Module: Mobile Application development (Android)

Session 8: Introduction to Andriod (Practice)

**Objectives**

1.      Create a reasonably simple Android application

2.      Understand the basics of an Android framework

3.      Become familiar with the Android Studio development environment

**Introductory Notes**

The text book is based on a five day training course given in industry to developers. It is a good example of what is expected of technical personnel.  A company flies an employee to a training site to receive a week of intensive training. Lectures go most of the day and, in the evenings, those attending will work through the tutorial assignments. At the end of the training sessions, the company expects that their employees are ready to hit the ground running and be immediately productive.

Of course, there are prerequisites. Most Android development is done in Java, though the framework for projects is very different than we might be accustomed to. There is no AWT, Swing, or Java FX. The life-cycle of applications are very different than those that are desktop/laptop based. Terminology is completely different so though we speak Java, the concepts will sometimes seem very strange. This being said, it is important to have a solid foundation is Java. Essential concepts are classes, objects, interfaces, anonymous listeners, inheritance, and polymorphism. Basic constructs like declaring and accessing variables/arrays and control structures (decisions/loops) should be second nature.

The bottom line is that we will be taking this five day course, but spreading it over a ten week quarter. The effort that you put into it will determine how much you get from it. Hopefully, the course will be challenging, but rewarding.

**The Development Environment**

The accepted development environment for Android is Android Studio. This was not the case when the text book was written (2013). At that time, most Android developers used Eclipse with plugins. The Android SDK (Software Development Kit) was a separate download. With Android Studio, everything is packaged in a single install and the download is available at<https://developer.android.com/sdk/index.html>. We need to install SDK versions 8 (Froyo), 16 (Jelly Bean), and the current version 22 or 23 (Lollipop) plus any additional development tools. Once the studio is installed, from Android Studio, click on Tools, then Android, then SDK Manager. You can select and then install what is needed.

**Bring you Notebook computer to class; we will be spending some time doing hands-on work.**

**Android Studio versus Eclipse**

Android Studio is very similar to Eclipse, which is good. However, there are some differences, which can be annoying. This will become apparent as we work through the book examples. In most cases, Google helps. Everything is there but the short keys and menu clicks don’t always match. Some of those I noticed as I did the labs this past summer follow (it is not an all-inclusive list).

a)      Displaying line numbers: file -> settings -> editor -> appearance -> show line numbers check box

b)      Undo is ^z (like Eclipse), but ^y is not redo (it deletes a line). For redo ^shift-z

c)      To refresh after copying files, click the refresh icon (circle with arrows) at the top left of the Studio development window. If you right click on the project there is also a synchronize option. However, this only applies to a single directory or file.

d)      Sometimes, when you add directories, you won’t see them in the navigator window on the left. There is a drop down at the top. Choose Project view instead of Android view to see all directories and sub-directories.

e)      To include needed imports, don’t use ^shift-o. Instead, click on the red reference to a class requiring the import and then click alt-enter.

f)       Note that the entry screens will differ somewhat from those of the text. In most cases, it is not too difficult to adjust to the changes. In a couple of cases, it takes some time to figure out the differences. Google helps. One that I found caused me trouble is file->new->directory or file->new->file. In Android Studio file->Android resource file or file->Android resource directory is needed because all project references are automatically updated. Otherwise, you get undefined warnings.

**Debug on an actual device**

You can debug most of the projects using the built-in Android emulator. However, sometimes it is better to use an actual device. This is especially true before submitting an app to Google-store for product approval. The following steps document the set up.

a)      On the device: settings -> about phone/tablet -> tap build seven times  to register as a developer  
settings->developer options->usb debugging  
setting->security->install from unknown sources

Settings->storage->top right settings button->ptp (camera mode)

    Note: You might have to change back when you want to be able to view the device file system

b)      In the Android project, add to the second gradle script file under the builds tag  
debug {  
    debuggable true  
}

c)      Connect the device using a usb cable

d)      When you debug, a dialog will appear showing the available devices. Select the connected device, not those listed for emulator execution.

**Part 1:  Programming**

Work through the GeoQuiz tutorial in chapters one through six. Complete all of the challenges at the end of chapters two and five.

Chapter four describes how to debug applications; if you feel confident in your debugging skills, you can skip that chapter. Otherwise, it is very valuable, describing basic techniques that are universal, no matter what language and environment you are using.

Chapter six has only one programming change that is described in listings 6.2-6.4. The rest of the chapter is relatively light reading. Note that the statistics describing version use are now very much out of date.

**Part 2: Synthesis questions**(Make sure to include the questions with your answers)

1.      What are major differences between Minimum Required SDK, Target SDK, compile with SDK, and Maximum SDK? How would you set each in a new project application?

2.      Define the following Android terms

a.      Context

b.      Activity

c.       Bundle

d.      Intent

3.      What are advantages to defining string values within the res/values folder? Which strings should you define there?

4.      How do you create a new resource layout? Be specific.

5.      How do you determine which SDK version is running?

6.      Define the term: model view controller. How is that applied in the GeoQuiz application?

7.      How is xml used in Android? Give two examples.

8.      What is the difference between the fill\_parent and match\_parent attributes?

9.      How are anonymous inner classes used in the GeoQuiz application?

10.  How is information passed between activities?

11.  What is the major difference between an explicit intent, and an implicit intent? How are each used in this application?

12.  How do you automatically generate getters and setters? Be specific.

13.  What bugs occur in the original implementation when the display is rotated?

14.  In chapter 6, Listing 6.4 has a problem. How did you get it to work?

15.  List five Android-based widgets and four attributes that go with each of them (other than width and height)?

16.  Describe the life cycle of an Android application.

17.  In the method, startActivityForResult(), what is the second argument for? How might this be a useful feature?

18.  What is the purpose of the method, onCreateOptionsMenu()?

19.  What are major differences between FrameLayout, RelativeLayout, and LinearLayout?

20.  What are the required attributes for all widgets?