Towards Improving the Portability of Personal Healthcare Data

セディック，アデル，モハメド，アブドゥルラーマン，アルアッバシ

https://doi.org/10.15017/1543995

出版情報：九州大学, 2015, 博士（工学）, 課程博士
バージョン：

Towards Improving the
Portability of Personal Healthcare Data

Seddiq Adel Alabbasi

August 2015

Department of Advanced Information Technology

Graduate School of Information Science and Electrical Engineering

Kyushu University

Abstract

A Personal Health Record (PHR) is digitized and stored for efficient patient management in hospitals or clinics. At the patient side, the health records are not digitally stored.

Nowadays people move from one place to another more than before. However, the digitized past health records do not move with them. The digitized data belong to the hospitals and the patients do not have digital access to them. Past efforts produced standard medical record formats and information messaging protocols. These standards were focused to enable interoperability among organizations. The reality is that the standards are not widely deployed and many hospitals are using their own stand-alone systems. Therefore, it is technically difficult for an individual to digitally carry standard PHRs with him/her. Healthcare data portability for an individual remained a big issue.

This research aims to improve PHR portability for an individual. We introduce a new application; we call it a PHB (Personal Health Book). This application will allow an individual to import PHRs from different hospitals, store them and share with other parties. The PHB can sit on an individual storing device (e.g. USB drive or a smart card) or can be installed as a web application.

In this work, we focused on three technical challenges: (1) How to import PHRs from multiple heterogeneous sources efficiently (2) How to store the imported PHRs in an efficient manner in terms of capacity and accessibility, and (3) How to export viewable PHRs to other trusted parties.

In this research, we proposed three items to address the above three technical challenges: (1) developed an import algorithm to read PHRs from heterogeneous structured hospital healthcare applications without loosing or missing any data (2) designed a decomposed database to efficiently store the imported data consuming minimum capacity, and (3) developed an algorithm to export the data to trusted parties.

In order to prove our proposed concept and monitor the performance, we developed a simulation to find the most optimum database design for designing the PHB database. We compared long database design and decomposed design in terms of number of rows generated, number of unnecessary data repetition, database size growth, number of generated cells, and size of wasted memory. The simulation constitutes 10,000 records, in different 3 cases (12, 2 and 5 number of hospitals), each case has different maximum number of unique items (up to 10, 0 and 5).

On an average the decomposed database design performed better than the long database design with 16.7% in number of rows generated, 88.9% in number of unnecessary data repetition, 40.29% in database size growth, 42.84% in number of generated cells and 79.87% in size of wasted memory.

The second part of the simulation was used to evaluate the designed database behavior in terms of data items storing and size. On an average (of first 10 records) the stored items were 106.28% less than the imported items from various PHRs, that is 984 bytes stored compared to 3,216 bytes imported.

Finally, we tested the above mentioned algorithms and the database design of our proposed PHB application by importing PHRs from two different hospitals - one from our Portable Health Clinic (PHC) Project Database in Bangladesh and one from a Japanese hospital. The database structures of the hospitals are unknown to our PHB application. The imported PHRs were transferred to XML format manually. We set a schema with minimal restrictions to validate the XML documents. With the help of the proposed import algorithm, the imported

PHRs were successfully stored into the PHB database. Then we exported the stored data and viewed them by using a web browser. We could read all the PHR data. However, the Japanese PHRs were manually translated and human bias was implied. The assumption was that the dictionary performed accurately. This can be treated as an ideal scenario.

The simulation environment we developed this time considered limited cases. This may not reflect the real time scenario. Healthcare data is very personal and thus sensitive. It is hard to test the portability of healthcare data by using real life data. The algorithms, and the database design need to be deployed for accurate evaluation. We have the plan to test the PHB in our current PHC Project running in Bangladesh.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Interconnecting data networks and commercial interest in consumer behavior and purchasing trends characterize personal data landscape [1]. Hyper connected services (social networks, telecommunications, corporations, hospitals) are updated by billions of connected devices, people and sensors that record trillions of transactions and behavior each day [2].

Many organizations use data analytics to gain monetary value out of volunteered personal data. Commercializing personal data has led to heated debate over data ownership [3] and protection, security, and context of freely given information. There is less discussion on understanding what is being uploaded to the personal data landscape and its exponential growth as personal data is replicated over as it passes from one data network to another. One approach is to simplify the information type to persistent data and transitory data and state how it is created.

Personal data retrieval concerns access and use of persistent records that adds value to the individual's welfare. Personal data retrieval is achieved by setting up a personal data locker between data stores and requesting parties. Personal data lockers are part of the Big Data dialogue [4] [5] and in the concept, organizations hand back data to a personal data store.

To access the personal data store, an individual uses a Personal Identity Module (PIM). The PIM queries and fetches relevant personal data from the data bank and stores it offline in a device. A requesting third party could access the data bank after an agreed authentication process. Data lockers enable data portability because a person can share their details with service/person at anytime.

Personal data smart management associates a persons' transitory data to a specific location and time and saves the transaction. Mobile devices are used for data transactions, such as for uploading medical records to online health databases [6] [7]. Smart cards are used by offline applications to authenticate an individual [8].

Personal Health Records (PHR) are a set of computer-based tools that allow people to access and coordinate their lifelong health information and make appropriate parts of it available to those who need it [9] [10]. PHR systems were developed in the late 1990s to target patients who were travelling and needed healthcare and for situations where patients were not able to provide their health information [11] [12], or have communication difficulties.

Currently, three types of PHRs have become available [13]. These include: Stand-Alone formats (PC, USB drive), where consumers store health information on personal computers but lack the ability to exchange information between consumers and healthcare provides; Web formats that are managed by third parties (such as Microsoft Healthvault [14], Dossia Consortium [15]) and allow consumers to maintain their health information on private online accounts accessed by a login ID and password; and Integrated PHRs with Electronic Health Records (EHR) where a healthcare provider (such as MyHealthVet [16] of the US Department of Veteran Affairs) combines patient entered content with EHR data

Health consumers are migratory, they move from one health institution to another. Thus the PHR must be portable. Nonetheless, without portable PHRs the number of medical errors may increase, which affects around 100,000 patients annually in the US as was shown in the landmark study by the Institute of Medicine in 1999 [17]. Until today medical errors have not diminished. Not having portable PHR does not only affects the patients but also has an increased financial impact on the health care industry.

Many PHRs are standalone products (non-tethered) with all information self-entered by the consumer. Others were integrated with official EHRs (tethered) offered by a health care provider or insurer. Some PHR vendors provided both stand-alone and integrated versions of their products, depending on whether the vendor was working directly with a patient or with a health care provider who, in turn, made the product available to patients.

Features varied, such as whether or not prescription refills were available through the PHR and whether or not secure messaging between providers and patients was available. Some PHR products actually allowed patients to see portions of their official EHRs.

The purpose also varied from one product to another. While a number of PHRs offered an overall health history to simplify health information exchange, others focused on a particular aspect of health, such as chronic pain or end of life issues, or on a specific population, such as children or migrant workers.

Most of the digitizing healthcare data initiatives are very much implemented to serve either a medical institution or people under some condition and infrastructure, which is mostly ruled by an organization rather than serving the person at his regular pace. A patient cannot have ideal portability level with existing contributions as they are designed to be used with a specific infrastructure with the assumption that all hospitals in the world follow their own standards. As some of them focuse on the healthcare data portability between organizations more than thinking about the patient who actually owns the data.

1.2 Research Objectives

The main objective of this research is to improve the portability of the Personal Health Records by proposing an improved version of the Personal Health Book by having the Personal

Health Book on the web and stand-alone synced together, where the Personal Health Book data can be accessed at anytime and from anywhere.

In this work, we define portability as *"The ability for a person to access, share and control/manage own data from anywhere at anytime"*, compared to what we observed from the existing contributions, no one is fulfilling all of the data portability attributes as mentioned in the definition. Even with the existence of systems/applications to integrate PHRs, portability of PHRs is still an issue due to the variable standards of existing PHRs, different technology back bones of medical firms, patients IT literacy and countries infrastructures.

Current contributions of Personal Health Records portability are divided into Web-based and stand-alone PHRs, which have problems with their dependencies as the:

Web-based: is fully dependent on the availability of an Internet connection, and if it was available, the rate of access might be reduced due to its quality and the compatibility of the used device. Sharing also depends on the shared target readiness to accept the sender's PHR format.

Stand-alone: is fully dependent on the availability of a PC/device to view the PHR, and that device operating system should be compatible with the PHR application on the stand-alone device. As well as it needs a compatible port for connectivity if the stand-alone device was not common. Sharing also depends on the shared target readiness to accept sender PHR format as well as if target was physically present at the same place with the sender, and his PHR application was compatible with the running operating system (if receiver was a stand-alone too).

There are minor contribution in combining these two formats to make them work as one system. However, they are designed for targeted audience and technology/standards.

1.3 Research Contribution

The main contributions of this research is to improve the portability of the Personal Healthcare Data. We proposed a Personal Health Book (PHB) architecture. We developed algorithms to efficiently import Personal Health Records history from previous hospitals and share them with the new hospital to visit. We have designed a PHB decomposed database that can efficiently store the imported data without loosing any original information. We tested our algorithms and database design by using a simulation developed by us.

These three contributions are described below:

First, design a decomposed data types managed database that is small in size and can be used for storing Personal Health Records on a Smart Card and with the same structure concept that can be used on the web-based Personal Health Book. Unifying the concept makes it easier to encourage healthcare system developers to adapt with the concept.

Second, the algorithm for importing Personal Health Records into the Personal Health Book. This algorithm along with a simple/minimum XML structuring requirement, Personal Health Records can easily get imported into the Personal Health Book. Regardless of what the content is and how it is formatted or named.

Third, the algorithm for exporting Personal Health Records from the Personal Health Book. With this algorithm, the Personal Health Book can export Personal Health Records to any standard or format as it can understand the database design concept and get the data from it easily. The code following this algorithm can be placed in a framework/API to connect between a Personal Health Book and any other authorized party that needs to export Personal Health Records from the Personal Health Book.

The PHB can benefit an individual in different ways. (1) An individual can easily share his/her life-time data with a doctor/hospital more efficiently (short period of time with less errors) so that the doctor can make a better decision. This will also open a new field of research on how to visualize the life-time healthcare data to demonstrate it to a busy doctor. (2) An individual can monitor and compare his/her health data to better understand his/her health status. (3) As the data will be stored both on the web and on a personal device, it can be used in places where medical support is limited such as under-served communities, and during emergency situations such as a disaster.

1.4 Thesis Outline

This section briefly presents the outline of this thesis and overviews of each chapter.

Chapter 2 explains the research motivation, discusses the current research and commercial initiatives on personal healthcare data management, evaluates the status of portability of different approaches, lists the requirements of personal healthcare data portability.

Chapter 3 proposes our approach to improve the portability of Personal Health Records and compares it with the existing Personal Health Book. This chapter focuses on the design and architecture of the proposed Personal Health Book.

Chapter 4 describes the database design of the Personal Health Book, the implementation of the simulations and experiment to evaluate the database design and the algorithms for importing & exporting Personal Health Records.

Chapter 5 presents the application of the Personal Health Book and demonstrates the use-cases of the Personal Health Book for developing countries, disaster affected areas and to create healthcare awareness among the individuals.

Chapter 6 concludes the core contributions of this research, and discusses the scope and limitations of our approach and points out several directions for future works.

Chapter 2

Research Motivation

2.1 Introduction

The exponential growth of data being created, uploaded and stored in remote IT databases is beyond extraordinary. Much of this accumulating data identifies individuals, their attributes and behavior traits, and is used by third parties for Big Data analysis and profit. Known as the 'personal data landscape', an individual's own data is often not available so individuals cannot analyze and assess their own lives. This is particularly relevant to health data, which is captured and stored by disconnected healthcare organizations, such as hospitals and is largely unavailable in a digital format to the patients.

This chapter will introduce personal data and some innovative concepts. It will focus on personal data portability of personal health data and steps taken to improve the interoperability issues between data systems. It will conclude with the basic requirements for personal health data portability.

2.2 Personal Data and its Management

2.2.1 Personal Data Landscape

Interconnecting data networks and commercial interest in consumer behavior and purchasing trends characterize the personal data landscape[1]. Hyper connected services (social networks, telecommunications, corporations, hospitals) are updated by billions of connected devices, as people and sensors record trillions of transactions and behaviors each day[2] leading to petabytes of data records being stored in millions of databases. Many organizations use data analytics to gain monetary value out of volunteered personal data. Commercializing personal data has led to heated debate over data ownership[3] and protection, security, and context of freely given information.

There is less discussion on understanding what is being uploaded to the personal data landscape and its exponential growth as personal data is replicated over as it passes from one data network to another (Figure 2.1). This section will define personal data, simplify the information type to persistent data and transitory data and state how it is created, retrieved and management over a person's lifetime.

2.2.2 Personal Data and its Management

According to EU Data Protection Directive (95/46/EC) [18] "personal data" shall mean any information relating to an identified or identifiable natural person ('Data Subject'); an identifiable person is one who can be identified, directly or indirectly, in particular by reference to an identification number or to one or more factors specific to his physical, physiological, mental, economic, cultural or social identity. The EU's definition is a detached view that aims to protect the privacy of an individual. It does not address how the individual can use and benefit from his or her own personal data. Personal data is actually a valuable digital resource for a person and the society. Advancement of ICT solutions will improve the management of

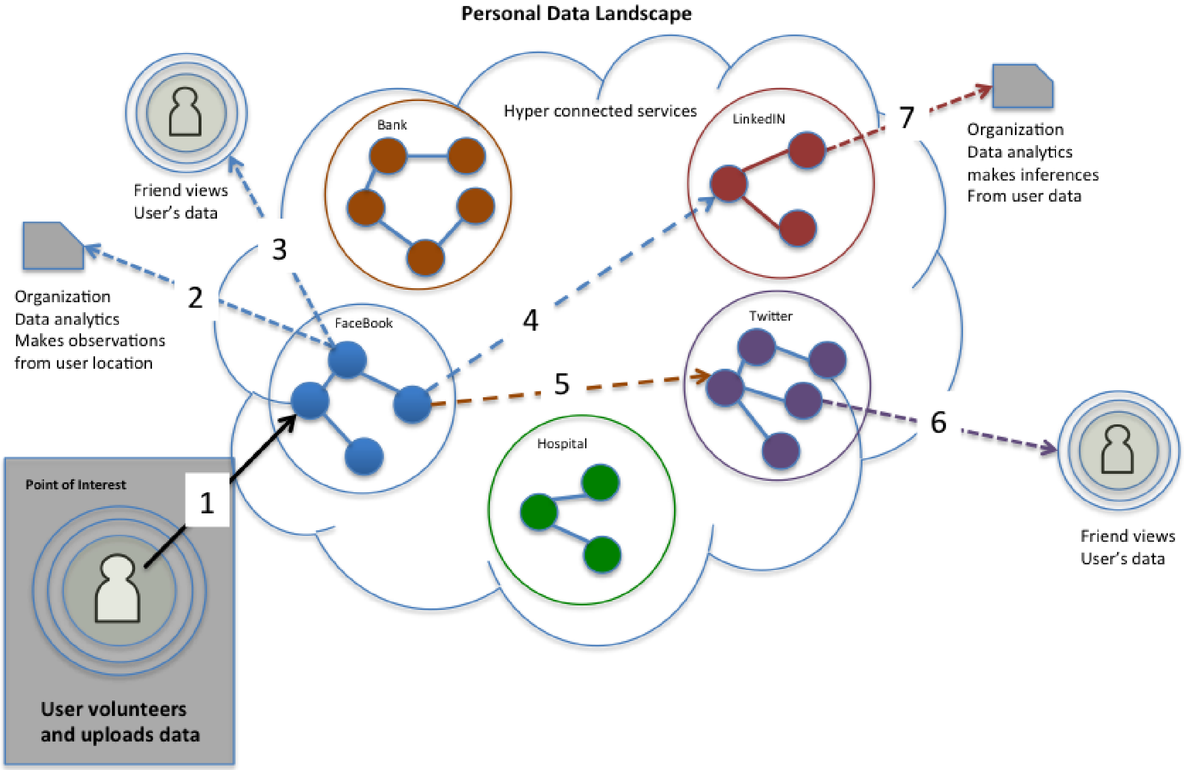


Figure. 2.1: Data Flow and Replication (steps 1-7) in the personal data landscape. At step 1, a user uploads his/her data on a social network (e.g. facebook), the same data can go to other organizations and social networks with or without the consent of the user as labeled by 2-7.

own data and enhance a more efficient interaction between an individual's data and institutional data networks in the personal data landscape.

2.2.3 Personal Data Creation

Personal data creation concerns persistent records that specifically identify an individual. It adds value to personal data and includes:

(1) **Personal data created on-behalf of a person and managed by organizations:** such as Electronic Health Records used in hospitals[19].

This type of personal data includes long-lived records such as academic certificates, professional certificates, employment records, health records[20] and residential details. The records are usually stored in disconnected or semi-connected data networks such as a school data warehouse and hospital servers.

(2) **Personal data created by a person and volunteered to the personal data landscape:**

This type of data includes comments, digital photos, and multimedia material. Data records are usually stored and managed in big connected environments such as social networks or service providers - Facebook, Google+.

(3) **Personal data created by a person but not shared and kept in a closed environment:** such as a personal computer [21][22].

This type of personal data includes digital photos, music files, and audio memos. Data records are usually stored in disconnected environment such as personal computer, optical disk, USB, and on a hard disk drive.

Personal data is not available at the individual's side, thus it should be retrievable.

2.2.4 Personal Data Retrieval

Personal data retrieval concerns access and use of persistent records that adds value to the individual's welfare. Personal data retrieval can be achieved by setting up a personal data locker between data stores and requesting parties. Personal data lockers are part of the Big Data dialogue [4] [5] and in the concept, organizations hand back data to a personal data store.

To access the personal data store, an individual may use a Personal Identity Module (PIM) to query and fetch relevant personal data from the data bank and store it offline in a device (Figure 2.2). A requesting third party could access the data bank after an agreed authentication process. Data lockers enable data portability because a person can share their details with service/person at anytime.

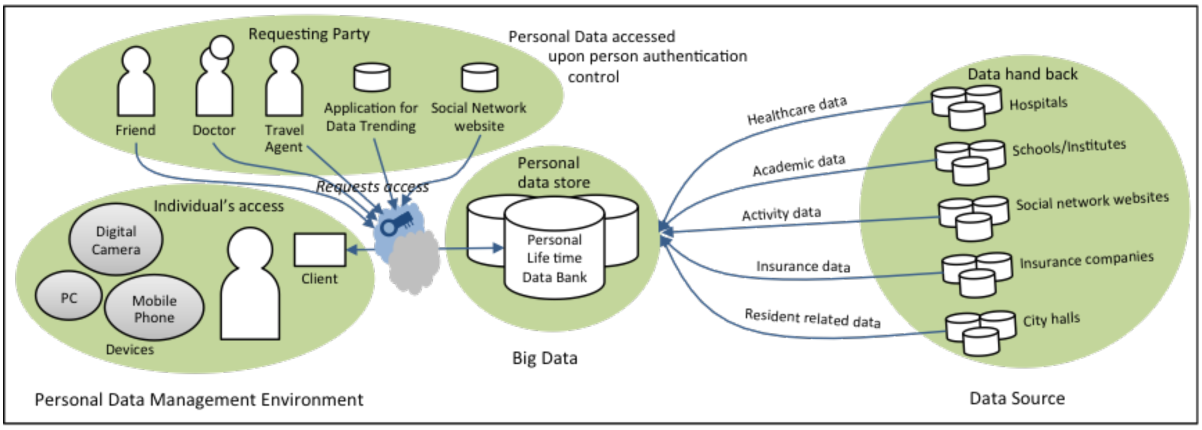


Figure. 2.2: Personal Data stores can used to concentrate personal data.

2.2.5 Personal Data Smart Management

Personal data smart management associates a person's transitory data to a specific location & time and saves the transaction. Mobile devices are used for data transactions, such as for uploading medical records to online health databases [6] [7]. Smart cards are used by offline applications to authenticate an individual [8]. Smart cards are useful in developing countries where individuals have limited personal data records. An example is the ePassbook application that enabled micro-finance members to maintain their financial records in Bangladesh [23] (Figure 2.3).

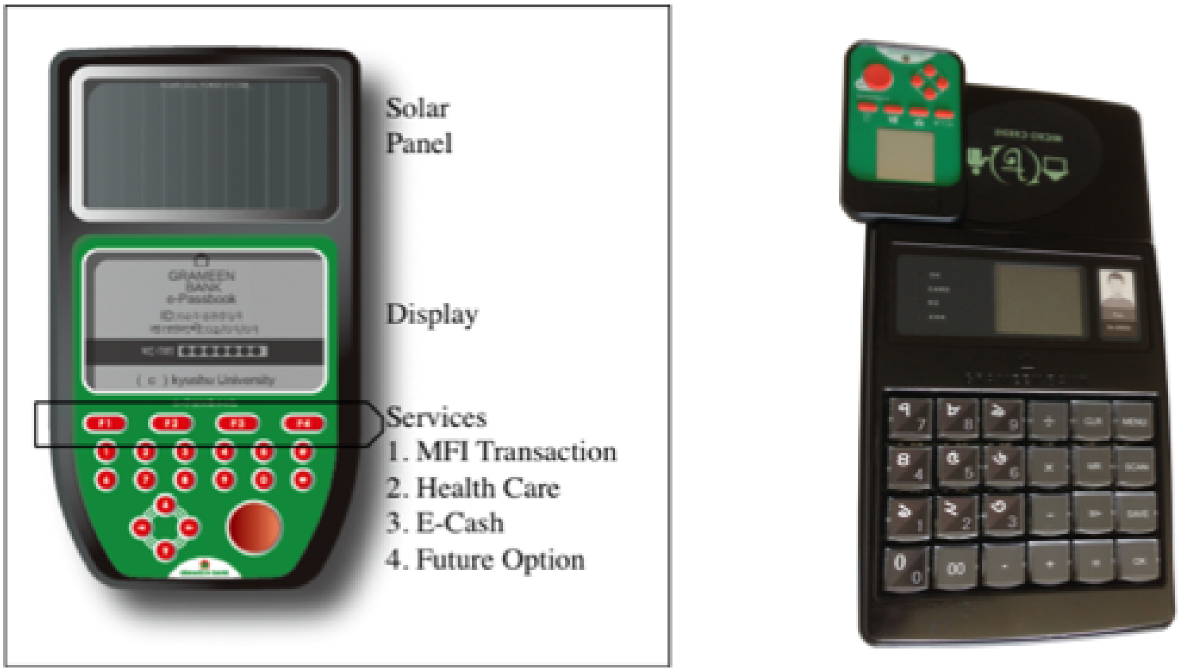


Figure. 2.3: ePassbook smart card and ePassbook reader for BOP.

The ePassbook offered multiple services (microfinance, health [24] and e-cash) to the rural poor at the Base Of the Pyramid (BOP). It stored Micro Finance transactions and health records. The individual viewed his or her own records using a display attached to the card. The ePassbook could transfer records and update a microfinance, health, or purchase transaction by using the inbuilt RFID communication interface to an ”ePassbook reader”.

The ePassbook can be enhanced to allow the IC chip to store a core union of citizen data and interface with a multitude of services. The ePassbook reader (Personal Data Controller) should communicate with the external networks and update the smart card (Figure 2.4). Upon a registration to a data network, the PDC should send personal data to an external data network (no filling of application form). Upon a successful commercial transaction with a data network, the receipt, e-ticket, or contract would be sent to the individual's account (e-mail).

It is important that the individual can modify ePassbook's citizen data by selecting pertinent information and deselecting sensitive information and save transactions offline. It is also critical that the information systems used to manage the person's data are secure from hackers and subversive threats.

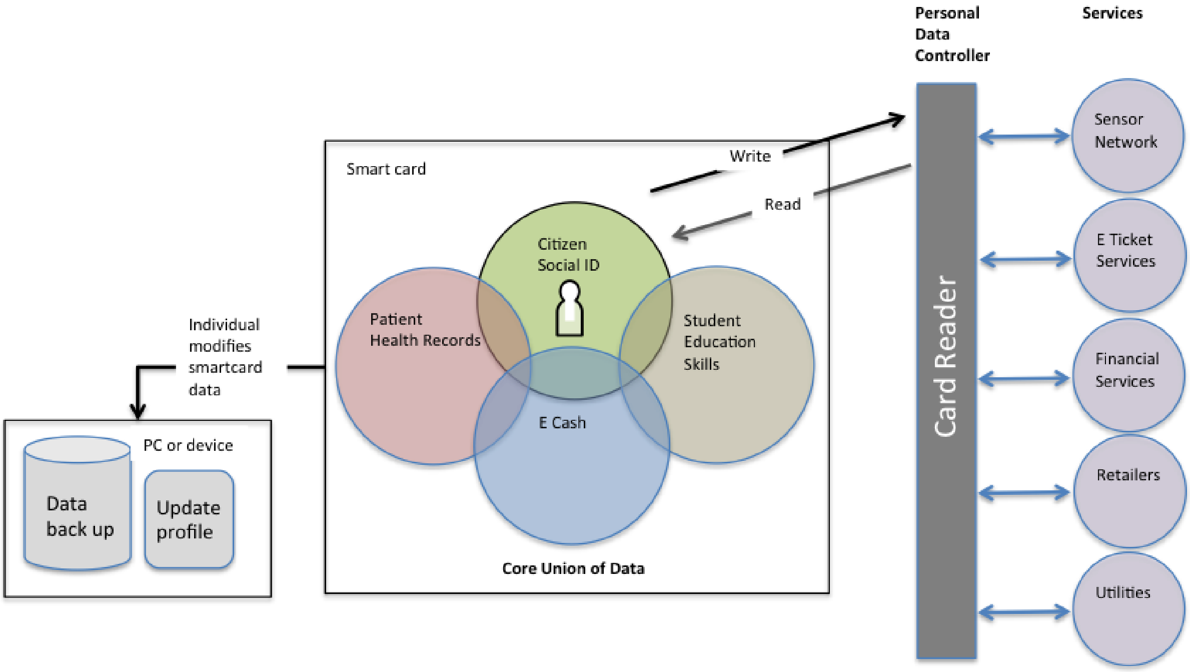


Figure. 2.4: Smart Card uses Personal Data Controller to interface with external services.

2.2.6 Personal Data Over a Lifetime

Personal data over a person's lifetime merges persistent data and transitory data and models their combined use over an individual's lifetime[25]. Persistent data is associated with clearly defined roles, achievements and images assigned to a particular time in a person's life. Transitory data is linked to consumption transactions and related to economic well being of a person.

Medical data is persistent and transitory because health changes over time. The expected volume of records produced by persistent and transitory data are shown in Figure 2.5. Persistent data would produce a low number of data records (academic qualifications and employment skills). Transitory data would contribute to a high level of data records (shopping receipts, tickets, smart meters). Medical data will have its own trend; it may decrease after childhood and then increase as a person approaches middle age. Showing frequency of data records against age will be of interest to different cohorts and may stimulate new research questions.

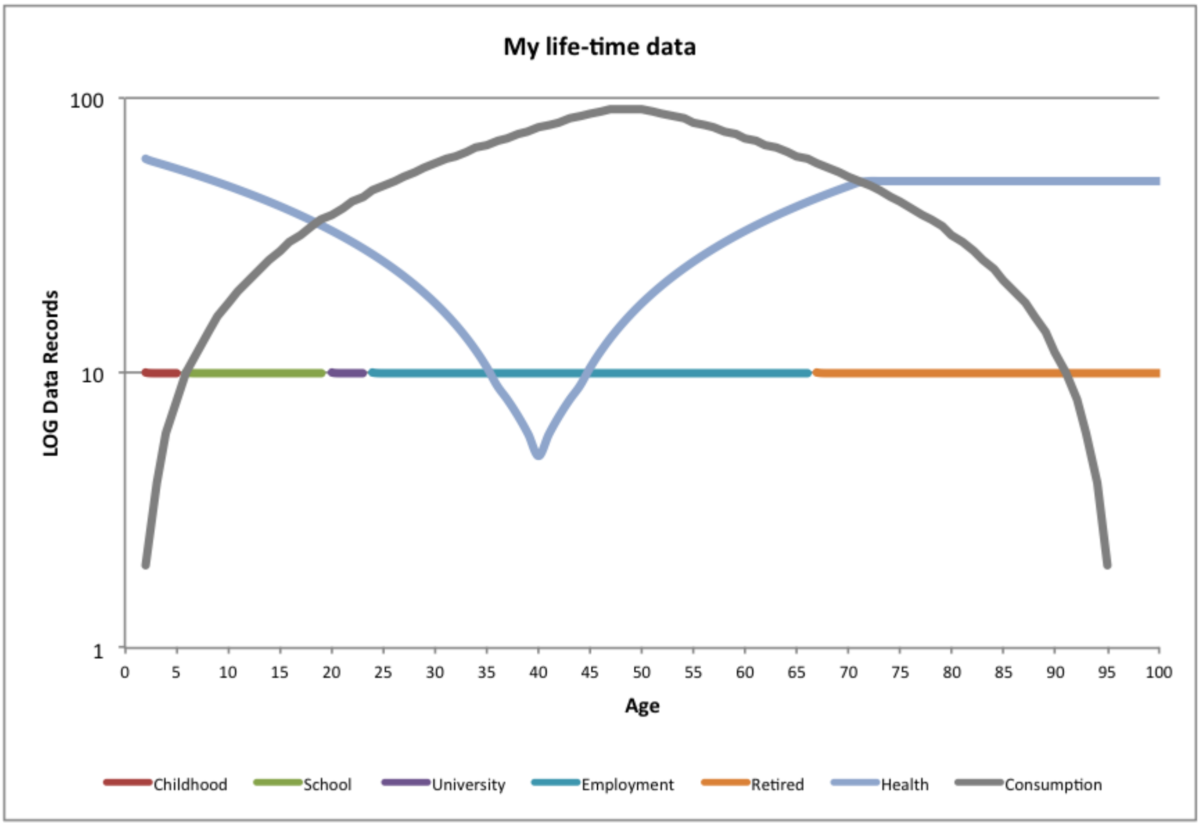


Figure. 2.5: Personal lifetime data shown as frequency of records against age.

2.3 Personal Data Portability

To enable an individual to gather, store and share their personal data assumes that the data records can be ported over different interfaces and between different data systems. There are initiatives to support personal data portability [5]. The general understanding is that "Data portability concerns the user being able to move and use own data across space and across time.[26]" and "Data portability is the ability for people to reuse their data across interoperable applications - the ability for people to be able to control their identity, media and other forms of personal data[27]". However, most personal data portability concepts depend/focus on data on the Internet and how it crosses interoperable applications through the Internet space. Some focus on personal data on the social networks only, and others build applications/frameworks for each service in which each service has its own framework.

For meaningful use, personal data portability should be fully independent from any technology, it should not be focused on one domain like social networks, but should be applicable to all domains that house important data (health, profession, consumption) that represents or effects an individual's life.

In this work, we define Data Portability as *"The ability for a person to access, share and control/manage own data from anywhere at anytime"*. With "anywhere" we refer to any geographical location, any technology and any application. Matching these criteria will be the most ideal portability (100% portable). This is not possible with disconnected data networks such as electronic health records within hospital and healthcare organizations.

2.4 Personal Healthcare Data

2.4.1 Electronic Health Records

The Electronic Health Record (EHR) was designed for billing and hospital administration purposes. It is a longitudinal electronic record of a patient's health information, generated by one or more encounters in the care delivery setting. Included in this information are the patient's demographics, progress notes, problems, medications, vital signs, past medical history, immunizations, laboratory data, and radiology reports [28].

The EHR is supposed to automate and streamline the clinician's workflow and has the ability to generate a complete record of a clinical patient encounter, as well as support other care-related activities directly or indirectly via an interface including evidence-based decision support, quality management, and outcomes reporting[29].

It is a record in digital format that is theoretically capable of being shared across different health care settings. In some cases, this sharing can occur by way of network-connected, enterprise-wide information systems and other information networks or exchanges.

EHRs may include a range of data, including demographics, medical history, medication and allergies, immunization status, laboratory test results, radiology images, vital signs, personal statistics like age and weight, and billing information [30].

Many healthcare organizations have implemented complex EHR systems, however, they all seem to experience compatibility and data portability problems. Hospitals tend to use their own standards, which makes it costly and time consuming to develop and maintain EHR systems and very difficult to send clinical records between institutions.

Electronic PHRs stored in hospitals share the same main components:

(1) Patient's details/profile

(2) Hospital details

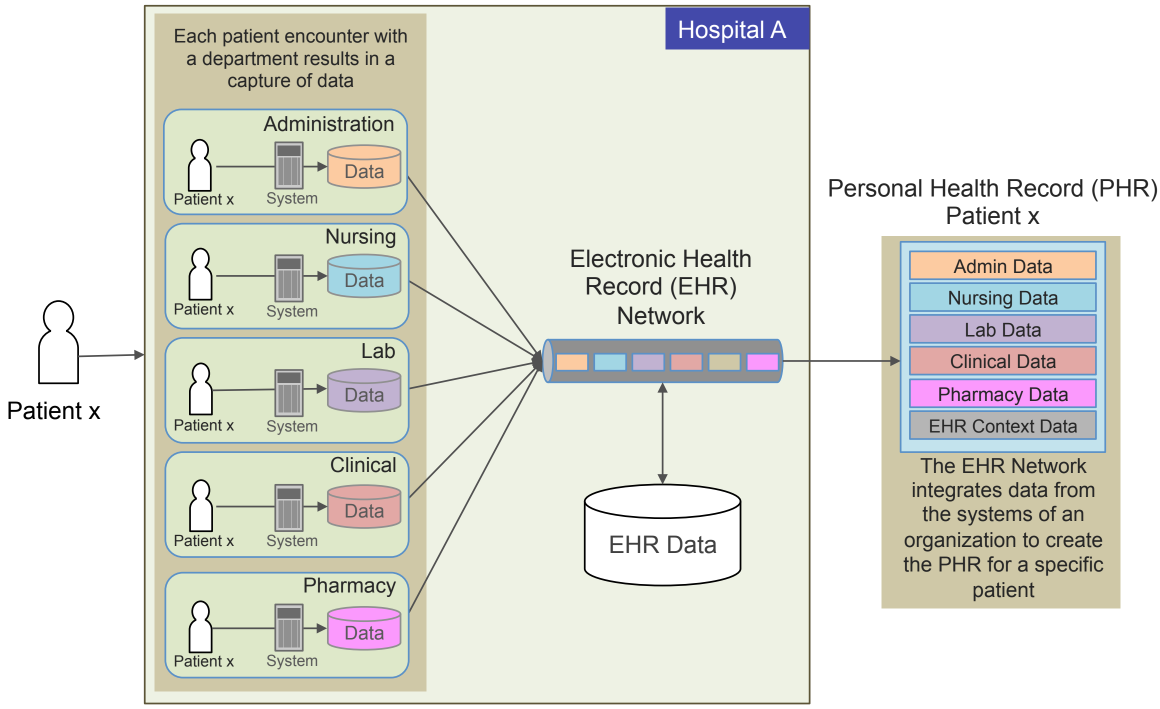


Figure. 2.6: Personal Health Records from EHR Concept Overview

(3) Doctor's/Physician's details

(4) Measurements details

(5) Drugs details

From these components, we find that some tend to be almost static (Patient's details), some will be semi-static (Hospital details and Doctor details), some will be static in details but growing with new data (Drug details) and the last component will be very dynamic and always contains new data (Measurements details).

Different hospitals generate different EHR's that will differ in data type (integer, char, date) or different format (size of each data, storing order of data) and data structure (names and number of items). Therefore it is necessary to design a database that can accommodate these variable natures of the source data.

2.4.2 Personal Health Records

Personal Health Records (PHR) are a set of computer-based tools that allow people to access and coordinate their lifelong health information and make appropriate parts of it available to those who need it [4][5].

PHR systems were developed in the late 1990s to target patients who were travelling and needed healthcare and for situations where patients were not able to provide their health information [6][7], or have communication difficulties.

Currently, three types of PHRs have become available [8]. They are:

(1) Stand-alone formats (PC, USB drive), where consumers store health information on personal computers but lack the ability to exchange information between consumers and healthcare provides.

(2) Web formats that are managed by third parties (such as Microsoft Healthvalut, Dossia Consortium) and allow consumers to maintain their health information on private online accounts accessed by a login ID and password.

(3) Integrated PHRs with Electronic Health Records (EHR) where a healthcare provider (such as MyHealtheVet of the US Department of Veteran Affairs) combines patient entered content with EHR data [16].

Although PHR systems have been available for over ten years, there is low uptake by the consumer due to many reasons, including the availability and of their own medical records in digital form and portability from healthcare organizations.

2.5 Current Initiatives Regarding Personal Healthcare Data Portability

There has been considerable effort to address the problem of personal data portability between data systems. There are standardization activities, commercial applications and academic research into the problem. The following sections and the remainder of the research will focus on healthcare data.

2.5.1 Electronic Health Standards/Protocols

Health Level 7 (HL7)

Health Level 7 (HL7) is an ANSI-accredited standards developing organization dedicated to providing a comprehensive framework and related standards for the exchange, integration, sharing, and retrieval of electronic health information that supports clinical practice, management, delivery and evaluation of health services [31]. HL7 maintains some of the most widely used standards for clinical information exchange around the world.

One of the HL7 tools is the Virtual Medical Record (vMR) [32] and is based on the HL7 V3 (Health Level 7 version 3) model and data types. It is a ”virtual” interface that is optimized for a point in time of clinical decision support. While it could be used as the basis for an EHR design, it is intended to be simplified to represent a snapshot and to omit important concepts that should exist in a real EHR.

OpenEHR

The OpenEHR [33] foundation is an independent, non-profit community that facilitates the creation and sharing of health records by consumers and clinicians via open-source, standards based implementations. It is based on ISO 18308 Conformance Statement. OPenEHR solutions have been implemented in hospitals in various countries. The OpenEHR approach enables a platform-based e-health software market and allows vendors and developers to use OpenEHR standardization to create back-end and front-end solutions interface via content models and terminology, and standardized service interfaces.

2.5.2 Extensible Markup Language (XML)

Open XML can be used to access EHRs and exchange health data. In fact, HL7 CDA is a document markup standard based on XML. The XML specification defines a standard way to add markup to documents [34].

The XML was created so that richly structured documents can be used and exchanged over the Internet. It is now common that almost all data systems support XML and its principal use is a data carrier/mediator between applications on the Internet or locally between applications. Research has shown that it is recommended to use XML as a standard to support data interoperability between healthcare systems [35], [36], [37] and [38].

2.5.3 Commercial Initiatives

Commercial applications mainly concern Internet secure health vaults. A health vault site is a remote database that stores individual health information. Access to a record is through a commercial account and there can get access for multiple individuals, so parents can manage the medical issues of their siblings or vice versus.

Microsoft HealthVault

Microsoft HealthVault [14] gives a person one place to access all of his health information online, with many ways to add information and connect with data from his healthcare providers. Unfortunately, hospitals cannot import directly to Microsoft Health Vault (Figure 2.7), unless they use a system familiar to them [39] (for example MS SharePoint or Dynamics). Otherwise, they need to type the data manually in the Health Vault website or through some third party paid services.

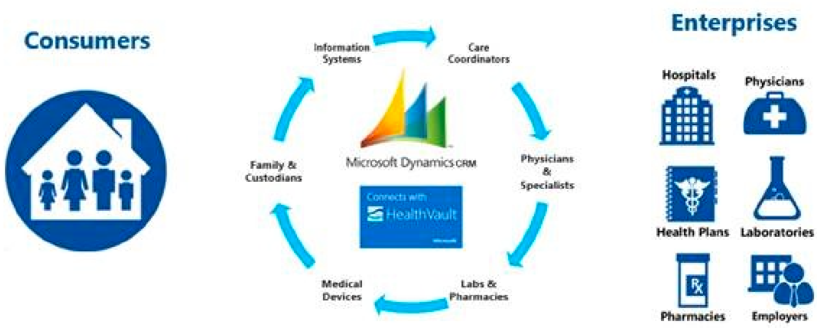


Figure. 2.7: MS HealthVault allows medical firms to connect automatically when using MS product

Indivo X

Indivo X [40] gives a person one place to access all of his health information online, with many ways to add information and connect with data from his healthcare providers. However, a hospital needs to follow indivo X standardization to be able to import a PHR, which costs money and time if every hospital had to follow their standard.

My HealtheVet

”My HealtheVet” [16] is Veterans Affair’s (VA) online PHR. It was designed for Veterans, active duty service members, their dependents and caregivers.

”My HealtheVet” helps the patient’s partner with their health care team. It provides opportunities and tools to make informed decisions and manage the patient’s healthcare.

With ”My HealtheVet”, America’s Veterans can access trusted, secure, and current health and benefits information as well as record, track and store important health and military history information at their convenience.

Registered Veterans enrolled in a VA facility can refill their VA prescriptions and in the future, view their VA appointments, receive copies of their VA electronic medical records, and electronically communicate with their health care team.

More than 1 million Veterans are now using secure messaging to communicate with their VA health care team. Even with all these features, ”My HealtheVet” is designed for the needs of veterans only, for specific health care types and only from VISN 4 VA medical centers. Sharing capabilities are only by viewing from the patient’s screen, print or download; there is no electronic sharing [41].

This means ”My HealtheVet” does not give patients the full access or the ability to manage or share their PHRs.

2.5.4 Academic Initiatives

Academic research tends to focus on patient-centered cloud based solutions and hardware devices.

Personal Health Book

Similar to Web formats and online PHRs health vaults, the Personal Health Book-based healthcare model is a web-based service where consumer PHRs are stored at a remote server and is shared with healthcare providers (such as a pharmacy) whom the consumer authorizes and has the capacity to import data from other information sources [42].

It targets medical firms with a specific technology on their side and patient with specific skill set (owning a device with web-browser and ability to use it) and specific access type (connectivity to the internet). Without an Internet connection, this Personal Health Book system will not be able to function.

Portable Digital Personal Health Record

A proposal to have a stand-alone PHRs stored on a Universal Serial Bus (USB) named Digital PHR (DPHR) [43]. As in the USA, computers are now widely used in the market, and many adult individuals have a basic working knowledge of a computer's function. The USB is the standard feature in many current computers.

The training period for learning new software could be considerably decreased by using easy to use templates, which are universally accessible by all computers regardless of their operating software or processor speed. Not only individuals can store their illness information in their DPHR storage devices but also they can maintain a wellness health record over a period of time. An optimal individual record should carry within it emergency information of an individual along with relevant demographic information, in addition to all the medical and surgical history. DPHR can be rapidly accessed using any standard or portable computer with any operating system.

2.5.5 Summary of the Current Initiatives

Many hospitals and healthcare organizations are implementing complex EHR systems, however since there are multiple choices on which standard and solution to follow, there is limited data portability between healthcare organizations and networks. This occasionally makes physicians question the viability of contemporary EHR IT systems.

As EHR systems are designed for care infrastructure managed by administrators, there is limited scope for individuals to access their own EHRs and build PHRs. This affects and restricts commercial application that allows consumers to store and manage their own health data using PHRs.

Academic researchers assume that consumers can access their own EHR and save them as PHRs on the cloud or in devices. An alternative viewpoint is to enable the individuals to interface with the care system and access their own health data.

Figure 2.8 summarizes the portability performance of the discussed contributions.

2.6 Basic Requirements for Personal Healthcare Data Portability

Accepting that individuals can access, manage and share their own PHRs, there are many issues with effectively managing personal health data. The basic requirements for adoption are usability, awareness, access, meaningful use, and clinical integration.

2.6.1 Usability of Personal Health Records

The primary key for personal healthcare uptake including data portability is that the system must be easy to use. Most health consumers are migratory, they move from one health insti-

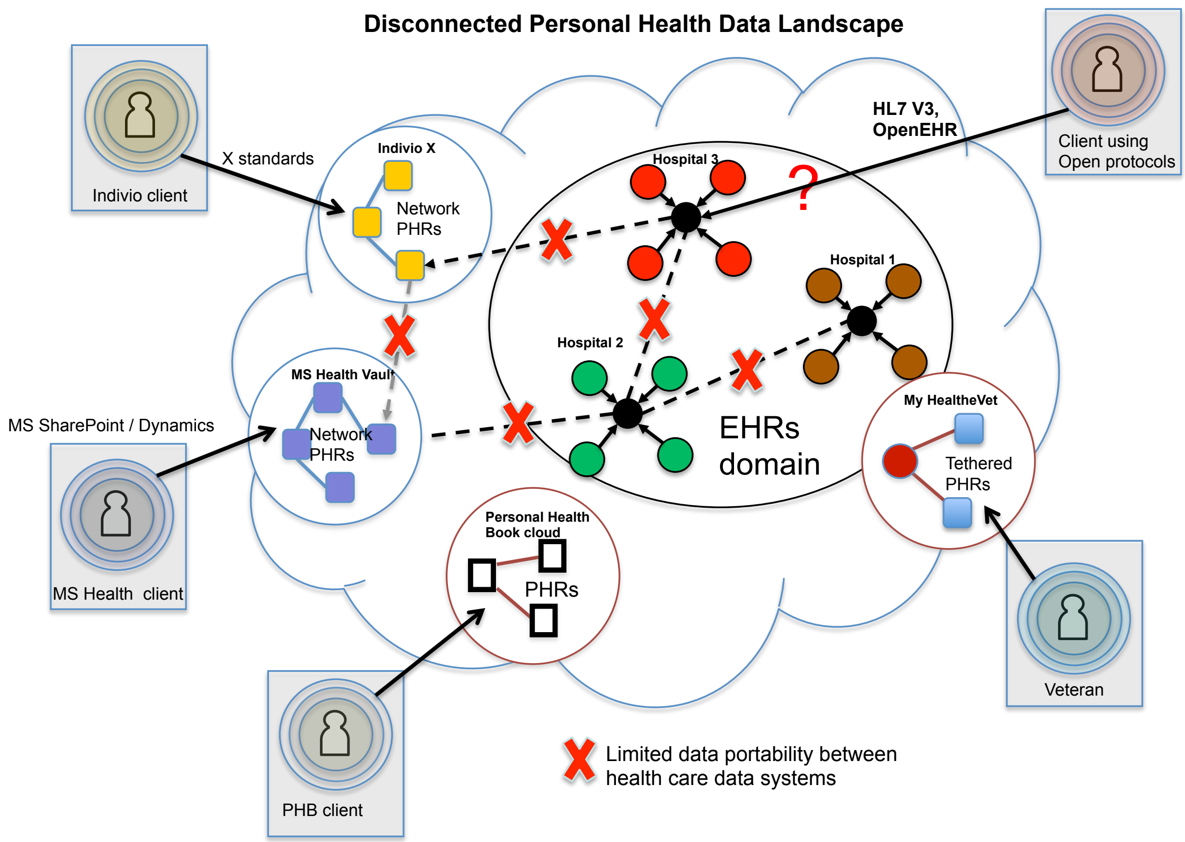


Figure. 2.8: Current contributed healthcare records portability.

tution to another. In the developing world, economic migration is emphasized with millions of illiterate people frequently moving from one region or country to another. Thus, a main criteria for portable PHRs is usability [44]. This is also applicable to the healthcare organizations receiving PHRs from new patients or migrants. An improvement in PHRs usability may decrease the number of medical errors, which affects around 100,000 patients annually in the US as was shown in the landmark study by the Institute of Medicine in 1999 [17]. Till today, medical errors have not diminished. It not only affects the patients, but also has an increased financial impact on the healthcare industry.

Awareness of Personal Health Records

The second most important requirement is the awareness of PHRs' tools and their value. Most individuals and many healthcare organizations are not aware of PHR systems and the rights of individuals have to access their own personal data.

Alongside the awareness of PHRs is e-health literacy. The individual must be computer literate and can make effective use of a multi-choice application. At the basic level, an individual's personal health information is recorded and stored in personal computer-based software and the consumer must have the capability to print, backup, encrypt, and import data from other sources such as a hospital laboratory. The simplest form of a PC-based PHRs would be a health history created in a word-processing program. The health history created this way can be printed, copied, and shared with anyone with a compatible word processor.

PHRs software applications provide more sophisticated features such as data encryption, data importation, and data sharing with health care providers. Some PHRs products allow the copying of health records to a mass-storage device such as a CD-ROM, a DVD, a smart card [8], or a USB flash drive [9][10].

PC-based PHRs are subject to physical loss and damage of the personal computer and the data it contains. Some other methods of device solutions may entail cards with embedded chips containing health information that may or may not be linked to a personal computer application or a web solution.

Access to Personal Health Records

The common method to access PHRs is for health consumers to connect to the Internet. Web-based PHR solutions are essentially the same as electronic device PHR solutions. However, web-based solutions have the advantage of being easily integrated with other services. For example, some solutions allow importing medical data from external sources. Solutions including HealthVault, and PatientsLikeMe [45] allow for data to be shared with other applications or specific people. Mobile solutions often integrate themselves with web solutions and use the web-based solution as the platform.

A large number of companies have emerged to provide consumers the opportunity to develop online PHRs. Some have been developed by non-profit organizations, while others have been developed by commercial ventures. These web-based applications allow users to directly enter their information such as diagnosis, medications, laboratory tests, immunizations and other data associated with their health. They generate records that can be displayed for review or transmitted to authorized receivers.

Despite the need for PHRs and the availability of various online PHRs providers, there has not been wide adoption of PHRs services. In fact, Google, being among the most innovative companies in the world, discontinued its PHRs service called Google Health on January 12, 2012. The reason cited for shutting down Google Health was that the service did not translate from its limited usage into widespread usage in the daily health routines of millions of people [46].

Meaningful use of Personal Health Records

A PHR system must have a favorable user perception. This has to be achieved through Integrated PHRs with official EHRs (tethered) that are offered by a healthcare provider or an insurer. Some PHR vendors have successfully provided both stand-alone and integrated versions of their products, depending on whether the vendor was working directly with a patient or with a healthcare provider who, in turn, made the product available to patients.

Even though integrated PHR systems exist, they are very small in contribution to the public with specific technology/audience types [47].

2.6.2 Clinical Integration of Personal Health Records

PHRs should be considered as a source of information for healthcare professionals. Many PHRs are stand-alone products (non-tethered) with all information self-entered by the consumer. Others were integrated with official EHRs (tethered) offered by a healthcare provider or an insurer. Features varied, such as whether or not prescription refills were available through the PHRs and whether or not secure messaging between providers and patients was available. Some PHR products actually allowed individuals to see portions of their official EHRs.

The purpose also varies from one product to another. While a number of PHRs offered an overall health history to simplify health information exchange, others focused on a particular aspect of health, such as chronic pain or end of life issues, or on a specific population, such as children or migrant workers.

The Medical Library Association (MLA)/National Library of Medicine (NLM) Joint Electronic PHR Task Force examined the current state of PHRs [48]. Of the 117 PHRs identified, 91 were viable. Almost half were stand-alone products and a number of them used national standards for nomenclature and/or record structure. Less than half were mobile device enabled, while some were publicly available, and others were offered only to enrollees of particular health plans or employees at particular institutions. A few were targeted to special health conditions.

2.6.3 Personal Health Records and their Portability

Even with the existence of systems/applications to integrate PHRs, portability of PHRs is still an issue due to the variable standards of existing PHRs, different technology back bones of medical firms, patients IT literacy and countries infrastructures.

We have prepared a table (Figure 2.9), by listing the portability attributes (as defined earlier) and evaluated the approaches for PHRs compared to what we believe to be an optimum portable solution; assuming the users have the basic capability to use ICT applications.

Web-based and stand-alone PHRs have problems with their dependencies as the:

Web-based: is fully dependent on the availability of internet connection, and if it was available, the rate of access might reduce due to its quality and the compatibility of the used device. Sharing also depends on the shared target readiness to accept the sender's PHRs format.

Stand-alone: is fully dependent on the availability of a PC/device to view the PHRs, and that device's operating system should be compatible with the PHRs' application on the stand-alone device. In addition, it needs a compatible port for connectivity if the stand-alone device was not common.

Sharing also depends on the shared target readiness to accept the sender's PHRs format as well as if the target was physically present in the same place with the sender, and his PHRs application was compatible with the running operating system (if the receiver was a stand-alone too).

Feature		Medical Firms	Web-based	Stand-alone	Ideal
Digital Access	Anywhere	✖	●	▲	✪
	Anytime	✖	●	■	✪
Digital Share	Anywhere	✖	●	■	✪
	Anytime	✖	●	▲	✪
Control/Manage	Anywhere	✖	✪	✪	✪
	Anytime	✖	✪	✪	✪

✪ = Excellent ● = Good ■ = Fair ▲ = Poor ✖ = Not Portable

Figure. 2.9: Portability Status of Different PHRs

2.7 Summary

The ubiquitous accumulation of data in personal data landscape identifies an individual's behavior and is largely unavailable or underused by the individual. Individuals should be able to use personal data to analyze and evaluate their own behavior, as well as benefit by sharing with other entities. In particular, a person's health data that is stored in PHRs must be available to the individual so they can manage their own health status over their own lifetime.

In reality, there are many barriers to real personal healthcare including data portability of EHR and PHRs between medical organizations. As healthcare is a trillion dollar industry there are many ongoing efforts to improve the standardization or applicability of health data portability. However, most of these activities are focused on the administration of hospitals and clinics, and commercial organizations.

There is a lack of activity to place the control of PHRs with the health consumer or individual themselves. Most care stakeholders will say that people are not interested in managing their own health data, as evident with the failure of Google Health. Perhaps it is the mechanism and ease-of-use that is undeveloped rather the lethargic behavior.

In the next chapter, we will introduce an innovative Personal Health Book that organizes PHRs and can port health data from one health organization to another.

Chapter 3

Design of the Proposed Personal Health Book

The problem of sharing health related digital information between healthcare organizations and between healthcare organizations and individuals is often the way the Electronic Health Records and PHRs are structured. Usually in the form of a vendor specific data table, medical records between organizations differ in the way the data file is formatted and organized, such as the naming, order, type, count and size of the data categories and items. In which a data category can be an identifier (hospital) and data items the related data to the identifier (name, address and telephone number). How the data categories and items types are defined (INT, FLOAT, CHAR and DATE) and the related differences in the Byte sizes.

These technical differences provide a barrier for both importing data (getting) and exporting data (pushing) between healthcare organizations that want to share data. As this is a major problem between hospitals, vendors can introduce middleware to provide interoperability between medical administrations, however this adds additional layer of complication and cost, and does not benefit individuals and their access to their own medical data.

This chapter will examine the data portability problem between healthcare organizations and individuals and propose an architecture for the PHB that may bridge the interoperability barrier for importing and exporting PHRs.

The goal of the PHB is that individuals will have an easy-to-use mechanism to access their own PHRs and viewing own personal medical data that are meaningful. The technical objectives of a digital PHB are the following:

(1) Read PHRs from heterogeneous structured healthcare applications and importing them, without loosing or missing any data.

(2) Efficiently store the imported PHRs.

(3) Export the PHRs to the individuals and trusted parties to view and share their medical history.

The chapter will define the Personal Health Book, provide the high level requirements for our proposed Personal Health Book, and will describe the architecture for a database oriented Personal Health Book.

3.1 Introducing Personal Health Book

The PHB is a collective of a person's PHRs that are presented according to the meaningful use, with the ability to share, manage and control the records.

3.1.1 Paper Format Personal Health Book

The simplest form of a PHB is a hard copy paper book with pages recording an individual's health check-ups. This idea has been implemented many times. For example, in Australia, The New South Wales (NSW) Government has issued PHRs called the 'Blue Book'. It's a booklet, bound in a blue plastic cover, and given to all parents in NSW after the birth of a baby.

The Blue Book records the child's health, illnesses, injuries, growth and development as well as immunizations until the age of five. It is safely kept, as it is a valuable health information for the child and parents throughout their lives [49][50].

Likewise, in Nigeria, the Wellbeing Foundation has issued a 'Personal Health Records' book where all medical records of a mother and child are recorded and stored for future reference and analysis. The book records all immunizations of a woman (especially pregnant women) from the day of first presentation at the hospital when she delivers and until her child is five years old. Although it is a paper book, it is hailed as a ground breaking health resource to ensure effective delivery, evaluation and monitoring of sustainable, effective public health strategies in maternal newborn and child health in Nigeria.

The hard copy Personal Health Books are convenient for the user and the health professional as they can be physically taken to any healthcare organization and the information has a meaningful use for both parties. However, medical data and historical information are not digitally portable, although key information pages are often pre-designed for computer readability. Successful paper PHBs can guide us in terms of usability and meaningful use in how a digital PHB should be designed.

3.1.2 Web-based Personal Health Book

Juha Puustjarvvi and Leena Puustjarvi introduced an electronic version of the Personal Health Book. Taking influence from commercial health vault applications, the web-based PHB allows an individual (patient) to have a centralized application to access, share and manage their own PHRs instances. Once the consumer accesses their remote health vault (online database) they can search and order their PHRs.

Figure 3.1 shows the infrastructure of the web-based PHB and how it can be accessed over the internet by the patient, medical firms, family and friends. The PHB database is a collection of XML files using the standard format of Continuity of Care Document (CCD) [51] and Continuity of Care Record (CCR) [52]. In order to overcomes processing restrictions for trend analysis and online graphical views, the authors proposed a cloud-based solution [53].

The web-based PHB addresses data portability between healthcare organizations and individuals, as PHRs would be digitally distributed. However, the solution requires all parties to have devices connected the Internet, which is often not feasible in different health context (hospitals, pharmacies) and in different environments, particularly in developing countries. Accepting these limitations, the web PHB database can store PHR data for years, it is accessible from any online location, and can be shared to trusted third parties.

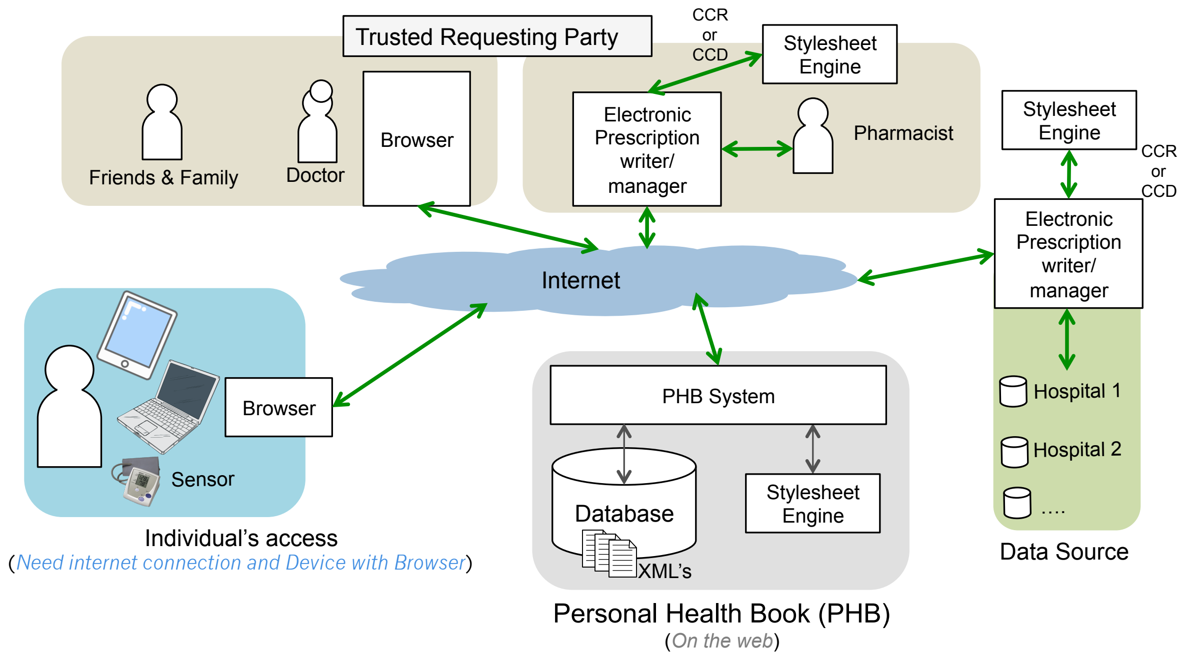


Figure. 3.1: Web-based Personal Health Book (PHB).

3.2 Proposed Personal Health Book

This research proposes a new version of the Personal Health Book. The proposed PHB has the best qualities of the paper PHB (usability and meaningful use) and the web-based PHB (digital accessibility and share-ability). To achieve this, we revisited the PHR "integrated approach" and focused on a stand-alone version of his PHB software application that stores different types of PHRs from different healthcare providers, with the capability to store on the web-based version. This is similar to a digital diary that houses ”personal lifetime data” and is an application with standard interfaces and its own database [25].

3.3 High Level Requirements of the Personal Health Book

The high level requirements of the PHB are to accept the healthcare organization’s imported PHR file efficiently, store the file and present it to the PHB owner in a unified view, and create an understandable report to the next health clinic.

In other words, the objectives are:

(1) Read PHRs from heterogeneous structured healthcare applications and importing them, without loosing or missing any data.

(2) Efficiently store the imported PHRs.

(3) Export the PHRs to the individuals and trusted parties to view and share their medical history.

To achieve these objectives there are three main PHB functions:

Gather function: import PHRs from various medical sources (clinics, hospitals), read PHR contents, and store PHRs into an Integrated PHB database.

Store function: save PHRs regardless of different data structure (names and number of columns) and different data characteristics: data type (integer, char, date); format (size of each data, storing order of data).

Share function: view the saved PHRs in a unified and understandable format. Viewing includes querying the saved PHRs to produce new types of reports. Unified means that although PHRs imported from various healthcare organizations will probably be in different formats, the PHB output will be in the same consistent format. This makes it easy for the viewer to read and analyze own medical history. Sharing also includes the PHB owner allowing access of the saved PHRs to trusted requested parties such as clinics, doctors and family. In all cases, the individual (the PHB owner) should have full control in terms of access, understanding, managing and sharing their own medical data.

A simple overview of the PHB is shown in Figure 3.2.

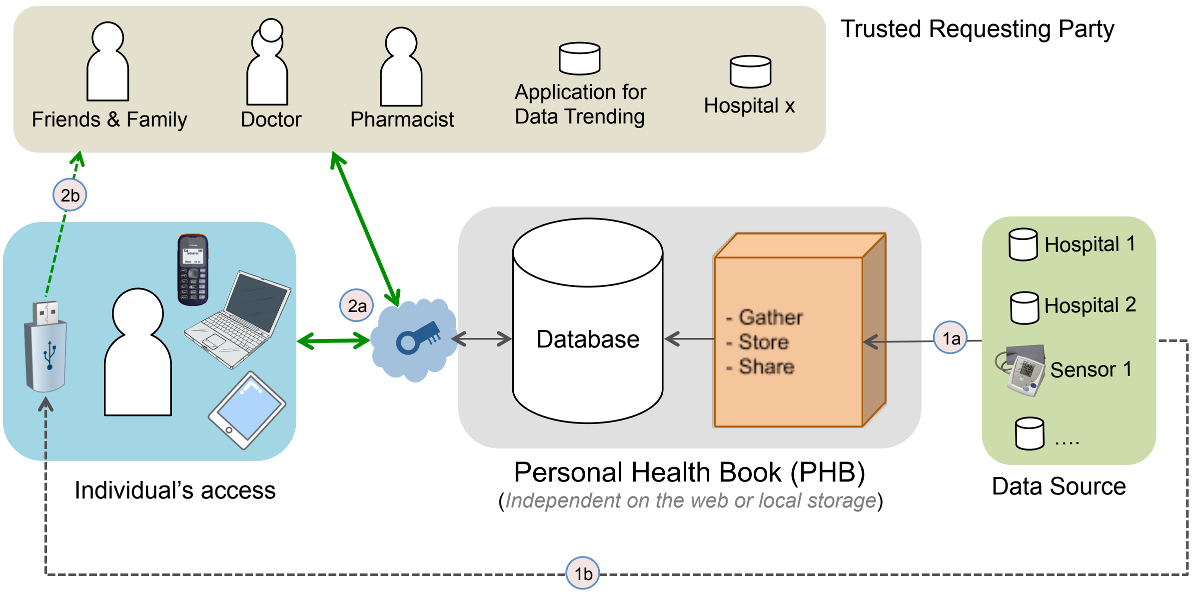


Figure. 3.2: Proposed Personal Health Book (PHB).

3.3.1 The Problem of Different Items and Formats

The most difficult requirement for the PHB is to import native PHRs from healthcare organizations when there are interoperability issues. The PHB 'gather function' tries to overcome this problem by expecting health providers to support XML as a common interface and the PHR as an XML document. This expectation does not fully solve the import problems as native PHRs have their own idiosyncrasies in terms of data formats and structures. Each care healthcare organization may have its own type of EHR and PHRs because of the IT vendor they chose for implementing their EHR systems or because of legacy reasons.

To conform to heterogeneity of native PHRs, the PHB has to resolve a range of data record concept common to all medical records:

- Document structure: with XML it is based on parent and child tree.
- Data Categories: considered as a parent or identifier (Hospital, Department, Doctor).
- Data Items: considered as a variable with label (name, height, urine protein).
- Common data items that are frequently used (name, blood pressure).
- Unique data items that are specific to a particular organization (specialist term).
- Data Values that are the values of the data items.

The number of data categories and data items in a scanned PHR will be different for each healthcare organization: be it internally, between hospital departments, or externally between hospitals or clinics. The labeling or naming of the item will also very between healthcare organizations. The values can have different meaning depending on the used scale.

Figure 3.3 shows the actual requirements of the PHB and internal structure. The gather function becomes a data import components and share function becomes a data export component.

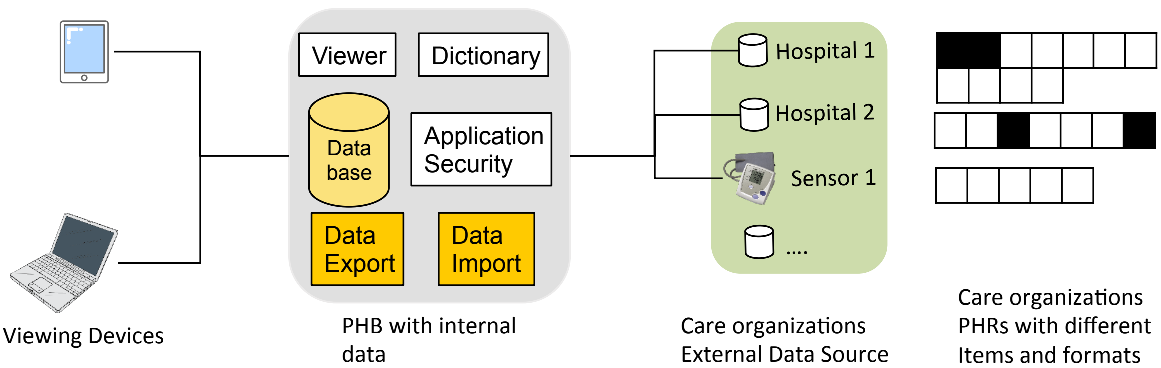


Figure. 3.3: PHBRequirements.

The expected variation in PHRs is demonstrated in Figure 3.4. There are different PHRs from different hospitals A, B, C & X. ”A” has less number of data items and has 2 unique items (A & B) that does not exist in ”B”, ”C” or ”X”. ”C” has 2 unique data items as well (F & G) that does not exist in ”A”, ”C” or ”X”. On the other hand, ”B” and ”X” don’t have any unique data items; that is all their items exist in ”A” and ”C” collectively.

Even for common data items, there will be variations in format or naming, for example, if we considered C from ”A”, ”B” and ”X” are common items and they represent body temperature, C from ”A” naming might be ”temperature”, but C from ”B” and ”X””s naming is ”body temperature”. Even though C from ”B” and ”X” have the same naming, but format might be different as C from ”B” value might be in Celsius and C from ”X” value might be in Fahrenheit.

If we compare data items from ”B” and ”X”, we find that they are similar in naming, number and order. One template will be generated and used for PHR’s coming from ”B” and ”X”.

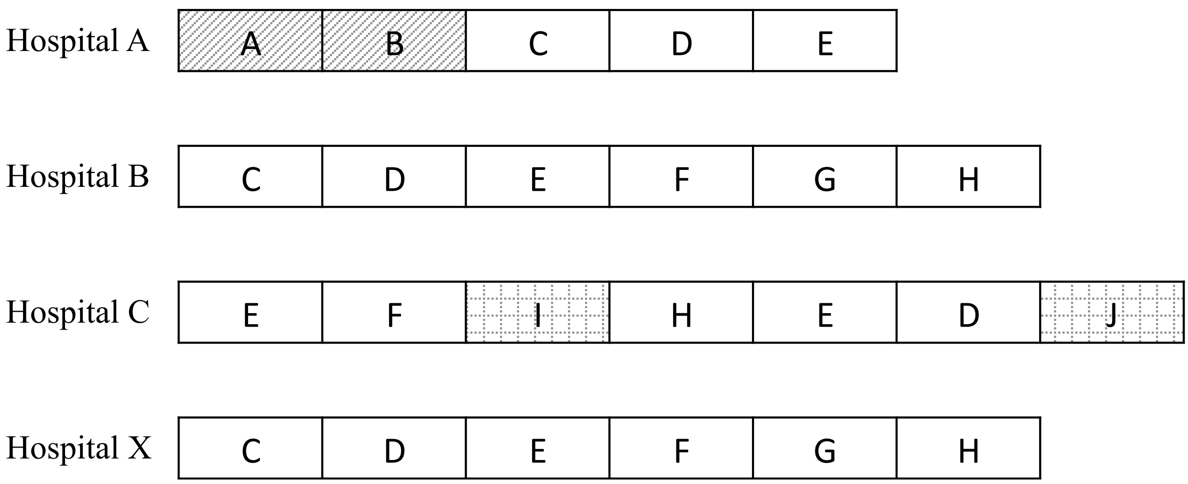


Figure. 3.4: Personal Health Records from Different Sources for Patient x

3.3.2 The Expected XML Format from Healthcare Providers

To accommodate the variation in native PHRs in the test phase of this research, we have created a rule on the XML file format and basic expectations from a healthcare organization. To achieve a successful import to the PHB database, the following is required:

(1) File format: XML

XML structure:

- Group items by category types.
- Each consecutive category must be nested to its parent.
- For more than one word category name, a category must have <type> item with the value of the category name.
- The items (Fields Naming) must be written in correctly spelled English.

(2) Minimum category types:

- Patient
- Hospital/Clinic
- Physician/Doctor
- Nursing/Checkup

These set of rules are translated into an XML schema file (XSD) used for XML validations, which also helps in the logic of reading the XML file content.
For future feasibility study of the PHB, we will make the XML rule less strict so that healthcare organizations will not have to make any adjustments.

3.4 Design of the Personal Health Book

This section introduces the overall design of the proposed PHB. The PHB is a software solution that stores PHRs on an independent database. It is portable to multiple devices and will run inside the device's operating system. The implementation is dependent on the devices: USB, mobile platform, or on the web server with a client browser application.

3.4.1 The Architecture of the Personal Health Book

The PHB application consists of eight components:

(1) Security Layer

(2) Application Component

(3) Data Import Component

(4) Data Dictionary

(5) PHB Database

(6) Data Export Component

(7) Notification Center

(8) Logger Component

The PHB internal architecture and the general data flow between the components are shown in Figure 3.5. The generic components (Security layer, Application, Notification and Logging) are based on typical APP implementations. The contributing components (Data Import, Data Dictionary, Database, Data Export) are part of this research and will be explained and tested in detail.

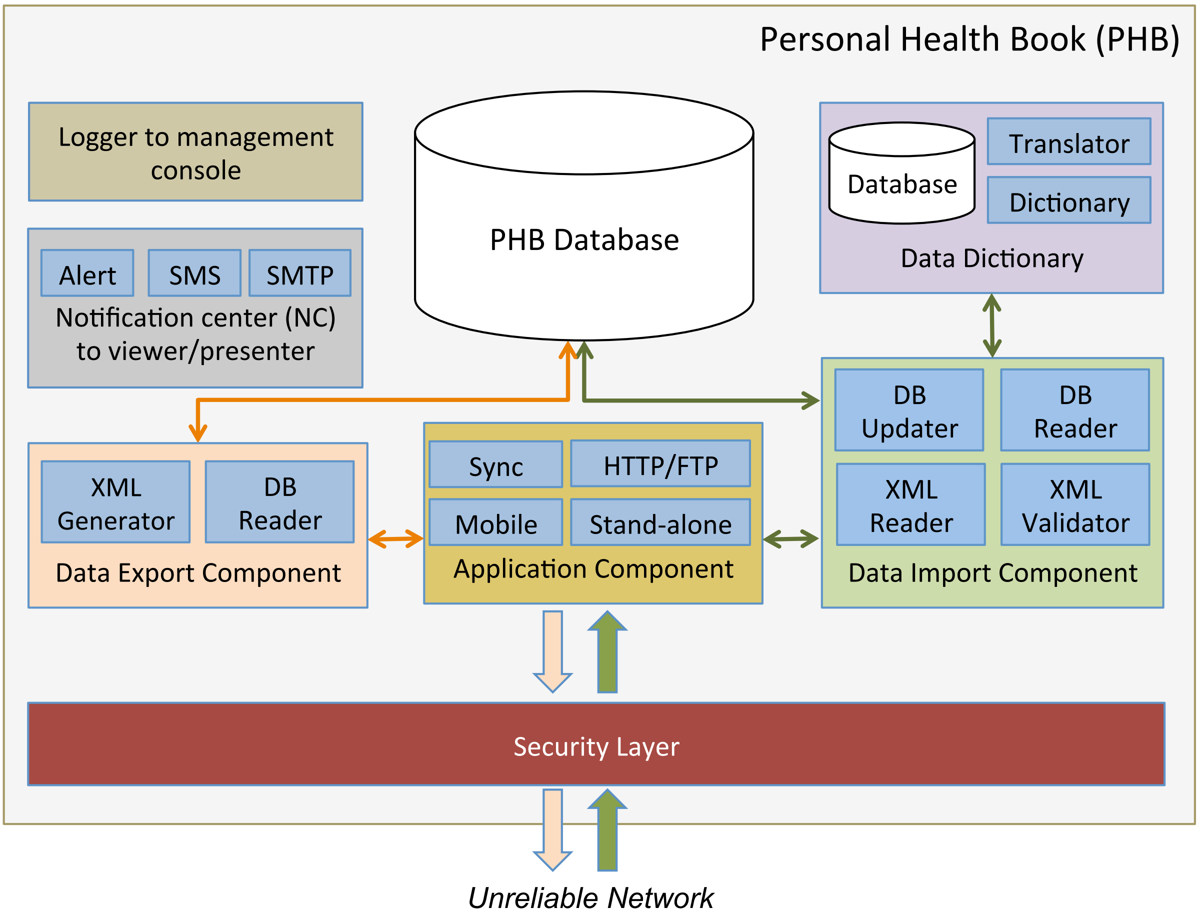


Figure. 3.5: Major Components of the Proposed Portable Health Book

3.4.2 Security Layer

The primary expectation for a PHB that stores an individual's personal data (PHRs and medical history) is privacy and data security. In the PHB application, this is managed by the security layer. As expected, the security layer will verify any attempt to connect to the PHB application for Access, Import or Export.

The PHB should be accessed by a web browser (HTTP), stand-alone application, mobile application or using an ftp service. Registered Users will need to login using their user name and password. Only authorized users can view shared data.

To improve privacy and data security, PHB data will be securely stored into the PHB database by encrypting all personal data [54]; and no access will be granted to anyone without securely logging in the system [55], by having a user name and password [56].

The Integrated PHB can get higher security by incorporating the geometric approach [57] or the token system [58] during login attempts.

3.4.3 Application Component

The operation of the Application Component is dependent on the type of the device that the PHB is ported onto and its messaging functionality is relevant to the device implementation model.

- If the PHB is ported on a USB, then the stand-alone model is followed with the PHB database being part of the application.
- If the PHB is ported on a mobile device, then the mobile APP model is followed and PHB database will use the device's (such as Android) own database that could be synced with the PHB database on the web.
- If the PHB is a client (web-based or stand-alone) and the PHB database is on the web, then cloud services will be used.

A synchronization sub-component will ensure that there is parity of files between the PHB database and the external backup.

3.4.4 Data Import Component

The Data Import component allows the import of PHR data from healthcare organizations. It is expected that the healthcare organization would push the patient's PHR from their system to the PHB.

When the Data Import Component receives the XML file in a predefined format, it scans and checks the file according to an acceptance logic based algorithm. During the checking procedure, the Data Import components interact with the PHB database. This will be described later.

In other words, to import the XML file from the healthcare organization source to the PHB database, there is an algorithm and the steps in pseudo code are as shown in the Algorithm 1.

The Data Import algorithm covers storing of the imported PHRs. To achieve that, there are sub-components.

- The XML validator verifies if the file in an XML is in the predefined format.
- The XML Reader opens the scans of the XML file.
- The DB reader checks if the XML categories (Hospital ID) and data items (Hospital Name, Address, telephone no.) exist in the PHB database.
- The DB updater creates new categories in the PHB database Categories table, new items in the Items table and new templates in the Templates table if they did not exist in the PHB database.

More information on the PHB database is give in a later chapter. The data flow and decision points are shown in Figure 3.6.

Algorithm 1 Personal Health Records Import Logic

```
1: Check imported file extension
2: if not an XML then
3:     go to Line 29
4: else
5:     Open XML file
6:     if not a valid XML then
7:                 ▷ Validate XML file structure against a predefined PHB schema (XSD)
8:         go to Line 29
9:     else
10:        Open the XML file and read the content
11:        Get all parent nodes
12:        for each parent node, check whether it exists in PHB ”Catagory” list
13:        if it does not exist then
14:            create a new category
15:        end if
16:        end for
17:        for each child under each parent node, check it exists in the ”Items” table in PHB
18:        if it does not exist then
19:            create a new item
20:        end if
21:        end for
22:        For the existing category, check if the imported child node matches the number and order under existing templates in PHB
23:        Create new templates for new categories and for the mismatch cases
24:        Insert new values into the ”Values” table in the PHB database according to templates ordering of the items
25:        Insert new records for new categories and link them to their values and templates
26:        Insert related medical records and link them to their sources, values and templates
27:    end if
28: end if
29: Close the XML file
```

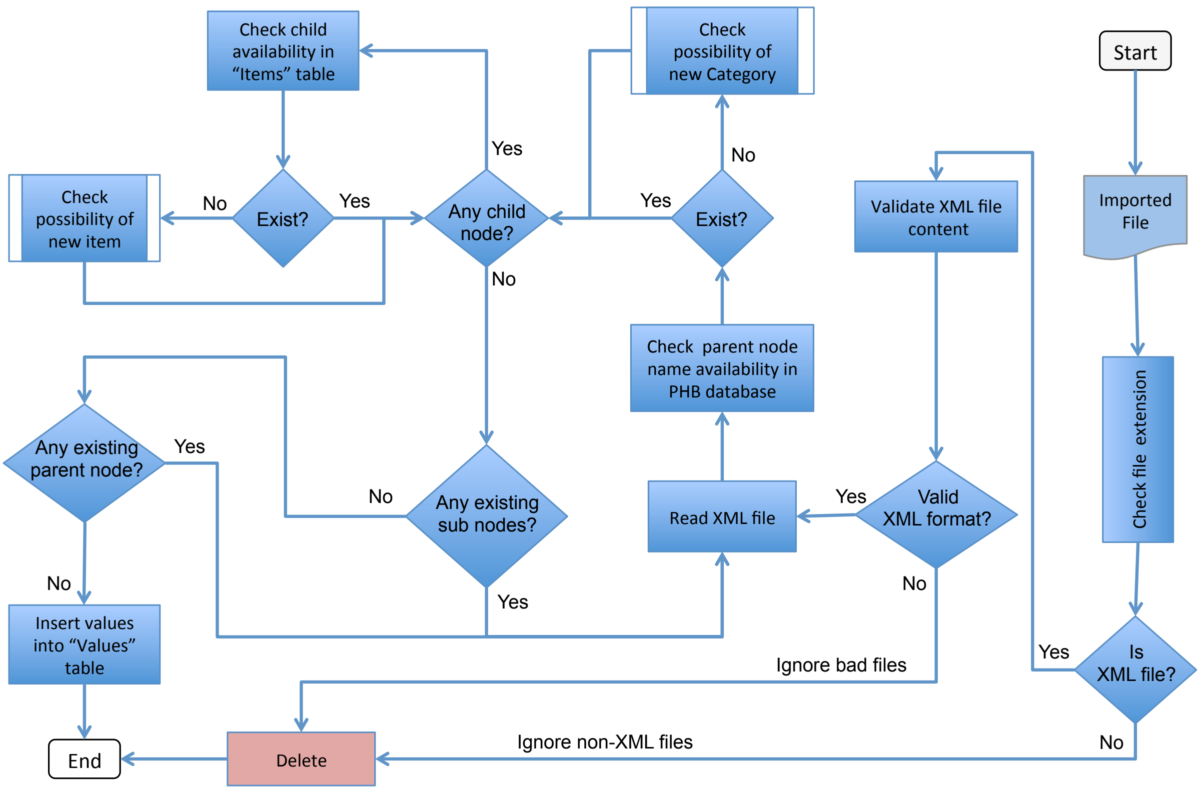


Figure. 3.6: The Integrated PHB Workflow for Gathering and Storing Personal Health Records.

3.4.5 Data Dictionary to Select Appropriate Terminologies

The healthcare profession uses a huge amount of terminology to describe an array of health conditions (such as infectious diseases, non-communicable diseases, laboratory tests, medicines, administration) and there is a need for an online data dictionary to satisfy the variability.

A data dictionary is a software utility triggered by a transaction process, queries an additional resource, and return with an answer (information). A data dictionary can also be used to help with language translation, which is important for low illiterate economic migrants who may use a PHB. Semantic Web tools: Resource Description Framework (RDF) [59], Linking Open Data (LOD) [60] and Web Ontology Language (OWL) [61] could be used as dictionary sources to understand the meaning of the imported data items.

In the PHB architecture, upon receipt of the XML PHR, the Data Import Component DB reader will send new data (categories, items) to the Data Dictionary Component (because it does not exist in the PHB database).

The ”Data Dictionary” component will check the meaning of the sent data by invoking the dictionary and/or translating the data (categories and items) even if it was not in English.

- If the data does not has a valid meaning, it will be compared with the content of the Frequent Used Terms (FUT) database to check for a match.
- If there was no match, the Notification Center will send a message/alert to the importer, requesting clarifications in the meaning.
- If the imported categories/items are English or has a match with the FUT data, the ”Data Dictionary” will inform the ”Data Import Component” with the results.

The ”DB updater” module of the ”Data Import” will store the imported categories/items into the PHB database. The data flow and decision points are shown in Figure 3.7.

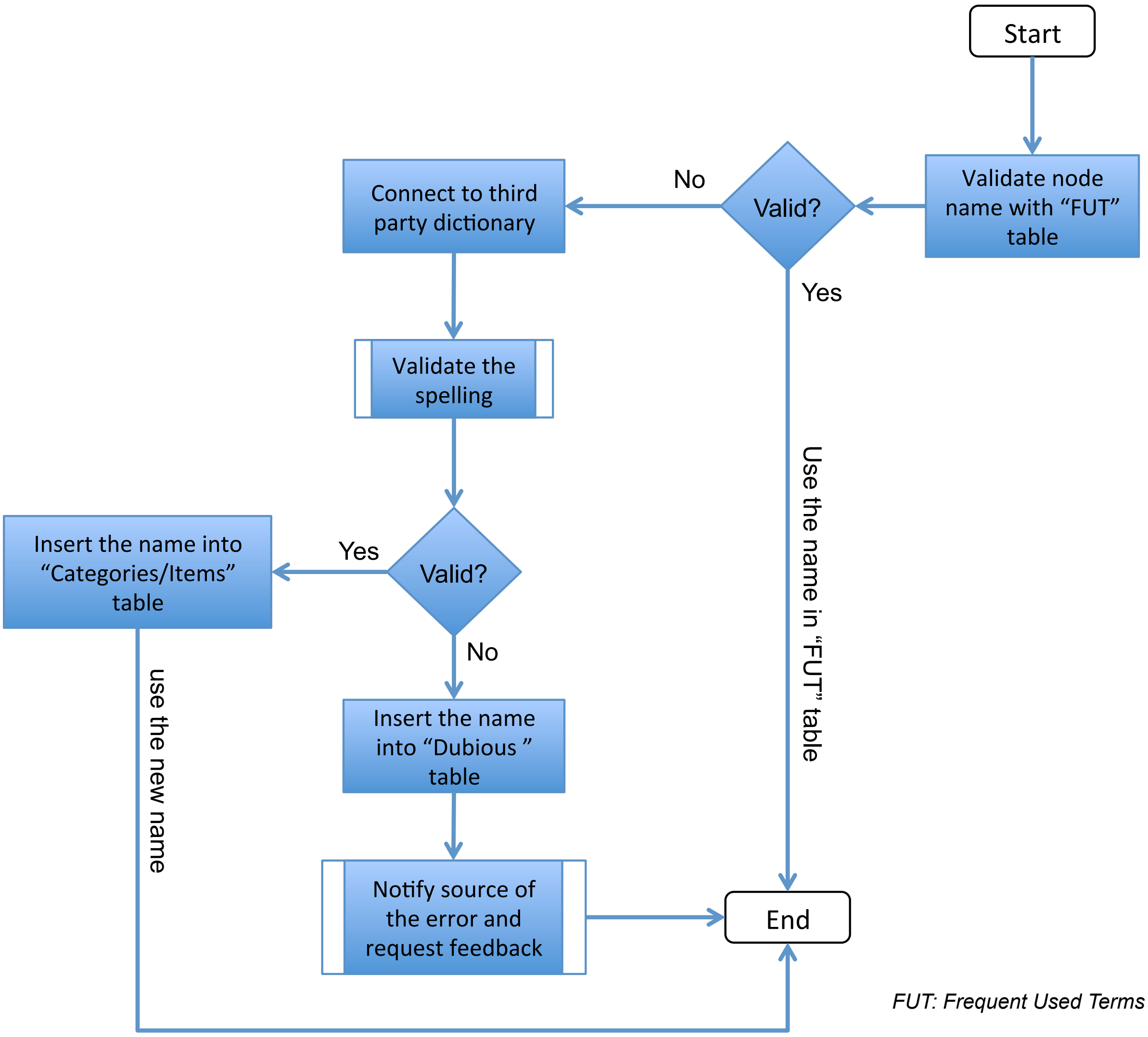


Figure. 3.7: Data Dictionary Workflow.

3.4.6 Personal Health Book Database

The PHB Database accommodates the different imported PHR formats. It is the main research contribution and the next chapter will fully cover its design and testing.

The role of the database is to store PHR data categories, data items and data values in the most optimal manner so it can be stored on very small devices (such as a smart card chip). One form of size optimization is to use the templates to reduce replicating categories and items. Templates are mentioned in the Data Import Component algorithm.

3.4.7 Data Export Component

The Data Export Component enables the individual (PHB owner) to view and share their medical history. It is expected that the PHB would push the PHR data or history to healthcare organization systems.

To export a PHR report from the PHB database to a destination (seeking) health clinic, the current concept will generate an XML file. To achieve a successful export that can be understood by the seeker, Algorithm 2 elaborates the steps that will occur.

The Data Export algorithm is used to generate a viewable and exportable medical information for stored PHRs.

To achieve this, there are sub-components:

- The ”Data Export Component” will react whenever a request for a PHR export or view is initiated.
- The ”DB Reader” module will query the ”PHB database”.
- The ”XML generator” will generate an XML file and will fill it with the queried data.

If the XML file generating was successful/unsuccessful, the Notification Center ”NC” will send an acknowledgement to the Export component requester.

Algorithm 2 Personal Health Records Export Algorithm

```
Get all root sources
Create a new blank XML file and open it to be filled
for each root source
    Get details of the root source by referring to templates
    Get all child sources of the root source
    for each child source
        Get details of child source by referring to templates
    end for
    Get all parent records
    for each parent record
        Get parent record details by referring to templates
        for each child of parent record
            Get details of child source by referring to templates
        end for
        Get all sub-records (and its details) of parent that their sources are sub-sources of the child source by referring to templates
    end for
    Get sub-records (and its details) that their sources that are not sub-source of the root source by referring to templates
    Fill all collected data into the XML file
end for
Close the XML file
Publish the generated XML file
```

More information on the PHB database is give in a later chapter. The data flow and decision points are shown in Figure 3.8.

3.4.8 Notification Center

The Notification Center enables the PHB software (USB, APP, Client) to inform the user (PHB owner) about the current internal process status and state.

- Local notifications are scheduled by the stand-alone APP and delivered on the same device.
- Remote notifications (push notifications) are sent by the web server to a remote service to push the notification to the device.

In the PHB software, there is potential for many notifications to be sent to the viewer.

(1) During Data Import (of an XML PHR):

  (a) If imported file was not valid, the PHR importing operation will stop, and Alert/Notify the importer with the result.

  (b) After a successful/unsuccessful import, the Notification Center ”NC” will send an acknowledgement to concern parties of importing results.

(2) During Data Dictionary process:

  (a) If there was no match, the Notification Center will send a message/alert to the importer, requesting clarifications in the meaning.

  (b) After a successful/unsuccessful import, the Notification center ”NC” will send an acknowledgement to concern parties of importing results.

(3) During Data Export (of XML report):

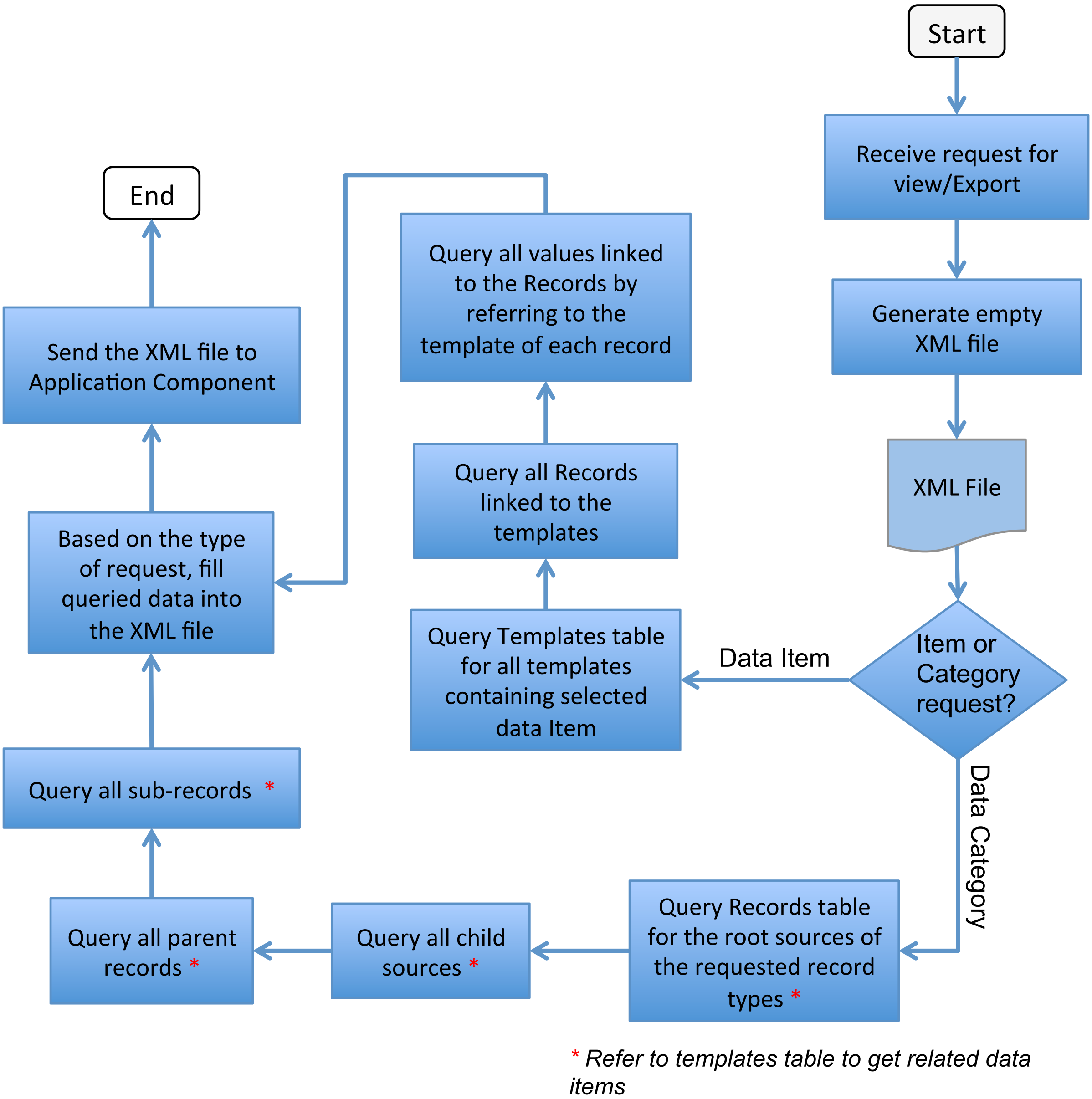


Figure. 3.8: Export Component Workflow

(a) If the XML file generated was successful/unsuccessful, the Notification Center ”NC” will send an acknowledgement to the export requester.

The Notification messages are not part of this research.

3.4.9 Logger Component

The Logger Component keeps track of the history of a process execution in the logs. There are various trouble-shooting use cases to monitor, store and review the internal PHB logs.
The logger functionality is not part of this research.

3.5 Summary

The proposed PHB software is an application that manages PHRs and is more suitable than the existing web-based PHB, as the overall dependencies to function are less. In addition, it can work with or without having an Internet access. It is more acceptable by people who do not want to place their PHRs on the web, which will not affect the functionality of the proposed PHB.

A simulation model is developed to test the performance of the PHB database. We will explain the details in the next chapter.

Chapter 4

Personal Health Book Database

4.1 Introduction

The Personal Health Book is designed for carrying a patient's health information as PHRs. The main criterion for the design is application size so it can run on any device including a smart card chip. We focus on optimizing and simplifying the logic so that the application can be implemented in budget-constrained environments. Different hospitals produce data in different data type (integer, char, date) or different format (size of each data, storing order of data) and data structure (names and number of columns). Therefore, it is necessary to design a database that can accommodate these variable natures of the source data. With all the variability in one storage, we had to think of answering these two questions:

(1) How the database should be designed in a way that can accommodate and adapt the variability of the imported PHRs?

(2) Could we store all imported PHRs into the PHB database without altering the meaning or loosing any part of it?

(3) Could we export viewable and understandable PHR that match the originally imported one?

4.2 Personal Health Book Database Design

4.2.1 Database Design Methodologies

We investigated which database design would best suit the PHB [62]. We considered three different database design methodologies: Wide Direction, Long Direction and Decomposed (see Figure 4.1).

In the Wide Direction database design, the items will be allocated in columns and in Long Direction design; the items will be allocated in rows. Wide direction will pre-allocate maximum number of columns to accommodate all the items. Long Direction design will allocate only required (minimum) number of columns. This design will have the flexibility to add new items.

We assume that the Wide Direction design will generate less number of rows and less repeated items. Hence, the database size will be smaller. However, because of its rigid design, this design will produce big number of null values and many of the imported data will not have a place and will be dropped. Therefore, a database that needs to accommodate new fields is not recommended for Wide Direction design.

The Long Direction design has advantages in not having any null values and because of its dynamic design; it can accommodate all imported data. However, it will generate big number of rows for each imported health record, and it will also allow a big number of data repetitions. Hence, the database size will grow very fast compared to the Wide Direction design.

The Decomposed Design is like a mixture of both the Wide Direction and the Long Direction to get the best of them. So it has advantages in not having any null values; it can accommodate all imported data. Even though the number of generated rows for each imported record and the number of data repetitions will be more than the Wide Direction design, they are less than Long Direction design.

Long Direction

Record	Item	Value
1	Date	10/10/2013
2	Hospital	Hospital x
3	Doctor	Doctor x

Multiple rows in one table for one imported record.

Wide Direction

Record	Date	Hospital	Doctor
1	10/10/2013	Hospital x	Doctor x

One row in one table for one imported record.

Decomposed

Record	Date	Hospital	Doctor
1	10/10/2013	1	1

Hospital	Item	Value
1	1	Hospital x

Doctor	Item	Value
1	1	Doctor x

Item	Value
1	Name

Items inserted in multiple linked tables for one imported record.

Figure. 4.1: Database Design Methodologies

We have listed all the assumed advantages and disadvantages of each database design (see Table 4.1), and we are planning to use the best method when designing the PHB database. As in the Wide Direction database design, there are expectations of data dropping (not getting stored into the PHB database), so we decided not to consider using it.

4.2.2 Personal Health Book Decomposed Database Design

The PHB database needs to manage files and data from multiple healthcare sources; it should adapt to any new data type and grow dynamically. It should work with future changes in the data without the need to change the database structure. That is the reason why we are considering the Decomposed Design for the PHB database [63], which is a mixture of both the Wide Direction and the Long Direction to get the best of both designs.

Entity Relationship Model

The entity relationship data model for an adaptive PHB database is shown in Figure 4.2. The ER diagram depicts the interconnections and relationships between entities: User, Record, Source, Category, Value and Item. For example, User and Record are connected via the relationship ”Owns” (one-to-many). In other words, the user will own many records. Similarly, entities Source and Record are connected via the relationship ”Imported from”. Here, a record will be imported from a source. In another scenario, Record and Value are connected via ”Has values”, which means that a record will have many values. Similarly, other connected entities have relationships in a meaningful way.

Personal Health Book Decomposed Database Structure

The PHB database consists of six tables (we used MS SQL to create the database concept, which works with most of database engines). The tables and their functions are described as follows:

Table. 4.1: Assumed Advantages and Disadvantages of Long, Wide and Decomposed Database Designs

Database Design	**Advantages**	**Disadvantages**
Long	• No blanks or ”Null” values • No data drop (can accommodate all received data)	• Bigger number of rows • Bigger number of data repetitions • Bigger size of database • Bigger memory waste • Bigger number of cells generated
Wide	• Small number of rows • Small database size • Small number of repetitions	• Big number of blanks or ”Null” values • Data dropping (can’t accommodate all received data)
Decomposed	• No blanks or ”Null” values • No data drop (can accommodate all received data) • Possibility of data types management	• Big number of rows • Big number of data repetitions • Big size of database • Big memory wasted • Big number of cells generated

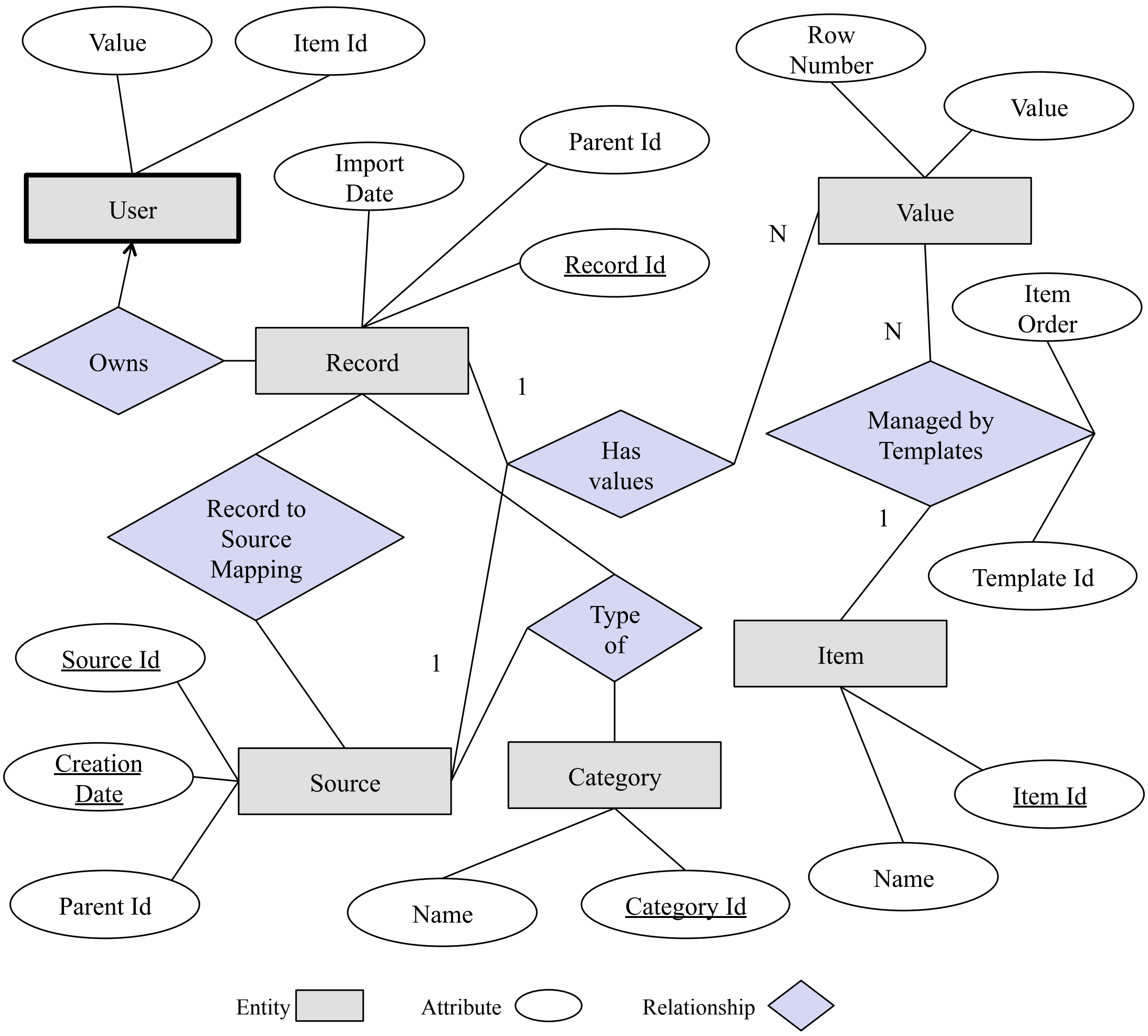


Figure. 4.2: Entity Relationship Model for the PHB Database

User Profile table: User personal details, such as name, date of birth, and address.

Items table: All data naming/labels are stored as an index (including unknown new items). An item can be name, age, temperature, height and description.

Categories table: This table is used to manage data types, such as the hospital, doctor, drug and checkup details.

Templates table: As PHB accepts all data coming from different sources with different structures, there is a need to keep track of each item in relation to each source.

Records table: All health records are stored in a Records table. For example, a record of one hospital visit has prescribed drugs and lab reports; these records have their own related data, different elements and more than one occurrence. They will be saved as sub-records in the records table. Even records source details will be kept in this table, such as hospitals, doctors, and drugs. A source can be a sub of another source (doctor is a sub of hospital). Each record stored in the database, will be related to a certain source, such as, a doctor can be a source of a record, or a drug is the source of the prescription. All sources (hospitals) to sub-sources (doctors) and records (checkup details) to sub-records (prescriptions) are linked through the "ParentId" column in the "Records" table.

Values table: All the values of the items that are specified in the templates table are stored here, which are actually the values of the sources and records details.

The database is decomposed and designed in such a way that there will never be a "Null" value or a blank stored in any table (Figure 4.3).

Querying the PHB database:

Every source and record will have a set of values linked together by the template table. To query the PHB database, the templates table will identify the items (naming/labeling) related to each value of the queried data type (source/record). Items related to the values are stored in the Items table, and every template is linked to a number of these items. If a hospital changes

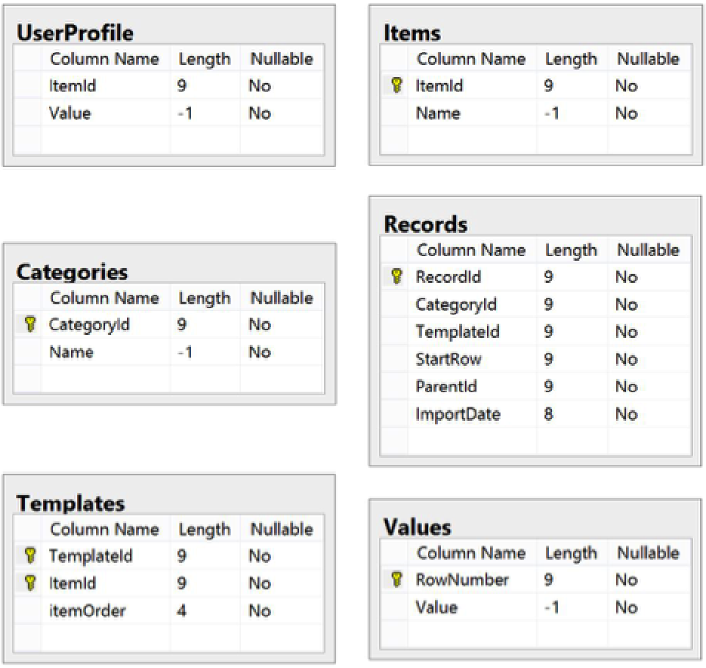


Figure. 4.3: The PHB Decomposed Database Design

the number of items, it will not affect the existing stored data because the date instance is kept. The example of a query would be to type in a keyword such as ’Blood Pressure’ between selected dates and the PHB would present the report. For a better and faster querying the PHB database we have scripted views; a view contains rows and columns, just like a real table. The fields in a view are fields from one or more real tables in the database. You can add SQL functions, WHERE, and JOIN statements to a view and present the data as if the data were coming from one single table [64].

Depending on the queried data and the depth of relevant data to be queried together, the complexity and length of the scripted view will be different, see Script 4.1 to see the simple view.

In Script 4.2, the scripted view is reading from four different tables using conditional INNER JOIN commands and viewing them in one table.

Script. 4.1: View querying records with their sources with their category name

```
SELECT SourceId, StartRow, EndRow, CategoryId, TemplateId, ParentId,
   CreationDate, Categories.Name AS SourceCategoryName FROM Categories
   INNER JOIN Sources ON Categories.CategoryId = Sources.CategoryId
```

Script. 4.2: View querying records with their source details and the category of each

```
SELECT r.RecordId, r.CategoryId, Categories.Name AS recordCategory, r.
   TemplateId, r.StartRow, r.EndRow, r.ParentId, r.ImportDate, s.SourceId
   , s.StartRow AS sourceStartRow, s.EndRow AS sourceEndRow, s.TemplateId
    AS sourceTemplateId, s.CategoryId AS sourceCategoryId,Categories_1.
   Name AS sourceCategoryName, s.ParentId AS sourceParentId, s.
   CreationDate AS sourceCreationDate FROM Sources AS s INNER JOIN
   RecordSourceMap ON s.SourceId= RecordSourceMap.SourceId INNER JOIN
   Records AS r ON RecordSourceMap.RecordId = r.RecordId INNER JOIN
   Categories ON r.CategoryId = Categories.CategoryId INNER JOIN
   Categories AS Categories_1 ON s.CategoryId = Categories_1.CategoryId
```

4.2.3 PHB Templates for Data Types Management

One of the main features in the PHB is the capability to accept all incoming data from multiple sources with different structures. This is possible with the help of the ”Categories” table an the ”Templates” that maps ”Items” and ”Values” tables with the ”Records” tables. Records structures will be stored in the ”Templates” table, any new data category (type) will be stored in the ”Categories” table, and any new data item (label) will be stored into the ”Items” table. The ”Templates” table will help to map all related data items together with their order of stored values in the ”Values” table (Figure 4.4).

Templates are independent components in the PHB database; it does not belong to any data category or record. The main goal of having templates is to keep related data items mapped

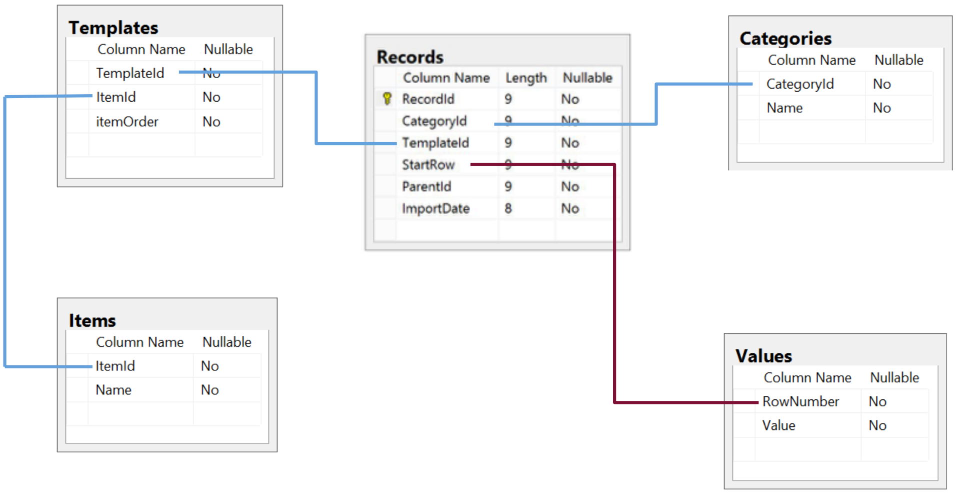


Figure. 4.4: Data Types Management in the PHB Decomposed Database

and in order. A template can be reused for records from different sources as long as the data items are the same. This key feature reduces memory consumption and gives a better meaning of the template.

4.3 Personal Health Book Database Simulation

4.3.1 Personal Health Book Database Simulation Setup Environment

In order to verify our assumptions, we have carried out a simulation with the setup listed in Table 4.2.

Table. 4.2: Simulation Environment

Tool	MS Excel
Technology	Excel Macros
Code	VB Script

The simulation was carried out for:

A. Database Design Methodology Comparison

We have assumed that Decomposed database design will be best for the PHB, so we needed to verify our assumptions by conducting a simulation to compare performances of the Decomposed database design and the Long database design approaches. Because of the rigidity of the Wide database design, we had to drop it out of our considerations for the PHB database design, as it does not serve our objective to have the PHB *”accepting all incoming data”*. We have considered the following evaluating factors in our comparison:

- Number of rows generated:

  This applies across all tables while importing EHR items into the database. For the very first import in the PHB database, Equation 4.1 will be invoked as R_1 is the first record in the PHB database, $R(g)$ is the number of rows inserted in the categories table, $R(r)$ is the number of rows inserted in the records table, $R(t)$ is the number of records inserted in the templates table, $R(m)$ is the number of rows inserted in the items table, and $R(v)$ is the number of rows inserted in the values table.

  $$R_1 = R(g) + R(r) + R(t) + R(m) + R(v) \tag{4.1}$$

  If it wasn't the first import in the PHB, but it was the first time for a healthcare organization to import, Equation 4.2 will be invoked as R_x represents the first time an import made by a healthcare organization and e is the existing common items.

  $$R_x = R(r) + R(t) + R(m - e) + R(v) \tag{4.2}$$

  If it was not the first import and not the first time for the healthcare organization to import, R_y will be invoked (see Equation 4.3, where i is the import sequence, h is the items related to the healthcare organization and c is the rows used for the data category of that healthcare organization.

$$R_y = \sum_{i=2}^{n} [R(r-h-c) + R(v)]_i \tag{4.3}$$

Equation 4.4, calculates the total generated rows R.

$$R = R_y + R_1 + R_x \quad \begin{cases} R_1 = 0 & \text{for all, except for the first record} \\ R_x = 0 & \text{for all, except for the new record structure} \end{cases} \tag{4.4}$$

- Number of repetitions:

  Number of non-value data repetitions could be items Id repeated or item name (label) repeated. For example, if we take ”Temperature” as an item, if the patient had checked his temperature several times, these values are not considered as repeated items even if they were similar as they differ in time stamp to build up health history progress. However, the name/label ”Temperature” or its Id will be considered repeated after it’s first time recurring.

  For the very first import in the PHB database, Equation 4.5 will be invoked as Rep_1 is recorded for the first in the PHB database, $Rep(g)$ is the number of repeats of the categories table, $Rep(r)$ is the number of repeats of the records table, $Rep(t)$ is the number of repeats of the templates table, $Rep(m)$ is the number of repeats of items table, and $Rep(v)$ is the number of columns of the values table.

$$Rep_1 = Rep(g) + Rep(r) + Rep(t) + Rep(m) + Rep(v) \tag{4.5}$$

  If it wasn’t the first import in the PHB, but it was the first time for a healthcare organization to import, Equation 4.6 will be invoked as Rep_x represents the first time an import is made by a healthcare organization.

$$Rep_x = Rep(r) + Rep(t) + Rep(m - e) + Rep(v) \tag{4.6}$$

If it was not the first import and not the first time for the healthcare organization to import, Rep_y will be invoked (see Equation 4.7).

$$Rep_y = \sum_{i=2}^{n} [Reps(r - h - c)]_i \tag{4.7}$$

Equation 4.8, calculates the total generated cells Rep.

$$Rep = Rep_y + Rep_1 + Rep_x \quad \begin{cases} Rep_1 = 0 & \text{for all, except for the first record} \\ Rep_x = 0 & \text{for all, except for the new record structure} \end{cases} \tag{4.8}$$

- Number of generated cells:

  That is the number of cells generated in all tables (number of rows multiplied by number of columns). For the very first import in the PHB database, Equation 4.9 will be invoked as C_1 is the first record in the PHB database, $C(g)$ is the number of columns of the categories table, $C(r)$ is the number of columns of the records table, $C(t)$ is the number of columns of the templates table, $C(m)$ is the number of columns of items table, and $C(v)$ is the number of columns of the values table.

$$C_1 = C(g) \times R(g) + C(r) \times R(r) + C(t) \times R(t) + C(m) \times R(m) + C(v) \times R(v) \tag{4.9}$$

If it wasn't the first import in the PHB, but it was the first time for a healthcare organization to import, Equation 4.10 will be invoked as C_x represents the first time an import is made by a healthcare organization.

$$C_x = C(r) \times R(r) + C(t) \times R(t) + C(m) \times R(m - e) + C(v) \times R(v) \tag{4.10}$$

If it was not the first import and not the first time for a healthcare organization to import, C_y will be invoked (see Equation 4.11).

$$C_y = \sum_{i=2}^{n} [C(r) \times R(r - h - c) + C(v) \times R(v)]_i \tag{4.11}$$

Equation 4.12, calculates the total generated cells C.

$$C = C_y + C_1 + C_x \quad \begin{cases} C_1 = 0 & \text{for all, except for the first record} \\ C_x = 0 & \text{for all, except for the new record structure} \end{cases} \tag{4.12}$$

- Database size:

  The database size is estimated by calculating the sum of total multiplications of the columns and the rows of same data types. For the very first import in the PHB database, Equation 4.13 will be invoked as S_1 is the size of the first record imported into the PHB database, $S(g)$ is the memory size consumed after inserting into the categories table, $S(r)$ is the memory size consumed after inserting into the records table, $S(t)$ is the memory size consumed after inserting into the templates table, $S(m)$ is the memory size consumed after inserting into the items table, and $S(v)$ is the memory size consumed after inserting into the values table.

$$S_1 = S(g) + S(r) + S(t) + S(m) + S(v) \tag{4.13}$$

If it wasn't the first import in the PHB, but it was the first time for a healthcare organization to import, Equation 4.14 will be invoked as S_x represents the size of the first import for a healthcare organization.

$$S_1 = S(r) + S(t) + S(m - e) + S(v) \tag{4.14}$$

If it was not the first import and not the first time for a healthcare organization to import, S_y will be invoked (see Equation 4.15).

$$S_y = \sum_{i=2}^{n} [S(r - h - c) + S(v)]_i \tag{4.15}$$

Equation 4.16, calculates the total memory size allocated S.

$$S = S_y + S_1 + S_x \quad \begin{cases} S_1 = 0 & \text{for all, except for the first record} \\ S_x = 0 & \text{for all, except for the new record structure} \end{cases} \tag{4.16}$$

- Size of wasted memory:
  The estimation of storage memory wasted caused by either the item name/label repetitions (Long direction DB) or by the item id repetitions (Decomposed DB). For all four cases mentioned earlier (first import in the system, first record for the healthcare organization, revisits and the total calculations) we use the same equation for calculating memory wasted W that would depend on the number of repetition multiplied by the size allocated for d_{size} decimal in the database (see Equation 4.17).

$$W = Rep \times d_{size} \tag{4.17}$$

B. The PHB database evaluation

For further evaluations of the decomposed PHB database design; as size is a key factor to have the PHB on a small local storage (such as a smart-card or a low memory USB), we have used the simulation to get these results:

- Database size growth compared to actual data imported.
- Database behavior toward similar imported data.
- Cumulative database size growth compared to cumulative size of imported data.

4.3.2 Simulation Algorithms

As soon as the simulation starts, it will execute the function $GenerateHospitals$ (see Algorithm 3) to generate data for hospitals according to the cases and input values for each case ($simCase$) with the number of hospitals ($numberOfHospitals$) and maximum number of unique items ($uniqueMax$) for each $simCase$.

The function $GenerateHospitals$ will first check the $simCase$ value to identify which case it will work with. Then it will loop from 1 to $numberOfHospitals$ to generate each hospital, where it will randomly set the number of unique items ($unique$) by executing the random operator (Rnd) and set it's range from 0 to $uniqueMax$. Then it will set the number of imported items ($TotalItems$) by adding $unique$ to minimum items for any hospital in the simulation ($fixedItems$) with a random generated number (starting from 1). After that it will calculate the number of items without it's repetition ($KeyItems$).

After generating all hospitals data for the current case, the simulation calculations will start through the procedure $simulateThisWork$. Then, the whole process will start over for the following case until all cases get simulated. Finally, return "Finished" at the end of the function $GenerateHospitals$.

Algorithm 3 Generate Hospitals for each Case

```
function GenerateHospitals(simCase)
    if simCase = 1 then
        Get numberOfHospitals
    else if simCase = 2 then
        Get numberOfHospitals
    else if simCase = 3 then
        Get numberOfHospitals
    end if
    for H = 1 To numberOfHospitals do
        HospitalNo = H
        unique = Int(((uniqueMax − 0 + 1) ∗ Rnd+0))
        TotalItems = fixedItems + unique + Int(((items − 1 + 1) ∗ Rnd + 1))
        HealthRecords = Totalitems − fixedItems
        KeyItems = HealthRecords + fixedItems − repeatedItems
    end for
    simulateThisWork(simCase)
    if simCase < 3 then
        GenerateHospitals(simCase + 1)
    else
        Return("Finished")
    end if
end function
```

Algorithm 4 shows the flow of the simulation calculations in the procedure *simulateThisWork*, as it will start by getting the number of hospitals (*numberOfHospitals*) and the number of imported PHRs (*numberOfRecords*) from the inputs form for the case *simCase*. Next, for each imported PHR, it will assign it to one of the hospitals in the *simCase* by a randomly selected hospital (*randomHospital*). After that, to decide how to simulate, it will check if there were any imported records earlier in the database. Then the simulation will randomly assign a doctor (*newDoc* - if it was a re-visit for the same doctor or the first visit). Next, it will check the value of the *NoOfImports*, as it will indicate if it is the first imported PHR from the current hospital or not; if it was for the first time, it will check if there was any available items by other hospitals (*availKey*) by executing the function *availableKeyItems*. Depending on the values of *newDoc*, $NoOfImports$ and *availKey*, the procedure will calculate the expected generated values of the number of rows ($rows$), the number of repeats ($repeats$), database size increment ($size$), wasted memory size ($waste$) and the number of generated cells ($cells$) for the Long (L) and Decomposed (D) database design methodologies.

4.3.3 Simulation Inputs

The simulation was carried out by considering three cases (see Table 4.3) with an input of the total number of PHRs (R), from (H) number of hospitals, and set maximum number of unique items (U). Unique items are the items that are introduced by a hospital uniquely, that is, no other hospital sends them.

Table. 4.3: PHB Test Environment

Cases	**Inputs**		
	R	**H**	**U**
Cases 1	10,000	12	10
Cases 2	10,000	2	0
Cases 3	10,000	5	5

4.3.4 Simulation Results and Discussions

1) Database design methodology comparison simulation results

After executing the simulation model, the simulation engine generates various PHR types with random attributes within the specified inputs, as shown in Tables 4.4, 4.5 and 4.6.

With the generated PHRs, the simulation script did the calculations for the comparison factors that we are going to use in our research. Table 4.7 lists the average calculations for the three cases for the Long database design, and Table 4.8 lists the average calculations for the three cases for the Decomposed database design; all cumulatively calculated.

Algorithm 4 Simulate the Changes in the Database by PHR Imports in Each Case

```
procedure SIMULATETHISWORK(simCase)
    if simCase = 1 then
        Get numberOfHospitals & numberOfRecords
    else if simCase = 2 then
        Get numberOfHospitals & numberOfRecords
    else if simCase = 3 then
        Get numberOfHospitals & numberOfRecords
    end if
    currnetRecord = 1
    while currnetRecord ≤ numberOfRecords do
        randomHospital = ((numberOfHospitals − 1 + 1) ∗ Rnd + 1)
        i = 1
        for H = 1 To numberOfHospitals do
            if NoOfImports <> 0 then
                i = 0
                Exit For
            end if
        end for
        newDoc = Int(((1 − 0 + 1) ∗ Rnd + 0))
        if NoOfImports = 0 then
            availKey = availableKeyItems(numberOfHospitals)
        else
            availKey = −1
        end if
        if availKey = −1 then
            Set the values of rows, repeats, cells, size, waste for L & D
            if newDoc = 0 then
                Adjust number of items by removing exiting doctor items
            end if
        else if availKey > 0 then
            if availKey < KeyItems then
                Set the values of rows, repeats, cells, size, waste for L & D
            else
                Set the values of rows, repeats, cells, size, waste for L & D
            end if
        end if
        fillResults(simCase, rows, repeats, cells, size, waste for L & D)
        NoOfImports = NoOfImports + 1
        currnetRecord = currnetRecord + 1
    end while
end procedure
```

Table. 4.4: Simulation Generated Contents: Case 1

No.	Total Items	Unique Items	Number of Imports
1	35	5	485
2	94	7	877
3	21	2	903
4	35	10	950
5	69	10	896
6	25	3	941
7	84	5	912
8	24	7	933
9	68	6	873
10	57	3	867
11	17	0	930
12	23	4	433

Table. 4.5: Simulation Generated Contents: Case 2

No.	Total Items	Unique Items	Number of Imports
1	14	0	4990
2	73	0	5010

Table. 4.6: Simulation Generated Contents: Case 3

No.	Total Items	Unique Items	Number of Imports
1	33	4	1228
2	16	5	2457
3	19	3	2520
4	17	2	2588
5	23	2	1207

Table. 4.7: Simulation Cumulative Calculations for ”Long Database Design”

	10	100	1000	10000
Number of rows	355	3264.67	31566.33	318535
Number of repeat	356.67	3430	33228.67	335931.33
Number of cells	1082.67	10065	97453.33	983470
Size	9034	84055.67	813947.33	8214160
Size of wasted storage	2853.33	27440	265829.33	2687450.667

Table. 4.8: Simulation Cumulative Calculations for ”Decomposed Database Design”

	10	100	1000	10000
Number of rows	477	3009.33	26711.67	267748.67
Number of repeat	305	1326.33	10048	98207.67
Number of cells	1156.33	7061.333333	62023.33333	620470.67
Size	10033	60718	531673.67	5316662.667
Size of wasted storage	2745	11937	90432	883869

For a better understanding of how much the Long Design behavior differs from the Decomposed Design, we have studied the average of the three cases of the first 100 PHR imports into graphical charts.

Number of rows In Figure 4.5, it is noticeable that the Decomposed database design started with more number of rows, then dropped down less than the Long database design, even though they are relatively close.

Database size In Figure 4.6, it is also noticeable that the Decomposed database design started with a bigger database size increment, then dropped down less than the Long database design, and there is a considerable gap between them.

Number of repeats In Figure 4.7, it is also noticeable that the Decomposed database design started with a big number of repeated items, then steeply dropped down and remained almost at a steady level, less than the Long database design, with vast gaps between them.

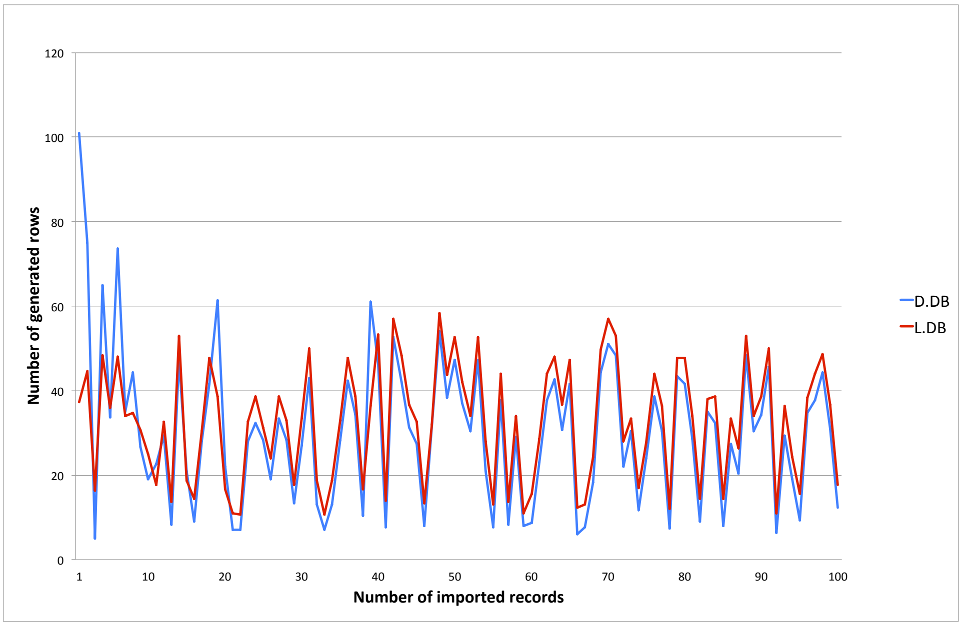


Figure. 4.5: Number of Rows vs. Number of imported records

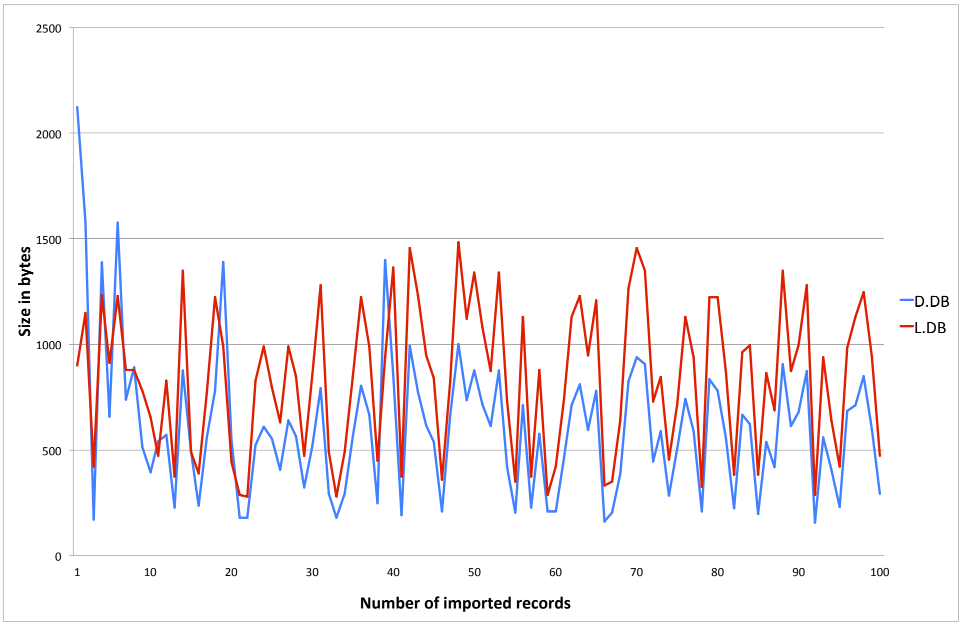


Figure. 4.6: Database size Increment vs. Number of imported records

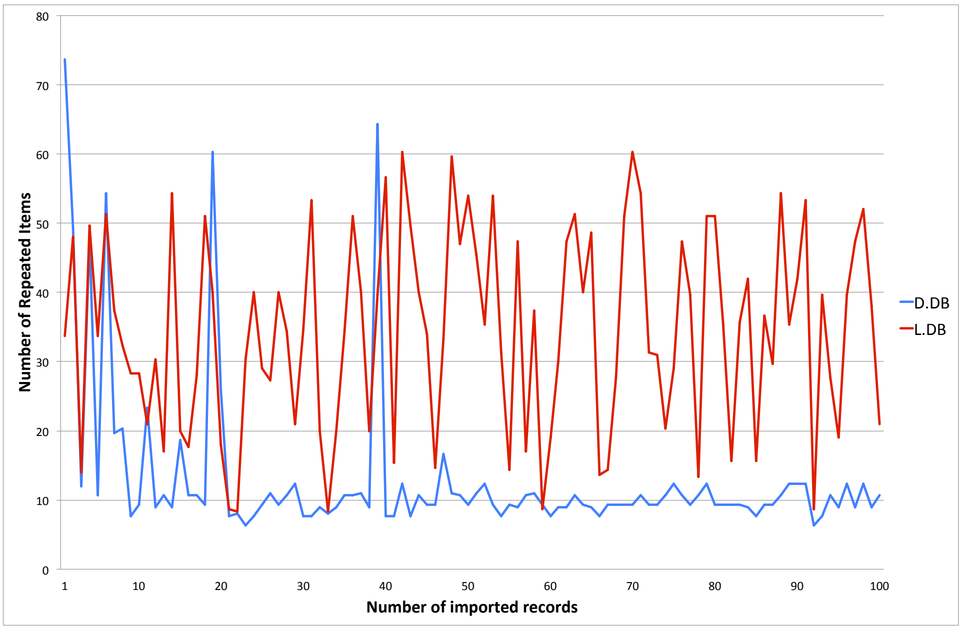


Figure. 4.7: Number of Repeated items vs. Number of imported records

Size of memory wasted Memory wasted (in bytes) is the memory size of the repeated items, as it shows in Figure 4.8. The chart has almost the same characteristics as the repeated chart; that Decomposed database design started with a big size of memory wasted, then steeply dropped down and remained almost at a steady level, less than the Long database design, and there are vast gaps between them.

Number of generated cells Theoretically, more tables with more columns and almost the same number of rows, the number of generated cells should be bigger. However, a good Decomposed database design should perform better. As it shows in Figure 4.9, the Decomposed database design started with big number of generated cells, then gradually dropped down and remained less than the Long database design, and there is a noticeable gap between them.

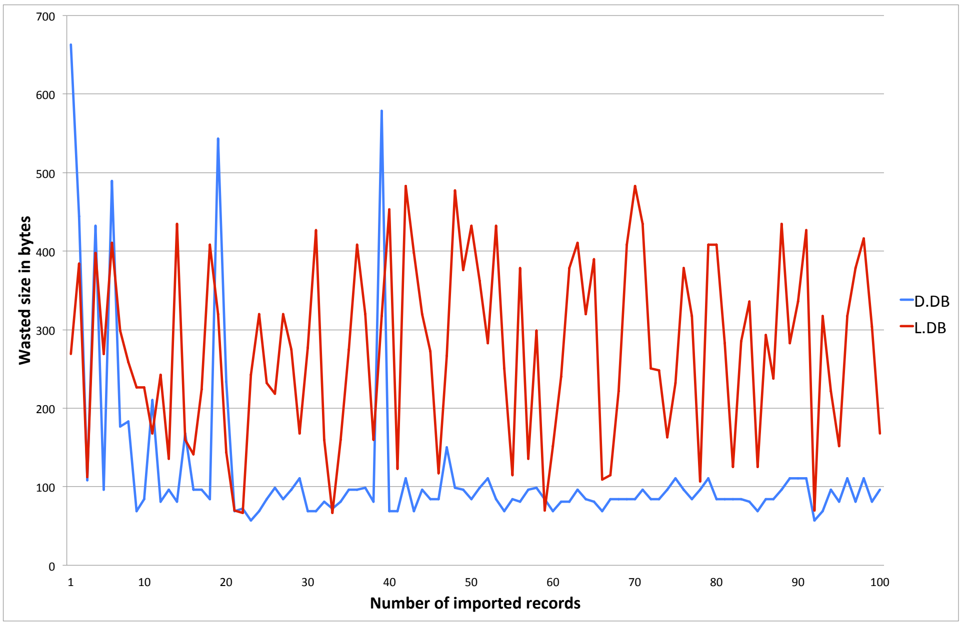


Figure. 4.8: Size of wasted storage vs. Number of imported records

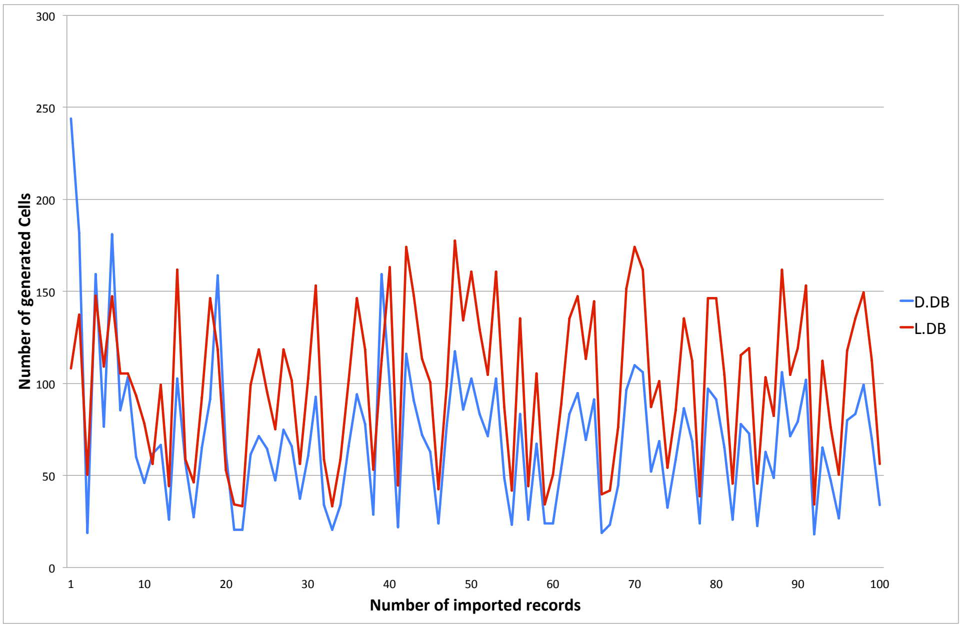


Figure. 4.9: Number of generated Cells vs. Number of imported records

In the previous figures, the Decomposed database design sometimes is elevating higher again, theses instances represent the very first time a PHR is imported from a hospital. That means if no more new PHR formats are getting imported, the ”decomposed” database design will have a better performance in the considered comparison factors. Thus, we have decided to use the Decomposed database design for the PHB.

2) The PHB database evaluation simulation results

As we are proposing to have the PHB on a portable local storage (USB or smart card), we had to evaluate our proposed database design to make sure it can get stored into such small sized environments (specially on the smart card). With the previous simulation we could estimate the average size the PHB could be (Application + Database), which is around 5 Mega Bytes for 10,000 imported PHRs.

We have carried out the simulation to evaluate how the PHB database will behave with imported PHRs, we have focused on how the management of data types will help when the items and will be stored in the database over time when:

A. One Patient’s PHRs imported by one hospital

Table 4.9 shows the imported ”key items” from one hospital for one patient, each visit was for a different purpose. Key items are distinct items in each PHR.

In Figure 4.10 we can see how many items (terms) will be stored into the PHB database for each visit compared to the originally imported items. Figure 4.11 shows the actual number of items stored cumulatively in the PHB database compared to what got imported, and how much it will save memory. In Figure 4.12 we wanted to show how many items will be stored into the PHB database for each visit compared to the originally imported items, where the patient would have the same checkups/measurements each time (same PHR structure). Figure 4.13, clearly shows the advantage of our Decomposed Design in saving storage by not saving un-necessary data into the database.

Table. 4.9: Simulation Cumulative Calculations for ”Decomposed Database Design”

Hospital Visits	Imported Items	Unique Items	Stored Items
Visit 1	25	5	25
Visit 2	82	7	62
Visit 3	11	2	2
Visit 4	23	10	10
Visit 5	57	10	10
Visit 6	13	3	3
Visit 7	74	5	5
Visit 8	14	7	7
Visit 9	58	6	6
Visit 10	45	3	3

B. Other cases

We carried out the simulation for:

- One patient’s PHRs imported by multiple hospitals.
- Multiple patients’ PHRs, considering the PHB is getting used on the web.

We got the same results as we got in Figures 4.10,4.11, 4.12 and Figure 4.13, whether patients visited the same hospital for the same checkups or they visited different hospitals for different variation of checkups. The result was almost identical.

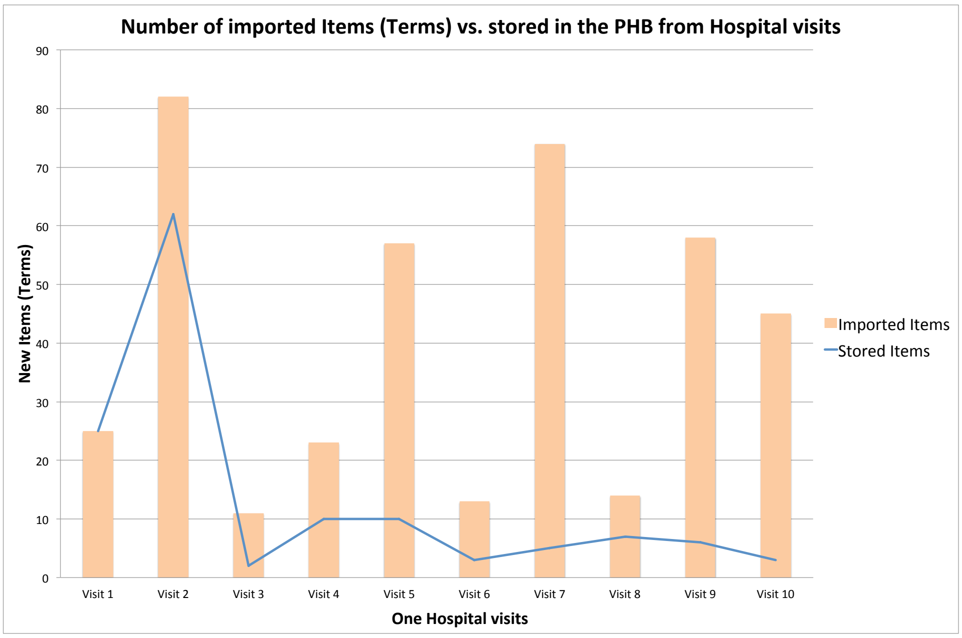


Figure. 4.10: Number of imported Items vs. stored in the PHB from Hospital visits for one Patient

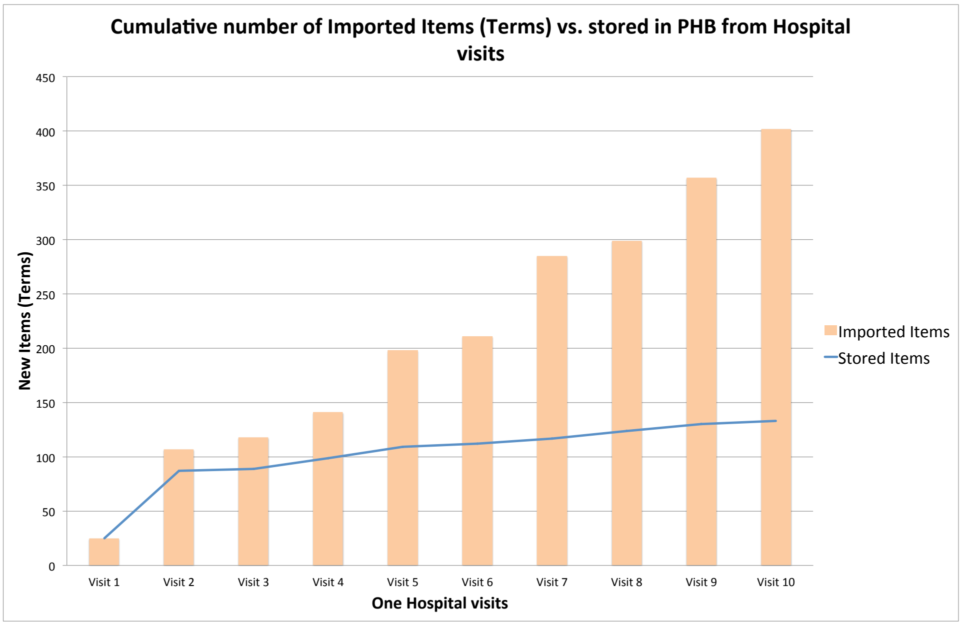


Figure. 4.11: Cumulative number of Imported Items (Terms) vs. stored in PHB from Hospital visits for one Patient

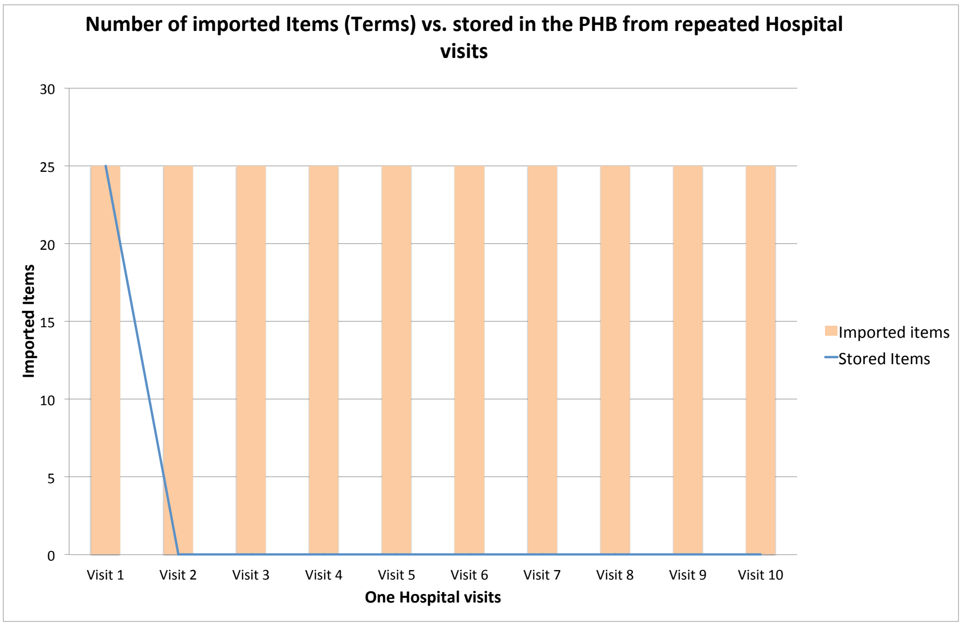


Figure. 4.12: Number of imported Items (Terms) vs. stored in the PHB from repeated Hospital visits for one Patient

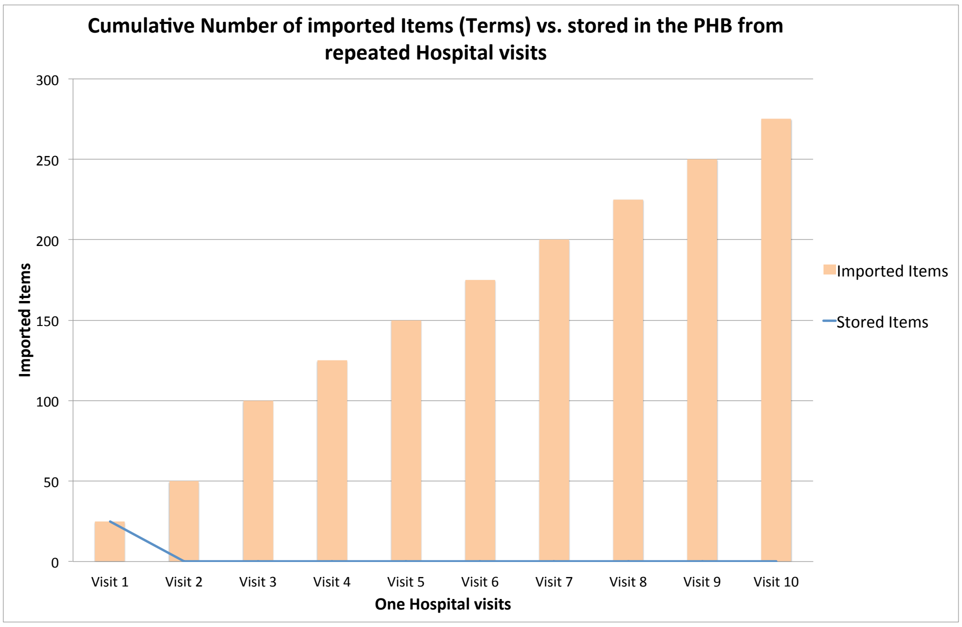


Figure. 4.13: Simulation output for Patient visits to one hospital for same repeated visits for one Patient

4.4 Personal Health Book Database Testing

4.4.1 Personal Health Book Database Test Setup Environment

We have implemented a test environment (web-based) of PHB, where an individual can upload own PHR in XML format (taking into consideration the minimum requirement for the formatting earlier explained). PHR samples were manually generated from the Portable Health Clinic (PHC) [65] test data and from a paper-based health report from a hospital in Japan.

As it shows in Table 4.10, test application developed using Visual Studio 2010 by coding asp.net pages with C# language, Microsoft Internet Information Services (MS IIS) version 7, Microsoft SQL server 2008 (MS SQL 2008). In the test, the output of the PHB an XML file generated after querying the PHB database, and viewed with the help of a pre-designed XML stylesheet (XSLT).

Table. 4.10: PHB Test Environment

Target	Test the PHB Database import and export
Tool	Visual Studio
Technology	Web application (asp.net)
Code	C#
Web server	MS IIS 7
Database	MS SQL 2008
PHR format	XML
XML stylesheet	XSLT

4.4.2 PHB Database Test Algorithms

Before importing any PHR to the PHB; the system will check the file extension, if it was not and XML file, it will not get imported. However, if it was an XML file, it will accept the file to be imported, then the XML content validation will start. As it is shown in Algorithm 5, the imported XML file will get validated before processing it through the function *xmlValidate*, where it will check the XML file integrity and compare it to one of the predefined XML schema (XSD). If it was not a valid XML file, the system will delete it.

Algorithm 5 Validating Imported XML File

```
function xmlValidate(XMLfile)
    XmlReader ehrFile = XmlReader.Create(XMLfile, setXSD(”002.xsd”));
    while ehrFile.Read() do
        { }
    end while
    if errorMessage = ”1” then
        return ”Valid XML file”;
    else
        System.IO.Files.Delete(XMLfile);
        return ”Not valid XML file”;
    end if
end function

procedure ehrSettingsValidationEventHandler(object sender, ValidationEventArgs e)
    if e.Severity = XmlSeverityType.Error then
        errorMessage = errorMessage + e.Message;
    else
        errorMessage = ”1”;
    end if
end procedure
```

Algorithm 6 shows the steps after validating the imported XML file. As the XML file reading process will start to collect all the data categories, data items and the data values. After that, the collected data will be compared to existing data into the PHB database and will store only new valid ones. The non-understandable data will need feedback from the importer to clarify the meaning. Then, it will check if there were matching templates, if not, it will create a new one. Finally, all data will be stored into the PHB database and will be linked according to how they were originally imported.

Algorithm 6 Reading and storing Imported XML File

```
function xmlProcess(XMLfile)
    Reading XML file
    System.IO.StreamReader file = new System.IO.StreamReader(XMLfile);
    while (line = file.ReadLine()) ! = null do
        parentsId.Add(Parent);
        groupsId.Add(Group);
        type.Add(line);
        itemsGroupId.Add(Group);
        itemsName.Add(line);
        itemsValue.Add(line);
    end while
    file.Close();
    // Start storing collected data into Database
    if parentsId.Count = groupsId.Count & parentsId.Count = type.Count then
        for i = 0 To parentsId.Count - 1 do
            // Check if the Category/item availability in the database
            if categoryItemCheckAvailablityByName(type[i].Trim()) = false then
                // Category/item doesn't exist, need to Insert new imported into database
            else
                // Category/item is available in the Database, no need to store imported one
            end if
        end for
    end if
    //Compare existing templates details with imported content
    for i = 0 To groupsId.Count - 1 do
        if itemsName[j]= dsTemplateWithItemNames.[n]["Item Name"] then
            //new template needed
        else
            //No need to create new template
        end if
    end for
    //Start inserting new source/records values
    if newSourceFlag = 1 then
        //Insert New source values
    end if
    for i = 0 To groupsId.Count - 1 do
        for j = 0 To itemsGroupId.Count - 1 do
            //Insert Records values
        end for
    end for
end function
```

4.4.3 PHB Database Test Results and Discussions

To test the PHB framework we randomly selected a real patient's PHRs (with personal details deleted) from the Grameen Portable Health Clinic database in Bangladesh (22,000 entries). We accessed 2 PHR records from a Japanese hospital "Hospital xyz" issued at different times. We imported 5 PHRs into the PHB environment. The PHB role was to reorganize the data from two health clinic visits by categorizing and grouping and presenting a unified, human understandable, view structure.

The Portable Health Clinic normally generates PDF files to print PHRs. For the test, we created an XML file to match the import requirements (see Figure 4.14(a)). The data was successfully imported in the PHB database.

With the help of the pre-designed XSL file to view the generated XML file in a way that it will generate multiple tables. The viewed PHB patient report is human readable, understandable and meaningful as seen in Figure 4.14(b). The reports of multiple visits from two healthcare organizations are shown in a descending order. Every category/type in the report is clearly separated from the others and contains the exact number of items that the original record contained. In addition, all childs components are displayed under their parent source.

4.5 Summary

In this chapter we explained in detail the structure of the Decomposed Design of the PHB database, and discussed how we reached the current design after the simulation we carried out to compare Long and Decomposed database design methodologies. After that, we carried out a simulation to get more results showing us how the Decomposed PHB database will behave when imported PHRs come from one or many hospitals, same or different type of PHRs for one or many patients. The algorithms of how the simulation was carried out were explained.

After that, we explained the experimental PHB that we implemented to test the import,

```xml
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<content>
  <Source>
    <Type>Hospital</Type>
    <Column1>Name</Column1>
    <Column2>Portable Health Clinic</Column2>
    <SubSource>
      <Type>Doctor</Type>
      <Column1>Name</Column1>
      <Column2>Dr. Ahmed</Column2>
      <Column1>Address</Column1>
      <Column2>Dhaka, Bangladesh</Column2>
      <Checkup_Details>
        <Type>Checkup Details</Type>
        <Column1>Checkup date</Column1>
        <Column2>01/12/2013 11:04 AM</Column2>
        <Column1>height</Column1>
        <Column2>153</Column2>
        <Column1>weight</Column1>
        <Column2>53.5</Column2>
        <Column1>bmi</Column1>
        <Column2>22.85</Column2>
        <Column1>waist</Column1>
        <Column2>89.3</Column2>
        <Column1>hip</Column1>
        <Column2>91.6</Column2>
        <Column1>waist_hip_ratio</Column1>
        <Column2>0.97</Column2>
        <Column1>temperature</Column1>
        <Column2>98.67</Column2>
        <Column1>oxygen_of_blood</Column1>
        <Column2>98.6</Column2>
        <Column1>bp_sys</Column1>
        <Column2>171</Column2>
        <Column1>bp_dia</Column1>
        <Column2>102</Column2>
        <Column1>blood_glucose</Column1>
        <Column2>76</Column2>
        <Column1>blood_glucose_type</Column1>
        <Column2>PBS</Column2>
        <Column1>blood_hemoglobin</Column1>
        <Column2>9.1</Column2>
        <Column1>urinary glucose</Column1>
        <Column2>-</Column2>
        <Column1>urinary protein</Column1>
        <Column2>++</Column2>
        <Column1>urinary urobilinogen</Column1>
        <Column2>+-</Column2>
        <Column1>pulse rate</Column1>
        <Column2>76</Column2>
        <Column1>arrhythmia</Column1>
        <Column2>Normal</Column2>
        <Column1>prescription</Column1>
```


(a) Imported PHR from PHC database, Bangladesh

Hospital	Last Visit
Name	Hospital xyz
Doctor	
Name	Dr. Yoko Naka
Checkup Details	
Checkup date	24/01/2015 10:00 AM
Metabolic	No
Medical History	No
Advice on Treatment	No
Blood Pressure	84/52
	⋮
Hospital	Older Visit
Name	Portable Health Clinic
Doctor	
Name	Dr. Ahmad
Address	Dhaka, Bangladesh
Checkup Details	
Checkup date	08/08/2014 10:00 AM
weight	53.5
BMI	22.85
waist	89.3
hip	91.6
waist_hip_ratio	0.97
	⋮
Hospital	Older Visit
Name	Hospital xyz
Doctor	
Name	Dr. sanji Jiro
Checkup Details	
Checkup date	21/06/2014 10:30 AM
Chest	84.0
BMI	23.1
Fat level	+ 5.0 %
Blood Pressure	102/60
	⋮
Doctor	Oldest Visit
Name	Dr. Yoko Naka
Checkup Details	
Checkup date	30/12/2013 11:04 AM
Blood Pressure	120/72
Body Measurements	
Height	167.4

(b) PHB Personal History Report

store and export algorithms and database capabilities. We explained the algorithms we used in our experiment and then discussed the methodology and the results by comparing the output with the imported PHRs.

In the next chapter, we will explain the applicability of the PHB and showcase some use cases for the PHB in the developed and developing countries and how to have a meaningful use of the PHB.

Chapter 5

Personal Health Book Use Cases

The portability of PHRs is impeded by many barriers that result in a disconnected personal health data landscape. The main barriers which affect the adoption of the Personal Health Book Application as a container of PHRs by individuals are:

- Awareness of PHRs.
- Usability of PHRs.
- Access to PHRs.
- Meaningful Use of PHRs.
- Clinical Integration of PHRs.

This chapter will discuss how the Personal Health Book can bridge the barriers and will present viable use-cases.

5.1 Personal Health Book and the awareness of PHRs

For the individual, the most important PHB adoption feature is the awareness of the portability of PHRs and their value. Most health consumers are not aware of their right as individuals to have access to their own personal medical data. Alongside PHR awareness is the health consumer must be computer literate and can make effective use of a multi-choice application. The PHB is a software application and can be ported on offline devices such as smart card, USB, intermediate Smart Phone and tablet applications, online note books and PCs.

Awareness Use-case: PHB application interface

This use case indicates how a health consumer can use the PHB when ported onto a USB card. Whenever the patient visits a hospital, the following steps occur:

(1) The pateint gives the USB smart card to the doctor or health-worker to plug it in their personal computer.

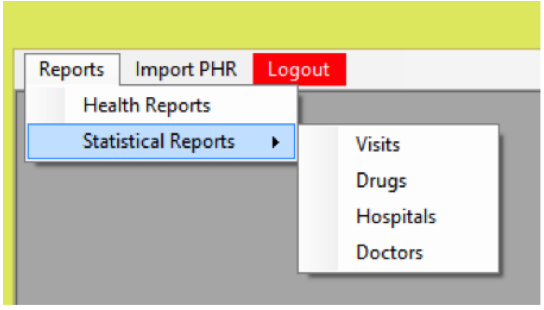
(2) The doctor/health-worker runs the PHB application, and a login screen will appear.

(3) The patient has to insert his username and password as shown in Figure.5.1(c).

(4) After a successful login, the PHB's main window will appear, and the user can navigate through the application via the main menu (see Figure.5.1(d)). The menue has three main items:

- Reports: Where the patient can view and share his health related reports (Health Reports) or understand his data statiscally (Statistical Reports).
  Because the PHB database is designed to get the use of each stored entity, it can produce health related reports as well as statistical reports such as number of visits, drugs, hospitals and doctors. These types of reports can help in understanding the extent a person is following up on his health and the address and contacts of all visited hospitals and doctors.
- Import PHR: Where doctor/healthcare worker can import new PHRs

(c) Login screen



(d) The PHB Application Main Menu

Figure. 5.1: Personal Health Book Stand-alone Application User Interface

- Logout: It is important that the patient logs out after he finishes from the PHB application to keep his data securely safe.

(5) If (Health Reports) was selected, health reports form will appear (see Figure.5.2), here the patient can view his health related data.

(6) The (Generate Report) button will produce the health report based on the patient's customisation and filtering.

(7) By clicking on the (Export) button, the patient can share a copy of the produced report with the doctor/health-worker, either in an XML or PDF format. As a secuerity and privacy protection measure, an authenticaion form will appear and the patient will have to enter his credintials to allow the export process.

(8) After the patient finishes the health checkup, the doctor/health-worker will be able to import the new PHR to the PHB application by selecting (Import PHR) from the main menu.

(9) To quit the application, the patient simply clicks on (Logout).

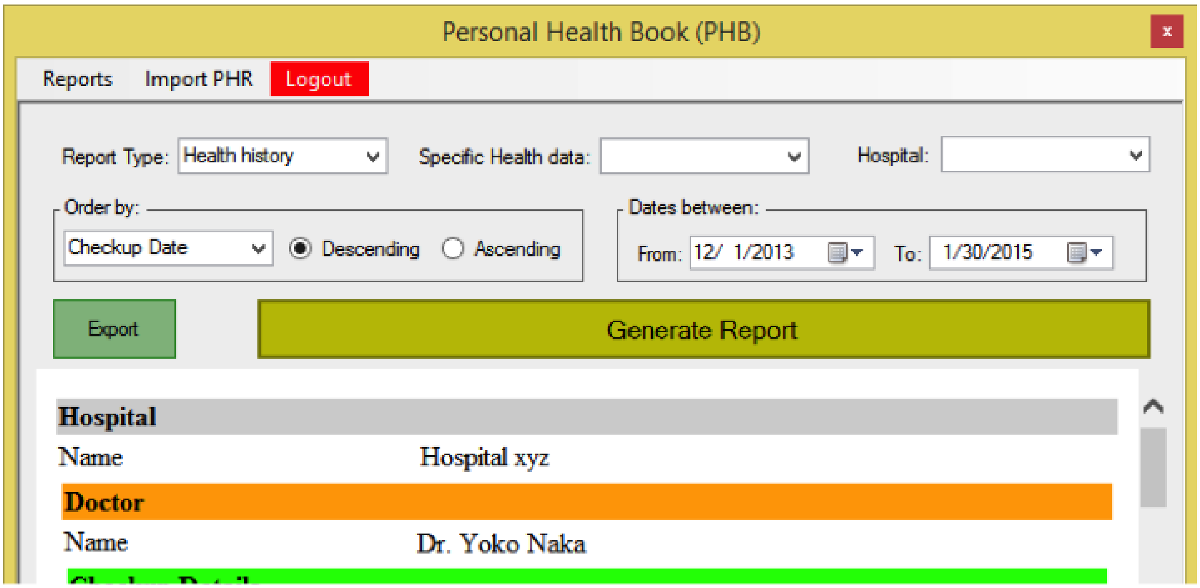


Figure. 5.2: Health Reports Form

5.2 Personal Health Book and Usability of PHRs

The most important PHB adoption feature as a viable solution is usability that the application enables true portability. All health consumers are migratory, they move from one health institution to another during their lifetime. In the developing world, economic migration is emphasized with millions of illiterate people frequently moving from one region or country to another. This section, discusses how the PHB is relevant to the usability issue.

5.3 Personal Health Book in Developing Countries

An important test for the PHB, if it can be usable to developing countries. A digital Portable Health Book solution in developing countries requires affordability (low cost) and simplicity (ease of use). Following these drivers are robustness, privacy, and security. For this arena, it is suggested that a USB or smart card [66] is a suitable and affordable solution for the low-income person. The ”ease of use” depends on the literacy level of the consumer and willingness of the healthcare provider to share the PHR. We envisage that the consumer will ask the healthcare provider to store the resultant PHR or EHR on their USB or Smart Card. This is applicable to rural community health clinics or visiting health providers such as the Grameen PHC in Bangladesh [65]. At the rural community, health clinics store computer records and give patients printed prescriptions. It is feasible to ask for a digital copy. Once imported in a USB or smart card, the consumer will store the PHR and then can view the PHB data using a PC or tablet.

As low literate consumers in developing countries often migrate from rural villages to urban centers, the consumer should be able to ask another healthcare administration or doctor in a hospital to import their latest PHR or health history from the USB or smart card they are carrying. This will give the health clinic an improved perspective on the consumer's medical history. Following the consultancy, the patient can ask for the results to be imported to the USB or smart card. The PHRs will increment and the consumer will ask the next health clinic to view the appended PHRs and so on.

To provide robustness to the Personal Health Book application, if requires extending the storage options to the low literate consumer. The ability to save the PHR to an external storage such as a cloud environment will overcome the risk of physically losing the USB/Smart Card or file corruption. In this instance, the consumer will ask the healthcare provider or a family member to export the PHR to a secure website. The consumer will need a username and a password to access their unique remote PHB database.

A web-based PHB service has the benefits that the low literate consumer can use a smart phone PHB application to manage and store the PHR both on the smart phones and on the Internet site. Once authorized by the consumer (health information owner), other interested parties such as family, friends and ultimately healthcare professionals, such as doctors, can view and import the PHRs.

Use-case: Low-literate Individual from a Developing Country

This use-case explains how the PHB could benefit millions of migrants in developing countries, as shown in Figure 5.3. Seddiq is a low literate middle-aged male consumer with hypertension and is examined at the rural clinic in Bangladesh. He would like to import the PHR. He has a choice of USB Smart Card with a PHB application (5 MB); a smart phone with PHB App but he needs a transmission method for importing the file; or asking the healthcare provider to export the PHR to a website (he has to type in the username and password). Seddiq, later migrates to an urban city, overeats and feels unwell. He visits the hospital and allows the administration to access and import his PHB. He chooses the website method and shares his username and password. After the consultation, he asks the administration to export a copy of the PHR to the web-based PHB. Seddiq's family rely on his earnings, they are concerned about his health status, and occasionally view a summary of his PHRs. During the festival season, Seddiq returns to the rural village and visits the health clinic for a check-up. Due to Internet outage, the community health clinic import his health records from his USB Smart Card. Using this approach, Seddiq, the people that depend on him, and health providers, have a convenient method to manage the PHR and make important decisions.

5.4 Personal Health Book and Access to PHRs

Ease of access to PHRs is key for the PHB usage. The simplest and most robust method is for health consumers to connect offline devices to the Internet and sync with a cloud service. Access simplicity and ease-of-use will benefit many user groups such as people in a hurry and elderly people.

Access to PHB is also important to the healthcare provider. Most healthcare organizations see the PHR as a summation of a check-up or examination, they are not willing to share a digital PHR, but they will give a paper copy of the test results and a prescription. Healthcare

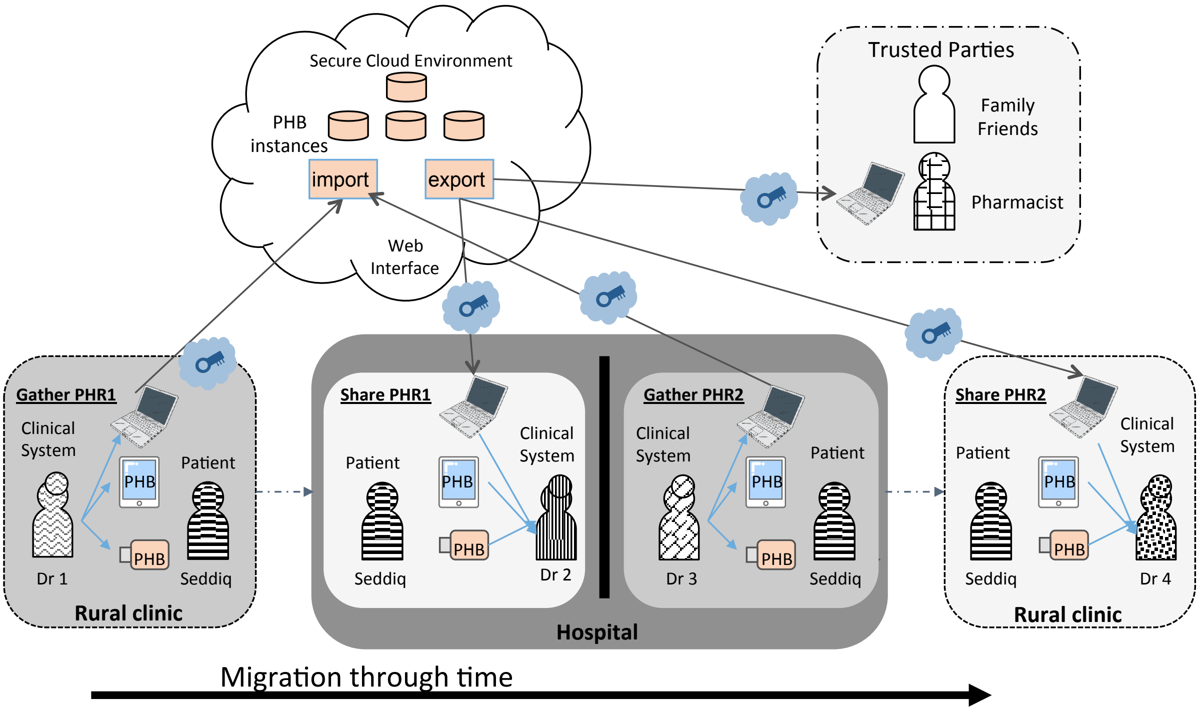


Figure. 5.3: The Portable Health Book Use-case in Developing Countries

providers rarely use previous PHRs as an import to their decision making, although they do check the data of referrals from other healthcare organizations. Taking this important fact into consideration, with the PHB we envisage that healthcare organizations will benefit when the patient asks the care administration to import the most recent PHRs from their PHB account on the web or locally in the USB or smart card or smart phone. Adhering to the patient's request, the healthcare organization will learn about the patient's medical history (allergies, past diseases, surgeries, complications) that can influence the hospital triage and early tests and avoid any medical errors. Should the healthcare organization accept the concept of PHB and adjust their IT EHR systems, patient centric data will increase their efficiency in terms of patient turn-around time, reducing errors.

There are many use cases that would demonstrate how the care administration would benefit from reliable digital PHRs. From citizens moving from region to region and clinic to hospital to individuals moving from one country to the next.

The end point of PHR access is when the patient asks for the results to be imported to his/her PHB. Care administration and physicians may be reluctant as this may place them at litigation risks if they made incorrect diagnosis or prescribed the wrong medicine. However, these would be less frequent cases and on the whole both the healthcare organization and the patient will benefit from a transparent record of the health event.

Technical access to the PHB provide robustness and requires extending the storage options so it can be stored on small and thin media (as thin as a business card or smaller). The ability to promptly sync the PHR to an external storage such as a web-based environment will overcome the risk of physically losing the USB/smart card/smart phone or file corruption.

Use-case: Person Living in Disaster Affected Area in a Developed Country

This use-case illustrates how simple access to an individual's PHRs will benefit public healthcare and health professionals in an disaster scenario, and is shown in Figure 5.4. Seddiq is an over-weight middle-aged male, who has immigrated to Japan and cannot speak Japanese fluently. He suffers from hypertension and respiratory diseases and is regularly examined at the city hospital. He has asked the health clinics to push his PHRs onto a PHB application that is embedded into a smart card. He is able to sync his smart card with the web-based and smart mobile application and create summary health reports. A strong earthquake occurs and destroys many buildings of the city's infrastructure and Seddiq becomes homeless. While waiting at the emergency shelter, the dust level is increased and Seddiq's experiences severe coughing and breathing problems. The busy Japanese paramedic keeps him aside for examination. Seddiq, shares his smart phone with the offline PHB application with the paramedic, who can see his medical history and treats him quickly with oxygen and blood pressure medication. Then, the paramedic moves on to the next casualty. Once recovered, Seddiq records the event and medication on his smart phone. When the connectivity to the Internet is repaired, he syncs his update with his smart card with the web-based PHB. Using this approach, the paramedic, Seddiq, and the people that depend on him benefit from the PHB. The time constrained paramedic does not need to spend time assessing Seddiq's problem, particularly if there is a language barrier or if Seddiq is in shock and cannot coherently communicate.

5.5 Personal Health Book and Meaningful Use

A personal health book must have a favorable use perception. The PHB should also be considered as source of information for healthcare professionals. Both portability issues can be resolved if we combine wellness activities with healthcare requirements. This can achieved is we have a simple model that all stakeholders will understand. A proposed model is the

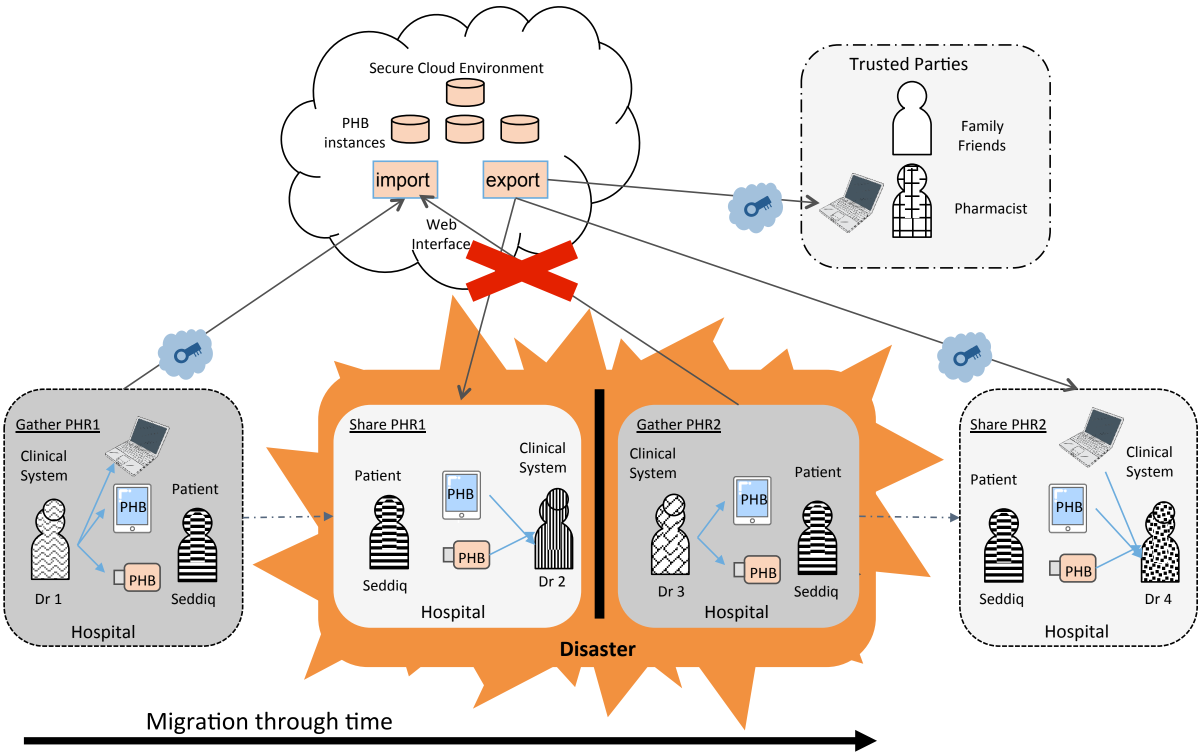


Figure. 5.4: The Portable Health Book Use-case in Disaster Affected Area

Portable Health Book cycle. In the cycle there are three keys events which are shown in Figure 5.5. The next sections will briefly describe what happens during each event.

5.5.1 PHB Cycle Measurement

An individual's wellness and clinical health is categorized, itemized and measured in data units. Current wellness indices are measured at home using Over the Counter (OTC) devices such as digital thermometers, weight scales, blood pressure machines, mobility (step counters), sleep apnea sensors (reference). Clinical health measured are measures at healthcare organizations (clinics and hospital) and include a myriad of measures (Blood tests, urine tests, tissue biopsies, xrays, MRI scans).

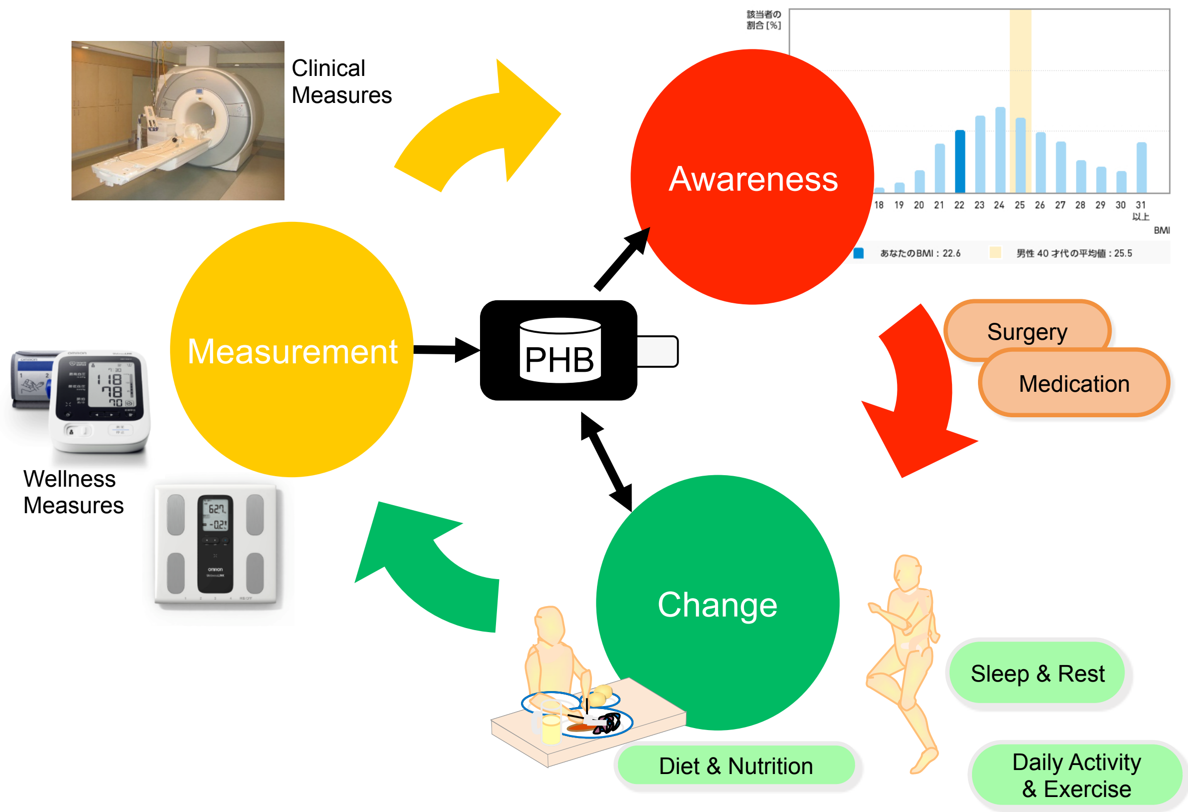


Figure. 5.5: PHB Cycle to add Meaningful Use to All Stakeholders

The results of the measurement categories, items and values are stored in a Personal Health Record (PHR). The PHB is an ideal store for the PHRs. The PHB should be able to capture both wellness and clinical measurements from the source. The type of devices the PHB resides will depend on the system that is producing the health data. For home devices, it is possible to interface with the web service such as Omron's Wellness System. Capturing the PHRs from the clinical sources hospital has been described in previous chapters.

5.5.2 PHB Cycle Awareness

Health awareness is based on the result and analysis of the wellness and clinical measurements. Wellness awareness is for the individual and can be a presentation of the current measurement or a graph showing a change in time: how weight and blood pressure has changed over the last week or month.

Clinical awareness is for the health professional or clinician and is often a presentation of the current test, measurement or examination. As per normal practice, the physician or health professional will make the patient aware of their health condition (the diagnoses). Awareness will also involve the next steps for the patient - the need for medicine, surgery or change in lifestyle.

In the PHB, health awareness is the XML report that can be viewed and shared. With technical improvement, the individual can search the PHB for particular data items, such as weight and blood pressure changes in the last six months and produce a report that can include graphs and tables. The PHB report will inform both the individual and whoever the report is shared with. When given to a physician, he will be able to see the individual's overall health trends and also compare them with the hospital tests and analysis.

5.5.3 PHB Cycle Change

Change is how the individual responds to the wellness and clinical awareness. Changes is the active component that the individual is responsible for and controls. For wellness, the awareness of overweight and high blood pressure should lead to change in diet, improved nutrition, increased exercise, or more rest. Following clinical advice, the individual will take prescribed medicine, may take time off from work, or change lifestyle such as reducing sugar intake or quitting smoking. To assess the impact of the changes to improve health requires measurements and the cycle continues.

In the PHB, change is actions taken before the next measurement (the next PHR). Depending on the point of interest, such as taking a prescribed medicine and changing diet, the individual can monitor the expected improvement.

5.5.4 PHB Cycle and Clinical Integration

The PHB can provide a combination of wellbeing health data and clinical data and provide 'meaningful use' to the individual. For example, once the individual is 'aware' of the non-communicable disease such as hypertension or diabetes, the individual can monitor changes (taking medicine, reducing lifestyle risks) through subsequent measurements, and then becomes more aware via the physicians follow-up analysis of the modifications. In another example, if an individual has had surgery and is recommended to do daily exercise, the health measurements can inform both the physician and the patient of the rate of recovery.

The PHB cycle can provide 'meaningful use' over a personal lifetime. It can represent be a digital version of the 'blue book' and 'personal Health Record book'. In USB format, this is applicable to developing countries such as Nigeria. Preferably, the individual's health data is stored on the web and managed by secure cloud services. The basic cycle of 'self' health monitoring and evaluation can be applied until old age.

5.5.5 Use-case: Wellness data with clinical data

Seddiq is an over-weight middle-aged male, has the PHB application on various devices and uses it to gather, store and share his medical history. Seddiq is diagnosed with type 2 diabetes. His physician has advised him to eat high-fiber and low fat foods, exercise regularly, monitor his blood sugar and take diabetes medication with insulin therapy. Seddiq decides to use his PHB App to help reduce the impact of type 2 diabetes. He changes his lifestyle, reduces intake of animal fat and confectionery sweets, starts exercising and goes to bed early. He measures his blood glucose, urine glucose, BMI and steps-per-day, and pushes the records onto his PHB APP. He queries the PHB database and creates weekly trends for all variables. He is aware of his progress and emails the report (as a pdf attachment) to his physician on a weekly basis. The physician quickly understands the progress or non-progress and can email a prescription for further type 2 diabetes, if needed. Using this approach, Seddiq has more control of his health status and the doctor is able to follow his progress with less need for Seddiq to visit the health clinic.

5.6 Summary

The Portable Health Book is a simple application with an independent database. In its simplicity it can be attractive to many stakeholders and their health needs. It can be used by migrants in the developing countries and thus alleviating many social problems. It can be used by the healthcare organizations to reduce medical errors and improve efficiency. It can accommodate wellness health data with clinical data and thus enabling the individual to understand their own health status in a more meaningful manner.

In the next chapter, the research objectives, approaches and research findings will be summarized. We also list our recommendations for further research.

Chapter 6

Conclusion and Future Work

6.1 Summary

We discussed the importance and the issues of the portability of the Personal Health Records. In order to improve the portability of Personal Health Records, we proposed a Personal Health Book application. The application can be installed at a personal device as a stand-alone or can be deployed on the cloud. We focused and proposed the following three technologies to improve total efficiency of the application - (1) a decomposed data types managed database design to improve the portability of the Personal Health Book, where data storage size will be small enough to be stored on a portable stand-alone storage such as a smart card/USB, (2) an import algorithm, which can understand the Personal Health Book database structure and has a level of understanding of the imported Personal Health Records content, (3) an export algorithm, which can understand the Personal Health Book database and read it while maintaining the link with related components during the export process. In the following, we summarize our key contributions.

First, we designed a decomposed data types managed database for the Personal Health Book that can be small in size. We introduced the existing major database design methodologies (Long, Wide, Decomposed), and discussed their pros and cons. We carried out a simulation to compare the designs in term of number of generated rows, number of unnecessary item repeats, database size growth, number of generated cells and wasted memory size which are the performance indicators for a good database design.

The simulation constituted of 10,000 records, in different 3 cases (12, 2 and 5 number of hospitals), each case had different maximum number of unique items (up to 10, 0 and 5).

On an average the decomposed database design performed better than the long database design with 16.7% in number of rows generated, 88.9% in number of unnecessary data repetition, 40.29% in database size growth, 42.84% in number of generated cells and 79.87% in size of wasted memory. As expected, the Decomposed design performed the best and we decided to use it for the Personal Health Book database.

The second part of the simulation used to evaluate the designed database behavior in terms of data items storing and size. On an average (of first 10 records) the stored items were 106.28% less than the imported items from various PHRs, that is 984 bytes stored compared to 3,216 bytes imported.

Second, we developed an algorithm for importing Personal Health Records into the Personal Health Book. With this algorithm along with a schema with minimal restrictions to validate the imported XML documents, Personal Health Records can easily get imported into the Personal Health Book, regardless of what is the content and how it is formatted or named. We carried out an experiment to import Personal Health Records from two hospitals in real world to verify the algorithm.

With the help of the import algorithm, the imported PHRs were successfully stored into the Personal Health Book database. However, the Japanese PHRs were manually translated and human bias was implied. The assumption was that the dictionary performed accurately. This can be treated as an ideal scenario.

Third, we developed an algorithm for exporting Personal Health Records from the Personal Health Book to other trusted parties. With this algorithm, we aim to export Personal Health

Records to any standard or format as it can understand the database design concept and get the data from it easily. Using the export algorithm, we exported the stored data and viewed them by using a web browser. We could read all data.

The algorithms can be placed in a framework/API to connect between the Personal Health Book and any other authorized party that needs to import/export Personal Health Records from the Personal Health Book.

6.2 Future Works

The proposed application did not consider the security and policy issues of the healthcare data when it is managed by an individual. Traditionally, the hospitals managed the data on behalf of the patients by maintaining policy and security issues.

From the technical aspect, it is predicted that technology (both hardware and software) will improve faster than anyone can imagine. The maintenance of sensitive healthcare data onto a smart card or smart phone (life time of these devices are only three years in maximum) will be an issue for non-technical people. Many people may not want to store their personal data on the cloud.

From the social impact aspect, we have experienced many disasters, unreached communities and aging population who can not manage their paper based health records, however it seemed very important to keep the records in a digitized form in a secondary source so that they can easily retrieve their records and share them in emergency situations. We feel that PHB could be an excellent alternative in order to support these segment of people.

The following list is the recommendations for further research:

- Evaluate the Personal Health Book while overloading it by importing big number of Personal Health Records from multiple sources.

- Investigate how to run the stand-alone Personal Health Book on different operating systems/platforms.
- Research the enhancement of the import and export algorithms to accept the existing Personal Health Records formatting standards as much as possible.

Acknowledgement

First of all I would like dedicate my achievement to my parents and my wife, for believing in me and their continuous support and encouragement to go further with my studies.

Then I would like to express my deep and sincere appreciation to Dr. Ashir Ahmed for his support and contribution. I could not complete this thesis without his support, guidance, encouragement and friendship over the three years and six months of my research.

I thank my supervisor Professor Akira Fukuda for his invaluable suggestions and guidance during regular lab meetings.

Beside to Prefosser Akira Fukuda, I would like to thank the judges, Professor Kazuaki Murakami and Professor Daisuke Ikeda, for giving me their time and valuable feedback on my research. Special appreciation to Professor Kazuaki Murakami for accepting my enrollment in Kyushu university and for his support and kindness, considering us as his children and taking care of us.

Also, I am very grateful to Professor Kunihiko Kaneko and Dr. Andrew Rebeiro-Hargrave for their contribution and guidance during our meetings.

I would also like to thank His Excellency The Minister of Interior, Lt-General Shaikh Rashid bin Abdullah Al Khalifa; The Director General of Ministry of Interior Court, Major-General Reyadh Eid Abdullah and the Director of the Directorate of Information Technology & Electronic Creativity, Major Ebrahim Mohammed Alsada for allowing me to get the opportunity to study abroad for the past three years and six months.

Thanks to Ms. Yoko Otsuru, Ms. Rika Shudo and Ms. Oiwa Yasura for their support and assistance, that helped me to concentrate on my research and finish my study in Kyushu University.

I would like to thank the Japanese Government for giving me the scholarship to study in Japan. I also would like to thank KDDI and Toyota for their fund support for my research activities.

I will not forget to thank xTURP lab members, Ms. Eiko kai, Ms. Jecinta Kamau, Mr. Yoshio Kajikuri and Mr. Kazi Mozaher Hossein for their inputs during xTURP weekly meetings and for their friendship and support.

References

[1] World Economic Forum Report: Rethinking Personal Data: Strengthening Trust, May 2012 http://www.weforum.org/issues/rethinking-personal-data.

[2] World Economic Forum Report: Unlocking the Value of Personal Data: From Collection to Usage, February 2013 http://www.weforum.org/issues/rethinking-personal-data.

[3] Ali M. Al-Khouri. Data Ownership: Who Owns My Data?. International Journal of Management & Information Technology, Vol.2, No.1, 8 Pages, 2012.

[4] Big Data, http://en.wikipedia.org/wiki/Big_data

[5] Seddiq Alkhan and Ashir Ahmed. Social Information Infrastructure and BIGDATA for Smart Services. Proceedings of e-CASE and e-Tech 2013, 18 pages, 2013.

[6] Krishna Kummamuru, Ajith Jujjuru, and Mayuri Duggirala. Generating Facets for Phone-based Navigation of Structured Data. Proceedings of the 21st ACM international conference on Information and knowledge management (CIKM'12), pp.1283-1292 , 2012.

[7] Ashir Ahmed, Takuzo Osugi, Rafiqul Islam Maruf, and Naoki Nakashima. Evolution of Remote Health-consultancy over Mobile Phone. Proceedings of the 2013 IEICE General Conference, 2013.

[8] Smart Card Alliance, Smart Cards and Biometrics in Healthcare Identity Applications Publication Date: May 2012 http://www.smartcardalliance.org.

[9] Claude Sicotte, Jean Louis Denis, Pascale Lehoux, and Francois Champagne. The Computer Based Patient Record Challenges Towards Timeless and Spaceless Medical Practice. Journal of Medical Systems, Vol.22, No.4, pp.237-256, 1998.

[10] Marion J. Ball, Melinda Y. Costin, and Christoph Lehman. The Personal Health Record: Consumers Banking in their Health. Stud. Health Technol. Informat., Vol.134, pp.35-46, 2008.

[11] Robert Steinbrook. Health Care and Amercian Recovery and Reinvestment Act. N.Engl. J.Med., Vol.360, No.11, pp.1057-1060, 2009.

[12] Maisie Wang, Christopher Lau, Fredrick A. Matsen, and Yonfmin Kim. Personal Health Information Management System and its Application in Referral Management. IEEE Trans on Information Technology in Biomedicine, Vol.8, No.3, 2004.

[13] Healthcare Information Management and Systems Society (HIMSS), 2007. HIMSS Personal Health Records Definition and Position Statement. Retrieved October, 22, 2009, from http://www.himss.org/content/files/phrdefinition071707.pdf

[14] https://www.healthvault.com

[15] http://www.dossia.org

[16] https://www.myhealth.va.gov

[17] Linda T. Kohn, Janet M. Corrigan, and Molla S. Donaldson, Editors. To Err Is Human: Building a Safer Health System. National Academy Press, 1999.

[18] Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995. Official Journal of the Europian Communities, No. L 281/31. 1995.

[19] ”Electronic Health Records”, `http://www.hims.org/ASP/topics_ehr.asp`

[20] Eiko Kai, Partha P. Ghosh, Sozo Inoue, and Ashir Ahmed. GramHealth BigData for Smart Healthcare Applications, Proceeding of the 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC13). 2013.

[21] Ofer Bergman, Ruth Beyth-Marom, Rafi Nachmias, Noa Gradovitch, and Steve Whittaker. Improved Search Engines and Navigation Preference in Personal Information Management. ACM Transactions on Information Systems (TOIS) TOIS Homepage archive, Vol.26, Issue 4, Article No.20, 2008.

[22] Ofer Bergman, Steve Whittaker, Mark Sanderson, Rafi Nachmias, and Anand Ramamoorthy. The Effect of Folder Structure on Personal File Navigation. Journal of the American Society for Information Science and Technology, Vol.61, Issue 12, pp.2426244. 2010.

[23] Ashir Ahmed, Koji Ishida, Masaharu Okada, and Hiroto Yasuura. Poor-Friendly Technology Initiative in Japan: Grameen Technology Lab. The journal of social business, Vol.1, No.1, 2011.

[24] Ashir Ahmed, Lutfe Kabir, Eiko Kai, and Sozo Inoue. GramHeath: A Bottom-up Approach to Provide Preventive Healthcare Services for Unreached Community. Proceeding of the 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC13). 2013.

[25] Seddiq Alabbasi, Andrew Rebeiro-Hargrave, Ashir Ahmed, Kazuaki Murakami, and Hiroto Yasuura. Personal Lifetime Data and Its Smart Management, Proceedings of the

2013 International Symposium on Intelligent System Engineering (ISISE 2013), Published in WIT Transactions on Information and Communication Technologies, Vol.63, pp.327-333, 2013.

[26] http://microformats.org/wiki/data-portability

[27] http://www.dataportability.org

[28] "Electronic Health Records Overview", Published by: National Institutes of Health National Center for Research Resources, April 2006. Available on: http://www.himss.org/files/HIMSSorg/content/files/Code%20180%20MITRE%20Key%20Components%20of%20an%20EHR.pdf

[29] Jennifer Allan, BSN, RN, and Jane Englebright, PhD, RN. Patient-Centered Documentation: An Effective and Efficient Use of Clinical Information Systems. Journal of Nursing Administration. Vol.30, No.2, pp.90-95, 2000.

[30] http://en.wikipedia.org/wiki/Electronic_health_record

[31] http://www.hl7.org

[32] http://wiki.hl7.org/index.php?title=Virtual_Medical_Record_(vMR)

[33] http://www.openehr.org/home

[34] http://www.xml.com

[35] https://msdn.microsoft.com/en-us/library/bb879915(v=office.12).aspx

[36] http://www.ibm.com/developerworks/data/library/dmmag/DMMag_2009_Issue2/Industry/

[37] http://www.jmir.org/2001/1/e7/

[38] http://xml.coverpages.org/healthcare.html

[39] http://blogs.msdn.com/b/healthvault/archive/2013/03/29/healthvault-at-convergence.aspx

[40] http://indivohealth.org

[41] http://www.visn4.va.gov/VISN4/MyHealtheVet.asp#sthash.MwZ8BBWH.dpuf

[42] Juha Puustjarvvi and Leena Puustjarvi. Personal Health Book: A Novel Tool for Patient Centered Healthcare. Proceedings of the International Conference on Health Informatics (HEALTHINF 2011), pp.386-393, 2011.

[43] https://ispub.com/IJH/5/2/5931

[44] Seddiq Alabbasi, Andrew Rebeiro-Hargrave, Kunihiko Kaneko, Ashir Ahmed, and Akira Fukuda, Personal Health Book Application for Developing Countries. International Journal of Advanced Computer Science and Applications (IJACSA), Vol.6, Issue 2, pp.120-128, 2015.

[45] https://www.patientslikeme.com

[46] http://www.google.com/intl/en/health/about/

[47] http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2900995/

[48] http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2900995/

[49] http://www.kidsfamilies.health.nsw.gov.au/publications/child-personal-health-record-(blue-book)/

[50] http://wellbeingfoundationnig.org/index.phpoption=com_content&view=article&id=1321&Itemid=786

[51] http://www.hl7.org/implement/standards/product_brief.cfm?product_id=6

[52] http://www.aafp.org/practice-management/health-it/astm.html

[53] Juha Puustjarvi and Leena Puustjarvi. Personal Health Book: A Cloud-Based Tool for Patient Centered Healthcare. Journal of Public Health Frontier, Vol.2, Issue 3, pp.146-155, 2013.

[54] Rajitha Tennekoona, Janaka Wijekoona, Erwin Harahapa, and Hiroaki Nishib. Per-hop Data Encryption Protocol for Transmitting Data Securely Over Public Networks. Procedia Computer Science, Vol.32, pp.965972, 2014.

[55] Stuart P. Goringa, Joseph R. Rabaiottib, and Antonia J. Jonesb. Anti-keylogging Measures for Secure Internet Login: An Example of the Law of Unintended Consequences. Computers & Security, Vol.26, Issue 6, pp.421426, 2007.

[56] Khosrow Dehnad. A Simple Way of Improving the Login Security. Coputers & Secruity, Vol.8, Issue 7, pp.607-611, 1989.

[57] Tzong-Chen Wu. Remote Login Authentication Scheme Based on a Geometric Approach. Computer Communications, Vol.18, Issue 12, pp.959963, 1995.

[58] Yen Sung-Ming and Liao Kuo-Hong. Shared Authentication Token Secure Against Replay and Weak Key Attacks. Information Processing Letters, Vol.62, Issue 2, pp.7780, 1997.

[59] http://www.w3.org/standards/techs/rdf#w3c_all

[60] http://en.wikipedia.org/wiki/Linked_data

[61] http://www.w3.org/standards/techs/owl#w3c_all

[62] Seddiq Alabbasi, Ashir Ahmed, Andrew Rebeiro-Hargrave, Kunihiko Kaneko, and Hiroto Yasuura. Design of a Personal Health Book for Ensuring Healthcare Data Portability, Proceedings of the International Conference on Applied and Theoretical Information Systems Research 2014, 15 pages, 2014.

[63] Seddiq Alabbasi, Ashir Ahmed, Kunihiko Kaneko, Andrew Rebeiro-Hagrave, and Akira Fukuda. Data Types Managed Database Design for Dynamic Content. Proceedings of IEEE TENCON 2014, 5 pages, 2014.

[64] `http://www.w3schools.com/sql/sql_view.asp`

[65] A. Ahmed, A. Rebeiro-Hargrave, N. Yohara, E. Kai, R. Hossein, and N. Nakashima. Targeting Morbidity in Unreached Communities Using Portable Health Clinic System. IEICE Transactions on Communications, Vol.E97-B, No.3, 2014.

[66] Francine L. Maloney and Adam Wright. USB-based Personal Health Records: An Analysis of Features and Functionality. Int. J. Med Informat, Vol.8, No.2, pp.97-111, 2010.

Published Papers

Journal Paper

(1) Seddiq Alabbasi, Andrew Rebeiro-Hargrave, Kunihiko Kaneko, Ashir Ahmed, and Akira Fukuda. Personal Health Book Application for Developing Countries. International Journal of Advanced Computer Science and Applications (IJACSA), Volume 6, Issue 2, pp.120-128, 2015.

International Conference

(1) Seddiq Alabbasi, Ashir Ahmed, Kunihiko Kaneko, Andrew Rebeiro-Hagrave, and Akira Fukuda. Data Types Managed Database Design for Dynamic Content. Proceedings of IEEE TENCON 2014, CD-ROM. 5 pages, 2014.

(2) Seddiq Alabbasi, Ashir Ahmed, Andrew Rebeiro-Hargrave, Kunihiko Kaneko, and Hiroto Yasuura. Design of a Personal Health Book for Ensuring Healthcare Data Portability. Proceedings of the International Conference on Applied and Theoretical Information Systems Research 2014, CD-ROM. 15 pages, 2014.

(3) Kunihiko Kaneko, Ashir Ahmed, and Seddiq Alabbasi. A Data Explorer System and Rulesets of Table Functions. Proceedings of the 5th International Conference on Society and Information Technologies: ICSIT 2014, CD-ROM. 6 pages, 2014.

(4) Seddiq Alabbasi, Andrew Rebeiro-Hargrave, Ashir Ahmed, Kazuaki Murakami, and Hiroto Yasuura. Personal Lifetime Data and Its Smart Management. Proceedings of the 2013 International Symposium on Intelligent System Engineering (ISISE 2013), Published in WIT Transactions on Information and Communication Technologies, Vol.63, pp.327-333, 2013.

(5) Seddiq Alkhan and Ashir Ahmed, Social Information Infrastructure and BIGDATA for Smart Services. Proceedings of e-CASE and e-Tech 2013, CD-ROM. 18 pages, 2013.