health information systems to the combined use of multiple standalone applications. Because of the rapid evolving nature of technology, health care institutions have to undergo frequent change. This requires clinicians to adapt to new software and devices. Time spent on providing optimal care therefore diminishes, while time spent on educating staff increases.

This thesis proposes a dashboard design that allows the caregiver to personalize its functionality according to patient's needs in terms of treatment. First, to get a thorough understanding of related subjects, a background information will be give in section 2. Based on this background information a design is proposed (section 3).

2 Background

A literature study has been conducted to situate the topic. Multiple subjects lay at the base of the design described in section 3. Of these subjects, three are key which are explained first: electronic health record systems, telemonitoring and dashboard design. The process followed to create the final prototype is explained hereafter. Finally, topics that should be kept in mind for further development are briefly described, as they are not part of the scope of the prototype.

2.1 Electronic health record systems

Health information systems are becoming an important part of health care. They not only support patient care, but also administrative and financial tools. At the heart of these systems lies the electronic health record. An electronic health record (EHR) is a repository of electronically maintained information about an individual's health status and health care, stored such that it can serve multiple legitimate uses and users of the record [1]. An electronic health record system (EHRS) provides tools to manage and interact with these records. These tools include providing reminders, data analysis and decision support. It helps the clinician to organize, interpret and react to data.

First, the advantages of EHRS over paper-based records are described to signify its importance. Secondly, the main components present in these systems is summarized. The functionality of an EHRS can be categorized into two types: a monolithic system tries to provide more general care, while smaller and more specialized systems cater to more specific types of care. The latter however, lacks certain functionalities and data, which requires the combined use of multiple systems. The last section compares the two types and highlights the benefits and drawbacks of both.

2.1.1 Moving away from paper

For modern medicine, the traditional paper-based medical records are not suited in today's world filled with technology. The drawbacks of information on paper are obvious when compared to digitally stored information.

Functionality Digital records allow systems to aggregate, process and generate statistics of the data it contains. For example, heart rate graphs with typical statistics can be generated and when the users hovers over a data point in the graph, the exact value is shown. Paper records containing tables or printed graphs that show the same data are limited in this regard.

A paper-based medical dossier can only store medical images, such as x-rays, but compared to a digital image, a lot of detail is lost. As such, other multimedia and other file types are can not be stored. With an EHRS, these file types can be attached with ease, provided the system supports them. Paired with the tools of the system, a lot more functionality can be achieved.

In terms of data input, an EHRS can detect and prevent false data input. The system can make sure all necessary data fields filled in and in the correct format. This results in more complete and accurate data gathering with less errors.

Information quality A summarizing paper noted that the use of EHRS leads to more complete, accurate, comprehensive and reliable data compared to paper-based records [2]. As mentioned in the previous paragraph, a digital system can impose rules on data fields to avoid missing or entering wrong data. In terms of comprehensibility, poor handwriting leads to wrong or loss of information, which is avoided on digital systems.

Accessibility Paper records are difficult to access because most of the time only one copy exists. Transferring these records to other branches or institutions therefore is difficult: the record has to be found in the often large medical dossier, has to be copied and then manually sent. Also, misfiling, flooding or fires can lead to irrecoverable loss of the data. Digital records avoids the last issue, but there can still be difficulties regarding data transfer as most institutions have their own database structures.

The process of moving data on paper to a digital format requires a lot of manual work. While some aspects can be automated, such as scanning and processing the paper forms with software, verification is still necessary. All the data fields from the documents need to find a place in the EHR, which frequently is not possible. Also, what do we do with unreadable data due to poor handwriting? What happens with duplicate forms with slightly different values? Because of these reasons, adopting an EHRS is a significant undertaking for an institution. As such, the benefits are not immediately apparent.

Because digital storage increases accessibility significantly, security measurements have to be taken. If not, data can be stolen, deleted or even altered. This also adds to the complexity of developing EHRS.

Efficiency An important task of an EHRS is to facilitate efficient care. An early study saw a 6% increase in productivity when EHRS were deployed in health care institutions [3]. However, other factors are at play, such as adoption rate. The transition from paper-based documenting to digital is accompanied with a learning curve, which can be steep. After the switch, the productivity will be lower at the start and as the users become more experienced with the software, it will increase.

2.1.2 Components of an EHRS

As mentioned before, EHRS do not simply store patient records. They consist of many components which ultimately define how well they perform in health care. Literature gives us multiple definitions of which components are essential. However, we summarize them in the following five [1]:

Integrated view of patient data An EHR must allow storage of a wide range of data types. This can be text, numbers, images, video and others. Some data can still be on paper due to lacking support of the EHRS, as mentioned in section 2.1.1. To display more complex data types, such as x-ray images, standards are used. A brief overview of such standards can be found in section 2.6.

Clinician order entry Order entry is the point at which the clinician has to make decisions or take action. An order entry system can assist the clinician by providing decision support. It also reduces errors and costs compared to paper order entry.

Clinical decision support A decision support system embed into an EHRS can aid the clinician by suggesting actions to take, when certain situations occur. If for example a patient is due for a vaccination, the system can notify the clinician by presenting a constructed order which needs to be confirmed or denied. The system can do this for a bulk of patients, so manual checkups are not required, thus saving time.

Access to knowledge resources When clinicians are writing notes or orders for a patient, clinical questions can arise. Instead of asking colleagues, the EHRS can pull literature to address the question. Due to the internet, a very large source of information is readily available.

Integrated communication and reporting support Communication is an important part in health care. Often clinicians of multiple institutions provide care for a particular patient. The effectiveness of patient care therefore is directly affected by communication which is why EHRS should provide tools to assist the clinicians. Most institutions are bounded by their own EHRS, which requires them to ask the other institutions for necessary data. A solution to this problem are Health Information Exchanges, which allows institutions to reach data beyond their own EHRS.

Throughout the years, many EHRS have been developed which may or may not integrate all of the above components. Should an essential component be missing, an institution may add another system. Although this has its benefits and drawbacks.

2.1.3 Monolithic vs. multiple EHRS

Health care institutions can opt for a monolithic EHRS or combine multiple EHRS to achieve all required functionality. To both there advantages and drawbacks [4]. Elements influencing this choice include IT infrastructure, safety risks, volume of care and frequency with which patients move facilities. However, it's possible that a single EHRS does not satisfy the requirements of an institution.

A reason for this is that certain branches require specially tailored software for their clinical practice, which a general EHRS lacks.

Functionality wise, a monolithic EHRS tends to appeal to more general types of care, whereas it lacks in very specific ones. Also, vendors that offer these all-in-one solutions have less experience with these specialties which reduces the chance it will be added to the system. Vendors of specific EHRS do have this expertise and can tailor the system to the needs of the customer. In this case, combining a system that supports general care with special care systems will seem like the best choice. However, other considerations have to be made.

The advantages of a monolithic system is that the data it uses is centralized. This ensures that all data can be accessible anywhere in the system and that it can be easier maintained. When multiple systems are in place, data has to be exchanged and can lead to difficulties finding certain pieces of information. The exchange of data is difficult, as each system probably has its own data structures. A potential solution is to create a new EHRS, solely for the purpose data collection and transformation with which all systems interface with. This leads to another system, requiring additional development and maintenance time.

2.2 Telemonitoring

Monitoring patients at a distance with the use of information technology is called telemonitoring [5]. The patient uses special monitoring devices to gather data which is sent to the health care provider. Telemonitoring allows the reduction of chronic disease complications as a result of better followup, provides health care without using hospital beds. It also reduces patient travel, time away from work and overall costs [6]. While many health conditions can be monitored, telemonitoring is most prevalent in chronic diseases. These diseases are the major cause of death in most countries and require many health care resources and costs.

2.2.1 Devices & health parameters

2.3 Dashboards

The primary goal of a dashboard is clear communication, which is achieved with an effective design. A formal definition is as follows: “A dashboard is a visual display of the most important information need to achieve one or more objectives; consolidated and arranged on a single screen so information can be monitored at a glance.” [7]. This definition mentions four key characteristics:

Dashboards are visual displays. It is important that data is represented visually. Displaying charts and other graphics allows more efficient communication compared to textual information. For example, trends and outliers are easier to spot in a line chart compared to a table with the same values.

Dashboards display information needed to achieve specific objectives. The data that is shown, must be of use for the job that needs to be done. It is possible that the data needs to be gathered from multiple sources and tailored to the specific context so an efficient visualization can be created.

A dashboard fits on a single computer screen. In order to see all information at a glance, scrolling must be prevented. If multiple screens are present, then it is no longer a single dashboard. This leads to another question: what type of display is the information shown on? Due to the wide variety of screen sizes and aspect ratios, a responsive dashboard would be most beneficial.

Dashboards are used to monitor information at a glance. Data that is important should be immediately noticed, whereas very specific details should be hidden. This means data must be summarized or aggregated in order to be effectively shown. However, if the user wishes to view the detailed data, the dashboard should provide a way to do so.

To create an effective dashboard, the user-base must be known and well understood.

Another important factor is the type of user. Some users can understand one visualization better than the other. To create an effective dashboard, the focus should be put on the user [8].

Many EHRS fail to provide a meaningful dashboard. We can only look at open-source systems as most health software is purchased and licensed.

2.4 MuiCSer

This section describes the MuiCSer framework, made for user-centred software engineering processes in a multidisciplinary context [9], which will be used to create the prototype. This framework focuses on optimizing the user experience during the entire software engineering cycle to ensure that the end-user's needs are fulfilled. By combining user-centred design and software engineering principles the user experience of the final product can be improved substantially of the final product.

2.4.1 Process

The MuiCSer process is summarized in figure 1. After each phase, the result is evaluated, verified and validated to ensure that the required functionality is present. The received feedback can in turn be used to reiterate over the previous phase. On the figure this is denoted with the light arrows, while the dark one represents the overall process direction.

New or legacy system At the start of the process an existing system in need of improvement is either evaluated or a new one has to be designed. This requires an analysis of the tasks and needs of the user, as well as the objects

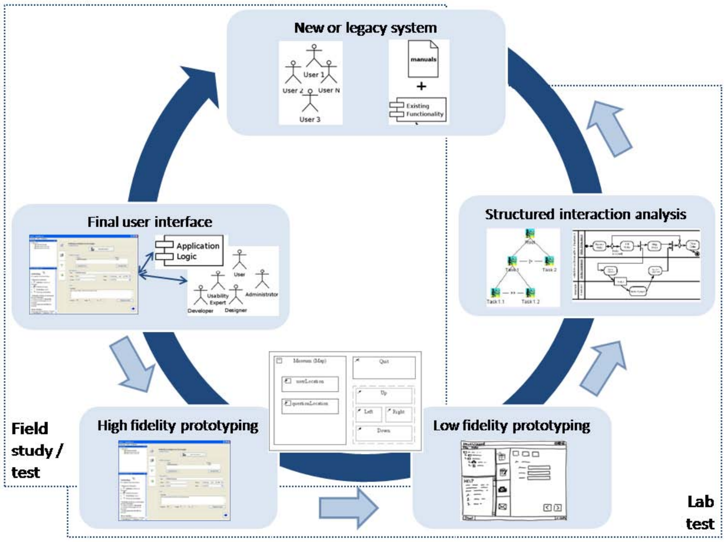

Figure 1: MuiCSer process

and resources required to perform these tasks. Personas and scenarios are the resulting artifacts of this phase. First, personas describe the personalities of the potential end users including hobbies, skills and the environment they surround themselves in [10]. Its goal is to uncover behavior patterns which can be of use when designing a user interface. Second, a scenario is a story describing the use of a fictitious system from the persona's point of view [10]. It tries to sketch the usage of the system for which a design must be made.

Structured interaction analysis During this phase, the results of the analysis are used to create task models. These models specify concrete tasks and goals which can be dissected into specific actions or steps the user has to take. These artifacts lay the foundation for designing a user interface which supports these tasks and goals.

Low fidelity prototyping When the actions have been specified using the task models, low fidelity prototypes can be designed. Paper sketches and mockups are such examples and are ideal for visualizing the layout of the software its user interface. Without spending too much time and resources, presenting such prototypes can yield valuable feedback from the end-user or customer. However, there is no interaction present. Typically multiple versions of these prototypes will be created until the customer is satisfied after which high fidelity prototypes can be developed.

High fidelity prototyping Creating high fidelity prototypes requires a lot more effort compared to low fidelity prototypes, as they offer functionality closely resembling the final product. However, the feedback will be much more valuable as not only design, but also functionality is tested.

Final user interface When the latest iteration of the high fidelity prototype satisfies all user requirements, the final user interface can be created. It would be beneficial to reuse the code from the prototype in order to save time and resources. As a final step, the task models are checked against the interface to check if all required functionality is present.

2.5 Customization

2.6 Standards

2.7 Privacy

Although not being a main focus, it is important to highlight the privacy concerns regarding health information data.

3 Design

A design has been created using the MuiCSer framework explained in section 2.4. The final design went through multiple iterations based on received feedback.

3.1 Proposal

3.2 Personas & scenarios

Based on the proposal mentioned in the previous section, personas and scenarios have been created to get a better understanding of how the system will work and its users. Each of these try to highlight the problems the users face and how a new system tries to solve them.

3.2.1 Jake

Persona Jake is a 25-year-old male who currently works as a nurse at the hospital in his city. He has been working for 4 years for the hospital and lives alone in his apartment. Still being a young adult, Jake grew up with technology. As such he experiences no difficulties when using new software on his computer or smartphone. His hobbies include music, playing the guitar and video gaming.

Currently, the workflow at the hospital is dated. A new health information system was introduced to summarize and gather all medical data in a singular space. Because this is a monolithic system, a lot of features that Jake does not need still clutter the screen. Navigating the system is a pain and customization is not present. Jake wishes to only see the features he uses most, while hiding the features he does not need.

Scenario Jake starts his first day using the new system by reading the manual that is accompanied with it. He starts the application and for each patient he is presented with a few standard layouts to choose from, based on illnesses: cardiovascular diseases, respiratory diseases, diabetes and few others.

After selecting a standard layout, Jake is given the opportunity to customize the dashboard. The dashboard contains all the wanted functionality of the system, where each “block” represents a module that can be added or deleted. Jake enters the edit mode enabling him to move the blocks around and arrange the order. He deletes a few modules and now he wants to add other modules. Jake opens the modules window where he is presented with all the available modules. A module is added to the dashboard when Jake selects it.

By examining the new module Jake quickly notices that on the dashboard a summary is given. But when Jake clicks on the module, the block expands to fill the whole screen where detailed information is given. Jake feels he is in control of the system and that it will improve his productivity.

This scenario highlights the benefits of customization. Hiding unwanted components, while adding the useful ones allow for a clear dashboard to be displayed.

The user is in control and is able to quickly view data of importance. It is also possible to view detailed data.

3.2.2 Dan

Persona Dan is a general practitioner since he graduated from university. He is on the job for 21 years and he is the preferred doctor in his town. Throughout the years Dan has used a multitude of systems and he always tries new ones to improve his workflow. Because he has been practitioning for such a long time, he has 500 patients that visit him at least twice a year. Some patients, especially the elderly, visit as much as once a month.

Most of Dan's patients visit for illnesses such as fever and a cold. To diagnose these illnesses no data is necessarily needed, just a description of what the patient feels will suffice most of the time. If Dan performs such a diagnosis, he promptly adds it to the information system and to the electronic health record of the patient.

However, if a patient visits that has a lot of problems regarding his blood pressure, then Dan has to perform a more complex diagnosis using historical data. In the current system that Dan uses, it is very difficult to search for this data. But what bugs Dan the most is that he has to do it every time this patient visits.

Scenario Because of Dan's ongoing curiosity, he tries a new health information system which allows customization for each specific patient. Because the diagnoses of most patients can be very different from time to time, Dan creates a default module group which is displayed for each patient, unless that patient has a specific module configuration. The default module group includes: past diagnoses, known allergies, patient information (blood type, last weight, height...) and medication list.

The first patient of the day describes what sounds like a fever. Dan confirms this and prescribes the patient some medicine. The diagnosis is added to the 'past diagnoses' module and the prescribed medication to the 'medication list' module. Dan did not need other health data to perform the diagnosis. Therefore, Dan does not change the configuration of this patient.

The next patient, an elderly woman, came for her second visit of this month. Dan knows from the past that it will probably be a heart problem. Dan searches for other modules labeled as 'heart problems' and adds them to the configuration of the elderly woman. Now both the default module group and the heart rate modules are present, which is unnecessary according to Dan. He removes some modules of the default module group. Now, the next time the elderly woman visits, that configuration will be loaded.

One specific patient had broken his arm three times in less than a year. When the patient came for a routine visit, Dan immediately added a module to easily view x-ray photos and view them in a timeline.

Again, the benefits of system customization for each patient are obvious. Initially, it will take some time to configure each dashboard for all of the patients, but the system will try to provide default layouts useful for certain types of care. Once the configuration is done, the time spent during a consultation will decrease.

3.3 Modules

4 Implementation

high fidelity

4.1 Technology stack

4.2 Overview

5 Usability test

A usability test has been conducted to test the effectiveness of the high-fidelity prototype. The test involves users filling in questionnaires before and after using the prototype. The tester was required to follow a predefined list of steps while using the prototype. Only once the final step was completed, the test was finished.

The following section will first describe in detail the setup of the whole usability test, such as the followed process, the documents (questionnaires, list of steps), the testing environment and information concerning the testers. Thereafter, the results of the test are shared and analyzed.

5.1 Process

5.2 Result

6 Discussion

TODO

7 Conclusion

todo

A Appendix

appendix

References

[1] Edward H Shortliffe and James J Cimino. *Biomedical informatics*. Springer, 2006.

[2] Kristiina Häyrinen, Kaija Saranto, and Pirkko Nykänen. Definition, structure, content, use and impacts of electronic health records: a review of the research literature. *International journal of medical informatics*, 77(5):291–304, 2008.

[3] Dwight C Evans, W Paul Nichol, and Jonathan B Perlin. Effect of the implementation of an enterprise-wide electronic health record on productivity in the veterans health administration. *Health Economics, Policy and Law*, 1(2):163–169, 2006.

[4] T Payne, J Fellner, C Dugowson, D Liebovitz, and G Fletcher. Use of more than one electronic medical record system within a single health care organization. *Applied clinical informatics*, 3(04):462–474, 2012.

[5] Guy Paré, Mirou Jaana, and Claude Sicotte. Systematic review of home telemonitoring for chronic diseases: the evidence base. *Journal of the American Medical Informatics Association*, 14(3):269–277, 2007.

[6] Stephane Meystre. The current state of telemonitoring: a comment on the literature. *Telemedicine Journal & e-Health*, 11(1):63–69, 2005.

[7] Stephen Few. Information dashboard design. 2006.

[8] Richard Brath and Michael Peters. Dashboard design: Why design is important. *DM Direct*, pages 1011285–1, 2004.

[9] Mieke Haesen, Karin Coninx, Jan Van den Bergh, and Kris Luyten. Muicser: A process framework for multi-disciplinary user-centred software engineering processes. In *Engineering Interactive Systems*, pages 150–165. Springer, 2008.

[10] Sabine Madsen and Lene Nielsen. Exploring persona-scenarios-using storytelling to create design ideas. In *Human work interaction design: Usability in social, cultural and organizational contexts*, pages 57–66. Springer, 2010.