# Project Outline for Software Design II (SE3A04)
## Mobile Medical Information System Application

Last Updated: September 9, 2013

With the advent of mobile operating systems, such as Android, iOS, and BlackBerry, the development of software applications for mobile devices has become increasingly popular. The aim of this project is to specify, design, and implement an application for a mobile device, in particular, one based on the Android platform.

## 1 Project Outline

Your local Medical Centre has recently taken steps towards transitioning to keeping digital medical records and is looking for solutions in the form of a mobile application which can manage the information. The idea is that this application will be available to a variety of users including doctors, nurses, administrators, patients, and perhaps others. The primary concern of the Medical Centre is related to the security of the information as it must maintain the confidentiality, integrity, and availability of the information. Since a variety of users will use the application, measures must be taken so that only those users which are authorised to access the information are able to do so. For example, a doctor may be able to access more information than a nurse, and a doctor may able to modify the information while a patient may not.

Suppose that you and your development team have been contracted to specify, design, and implement a mobile medical information system for the Medical Centre.

The software should meet the following minimum requirements:

1. The application must store the medical information in an encrypted form using the cryptosystem of your choice. This information will include patient medical records and histories, patient charts, appointment dates and times, etc.
2. The application must implement some kind of access control (e.g., Access Control Matrix, The Bell-LaPadula Model, Biba's Model) to ensure that only those users which are authorised to access the information are able to do so.
3. The application must allow for authorised users to create, modify, and remove patients, charts, appointments, etc. For example, only an administrator may add users to the system, edit user profiles, or remove users, while only doctors may create, edit, or delete patient charts.
4. The application must use the information stored for a particular patient to generate reminders for patients to schedule appointments. For example, the medical history of a patient may contain the date of their last tetanus booster. Using this information and the current date, the application must remind the patient and their doctor that an appointment should be scheduled to update the tetanus booster when required.
5. The application must provide at least one plot allowing for the visualisation of some kind of data stored in the medical information database (e.g., patient weight vs. time).

***Note*** The implementation of the requirements listed above constitutes the minimum requirements for project completeness. Any additional functionality will be rewarded. Each team is ***required*** to ***implement at least one additional innovative feature*** not mentioned in the outline. Creativity will be generously rewarded.

# 2 Technology

The Android Software Development Kit (SDK) is natively written in Java. However, there exists the Scripting Layer for Android (SL4A) which allows you to develop your android applications in scripting languages such as Python.

You are free to use Java, Python, or another scripting language supported by SL4A, as long as it meets the specified requirements. However, we are better equipped to provide you technical support with Java.

# 3 Tools

The **Android SDK** has the tools, sample code, and documentation you will need to create applications. The Android SDK is available at: `http://developer.android.com/sdk/index.html`.

If you plan to implement your application using Python, you will need the **Scripting Layer for Android (SL4A)**. SL4A provides interactive interpreters, script editing, and script execution for various scripting languages. SL4A is available at: `http://code.google.com/p/android-scripting/`.

You are free to use any software design tool, i.e., StarUML, Visio, etc. and any version management tool, i.e., SVN, Google Code, SF.net, etc., as long as it meets the project requirements.

# 4 Resources

***Note*** A large portion of this project is to do the required background research on working with the Android platform and information security. Keep in mind that a substantial component of any software project is to solve and/or eliminate the underlying technical difficulties. This often means hitting manuals and Google. To start you off, we provide a narrow selection of references and resources.

## 4.1 Android, Java, and Python

There are numerous online resources, including code samples, tutorials, screen-casts, etc. to help you get started with the development of your Android Application.

For those of you wishing to implement your application in Java, a good reference is the following:

1. Shane Conder and Lauren Darcey. *Learn Java for Android Development*. Tuts+. 2010.
   Available: `http://mobile.tutsplus.com/series/learn-java-android-development/`
   - This is a 13 part series of tutorials to help you get started with Java with a focus on developing Android applications.

For those of you wishing to script your application in Python, some good references are:

1. Paul Ferrill. *Pro Android Python with SL4A*. APress. 2011.
   - This book is available free online in PDF form.
2. Paul Barry. *Python for Android*. Linux Journal. 2011.
   Available: `http://www.linuxjournal.com/article/10940`

- This is a good beginners tutorial to help install the Android SDK, SL4A, and how to configure things to get them to work. The tutorial is written for Linux but can easily be used for Windows and Mac OS X as well.

For general reference, the Android Developers site is a good place to look:
`http://developer.android.com/index.html`

### 4.2 Information Security

There is a wide variety of resources to help familiarise yourself with information security including access control and encryption/decryption algorithms. We recommend the following textbooks to give you a general overview of many of the concepts required to complete this project.

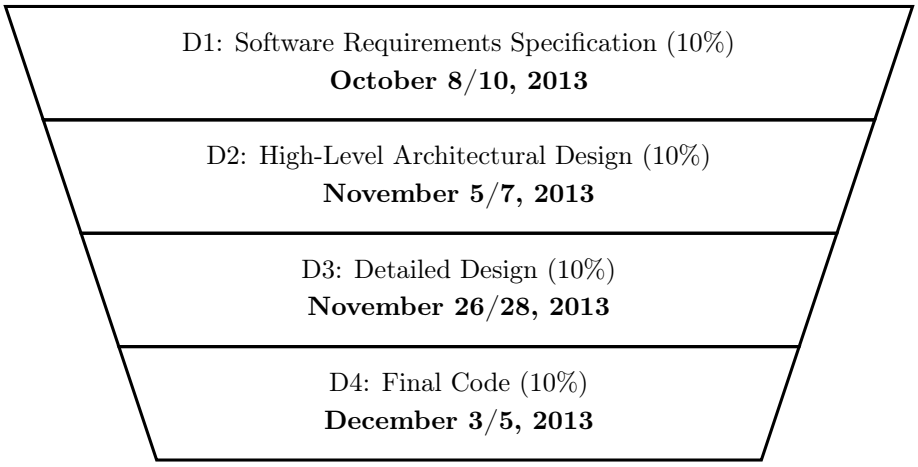
1. Matt Bishop. *Computer Security: Art and Science*. Addison-Wesley. 2002.
   - In this book, you will find information about
     - Access Control Matrix: Chapter 2 (pages 31–46)
     - Confidentiality Policies: Chapter 5 (pages 123–150)
     - Integrity Policies: Chapter 6 (pages 151–167)
     - Basic Cryptography: Chapter 9 (pages 217–241)

***Note*** Please keep in mind that you can find many of these ideas online in the form of tutorials and lecture notes, among others.

## 5 Deliverables

Every team will submit one team copy of each of the following deliverables on the specified due date. Each deliverable must have a list of the contributions of each team member and must be signed and agreed by each team member.

***Note*** The due dates may be adjusted by the instructor as circumstances dictate.



***Note*** The project described in each deliverable may be a subset of the project described in the previous deliverable. You may cut features as you progress through the semester. This idea is illustrated in the inverted trapezoid diagram. However, the final implementation must contain ***at least*** the ***minimum project requirements*** as specified above, plus ***one additional innovative feature*** of your own design. Creativity and additional effort will be rewarded, so think big at the beginning and keep your options open as long as possible.

# 6 Log Books

Each team member is expected to keep a ***hand-written log book*** that contains all the notes of the work done during the team meetings and your individual contributions to the project done on your own time. Log books will be randomly requested to evaluate a student's participation during the course of the project, and may be used as a resource in the event of a dispute within a group. Be sure to bring your log book to all tutorial sessions. Remember that the log book ***must*** be a physical book.