

**IMPLEMENTING AN E-HEALTH CARE SERVICE ORIENTED ARCHITECTURE  
USING YII FRAMEWORK**

**By**

**Lotino, Wilfred John Z.**

**Lua, Marc Oliver Y.**

**Sina-on, Razelle Jhan D.**

**SCHOOL OF ARTS AND SCIENCES**

**ATENEO DE DAVAO UNIVERSITY**

**MARCH 2013**

## **ABSTRACT**

The advancement of E-Health care applications has enabled the proponents to develop an E-Health Care Service Oriented Architecture that could be used by hospitals as well as doctor's clinics. Since E-health care has a very wide scope, the proponents conducted surveys in hospitals to find out the most essential features of an E-health care application. The proponents also conducted an Object Oriented Analysis and design in hospitals to develop the database design of the application. Using the Yii Framework's Object-Relational Mapping, the proponents implemented the database design. The proponents were able to develop an E-health care application prototype and multiple hospitals could avail the application as a service online rather than setting up their own servers.

***Keywords:***

Yii Framework, Twitter Bootstrap, E-Health Care

## **CHAPTER 1**

### **INTRODUCTION**

#### **1. Background of the Study**

The internet has already affected the lives of every individual from teenagers to old people. E-health care is one of the areas of the health care industry which has used the internet to deliver health information to patients and reduce the need for hospital care. A lot of countries are already adapting E-healthcare. In hospital information systems, medical records are only available to the hospital itself. Doctors could not see what is happening to their patient in the hospital and despite their busy schedule they would need to go to the hospital to check their patients. Through service-oriented architecture it is easier for doctors to monitor their patients. Most hospitals in Davao City use hospitals systems with servers to be administered. Updates to the system are done manually by hired or contracted programmers. In a service-oriented architecture system updates are done without the need of a system administrator and health system would benefit a lot from it since systems would have less time in upgrades. The proponents would like find out the feasibility of E-Healthcare application in a service-oriented architecture in Davao City.

#### **2. Technology Application Context (IT)**

Medical records are data which should be accurate and up-to date. Nowadays, medical records of patients in the hospital are only updated when the patient is admitted. There are still a lot of things happening to the patient outside the hospital such as consultations in clinics and examinations in medical laboratories. Software as a service (SaaS) is a category of cloud computing services that provides the users of an “on-demand software” that could be accessed through a web browser or other devices that has access to the internet. Using this architecture, we could enable the collaboration of clinics

with certain hospitals. E-healthcare delivered in a service-oriented architecture would enable data to be centrally hosted on a server provided by the service provider which different users such as doctors could access and update. The security of the application would be implemented by accounts given to users where they are only the ones who could access the system. The Yii Framework technology would be used by the proponents to develop the functionalities and modules of the application.

### **3. Objective of the Study**

The general objective of this study is to provide a service-oriented application for hospitals that would enable the collaboration with medical practitioners (doctor, nurse) and medical establishments (clinics).

- To identify the most essential E-Health Care Services (patient admission, scheduling, laboratory, medical history etc.) and create an application that is in a service-oriented architecture.
- To provide an application that would assist the collaboration of clinics and hospitals such as Doctor's accessing relevant information about a patient that is admitted in the hospital for a better decision making.
- To help doctors manage their time and be informed about the tests done to patients or vital signs taken by nurses even they are not in the hospital.
- To test the effectiveness of e-health care application in a service-oriented architecture in the Davao City.
- To learn how to use Yii Framework as a technology in developing applications.

#### 4. **Significance of the Study**

Hospitals would be able to collaborate with clinics and more time would be given in the treatment of patients rather than encoding and finding out the medical history of the patients. Health records would be consistent between collaborated health establishments. Doctor would manage their time better since they are updated with the medical records of patients even when they are in the clinic. The study would find out the effectiveness of E-Health applications in Davao City.

#### 5. **Scope and Limitation of the Study**

The scope of our project would be doctors and small hospitals in Davao City. The E-Health Service application would only include the medical records, laboratory, consultation, physical exams, appointments and e-prescribing. The application would not cover the Billing and the facilities of the hospitals.

#### 1.6 **Definition of Terms**

- **Cloud computing** - is the delivery of computing as a service rather than a product, whereby shared resources, software, and information are provided to computers and other devices as a metered service over a network (typically the Internet).
- **Health Care Services** – assists the diagnosis, treatment, and prevention of disease, illness, injury, and other physical and mental impairments in humans.
- **Users interface** – is one of the most important parts of any program because it determines how easily you can make the program do what you want.
- **Framework** - basic supporting structure of anything.
- **Database** – large amount of information stored in a computer.
- **Twitter Bootstrap** - a front-end design tool that provides built in widgets for faster web development. It also adapts to mobile phones using its responsive CSS.

- **Model View Controller (MVC)** - a computer software design pattern that separates the representation of information from the user's interaction with it.
- **MySQL** - an open source relational database management system

## **CHAPTER 2**

### **REVIEW OF RELATED LITERATURE AND WORKS**

#### **2.1 Survey of Existing Technologies**

The proponents studied different methods for developing applications in a service-oriented architecture. Instead of having one database for each client, the proponents decided to use relational database in designing the application. Object Oriented Analysis would be used to study the process of hospitals. In an Object Oriented Analysis each object is treated as a class that is related to other classes. OOA is effective in the proposed application since there would be a lot of users and interactions of the users. Through OOA, the proponents would not repeat writing the same code and changes to the application would be implemented easier. E-Health care is also a very broad field and these systems should be maintainable and scalable.

Since the application would be web based, the proponents reviewed different web technologies for developing web applications. The proponents tried using Grails framework which uses the groovy language which is based from Java. The team had no background in developing Java enterprise and Java web application so the team decided to go with PHP which is language that the proponents are more comfortable.

The proponents studied different PHP frameworks and they decided to use Yii Framework. Yii Framework has a very good documentation and has a lot of tools that would speed up the development of the features of the application. Yii Framework is an Object Oriented Programming framework that would implement the object oriented design of the application. The framework provides AJAX based widgets for displaying data in order for less styling. It also provides a code generator for the CRUD functions of a model which is very clean and clear.

#### **2.2 Related Works**

**OpenEHR** - is a Free and Open Source electronic health records and medical practice management application that can run on Windows, Linux, Mac OS X, and many other platforms.

**OpenMRS**- is a collaborative open source project to develop software to support the delivery of health care in developing countries.

**Care2x** - Integrated Hospital Information System built using PHP, mySQL and PostgreSQL.

**GNU Health** - GNU Health is a free Health and Hospital Information System written in Python and using Tryton framework.

**CureMd SMART Cloud** – is a web-based specialty EMR, Practice Management, Patient Portal and billing services.

**Practice Fusion** – is a free web-based Electronic Medical Record (EMR). It has features such as charting, scheduling, e-prescribing (eRx), medical billing, lab integration and referral letters.

**“E-Health Care Applications”**

<b>nEMR</b>	<b>OpenMRS</b>	<b>Care2x</b>
Demographics	<ul style="list-style-type: none"> <li>• Central concept dictionary</li> <li>• Privilege-based access</li> </ul>	<ul style="list-style-type: none"> <li>• Doctor’s Dashboard</li> <li>• E-Prescribing</li> </ul>
Scheduling	<ul style="list-style-type: none"> <li>• Patient repository</li> <li>• Multiple identifiers per patient</li> </ul>	<ul style="list-style-type: none"> <li>• Appointments</li> <li>• Surgery</li> </ul>
Diagnostic Medical	<ul style="list-style-type: none"> <li>• Data entry</li> <li>• Data export</li> <li>• Standards support</li> </ul>	<ul style="list-style-type: none"> <li>• Nursing</li> <li>• Outpatient</li> <li>• Wards</li> </ul>
Integrations	<ul style="list-style-type: none"> <li>• Modular architecture Patient</li> </ul>	<ul style="list-style-type: none"> <li>• Laboratory</li> </ul>



<b>Billing</b>	workflows	• Pharmacy
<b>Decision Rules</b>	• Relationships	• Admission
<b>Portal</b>	• Patient merging	
	• Reporting tools	
	• Person attributes	

<b>GNU Health</b>	<b>CureMD SMART Cloud</b>	<b>Practice Fusion</b>
<ul style="list-style-type: none"> <li>• Strong focus in family medicine and Primary Health Care</li> <li>• Diseases and Medical procedures standards (ICD-10 / ICD-10-PCS)</li> <li>• Prescription writing</li> <li>• Billing</li> <li>• Epidemiological and other statistical reports</li> <li>• Patient Administration</li> <li>• Doctor Administration</li> <li>• Lab Administration</li> <li>• Medicine / Drugs information</li> </ul>	<ul style="list-style-type: none"> <li>• Electronic medical records</li> <li>• Practice management</li> <li>• enterprise scheduling</li> <li>• Intelligent billing</li> <li>• Workflow management</li> <li>• e-prescribing (eRx)</li> <li>• E-labs</li> <li>• Document imaging</li> <li>• Clinical knowledgebase</li> <li>• Clinical alerts</li> <li>• Device connectivity</li> <li>• Patient education</li> <li>• Data mining reports</li> <li>• Patient portal</li> </ul>	<ul style="list-style-type: none"> <li>• Charting</li> <li>• Scheduling</li> <li>• e-prescribing (eRx)</li> <li>• Medical Billing</li> <li>• Lab Integration</li> <li>• Referral Letters</li> </ul>

*Features of Existing E-Healthcare Applications*

The proponents also reviewed existing systems in E-health care in order to find out and understand how e-healthcare systems work. The proponents also studied how the features are designed in the existing systems. The proponents learned that all the existing e-health care systems contain medical records which are the core of every e-healthcare application. Most of these applications are not yet implemented as a service oriented architecture and these applications are given as a product that is to be installed in a server or a personal computer of a user. Some of these systems are implemented in service-oriented architecture but are proprietary. Our proposal is to develop an e-health care application prototype that has a service oriented architecture that is to be tested with different health practitioners in Davao City. The proponents would like to know the most essential features of a hospital system and implement those essential features. The

application would be mainly used by hospitals and the persons in the hospital such as the admission staff, laboratory staff, doctor, nurses and patients.

## **CHAPTER 3**

### **PROJECT DESIGN AND METHODOLOGY**

#### **3.1 Operational Framework**

*Figure 1.Operational Framework*

The application provides electronic health services that were developed using technologies such as Yii Framework and Twitter Bootstrap. This application would be accessible by a user who has a device that has access to the internet. Hospitals are the main users of the application. The hospital administrator could add the hospital staffs,

nurses, doctors, and patients. The doctor could also use the application in his clinic and he could also add an assistant account to assist him in encoding.

## **3.2 Methodology**

### **3.2.1 Identification of E-Health Services**

The features of the E-health care service were identified through surveys from doctors, nurses, patients and hospital staffs such as admission staff and laboratory staff.

### **3.2.2 Creating the Database**

The proponents gathered different medical forms from hospitals. The proponents also interviewed doctors to know the process on how doctors get the medical records of patients. The proponents also conducted an Object Oriented analysis and design on hospitals. Based from the data we have gathered, the proponents design the database of the application.

### **3.2.3 Creating the Web Application**

The application was made with the use of Yii Framework and Twitter Bootstrap. Yii enabled the implementation of the database design using the frameworks' Object-relational mapping and Model-View-Controller pattern. Twitter bootstrap was used to design the views component of the application.

### **3.2.4 Deployment to Cloud Server**

The application was deployed to the Hopkins server of Ateneo de Davao University using SSH and wget.

### **3.2.5 Testing**

The testing was done with different health practitioners in Davao City.

## **CHAPTER 4**

## TECHNOLOGY BACKGROUND

**PHP** is a general-purpose [server-side scripting language](#) originally designed for Web development to produce dynamic Web pages. It is one of the first developed server-side scripting languages to be embedded into an HTML source document rather than calling an external file to process data.

**Yii Framework** is a free, open-source Web application development framework written in PHP5 that promotes clean code and encourages rapid development. It works to streamline your application development and helps to ensure an extremely efficient, extensible, and maintainable end product.

**MySQL** is the world's most used open source relational database management system that runs as a server providing multi-user access to a number of databases.

**Software as a Service** is a type of cloud computing that delivers a single application through the browser to thousands of customers using a multitenant architecture. On the customer side, it means no upfront investment in servers or software licensing.

**Twitter Bootstrap** is a simple and flexible HTML, CSS, and Javascript for popular user interface components and interactions.

## RESULTS AND DISCUSSION

### 5.1 Identification of E-Health Services

#### 5.1.1 Surveys and Interviews

The e-health services were identified through surveys and interviews from medical practitioners, IT staff and patients. The results came from hospitals and the results were summarized. The medical practitioners, IT staff and patients suggested the following:

Out of 103 surveyed papers there are (14) Physician, (8) IT Staff, (18) Hospital Staff, (31) Nurses, (32) Patients.

<b>Legend:</b>	
1 – Least Important	4 – Very Important
2 – Seldom Important	5 – Most Important
3 – Often Important	

	<u>1</u>	<u>2</u>	<u>3</u>	<u>4</u>	<u>5</u>	<u>POINTS</u>
<b><u>PHYSICIAN/DOCTOR</u></b>						
PHYSICIAN APPOINTMENT			3	7	4	57
PHYSICIAN ORDERS				14		56
• <b>MEDICAL RECORDS</b>				6	8	64
• <b>DIAGNOSTIC REPORTS</b>					14	70
DOCTORS LIST			2	8	4	58
<b><u>PATIENT</u></b>						
• <b>PATIENTS REGISTRATION</b>			3	12	17	142
• <b>PATIENTS ADMISSION</b>			4	14	14	138
BILLING		3	12	15	2	112
PATIENTS LIST	2	5	11	13	1	102

	<u>1</u>	<u>2</u>	<u>3</u>	<u>4</u>	<u>5</u>	
<b><u>NURSES</u></b>						
NURSNG CHARTS			11	14	7	124
• NURSES LIST			9	14	9	128
• NURSE ON DUTY SCHEDULE				17	15	143
NURSING LOG BOOK				16	16	144
<b><u>IT STAFF</u></b>						
• PHYSICIAN APPOINTMENT				3	5	37
PHYSICIAN ORDERS			2	3	3	33
MEDICAL RECORDS			1	2	5	36
DIAGNOSTICS REPORTS		2	2	3	1	27
• PATIENTS REGISTRATION					8	40
PATIENTS ADMISSION				3	5	37
BILLING			4	2	2	30
PATIENTS LIST				6	2	34
NUSING VITAL SIGNS			4	3	1	29
NURSES LIST		3	3	2		23
NURSE ON DUTY SCHEDULE		2	4	2		24
NUSING LOG BOOK		4	3	1		21
NEWS				3	5	37
OUTPATIENT LABORATORY			3	2	3	32
LABORATORY REPORTS				4	4	36
WARD MANAGEMENT			3	4	1	30
PHARMACY MANAGEMENT			1	3	4	35
<b><u>HOSPITAL STAFF</u></b>						
• NEWS		2		7	9	77
OUTPATIENTS LABORATORY			3	15		69
• LABORATORY REPORTS				13	5	77
WARD MANAGEMENT		2	3	8	5	70

PHARMACY MANAGEMENT			5	5	8	75
---------------------	--	--	---	---	---	----

- *Top 2 for each user type of the application.*

The Features that the proponents have identified that would be feasible to develop given the short time:

1. Patient List
2. Patient Accounts
3. Patient Medical History
4. Doctor List
5. Doctor's Appointments
6. Doctor Assistant Accounts
7. E-Prescribing
8. Patient Physical Exam
9. Hospital Staff Profiling
10. Nursing Vital Signs Input
11. Laboratory Results
12. Laboratory Test Profiling
13. Patient-Doctor Messaging

The doctor's assistant account was recommended by Mr. Antonio Bulao II. The proponents also added more features such as E-Prescribing and Messaging. We did not include the facilities such as the wards and billing since it would not be feasible to include those given the period of time of the project. We only included the core e-health services which are the medical records which are needed more to provide better health care. Even though nurse list and nurse on duty-schedule are top of the survey for nurses, the proponents decided to put the nursing vital signs input as the feature since it would benefit the doctor and patient more than the other features.

## **5.2 Creating the Database**

The tables in the database were created based from different methodologies. The proponents used the forms that we have gathered from different hospitals such as the laboratory module, nursing vital signs input and admission of patients. The team also interviewed doctors and used the data gathered in designing the consultation and the



medical history of the patients. The team also studied existing systems of e-health care applications and implemented some of the features such as appointment and e-prescribing. Some tables were created based on research from books such as the patient's physical exam.

The proponents did an object oriented analysis on hospitals and created the physical design of the database using MySQL workbench. The proponents studied who are the users of the application and implemented them as models.

• Doctor	Doctors are needed in order to diagnose patients and give orders for the hospital staffs. They are also responsible for the treatment of patients.
• Doctor Assistant	Doctor assistant are the one who would encode the medical records of a patient after consultation and they could set appointments also.
• Nurse	Nurses are needed in order to check the condition of the patient in the hospital from time to time and the data from nursing charts are very important to doctors.
• Hospital Administrator	Hospital Administrators are responsible for adding and updating the users of the system.
• Admission Staff	Admission staffs are responsible for encoding the basic information of patients during admission and their medical history.
• Laboratory Staff	Laboratory Staff are the responsible for encoding the lab results taken by patients. They are also the ones who would create templates for forms.
• Patient	Patients are needed in order for a better interaction with doctors and to be a part of the health care process.

*Figure 4. Database Design for the Users of the Application*

### 5.3 Creating the Web Application

Based from the physical design the proponents created, Yii framework was used to implement the design of the database in a Model-View-Controller pattern. Yii Code Generator generates an Active Record class using the Object Relation Mapping technique. All of the models of the application are implemented as Active Record classes. The CRUD operations are implemented as Active Record methods.

*Figure 2. Yii Code Generator*

```
class Patient extends CActiveRecord
{
    public static function model($className=__CLASS__)
    {
        return parent::model($className);
    }

    public function tableName()
    {
        return 'patient';
    }

    public function relations()
    {
        return array(
            'p'=>array(self::BELONGS_TO, 'User', 'login_id'),
        );
    }
}

class User extends CActiveRecord
{
    .....
```

```

public function relations()
{
    return array(
        'doctors'=>array(self::HAS_MANY, 'Doctor', 'login_id'),
        'patients'=>array(self::HAS_MANY, 'Patient', 'login_id'),
    );
}
}

```

*Figure 3. Code for the Active record database relations*

### 5.3.1 User Classes and Login

The proponent created a class for each user and has the user id as foreign key. The User class would enable the separation of one hospital to another and distinguish the users of the application. The hospital administrator account is created in the User table with the hospital id as foreign key and that included the id of the hospital as foreign key and “admin” as the user type.

The default login for the application is provided by Yii Framework. It uses the CWebUser class in storing the details of the login. The user table contains a user\_type field that would distinguish different users.

```

Yii::app()->user->setState('roles', $user->user_type);

```

The setState function sets the user type of the user to the roles variable of the CWebUser class. This would be later used to control the access of the users to certain functions of the application.

In the generated code of the controller, there is a method called accessRules() that would control the access of the users.

*Figure 4. Code for Controlling the Access of Users*

### 5.3.2 Patient Admission, Consultation, Physical Exam

The admit table of the database would contain the admission dates and discharge dates of the application. The social table stores the information about the patient social habits such as the frequency of smoking and alcohol drinking.

In the consultation the data is stored in the patient\_encounter table of the database. This contains the doctor findings of the patients, chief complaints, diagnosis and management of the patient. Together with the consultation is the physical exam. We used the Tabs widget of Twitter bootstrap in creating the tabs of the physical examination.

```
<?php $this->widget('bootstrap.widgets.BootTabbable',array(
    'placement' => 'top',
    'tabs' => array(
        array('label' => 'Vitals Signs', 'content' =>
$this->renderPartial('Encounter/_vitals', array(
    'model' => $model,
    'form' => $form,
    'allow' => $allow), true), 'active' => true),
```

If the finding of the doctor is abnormal during the physical exam, we used jQuery to check if the abnormal field is checked then it would show the checklist of abnormalities of the patient.

```
$('#_cardioCondition').prop('checked') == true) ?
$('#_cardio').show(250) : $('#_cardio').hide(250);
```

The data is stores in the physical\_exam table of the database. It is stored as Strings of 1 and 0.

*Figure 5. Database design for Admission, Consultation and Patient Physical Exam*

### **5.3.3 Laboratory**

We designed the database of the laboratory module by creating a field table which would contain all the fields of all the forms of the examinations. The field table contains an input name that is displayed during adding the results and an output name that is displayed when viewing the laboratory results. Lab\_Tests table would contain all the test of a particular hospital. Lab\_Field would bind the existing fields to a form of a hospital.

*Figure 6. Database Design for Laboratory*

### **5.3.4 Patient Medical History**

In the patient profile, we displayed the different kinds of medical history such as the illnesses, medication, allergies and family history. We created a patient\_issue table to store the medical history. In the patient\_issue table, we created a field that would identify the kind of medical history of the patient (Allergy, Medication, Illness). In the family history, we created a family table that would contain the name of the medical problem and the family member and its relation to the patient.

*Figure 6. Database Design for Patient Medical History*

### **5.3.5 Appointment**

The appointment table has the fields date and time of the appointment. The details of the appointment would be stored in the reason field of the table. We created a different User Interface for the patients account. The patient's reply to the appointment would be stored in the reply field.

### **5.3.6 E-Prescribing**

The proponents asked for a sample prescription from a doctor in Ateneo de Davao Clinic. We created a prescription table that contains the fields that should be in a standard prescription. The table contained the drug\_name, drug\_form, brand\_name, dosage, period and written\_date. Drug name is the generic name of the drug. Drug form can be in a form of tablet, capsule or syrup. Dosage is how often is the patient going to take the drug within a day. Period is how long the patient would take the drug. The drug would also be added in the medical history of the patient.

### **5.3.7 Messaging**

The messaging module was created by adding an extension from yii framework called yii-messaging. The extension was put in the extensions folder in the application. It was defined in the config.php in yii.

*Figure 7. Database Design for Appointment, Prescription and Messaging*

### **5.3.8 Nursing Vital Signs Input**

The proponents created a nurse\_patient table that would contain the fields of data that nurses are collecting during their rounds in the hospital. This would contain the date and time of collection, blood pressure, respiratory rate, pulse rate and temperature. The data can be viewed by the doctor.

*Figure 8. Database Design for Nursing Vital Sign Input*

### **5.3.9 Controllers**

The controllers of the application were used to pass data from the view and are processed in the model. The controller enables the creation of new model that is to be saved in the database.

```
User::model()->findAllByAttributes(array('hid' => Yii::app()->user->hid, 'user_type' => 'patient'));
```

*Figure 9. UML for the Controllers*

### **5.3.10 Views**

The proponents used Christopher Niska's extension of Yii Framework that enabled the integration of Twitter Bootstrap into the framework's widgets. The integration was done by adding the downloaded Twitter Bootstrap folder into the extensions folder of Yii and defined bootstrap in the components in the main configuration of the framework.

```
'bootstrap'=>array(  
    'class'=>'ext.bootstrap.components.Bootstrap',  
),
```

The views of the application were all designed with Twitter Bootstrap interfaces such as the grid system and the widgets such as the tables, detailed view and the tab

## **5.4 Deployment to Server**

The proponents used Secure Shell to connect to the Hopkins server of Ateneo de Davao University.

*Figure 9. Connecting to Port 22 of Host Name using PUTTY*

The proponents also used dropbox for transferring files to the server. Wget is used to download the files from the dropbox server to the dropbox server.

### 5.5 Testing

To prove the effectiveness of the study, the application was tested with different health practitioners from Davao City. The proponents let the users use the application and provided user acceptance forms for evaluation. Each user could rate the functionality of the system from 1 to 5 and give comments for each function.

For our user acceptance test, we have prepared a questionnaire for the testers so that they could rate our application and gave some comments and recommendations. We tested our system in Esperanza, Sultan Kudarat and in Ateneo de Davao University. Since our project is a web application, we had to look testers like patients, students and especially doctors. We got a total of 40 testers; 9 of them are doctors and 31 of them are patients and students.

Summary of the user acceptance forms:

<b>Features</b>	<b>5</b>	<b>4</b>	<b>3</b>	<b>2</b>	<b>1</b>
Fast and easy to use	10	26	4		
Web Application's design is user friendly	19	18	3		
Service Oriented Architecture	12	23	5		
Patient Account	11	25	3	1	
Patient Medical History	14	22	4		
Doctor's Appointments	16	17	6		
Doctor Assistant Accounts	18	16	7		
E-Prescribing	14	16	10		
Patient Physical Exam	12	24	4		
Hospital Staff Profiling	15	21	4		
Nursing Vital Signs Input	15	17	8		
Laboratory Results	11	24	5		
Laboratory Test Profiling	12	17	11		
Patient-Doctor Messaging	6	9	24	1	



## **CHAPTER 6**

### **CONCLUSION AND RECOMMENDATIONS**

#### **1. Conclusion**

The program met the following objectives as proposed:

- The application was able to provide e-health services such as medical records, laboratory results, physician consultation, patient appointments, e-prescribing, patient physical examination, nursing charts.
- The application was implemented in a service-oriented architecture using Relational Database.
- The application enabled the team to learn a PHP Framework through Yii and graphical user interface tools such as Twitter Bootstrap.

#### **6.2 Recommendation**

The application could be improved if the proponents would have more time since the project has a big scope. One of the recommendations is to add more e-health service to the application like billing and facility management for hospitals. The application

could have an offline application that would sync with the cloud server when there is an internet connection since hospitals system are very crucial and the application should also work even when there is no internet connection in order to not affect the hospital process. Another recommendation is to improve the security of the site such as putting Two-Factor authentications. A mobile application would also be beneficial.

## BIBLIOGRAPHY

**Bickley MD, L. (2008). Bates' Guide to Physical Examination and History Taking, 10th Edition. New Jersey: Westfield.**

Physician Specialty Description. *Boone County Hospital*. Retrieved from <http://www.boonehospital.com/bch.nsf/View/SpecialtyDescriptions>

**Knorr, E & Gruman, G. (2008). What cloud computing really means. Info World.**  
Retrieved February 2, 2013 from <http://www.infoworld.com/d/cloud-computing/what-cloud-computing-really-means-031>

SOAP Notes. Retrieved from

<http://mededucation.bjmu.edu.cn/reportreference/reference/chapter%20one/soap%20notes.htm>

**International Classification of Diseases. World Health Organization. Retrieved from <http://www.who.int/classifications/icd/en/>**

Yii Framework. (2011-2012). *Create your own Validation Rule*. Retrieved from <http://www.yiiframework.com/wiki/168/create-your-own-validation-rule/>

Yii Framework. (2012). *Simple RBAC*. Retrieved from <http://www.yiiframework.com/wiki/328/simple-rbac/>

Niska, C. (2012). Bringing together the Yii PHP framework and Twitter Bootstrap. *Yii-Bootstrap*. Retrieved from <http://www.cniska.net/yii-bootstrap/>

Yii Framework. (2011). *Declaring Validation Rules*. Retrieved from <http://www.yiiframework.com/doc/guide/1.1/en/form.model#declaring-validation-rules>

*Adding a Timepicker to jQuery UI Datepicker*. (2012). Retrieved from <http://trentrichardson.com/examples/timepicker/>

Yii Framework. (2011, November 24). *Private-Messaging*. Retrieved from <http://www.yiiframework.com/forum/index.php/topic/26111-extension-private-messaging/>

[Yii Framework. \(2011, November 26\). Examples of using the CHTML class of Yii.](http://www.yiiframework.com/wiki/48/by-example-ckeditor/) Retrieved from <http://www.yiiframework.com/wiki/48/by-example-ckeditor/>

Uus, E. (2013). *Form design - Complex dynamic rows. Visual Yii*. Retrieved from <http://www.eha.ee/labs/yiiplay/index.php/en/site/extension?view=dynamicrows>

Yii Framework. (2012, July 7). *PDF Extension for Yii*. Retrieved from <http://www.yiiframework.com/extension/pdf/>

Yii Framework. *Yii Framework API Class Reference*. Retrieved from <http://www.yiiframework.com/doc/api/>

