School of Computing and Information
University of Pittsburgh

Lab Project 4
Secure Mobile Application Development
Integration of Notification on Smart Watch

Version 1.0

This lab is based on mHealth project by *Haobing Huang* and by *Runhua Xu*.

School of Computing and Information
University of Pittsburgh

Read the following guidelines before working on the project

Lab Project Goal:

The goal of this project is to learn the basic skills for developing a secure mobile health app. This project will involve the following tasks.

- Secure the outsourced medical reminder data
- Add wearable features to notifications

General Guidelines:

For this project, you will first need to review the developer's guides for **Android**, **Google Firebase**, **Cloud Key Management Service** and **Android Wear** if you don't have the relevant background and experiences. Check the following materials:

- Documentation for app developers
  [https://developer.android.com/docs/]
- Documentation for Google Firebase
  [https://firebase.google.com/docs/]
- Documentation for Google Cloud Key Management Service
  [https://cloud.google.com/kms/docs/]
- Documentation for Android Wear
  [https://developer.android.com/training/wearables/]

NOTE:
Please note that we have integrated the Firebase by providing our LERSAIS's ***google-services.json*** file in the project. It may not be allowed you to manage the backend server, namely, the Firebase console. If you want to debug/test your project using your own Firebase console. Please follow the instructions provided by Firebase and replace the ***google-services.json*** file that have been provided to you in the project skeleton with your own.

Moreover, we have also integrated the Google Cloud Key Management Service and provided the ***LERSAIS-mHealth-KMS-bdd9f7acef42.json*** file in the assets folder in the skeleton. Please don't create yours.

Our LERSAIS lab has real android phones and smart watches for testing in case you need. Please schedule a time slot with TA.

School of Computing and Information
University of Pittsburgh

Secure Outsourcing Medical Reminder
and
Wearable Features to Notifications

A Healthcare Scenario [extended version]

In your prior three lab projects, we have the following scenario:

Suppose that a patient needs to go to different hospitals/clinics for different healthcare services/checkup/treatment. It is often tedious to fill the medical forms to provide his/her medical history information such as allergy history, family genetic history, etc. Furthermore, the staff/information systems of the hospital or health clinics/units may be not fully trustworthy to store and manage his/her medical history record.

Thus, the goal here is to design a mobile healthcare application to help a user manage the medical history record. It is a user-centric medical record management application where users have full and complete control of their healthcare data. The medical history record is sensitive data for patients because of privacy concerns. Thus, it is necessary to encrypt the medical history record before outsourcing it to the cloud.

Then, we have considered the medical history record usage scenario. Suppose that a patient is receiving treatment in the treatment room and he/she needs to present his/her medical history record to a group of people including attending physician, assistant physician, trainee physicians and nurses in the treating room. Thus, it is necessary to design a proximity-based data access control approaches with consideration of privacy concerns.

Suppose that a patient has received the medical treatment. Before he/she leaves the hospitals/clinics, he may get some medical advice documents/papers. It is inconvenient to take or keep such documents/papers by hand. Thus, we plan to add a *medical case* feature in the app.

Specifically, users can take a photo for each medical advice document. The app can recognize the text in the photo. It is allowed to save both the photo and recognized text in the cloud database. Users can retrieve/view the medical advices anytime anywhere. In addition, as in project 2, these medical advice data should also be kept confidential in the cloud database due to privacy concerns.

In your lab project 4, we have the following additional application requirements:

Suppose that a patient has received the medical treatment and left the hospitals/clinics with medical advice. Then, one medical advice is that the patient should take the medicine regularly. For instance, the patient should take xxx pills every morning when he/she gets up. Thus, the goal here is to allow the patient to add medical reminder that will remind the patient to take pills at time by sending notification to both cell phone and wearable devices such as smart watch. In addition, as in project 2 and project 3, these medical reminder data should also be kept confidential in the cloud database due to privacy concerns.

School of Computing and Information
University of Pittsburgh

Task: Add and List Medical Reminder Items

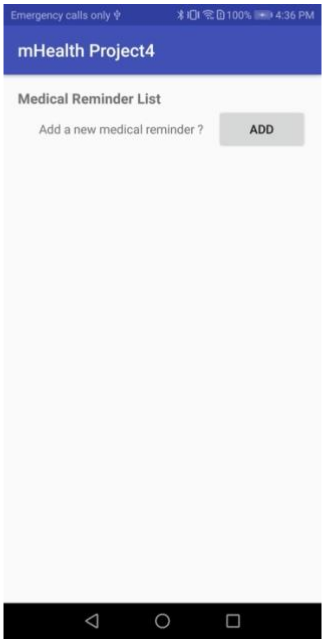
This task requires you to add a medical reminder item and save the encrypted medical reminder item into the cloud database. In addition, the reminder should also be notified on both cell phone and wearable devices, namely, android smart watch. Specifically, the requirements/procedures are described as follows:

1. A typical medical reminder in the provided skeleton includes five properties:
    a. The identity of the medical reminder, which can also be used to present the unique notification id in the android system in case we have more than one reminder.
    b. The title name of the medical reminder
    c. The content of the medical reminder
    d. The trigger time of the reminder.
    e. The repeating interval time of the reminder.

    Even though the model, **MedicalReminder**, is provided, you can also modify if you need.
2. The app can notify the reminder to the user.
3. The repeating reminder can be cancelled.
4. The app can encrypt the medical reminder and save to the cloud database
5. The app can list all medical reminder items that a user has.

Go through the code provided in the skeleton. The interface/view is already provided for you, so you do not need to worry about design it. We have provided the model class, namely, **MedicalReminder**, and the view/interface for adding a medical advice.

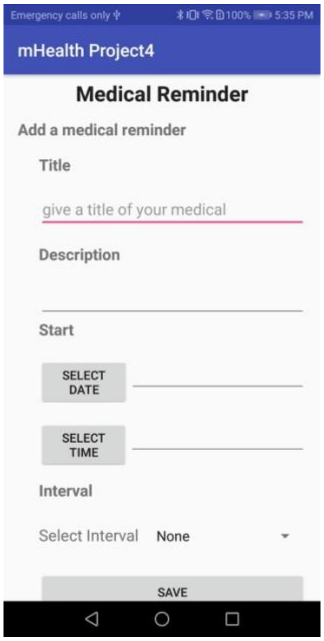
Sample Views from Cell Phone



mHealth Project4

Medical Reminder List

Add a new medical reminder ? ADD



mHealth Project4

Medical Reminder

Add a medical reminder

Title

give a title of your medical

Description

Start

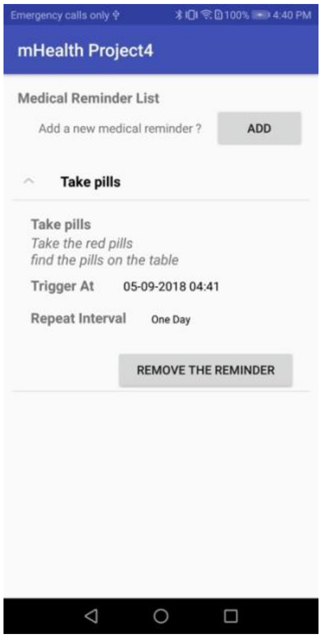
SELECT DATE

SELECT TIME

Interval

Select Interval None

SAVE



mHealth Project4

Medical Reminder List

Add a new medical reminder ? ADD

Take pills

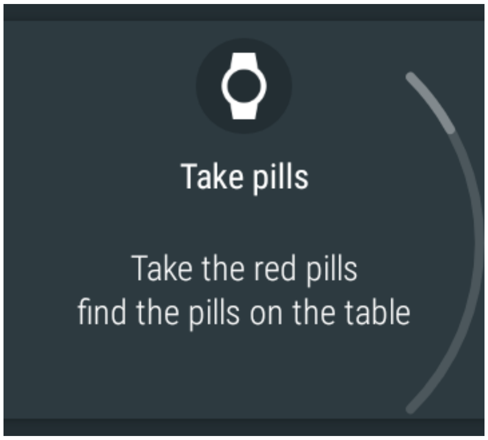
Take pills
Take the red pills
find the pills on the table

Trigger At 05-09-2018 04:41

Repeat Interval One Day

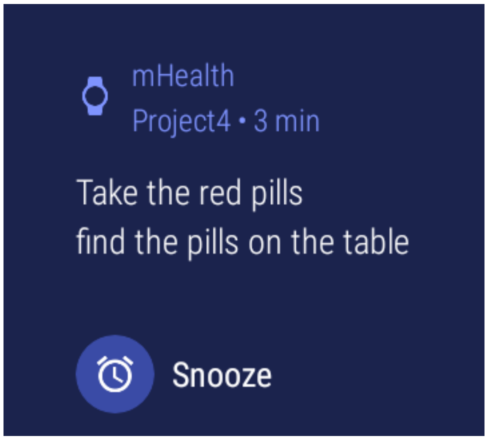
REMOVE THE REMINDER

Sample Views from Smart Watch



Take pills

Take the red pills
find the pills on the table



mHealth Project4 • 3 min

Take the red pills
find the pills on the table

Snooze

Specifically, you will accomplish the following tasks:

Task 1 Add a medical reminder to the cloud database (MedicalReminderAddActivity)

Task 1.1 Create the medical model from the view
Task 1.2 Sava the medical model to the cloud database

Note that you can take the case of outsourcing secure medical advice as example to handle Task 1.2

Task 2 Schedule a medical reminder notification to users. (MedicalReminderAddActivity)

Task 2.1 Create notification channel
Task 2.2 Generate a notification include the require message
Task 2.3 Setup the repeating alarm to notify users

*Note that you should first learn to use **AlarmManager** to repeat a schedule. Then, specify the schedule as notification schedule. Finally, specify the notification with customized design, for instance, **BigTextStyle**.*
Here are some recommended steps:

1. *Create notification channel*
2. *Build the BIG_TEXT_STYLE*
3. *Set up main Intent for notification*
4. *Create additional Actions (Intents) for the notification, e.g., dismiss action, snooze action.*
5. *Build and issue the notification*
6. *Add a schedule (broadcast service) to trigger the notification*
7. *Use the AlarmManager to repeat the schedule*

Here are references you can learn from:

- *https://developer.android.com/training/wearables/notifications/creating*
- *https://developer.android.com/training/scheduling/alarms*

Task 3 List all medical reminder items. (MedicalReminderListActivity)

Task 3.1 present the medical reminder list
Task 3.2 remove the alarm (ExpandableMedicalReminderListAdapter)
Task 3.3 remove the medical reminder from database (ExpandableMedicalReminderListAdapter)

Note that we use an ExpandableListView to present all medical reminder items, which indicates a user may have more than one medical reminder item. The model of MedicalReminder and the adaptor of ExpandableMedicalReminderListAdapter have been provided. Please check the related documentation of android to learn how to use it. In addition, you can take what you have did in project 3 (also provide in the skeleton of project 4), namely, how to present the Medical Advice Record as the reference.
Here is reference for managing a list of data in Firebase:

- *https://firebase.google.com/docs/database/android/lists-of-data*