

School of Computing and Information  
University of Pittsburgh

# **INFSCI 2955: Special Topics in Security Assured Health Informatics**

## **Lab Project 1 Secure Mobile Application Development**

Version 1.5

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

*Read the following guidelines before working in 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.

- Build a basic health app in mobile platform
- Implement basic security features:
  - Registration and Authentication
  - Basic Profile Management
  - Demo of Medical Record Management
  - Authentication Reset

## General Guidelines:

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

- Documentation for app developers [ <https://developer.android.com/docs/> ]
- Documentation for Google Firebase [ <https://firebase.google.com/docs/> ]

There should be other free tutorials or learning materials available in the web that you can learn from.

### NOTE:

Please note that we have integrated the Firebase by providing LERSAIS's **google-services.json** file in the project. It may not allow 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.

## Build Your First Application

### Task 1: Build your first app and familiarize with the development environment

Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA. Check the following guides to be familiar with Android Studio.

- <https://developer.android.com/studio/intro/>

Follow the following guides provided by official documentation to build your first app

- <https://developer.android.com/training/basics/firstapp/>

Here is another helpful tutorial for you in case you need it.

- <https://www.tutorialspoint.com/android/index.htm>

## Build A Healthcare Application

### A Healthcare 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 not be 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.

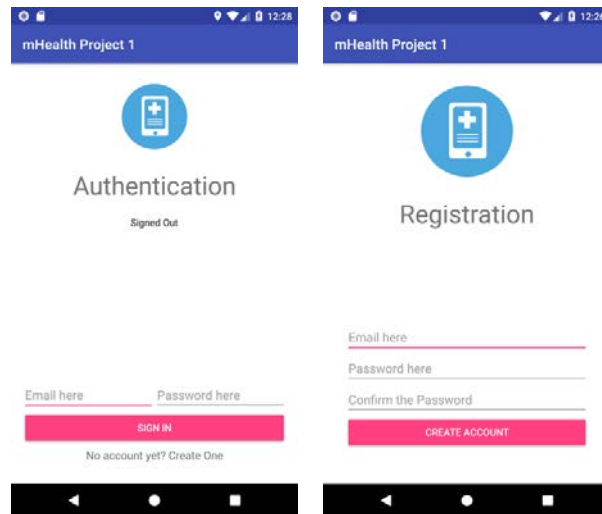
To help you start, we have provided basic skeleton of the program of the app. You can follow the instructions to complete this project step by step. To begin, you need to download the provided project package file and import it into your Android Studio.

Note that this project utilizes or is integrated with Google Firebase. Firebase can help you build apps fast, without having to manage infrastructure. In essence, we treat the Firebase as the backend/server-side infrastructure to support the operation of the mobile app. As the Firebase has been integrated, you do not need to worry about the backend.

## Task 2: Registration and Authentication

The project adopts the password authentication feature of Firebase (check the following link to find more details) to accomplish the registration and authentication of users in our app.

- <https://firebase.google.com/docs/auth/android/password-auth>



Task 2.1 Go through and learn the whole skeleton of the project.

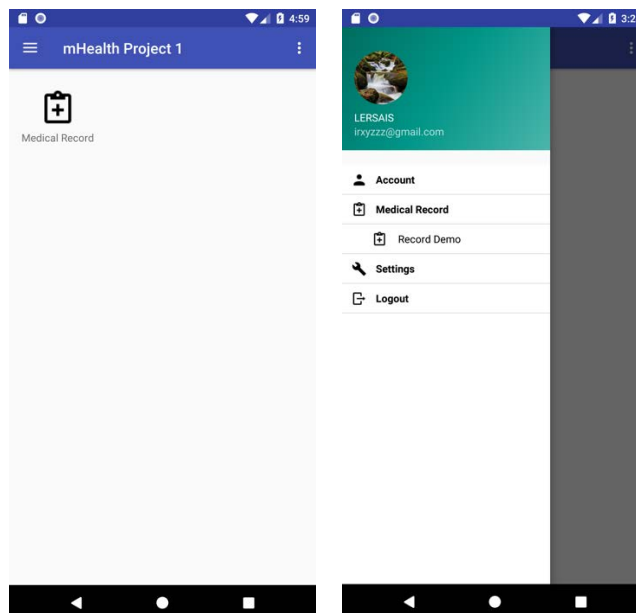
Go through the activity in **SignupActivity** and **LoginActivity** and implement the missing parts. The interface/view is already provided to you, so you do not need to worry about designing it.

Task 2.2 Implement the registration feature (SignupActivity)

Task 2.3 Implement the authentication feature (LoginActivity)

Task 2.4 Implement the sign out function (MainActivity).

If you have successfully completed Task 2, you will be able to register and sign into the app. A sample snapshot is as follows (an emulator with Android API level 26).



### Task 3: Basic Profile Management

We adopt Firebase to authenticate users; it also provides basic profile management feature. Check the following provided guides to for this task.

- [https://firebase.google.com/docs/auth/android/manage-users#update\\_a\\_users\\_profile](https://firebase.google.com/docs/auth/android/manage-users#update_a_users_profile)
- <https://firebase.google.com/docs/storage/android/start>
- <https://firebase.google.com/docs/storage/android/upload-files>

Go through the activity in **ProfileActivity** and implement the missing parts. As the view is provided, you do not need to worry about designing it.



Task 3.1 Allow user to update display name (ProfileActivity)

Task 3.2 Allow user to update profile photo (ProfileActivity)

Task 3.3 Display user's name and photo (ProfileActivity)

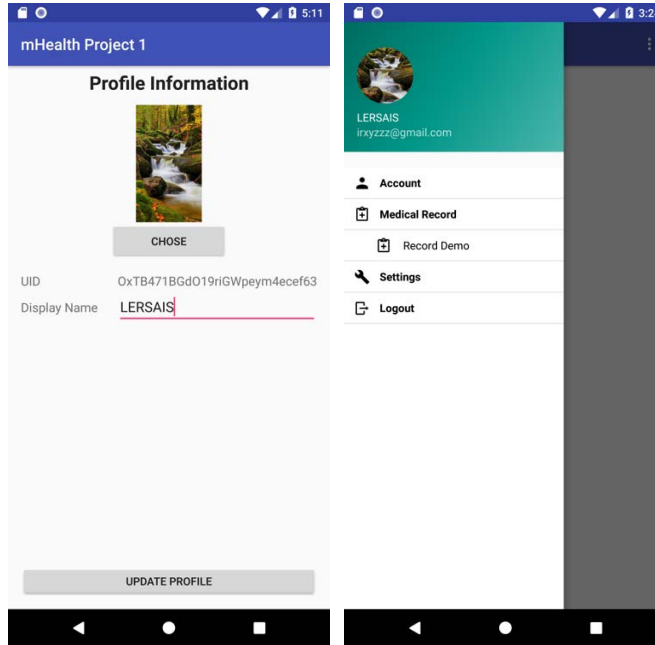
Tips: We suggest that you use the Storage service provided by Firebase to store and manage a user's photo file. If you accomplished Task 3 successfully, you will be allowed to update the *profile photo* and *display name*.

Tips: a new intent can be created with "ACTION\_PICK" and "MediaStore.Images.Media.INTERNAL\_CONTENT\_URI" and then gallery activity will be started with the given request code.

Tips: override the onActivityResult() where the photoImageView will be updated with the data of the intent created in the previous step.

A sample snapshot is as follows.

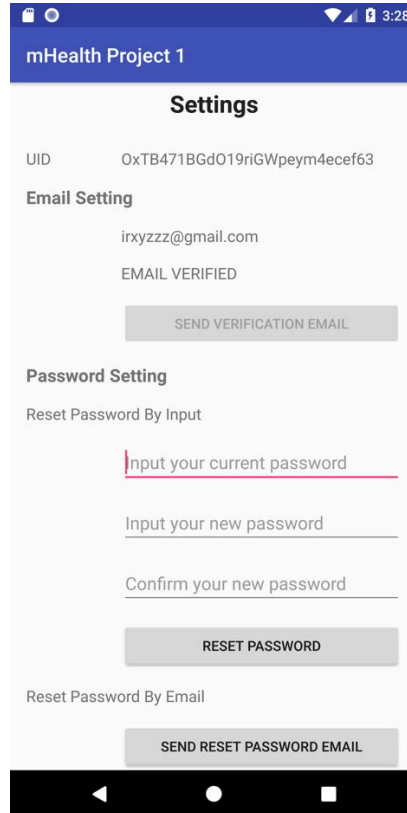
School of Computing and Information  
University of Pittsburgh



## Task 4: Password Reset

Go through the registration activity in **SettingActivity** and implement the missing parts. As the interface/view is provided, you do not need to worry about that.

- [https://firebase.google.com/docs/auth/android/manage-users#set\\_a\\_users\\_password](https://firebase.google.com/docs/auth/android/manage-users#set_a_users_password)
- [https://firebase.google.com/docs/auth/android/manage-users#send\\_a\\_password\\_reset\\_email](https://firebase.google.com/docs/auth/android/manage-users#send_a_password_reset_email)



Task 4.1 reset password by input (SettingActivity)

Task 4.2 reset password by sending email (SettingActivity)

## Task 5: Demo of Medical Record Management

This is an open task, namely, we don't provide the view/GUI and skeleton code. You should design your view and add your code to support the following:

- The medical record should include the following information
  - General Information: Name, DoB, Gender, Marital Status, Occupation, Contact
  - Past Medical History: Allergy, Heart attack, Rheumatic Fever, Heart Murmur,
  - Family Diseases: Father, Mother, Sibling
  - Habits: Alcohol, Smoking, Cannabis
  - Others: comments

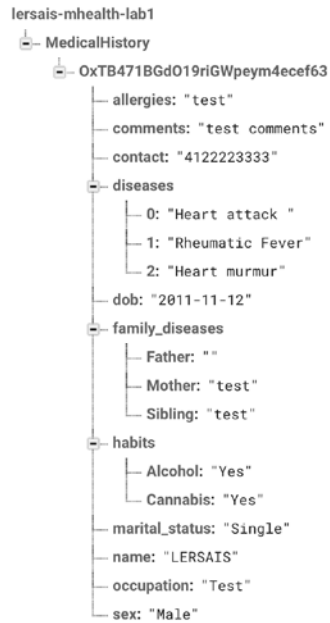
Here is a sample design view.

The image displays two side-by-side screenshots of an Android application titled "mHealth Project 1". Both screenshots show a "Medical Record Edit" form. The left screenshot shows the "General" section with fields for Name, Date of Birth, Sex (Male/Female), Marital Status (Single/Married/Divorced/Widowed), Occupation, and Contact. Below this is the "Past Medical History" section with an "Allergies" field and checkboxes for Heart attack, Rheumatic Fever, and Heart murmur. The "Family Diseases" section is partially visible at the bottom. The right screenshot shows the "Family Diseases" section with input fields for Father, Mother, and Sibling. Below that is the "Habits" section with radio buttons for Alcohol and Cannabis (Yes/No). The "Others" section has a "Comments" field. A "SAVE" button is located at the bottom right of the form.

- The medical record should be stored in Firebase Database
  - Firebase database has been integrated.
  - Use reference: <https://firebase.google.com/docs/database/android/start/> to learn how to use Firebase Database.
  - Your data should be stored in the root "MedicalHistory/[UID of user]". Here is an example.



School of Computing and Information  
University of Pittsburgh



- You can design the specific structure to store the medical record data under "MedicalHistory/[UID of user]/" by yourself.