e-Health – For Continuity of Care
C. Lovis et al. (Eds.)

doi:10.3233/978-1-61499-432-9-768

Towards a Trial-Ready Mobile Patient Questionnaire System

Michael STORCK[a,*,1], Benjamin TRINCZEK[a,*], Martin DUGAS[a] and Fleur FRITZ[a]
[a]*Institute of Medical Informatics, University of Münster, Münster, Germany*

Abstract. Gathering patient-reported data enables physicians to know a patient and his/her health-related needs in a more comprehensive way. One method to collect these data is to let the patient fill out electronic questionnaires on a mobile device, e.g. a tablet. Although having successfully implemented a prototype, the mobile questionnaire system at this site has been stretched to its limits within both routine care and clinical trials. By collecting user feedback and analysing the capabilities and limits of the underlying framework, we identified key issues of the prototype. A new implementation addresses these obstacles while keeping the overall application logic and usability. This leads to a trial-ready mobile patient questionnaire system.

Keywords. Questionnaires, Quality of Life, m-health, patient reported outcomes, medical documentation

Introduction

To guarantee the best treatment in patient-centred care settings, medical experts need to know not only a patient's clinical picture, but also his/her social context and preferences, which implies collecting all relevant information. Unfortunately, due to the increasing workload of clinicians, the available time at the bedside and thus the possibility to collect this information is decreasing.
Patient Reported Outcomes (PRO), i.e. patient data gathered and provided by him-/herself or the patient's relatives without the physician's assistance, can be helpful to get a better understanding of the patient's context and his/her perception. This information is useful to find the precise diagnosis and the best therapy for the individual patient, which can lead to improved, personalized healthcare.

Gathering patient-based data is usually done via paper-based questionnaires that are filled out by him/her in the waiting room. There are two different approaches to transfer the questionnaire results into the Electronic Health Record (EHR). First a medical staff member can typewrite it into the EHR, to make it electronically available for both routine care and research purposes (following the Single Source approach [1]). The second approach is to scan the paper-based questionnaire and make it available within the EHR documentation of the patient. The approaches contain several drawbacks: typewriting or interpretation errors might occur, the data of scanned questionnaires is not accessible in a structured way, delivery of patient data to the physician is delayed and the task itself is both cumbersome and cost-ineffective. To

[1] Corresponding Author.
* equally contributed

overcome the media disruption, some projects already implement an electronic questionnaire system that allows the patient to answer a digital survey [2]. At this site, such a system called "Mobile Patient questionnaire (MoPat)" has been implemented prototypically in 2010 and was evaluated regarding usability and cost-efficiency [3]. The implementation provides web-based patient questionnaires. In contrast to other systems, completed questionnaires are exported to the EHR System instantaneously in a data format that is both re-usable and compliant to the EHR system's data structure.

Although being successfully deployed, the prototype has been stretched to its limits, not only because of the recent security issues of the underlying programming framework [4], but also in terms of infrastructural conditions. To make the system future-proof and scalable, the existing prototype, its usage and the underlying infrastructure have been analysed and changed accordingly. This manuscript describes the issues identified and the newly implemented solutions.

1. Methods

During usage of the MoPat system in the participating clinics, we asked ten clinicians about occurring problems and feature requests for additional scenarios. Either direct, open interviews via telephone or email have been conducted. The interviews were unstructured to give the users the possibility to describe their impressions and features comprehensively. The information given by the users were analysed and the issues and feature requests were prioritized by importance.

After gathering the user's feedback, we conducted code reviews to estimate the amount of work that had to be invested to solve the issues addressed and features requested. The code review considered recent issues like security, feature potential of the underlying programming framework and the staff's knowledge about it.

After analyzing on the one hand the user requests and on the other hand the potential of the programming framework, we decided to develop a new version of MoPat. As the use of a web-based approach turned out to be a good choice we decided to follow the approach.

We introduced test automatisms that check the source code with unit tests each time a new version is built or a code review has been conducted. Additionally, monitoring services that observe the currently running version and check basic features automatically using the Selenium framework on a frequent basis were deployed.

2. Results

2.1. Issues of the prototype and feature requests

The users mainly reported one issue and one feature request: missing network connectivity and readiness for the conduction of clinical trials.

Although network connectivity is increasing within the hospital, a wireless network connection is only consistently available at the registration desk. Unfortunately, the web-based prototype was built to only request a single question from the server, present it to the patient, send the answer(s) back to the server and request the next question consecutively. Due to this implementation, a network connection is necessary during the patient's survey. Since the patient usually conducts

the survey in the waiting or patient room, network connectivity at the registration desk alone is not sufficient. Two obvious solutions might be to provide wireless network at the waiting/patient room or to use cellular data. The first one does not cover the scenarios of deployment in additional clinics or moving patients. The second one does not comply with the local and probably most data protection guidelines as long as the tablet computer displays patient data. Thus, the implementation has to be adapted.

The prototype was already able to export PRO data in an XML format, which is both human readable and machine processable. Since patient data that is available in such a format is well-structured, it can be both imported into an EHR and be used within clinical trials. However, the availability of structured data alone is not sufficient for utilizing a tool within clinical trials. Data formats acknowledged by the Food and Drug Administration (FDA), e.g. CDISC ODM [5], are necessary to enable researchers to submit their trial results. Additionally, regulatory requirements, such as documentation of the development, extensive software tests and code reviews, have to be met as well.

The technical analysis of the prototype led to two results: (1) the underlying framework is still vulnerable to common web-based attacks, such as Cross-Site-Scripting (XSS) or Denial-Of-Service (DOS), which in part allows the attackers to gain information of/from the application or even compromise the system; (2) new features, like additional types of questions, export to CDISC ODM files or full audit logs would consume a lot of programming effort. The utilized framework is designed for rapid software prototyping, which means that requirements are quickly implemented by following the "convention over configuration" pattern and creating modern graphical user interfaces automatically. This property supports the development of prototypes, but makes it cumbersome for developers to implement complex additional application logic. Additionally, the local knowledge about the framework was not as extensive as about other programming languages and frameworks.

2.2. Solutions in the new mobile questionnaire system

The resulting new version of the MoPat system covers all features of the prototype: It supports questions for which the patient can choose one to several answers (multiple choice), write a free text answer, pick a certain date, choose a value on a visual analogue scale (both horizontally and vertically) or Likert scale. Each question is presented separately on a single screen to guarantee high usability.

To address the connectivity issues, the questionnaire(s) is/are completely transmitted in the beginning of the survey, and network connectivity is only necessary at the end of the survey. Until then, each new answer given by the patient is sent to the server following the "fire and forget pattern", which means that the request is fired to the server but the server's response is ignored. The jQuery-based business logic at the client-side takes care of displaying the next question and storing the given answers as long as the server did not confirm the latest update in the background. This implementation has even been successfully tested within an investigator-initiated trial (IIT) at another site. The application server deployed at this site delivers the complete questionnaire via a VPN-secured cellular connection to the remote client; no lag could be identified while answering the questions.

A template-based export allows administrators to upload an empty file either in the XML structure already known or ODM. The application detects placeholders to put the patient's responses into and exports these PRO data following the given template.

The issue of the utilized programming framework is solved by the use of a well-known and better supported programming framework for Java, Spring [6]. Java is a commonly used programming language, hence the further development of the redeveloped application is ensured.

Since MoPat is developed as a web-based application, the user interface's look and feel is not like a native application. The new MoPat version uses the jQuery Mobile Framework to improve the look and feel and the input elements. This framework enhances the web site elements with touch-improved behavior. As you can see in figure 1 the graphical user interface is well structured with navigation at the top and providing one single question on every screen.

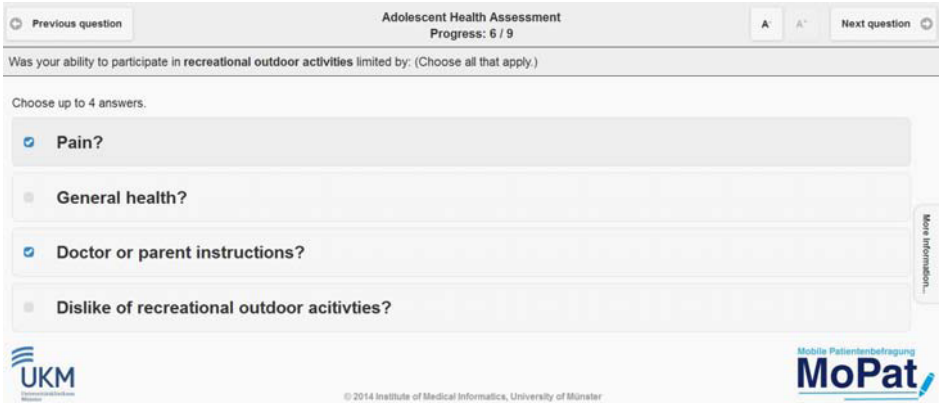

Figure 1. Screenshot of the new version of MoPat used by the patient.

Most clinics provide a single sign-on service to authenticate users within the incorporated software systems. The new MoPat version provides the ability to authenticate users within MoPat via a configurable LDAP (Lightweight Directory Access Protocol) implementation. This simplifies user management and authentication for the clinical users, because no additional passwords are needed.

The application contains a complete audit logging mechanism. It logs the user's and system's action each time a patient's data point is read, written, updated, deleted, transferred or received, together with the calling function and date of processing. The application is only permitted to add additional log entries into the separate database. In summary all identified features and issues are listed in Table 1 including their implementations in the different MoPat versions.

Table 1. Features/issues identified and addressed by the MoPat versions

Feature / issue	MoPat	MoPat redesigned
network connectivity	Required constantly	Required selectively
XML export	Supported	Supported
EDC-compliant export	Not Supported	Supported
XSS vulnerability	Strong	Weak
Adding new question types	Difficult	Easy
Audit logging	Not Supported	Supported
Touch capabilities	Weak	Strong
LDAP authentication	Not Supported	Supported

3. Discussion

The user's feedback has shown that the existing prototype already covered all functional and usability requirements. Nevertheless, additional infrastructural and security issues have been discovered and regulatory needs have been identified. Thus, the existing implementation was not sufficient for a routine, multicenter utilization (within clinical trials) that is both scalable and sustainable.

Although dealing with an infrastructural problem, we were able to solve the mainly reported issue with a design pattern within the program. This does not only enable the application to behave more robustly to connectivity problems, it also takes off the patient's impression of not using the application in the right way due to non-respondence.

In theory the old MoPat could support the documentation process within clinical trials, but the export abilities where not sufficient to transfer the stored data in MoPat to an electronic data capture (EDC) system for clinical trials. In addition a system used in clinical trials has to fulfill certain requirements in code quality, e.g. comprehensive documentation or testing of source code. The new source code is tested with unit tests and the running application, especially the client-side implementation, is automatically tested with Selenium. Thus, the chance to achieve a qualified software certification is higher for the new developed system. To handle the export of the stored data to an EDC system, the new implementation has an integrated ODM export mechanism. This XML data format can be imported by several EDC systems, e.g. OpenClinica, hence no additional transformation has to be performed to make PRO data gathered with MoPat available to EDC systems. By improving the software quality and data export capabilities, we achieved to provide a PRO system that can be utilized in clinical trials.

In conclusion, the new implementation addresses all identified issues both from organizational/infrastructural and development point of views. This leads to a mobile questionnaire system for PRO that is – due to its new capabilities and configurability - ready to be utilized in different scenarios. Both routine PRO data collections in connection with an EHR system and standalone implementations with (direct) exports to an EDC system are now possible and successfully tested.

References

[1] Dugas M, Breil B, Thiemann V, Lechtenbörger J, Vossen G. Single source information systems to connect patient care and clinical research. Stud Health Technol Inform. 2009;150:61–5.

[2] Rogausch A, Sigle J, Seibert A, Thüring S, Kochen MM, Himmel W. Feasibility and acceptance of electronic quality of life assessment in general practice: an implementation study. Health Qual Life Outcomes. 2009;7:51.

[3] Fritz F, Balhorn S, Riek M, Breil B, Dugas M. Qualitative and quantitative evaluation of EHR-integrated mobile patient questionnaires regarding usability and cost-efficiency. Int J Med Inf. 2012;81(5):303 – 313.

[4] Rubyonrails Ruby On Rails: CVE security vulnerabilities, versions and detailed reports [Internet]. [cited Feb. 3rd 2014]. Available at: http://www.cvedetails.com/product/22568/Rubyonrails-Ruby-On-Rails.html?vendor_id=12043

[5] Clinical Data Interchange Standards Consortium (CDISC). ODM Certification & Archiving and Interchange of Metadata [Internet]. 2013 [cited Nov. 12th. 2013]. Available at: http://www.cdisc.org/odm

[6] Walls C. Spring in action. 3rd ed. Shelter Island: Manning; 2011. 400 p.