Developing Secure Software
Final Project Report
Secured Healthcare Management Application

University of Pittsburgh
School of Computing and Information
INSFCI 2620 Developing Secure Systems
Final Project Report
Chi-Heng Hung CHH162

Table of Contents

1. Introduction

The Secured Healthcare Manage Application is a mobile application that we designed for user to store and organize their healthcare information. Our main goal is to develop a healthcare application using a secured software developing lifecycle in order to ensure that the user's information that are processed in this application is are handled safely and the information stored in the application is properly encrypted, so that user can store their healthcare data in this application and don't need to worry the confidentiality, integrity and privacy of their data.

1.1 Timeframe

	Description of Work	Time
Requirement Analysis	Review and list all the requirement of Secure SDLC and HIPAA for the application	11/12-11/18
System Design	Design the system according to the requirements that are listed in the previous step	11/19-11/25
Implementation	Implementation the system design into code	11/26-12/09
Testing	Make sure that the code is working as expected and ensure the requirements and regulations are met	12/10-12/15

1.2 Scope

The project is aimed to use a secured software developing lifecycle to develop a healthcare application for Android system with proper security features, like hash function and encryption, to secure user's information. The application should be able to let the user manage medical records, medication records, insurance information and emergency contact information.

1.3 References

- U.S. Department of Health & Human Services www.hhs.gov/hipaa
- Adobe PhoneGap phonegap.com
- Apache Cordova cordova.apache.org
- Ratchet goratchet.com
- CryptoJS code.google.com/archive/p/crypto-js/

- Font-awesome fontawesome.com

2. Background and Objectives

As the progress of technology, people start to store data in digital copy instead of paper copy, including their healthcare data. However, healthcare data includes lots of sensitive data that should be taken cate carefully to protect user's privacy, therefore we are planning on developing a mobile application that can enable users to manage their health records and medical records in which users does not need to concern the privacy of their data that is stored in the application.

2.1 Customer

Any customer that is looking for a mobile application to manage their healthcare information but also want to ensure the security and privacy of their healthcare data.

2.2 Objective

- Develop an application that user can manage health records and medical records
- Provide an interface to show user's emergency contact for emergency situations
- Develop the application using a Secured Software Develop Life Cycle to insure security in the application
- Insure the application meets the regulations of HIPAA (Health Insurance Portability and Accountability Act)

3. Project Results

In this part we will show the project results by going through the requirements and explain them one by one in detail, then show how we develop this application by going through the software development lifecycle.

3.1 Requirements

3.1.1 Requirement Compliance Matrix

ID	Requirement Description	Completed
R01	Manage health records	Yes

ID	Requirement Description	Completed
R02	Manage medication records	Yes
R03	Manage insurance information	Yes
R04	Interface for emergency contact information	Yes
R05	Developing using secured SDLC	Yes
R06	User Authentication	Yes
HIPAA appliance requirements		
R07	Encrypted communications	Partially
R08	PHI in used protection	Yes
R09	PHI at rest protection	Yes
R10	PHI in transit	Yes
R11	Minimum use and sharing of PHI	Yes
R12	Agreements in place	Yes
R13	Emergency access procedures	Yes
R14	Passcode requirements	Yes

3.1.2 Requirements Compliance Summary

Total number of requirements	14
Number of requirements implemented	13
Requirements partially fulfilled	1
Requirements not fulfilled	0

3.1.3 Requirements explanation

R01: Manage health records

Enable user to record health information that includes the date, hospital, doctor, problems, treatments and revisit date. User can add the health records and view it in an organized manner afterward.

R02: Manage medication records

Enable user to record medication information that includes the medicine, dosage, frequency, start and end date. User can add the medication records and view it in an organized manner afterward.

R03: Manage insurance information

Enable user to save insurance information that includes the company name, company phone, policy type, co-pay information, member name and member ID. User can add the insurance information and view it or edit at any time.

R04: Interface for emergency contact information

Enable user to save emergency contact information that includes name, date of birth, blood type, weight, height, allergies, contact name and contact phone. User can add the emergency contact information and view it or edit at any time. The emergency contact information can also be viewed in the sign-in page that does not

requires sign-in to view.

R05: Developing using secured SDLC

The application is developed under a secured waterfall software developing lifecycle, the details of each stage in the lifecycle will be discussed in section 3.2.

R06: User Authentication

User is required to use the password they set up to gain access to the application every time. The password is required to have at least 8 characters with at least one lowercase letter, one uppercase letter, one number and one special characters. The password is hashed and stored in the application for future authentication use.

R07: Encrypted communications

The application is designed to work offline, without internet connection, therefore there will not be communication part that requires encryption. The only communication is the backup process that allows user to back up their data with email function of the Android device, so the security depends on the email function of the Android device, not on us, therefore this requirement is marked as partially completed.

R08: PHI in used protection

The PHI that is in used in this application are decrypted from the storage every time it is called and will not be stored in plaintext after used.

R09: PHI at rest protection

The PHI that is at rest in this application is fully decrypted using AES with 256 bits key at all time, excepted the emergency contact information, which is not encrypted because of the purpose of these information are to be accessible under emergency circumstances.

R10: PHI in transit

The PHI that is used in this application are stored offline and are not transit to any other user, application or server.

R11: Minimum use and sharing of PHI

The PHI that is used in this application will not be shared with any other user, server or application.

R12: Agreements in place

We provided a terms and agreement that includes the permitted uses and disclosures by business associate on the sign-up page and provided a clear instruction on the sign-up page that they must agree the terms and agreement to use this application

R13: Emergency access procedures

Once user have signup and create emergency contact in the

application, an emergency contact button will appear in the sign-in page that allow users to access without login. This can provide other user to access the emergency contact under emergency that the user might not be able to sign-in the application.

R14: Passcode requirements

Every time the user switch between apps or return to the menu on their phone, the session that is used in this application will be deleted and will require the user to login again to regain the access permission to the information in this application.

3.2 Software Development Lifecycle

3.2.1 Overview

Considering that this is a small project with only one developer, we choose to use the waterfall software development lifecycle. In order to ensure the security features are implemented in an efficient manner, we took security features into consideration in each steps of the lifecycle.

3.2.2 Stage 1 - Requirement Analysis

In this stage we identify the requirements of this application. First, we list all the requirements that a healthcare application needed, then we consider the factors that are related to the security features that will affect the security of the application, which includes confidentiality, availability and integrity. For this application, we focus on the confidentiality part to ensure the user's privacy. At the same time, we also go through the requirements and recommendations of the HIPAA regulation to make sure that all the proper requirements are identified and listed. The requirements we identified are listed in section 3.1.

3.2.3 Stage 2 – System Design

In this stage, we start to design our application with the requirements listed in previous step and design proper mechanism to fulfill each of them. There are few main mechanisms that are used in the application in which we want to mention here.

The first one will be how user's password are handled. As the user create their account on the sign-up page, we will take the password, salt it with user's email then hash it with hash function SHA-256, the result we get will be used as the session key. Next, we take the session key, salt it with the user's email then hash it with hash function SHA-256 again, the result we get will be stored along with user's email in the application for future authentication use. The user's plaintext password will not be stored in any part of these steps. Every time the system

detected the user pause, switch, or close the application, it will delete the session file we created previously. When user return to the application, they will be asked to re-enter the password, we will repeat the two steps hashing mechanism again and compare the result with the file stored to identify if the password is correct. This is how we handled user's password.

The second one we are going to talk about will be how user's healthcare data will be encrypted. The encryption algorithm we choose to use is AES with 256 bits key which is the session key we generated in previous step. Every time user creates or adds data into the application, we will encrypt it with the session key and stored it, once they are called or needed, we when then decrypted it with the session key and display it on screen. User's data will not be stored in plaintext in any format or any time during all the process.

The last one we are going to talk about are some extra mechanisms that we used to protect the application and user's information. The application is a web-based application, this application is programmed in HTML, CSS and JavaScript, then compiled and packed by PhoneGap into an Android application. Which means that some common vulnerabilities on web application might appears on our application too, therefore we put some efforts on it to eliminates the vulnerabilities. We used input validation for every input the user may make to prevent any kind of malicious code injection. We also applied the Content Security Policy and Subresource Integrity check to prevent unwanted scripts and code from executing in our application. By doing so, we can at least eliminate two vulnerabilities that are listed in OWASP Top 10 2017.

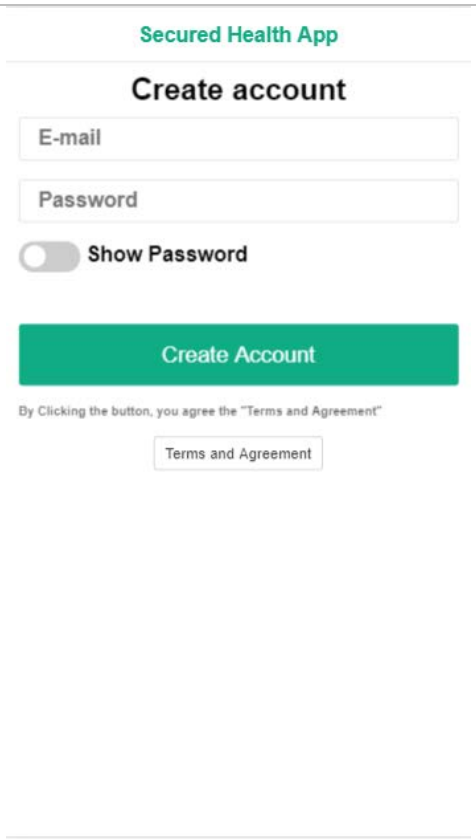
3.2.4 Stage 3 – Implementation

In this part, we followed the design in previous step to implement them one by one. We did meet several problems during this stage, but we are able to solve them eventually. In this part we use the Sublime Text Editor as our main editors, along with PhoneGap CLI and PhoneGap Developer as the testing and debugging platform.

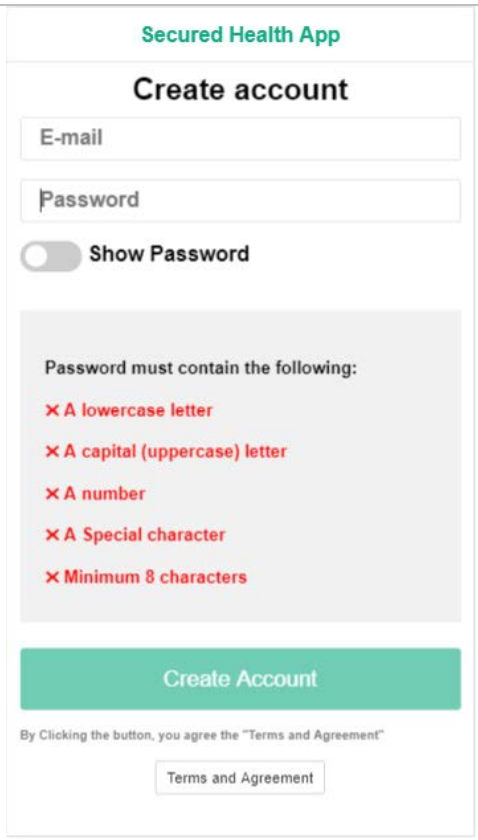
3.2.5 Stage 4 – Test

In this part, we reviewed the requirements that is identified and listed in stage 1 to ensure that every requirement is fulfilled. We also tried all the functions designed in stage 2 are properly implemented and no unwanted functions are implemented. We do find some bugs and unexpected feature and have them fixed or removed in this stage.

3.3 Application Screenshot

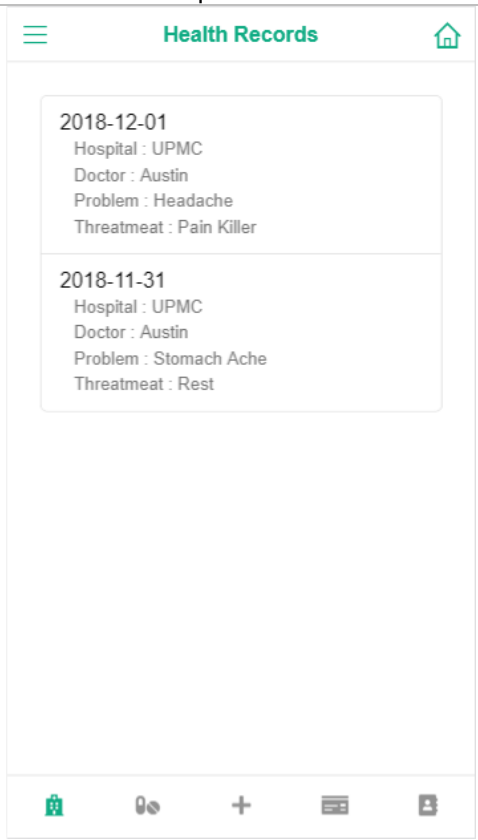


The create account page

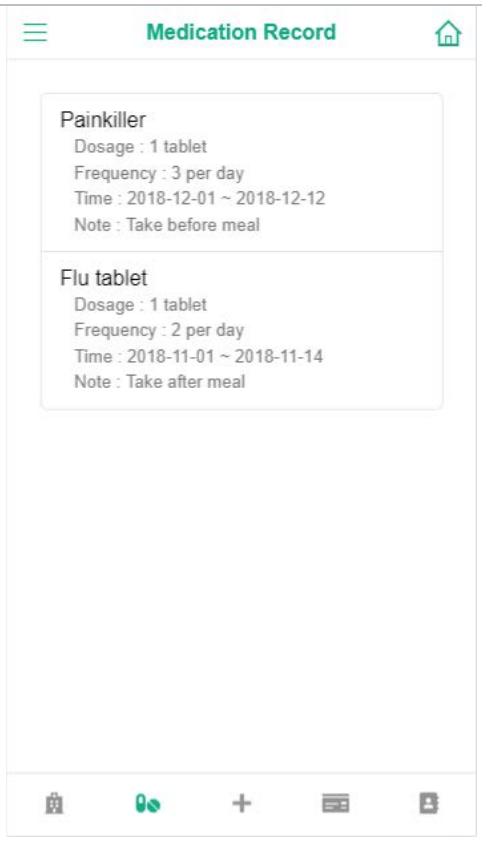


Create account page with password requirement

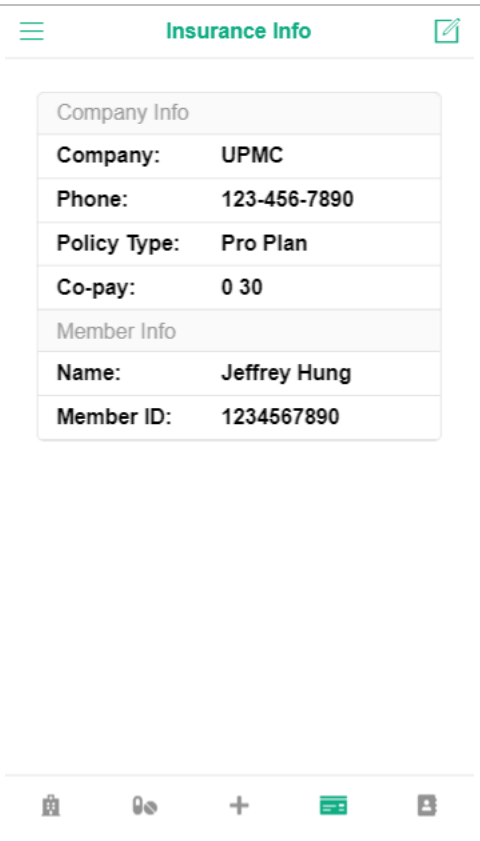
Home page



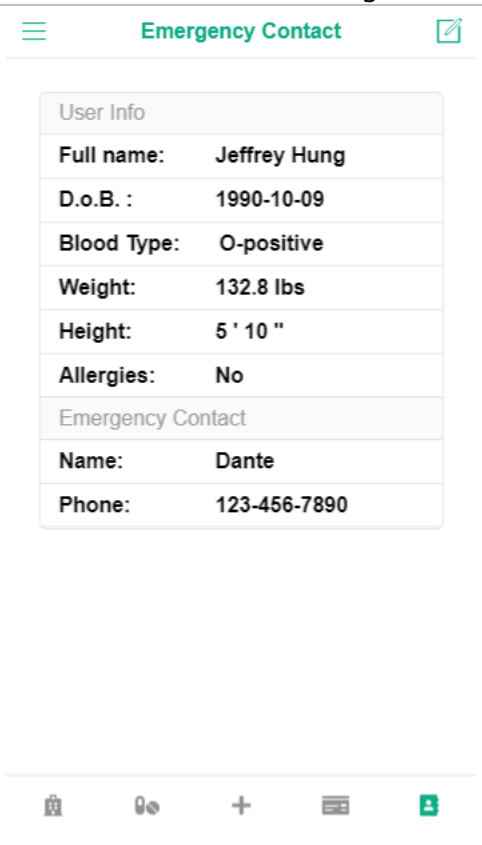
Heath Record page



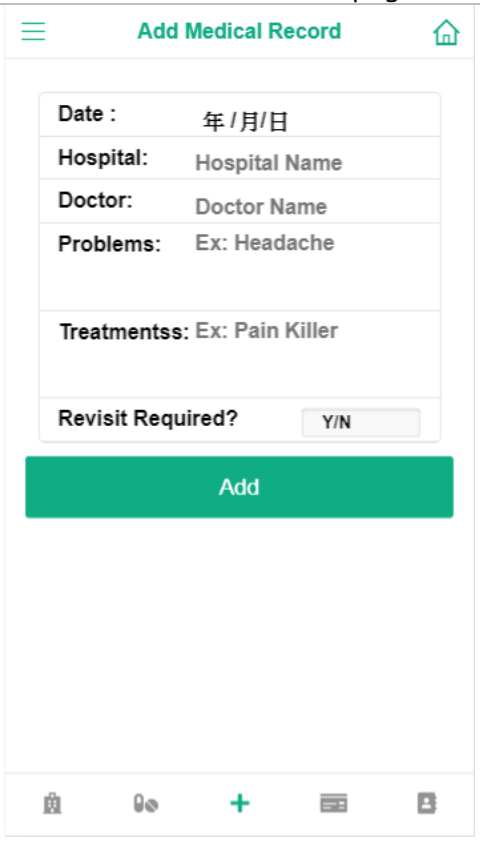
Medication Record Page



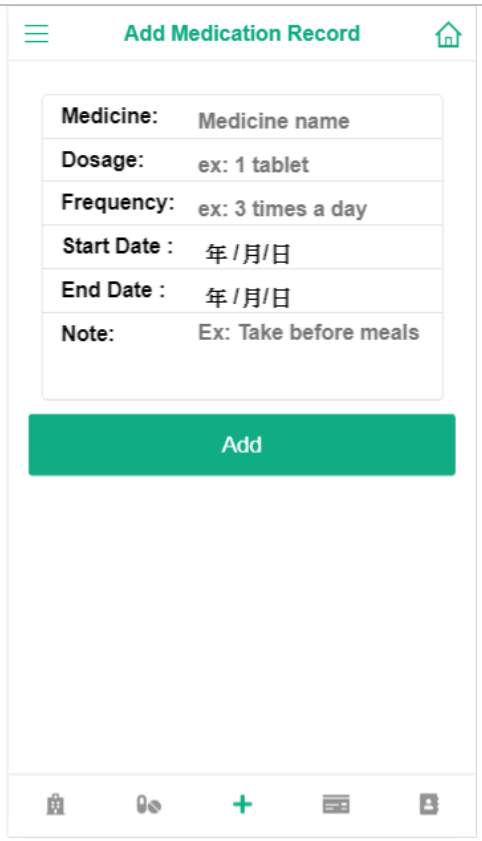
Insurance Information page



Emergency Contact page

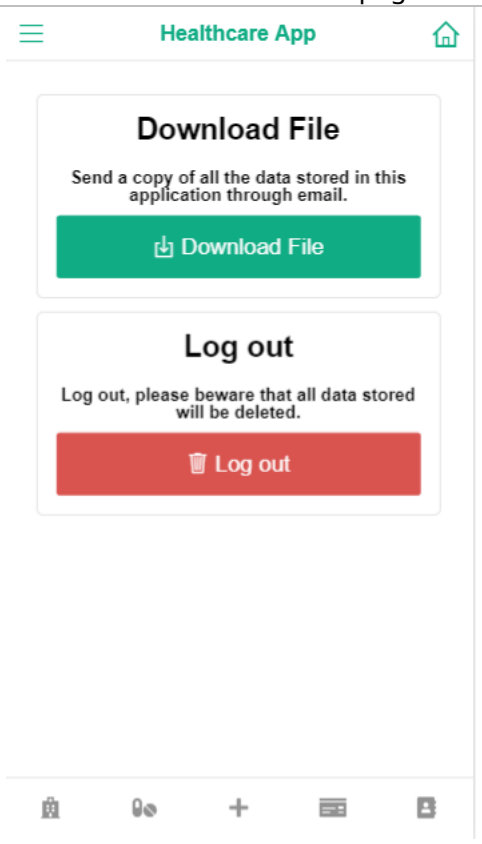


Add medical Record page

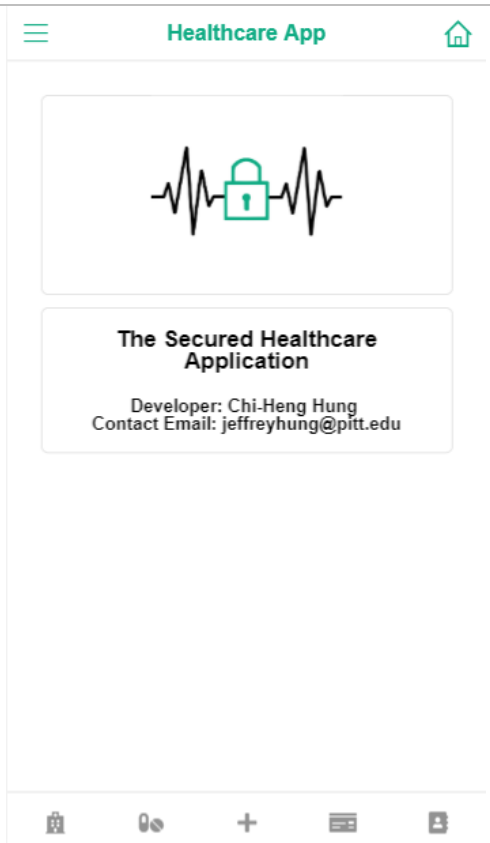


Add Medication Record page

Sidebar with more functions



Setting page



Support Page

4. Project Experiences

In this part of the report, we will talk about the things we learned and some future works of this project.

4.1 Positive Experiences

The main experience we learn from this project is how to build an application using a secured software development lifecycle, in the past we do work on several different projects before, but security is always not the main goal. In this project, we lean to think in a security aspect during every process of the development in order to make sure that security mechanisms are properly implemented in every detail of the application. We also learn how to build a web-based mobile application through this project, we learn to use the tools like Cordova, PhoneGap, which we have never tried before.

4.2 Improvement Possibilities

For the software developing process, we appeared to be not so familiar with the lifecycle and spent some time getting familiar with it, in the future if we are using the same secured software developing lifecycle, we can surely get handy in a short time and improve the efficiency throughout the process.

For the secured healthcare application that we developed, we mainly focused on the confidentiality part of user's data, in the future, we can also implement some features to insure the integrity of user's information and add some more features to make this application more useful.