Towards a Norwegian Implementation of Electronic Personal Health Records

Torstein Jensen[1], Knut Halvor Larsen[1], and Anders Kofod-Petersen[2]

[1] Department of Computer and Information Science,
Norwegian University of Science and Technology,
7491 Trondheim, Norway
{torstej|knuthalv}@stud.ntnu.no
[2] Norwegian Centre of Electronic Health Records,
7489 Trondheim, Norway
anderpe@idi.ntnu.no

Abstract. Today, most healthcare systems are built for the convenience of healthcare providers. Since the 1990s there has been a growing interest on involving the patient more in his or her own treatment, and in that way improve the quality of care. One of the means of doing that is to develop software that puts the patient in the centre while allowing the providers to carry on with their normal work habits. The electronic personal health record is one possible solution to this challenge. In this paper we present an implementation of the personal health record and describe how we adapted it to Norwegian standards.

1 Introduction

The concept "Personal Health Record" (PHR) has traditionally been used to encompass systems that focus on the patients, by making it easier for them to exercise their legal rights concerning their patient records. This right includes access to the content in the record and determining who should be able to read and write in it. Additionally, the PHR offers functions that enable patients to contribute to their records and ensure better communication with healthcare providers. These functions are meant to stimulate patients' awareness and reflection, and finally give a better quality of care. In Norway, there has recently been an increased focus on patient centred care. Several actors have come up with solutions that are either similar to, or a step towards a full-scale implementation of the PHR. The focus in this paper is on a solution from the USA called Indivo.

The rest of the paper is structured as follows: Some background information on ongoing work related to the PHR in Norway is given in Section 2; Section 3 describes the guardian angel manifest; This is followed by Section 4, which introduces an implementation of the PHR founded on the guardian angle manifesto called Indivo. In Section 5 some information about Norwegian healthcare informatics is given and we describe how Indivo is adapted to allow for the Norwegian standards. We present our conclusion and further work in Section 6.

2 Background

In Norway, some work has recently focused on allowing patients access to their health records. “MyRec” [1] is a web portal, which is joint venture between several Norwegian hospitals and health institutions. When connected, the patient can gain knowledge about his medical situation and share information with the different actors. The portal offers information tailored to special patient groups, forms tool, search engine, help and self-service administration, and means for communication between patient and healthcare actors. As of today, the portal offers no information from the Electronic Health Record (EHR), but there is ambition to include this in a controlled way.

“WebChoice” is a project connected to “MyRec”, which focuses on cancer patients [1,2]. It is a web-based support system that makes it easier for patients to report, prioritize and monitor their symptoms and problems. In addition, it offers resources with relevant information for the patient both with regards to his or her disease and with regards to the patient and his or her family. There is also a discussion forum where cancer patients can share knowledge and experiences, and receive support from health personnel. Like “MyRec”, “WebChoice” offers no information from the EHR.

In addition, the “Kjernejournal” project [3], which focuses on sharing of parts of the EHR between several health actors, is worth mentioning. In the early stages the goal of the project is to ensure better handling of drugs when health actors cooperate on treating a patient, in order to reduce the amount of adverse effects. The Kjernejournal is created by copying the relevant part of the EHR to a separate server with high availability and allow other actors to contribute and read data in an automated fashion. In this way the data is kept consistent and readily available. The patient in question decides on the access policies together with his or her primary physician. The Kjernejournal covers some aspects of the PHR in that it supports a collection of medical information for the patient. However the data is not meant for the patient to read; it is not prepared in a readily accessible way and it is not the type of data that engages the patient in his or her treatment. Also, there is no room for patient provided information.

Currently, the Norwegian Research Centre of Electronic Health Records is investigating the applicability of a PHR as a supplement to the Kjernejournal [4]. They propose an architecture where the Kjernejournal is kept intact, and a separate PHR is connected to it. The information exchange happens in a read-only manner. This allows for the aforementioned exchange of information between the different healthcare actors, however the data can also be sent from the Kjernejournal to the PHR where it can be read by the patient. The patient can also provide information like “Patient stories” and experience reports to the PHR, which then can be read by the appropriate healthcare actors.

3 Guardian Angel

The guardian angel manifest [5] deals with two classes of problems related to patient treatment. One is that the patient gets alienated from participating in decision-making concerning his or her own healthcare. The problem here is that the patient has a lack of understanding and a lack of trust in his or her providers due to two reasons: Fist of all, the medical practice has over the years become more and more capable and complex, hence it is more difficult to explain medical related information to the patient in a way that makes it understandable for him or her. Also the pace of medical care has increased considerably, which causes less time for the provider to inform the patient. The second problem lays in the fact that there is little integration between healthcare providers. This makes it hard to share patient data, for instance when the patient moves and thus changes hospital and providers.

To improve the situation it is proposed to shift the focus away from information systems built for the convenience of the healthcare providers, and instead build systems that focus on the patients. Such a new system, called “Guardian Angel”, is meant to integrate the patient’s health related information over a lifetime. Hence, it is at a minimum supposed to contain a complete medical record of the patient, something, which is very difficult to reconstruct as the patient moves through life. Other functionalities include being able to collect patient data, check, interpret and explain medically relevant plans and facts, adapt advice based on the patients previous experiences and preferences, and monitor progress and help educate, encourage and inform the patient.

The believed advantages are: improved medical decision making, a reduction in healthcare errors, allowing persons to make better personal decisions, and improvements in effectiveness and efficiency of healthcare. Also, the healthcare providers will have access to accurate and up-to-date data, a better opportunity to discover changes in the patient’s health, and be able to communicate better with the patients.

When building a guardian angle, flexibility is an important keyword. The software has to be easy to adapt to new standards that will evolve in the future. Also, over time the understanding of requirements might change, or new functionality has to be added. Hence it is important to develop an open architecture consisting of components provided by generic and preferably standards-based facilities.

4 Indivo

Indivo, which is a project at Harvard Medical School, MIT, and Children’s Hospital Boston, is based on the guardian angel manifest [6]. It is basically designed as a personal health record where the patients have control over a complete secure copy of their own medical record. Hence, it is not meant to be the primary record of the healthcare system but rather a collection of medical data across the patient’s history. Depending on the server settings, the patient is allowed to read, write and modify components, and he or she decides who shall have access to the different parts of the PHR. Granting rights to already registered persons, groups or roles does this. The access control can be done on a fine-grained level, e.g. for each document in the patients medical record. Indivo is open source and free, and built to public standards. It is also module based and easy to configure to different needs.

One of the main ideas with the PHR is to enable better communication between the primary physician and patient. Indivo supports this by including support for messages. It also supports lots of other document types and new ones can easily be added due to Indivo's flexible structure. New information can be sent to the patient from any client (e.g. an EHR system) as long as the information is sent using Indivo's communication protocol.

Indivo can be divided into three layers: client, server and store. Each layer can be located at different physical locations, and both the client and the store are designed in a pluggable fashion such that a number of different types can be used. As most of the data processing happens in the server, this can be regarded as "heart and soul" of the architecture. Indivo implements a model-view-controller architectural pattern.

Indivo uses a communication protocol called IndivoTalk to communicate between the client and the server. This protocol is based on XML messages and request-response interaction. Adding new XML schemas and performing small modifications to the server source code can quite easily extend it.

The following subsections describe each layer of the Indivo architecture in more detail. A complementary illustration is given in Figure 1.

Client The client is the layer used to communicate with the Indivo server. As long as it uses the IndivoTalk protocol the client can be any software process regardless of the platform and implementation language.

The Indivo actor is a registered user of the system. Each user has his own record and some attributes, e.g. roles and group memberships. Each role has some privileges. For an example, a user logging in with a patient role might be allowed to update, read and add documents to his/her own record, while a user logging in with a provider role must be given privileges by the patient in order to do the same to the patient's record.

Server The server is a Java 2 Enterprise Edition (J2EE) compliant servlet. Using the servlet technology has several advantages. The Secure Hyper Text Transfer Protocol (HTTPS) can be used for encryption of IndivoTalk messages between the client and the server. Also the manipulation of requests is simplified by using the Java Servlet Application Programming Interface (API).

The server consists of the three different main parts: The communication layer, the action response layer and the authorization module.

The communication layer is responsible for accepting XML IndivoTalk messages and converting them into program objects, which are then sent on to the action response layer. When a response arrives from the action response layer it

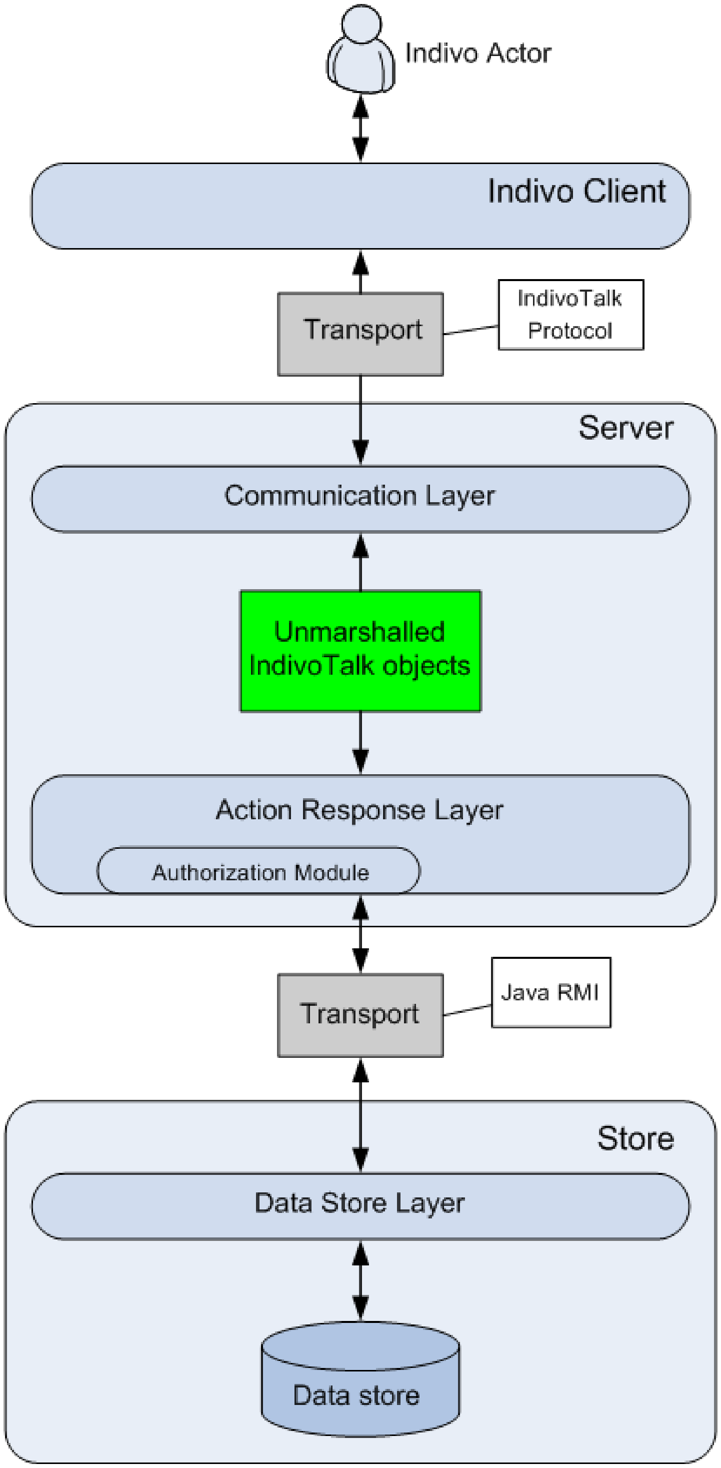

Fig. 1. Indivo architecture (adapted from [6])

is converted back into XML messages and sent to the client. This processing is done using Java Architecture for XML Binding (JAXB). The messages are automatically marshalled and unmarshalled according to a specified schema, which in this case consists of the different IndivoTalk elements.

The action response layer processes the actions received from the communication layer and maintains information about the different sessions. There is one action responder for each type of action, and each one of these has its own processing and authorization procedure. The layer reads several XML configuration files in order to create the different responders, and these files contain among other things information about the data store and authorization engines. It is the responsibility of the action response layer to delegate authorization to the authorization module before the request may be executed.

The authorization module is an implementation of the Extensible Access Control Markup Language (XACML). Depending on the action responder (and type of action) there are either one or two authorization steps. The first one decides whether the user is allowed to perform the type of action requested, and the second finds out whether there are any record policies that prohibit the user from performing the action. In the first case the authorization policies are read from a configuration file, while the record based policies are read from the data store. The advantage of using XACML is that it is very flexible.

Store The data store contains the different records. Each record consists of several documents and each document consists of several versions. The different document types are defined with XML schema at the server, and adding new document types are easy.

Technically speaking the default store is a Berkeley Database consisting of several encrypted XML files. They are accessible to the action response layer using Java Remote Method Invocation (RMI). Encryption can occur before the data is sent from the server or at the store. One advantage of encrypting the data at the server is that the encryption key is kept separate from the encrypted data.

Other store types can replace the default store, as long as they adhere to the Indivo API, which consists of method calls that the server uses when it communicates with the store.

5 Adapting Indivo to the Norwegian Structure

Healthcare informatics in Norway have the last few years been driven by a series of government strategic plans where the latest, Te@mwork 2007, is scheduled to end in 2007. The plans have led to large initiatives in the IT- and healthcare sector. The development of “Norsk Helsenett”, a closed network for electronic communication and cooperation in the healthcare sector, is one of the outcomes from the initiative. Another important area for the strategic plans has been standardization work. For this purpose the government formed Norwegian Centre for Informatics in Health and Social Care (KITH) in 1990. KITH contributes

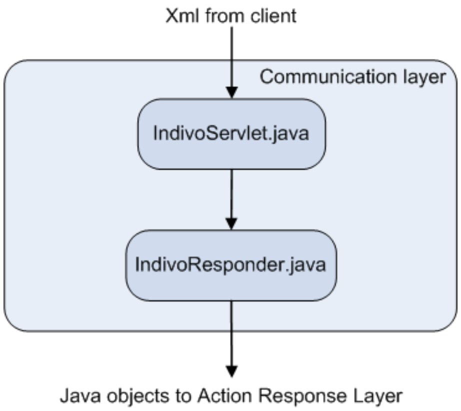

Fig. 2. Original module

with coordination of the IT development in the sector. A large part of this work is development and maintenance of codes and standards. KITH also takes part in international standardization. Contemporary with the development of electronic solutions in the healthcare sector an evolving interest and concern in patient security also increases. The Data Inspectorate does evaluation of systems fulfilment for patient security and Norwegian law. A possible implementation of a personal health record could very well be problematic for the Data Inspectorate, but this is outside the scope of this article.

Implementing KITH standards within Indivo is an important step towards adapting Indivo to the Norwegian structure. This implies the employment of the content and EHR standards defined by KITH. An important part of these standards is the ability to send and receive standardized information to and from other healthcare systems, a prerequisite for a personal health record system. A key element in this information sharing is the “Hodemelding” [7] , which is based on XML. The Hodemelding is supposed to create a common message header with basic information about sender and receiver that wraps the health information documents. This therefore leads to a common interface for communication easy to implement.

Thus making Indivo able to receive a Hodemelding is an important step towards a Norwegian implementation of a PHR. Our intention was to implement support for Hodemelding without making too many modifications to the original code. Indivo uses XML for communication between the server and the client, this is done by altering the module that handles requests and responses. An illustration of the original module is given in Figure 2 and the altered module in Figure 3.

By adding a new Hodemelding JAXB context to the original IndivoTalk context the system is able to receive and unmarshall incoming XML that is based on the Hodemelding XML-scheme. Next the system does a simple instance check of the object that has been unmarshalled to find out whether it is a request

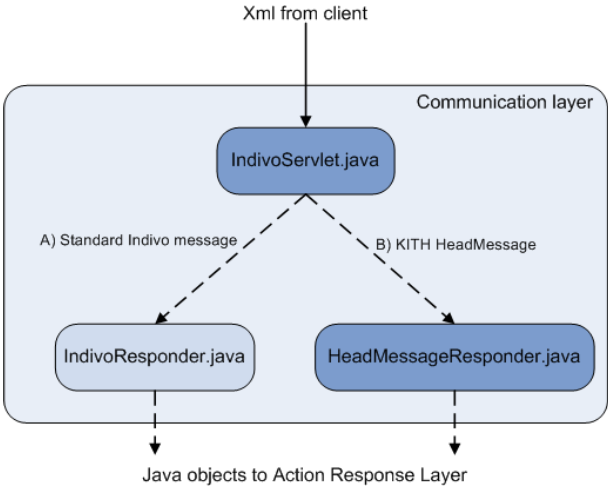

Fig. 3. Altered module

from the IndivoTalk protocol or a Hodemelding. To handle the specific content of a Hodemelding and the different documents it can contain we created a new module. Functionality in this module is only used to interpret the Hodemelding, so for standard tasks like authentication and adding documents the module still utilize the native Indivo functions.

Besides handling Hodemelding the Indivo server should also be able to handle specific documents as defined by KITH. The various document content standards are defined in XML-schemas from KITH. Because Indivo also utilize XML-schemas to define its content documents, only small modifications have to be done to add new document types to the server. As an example of this we added the discharge summary [8] messages to Indivo. In addition to this we added a document called application receipt [9] used for confirmation of received messages with or without errors. This receipt is sent like any other document inside a Hodemelding, but only if the original sender requests one. In our implementation of the new module used to handle the Hodemelding the receipt is automatically generated and sent if needed.

In this we have not only added functionality to receive document, but also made changes to the kind of documents that are available in the PHR. With these rather small changes we have made the Indivo system one step closer to use in a Norwegian setting.

6 Conclusion and further work

Indivo is a promising open source program that aims at creating a web-based personal health record. The fact that Indivo uses XML both in communication and configuration makes it a highly adaptable and modifiable system. At the time of writing the current version, 3.0 beta, has a PHP based user interface that will help the patient control his record. Although there currently is some difficulties installing and running Indivo, we look forward to promising future releases.

In our work with Indivo we have managed to prove its suitability as a personal health record system within a Norwegian structure. With some changes to configuration XML files and rather small changes to the implementation itself, we have made Indivo able to receive a Hodemelding, handle its content, and if needed send an application receipt in a new Hodemelding back to the sender. As a practical example we have used the Norwegian standard for a discharge summary as content for the Hodemelding and the PHR. This ability to use the Hodemelding for communication is paramount for any healthcare system used in Norway. In addition, the fact that Indivo utilize a model-view-controller architectural pattern makes it easier to add or change any logic layer.

There is still much work to be done before Indivo fully can be deployed in Norwegian healthcare. Disregarding the problems concerning legislation there are still technical challenges. Researchers at NSEP and students at the NTNU are currently involved in different aspects of the personal health record and Indivo. This consists of adopting a Norwegian document structure based on case, document and fragment, into Indivo, handling and presentation of prescriptions, and research into how to enable patients to select roles for access.

7 Acknowledgements

The authors would like to thank Hallgeir Stueness for his assistance.

References

1. Jor, S.: ”MyRec” (MinJournal) – the patient dialogue. In Kofod-Petersen, A., Brasethvik, T., eds.: Proceedings of the International Symposium on Personal Electronic Health Records (ISePHR 2006). Number 06/2006, Trondheim, Norway, Department of Computer and Information Science, NTNU (2006) 27–30
2. Senter for pasientmedvirkning (SPS): Webchoice - internett-tjeneste for kreftpasienter. (http://avd.rikshospitalet.no/syf/forskning/prosjekter/webchoice05.htm) Last visited: 30/10-2006.
3. Nystadnes, T.: Løsningsskisse fyrtårn Trondheim: Legemiddelopplysninger i Samtykkebasert kjernejournal. Technical Report 29/05, Norwegian Centre for Informatics in Health and Social Care (2005)
4. Brasethvik, T., Kofod-Petersen, A.: Eigenjournal: A personal collaborative medical journal. In: Proceedings of the 1st International Workshop on Health Pervasive Systems, Lyon, France, IEEE Computer Society Press (2006) 52–56
5. Szolovits, P., Doyle, J., Long, W.J., Kohane, I., Pauker, S.G.: Guardian angel: Patient-centered health information systems. Technical Report TR-604, Massachusetts Institute of Technology, Laboratory for Computer Science (1994)
6. Simons, W.W., Mandl, K.D., Kohane, I.S.: The PING Personally Controlled Electronic Medical Record System: Technical Architecture. J Am Med Inform Assoc **12** (2005) 47–54

7. KITH: Standard for hodemelding - informasjonsmodell og xml meldingsbeskrivelse. (http://www.kith.no/upload/3037/R01-06-Hodemelding-v12.pdf) Last visited: 27/10-2006.
8. KITH: Elektronisk utveksling av epikrise. (http://www.kith.no/upload/1075/R26-02EpikriseMelding-v1.pdf) Last visited: 27/10-2006.
9. KITH: Applikasjonskvittering - informasjonsmodell og meldingsbeskrivelse. (http://www.kith.no/upload/1187/R15-04AppRec-v1.pdf) Last visited: 27/10-2006.