

AN ADVANCED PERSONAL HEALTH RECORD PLATFORM
FOR PATIENT EMPOWERMENT

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

ŞENAN POSTACI

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

JULY 2012

Approval of the thesis:

**AN ADVANCED PERSONAL HEALTH RECORD PLATFORM
FOR PATIENT EMPOWERMENT**

submitted by **ŞENAN POSTACI** in partial fulfillment of the requirements for the degree of **Master of Science in Computer Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. Adnan Yazıcı
Head of Department, **Computer Engineering**

Prof. Dr. Nihan Kesim Çiçekli
Supervisor, **Computer Engineering Department, METU**

Prof. Dr. Asuman Doğaç
Co-supervisor, **Computer Engineering Department, METU**

Examining Committee Members:

Prof. Dr. Fazlı Can
Department of Computer Engineering, BILKENT

Prof. Dr. Nihan Kesim Çiçekli
Department of Computer Engineering, METU

Prof. Dr. Özgür Ulusoy
Department of Computer Engineering, BILKENT

Assoc. Prof. Dr. Pınar Şenkul
Department of Computer Engineering, METU

Assoc. Prof. Dr. Ahmet Coşar
Department of Computer Engineering, METU

Date:

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: ŞENAN POSTACI

Signature :

ABSTRACT

AN ADVANCED PERSONAL HEALTH RECORD PLATFORM FOR PATIENT EMPOWERMENT

Postacı, Şenan

M.S., Department of Computer Engineering

Supervisor : Prof. Dr. Nihan Kesim Çiçekli

Co-Supervisor : Prof. Dr. Asuman Doğaç

July 2012, 79 pages

In recent years, many Personal Health Record (PHR) systems have been developed to retrieve patients' Electronic Health Records (EHRs) from external sources. However, current PHRs can provide access to only a small number of EHR systems, since there are many different interfaces, data formats and medical terminologies among different systems. When this is the case, all these diversity yields high integration costs. Development of such systems is difficult and expensive because of the reasons such as accessing to evidence based medical information, utilization of social networks to share information, incorporation of available medical knowledge models, etc. Due to the technical diversity of external information systems, a developer of a PHR system faces a difficult integration process when he wants to integrate a new source or service.

Integration of medical devices is also important and necessary in a PHR system. However, most of the medical device vendors use proprietary formats and protocols for communicating their devices with external systems; again, causing high integration efforts and costs.

In this thesis, these problems and challenges are addressed by providing an on-line personal

healthcare management platform, i.e. eSaglikKaydim which is built on top of a highly modular architecture and provides services based on worldwide standards. In this way, eSaglikKaydim platform can be integrated with any external health information service and medical device so that it maximizes the data variety retrieved from all kinds of external health data sources.

The work presented in this thesis is part of the OSAmI project supported by European ITEA and funded by the TÜBİTAK TEYDEB Project No: 9080004.

Keywords: patient empowerment, health records, external information sources, medical devices, medical standards and terminologies

ÖZ

HASTALARIN YETKİLENDİRİLMESİ İÇİN GELİŞMİŞ BİR KİŞİSEL SAĞLIK KAYDI PLATFORMU

Postacı, Şenan

Yüksek Lisans, Bilgisayar Mühendisliği Bölümü

Tez Yöneticisi : Prof. Dr. Nihan Kesim Çiçekli

Ortak Tez Yöneticisi : Prof. Dr. Asuman Doğaç

Temmuz 2012, 79 sayfa

Geçtiğimiz yıllarda, hastaların harici sistemlerden Elektronik Sağlık Kayıtları'nı (EHR - Electronic Health Record) çekebilmeleri amacıyla birçok PHR (Personal Health Record - Kişisel Sağlık Kaydı) sistemi geliştirilmiştir. Ancak, günümüzdeki PHR sistemleri, farklı sistemler arasında birçok farklı arayüz, veri formatı ve terminoloji bulunduğu için, sadece küçük sayıdaki EHR sistemlerine erişim sağlayabilmektedir. Durum böyle olunca, tüm bu farklılıklar yüksek entegrasyon masraflarına neden olmaktadır. Bu tip sistemlerin geliştirilmesi, kanıt tabanlı tıbbi bilgilere erişim, bilgi paylaşım için sosyal ağların kullanımı, mevcut tıbbi bilgi modellerinin birleştirilmesi gibi nedenlerden dolayı hem zor hem de pahalıdır. Harici bilgi sistemler arasındaki teknik farklılıklar nedeniyle, bir PHR sistemi geliştiricisi, kendi sistemlerini yeni bir kaynak ya da servisle entegre etmek istediğinde, zorlu bir entegrasyon süreciyle karşılaşmaktadır.

Bir PHR sisteminde, tıbbi cihazların entegrasyonu da önemli ve gereklidir. Ancak, birçok tıbbi cihaz üreticisi harici sistemlerle iletişim için özel veri formatları ve protokolleri kullanmakta, bu durum da yine, yüksek entegrasyon çaba ve masrafları gerektirmektedir.

Bu tezde, özde bahsedilen problemleri ve zorlukları çözmek için, yüksek modüler bir mimari üzerine kurulmuş ve dünya çapında standartlar tabanında servisler sunan çevrimiçi bir kişisel sağlık platformu sunulmuştur. Bu sayede, sözü geçen platform, bir çok harici bilgi sistemi ve tıbbi cihaz ile entegre olabilmekte, böylece her çeşit harici sistemden gelebilecek tıbbi verinin çeşitliliği artırılabilmektedir.

Bu tezde sunulan çalışma, ITEA tarafından desteklenen ve fonu TÜBİTAK TEYDEB Proje No: 9080004 tarafından karşılanan OSAmI projesinin bir parçasıdır.

Anahtar Kelimeler: hastaların yetkilendirilmesi, sağlık kayıtları, harici bilgi kaynakları, tıbbi cihazlar, tıbbi standartlar ve terminolojiler

To my family

ACKNOWLEDGMENTS

I would like to express my sincere gratitude and appreciation to my supervisor, Prof. Dr. Nihan Kesim iekli, for her encouragement, guidance and support throughout this study.

I would also like to express gratitude to my co-supervisor, Prof. Dr. Asuman Doęa for her guidance and support during my study.

I am deeply grateful to my family for their love and support. Without them, this work could not have been completed.

I am deeply grateful to Tuncay Namlı, Göke Banu Laleci Ertürkmen, Yıldıray Kabak and Mustafa Yüksel without whose invaluable guidance and contribution, this work could not have been accomplished. Furthermore, I thank Erdem Alpay, Suat Gönül, Alişan Yılmaz, Mert Gençtürk, Volkan Başar and all the other colleagues at SRDC, whose help, stimulating suggestions and encouragement helped me in all the time of research for and development of this thesis.

I would like to thank the Scientific and Technological Research Council of Turkey (TÜBİTAK) for providing the financial means throughout this study.

Finally, my special thanks go to my cousin Sinan, my friends Melih, Aykut, Serkan, Halil, İlke, Ömer, Yasemin, Mert and İnci for their help, support and cheerful presence through the course of this study. Thanks for giving me a shoulder to lean on whenever I need.

TABLE OF CONTENTS

ABSTRACT	iv
ÖZ	vi
ACKNOWLEDGMENTS	ix
TABLE OF CONTENTS	x
LIST OF TABLES	xiii
LIST OF FIGURES	xiv
CHAPTERS	
1 INTRODUCTION	1
2 BACKGROUND ON ENABLING TECHNOLOGIES AND STANDARDS	7
2.1 Enabling Technologies	7
2.1.1 OSGI (Open Services Gateway Interface)	7
2.1.2 Adobe Flex	10
2.1.3 Adobe BlazeDS	11
2.1.3.1 RPC Services	12
2.1.3.2 Messaging Service	13
2.1.4 Hibernate	15
2.2 Base Standarts and Specifications	16
2.2.1 Continuity of Care Record (CCR)	16
2.2.1.1 CCR Header	17
2.2.1.2 CCR Body	18
2.2.1.3 CCR Footer	19
2.2.2 eXtensible Access Control Markup Language (XACML)	20
2.2.3 Medical Terminologies	22

	2.2.3.1	SNOMED CT (Systematized Nomenclature of Medicine–Clinical Terms)	23
	2.2.3.2	LOINC (Logical Observation Identifiers Names and Codes)	23
	2.2.3.3	ICD 10(International Classification of Diseases - Version 2010)	23
	2.2.3.4	RxNorm	24
3		ESAGLIKKAYDIM DESIGN, IMPLEMENTATION AND FEATURES . . .	25
	3.1	eSaglikKaydim Requirements and Design	25
	3.1.1	eSaglikKaydim System Architecture	25
	3.1.2	eSaglikKaydim Architecture Components	26
	3.1.2.1	Infrastructure Bundles	26
	3.1.2.2	System Service Bundles	28
	3.1.2.3	PHR Application Bundles	29
	3.1.2.4	PHR Application API Bundles	31
	3.2	Modules of eSaglikKaydim with Implementation Details	35
	3.2.1	BlazeDS Connector	35
	3.2.2	JSON Connector	37
	3.2.3	XPHR Intergration Module	40
	3.2.4	QED Intergration Module	43
	3.2.5	FMIS Intergration Module	45
	3.2.6	NHIS Intergration Module	45
	3.2.7	Consent Editor Module	46
	3.2.8	Accu-Chek Glucometer Device Intergration Module . . .	47
	3.2.9	A&D Blood Pressure Device Integration Module	48
	3.2.10	Health Modules	48
4		USE CASE: INTEGRATION OF ESAGLIKKAYDIM WITH NATIONAL HEALTH INFORMATION SYSTEM AND DEMONSTRATION OF A SCENARIO	53
	4.1	Diabetes History	54
	4.2	Pregnancy History	55
5		RELATED WORK	59

6	CONCLUSION AND FUTURE WORK	61
	REFERENCES	63
	APPENDICES	
A	SCREENSHOTS FOR USE CASE DEMO SCENARIO	66

LIST OF TABLES

TABLES

Table 2.1	Different Bundle States	10
Table 5.1	Comparison with other PHR systems	60

LIST OF FIGURES

FIGURES

Figure 1.1	The OSAmI Commons Platform	5
Figure 2.1	OSGI Framework Layers	8
Figure 2.2	A Sample MANIFEST.MF File	9
Figure 2.3	How Flex Works	11
Figure 2.4	BlazeDS Client-Server Relation	12
Figure 2.5	Main Features of BlazeDS	13
Figure 2.6	BlazeDS Channel-Endpoint Interaction	14
Figure 2.7	Hibernate API	15
Figure 2.8	An example of Hibernate mapping file	16
Figure 2.9	General Structure of a CCR Document	20
Figure 2.10	XACML Processing Environment	21
Figure 2.11	XACML Building Blocks	22
Figure 3.1	eSaglikKaydim Architecture	27
Figure 3.2	Fundamental Bundles	28
Figure 3.3	Platform Data Sources and Interoperability Architecture	32
Figure 3.4	The BlazeDS Connector	37
Figure 3.5	JSON Message Attributes	38
Figure 3.6	Sequence Diagram of JsonRPC Login and Service Call	39
Figure 3.7	XPHR Actor Diagram	41
Figure 3.8	XPHR Extract Elements	42
Figure 3.9	QED Actor Diagram	43

Figure 3.10 QED Record Query Parameters	44
Figure 3.11 QED Record Query Codes	44
Figure 3.12 Consent Editor Rule Creation	46
Figure 3.13 Consent Editor Architecture	47
Figure 3.14 Consent Editor Server Architecture	48
Figure 3.15 getRecords result	49
Figure 3.16 Information Source Types	50
Figure 3.17 Details of a record	51
Figure 3.18 A pop up windows to add new problems	51
Figure 3.19 List of Problem Terminologies	52
Figure A.1 Hospital Visit Records of Mrs. Aymaz	66
Figure A.2 Problem and Diagnosis Records of Mrs. Aymaz	67
Figure A.3 Procedure Records of Mrs. Aymaz	68
Figure A.4 Procedure Records of Mrs. Aymaz (cont.)	69
Figure A.5 Medication Records of Mrs. Aymaz	70
Figure A.6 Medical Equipment Records of Mrs. Aymaz	71
Figure A.7 Consent Rules of Mrs. Aymaz	72
Figure A.8 Diabetes Diagnosis of Mrs. Aymaz	73
Figure A.9 Diabetes Goals of Mrs. Aymaz	74
Figure A.10 Diabetes Calendar and Glucose Measurements of Mrs. Aymaz	75
Figure A.11 Immunization Records of Mrs. Aymaz	76
Figure A.12 Tests and Laboratuary Results of Mrs. Aymaz	77
Figure A.13 Graphical Analysis of Glucose Tests of Mrs. Aymaz	78
Figure A.14 Pregnancy Record of Mrs. Aymaz	79

CHAPTER 1

INTRODUCTION

A new trend in healthcare is empowering patients through Personal Health Record (PHR) systems so that they can take a more active role in the management of their personal health. PHRs are not only electronic repositories of health information controlled or accessed by patients. They are also integrated with a wide variety of healthcare information technology systems including the personal medical devices to obtain the patients' current physiological status; the clinical decision support systems for patient-physician shared decision making; and the evidence-based medical sources on the Internet to automatically retrieve data according to the patient context. Today, there are many implementations of PHR systems and these implementations have ranged from online accessible web pages to patient portals that give them access to their Electronic Health Records (EHRs) [1].

In other words, PHR systems are being developed so that the patients can actively manage their own health and health care. Currently, there are already a number of PHRs with varying degree of capabilities. Typically they contain information about an individual's diagnoses, medications, allergies, lab test results, immunization records, and other personal health information. Many PHRs also provide linkages to convenience tools (e.g. requesting appointments, requesting prescription renewals, asking billing questions) and communication tools to assist the patient in connecting with various health care professionals.

Recently a major report on "The Value of Personal Health Records" by the Center for Information Technology Leadership (CITL) [2] modeled the differences between the existing PHR architectures, both those that currently exist in the healthcare marketplace as well as the future PHRs. CITL defined the architectures using four dimensions: i) methods of data incorporation, ii) types of data systems, iii) number of data sources, and iv) type of data exchange.

These four dimensions are then used to distinguish the four PHR architectures: provider-tethered, payer-tethered, third-party, and interoperable. The first three PHR architectures are the most prevalent in the current market. The report concludes that the true added value of the PHR systems will be achieved when future PHR systems will allow all patients to access data from providers, payers, or other organizations that have different data systems but which support standard interfaces and the semantic interoperability.

- **Provider-tethered PHRs:** A provider-tethered PHR benefits the provider by enabling patients to view lab results, edit and supplement their medication lists, and allow for automated appointment scheduling from this provider, but would not help patients to access their records in other providers or track their healthcare expenditures or learn more about their levels of coverage in the payer systems.
- **Payer-tethered PHRs:** Payer-tethered PHRs benefit the payer by allowing patients to track their healthcare expenditures or learn more about their levels of coverage but would not enable patients to view lab results, edit and supplement their medication lists, and allow for automated appointment scheduling.
- **Third-party PHRs:** The third-party PHRs allow patients to access data from other providers, payers, or other organizations that have different data systems. But this requires the construction of separate interfaces to each data source, which are costly without standards.
- **Interoperable PHRs:** The interoperable PHRs will allow all patients to access data from providers, payers, or other organizations that have different data systems but which support standard interfaces and the semantic interoperability.

As pointed out in the report on "The Value of Personal Health Records" by the Center for Information Technology Leadership (CITL), to be able to exploit their potential, the future PHR systems must be interoperable with different data systems rather than being limited to "Provider-tethered" or "Payer-tethered" models. For this, they must support standard interfaces and communication protocols for technical interoperability and the semantic interoperability must also be resolved. However, a recent survey [3] investigating the major 48 PHR systems on the market today discovered that "...none of the PHRs use existing medical standards for the storage and communication of their data". Only three systems supported HL7v2,

one could handle HL7v3 messages for data exchange. Four products support Continuity of Care Record (CCR) [4] for content. And regarding standardized data exchange, none of the products claimed fully interoperable support for data exchange through an IHE integration profile [5]. The survey concludes that "At the present time, PHRs have implemented only basic features allowing only basic patient empowerment. The breakthrough for empowerment will require cost intensive implementations of medical standards to provide interoperability and health services which let the users make a choice for the usage of a PHR."

In recent years many regional and national healthcare networks based on Electronic Health Records (EHRs) have been established all over Europe. The initial motivation of these networks was the ubiquitous provisioning of patient data for supporting structured care scenarios with the aim of reducing time and costs for exchanging patient data among co-operating health care professionals. In addition PHRs are becoming more popular in Europe. These systems span a wide range of use cases including personal health management, wellness and prevention. However, current PHR systems have access only to a limited set of EHR systems because of high integration costs with respect to the diversity of interfaces, data formats, and vocabularies used by the different systems.

Development of a personal healthcare platform is a difficult and costly undertaking because it involves several tasks including managing EHRs, PHRs and Personal Health Devices; accessing to evidence based medical information; utilization of social networks to share information; provisioning of shared decision support and incorporation of available medical knowledge models and predictive algorithms. Realizing these functionalities requires not only specialized knowledge in technology and medicine but also tedious and costly integration tasks. As a result, personal healthcare mechanisms are currently only provided in a fragmented fashion, as standalone applications or services. Technically, it is possible to use these applications as external services on top of each EHR and PHR platform but given the existing semantic and technical diversity of platforms, the developer of a personal healthcare service will face the challenge of an expensive integration process each time he wants to integrate another source or service.

Personal Health Devices are also essential to the practice of modern personal healthcare services. Although a majority of the Personal Health Device vendors are still using proprietary formats and protocols for communicating their devices with external systems, some vendors

involved in standardization activities have started implementing the most prominent standards in this domain, i.e., ISO/IEEE 11073 standard set and Continua Health Alliance [6] guidelines, for their new portfolio of devices. ISO/IEEE 11073 standard set is based on an object-oriented domain information model called the "Domain Information Model (DIM)" [7], and Continua Health Alliance guidelines detail the use of ISO/IEEE 11073 standard set for specific medical devices, by exploiting ISO/IEEE 11073 Device Specializations.

Technical interoperability is another important issue and can be achieved by using the relevant existing standards such as IHE QED [8] and IHE XPHR [9].

In this thesis, we address all these problems and challenges by providing an on-line personal health platform based on a CCR based repository. It overcomes these challenges by having a highly modular architecture and providing services based on world-wide standards. The platform can be integrated with any external health information system or medical device by developing necessary adapters and these adapters can be plugged in easily, even during the run-time of the platform without effecting other modules and adapters, thanks to its modular architecture. In this way, the platform maximizes the data variety retrieved from any kind of external health data sources.

This thesis has been realized as a part of the OSAmI project (Open Source Ambient Intelligence) [10] supported by European ITEA and funded by TÜBİTAK TEYDEB Project No 9080004. The European ITEA research project OSAmI aims to create a platform that is joint component and service-oriented, and this platform is based on a common architecture and principles. Many developers can produce industry-related solutions by using reusable ambient intelligence applications provided by this platform. OSAmI project consists of several networked several sub-projects which aim to develop solutions for different industry categories by using the same OSAmI architecture and principles.

In other words, OSAMI targets open source common foundations for a dynamic service-oriented platform (Figure 1.1) which is able to personalize itself in large diversity of cooperating. According to the vision of this project the relationship between humans, computers and electronic devices has evolved rapidly, defining technology eras with shifts in the related information technology (IT) business leadership. From the one-to-many relationship - computer versus human users - in the enterprise during the mainframe era in the 1960s, computers moved to the family environment with the personal computer (PC) in the 1980s. With the

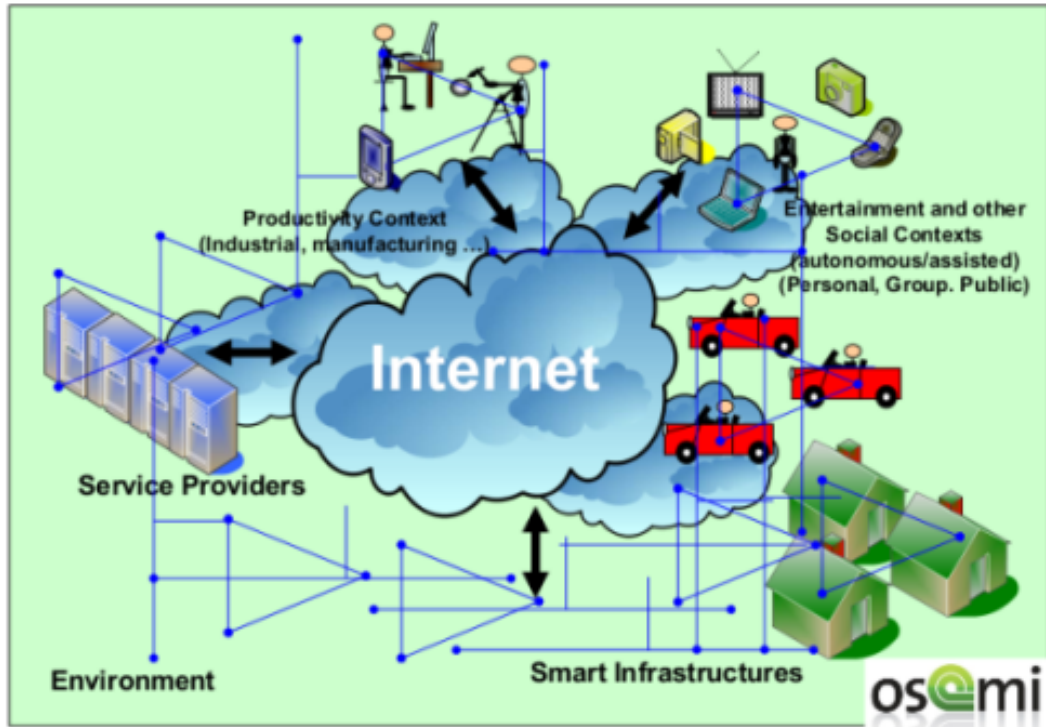


Figure 1.1: The OSAmI Commons Platform

mobile phone, they established a personal one-to-one relationship ten years later. The emergence of Ambient Intelligence is consequence of moving to a one-to-many relationship with phone complements, WiFi routers, gaming consoles, MP3 players, set-top boxes (STBs), TVs and infrastructures with impressive computing and storage capabilities. This environment is an enabler for a new concept of global and transversal platform that will exploit the real potential of the network affecting all business areas. In this convergence process, the Ambient Intelligence can be defined as an automated service provider embedding the devices. The Ambient Intelligence platform will be able to personalize itself dynamically in devices according to the context (i.e. physical containers, user needs and environment). The OSAmI consortium shares the vision of this platform emerging from a community process. [11]

Thanks to the modular architecture of the personal health platform developed that is presented in this thesis, which will be described in Section 3, it is being used to develop a pilot diabetes monitoring application in Turkey within the scope of the FP7-ICT-288209 EMPOWER Project [12]. The diabetes patient pathways are developed based on the Turkish "National

clinical guideline for management and treatment of diabetes in adults”.

The PHR platform will also be used for FP7-ICT-PSP 297260 PALANTE Project [13]. PALANTE Project provides a service for managing and monitoring Ankylosing Spondylitis disease. This service, namely Virtual Arthritis Clinical Service (VACS), is being develop as a module and also will be integrated with other modules of the PHR system.

The organization of this thesis is as follows: Chapter 2 briefly summarizes the background on the enabling technologies and standards. In Chapter 3, the design and implementation of the PHR platform are explained. A use cases scenario of the platform about a diabetic patient who gets pregnant is described in Chapter 4. In Chapter 5, the related work is explained on existing PHR systems. Finally, Chapter 6 concludes this thesis and presents the future work.

CHAPTER 2

BACKGROUND ON ENABLING TECHNOLOGIES AND STANDARDS

2.1 Enabling Technologies

2.1.1 OSGI (Open Services Gateway Interface)

The OSGi alliance is a consortium in worldwide, established by leading technology innovators that has extensive expertise about creating open specifications for software based on JAVA technology [14].

In order to achieve modularity, OSGI Alliance defined a specification called OSGI Service Platform [15]. This platform can be investigated in two categories: OSGI framework and standard service specifications set. OSGI framework, provides the main services that enable service oriented applications to be installed and executed. One of the most important properties of this framework is that the applications running on top of this framework can be managed dynamically. That means, an application running on OSGI framework can be stopped, restarted, installed and deleted at runtime without affecting other applications running on top of the same framework. Thanks to this property, OSGI can be used as not only a service network gateway, but also an application framework. Thus, Eclipse project, developed by IBM, is configured to be run on top of OSGI framework and so, plug-in applications can be plugged in and out easily on the framework.

There are 5 main layers on the basis of JAVA based OSGI framework. **Execution environment** runs on the operating system layer and responsible for the execution of the applications. **Module layer** provides dependency management and versioning of OSGI bundles (applica-

tions). **Lifecycle layer** manages the lifecycle operations such as execution, restart, installation and deletion of the applications. In order for these operations to be executed, the OSGI bundles register themselves to the **service registry**. This registration process is controlled by the service layer. Finally, the **security layer** handles the security aspects by limiting bundle functionality to pre-defined capabilities.

OSGI framework provides many benefits for the application developers:

- It allows applications that uses different versions of the same software libraries to run without any problem. So it prevents version conflicts.
- Bundles are designed for a specific purpose and can be used in different systems as components. This approach led to birth of component based programming concept.
- Bundles can be executed and stopped at runtime. Therefore, there is no need to stop the whole system to update small parts of the system.
- There are many open source application developers, so this standard reaches many other developers and quickens the adaptation to this standard.

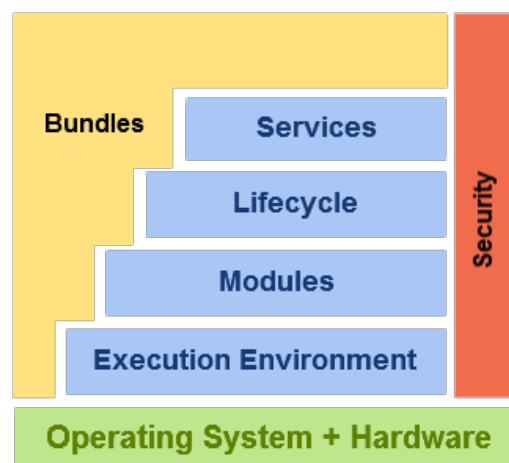


Figure 2.1: OSGI Framework Layers

Bundles

A bundle consists of several JAVA classes, additional resources (libraries, configuration files,

etc.) and a MANIFEST.MF file where all the information related to the bundle is kept. An example of a typical MANIFEST.MF file with OSGI Headers can be seen in Figure 2.2.

```
Bundle-Name: PHR Manager
Bundle-SymbolicName: tr.com.srdc.phr.manager
Bundle-Description: Manages all activities of PHR Platform
Bundle-ManifestVersion: 2
Bundle-Version: 1.0.0
Bundle-Activator: tr.com.srdc.phr.manager.bundle.PHRManager
Import-Package: org.osgi.framework;version="1.3.0"
Export-Package: tr.com.srdc.phr.manager.util
```

Figure 2.2: A Sample MANIFEST.MF File

The meaning of the contents in the example is as follows [16]:

- **Bundle-Name:** Denotes a name, which is in a human-readable form, for the bundle.
- **Bundle-SymbolicName:** This is the only required field of a MANIFEST.MF file and denotes the unique identifier of a bundle.
- **Bundle-Description:** Describes the bundle's functionality.
- **Bundle-ManifestVersion:** Denotes the OSGI specification version.
- **Bundle-Version:** Denotes the version of the bundle.
- **Bundle-Activator:** Denotes the Activator class name which is executed when the bundle starts.
- **Export-Package:** Denotes all the JAVA packages in a bundle that can be shared with other bundles.
- **Import-Package:** Denotes all the JAVA packages which are imported to be used from other bundles.

Bundles can be dynamically installed, started, stopped, updated and uninstalled on runtime and this management is handled by the lifecycle layer. Each bundle in an OSGi framework has different states during its life cycle, and these states are explained in the following table [17]:

Table 2.1: Different Bundle States

Bundle State	Description
INSTALLED	The bundle has been installed.
RESOLVED	All Java classes that the bundle needs are available. This state indicates that the bundle is either ready to be started or has stopped.
STARTING	The bundle is being started, the BundleActivator.start method will be called, and this method has not yet returned. When the bundle has an activation policy, the bundle will remain in the STARTING state until the bundle is activated according to its activation policy.
ACTIVE	The bundle has been successfully activated and is running; its Bundle Activator start method has been called and returned.
STOPPING	The bundle is being stopped. The BundleActivator.stop method has been called but the stop method has not yet returned.
UNINSTALLED	The bundle has been uninstalled. It cannot move into another state.

2.1.2 Adobe Flex

Flex is a rich Internet application development framework that allows developers to easily create Rich Internet Applications (RIA) that can run on many web browsers. For a web browser to be able to execute Flex based RIA's, Adobe Flash Player must be installed since, Flex is based on Flash runtime environment.

Adobe Flex comes with a standard development kit (SDK) which provides rich Internet application developers reusable and extendable user interface components, remote data retrieval services and many different event handling mechanisms. With these components, developers can create interactive and visually dynamic applications easily. Briefly, Flex provides developers the following facilities [18]:

- Designing and creating dynamic graphical user interfaces that can run on any browser with Adobe Flash Player plug-in.
- A runtime environment that can be accessed worldwide.
- The power of ActionScript programming language.

- Integrating multimedia elements such as sound and video.

In order to develop RIA's with Flex, two languages are used: an XML based language, namely MXML, used for designing application display elements and ActionScript which is an ECMAScript-compliant [19] object-oriented programming language that is used for creating application logic. The code combination, combination of MXML and ActionScript, must be compiled into binary SWF [20] files. [18]

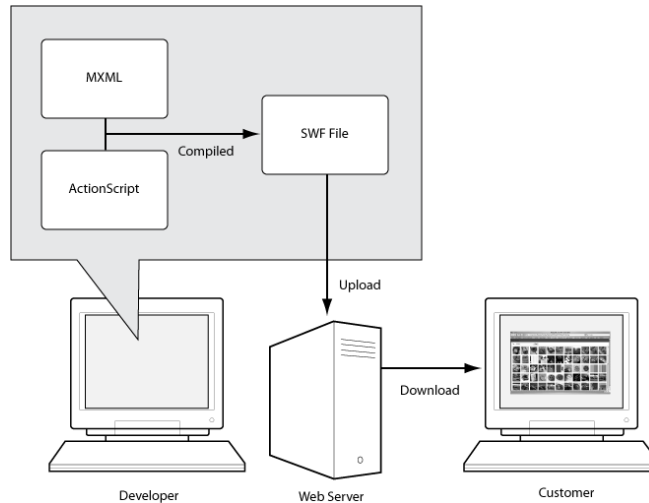


Figure 2.3: How Flex Works

Flex provides various mechanisms in order to communicate with remote servers. A Flex application can make HTTP and Web Service calls and also it supports low-level socket communication with other systems. However, the most important technology, which enables Flex applications to directly call methods of remote JAVA objects, is Adobe BlazeDS. It is considered to be the fastest way of communication between Flex applications and JAVA based servers and it is described in the following subsection.

2.1.3 Adobe BlazeDS

RIA's have become very popular with the new technologies like Adobe Flex/AIR, Ajax, Java/JavaFX and Microsoft Silverlight. These technologies provide some data access and interconnectivity services to interact with server environments, such as PHP, Java, Microsoft ASP .Net. Adobe has provided an open source server-based Java remoting and web messaging

technology, BlazeDS, that enables developers to easily connect to back-end distributed data and push data in real-time to Adobe Flex and Adobe AIR applications for more responsive rich Internet applications (RIA).

BlazeDS provides mechanisms that allow developers to create applications to connect a client application to a server-side application and share data in the server-side among multiple clients that are connected to the same server. BlazeDS also provides real-time messaging among multiple clients. [21]

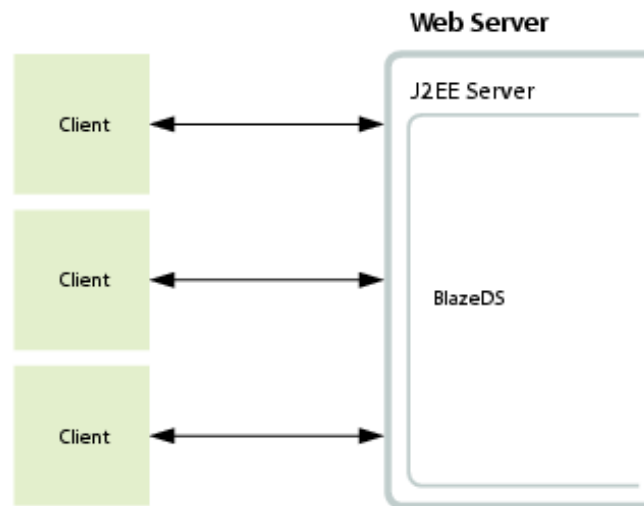


Figure 2.4: BlazeDS Client-Server Relation

Adobe BlazeDS API provides two important services to RIA developers, namely RPC and Messaging services

2.1.3.1 RPC Services

RPC Services enable Flex applications to call methods on remote servers and they provide 3 different mechanisms for this purpose:

- **WebService:** Flex applications can call web service methods on a remote server.
- **HTTPService:** Flex applications can communicate with the remote server over HTTP.
- **RemotingService:** Flex applications can directly call methods of JAVA objects on a remote server. This is one of the most important features of Adobe BlazeDS technology

since being able to call methods of JAVA object directly provides a huge performance advantage. When data requested from a remote server, requested data, which is in JAVA object form, is serialized into AMF (Action Messaging Format) [22] and transferred to client side. When client receives the data, it is deserialized into ActionScript objects and consumed.

2.1.3.2 Messaging Service

The Messaging Service enables asynchronous communication between a client and a server. It also enables bridging by using JMS [23] Adapter for JMS topics and queues on an embedded or external JMS Server. This bridging operation enables message exchange between Flex clients and JAVA applications. [21]

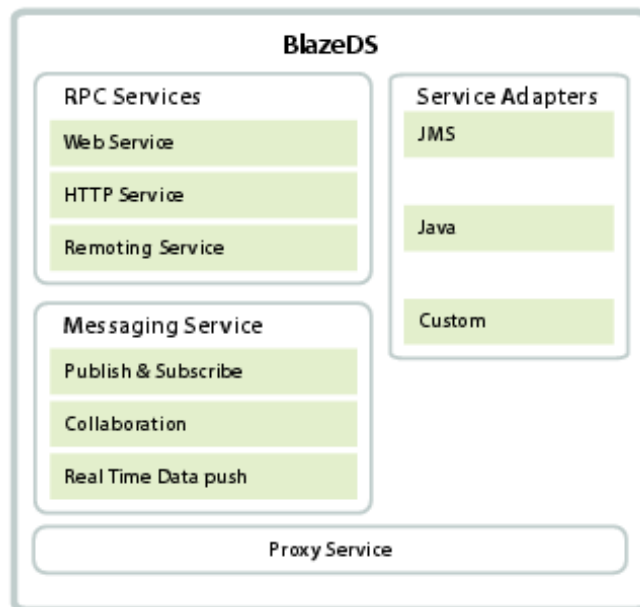


Figure 2.5: Main Features of BlazeDS

Message-Based Framework

To enable asynchronous communication between a client a server, BlazeDS uses a message based framework. In this framework, two exchange patterns are used. In the first pattern, a request-response mechanism is used. The server receives a request from the client and processes it. After the request is processed by the server, the client retrieves a response from the

server that contains the processing outcome. This patterns is also used by the RPC services, as well.

In the second pattern, a publish-subscribe mechanism is used. The clients subscribe to a server, and the server routes the messages to all the clients that are subscribed to itself.

Channels and Endpoints

The clients benefit from channels, in order to be able to send messages across the network. Channels contain information about the message formats, network protocols and behaviors, message destinations and the application code. Messages are translated by these channels and delivered to the endpoints of the servers.

To ensure the consistent communication between clients and server, channels dictate an ordering based mechanism. In this way, messages sent by the clients to servers have corresponding responses from the server.

Clients communicate with the endpoints of the server via channels. The endpoint of the server deserializes the messages and passes them to the message brokers in a generic JAVA form. Then, the message broker sends these messages to the appropriate clients.

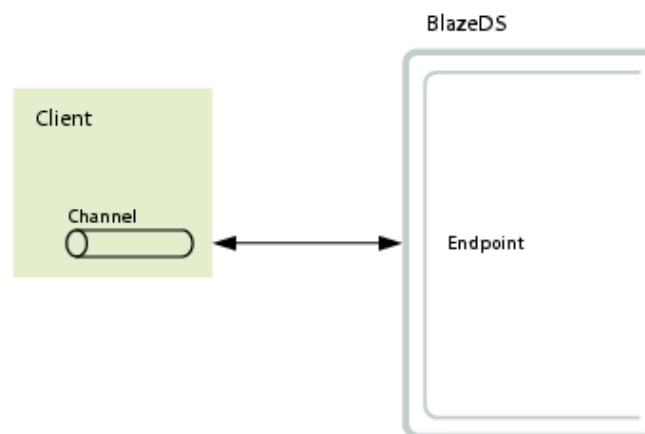


Figure 2.6: BlazeDS Channel-Endpoint Interaction

2.1.4 Hibernate

Working at the same time with object-oriented paradigm and relational database is an exhaustive work. As the number of classes increase, the number of the database tables and methods to map these objects to necessary columns in a table increases dramatically. Hibernate is a JAVA framework that provides ORM mappings between JAVA objects and database tables; and between JAVA data types and SQL data types. ORM stands for object/relational mapping and means mapping of data to a relational data model. In this framework each JAVA object corresponds to a database table, so each object can be persisted in the database easily and reduces the amount of coding and ease a developers work by data handling in SQL and JDBC [25]. Additionally, it provides querying mechanisms in both SQL and HQL (Hibernate Query Language), which is derived from SQL, languages.

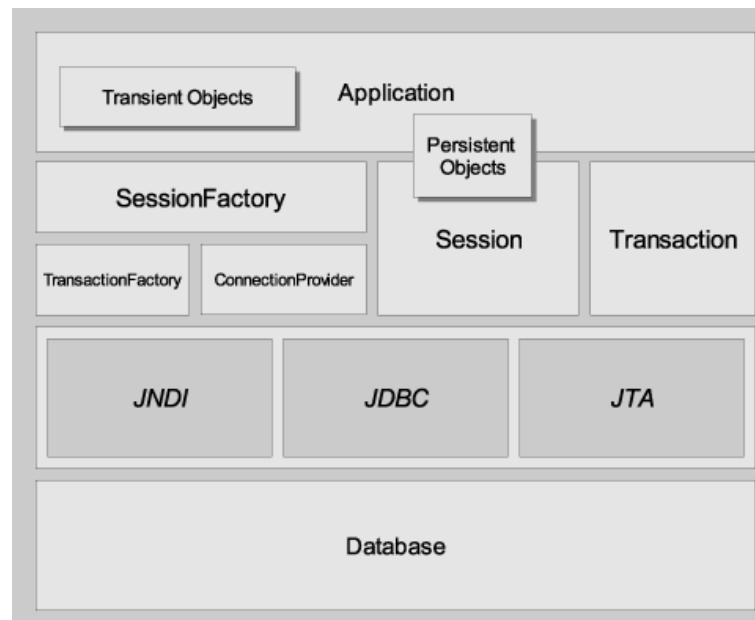


Figure 2.7: Hibernate API

Mapping is realized by writing simple XML files to load and store objects telling Hibernate that which object is persisted into which table of the which database (Figure 2.8). These XML files are easily generated by several tools and plugins. In these files, relations like one-to-many, many-to-many can be handled and each field of a JAVA class is mapped to a column of a database table. [24]

```

<?xml version='1.0' encoding='utf-8'?>
<!DOCTYPE hibernate-configuration PUBLIC
    "-//Hibernate/Hibernate Configuration DTD 3.0//EN"
    "http://hibernate.sourceforge.net/hibernate-configuration-3.0.dtd">

<hibernate-configuration>

    <session-factory>

        <!-- Database connection settings -->
        <property name="connection.driver_class">org.hsqldb.jdbcDriver</property>
        <property name="connection.url">jdbc:hsqldb:hsql://localhost</property>
        <property name="connection.username">sa</property>
        <property name="connection.password"></property>

        <!-- JDBC connection pool (use the built-in) -->
        <property name="connection.pool_size">1</property>

        <!-- SQL dialect -->
        <property name="dialect">org.hibernate.dialect.HSQLDialect</property>

        <!-- Enable Hibernate's automatic session context management -->
        <property name="current_session_context_class">thread</property>

        <!-- Disable the second-level cache -->
        <property name="cache.provider_class">org.hibernate.cache.NoCacheProvider</property>

        <!-- Echo all executed SQL to stdout -->
        <property name="show_sql">>true</property>

        <!-- Drop and re-create the database schema on startup -->
        <property name="hbm2ddl.auto">create</property>

        <mapping resource="events/Event.hbm.xml"/>

    </session-factory>

</hibernate-configuration>

```

Figure 2.8: An example of Hibernate mapping file

In Figure 2.8, content of a hibernate mapping file is shown. In this file, database configurations and attributes such as connection pool size, session management, SQL dialect, etc. are defined as properties. Moreover, a hibernate mapping resource that handles mapping of JAVA objects to database tables is added to the hibernate file.

2.2 Base Standards and Specifications

2.2.1 Continuity of Care Record (CCR)

To define the PHR summary data, ASTM International [26] provided the E2369-05 Standard Specification for Continuity of Care Record (CCR) [4] in 2005. CCR is one of the most widely used standard that defines a core data set for the most relevant administrative, demographic,

and clinical information facts about a patient's healthcare, covering one or more healthcare encounters. It contains various sections such as patient demographics, insurance information, diagnosis and problem list, medications, allergies and care plan.

The idea behind the CCR standard is very simple: it contains certain relevant health information - such as one's diagnoses and conditions, medications, allergies, and past procedures - can be reliably and unambiguously tagged using XML within a single file, then this information can then be passed across communications networks where it can be read and understood by servers and software applications in a vendor-neutral manner. Software vendors who agree to share this set of computable health data about a person gain value in the same way that any network effective product or service gains value, e.g. telephones and faxes. [27]

CCR is commonly represented and transmitted as XML documents and an XML schema (XSD) is provided by ASTM to ensure interoperability among various systems. According to this schema, a CCR document consists of 3 main sections, namely CCR Header, CCR body and CCR Footer.

2.2.1.1 CCR Header

CCR Header defines the properties of a CCR document. These properties and their usage are described as follows:

- **CCR Document Object ID:** A uniquely generated identifier to identify each CCR document.
- **Language:** The language of the CCR document.
- **Version:** The version of the CCR implementation that is used to generate the CCR document.
- **Date / Time:** The exact date and time of creation of the CCR document.
- **Patient:** The patient himself/herself whose health data is aggregated to construct the CCR document.
- **From:** The field that denotes who or what generated the CCR document.
- **To:** The field that denotes who or what received the CCR document.

- **Purpose:** The reason why the CCR document is constructed.

2.2.1.2 CCR Body

CCR Body section includes patient specific clinical information. This clinical information is denoted by 17 subsections which are described as follows:

- **Payers:** Includes information related to patient's payers, third party insurances and any financial statements related to patient's health.
- **Advance Directives:** Includes information of patient's advanced directives, any kind of healthcare documents and location of these documents.
- **Support:** List of patient's contacts who provides support during the treatment process.
- **Functional Status:** Includes information related to patient's functional status such as competency, ability to care himself/herself, daily activities.
- **Problems:** Includes information related to patient's current and past clinical problems, conditions, diagnoses, symptoms, findings, and complaints.
- **Family History:** Includes information related to patient's genetic relatives in terms of possible health problems.
- **Social History:** Includes information related to patient's personal, social, occupational history.
- **Alerts:** Includes information related to patient's current and past allergies and adverse reactions.
- **Medications:** Includes information related to patient's current and past medications.
- **Medical Equipment:** Includes information related to patient's current and past implanted and external medical devices and equipments.
- **Immunizations:** Includes information related to patient's current and past immunizations.

- **Vital Signs:** Includes information related to patient's current and past basic health information such as blood pressure, pulse, blood type, height, weight, body - mass index, head circumference, etc.
- **Results:** Includes information related to patient's current and past laboratory results.
- **Procedures:** Includes information related to patient's current and past surgical, diagnostic, therapeutic procedures or treatments.
- **Encounter:** Includes information related to patient's current and past encounters and hospital visits.
- **Plan of Care:** Includes information related to patient's planned procedures, encounters, interventions, etc.
- **Healthcare Providers:** Includes information related to patient's current and past healthcare providers.

2.2.1.3 CCR Footer

Contains data defining all of the actors, as well as information about external references, all text comments, and signatures associated with any data within a CCR document and defines the following parameters:

- **Actors:** Includes information related to all the individuals, organizations, locations, and systems included within the CCR document.
- **References:** Includes information concerning all the references within the CCR document to external data sources where they can be URL's, clinical documents, electronic patient records, papers, etc.
- **Comments:** Includes information related to all the comments associated with any kind of data within the CCR document.
- **Signatures:** Contains all signatures associated with any kind of data within the CCR document.

The general structure of a CCR document can be seen in Figure 2.9

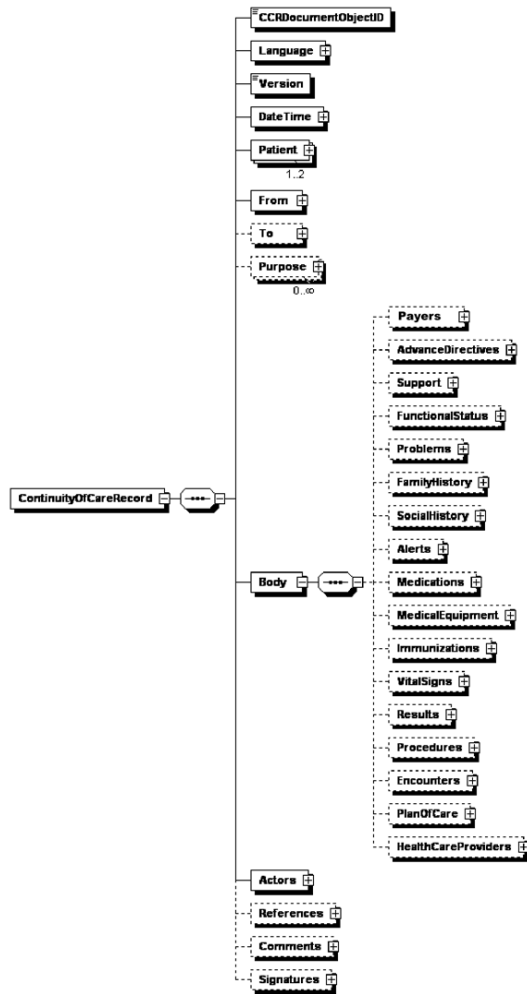


Figure 2.9: General Structure of a CCR Document

2.2.2 eXtensible Access Control Markup Language (XACML)

eXtensible Access Control Markup Language (XACML) is a standard defined by OASIS [28] and defines an XML based policy language as well as an access control mechanism language. The policy language describes the access control requirements while the access control language enables querying whether a given action should be allowed or not. The access control language also defines the Request and Response protocols between clients and the policy decision environments.

XACML defines access control rules for four main types: Subject, Resource, Action and Environment. Simple data sources to define a policy and request-response messages are con-

sidered to be the attributes of these main types. For example, attributes such as name, email address, role of a subject can be used in policies. Resource attributes can be resource identifier, time created, etc. Action attributes are unique identifier like "read", "write", "update" and "delete"; and these attributes can be combined to form more complex structures. Environment attributes define the current environment that is independent of other structures (i.e current time).

XACML uses XPath [29] standard for XML based resources. For instance, an EHR document may contain personal information of a patient and the access policy may include the age information of the patient. In that case, age information can be retrieved by XPath expression defined in the policy description. In this way, instead of the whole resource, only allowed sections of an XACML policy can be accessed. However, to perform these operations, the structure of the resource must be certain: The resource must be XML based and known by the XACML policy authors. In the healthcare domain, this can be achieved in organized and customized systems by means of content and communication.

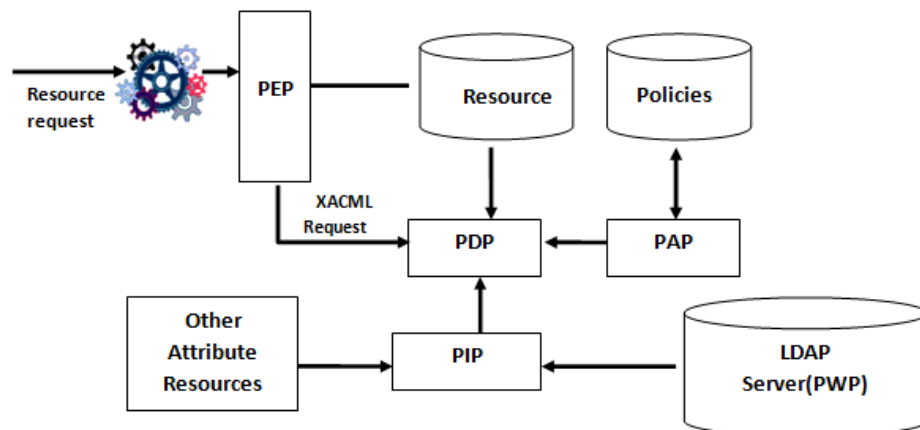


Figure 2.10: XACML Processing Environment

As it can be seen in Figure 2.10, XACML definitions define a processing environment model. In this processing environment, there are four main elements: Policy Enforcement Point (PEP), Policy Decision Point (PDP), Policy Administration Point (PAP), and Policy Information Point (PIP).

PEP is an access control mechanism entry point that separates XACML processing envi-

ronment from the application environment. It requests access decisions from PDP and this request contains attributes related to the resource, subject and action. These attributes are obtained by a service that is connected to the resource. However, this mechanism is not dynamic since, all attributes should take place in the request message and PEP must be aware of all the attributes in order to execute the policies. In this case, PIP solves this problem. When a value of attribute is need, a query is sent to PIP and PIP returns the response to PEP. [30]

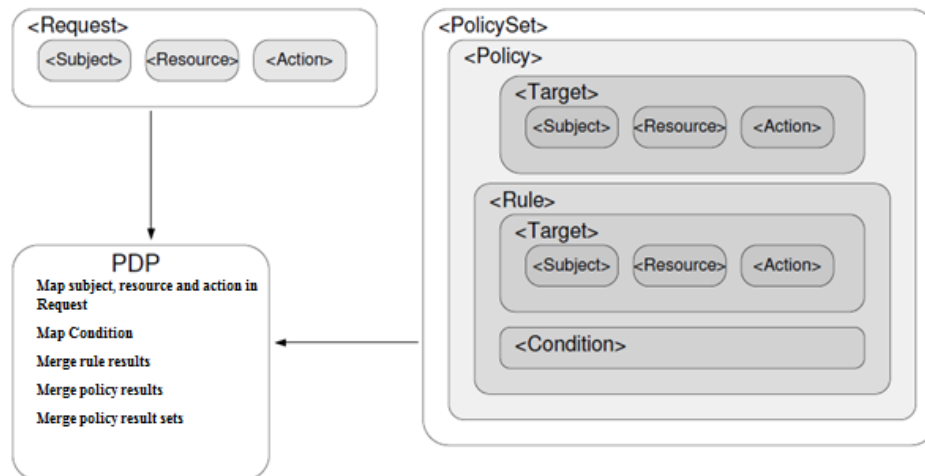


Figure 2.11: XACML Building Blocks

2.2.3 Medical Terminologies

In medicine, the clinical data structures such as EHR and PHR content templates refer to controlled vocabularies or terminologies to express semantics; i.e., the meaning of the terms used. For example, the observation for a patient can be expressed as a "heart attack" or a "myocardial infarction", and these mean the same thing to medical professionals. But unless the term is associated with a unique code from a code system, automated processing of the exchanged term is very difficult because an application, programmed to use "heart attack", would not understand "myocardial infarction". When the observation refers to a medical terminology system such as SNOMED CT and a code like "22298006 (Myocardial infarction)" is used to represent the observation, the meaning can be shared in a consistent and automated way as long as all involved applications use this code.

Therefore, there are many terminology standards such as Systematized Nomenclature of Medicine - Clinical Terms (SNOMED-CT) [31], Logical Observation Identifiers Names and Codes (LOINC) [33], International Classification of Diseases - 10th Revision (ICD-10) [35] and these terminologies are used for classification of different types of patient data like diseases, laboratory results, procedures, etc. In order to facilitate interoperability with external systems, the patient data should be coded with relevant terminology systems to the extent possible, e.g. apart from free text notes.

The most important medical terminologies are described in the following subsections:

2.2.3.1 SNOMED CT (Systematized Nomenclature of Medicine–Clinical Terms)

SNOMED CT is the most advanced clinical vocabulary available in any language in the world. It provides a machine-readable, well-formed clinical health terminology. Moreover, a meaning-based retrieval mechanism is enabled to provide effective access to required information. It also provides better communication and improves recording of EHR information. [31] [32]

2.2.3.2 LOINC (Logical Observation Identifiers Names and Codes)

To identify tests applied in laboratories and clinics, LOINC provides a set of universal names and identifiers for these tests. Today, medical terminology exchange is performed using HL7 messages by most of the laboratories and clinical services. However, the tests are identified by some internal identifiers, so it causes much confusion among healthcare services. LOINC addresses this issue by providing universal names and identifiers for laboratories and clinical systems. [34]

2.2.3.3 ICD 10(International Classification of Diseases - Version 2010)

ICD-10 by World Health Organization is the medical classification for all general epidemiological, health management purposes and clinical use; and this classification includes diseases, symptoms, abnormal findings, complaints and etc. Today, there are, approximately, 16000 different codes in ICD-10 medical terminology set. [35]

2.2.3.4 RxNorm

RxNorm is both a naming system for generic and branded drugs; and a semantic interoperation tool between pharmacy knowledge base systems and drug terminologies. RxNorm specialized in medicines and drugs, and it assigns unique identifiers for them so that it provides an efficient and unambiguous way of connection about drug-related issues among computer systems. [36]

CHAPTER 3

ESAGLIKKAYDIM DESIGN, IMPLEMENTATION AND FEATURES

Chapter 2 explained the data elements that a PHR system must contain and these elements are already included in CCR data model. Without having a modular architecture, managing all CCR sections and data element require a hard work.

3.1 eSaglikKaydim Requirements and Design

eSaglikKaydim provides a 7/24 accessible on-line platform for patients based on CCR data model. Besides, providing medical terminologies and worldwide standards, it aids patients to lighten their workload of creating health records. Also, by integrating eSaglikKaydim with external information systems, patients do not have to create their records, they just retrieve their health data from these systems. Therefore, in order to provide an easy-to-use and interoperable platform, the requirements of eSaglikKaydim platform have been elicited by examining CCR specifications.

3.1.1 eSaglikKaydim System Architecture

eSaglikKaydim comprises of various modular personal health applications. System's every application has a distinctive task such as recording and management of personal information; recording and management of medical information like allergies, drugs, laboratory results; tracking and management of diets, exercises and various illnesses. Owing to OSGI technology, every application within system can be plugged in or out easily.

The user interface of eSaglikKaydim, which was developed using Flex programming language has the duty of holding abovementioned applications. When eSaglikKaydim applications plugged into system, they are automatically installed and start running after user's choice and the interfaces of these applications appear on the user interface.

Every user has a personal account within SaglikKaydim and users can securely log into system by means of these accounts. Personal information of the users are kept securely and this information can be shared with their doctors, based on security measures defined by user. After logging into system, user can utilize applications he chose according to his needs and preferences. eSaglikKaydim applications can be added or removed during system's run, so users can start using new applications instantly. Moreover, general interface enables the users to view properties and control settings of the old eSaglikKaydim applications.

eSaglikKaydim consists of not only user application modules; it also contains modules that can provide services for clinical information systems, home caring services, medical information systems and other systems that can communicate with the platform. For example, a home caring system can record patient's various measurements to the platform by communicating with module that provides mentioned service. In the same way, a clinical information system can pull a list of patient's allergies and drugs used in the past, who is registered to the platform.

eSaglikKaydim architecture (Figure 3.1) is described in detail in the following section.

3.1.2 eSaglikKaydim Architecture Components

3.1.2.1 Infrastructure Bundles

- **PHR Manager:** PHR Manager is the main bundle of the system and provides required infrastructure for other bundles' operations. It provides following services for other bundles:
 - Configuration Management Service: This service includes configuration settings (platform user roles, application categories) and enables configuration of other bundles according to these settings.
 - Application Registry Service: This service enables registry of PHR Application

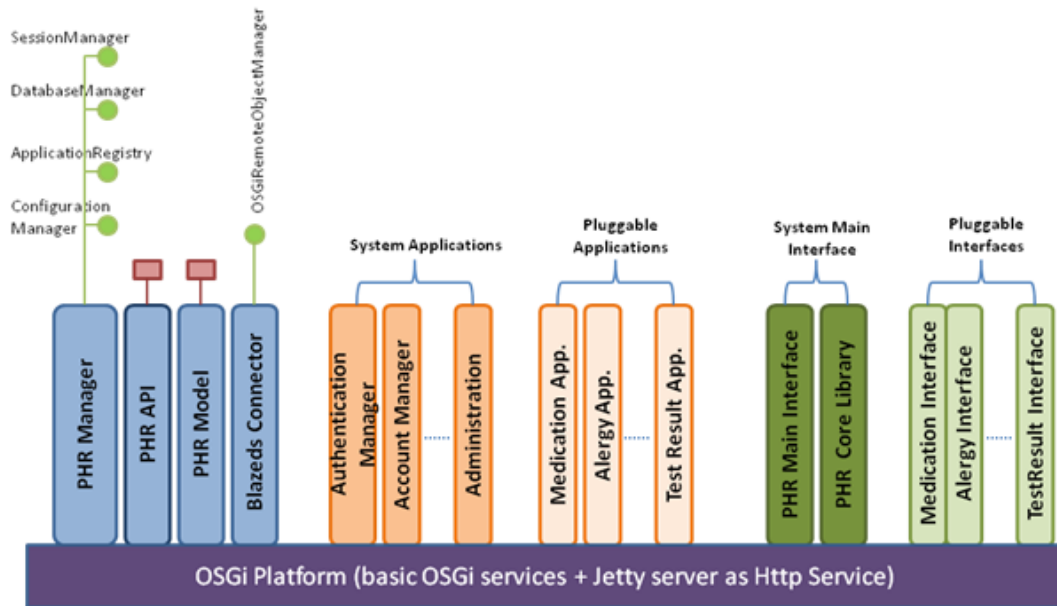


Figure 3.1: eSaglikKaydim Architecture

bundles, together with application information, to the system or deletion of it from the system. Moreover, it responds to System Service Bundles' queries

- Database Management Service: It enables the platform to control user logins.
- Session Management Service: This service manages users' authentication and sessions during their visits to eSaglikKaydim.
- Terminology Service Record Service: This service is used as a log file for HL7 Common Terminology Service based service which runs on the platform. Also, it responds to terminology queries, according to pre-recorded terminology services.
- **PHR Model**: The data model used in the platform is created on CCR health record standard. Every bundle in the system can access data model through this bundle. Moreover, Hibernate technology is used to easily access and manage data model.
- **PHR Common Terminology Service**: It provides data model and interfaces suitable for HL7 Common Terminology Specification.

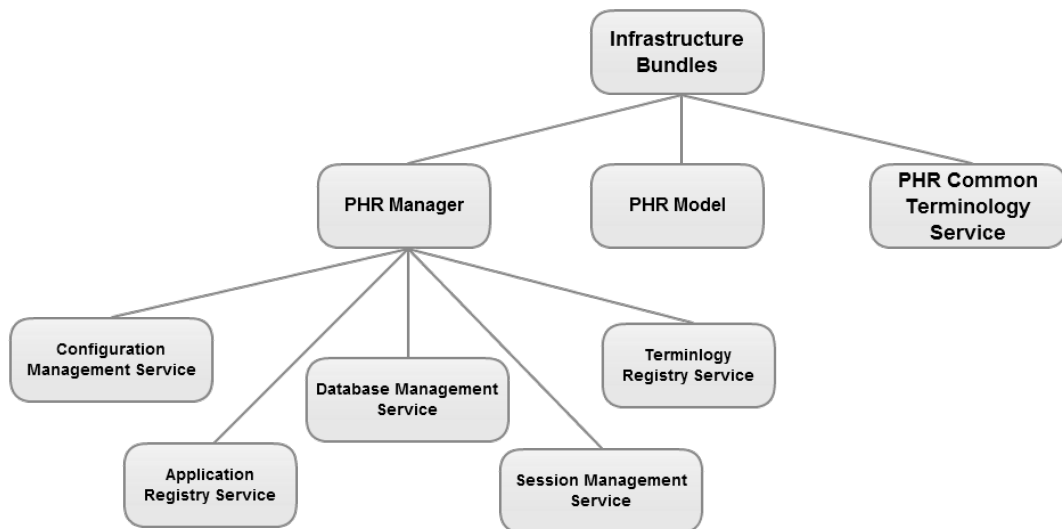


Figure 3.2: Fundamental Bundles

3.1.2.2 System Service Bundles

These are bundles which manage processes oriented to overall system. They provide auxiliary services for infrastructure and platform's general interface:

- **Login Manager:** Provides secure login into system for users.
- **Account Manager:** Provides access and management of accounts for users. Also, enables users to manage their PHR applications preferences in the platform.
- **BlazeDS Service:** Provides Adobe BlazeDS technology to other OSGI bundles in the system to enable generating Flex interfaces, which can communicate with Rich Internet Applications developers' OSGI bundles that use Adobe Flex.
- **JSON Service :** Provides JSON services to other OSGI bundles to enable conversion of CCR data model into JSON format.
- **USBS (NHIS) Service:** Provides data transfer for relevant bundles by retrieving data from Turkish Ministry of Health's National Health Information System [37].
- **AHBS (FMIS) Service:** Provides data transfer for relevant bundles by retrieving data from Turkish Ministry of Health's Family Medicine Information System [38].

- **Configuration Manager:** Provides interfaces for platform managers to administrate settings related to platform and registered applications.
- **UMLS CTS Servis:** Provides HL7 CTS definitions suitable health terminology services for queries on terminology systems that can be stored in UMLS (Unified Medical Language System) [39] based databases.

3.1.2.3 PHR Application Bundles

PHR applications are applications that are presented to user through a web interface and can be easily added or removed. Every PHR application consists of at least two bundles. The first of these is main PHR application bundle which can communicate through Flex web interface. This bundle must execute PHR Application API, provided by infrastructure, in order to perform fundamental functions and can exploit other services (Platform Services, Terminology Services, PHRModel etc.) simultaneously. Second bundle provides compiled Flex application interface to user over the web, through platform's HTTPService. At this point, Flex BlazeDS Service bundles play key role by providing connection between two bundles.

Another significant point in the platform is installation of applications to the system. During installation some technical information of the applications like characteristic credentials, name, description, category (disease management, health measurement devices, lifestyle support), remote object addresses and configuration parameters are provided to the system.

At present, following PHR applications are in the system:

- **PHR Alert:** Provides patients access to and management of allergy and reaction information (recording, updating, deleting)
- **PHR Basic Health Information:** Enables patients to manage characteristic information like height, weight, blood type, etc.
- **PHR Condition:** Enables patients to access and manage information like clinical problems, complaints, diagnostics and prognosis.
- **PHR Family History:** Enables patients to record and manage genetic information of patients with regards to possible health problems.

- **PHR Healthcare Provider:** Enables patients to record and manage information of persons responsible for caring and treating during illness.
- **PHR Hospital Visit:** Enables patients to access and manage examination information.
- **PHR Immunization:** Enables patients to access and manage immunization and vaccination information.
- **PHR Insurance Provider:** Enables patients to access and manage health insurance providers information.
- **PHR Medication:** Enables patients to access and manage information on used drugs.
- **PHR Personal Information:** Enables patients to access and manage their personal information.
- **PHR Procedure:** Enables patients to access and manage information about surgical, diagnostic therapeutic interventions.
- **PHR Social History:** Enables patients to access and manage information on business life, personal and social environments.
- **PHR Test Result:** Enables patients to access and manage laboratory, diagnostic or therapeutic results information.
- **PHR Diabetes Management:** Enables patients and health service providers to access and manage diabetes health plans organized according to patient's needs. Users can keep track of measurements, disease events, graphical analysis and insulin intake. Also, this application can process data obtained from devices like glucose meter, blood pressure monitoring device.
- **PHR Diet:** This application enables patients and their dieticians to manage diet programs arranged for patients. Patients can take a look at foods recorded in the system, find out nutritional values and record food they had via calendar based interface to the system. Also, users can track diet programs and share these programs with another users.
- **PHR Workout:** This application enables patients to manage prepared exercise programs. Patients can check up on numerous activities recorded in the system and prepare programs as they want or can check exercise

- **PHR Migraine:** This is an application that enables migraine patients to track their illness. While patient can record various information about sickness (symptoms, pain location etc.) into system, doctors can see statistical and graphical data generated in line with these information.
- **PHR Pregnancy Monitoring:** Enables pregnant patients to record events they encountered during pregnancy, doctor appointments, measurements and monitor them with their doctors. Also, weekly informative messages on interface about fetus development can be of help to patient.
- **PHR Message Board:** Enables patients to communicate with their doctors. Besides, when adjustment changes performed in the system (change of insulin profile, setting new blood pressure goals), it will send informative messages to the user.
- **PHR Hypertension:** In general, application provides shared environment for diabetes patients and their doctors and allow patients to regularly record blood pressure measurements.
- **PHR Clinical Documents:** With this application, patients can upload clinical documents to the system or download them from the system. Also, application can generate HL7 CDA [40] compatible medical reports from system's PHR information.
- **PHR Baby Monitoring:** With the help of this application, patients can easily record measurements about their children/babies and control their development.

3.1.2.4 PHR Application API Bundles

These bundles provide services to external systems like clinical information systems, home care systems, medical information systems to work together on specific aims. This aim may be recording of new information, retrieval of PHR records or any interaction between eSaglikKaydim and other external systems. Since these bundles are used in relevant PHR applications, CCR data model is used for PHR records. But in general sense, CCR is not used as a messaging standard or clinical document standard. Because of this, original services are expected to be based on HL7 CDA clinical document standard.

Platform architecture designed to provide the interoperability of both external data sources and various application interfaces is shown in Figure 3.3. As it is seen in the figure, the

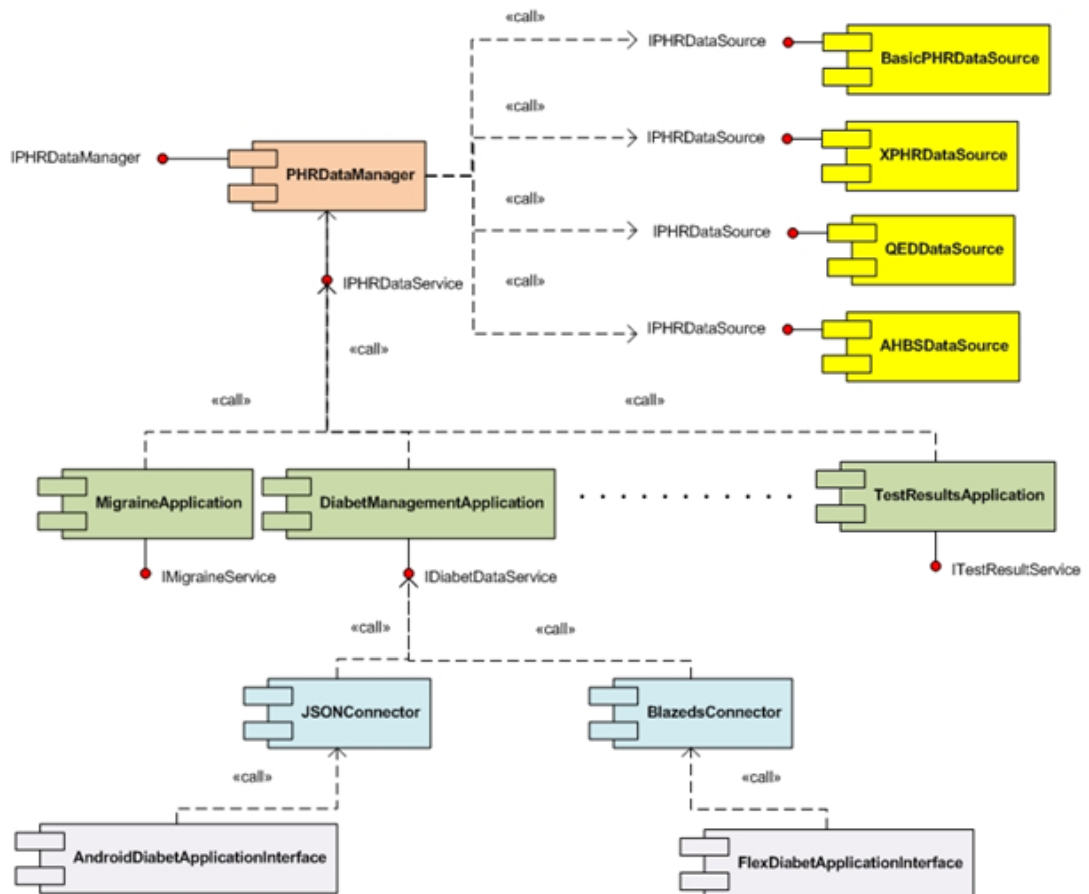


Figure 3.3: Platform Data Sources and Interoperability Architecture

component called PHRDataManager was developed within the architecture to provide access to data for applications from a single interface and port. This component implements IPHRDataServis interface, thus enabling a service which would allow all applications in the platform to record, query, update or delete the patients' records. As was mentioned in previously, ASTM CCR based data model is used as common model within the platform. This model presents a model which could represent all records in health sector. Alongside with this, although it provides basic data model for basic records (allergies, problems, procedures), CCR can be used in many different ways to create new specific record types by combining basic elements. In order to avoid this different utilization and to create compatible and consistent personal health record model within the platform, a record hierarchy was formed.

The modeling of each record type in this hierarchy is specified by rules. Actually, this method is very similar to Clinical Care Document (CDA) and CDA templates, which are based on

CCR Standards. What is done here is creating record template and arranging them in specific hierarchies. For example, while record type named "Results" holds all kinds of results of patient, Laboratory Result which is sub record type of it, includes medical test results and measurements; and sub record type Health Events includes various information (i.e. sleeping order, symptoms, pain logs) recorded about the patient's condition. In this way, the rules will be given indicating minimum number of elements for each record type and how these elements will be expressed within CCR. Within these rules, applications will be able to access records they needed in accordance with their own requirements. A UUID is assigned for every record type.

IPHRDataService interface which is implemented by PHRDataManager provides following basic operations for platform applications:

- **getRecords(credentials:UserCredential, recordTypeID:String, patientID:String, recordQuery:BasicRecordRequest):** Operation, where applications can query specific records belonging to specific patient, by giving record type ID. Query is performed by giving patient's unique identifier within the platform. Query can be performed for single record or for all records of that type. ID information of the user for whom the query is performed is given as an input of the operation. This ID information is used during evaluation of user's access to these records.
- **addRecord(credentials:UserCredential, recordTypeID:String, patientID:String, recordStorageRequest:BasicRecordRequest):** Enables application to add specific record type to patient's record. Record is approved before addition according to the rules specified for record type.
- **updateRecord(credentials:UserCredential,recordTypeID:String, patientID:String, recordUpdateRequest:BasicRecordRequest):** Used for changing patient's information in specific record. Record pairing is implemented by special record value within the request.
- **deleteRecord(credentials:UserCredential, recordTypeID:String, patientID:String, recordDeleteRequest:BasicRecordRequest):** Used for deleting patient's specific record.
- **getPHRRecordTypes():** Provides metadata of all record types in the platform.

- **isPHRRecordSupported(recordTypeID)** : Used to query the existence of specified record type in the platform.

As it can be seen from these operations, applications would be able to access patient records from a very simple interface. Alongside this, IPHRDataManager is another interface applied by PHRDataManager component. This interface registers the operations performed by IPHRDataSource interface, which provides record types and adapters that provides records, and the applications that provides these records to PHRDataManager component. As these records are created dynamically, it's possible to dynamically perform operations like adding new types of records or integrating with another data source while system continues to run.

There may be data sources that can provide more than one patient records from external or from within the platform. For integration of these sources, adapters applying IPHRDataSource interface should be developed and attached to the system. This interface also includes operations similar to IPHRDataService interface. Besides, it provides operations concerning patient's permission and configuration parameters for this data source. This way, patient can decide from which data sources his data can be extracted and transferred to the platform. One of these data sources, BasicPHRDataSource, represents platform's own database and keeps data in CCR model by directly using Hibernate technology. Also, this data source is a component that defines basic record types. These data sources needn't have to support all record types. They submit supported record types when they attach themselves to the system. This way, PHRDataManager component routes queries from PHRDataService to data source adapter that supports record type, taking into account patient's preferences. Afterwards, it returns assembling the data collected from data sources. For adding, deleting and updating actions, the actual location of the data and patient's primary data source information is considered and these processes are redirected to relevant sources. The default primary data source for platform is BasicPHRDataSource.

PHRDataService adapters have to provide a category implementing IPHRRecordService interface for each record type that support. This way, they perform operations by routing record operations received via IPHRDataSource to relevant record service. Operations of interface are stated below:

- **getRecord(patientID, recordRequest:BasicRecordRequest)**

- **addRecord(patientID, recordStorageRequest:BasicRecordRequest)**
- **updateRecord(patientID, recordUpdateRequest:BasicRecordRequest)**
- **deleteRecord(patientID, recordDeleteRequest:BasicRecordRequest)**
- **validateRecord(record:Object):** Controls record's compliance with record type rules and templates and if it isn't, returns validation report.
- **getInfoSourceContainer(record:Object):** In the platform, every record's creator, source (i.e. patient may be the source of record or medical device or doctor) and information must be kept together with record. According to record type, this operation returns data element that holds this information in CCR model.
- **canFilterParentRecords():** If record services provide one of the child record types, in order to benefit from the hierarchy of the record types, parent records can be filtered. This operation detects if the record service has this capability.
- **filterParentRecords(parentRecords:List<Object>):** Enables choosing and returning child records by filtering records that came in main record type. For example, if record type Blood Pressure Result is defined under Results record type, the service that will support this type can filter which of the Results records are in the Blood Pressure type. This way, by using this operation of record type's basic record service, PHRDataManager can assemble data for data source which doesn't support that record type but supports any record type from higher levels in hierarchy.

3.2 Modules of eSaglikKaydim with Implementation Details

3.2.1 BlazeDS Connector

The importance of Adobe BlazeDS technology for Rich Internet Application developers is described in Chapter 2. Due to its design, BlazeDS can only work on an application server, so it can not work in OSGI framework directly. Therefore, in order to serve eSaglikKaydim modules, which are developed using Flex, from an OSGI framework, **BlazeDS Connector** bundle was developed. This module enables developers to identify any JAVA object as remote object whose methods can be called from the Flex applications.

This module provides the following services:

- **(un)registerBundleClassLoader(ClassLoader loader):** Remote methods or messages between Flex applications and OSGI bundles may contain user-defined parameters. In order for OSGI environment to process these parameters, OSGI bundles need class loaders that includes classes related to the these parameters. Therefore, bundle developers must call these methods to be able use parameters in objects that are assigned as remote objects.
- **registerRemoteObject(String destinationID, Object service, boolean secure, List<String> authorizedRoles):** For a JAVA object to be accessed from Flex applications, first thing to do is to register these objects to **BlazeDS Connector** module. After a remote object is registered, as long as this module and OSGI bundles that contain the remote objects are not stopped; and remote object registry is not removed, Flex applications can call methods of these remote objects infinitely many times.

The first parameter of this methods, `destinationID`, is the unique identifier of the remote object that will be accessed by Flex applications. The second parameter, `service`, the object itself that is assigned as remote object. Third and fourth parameters include identity verification and authorization information. If a secure connection is desired, third parameter, `secure`, must be true. After that, BlazeDS will apply necessary security restrictions. For identity verification, an identity verification service, which has a method that requires a parameter for identifier (user ID, session ID, etc.) and a parameter for security (password, session ID, etc.) and returns the role of the user, must be registered. For authentication, the role information returned by the methods of identity verification service must be compared with roles in the fourth parameter, `authorizedRoles`.

- **unregisterRemoteObject(String destinationID):** When a JAVA remote object that is accessed by Flex application is no longer required, this method is called.
- **registerMessagingDestination(String destinationID, Object service, boolean secure, List<String> authorizedRoles):** If BlazeDS Messaging Service is required rather than Remote Object Service, this method is called and it executes in the same way as **registerRemoteObject** does.
- **unregisterMessagingDestination(String destinationID):** Removes the messaging service identified by the `destinationID`.

- sendMessage(String destinationID, Object msgBody, String sourceApplicationID, Set<String> msgTargetSet, Map<tring, String> selectorHeaders):** It is called for sending messages to Flex applications. As BlazeDS Messaging Service provides a publish-subscribe mechanism, messaging can be accomplished in three different ways: sending messages to all Flex applications, sending messages to a group comprised of specific Flex applications and sending messages to Flex applications according to their destinationID. This method supports these three mechanisms. The first parameter, destinationID, is the required identifier for messaging destination. Second parameter, msgBody, contains the data to be sent. Third parameter, sourceApplicationID, is the identifier of the JAVA server. This parameter may be important if the same destinationIDs have multiple message sender to identify who sends the message. Fourth parameter, msgTargetSet defines the group that the message will be delivered to. The last parameter, selectorHeaders, has information related to the specific Flex applications that the message will be delivered to.

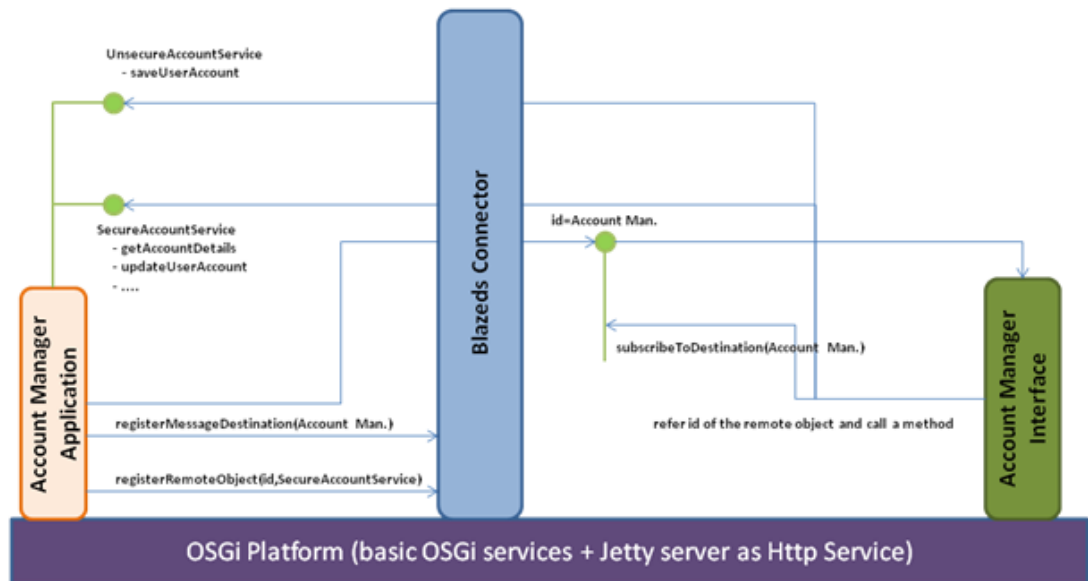


Figure 3.4: The BlazeDS Connector

3.2.2 JSON Connector

JSON Connector bundle was developed for applications and clients, which have low processing power, will be developed in mobile device environments and would rather use simple

JSON interface instead of complex structures like XML-AMF, to directly utilize services generated in OSGI-based servers. JSON (JavaScript Object Notation) [41] is a data transmission format which is easy for humans to understand and easy for computers to process. The aim of this bundle is to develop interfaces for Android-based mobile system developers as an API. However, this service generally can be used to open any service as JSON remote procedure call in OSGI environment. As a design, this service has much similarities with BlazeDS Connector module.

Service generates an HTTP endpoint and routes remote procedure calls received from this endpoint in specified format and coded in JSON to requested operations pre-registered to JSON Connector. Remote services, which are used as in Adobe BlazeDS, should be registered beforehand to JSON Connector. These registrations are called destinations and a singular ID is given to each destination. JSONRPC requests are sent within a JSONMessage envelope object.

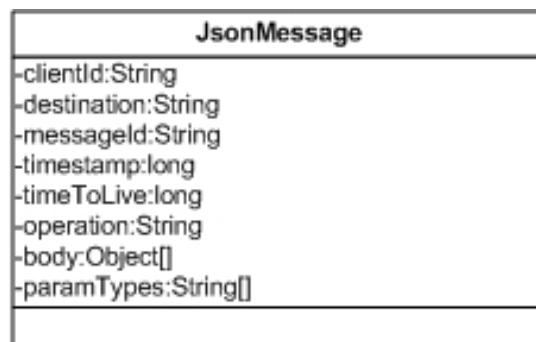


Figure 3.5: JSON Message Attributes

As can be seen from Figure 3.5, JsonMessage objects consist of certain parameters in order to make remote procedure calls. Of these parameters, clientID is caller system's singular Id, destination is singular name given to service targeted by call, messageId is singular Id given to pair return response with message, timestamp is message sending time, timeToLive is validity period of message, operation is the name of method to be called, body is parameters of method to be called, paramTypes denotes classes of parameters.

Figure 3.6 describes a standard JSON RPC call and process of user's entrance into system before this call. As can be seen, process starts with HTTP Request coming to JsonRPCServlet, which responds to Json RPC requests. This servlet picks message coded in JSON from within

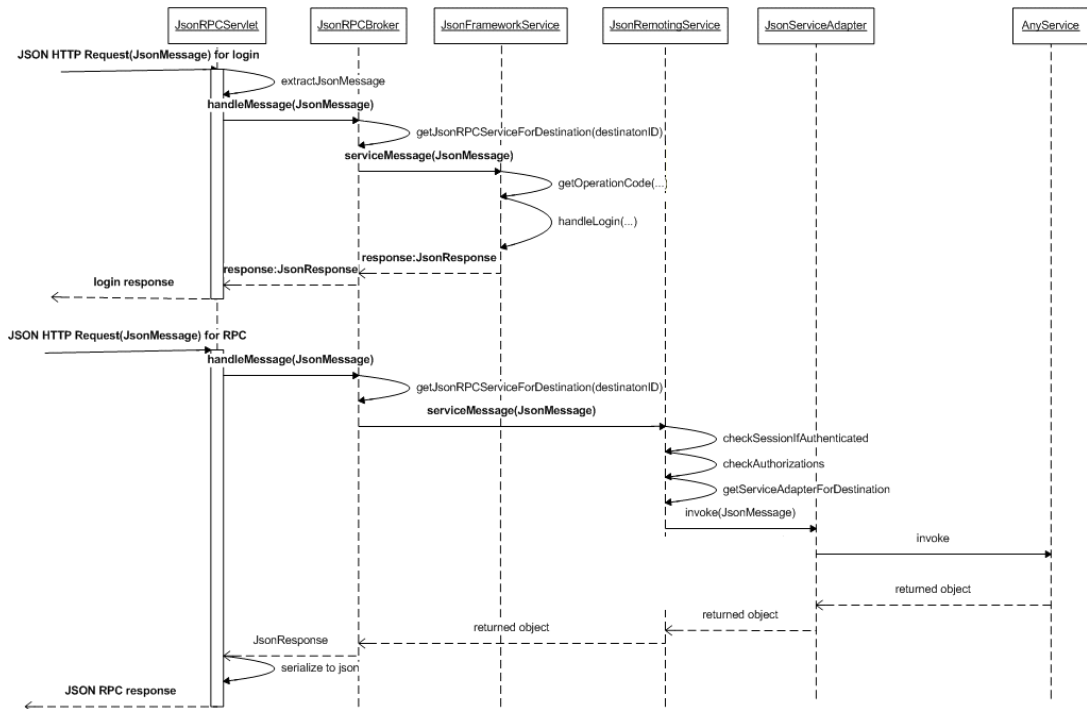


Figure 3.6: Sequence Diagram of JsonRPC Login and Service Call

HTTP Request and creates `JsonMessage` object by parsing this message. Afterwards, it migrates this message to `JsonRPCBroker`, which we can consider as service manager and which stands as singular (singleton) within component. `JsonRPCBroker` picks destination information from message and finds service that implements `IJsonRPCService` interface, to where destination is recorded. In order to support different messaging processes, more than one `IJsonRPCService` of this kind are enabled within JSON Connector. **JsonFrameworkService** and **JsonRemotingService** are implemented in current application. `JsonFrameworkService` is a general service which supports management operations like opening and closing session for service utilization of user. `JsonRemotingService` is a service which is used for invoking any destination service's methods.

Security specifications of a destination are also amongst metadata recorded during destination's record. Within this scope, person who recorded destination may ask that user, who will remotely access destination methods before destination's transfer, will have to open session in JSON Connector and will have to make authorization control. On the upper section of process shown in the figure, at first, user's login to service is displayed. If, after this procedure, user is authorized for service with identification information he provided, `JsonSession` is created and

recorded within HttpSession. Calls received after this are implemented through this created session.

In standard call situation, a call, that is to say JsonMessage object, reaches JsonRemotingService. After executing session and authorization controls as per metadata given during destination record, JsonRemotingService calls JsonServiceAdaptor's invoke method, which was assigned to destination during record. JSON service adapters may have different implementation according to preferences within platform. These applications can be assigned to destination by extending JsonServiceAdaptor abstract class while recording destination. Such an adapter named PHRJsonRemotingAdapter was developed for eSaglikKaydim personal health platform. Developers who will use JSON Connector in different environments can develop adapters in the same way. Fundamental function of these adapters is to invoke relevant method of service object recorded in destination by using method name and parameters from received JsonMessage object. Object returned by method is transmitted upwards, finally generating JsonResponse message and message sent to JSON serialized.

3.2.3 XPHR Intergration Module

IHE Exchange of Personal Health Record Content (XPHR) integration profile describes contents and format of intermediary records for import of summary information obtained from a PHR or EHR system to another PHR or EHR system and is within IHE Patient Care Coordination (PCC) technical framework. The purpose of this profile is to support interoperability between PHR systems used by patients and the information systems used by healthcare providers. Even though it doesn't cover all the data exchange requirements, it's one of the rare specifications designed for this purpose at the present time.

Actor diagram of XPHR is shown below. Here, Content Creator must have the ability to create content suitable for XPHR and Content Consumer must have the ability to process this content for profile suitability. XPHR integration module can convert records within platform to XPHR records and share them with other systems and at the same time can convert XPHR records to CCR model within platform and share them as well.

XPHR defined two content modules named XPHR Extract and XPHR Update. XPHR Update is used to report to the other side changes that occurred in XPHR Extract module sent before.



Figure 3.7: XPHR Actor Diagram

XPHR uses HL7 CDA as a content format and defined CDA templates to specify content details. Because these templates are based on CCD and CCD is based on CCR, it became easier to convert XPHR records to the platform's CCR model. Elements that may be within XPHR Extract are shown in Figure 3.8. Within this profile, equivalents of these elements in CCR are also given. Pairing was implemented with help of these mappings and is applicable within module.

As it was indicated, XPHR defines only content and format of imported records. Use of IHE Cross Enterprise Reliable Document Interchange (XDR) is recommended for transmission of these records. Basically, this profile gives definition of web service that has to be implemented by receiver of clinical document. Sender transmits document by calling this service. Developed XPHR integration module receives XPHR documents by implementing this web service and delivers them to the platform so that platform saves them to its database by converting them to CCR model. Besides this, in the same fashion module can summarize patient records in the platform and convert them to XPHR Extract (HL7 CDA) content.

Data Element Name	Opt	Template ID
Personal Information Name Address Contact Information Personal Identification Gender Date of Birth These components are required of all Medical Documents	R	1.3.6.1.4.1.19376.1.5.3.1.1.1
Personal Information Marital Status This component is optional in Medical Documents , but required if known in this specification.	R2	1.3.6.1.4.1.19376.1.5.3.1.1.1
Personal Information Race Ethnicity Religious Affiliation [2] These components are optional in Medical Documents	O	1.3.6.1.4.1.19376.1.5.3.1.1.1
Languages Spoken	R2	1.3.6.1.4.1.19376.1.5.3.1.2.1
Employer and School Contacts	O	1.3.6.1.4.1.19376.1.5.3.1.2.2
Hazardous Working Conditions	O	1.3.6.1.4.1.19376.1.5.3.1.1.5.3.1
Patient Contacts	R2	1.3.6.1.4.1.19376.1.5.3.1.2.4
Healthcare Providers	R	1.3.6.1.4.1.19376.1.5.3.1.2.3
Insurance Providers	R2	1.3.6.1.4.1.19376.1.5.3.1.1.5.3.7
Pharmacy	R2	1.3.6.1.4.1.19376.1.5.3.1.2.3
Legal Documents and Medical Directives	R2	1.3.6.1.4.1.19376.1.5.3.1.3.34
Allergies and Drug Sensitivities	R	1.3.6.1.4.1.19376.1.5.3.1.3.13
Conditions	R	1.3.6.1.4.1.19376.1.5.3.1.3.8
Conditions (cont)	R	1.3.6.1.4.1.19376.1.5.3.1.3.6
Surgeries	R2	1.3.6.1.4.1.19376.1.5.3.1.3.12
Medications – Prescription and Non-Prescription	R	1.3.6.1.4.1.19376.1.5.3.1.3.19
Immunizations	R2	1.3.6.1.4.1.19376.1.5.3.1.3.23
Doctor Visits / Last Physical or Checkup	O	1.3.6.1.4.1.19376.1.5.3.1.1.5.3.3
Hospitalizations	O	1.3.6.1.4.1.19376.1.5.3.1.1.5.3.3
Other Healthcare Visits	O	1.3.6.1.4.1.19376.1.5.3.1.1.5.3.3
Clinical Tests / Blood Type	O	1.3.6.1.4.1.19376.1.5.3.1.3.28
Pregnancies	O	1.3.6.1.4.1.19376.1.5.3.1.1.5.3.4

Figure 3.8: XPHR Extract Elements

3.2.4 QED Intergration Module

IHE Query for Existing Data Profile (QED) enables query on databanks of various information systems generated for medical information like problems, medications, inoculations. Because PHR systems are one of the fundamental of these databanks, it has great importance for eSaglikKaydim and interoperability devices. Unlike XPHR, it enables transfer of specific information depending on queries, instead of entire PHR record. As shown in Figure 3.9, two actors are specified. Just like XPHR, QED integration module, implements roles of both actors. In Clinical Data Consumer role it converts record requests formatted in PHRDataSource coming from interfaces to QED queries and can forward them to external systems supporting QED and can convert incoming records to CCR and submit them to the platform. As Clinical Data Source it can open patient records registered in the platform to external systems in QED compatible format.

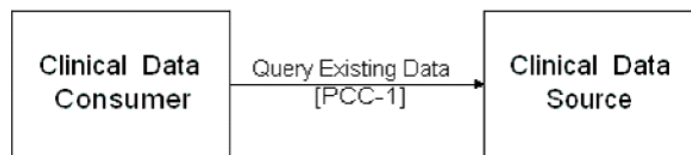


Figure 3.9: QED Actor Diagram

Query parameters specified in QED profile are shown in Figure 3.10. The most important of these parameters, careProvisionCode, indicates which one of patient's records are requested. Codes that should be used for query in careProvisionCode by profile are shown in 3.11

Because these codes do not include all records in the platform, the module is able to retrieve other records in conformance with QED protocol.

Parameter Name	Cardinality	Data Type	Vocabulary Domain	Clinical Data Consumer	Clinical Data Source
careProvisionCode	0..1	CD		O	R
careProvisionReason	0..*	CD		O	O
careRecordTimePeriod	0..1	IVL<TS>		O	R
clinicalStatementTimePeriod	0..1	IVL<TS>		O	R
includeCarePlanAttachment	0..1	BL		R	R
maximumHistoryStatements	0..1	INT		O	R
patientAdministrativeGender	0..1	CE	AdministrativeGender	O	R
patientBirthTime	0..1	TS		O	R
patientId	1..1	II		R	R
patientName	0..1	PN		O	R

Figure 3.10: QED Record Query Parameters

Actor Option	Code	Returns	Template Id
Vital Signs Option	COBSCAT	All Vital Signs	Vital Signs Observation
	Any Code from the Vital Signs Table below	The vital sign identified by the code	Vital Signs Observation
Problems and Allergies Option	MEDCCAT	All problem entries	Problem Entry
	CONDLIST	All Concern Entries	Concern Entry
	PROBLIST	All Problem Concerns	Problem Concern
	INTOLIST	All Allergy Concerns	Allergy and Intolerance Concern
	RISKLIST	All Risks ¹	Concern Entry
Diagnostic Results Option	LABCAT	All Lab Results	Simple Observations
	DICAT	All Imaging Results	Simple Observations
Medications Option	RXCAT	All Medications	Medications
	MEDLIST	All Medications	Medications
	CURMEDLIST	All active medications	Medications
	DISCHMEDLIST	Discharge Medications	Medications
	HISTMEDLIST	All Historical Medications	Medications
Immunizations Option	IMMUCAT	All Immunizations	Immunizations
Professional Services Option	PSVCCAT	All professional service entries	Encounters Procedures Entry

Figure 3.11: QED Record Query Codes

3.2.5 FMIS Intergration Module

Family Medicine Information System (FMIS) is used by family practice doctors after Family Medicine Information System was put into practice in Turkey and is a system which collects some basic patient records in central place. FMIS central system connects with interfaces used by family doctors via web services. Within this framework, web service operations are available intended for management functions performed by family doctors. Operations *SonMuayeneListele* and *HastaKarti*, which are important for integration, are processes that provide patient records and basic information. Integration of platform with FMIS is implemented by developing client for these two services. These clients are packed as *PHRDataSource* and take place within interoperability infrastructure. Records retrieved from FMIS by integration are listed below:

- Procedures /Operations
- Immunizations
- Problems
- Test Results
- Medications
- Pregnancy Information

FMIS provides special data structure to system for record contents. This data structure is coded with XML and used within web service. This data structure is processed by means of developed module and obtained values can be converted to CCR model.

3.2.6 NHIS Intergration Module

National Health Information System is secure, integrated, fast and expandable information and communication platform which collects every type of data produced in health organisations directly from production place in accordance with standards; aiming to increase efficiency and quality in health services by producing proper information from collected data for all parties. At this moment, 55 million out of 72 million Turkish citizens have their EHR created in NHIS. In integration with NHIS, the following records are obtained from the system:

- Procedures /Operations
- Immunizations
- Problems
- Test Results
- Medications
- Pregnancy Information
- Allergies
- Encounters

NHIS provides special data structure for record contents. This data structure is programmed using XML and used within web service. By means of developed module, this data structure is processed and obtained values can be converted to CCR model.

3.2.7 Consent Editor Module

This module provides an environment for patients enabling them to create access control rules and organize their privacy plans. In addition, this module can configure identity information models to define access control rules based on different subjects. Besides, this module can be configured for the awareness of users and their private information. Finally, it can construct rules in XACML [30] which is the most common access control language. (Figure A.7)

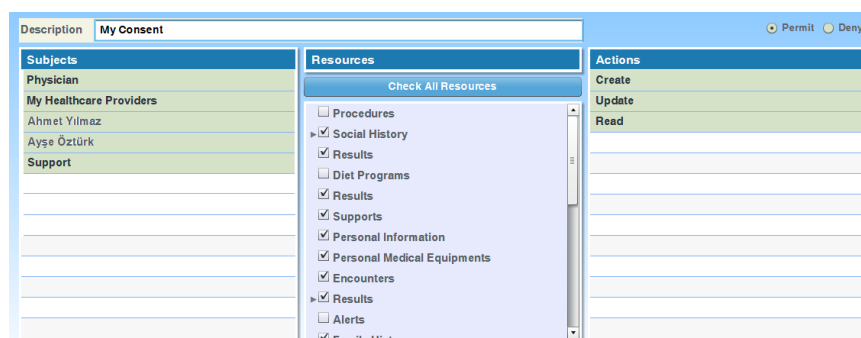


Figure 3.12: Consent Editor Rule Creation

Consent Editor module has a three-tiered architecture, each executes separately handling different jobs. The presentation tier provides graphical user interface for users and the interaction are handled by this tier. Service tier provides services for saving and retrieving users' consent information and rules. Data access tier handles the communication between the Consent Editor and PHR system. This tier also provides a JAVA interface, named **PHRInterface**, which must be implemented to provide consent related service. This interface has methods that must be implemented according to the needs of the system.



Figure 3.13: Consent Editor Architecture

The overall architecture of Consent Editor Module can be seen in Figure 3.13. Client part represents the presentation tier and the server part represents both service and data access tiers. The PHRDatabase is the database where all the information related to a patient is stored.

The service and data access tiers can be seen in detail in Figure 3.14. The Consent Editor Service provides services to the presentation tier as mentioned before. The PHRInterface represents the data access tier and it defines how data is accessed and varies according to the implementation in the system.

3.2.8 Accu-Chek Glucometer Device Intergration Module

This module enables patients to retrieve blood glucose measurements from Accu-Chek Glucometer device. Information recorded to device is uploaded to system through this module, which checks new measurements. The retrieved XML message from this device is vendor's proprietary format, and when a message is retrieved, it is parsed; the extracted glucose measurement information is converted to CCR format and stored in the repository.

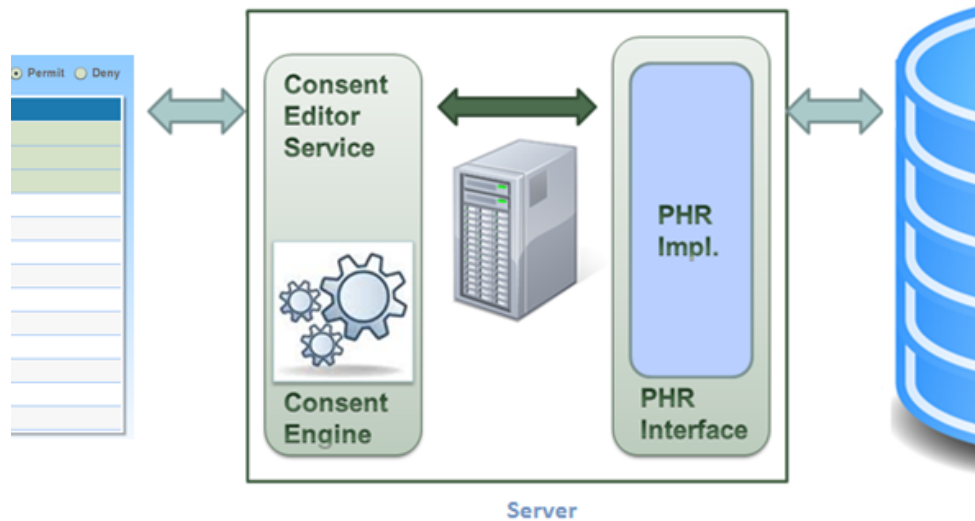


Figure 3.14: Consent Editor Server Architecture

3.2.9 A&D Blood Pressure Device Integration Module

This module enables patients to retrieve blood pressure measurements from A&D Blood Pressure Monitoring device. Information recorded to device is uploaded to system through this module, which checks new measurements. The retrieved message conforms to IEEE11073 standard, and when a message is retrieved, it is parsed according to IEEE11073-20601 device specifications; the extracted blood pressure measurement information is converted to CCR format and stored in the repository.

3.2.10 Health Modules

Last but not least, this last section describes the general structure and execution of health modules. As mentioned in the previous sections, eSaglikKaydim's most important feature is having modular health modules (Section 3.1.2.3). Based on CCR model, processed health data transferred back and forth from/to the OSGI server. To describe execution of a health module, **Problems** health module will be explained as an example in the rest of this section.

When a health module is first loaded - in our case Problems health module, it immediately calls **getRecords** remote method to retrieve CCR records from server. Here, the request sent

by the health module is delivered by BlazeDS Connector module to corresponding record service to be handled. The record service retrieves all "Problem" records from database and returns them to the Flex side and Flex side lists all the records as in Figure 3.15.

The screenshot shows a patient portal for NİLÜFER AYMAZ. The main content area is titled "Conditions" and contains a table of medical records. The table has columns for Name, Start Date, End Date, Status, and Source. The Source column is highlighted with a red box, and each entry in this column contains a green circular icon with the text "USBS".

Name	Start Date	End Date	Status	Source
Nocturia and Poliuria (1)	22/05/2009		Intermittent	USBS
Acute upper respiratory tract infection (1)	31/01/2009		Active	USBS
Type II Diabetes Mellitus (1)	22/05/2009		Intermittent	USBS
Type II Diabetes Mellitus (1)	26/11/2009		Intermittent	USBS
Pregnancy, confirmed (1)	03/05/2010		Active	USBS
Diabetes Mellitus in Pregnancy (1)	03/05/2010		Active	USBS
Anemia (1)	03/05/2010		Active	USBS
Type II Diabetes Mellitus (1)	19/08/2010		Intermittent	USBS
Anemia (1)	19/08/2010		Active	USBS
Anemia (1)	24/11/2010		Active	USBS
Cesarean Delivery (1)	25/11/2010		Active	USBS
Neck ache (1)	25/11/2010		Intermittent	USBS
Type II Diabetes Mellitus (1)	13/12/2010		Intermittent	USBS
Irregular menstruation (1)	05/01/2011		Suspected	USBS
Anemia, other (1)	18/04/2011		Intermittent	USBS
Migraine with aura (1)	18/04/2011		Recurrent	USBS

Figure 3.15: getRecords result

Records may have properties such as status, start and end date, and etc. and these properties are also provided in a list format. Another important property is the *information source* and they are displayed as small icons at the end of each record. Information source specifies the source or author of the information and it can be patient himself/herself, a healthcare professional, external health information systems or medical devices. Since records that are added by patients may not be reliable, information source concept helps doctors to check the *reliability* of the health data.

As it can be seen in Figure 3.16, in eSağlıkKaydım, following information source types are used:

1. **Patient Source:** Information shown with this icon are collected directly from the patient.

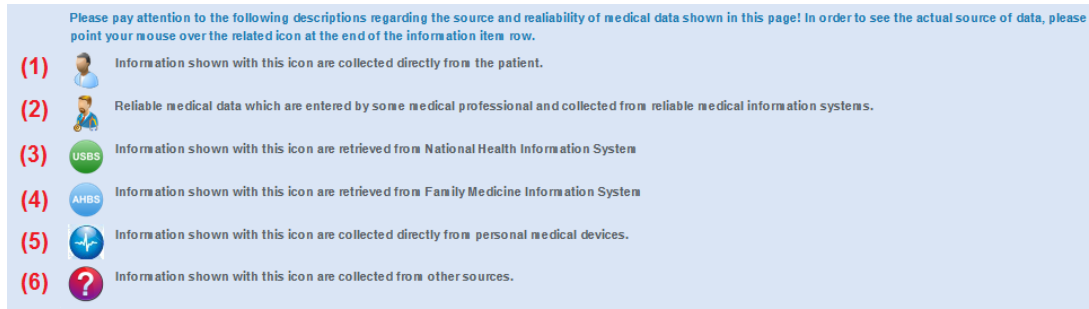


Figure 3.16: Information Source Types

2. **Health Professional Source:** Indicates reliable medical data which are entered by a medical professional.
3. **NHIS Source:** Information shown with this icon are retrieved from National Health Information System.
4. **FMIS Source:** Information shown with this icon are retrieved from Family Medicine Information System.
5. **Device Source:** Information shown with this icon are collected from personal medical devices.
6. **Unknown Source:** Information shown with this icon are collected from other sources.

By clicking on record, detailed information related to that record is displayed. Here users can edit or delete their records, or comments on them (Figure 3.17).

To add new problems, user should click on *Add Conditions* button. After that a pop up windows is displayed requesting necessary information from user as in Figure 3.18. In this window, users must fill in required fields such as problem name, status and start and end dates. All these information are coded with medical terminologies in order to be interoperable with external systems. As the user selects an item from a menu or a list, code information of the corresponding field is entered immediately.

The important thing is here that, the problem name must be selected from a medical terminology list provided by eSaglikKaydim. This list is displayed by clicking on *Browse problem names...* as it can be seen in Figure 3.19.

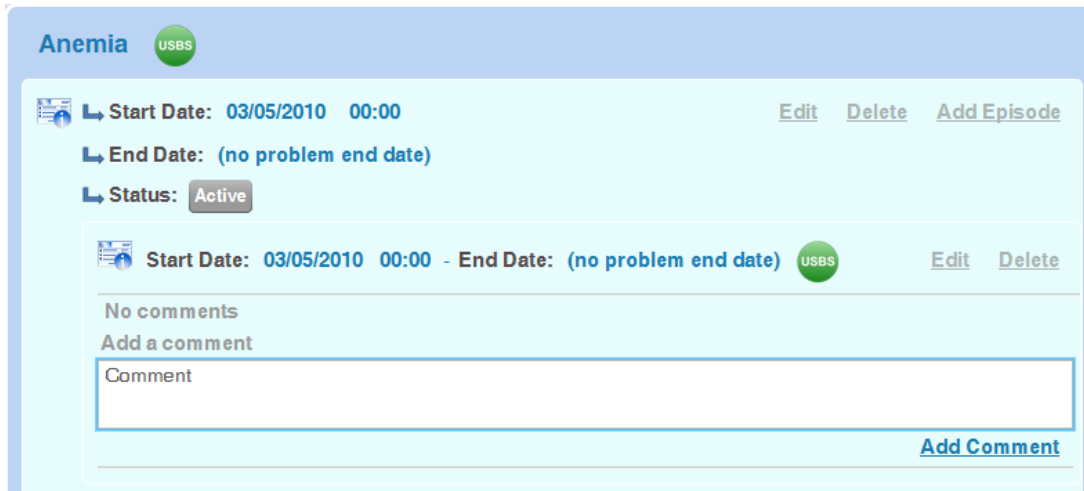


Figure 3.17: Details of a record

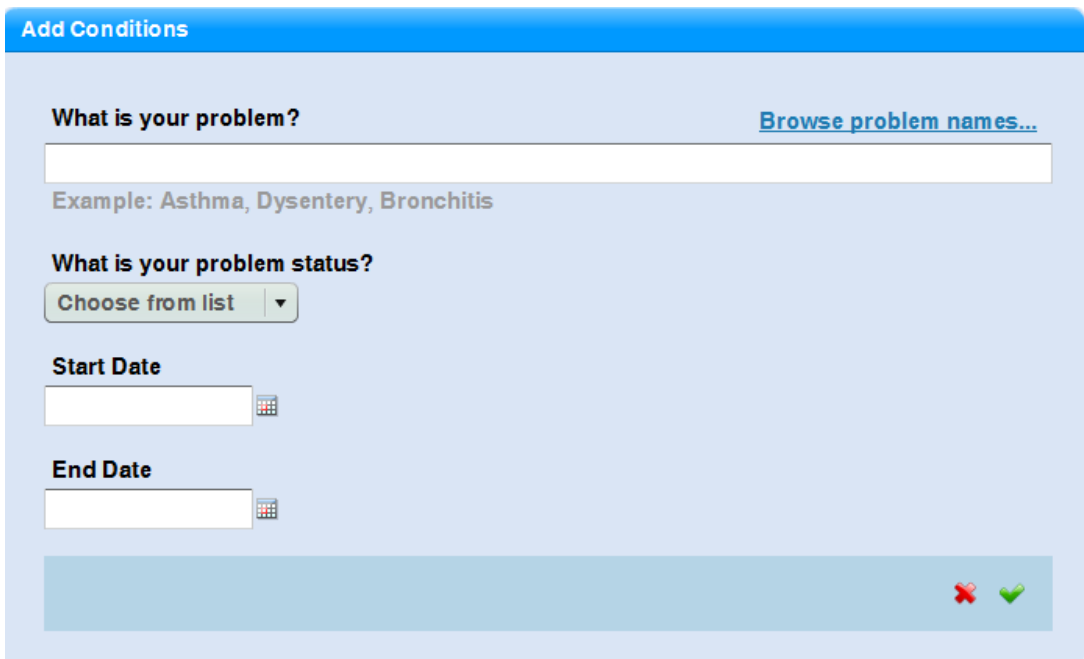


Figure 3.18: A pop up windows to add new problems

Problem Names		
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z <input type="text"/>		
Conditions		
Aarskog Syndrome		+
Aase Syndrome		+
Abdominal Aortic Aneurysm (AAA)		+
Abducens Nerve Palsy		+
Abnormal EKG		+
Abnormal menstrual periods		+
Abnormal Pap Smear		+
Abnormal Pap Smear Results		+
Abnormal Uterine Bleeding		+
Abnormal Vaginal Bleeding		+
ABO Incompatibility		+
Abscess, anorectal		+
abscess, peritonsillar		+
Abscess, skin		+
Acançosis Nigricans		+
Acetabular Labrum Tear		+
Achilles Tendinitis		+

Figure 3.19: List of Problem Terminologies

CHAPTER 4

USE CASE: INTEGRATION OF ESAGLIKKAYDIM WITH NATIONAL HEALTH INFORMATION SYSTEM AND DEMONSTRATION OF A SCENARIO

In this section, the integration of eSaglikKaydim with Turkish National Health Information System (NHIS) [37] will be described with a use case scenario. NHIS of Turkey became operational on January 15, 2009. As of February 2012, 98% of the public hospitals and 60% of the university hospitals and private hospitals are connected to the NHIS sending the EHRs of their patients daily. The total number of connected healthcare organizations is around 1200 and currently there are more than 650 million EHRs in total. The Family Medicine Information System (FMIS) [38], on the other hand, is another national infrastructure for the exchange of primary care records among general practitioners and the Ministry of Health. Both systems have service based architectures implementing Web services to store and retrieve EHRs.

eSaglikKaydim is a modular and flexible PHR system allowing access to personal health records of a patient with proper security mechanisms in place. It provides an environment for patients to keep track of their medical health records as it can be integrated with NHIS and FMIS. This section describes the pregnancy process of a diabetes patient, whose data is retrieved from NHIS and integrated into eSaglikKaydim, in detail. For the sake of privacy and secrecy of the private health data, the identity of the patient is changed.

The main character of this scenario is a female patient, Nilufer Aymaz who was born in 1983, married and lives in Buca, Izmir. She frequently visits Izmir Buca Seyfi Demirsoy State Hospital and Izmir Buca Obstetrics and Gynecology Hospital. At the end of this section, screen shots from eSaglikKaydim is provided in the Appendix section so that the process can

be tracked better.

4.1 Diabetes History

Mrs. Aymaz has diabetes history in her ancestors. There appear a number of infections/wounds in her body and she visits the bathroom very frequently. Therefore, she goes to her family doctor and the family doctor refers her to Izmir Buca Seyfi Demirsoy State Hospital (Figure A.1) with the Nocturia and Poliuria diagnosis in May 2009 (Figure A.2).

In Izmir Buca Seyfi Demirsoy State Hospital, the patient is admitted to the clinic of Endocrinology and Metabolism and in the clinic "Fasting Blood Sugar Test" and "Post-prandial Blood Sugar Test" are performed (Figure A.3). According to the results, on May 22nd 2009, she was diagnosed as Type II Diabetes Mellitus (Figure A.2). In her treatment, the endocrinologist doctor Ahmet Yilmaz recommends drug treatment that involves two types of oral anti-diabetic to the patient and the prescriptions are as follows (Figure A.5):

- DIAMICRON MR 60 MG 60 TABLET, twice a day on an empty stomach
- DIAFORMIN 1000 MG 100 TABLET, twice a day on a full stomach

In addition to that the doctor recommends Self-Monitoring of Blood Glucose (SMBG) by the patient and for this monitoring, prescribes the following glucometer device (Figure A.6):

- ACCU-CHEK Active meter (Model GC)
- ACCU-CHEK SENSOR COMFORT GLUCOSE TEST STRIP

The doctor recommends the patient to measure her blood sugar twice before the meals (fasting glucose level) and enter the values to eSaglikKaydim PHR System. She is also told to enter medication compliance to the PHR system. Furthermore, the doctor states that in order to enhance the stability of blood glucose level, the patient needs to follow a diet and therefore she is referred to the dietician. Right after the encounter and before the dietician visit, the accompanying nurse introduces the eSaglikKaydim PHR System and how to use and upload the values from the glucometer device to the PHR system automatically to Mrs. Aymaz

(Figure A.10). Mrs. Aymaz registers to the PHR system and gives consent to the Dr. Yilmaz and referred dietician to reach her PHR data with the Consent Editor Tool (Figure A.7). Additionally, she retrieves her EHR data from the Izmir Buca Seyfi Demirsoy State Hospital automatically with the EHR/PHR Interoperability Tool.

Meanwhile, the Dr. Yilmaz logs in to the PHR system and registers the diabetes mellitus diagnosis for Mrs Aymaz in Diabetes Monitoring module (Figure A.8). Furthermore, the doctor enters the goals, which is to maintain A1C level below 6.5% (Figure A.9) (Glycosylated hemoglobin - also called hemoglobin A1c, HbA1c, A1C, or Hb1c; sometimes also HbA1c - is a form of hemoglobin that is measured primarily to identify the average plasma glucose concentration over prolonged periods of time) [46], and above mentioned treatment activities (together with the prescriptions and medications)

Until the next visit in November 2009, the patient uses the medication regularly, obeys to the diet and therefore her life quality increases back. She also enters blood sugar level and her compliance to the PHR System. Both the doctor and dietician can monitor the executed activities by the patient through the PHR System, as the patient gave her consent to them beforehand.

In November 2009, she visits the hospital and according to the tests, it is observed that everything goes well and all lab results values are within the expected limits. Therefore, the drug treatment continues as before and there are only minor changes in the diet.

4.2 Pregnancy History

At the beginning of 2010, Mrs. Aymaz gets pregnant. After she confirms her pregnancy by personal testing methods, she goes to Izmir Buca Obstetrics and Gynecology Hospital. In this hospital, the nurse in the reception assigns the patient to gynecologist Dr. Ayse Öztürk. Before the encounter, the patient gives consent to Dr. Öztürk with the Consent Editor Tool of the PHR. During the encounter, through the PHR system, Dr. Ayse Öztürk sees that the patient has Type II diabetes and therefore the doctor requests the following tests (Figure A.3):

- Hemoglobin A1c Test
- Kidney Examination

- Fundoscopy (a test that allows a health professional to see inside the fundus of the eye and other structures using an ophthalmoscope - or funduscope. It is done as part of an eye examination and may be done as part of a routine physical examination. It is crucial in determining the health of the retina and the vitreous humor) [47].

According to the test results, it is observed that there are no severe problems and therefore the patient continues with her pregnancy. After that, the patient is informed about the diabetes treatment in the pregnancy. Especially, the possible complications and the frequency of hospital visits are told.

As patients who suffer from diabetes can not take oral antidiabetic medicine, Mrs Aymaz starts to have insulin treatment immediately. Her physician recommends her to use following regular insulin and insulin pen (Figure A.5):

- HUMULIN-R 100 IU/ML 3 ML, twice a day, thirty minutes before meals. 20 IU in the mornings, 10 IU in the mornings

According to the tests the patient is also diagnosed as anemia and therefore the following drug is prescribed (Figure A.5):

- MEGADYN FILM TABLET 30 TB - Once in a day

As a result of these medication and treatment changes, the diet of the patient is also updated and entered to eSaglikKaydim.

16th week

- General Ultrasonography
- Triple test (E3-HCG-AFP) in order to detect down syndrome and neural tube defect (Figure A.12)

18th week

- As the patient is diabetic pregnant a second level full ultrasound test is performed to in case of anomaly risk.

- With Obstetric Colored Doppler US, the blood flow in the blood vessels of the baby is inspected. (Figure A.3)

20th week

- Fetal echocardiography is performed to check whether there is heart and vessel anomalies (Figure A.3)
- The anemia still continues. Mrs Aymaz starts to take a new medicine: GYNO-FERRO SANOL CAPSULE 30 CAP, once a day. (Figure A.5)

24th week

- General Ultrasonography (Figure A.3)
- Tetanus Immunization (Figure A.11)
- During the examinations, her physician notices the high blood glucose level in her blood to the glucometer measurements. Hemoglobin A1c Test is applied again. The results confirm the high amount of glucose in her blood and her insulin dosage is rearranged: 25 IU in the morning, 15 IU in the evenings. (Figure A.12)

30th week

- 3 Dimensional Ultrasonography (Figure A.3)
- According to the examinations the insulin treatment of the patient is determined as effective.

34th week

- As she passes the 32nd week, in addition to ultrasonography, to examine the development of the baby, Non-Stress Test (NST) and Fetal Biometry and Biophysical Profile Scoring tests are performed. (Figure A.4)

37th week

- Towards the end of pregnancy, her complaints increase. At the same time, there is above-normal increase in weight.
- In order to assess the possibility of premature birth, amniocentesis is applied and it is observed that the lungs of the baby are not mature enough. Therefore, premature birth option is abandoned. (Figure A.4)

38th and 39th weeks

- Ultrasonography
- Non-Stress Test (NST)
- Fetal Biometry and Biophysical Profile Scoring (Figure A.4)

39th week

- As a result of increasing contractions, Mrs Aymaz is taken to the hospital very early in the morning (November 25th 2010)
- The doctors decide cesarean delivery. (Figure A.4)
- At 06:23, the patient gives birth to a 3.7kg female baby. (Figure A.14)
- On November 27th, the patient is discharged.

After that birth, the treatment of the patient is re-adjusted to the previous settings, where she uses oral anti-diabetic medications.

CHAPTER 5

RELATED WORK

There are a number of initiatives for developing platforms to handle patients' personal health-care records, but none of them is as advanced and comprehensive as eSaglikKaydim platform. In this section, the details of these platforms will be described. The information related to the PHRs are gathered from a recent survey [3] investigating the major PHR systems.

Salzburg Research PHR system which is developed by Salzburg Research is a PHR system solution developed within the iCARDEA [42] project that focuses on cardiovascular patients. The PHR system is designed as a Patient Empowerment Framework and aims to complement existing PHR systems and Personal Health Applications. The PHR system modular approach provides services and integration features supported by Enterprise Integration Patterns for different purposes, e.g. monitoring and managing personal health data, medication or information material. Since the main focus is on cardiovascular patients, the PHR system is not as comprehensive as eSaglikKaydim.

Another example is ICW's LifeSensor [43] which is an interactive PHR system particularly aimed at payers who can use to pull together patient data from all stakeholders and assure that members get easy access to health data in order to make the most cost-effective health care decisions. It also supports health and wellness programs, by providing interactive chronic disease add-on packages and health risk assessments. However, this system does not allow device integration which is a very crucial need within a PHR system. In addition, the system uses its own data model, so it is not an interoperable solution.

The most similar effort to a part of our work is MyChart [44] which enables patients to review most of the contents of the medical records, except for progress notes. The patients can view

their diagnoses, active medications, allergies, health maintenance schedules, immunizations, test results (including graphical display), radiology results, appointments, and demographics. In many cases, relevant health educational resources are automatically linked to key terms or phrases in the patient’s medical record, such as a diagnosis of diabetes. In addition, patients can communicate with the physician office to request an appointment, request a prescription renewal, update demographic information, update immunization status, or update a health maintenance procedure. Very many healthcare providers across the US such as Cleveland Clinic, Allina Healthcare Group , University of California, Davis , Baylor Clinic are using MyChart.

Another effort about PHR system is Google Health [45] and it allows users to volunteer their health records: either manually or by logging into their accounts at partnered health services providers, into the Google Health system, thereby merging potentially separate health records into one centralized Google Health profile. Volunteered information can include ”health conditions, medications, allergies, and lab results”. Once entered, Google Health uses the information to provide the user with a merged health record, information on conditions, and possible interactions between drugs, conditions, and allergies. However, the only technical standard from the health care industry being implemented by Google Health is the CCR (Continuity of Care Record).

Table 5.1: Comparison with other PHR systems

PHR System	Device Integration	External Source Integration	Interoperability Standards
eSaglikKaydim	Yes	Yes	Yes
Salzburg Research PHR	Yes	No	Yes
LifeSensor	No	Yes	No
MyChart	Yes	Yes	Yes
Google Health	No	Yes	Yes

In the above table, the comparison between eSaglikKaydim and other PHR systems can be seen. As it can be inferred from the table, eSaglikKaydim provides all crucial features that a PHR system must have, such as device integration, external health information integration and interoperability standards.

CHAPTER 6

CONCLUSION AND FUTURE WORK

Today, patient empowerment through personal health record systems is an important rising trend. According to this trend, in order for treatment process to be effective, patient empowerment is very important and therefore patients must be involved in the treatment process.

When this is the case, PHR systems are becoming more common, creating opportunities for the delivery of healthcare services and the promotion of personal health through several services. Linking these systems to external health information systems and medical devices allow patients to access even more and reliable health data from health professionals.

The bottleneck of these systems is yielding high costs for integrating PHR systems with external health information systems and medical devices; and lack of supporting worldwide interoperability standards. eSaglikKaydim addresses this issue by providing a modular architecture enabling developers to easily integrate any information source by just writing necessary adapters. Thanks to the powerful modularity enabled by the OSGI technology, eSaglikKaydim can be integrated with any EHR systems, health information systems, databases and portals containing/carrying medical information and etc.

eSaglikKaydim provides a web based platform to easily create health information with CCR data model by embedding worldwide medical terminology standards. Therefore, each record in the repository is fully interoperable with external systems. Furthermore, by linking eSaglikKaydim with external health information systems, patient health data can be retrieved from these systems.

The contributions of this thesis are as follows:

- A personal health care platform, which is based on a modular infrastructure enabling development and integration of new modules and adapters, was developed. This architecture was supported by worldwide standards and terminologies to enable interoperability with various external information sources.
- Several modules and adapters for integration with external health information systems (NHIS and FMIS) and medical devices were developed.
- APIs for third party developers were implemented so that they can develop their own health applications.
- Due to their design, important technologies can not be used in the OSGI framework directly. Modules for these technologies were developed so that they can be executed within the OSGI environment.

This work was presented to many Turkish Ministry of Health representatives, family doctors and deputy secretary of ministry himself. According to the comments and feedback received, eSaglikKaydim was improved to meet the needs discussed in several meetings. Finally, It is decided to deploy the platform for public usage and necessary arrangements are still ongoing.

The work in this thesis is also being used ongoing European Commission's FP7 projects: FP7-ICT-288209 EMPOWER[12] and FP7-ICT-PSP 297260 PALANTE[13]. Within the scope of these projects, health modules for specific diseases according to certain guidelines are being developed. Thanks to the modular architecture, eSaglikKaydim can be used in many more FP7 ICT projects.

The presented work is supported by the European ITEA and funded by TÜBİTAK TEYDEB Project No: 9080004.

As a future work, the platform will be improved for pilot deployment. Furthermore, new adapters and modules will be developed in order to provide API for mobile platforms such as Android mobile phones or tablets and iOS devices. As technology develops very fast, user interfaces, which are compatible with mobile devices, written in JSF (JAVA Server Faces) [48] or HTML5 [49] will be developed.

REFERENCES

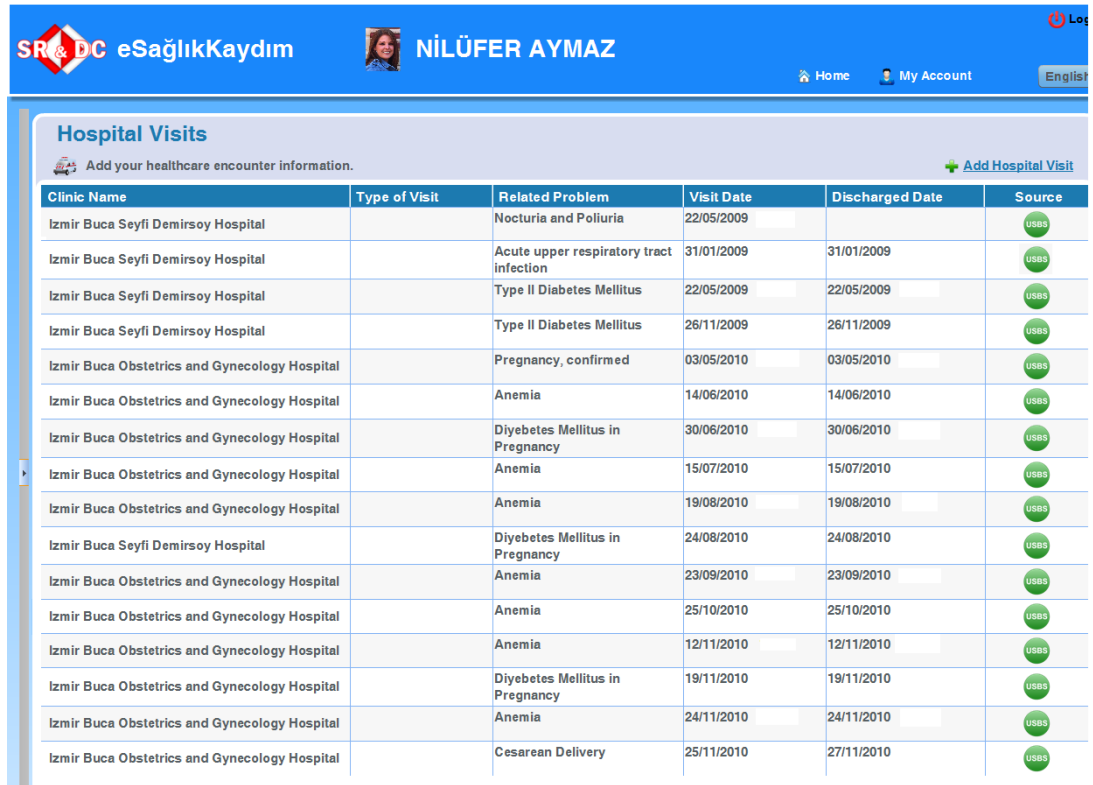
- [1] John D. Halamka, MD, Kenneth D. MANDL, MD, MPH, Paul C. Tang, MD. Early Experiences with Personal Health Records. *Journal of the American Medical Informatics Association*, 15(1), Feb 2008.
- [2] Center for Information Technology Leadership (CITL). http://www.citl.org/publications/_pdf/CITL_PHR_Report.pdf
- [3] Axel Helmer, Myriam Lipprandt, Thomas Frenken, Marco Eichelberg, Andreas Hein. Empowering Patients through Personal Health Records: A Survey of Existing Third-Party Web-Based PHR Products. *electronic Journal of Health Informatics*, 6(3), 2011.
- [4] ASTM E2369 - 05e2 Standard Specification for Continuity of Care Record (CCR). 2005. <http://www.astm.org/Standards/E2369.htm>, last visited on June 2012.
- [5] IHE Profiles. <http://www.ihe.net/profiles/>, last visited on July 2012.
- [6] Continua Health Alliance. <http://www.continuaalliance.org/index.html>, last visited on June 2012.
- [7] ISO/IEEE 11073-10201:2004(E) Health Informatics Point-of-care medical device communication Part 10201: Domain Information Model.
- [8] Query for Existing Data (QED). http://www.ihe.net/Technical_Framework/upload/IHE_PCC_Query_for_Existing_Data_QED_Supplement_TI_2008-08-22.pdf, last visited on June 2012.
- [9] Exchange of Personal Health Record Content Profile. http://wiki.ihe.net/index.php?title=Exchange_of_Personal_Health_Record_Content_Profile, last visited on June 2012.
- [10] Open Source Ambient Intelligence (OSAmI). <http://www.osami-commons.org/>, last visited on June 2012.
- [11] OSAmI-Commons. <http://thewiki4opentech.org/index.php/OSAmI-Commons>, last visited on June 2012.
- [12] Support of Patient Empowerment by an intelligent self-management pathway for patients (EMPOWER). The European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement no 288209. <http://www.empower-fp7.eu/>, last visited on June 2012.
- [13] PATient Leading and mANaging their healThcare through EHealth (PALANTE). the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement no 297260.
- [14] About the OSGi Alliance. <http://www.osgi.org/About/HomePage>, last visited on June 2012.

- [15] Services Gateway Initiative. OSGi Service Platform Specification. Version 3.0, March 2003.
- [16] Creating OSGI Bundles. <http://blog.springsource.com/2008/02/18/creating-osgi-bundles/>, last visited on June 2012.
- [17] Introduction to OSGI. <http://ctpjava.blogspot.com/2010/09/introduction-to-osgi.html>, last visited on June 2012.
- [18] Adobe Learning Resources. Get Oriented to Flex. <http://learn.adobe.com/wiki/display/Flex/Get+oriented+to+Flex>, last visited on June 2012.
- [19] ECMAScript. <http://www.ecmascript.org/>, last visited on June 2012.
- [20] SWF Technology Center. <http://www.adobe.com/devnet/swf.html>, last visited on June 2012.
- [21] BlazeDS Developer Guide. http://livedocs.adobe.com/blazeds/1/blazeds_devguide/, last visited on June 2012.
- [22] Action Message Format – AMF 3. http://download.macromedia.com/pub/labs/amf/amf3_spec_121207.pdf, last visited on July 2012.
- [23] Getting Started with Java Message Service (JMS). <http://java.sun.com/developer/technicalArticles/Ecommerce/jms/>, last visited on June 2012.
- [24] HIBERNATE - Relational Persistence for Idiomatic Java. http://docs.jboss.org/hibernate/orm/3.2/reference/en/html_single/, last visited on June 2012.
- [25] Java SE Technologies - Database. <http://www.oracle.com/technetwork/java/javase/tech/index-jsp-136101.html>, last visited on July 2012.
- [26] ASTM International. <http://www.astm.org/>, last visited on June 2012.
- [27] Kuraitis, V. Untangling the Electronic Health Data Exchange. June 19, 2008. <http://e-caremanagement.com/untangling-the-electronic-health-data-exchange/>, last visited on June 2012.
- [28] OASIS - Advancing Open Standards for the Information Society. <https://www.oasis-open.org/>, last visited on July 2012.
- [29] XML Path Language (XPath). <http://www.w3.org/TR/xpath/>, last visited on July 2012.
- [30] OASIS eXtensible Access Control Markup Language https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml, last visited on July 2012.
- [31] SNOMED Clinical Terms® (SNOMED CT®). http://www.nlm.nih.gov/research/umls/Snomed/snomed_main.html, last visited on June 2012.
- [32] SNOMED Clinical Terms® (SNOMED CT®). <http://www.ihtsdo.org/snomed-ct/>, last visited on June 2012.
- [33] Logical Observation Identifiers Names and Codes (LOINC®). <http://loinc.org/>, last visited on June 2012.

- [34] Logical Observation Identifiers Names and Codes (LOINC®) Users' Guide. <http://loinc.org/downloads/files/LOINCManual.pdf>, last visited on June 2012.
- [35] International Classification of Diseases (ICD-10). <http://www.who.int/classifications/icd/en/>, last visited on June 2012.
- [36] RxNorm. <http://www.nlm.nih.gov/research/umls/rxnorm/>, last visited on June 2012.
- [37] Kose I., Akpınar N., Gurel M., Arslan Y., Ozer H., Yurt N., Kabak Y., Dogac A. Turkey's National Health Information System (NHIS). eChallenges Conference, Stockholm, October 2008, pp. 170-177.
- [38] Family Medicine Information System (FMIS). http://www.ailehekimligi.gov.tr/index.php?option=com_content&view=article&id=760:aile-hekimlii-bilgi-sistemi-ahbs-nedir&catid=88:personel-cin-&Itemid=227, (In Turkish), last visited on June 2012.
- [39] Unified Medical Language System® (UMLS®). <http://www.nlm.nih.gov/research/umls/>, last visited on July 2012.
- [40] CDA® Release 2. http://www.hl7.org/implement/standards/product_brief.cfm?product_id=7, last visited on July 2012.
- [41] JSON. <http://www.json.org/>, last visited on July 2012.
- [42] An Intelligent Platform for Personalized Remote Monitoring of the Cardiac Patients with Electronic Implant Devices (iCARDEA). The European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement no 248240. <http://www.srdc.com.tr/projects/icardea/>, last visited on August 2012.
- [43] LifeSensor. <https://www.lifesensor.com/us/us/us-hn/payers.html>, last visited on June 2012.
- [44] Cleveland Clinic MyChart. <https://mychart.clevelandclinic.org/>, last visited on August 2012.
- [45] Google Health. <http://www.google.com/health/>, last visited on August 2012.
- [46] Diabetes Health Center. <http://diabetes.webmd.com/guide/glycated-hemoglobin-test-hba1c>, last visited on June 2012.
- [47] Fundoscopy - Fundoscopy Eye Exam, Reasons, Procedure to Conduct and Eye Care Post Fundoscopy. <http://www.medicalhealthtests.com/medical-tests/fundoscopy.html>, last visited on June 2012.
- [48] JavaServer Faces Technology. <http://www.oracle.com/technetwork/java/javaserverfaces-139869.html>, last visited on July 2012.
- [49] HTML5. <http://www.w3.org/TR/html5/>, last visited on July 2012.

APPENDIX A

SCREENSHOTS FOR USE CASE DEMO SCENARIO




SR&DC eSağlıkKaydım NİLÜFER AYMAZ Home My Account English

Hospital Visits


Add your healthcare encounter information. [Add Hospital Visit](#)

Clinic Name	Type of Visit	Related Problem	Visit Date	Discharged Date	Source
Izmir Buca Seyfi Demirsoy Hospital		Nocturia and Poliuria	22/05/2009		USBS
Izmir Buca Seyfi Demirsoy Hospital		Acute upper respiratory tract infection	31/01/2009	31/01/2009	USBS
Izmir Buca Seyfi Demirsoy Hospital		Type II Diabetes Mellitus	22/05/2009	22/05/2009	USBS
Izmir Buca Seyfi Demirsoy Hospital		Type II Diabetes Mellitus	26/11/2009	26/11/2009	USBS
Izmir Buca Obstetrics and Gynecology Hospital		Pregnancy, confirmed	03/05/2010	03/05/2010	USBS
Izmir Buca Obstetrics and Gynecology Hospital		Anemia	14/06/2010	14/06/2010	USBS
Izmir Buca Obstetrics and Gynecology Hospital		Diabetes Mellitus in Pregnancy	30/06/2010	30/06/2010	USBS
Izmir Buca Obstetrics and Gynecology Hospital		Anemia	15/07/2010	15/07/2010	USBS
Izmir Buca Obstetrics and Gynecology Hospital		Anemia	19/08/2010	19/08/2010	USBS
Izmir Buca Seyfi Demirsoy Hospital		Diabetes Mellitus in Pregnancy	24/08/2010	24/08/2010	USBS
Izmir Buca Obstetrics and Gynecology Hospital		Anemia	23/09/2010	23/09/2010	USBS
Izmir Buca Obstetrics and Gynecology Hospital		Anemia	25/10/2010	25/10/2010	USBS
Izmir Buca Obstetrics and Gynecology Hospital		Anemia	12/11/2010	12/11/2010	USBS
Izmir Buca Obstetrics and Gynecology Hospital		Diabetes Mellitus in Pregnancy	19/11/2010	19/11/2010	USBS
Izmir Buca Obstetrics and Gynecology Hospital		Anemia	24/11/2010	24/11/2010	USBS
Izmir Buca Obstetrics and Gynecology Hospital		Cesarean Delivery	25/11/2010	27/11/2010	USBS

Figure A.1: Hospital Visit Records of Mrs. Aymaz

SIR&DC eSağlıkKaydım  **NİLÜFER AYMAZ** Logout

Home My Account English



Patient: **NİLÜFER AYMAZ**
 B. Date: **1983-05-23**
 Tel: **02324482799**
 Email: **niluferaymaz@gmail.com**

- Account Management
- Disease Management
- Personal Information
- Health Record
 - Tests
 - Basic Health Information
 - Hospital Visits
 - Medical Equipments
 - Procedures
 - Medications
 - epSOS
 - Allergies
 - Immunizations
 - Conditions
- My Health Center

Conditions

Add your clinic problems, conditions, diagnoses, symptoms, findings, and complaints. [Add Conditions](#)

Name	Start Date	End Date	Status	Source
Nocturia and Poliuria (1)	22/05/2009		Intermittent	USBS
Acute upper respiratory tract infection (1)	31/01/2009		Active	USBS
Type II Diabetes Mellitus (1)	22/05/2009		Intermittent	USBS
Type II Diabetes Mellitus (1)	26/11/2009		Intermittent	USBS
Pregnancy, confirmed (1)	03/05/2010		Active	USBS
Diyabetes Mellitus in Pregnancy (1)	03/05/2010		Active	USBS
Anemia (1)	03/05/2010		Active	USBS
Type II Diabetes Mellitus (1)	19/08/2010		Intermittent	USBS
Anemia (1)	19/08/2010		Active	USBS
Anemia (1)	24/11/2010		Active	USBS
Cesarean Delivery (1)	25/11/2010		Active	USBS
Neck ache (1)	25/11/2010		Intermittent	USBS
Type II Diabetes Mellitus (1)	13/12/2010		Intermittent	USBS
Irregular menstruation (1)	05/01/2011		Suspected	USBS
Anemia, other (1)	18/04/2011		Intermittent	USBS
Migraine with aura (1)	18/04/2011		Recurrent	USBS


Figure A.2: Problem and Diagnosis Records of Mrs. Aymaz

SR&DC eSağlıkKaydım		NİLÜFER AYMAZ		Logout
		Home	My Account	English
Ultrasonography		23/09/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
3D Ultrasonography		23/09/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Glucose Test		24/08/2010	Izmir Buca Seyfi Demirsoy Hospital	USBS
Ultrasonography		19/08/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Glucose Tolerance Test		19/08/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Complete blood count		19/08/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Fetal echocardiography		15/07/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Obstetric Ultrasonography		15/07/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Obstetric Colored Doppler US		30/06/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
3D Ultrasonography		30/06/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Triple test (E3-HCG-AFP)		14/06/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Ultrasonography		14/06/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Kidney Examination		03/05/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
HbA1C Test		03/05/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Fundoscopy		03/05/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Obstetric Ultrasonography		03/05/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
HbA1C Test		26/11/2009	Izmir Buca Seyfi Demirsoy Hospital	USBS
Post-prandial Blood Sugar Test		26/11/2009	Izmir Buca Seyfi Demirsoy Hospital	USBS
HbA1C Test		22/05/2009	Izmir Buca Seyfi Demirsoy Hospital	USBS
Post-prandial Blood Sugar Test		22/05/2009	Izmir Buca Seyfi Demirsoy Hospital	USBS


Figure A.3: Procedure Records of Mrs. Aymaz

SR&DC eSağlıkKaydım		NİLÜFER AYMAZ		Logout
		Home	My Account	English
Cesarean Delivery		25/11/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Internal jugular subklavian		25/11/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Non-Stress Test		25/11/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Ultrasonography		25/11/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Fetal biometry and biophysical profile scoring		24/11/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Non-Stress Test		24/11/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Ultrasonography		24/11/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Non-Stress Test		19/11/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Ultrasonography		19/11/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Amniocentesis		12/11/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Fetal biometry and biophysical profile scoring		12/11/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Ultrasonography		12/11/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Non-Stress Test		12/11/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Post-prandial Blood Sugar Test		12/11/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Fetal biometry and biophysical profile scoring		25/10/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Non-Stress Test		25/10/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
Ultrasonography		25/10/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS
3D Ultrasonography		23/09/2010	Izmir Buca Obstetrics and Gynecology Hospital	USBS

Figure A.4: Procedure Records of Mrs. Aymaz (cont.)

SR&DC eSağlıkKaydım  NİLÜFER AYMAZ Logout

Home My Account English














Patient: NİLÜFER AYMAZ
B. Date: 1983-05-23
Tel: 02324482799
Email: niluferaymaz@gmail.com

- Account Management
- Disease Management
- Personal Information
- Health Record
 - Tests
 - Basic Health Information
 - Hospital Visits
 - Medical Equipments
 - Procedures
 - Medications
 - epSOS
 - Allergies
 - Immunizations
 - Conditions
- My Health Center

Medications

Add your medication information. [Add Medications](#)

Name	Start Date	End	Route	Frequency	Strength	Dose	Source
AUGMENTIN BID 625 MG 14 FILM TABLETS	31/01/2009		Oral	Once a day		1	
DIAMICRON MR 60 MG 60 TABLETS	22/05/2009		Oral	Twice a day		1	
DIAFORMIN 1000 MG 100 FILM TABLETS	22/05/2009		Oral	Twice a day		1	
DIAMICRON MR 60 MG 60 TABLETS	26/11/2009		Oral	Twice a day		1	
DIAFORMIN 1000 MG 100 FILM TABLETS	26/11/2009		Oral	Twice a day		1	
HUMULIN-R 100 IU/ML 3 ML 5 CARTRIGE	03/05/2010		Injection	Twice a day		1	
MEGADYN FILM TABLET 30 TB	03/05/2010		Oral	Once a day		1	
GYNO-FERRO SANOL CAPSULE 30 CAP	15/07/2010		Oral	Once a day		1	
HUMULIN-R 100 IU/ML 3 ML 5 CARTRIGE	24/08/2010		Injection	Twice a day		1	
DIAMICRON MR 60 MG 60 TABLETS	13/12/2010		Oral	Twice a day		1	
DIAFORMIN 1000 MG 100 FILM TABLETS	13/12/2010		Oral	Twice a day		1	

Descriptions

Please pay attention to the following descriptions regarding the source and reliability of medical data shown in this page! In order to see the actual source of data, please point your mouse over the related icon at the end of the information item row.






-  Information shown with this icon are collected directly from the patient.
-  Reliable medical data which are entered by some medical professional and collected from reliable medical information systems.
-  Information shown with this icon are retrieved from National Health Information System
-  Information shown with this icon are retrieved from Family Medicine Information System



Figure A.5: Medication Records of Mrs. Aymaz

SR&DC eSağlıkKaydım  NİLÜFER AYMAZ Logout

Home My Account English

Medical Equipments

Add your medical equipment information [Add Medical Equipment](#)

Name	Start Date	End Date	Status	Source
ACCU-CHEK Active meter (Model GC)	22/05/2009			
ACCU-CHEK SENSOR COMFORT GLUCOSE Blood Sugar Measurement Strips	22/05/2009			

Descriptions

Please pay attention to the following descriptions regarding the source and reliability of medical data shown in this page! In order to see the actual source of data, please point your mouse over the related icon at the end of the information item row.








-  Information shown with this icon are collected directly from the patient.
-  Reliable medical data which are entered by some medical professional and collected from reliable medical information systems.
-  Information shown with this icon are retrieved from a National Health Information System
-  Information shown with this icon are retrieved from a Family Medicine Information System
-  Information shown with this icon are collected directly from personal medical devices.
-  Information shown with this icon are collected from other sources.

Figure A.6: Medical Equipment Records of Mrs. Aymaz

SR & DC eSağlıkKaydım  NİLÜFER AYMAZ Logout

Home My Account English

MENU


Consent Management Editor

Description: **My Consent** Permit Deny

Subjects	Resources	Actions
Physician My Healthcare Providers Ahmet Yılmaz Ayşe Öztürk Support	<input type="checkbox"/> Procedures <input checked="" type="checkbox"/> Social History <input checked="" type="checkbox"/> Results <input type="checkbox"/> Diet Programs <input checked="" type="checkbox"/> Results <input checked="" type="checkbox"/> Supports <input checked="" type="checkbox"/> Personal Information <input checked="" type="checkbox"/> Personal Medical Equipments <input checked="" type="checkbox"/> Encounters <input checked="" type="checkbox"/> Results <input type="checkbox"/> Alerts <input checked="" type="checkbox"/> ...	Create Update Read

Add new conditions or obligations for access rules

Figure A.7: Consent Rules of Mrs. Aymaz


SR&DC eSağlıkKaydım  NİLÜFER AYMAZ Logout

Home My Account English


Diabetes Management My Care Plan My Diabet Profile Decision Supporters Help

Diabet Calendar My Current Diabetes Care Plan

Care Plan Del



Instructions



Diagnosis

Diagnosis for Diabetes:

Diagnosis Date:

Hypertension: Do you have hypertension problem?

Eye Problem: Do you have an eye problem?

Diagnosis List:

Type	Diagnosis	Date	Remove
2	Diabetes Melitus(2)	2009-05-22	✖

Care Plan Practitioners:

Name	Surname	Clinic	Remove
Ahmet	Yılmaz		✖

Care Plan for Diabetes Management

Type 1 Diabetes

Type 2 Diabetes

In type 2 diabetes, either the body does not produce enough insulin or the cells ignore the insulin. Insulin is necessary for the body to be able to use glucose for energy. When you eat food, the body breaks down all of the sugars and starches into glucose, which is the basic fuel for the cells in the body. Insulin takes the sugar from the blood into the cells. When glucose builds up in the blood instead of going into cells, it can lead to diabetes complications.

Hypertension & Diabetes

Figure A.8: Diabetes Diagnosis of Mrs. Aymaz

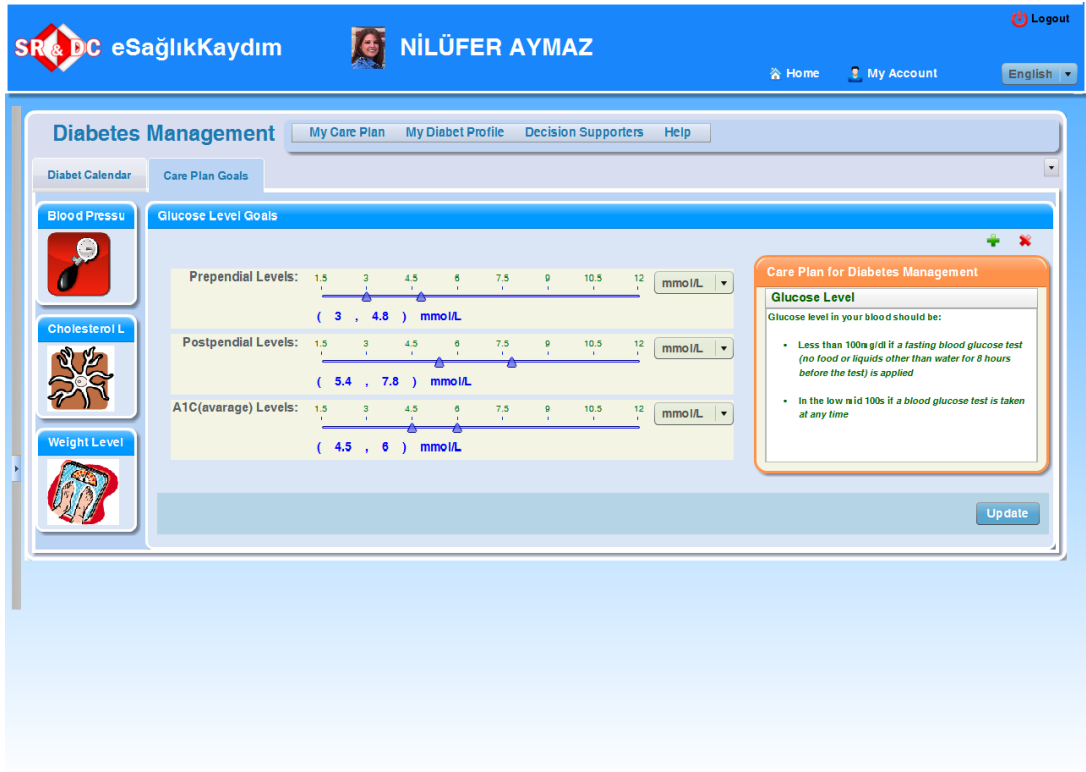


Figure A.9: Diabetes Goals of Mrs. Aymaz

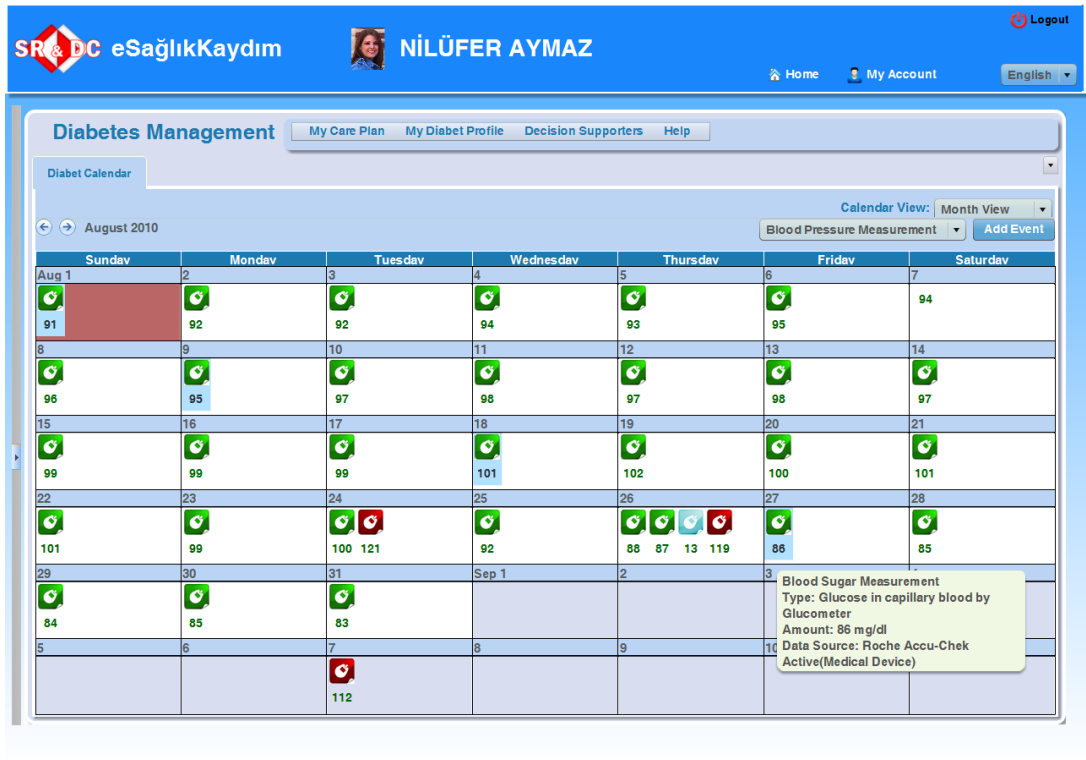





Figure A.10: Diabetes Calendar and Glucose Measurements of Mrs. Aymaz

SR & DC eSağlıkKaydım  NİLÜFER AYMAZ Logout

[Home](#) [My Account](#) English

Immunizations

 Add your immunization information. [Calendar View](#) [Add Immunization](#)

Name	Date	Source
Tetanus Immunization	19/08/2010	

Descriptions

Please pay attention to the following descriptions regarding the source and reliability of medical data shown in this page! In order to see the actual source of data, please point your mouse over the related icon at the end of the information item now.









-  Information shown with this icon are collected directly from the patient.
-  Reliable medical data which are entered by some medical professional and collected from a reliable medical information systems.
-  Information shown with this icon are retrieved from a National Health Information System
-  Information shown with this icon are retrieved from a Family Medicine Information System
-  Information shown with this icon are collected directly from a personal medical devices.
-  Information shown with this icon are collected from other sources.

Figure A.11: Immunization Records of Mrs. Aymaz

SR & DC eSağlıkKaydım  **NİLÜFER AYMAZ** Logout

Home My Account English



Patient: **NİLÜFER AYMAZ**
 B. Date: **1983-05-23**
 Tel: **02324482799**
 Email: **niluferaymaz@gmail.com**

Account Management
 Disease Management
 Personal Information

Health Record

- Tests
- Basic Health Information
- Hospital Visits
- Medical Equipments
- Procedures
- Medications
- epSOS
- Allergies
- Immunizations
- Conditions

Tests

Add your laboratory, diagnostic, and therapeutic results. Add Test

Test Name	Last Test Information
Urinalysis (1)	25/11/2010
Obstetric Ultrasonography (2)	1644 - 15/07/2010
Glucose Test (1)	109 - 24/08/2010
Glucose Test (5)	93 - 18/04/2011
Cholesterol (2)	215 - 18/04/2011
Prothrombin time (1)	12 - 25/11/2010
Fetal biometry and biophysical profile scoring (3)	24/11/2010
Triple test (E3-HCG-AFP) (1)	1.21-0.44-1.32 - 14/06/2010
HDL Cholesterol (2)	58 - 18/04/2011
LDL Cholesterol (2)	147 - 18/04/2011
3D Ultrasonography (2)	23/09/2010
Blood Urea Nitrogen (BUN) (1)	13/12/2010
Post-prandial Blood Sugar Test (3)	129 - 12/11/2010
Non-Stress Test (5)	25/11/2010
APTT (1)	29 - 25/11/2010
HbA1C Test (5)	131 - 13/12/2010
Triglycerid (2)	167 - 18/04/2011
Glucose Tolerance Test (1)	111 - 19/08/2010
Ultrasonography (8)	05/01/2011
Complete blood count (4)	13,0 - 05/01/2011

Figure A.12: Tests and Laboratory Results of Mrs. Aymaz

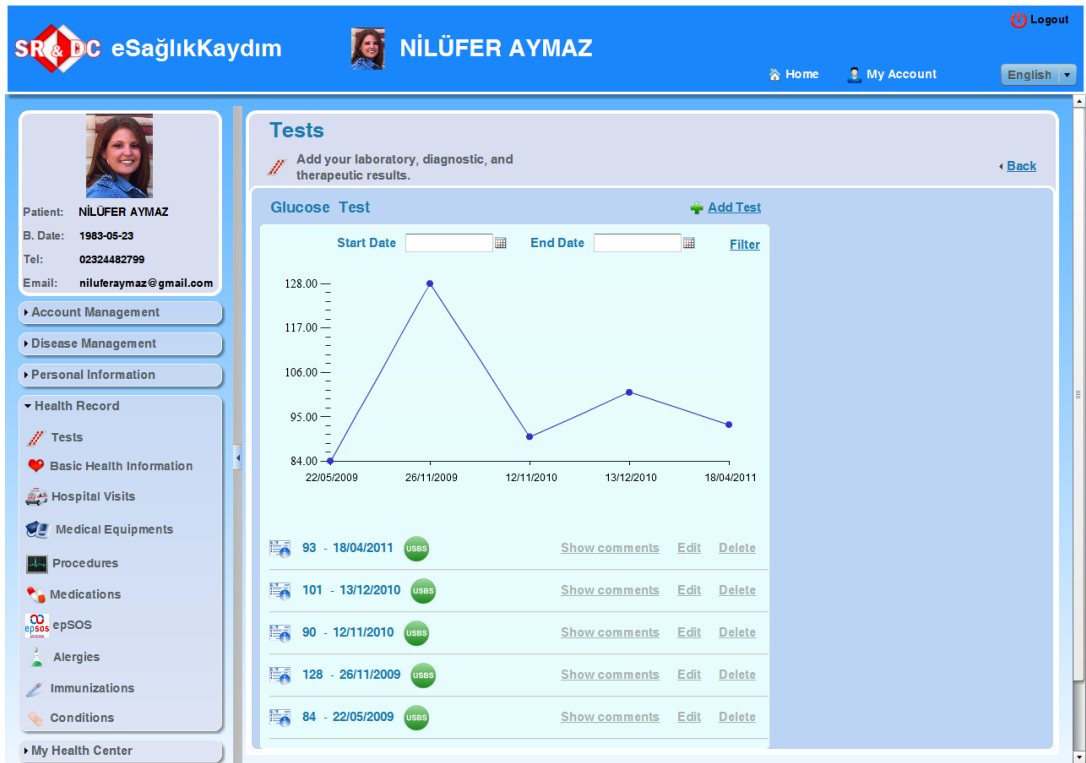




Figure A.13: Graphical Analysis of Glucose Tests of Mrs. Aymaz

SR&DC eSağlıkKaydım  NİLÜFER AYMAZ Logout

[Home](#) [My Account](#) English

Pregnancy Monitoring

[Pregnancy Monitoring](#) [Pregnancy Analysis](#)
[Pregnancy Calendar](#)
[New Pregnancy Record](#)

Start Date	End Date	Source
25/02/2010	25/11/2010	

Descriptions

Please pay attention to the following descriptions regarding the source and reliability of medical data shown in this page! In order to see the actual source of data, please point your mouse over the related icon at the end of the information item row.







-  Information shown with this icon are collected directly from the patient.
-  Reliable medical data which are entered by some medical professional and collected from reliable medical information systems.
-  Information shown with this icon are retrieved from National Health Information System
-  Information shown with this icon are retrieved from Family Medicine Information System
-  Information shown with this icon are collected directly from personal medical devices.
-  Information shown with this icon are collected from other sources.

Figure A.14: Pregnancy Record of Mrs. Aymaz