International Journal of Latest Trends in Engineering and Technology
Special Issue SACAIM 2016, pp. 106-114
e-ISSN:2278-621X

WELLNESS DATA SHARING FROM PERSONAL HEALTH RECORD (PHR) TO ELECTRONIC HEALTH RECORD (EHR)

Sarita Pais[1] and Xiaoci Xu[2]

Abstract- Patients have the option to maintain their health data and medical information in a Personal Health Record (PHR) system. However health professionals use different systems such as Electronic Health Record (EHR) to manage their patients' medical records. PHR systems provide greater benefits when integrated with EHR systems. The data and its format are different in the two systems. Hence this research project looks for solutions for data exchange between PHR and EHR systems. This research project followed design science research methodology to build an ecosystem and evaluate it through experiments using hypothetical data. Existing open source PHR and EHR systems were evaluated and literature related to PHR and EHR were reviewed in this project. HL7 is an international standard which is used for transferring health data between medical systems. This project set up the ecosystem using Microsoft HealthVault and OpenMRS. The ecosystem readthe data file containing blood glucose readings and converted the data into a format which Microsoft HealthVault can accept. Microsoft HealthVault exports the data in CCD/CCR data format. The ecosystem converted CCD/CCR data to CDA data format in OpenMRS. The ecosystem allowed the patient to share self-managed health data with health professionals and provided them with the patient's background health data enabling better consultation with patients. The findings of this project is a proof of concept which can demonstrate that patient managed health data from PHR system can be integrated into EHR system.

Keywords –PHR, EHR, blood glucose readings, OpenMRS, Microsoft HealthVault

I. INTRODUCTION

Patients can manage their health data through portals called Personal Health Record (PHR). Such patient portals are promoted by government and health providers around the world. Patients are considered active participants in their wellbeing. PHRs provide concise information about patient healthenabling them to make informed decisions and take advice from health professionals. Patients are in possession of their own health records which are electronically saved and can be retrieved from anywhere. Patients are willing to take charge of their own health records and realisedthe perceived benefits from these systems. A study conducted in New Zealand [1]on a patient portal indicated that patients were happy with the improved communication with their health provider and access to their health data.

Most tethered PHRs receive health data from Electronic Health Record (EHR) but do not allow patients to share their self-managed health data with EHR. Patients with chronic diseases likediabetes maintain daily glucose readings taken at different times during the day such as two hours before breakfast (fasting) and two hours after a meal. These readings are usually stored in a glucose meter. A software application is often provided with the glucose meter to download readings to a computer or smart phone. However the data is not integrated into a medical system such as EHR.

[1] *Faculty of Business and Information Technology Whitireia Polytechnic, Auckland Campus,New Zealand*
[2] *Faculty of Business and Information Technology Whitireia Polytechnic, Auckland Campus,New Zealand*

The United States, United Kingdom and Australia have promoted PHR Systems. In New Zealand the concept is still new and several medical centres are promoting a PHR system called ManageMyHealth to allow their patients to view their health records, book appointments with the medical centre and renew prescriptions. However the functionality of the portalis limited. For example, patients cannot upload their daily health records through this portal.

Genitsaridi et al. [2] analyzed the basic requirements for a customizable and extendable PHR system as depicted in Figure 1. One of the basic requirements in a PHR system is self-health monitoring. Patients should be able to upload necessary health and wellness data about themselves such blood glucose readings, blood pressure readings from devices used at home into PHR system.

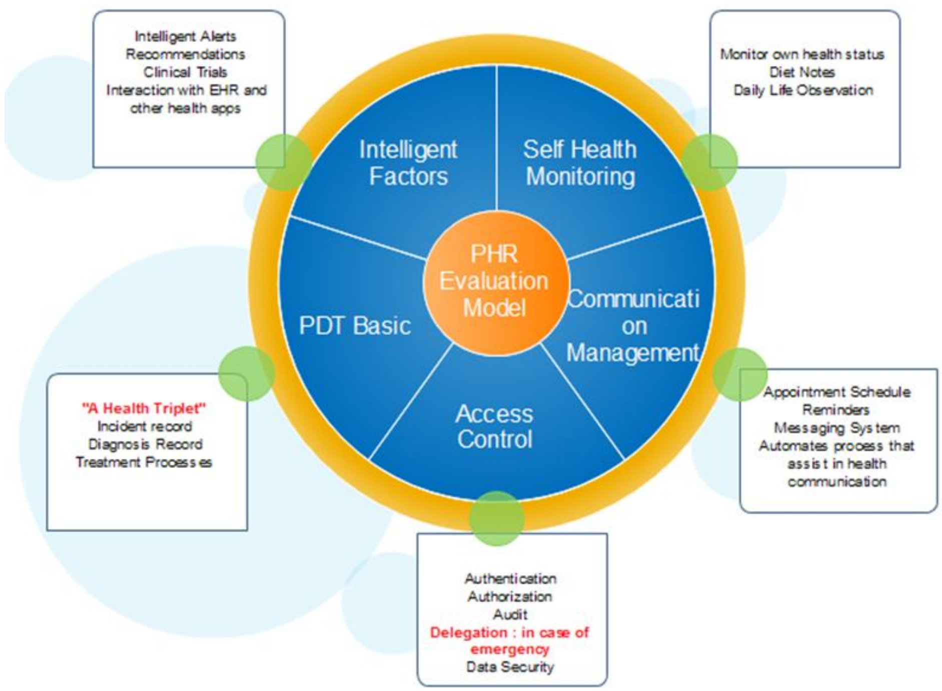

Figure 1: PHR System Evaluation Model (Adapted from Genitsaridi et al., (2013))

The current research study was undertaken to review various open source PHR and EHR systems. Design Science Research Methodology (DSRM) [3] was applied to build a proof of concept artefact to send patient managed data to an open source EHR system as outlined in Table 1.

Table -1 Artefact based on DSRM

Hevner et al. (2004) [3]	**Current Research Study**
Design as an artefact	Use existing open source software and extend functionality to include additional data.
Problem relevance	To share patient managed health data with clinicians and store in an EHR
Design evaluation	Blood glucose readings maintained in glucose meter and PHR used as input to the artefact
Research	Proof of concept of storing patient managed data in an EHR system

contributions	
Research rigor	Existing literature around PHR and EHR systems studied to include new functionality of adding patient data into EHR system. Use data from PHR system and glucose meter as input to new system.
Design as search process	Review functionalities available in various open source EHR projects.
Communication of research	Through journal articles presented to business and technology oriented audiences.

II.PROPOSED PROOF OF CONCEPT ARTEFACT

A. Selection of PHR system –

PHR systems under the Free and Open Source Software (FOSS) were reviewed. Microsoft HealthVault [4], Indivo X [5] and iTrust [6] were identified as suitable to the project. Most of these software systems did not come with good documentation and was not easy to install. At the time of the research project Indivo X stopped hosting a development sandbox. iTrust was abandoned for similar issues during installation.

MicrosoftHealthVault had SDKs for different platforms such as .net, Java and Android. It is a cloud based PHR platform which helps patients to collect, store and share health information with family members and healthcare professionals. It provides a feature-rich development platform to develop health related applications and allows connectivity with different types of health devices and third-party applications to help patients manage their health data. The health data can be used by different authenticated organizations such as hospitals, healthcare associations, laboratories. It supports computers and mobile terminals with different operating systems. It also supports 25 apps and 188 devices including blood pressure monitors, pedometers, glucose meters and more. Microsoft HealthVault served as the PHR system of choice in the research project.

B. Selection of EHR system

EHR systems such as TolveneCHR [7], OpenEHR [8], OpenEMR[9] and OpenMRS [10] were evaluated to identify a suitable EHR for the proposed ecosystem.The Tolvenplatform contained three modules:TolveneCHR, TolvenePHR and Tolven Analytics where eCHRis the EHR module; ePHR is the PHR module. The TolvenePHR is a personal health record system which helps patients to record, manage and share their health information. The Health Analytics solution assists the users to extract or analyze data for statistical purposes.

Helms and Williams [11] compared the privacy and security of the four EHR systems namely Tolven, iTrust, OpenMRS, OpenEMR based on access to patient data and patient data privacy. All four systems had similar evaluation results.

The EHR system used in this research project was OpenMRS which is an open source project predominantly used in many developing countries. The technical architecture had to be studied before adding new features to accept patient generated wellness data. OpenMRS system was chosen to extend the system functionality to include patient managed data like glucose readings. The OpenMRSarchitecture is structured into three layers: database, service and presentation. The database layer comprises of MySQL database. The service layer is based on Java language and follows Spring framework. The presentation layer uses HTML, JSP and jQuery.

The conceptual data model is explained in Figure 2: A patient visits a medical centre. Each visit is recorded with a visit_Id. A visit also has a visit type, start time and end time. Within the visit different encounters can be created. Each encounter has a unique encounter_Id, encounter type, time, location and provider. Within an encounter, different observations can be created according to the requirements.

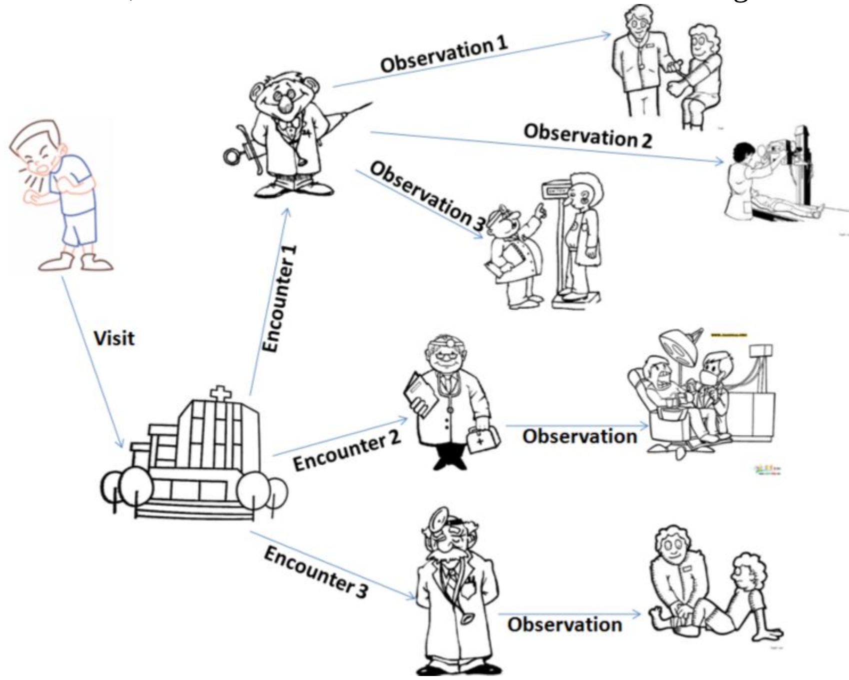

Figure2: OpenMRS Conceptual Data Model

There is no separate provision available for patient managed data to be captured in the openMRS. Data from glucose meter and PHR system like Microsoft HealthVault was recorded as observation under openMRS. Although it worked to write data into the system, the process was cumbersome. The system had to create a chain of events and identified the particular health care professional related to the particular observation.

Figure 3describes the architecture of the ecosystem. In the ecosystem, devices and services are separated into three categories: patient data collection components, PHR and EHR components. The patient data collection component includes all the devices used for collecting patient data; forexample,blood glucose meter. The PHR components include a personal computer (tablet, mobile phone or other mobile devices), an application server, a Microsoft authentication server and a Microsoft storage server. The EHR components include an application server, OpenMRS server and a computer or other devices used by health care professionals. The application server runs as a middleware shared by both PHR and EHR systems.

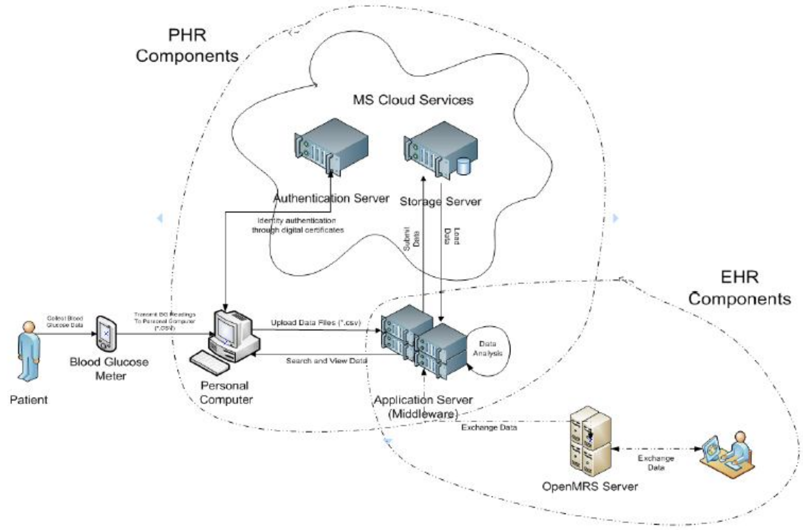

Figure 3: Ecosystem Architecture

Since Microsoft HealthVault and OpenMRS use different data schemas, a data mapping from source data to target data schema was required to match the data between these two systems. Figure 4 shows data mapping between Microsoft HealthVault and OpenMRS. For example, blood glucose reading from Microsoft HealthVault is stored under an observation in OpenMRS for the matching patient ID.

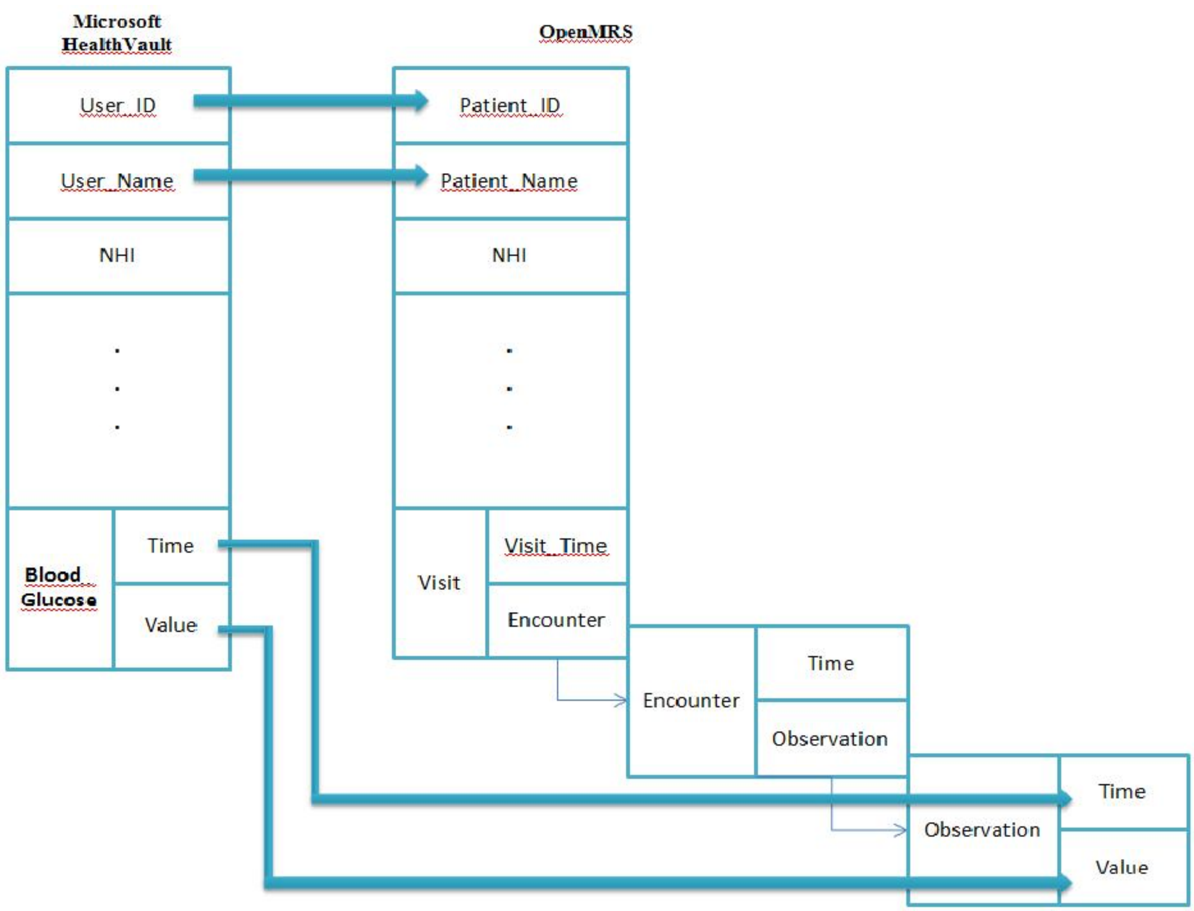

Figure4: Data mapping from Microsoft HealthVault to OpenMRS

A middleware software which maps the data schema from PHR system to EHR system needs to be written [12]. The schema mapping techniques from database literature were applied. In the absence of a middleware, data from PHR was manually mapped to the EHR system.The mapping included the schema and its formats. For example, conversion of CCR/CCD to CDA. The middleware software components reside in the application server.

The middleware in this structure is essential. Several services are provided by the application server. The application server analyzes the CSV data file and extracts useful information from it such as date, time and value. Since CCR/CCD data format follows the HL7 standard it can be converted to CDA data format. The application server can send XML file which is in CCD/CCR data format and created by Microsoft HealthVault directly to the OpenMRS server. Another essential function of the application server is to map the data between the PHR and EHR components. Each patient is given aunique identity in the system. The business identity (NHI) was also maintained.

OpenMRS has the ability to generate and exchange patient's clinical summaries using the Clinical Document Architecture (CDA) model which is an xml-based HL7 version 3 standard for clinical documents.

III. Experiment and Result

The working of the ecosystem is demonstrated through the sequence diagram in Figure 5below.The sequence diagram shows the order of interactions between different components of the ecosystem. These interactions as shown as input and output messages as explained in Table 2. The messages are numbered to describe the steps for a simple demonstration.

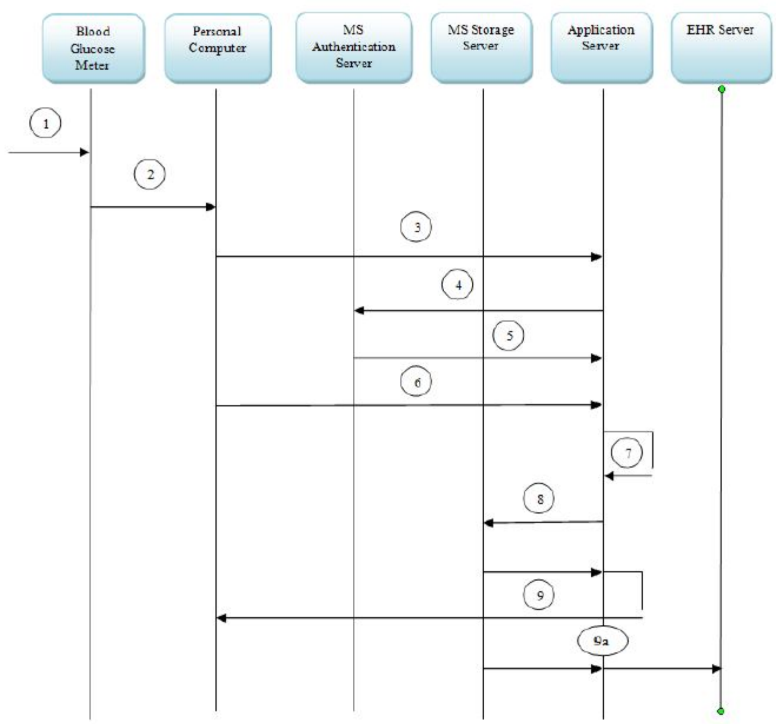

Figure 5: Demonstration of the Ecosystem

Each step from Figure 5 is explained in Table2 below.

Step #	Message	Description
1	Collect patient blood glucose readings	Patient obtains blood glucose readings using a blood glucose meter at home.

2	Download blood glucose readings to PC	Patient connects the blood glucose meter to a personal computer (mobile phone). Most glucose meters come with supporting software to download data. In most cases blood glucose meter readings are saved as a CSV file containing three main data pieces: date, time, blood glucose reading.
3	Patient login to PHR portal	Patient logs in to the PHR system.
4	Authentication	If the patient has not been authenticated the application server redirects the browser to Microsoft authentication server for user identity authentication.
5	Returning to application server	The browser redirects to application server after authentication.
6	Uploading data	Patient uploads the CSV file to application server.
7	Analyzing data	Application server analyses the CSV file.
8	Storing data	Application server submits data to MS storage server.
9	Displaying data	Application server loads data from MS storage server and display the data in the browser of the personal computer (mobile phone).
9a	Transmitting data	Application server loads data from MS storage server and then writes to the OpenMRS server.

In the ecosystem, the following functions were developed:

Send data files to application server: A jsp program was written to send data files to the application server. Users click "Select File" button to select a CSV file which was generated by blood glucose meter. On clicking "Upload" button, the CSV file is sent to application server.

Analyzing the CSV file: To analyze CSV file, a class named CSVAnalyserwas created. In this class, a file name "readFile" was sent as a String parameter to a method and a list of BloodXDatawas returned. Each line of this list contained a date, time and value to indicate a record of blood glucose reading data.

Upload data to the Microsoft HealthVault storage server: A servlet named "UploadServlet" was used to control the operation of the CSV file. An example of CSV data file and a class named "BloodGlucosePage" was used to analyse the blood glucose values. An example of CCD/CCR data format exported from Microsoft HealthVault is demonstrated in Figure 6.

Figure 6: An example of CCD/CCR data format exported from Microsoft HealthVault

Write data to OpenMRS database: A class named "CreateVisitFromFiles" was used to write data to OpenMRS database. In this class, methods of "createVisit", "createEncounter" and "createObservaton" are called sequentially. The value of blood glucose is written into OpenMRS database.

IV.CONCLUSION

A good PHR system not only help patients manage their own health data but also provides practical information to health care providers. Microsoft HealthVault was the selected PHR in this project. It can transfer patient data in CCD/CCR data format. EHR systems are essential for health care professionals to manage their patientdata. OpenMRS is an open source EHR system which provides the developers Java APIs. It can import patient data in CDA data format. This research study created an ecosystem for sharing data from Microsoft HealthVault toOpenMRS. Microsoft HealthVault exports data in CCD/CCR format and OpenMRS imports data in CDA format. CCD/CCR and CDA formats implement the HL7 standard. HL7 was used for data sharing between PHR and EHR systems.

Through this research study it was possible to build an ecosystem involving PHR and EHR systems. Patients are empowered to use the PHR system and send their health and wellness data to the EHR system. Thus the research proved that although there were different systems with different data formats, the data was able to be sent from one system to another using HL7 standard. Open source software are particularly useful to build these type of systems with minimal effort.

GLOSSARY

API	Application Program Interface
CDA	Clinical Document Architecture
CCD	Continuity Care Document
CCR	Continuity Care Record
EHR	Electronic Health Record
HL7	Health Level-7
HTML	HyperText Markup Language
JSP	JavaServer Pages
NHI	National Health Index

PHR Personal Health Record
SDK Software Development Kit

REFERENCES

[1] Y. Gu, K. Day, "Propensity of people with long-term conditions to use personal health records", Studies in Health Technology and Informatics, 188, pp. 46–51, 2013.

[2] I. Genitsaridi, H. Kondylakis, L. Koumakis, K. Marias, M. Tsiknakis, "Towards Intelligent Personal Health Record Systems: Review, Criteria and Extensions", Procedia Computer Science, *21*, pp. 327-334, 2013.

[3] A. R. Hevner, "Design Sciences in information systems research", MIS Quarterly, pp.1-49, 2004.

[4] Microsoft HealthVault, Developer Center, Accessed from https://msdn.microsoft.com/healthvault.

[5] IndivoX, Accessed from http://indivohealth.org/research

[6] iTrust, Accessed from http://agile.csc.ncsu.edu/iTrust/wiki/doku.php?id=requirements

[7] Tolven, Accessed from http://www.tolven.org

[8] OpenEHR, Accessed from http://www.openehr.org/

[9] OpenEMR, Accessed from, http://www.open-emr.org/wiki/index.php/Zend

[10] OpenMRS, Accessed from, http://openmrs.org/

[11] Helms, E., & Williams, L. "Evaluating access control of open source electronic health record systems", In Proceedings of the 3rd Workshop on Software Engineering in Health Care, pp. 63-70, 2011.

[12] P. A. Bernstein, J. Madhavan, E. Rahm, "Generic schema matching, ten years later", Proceedings of the VLDB Endowment, 4(11), pp. 695-701, 2011.