Introduction to Mobile Device Programming

M3G621212

Lab Session 1

This document assumes that you are sitting in front of a computer which has Eclipse with the Android Plug-in already installed. In the University labs you will find Eclipse and the Android Software Development Kit installed in the VMWare image SEB-VM-Win7x64.2014v1.

Once this image is running start up Eclipse for Android. There should be an "Eclipse" icon on the VMWare Desktop.

In this lab session you will be require to locate, produce and deploy a number of small Android Applications. If you have an Android phone you will be able to deploy the apps to your phone as well. In order to do this you will need an appropriate cable to connect your phone to the computer running Eclipse.

REMEMBER the programs that you create will be lost if you leave them on the lab machine. The lab machines reboot every night and your work will be wiped! Make sure that you copy any projects to a USB memory stick or to a portable hard drive.

Task 1

Your first task is to Run an Android Project on a particular Android Virtual Device (AVD). The AVD is an emulator which allows you to run an application on a software "simulator" of an Android mobile device. There are a number of Android AVDs' set up in the lab image. These emulators are summarised below.

AVD	Operating System
android2p1	Android 2.1
android2p2	Android 2.2
android4p2	Android 4.2
android4p4	Android 4.4

If Eclipse is not running start it now. Accept the default location for the workspace.

Locate ITMDPLabProject1. You should find this on Blackboard as a zip file. Download and unpack this file to a suitable location.

Now use the Import feature from the file menu to get the file into the Android Workspace. Once the project has loaded in the Eclipse environment right click on the project name and select run as Android Application. The project should now automatically deploy to a specific virtual machine. The virtual machine is specified in project properties. If the specific emulator is not running it should be started for you. In this case the application should deploy to androidavd4p2 which is an Android Virtual machine set up to emulate the Android Operating System for version 4.2.

Note you may have to redeploy the application once the emulator is running because the deployment may "time out" if the emulator is slow to start up.

Once the application is running view the Java source code and answer the following questions.

Q1

What is the name of the object used to display the message "Welcome to Mobile Device programming?

What is the class of the object?

Investigate this class and write a few sentences outlining the behaviour that this class displays.

Q2

What is the name of the object used to Display the text "Press to Exit"?

What is the class of the object?

Investigate this class and write a few sentences outlining the behaviour that this class displays.

What is the signature of the piece of code that handles the click event on this object?

What happens when this object with the text "Press to Exit" is clicked (you would touch this on a real Android device!)

Q3

Now locate the xml file main.xml which is in the layout folder of your Android project. Look at the entries in this file and think through how this relates to the layout that you viewed when the application ran. Write a few sentences describing how the graphical components are arranged.

Task 2

You can start an emulator directly without deploying a project to it. You do this by starting a "DOS" session. Click on the windows symbol on the bottom left of the screen and enter cmd in the search box. Now click on cmd which should appear as the only entry in the list of programs.

So if you have created an Android Virtual device called android4p2 you would run the emulator from the command line like this.

emulator -scale 0.75 –avd android4p4

This command works from the command line if the PATH environment variable in the labs includes the appropriate directory in the android/eclipse installation which contains the "emulator" command. The "settings" for each emulator are kept in separate files.

If the PATH environment variable does not include the Android installations directory you will need to follow the instructions below.

The Android installation directory is likely to be

c:\eclipseandroid\adt-bundle-windows-x86_64-20140321

At the Dos command prompt type the following command and press return.

cd c:\eclipseandroid\adt-bundle-windows-x86_64-20140321\sdk\tools

Now type :

emulator -scale 0.75 –avd android4p4

Alternatively you can change directory one level at a time. See below for sequence.

cd c:\
cd eclipseandroid
cd adt-bundle-windows-x86_64-20140321
cd sdk
cd tools

Now type :

emulator -scale 0.75 –avd android4p4

The scale factor is used to ensure that the emulator appears properly on your screen. Some of the "skins" available when you create the virtual device are too large for the screen.

See supplementary lab notes Section 2 for more details.

Be patient though it may take a few minutes for the emulator to start up completely. You are advised not to close the emulator down during the lab session but to keep it running and just redeploy your Android app to the emulator when you have made changes.

Once the emulator is up and running you can deploy other projects to the emulator.

Locate ITMDPLabProject2. You should find this on Blackboard as a zip file. Download and unpack this file to a suitable location.

Now use the Import feature from the file menu to get the file into the Android Workspace.

Once the project has loaded Make sure that project properties for Run/Debug are set appropriately in the Eclipse environment. To do this Right click on the project | click on Run/Debug Settings | Click on ITMDPLabProject2 | Edit | Target | android4p4 should be ticked. If not do so and the select Apply | Ok

Once you have checked the Run/Debug settings right click on the project name and select run as Android Application. This time your app should deploy to android4p4.

View the application once it is running and answer the following questions.

Q1

What is the purpose of the object called nameEntry?

What is the class of this object?

Q2

Now locate the xml file main.xml which is in the layout folder within the project. Write a few sentences about how the xml code arranges the graphical components on the screen.

Task 3

You can create your own virtual devices onto which you can deploy your Android project. To do this read section 1.1 of Supplementary Lab Notes on the Android SDK and the Eclipse Integrated Development Environment. If you work your way through section 1.1 you should now have created an Android Virtual Device called labavd1. This will allow you to customise the Emulator in specific ways. In the next task you will create a new Android project and deploy your application to the emulator that you have created.

Task 4

In this task you will simply create a project and deploy this project to the emulator you created in Task 3. See section 3.5 of Supplementary Lab Notes. Call the project LabTest1, select a Build Target of Android 4.2 and enter org.me.myandroidstuff for the package name and then click Finish. This sequence of actions generates a very simple Android application.

If you have been successful you should have an entry for LabTest1 in the package explorer.

In Eclipse, right click on the Entry LabTest1 and select Properties | Run debug settings | New|Android Application|androidtest4p2 for name and LabTest1 for Project|Click Target and select labavd1 for virtual device|Apply | OK | OK

Right Click on LabTest1 again but this time select Run As | Android Application. It may take a few minutes for the application to deploy so be patient. If all is well you will have deployed your app to the emulator labavd1

If all is well you should be able to see a message such as that shown below at the top of the emulator screen.

Hello World LabTest1

The target platform for the applications that we develop will be Android 4.2 (Jelly Bean)

Once the application is running answer the following questions.

Q1

What is the type of the Java class that was generated by the "wizard" in Eclipse?

Investigate this class and write a few sentences about what it provides the developer with.

Q2

What methods are generated by the wizard with the class that you identified in Q1? Can you work out what each method is responsible for?

Task 5

Now that you have generated an Android application you need to spend a bit of time looking through the files that make up the project. Use the document **"Android Development – The Basic Overall Picture of an Android Application"** to help you in this process. You should have been given a copy of this document. You will also find it on GCULearn in the lab area for this module.