IEMS 5722

Mobile Network Programming and Distributed Server Architecture

2015-2016 Semester 2

Assignment 0: Android Development Basics

Due Date: 29th January, 2016 (Friday)

Notes

i.) You are strongly advised to go through all the steps in this assignment in order to get familiar with the Android development environment

ii.) Pay attention to Section 4, where you are required to perform some tasks

iii.) See the instructions at the end of this assignment, follow them to submit your files for marking

iv.) Late submissions will receive 30% mark penalty

v.) This assignment accounts for 2% of your final grade.

1. Aim

To learn the fundamentals of Android development from project creation to installation on a virtual or physical device

2. Objectives

- Set up the Android development environment
- Create a Hello World Android application
- Understand the Android project structure
- How to run an app in the emulator
- How to deploy the app to a physical device

3. Instructions

3.1 Setting up the development environment

Please ensure you have at least **JDK 7** installed before continuing. Note that JDK and JRE are different. JDK is required for development. You can get the latest version from http://www.oracle.com/technetwork/java/javase/downloads/index.html

3.1.1 Android Studio

The Android Studio is the standard environment for developing Android app development and is the recommended option for this course.

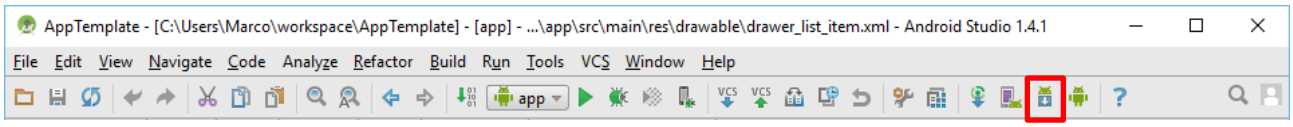
Download from http://developer.android.com/sdk/installing/index.html?pkg=studio

Once you have downloaded the package, follow the online instructions to install it on your computer. When you start Android Studio for the first time, it will ask you to specify a workspace. This is where your project's files will be saved. You can choose a default location to use or change it later if you want.

3.1.3 Installing the Android SDK Packages

Android Studio does not contain all the packages needed for app development yet. These packages can be downloaded using an integrated tool called the Android SDK Manager.

In Android Studio, click **SDK Manager** in the toolbar. Its icon is highlighted in the below screenshot.



Click on Android SDK Manager to show the different packages available. By default, the Android Studio includes the current version, but you will need to download other versions if you want to build your app against older Android versions. Targeting earlier versions let you reach a larger number of users. You will also need to use the SDK manager to download later versions of the API when they are released in the future.

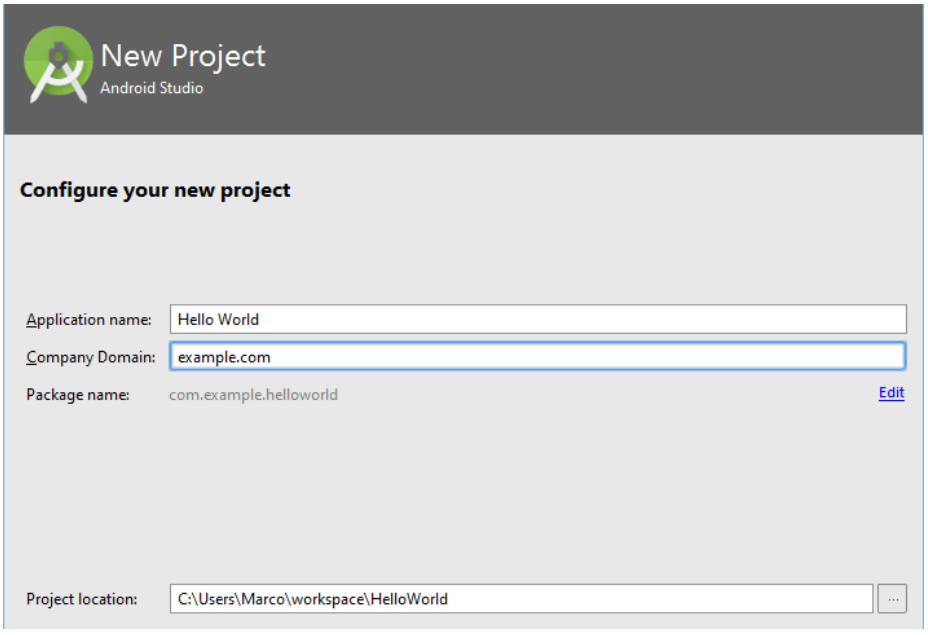
For this course, you will develop apps that support Android 4.0 (API 15) or later versions. You can download the SDK Platform for API 15 and later In addition, download the Android SDK Tools, Android SDK Platform-tools and Android SDK Build-tools (latest version).

Android Studio may need to be closed in order for Android SDK Manager to install the new packages.

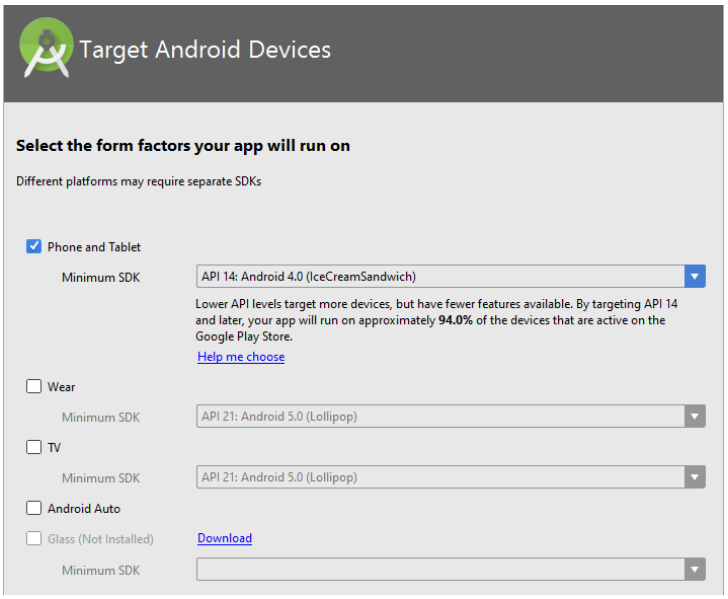
3.2. Creating your first app

To create an Android project, go to ***File*** > ***New*** > ***New Project***

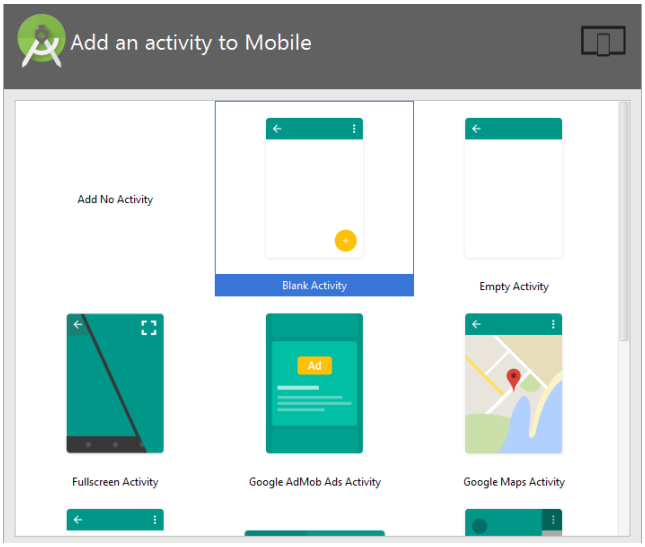
The New Android Application dialog appears. In the Application Name field, enter “Hello World”. This is the name that shows up in the ***Android application launcher***.
Enter your company domain to finalize your app’s package name following the Java package naming rules as seen in below.



Click Next to select what types Android devices and versions to support. For this exercise, only the Phone and Tablet option needs to be checked. The ***Minimum Required SDK*** is the lowest version of Android that your app supports. Lower versions target more devices, but less SDK features will be available. Set this to API 14 for this exercise.

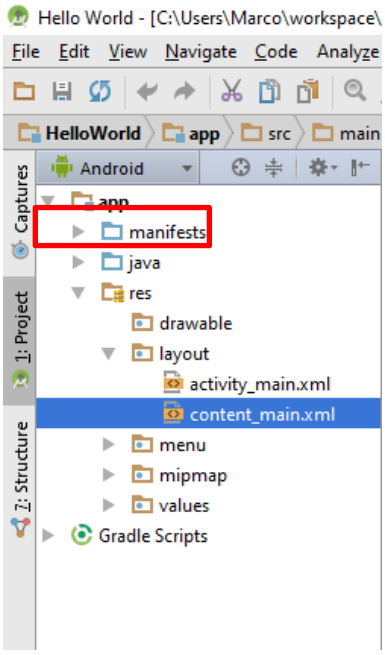


When you click next, it will let you customize the initial app template options such as having navigation menus or full screen views. We will use the Blank Activity for this example - this will create your project in your workspace with an empty activity. An example of the dialog is seen below. You are encouraged to try out the other ones to see how they work.



Click next to set up the activity. The blank activity has four options - the ***Activity Name***, ***Layout Name***, ***Title*** and the ***Menu Resource Name***. This screen defines your initial activity - when your app is launched, this activity and layout will be the entry point into your app.
You can leave these as default for now. Click Finish to start your project.

You should be back in Android Studio with a new HelloWorld project files visible in the ***Android*** project view panel (right).



If you choose a different option in the Android menu highlighted in red, you can visualize your project resources differently. This is useful depending on what you want to see (e.g. The *Project* option most closely resembles the structure from Windows Explorer)

To see what Android Studio has created, explore the *project tree*.

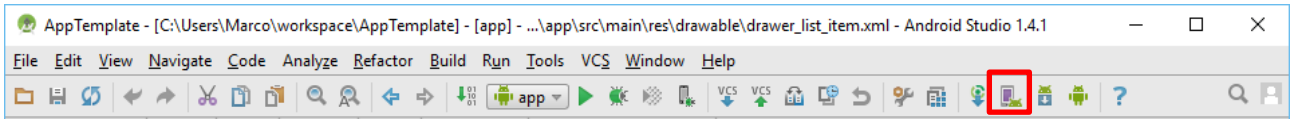
Note it has created a HelloWorld project, using the project settings we chose previously. The *java* folder contains the Java source files, in the package *com.example.helloworld*.

Likewise, the *activity_main.xml* file has been created in the *res/layout* folder.

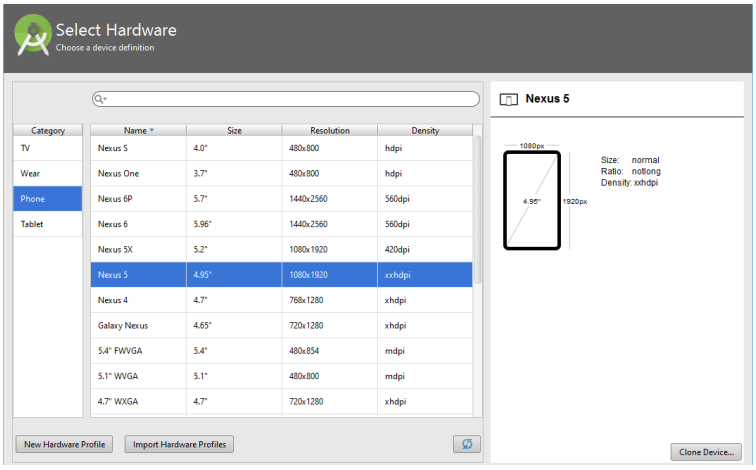
3.3 Running your app in the emulator

3.3.1 Create an emulator profile

In Android Studio, go to ***Android Virtual Device (AVD) Manager*** (button highlighted below). From here you can create, edit and delete your emulator profiles. Click on the Create Virtual Device button to define a new emulator profile.



Emulator profiles allow you to test your app with different settings so you can see how your app behaves even if you don't have a particular device physically. A Nexus 5 is selected below. This is sufficient for this exercise - leave the options as default and create the profile. If you want to simulate a particular device, you will need to look up the particular devices profile.



3.3.2 Starting the emulator

In the ***AVD Manager window***, choose the emulator you wish to run and click the Start action. This will bring up the Launch Options window, where you can scale the emulator display size. You may need to experiment with this to fit your display best.

When you click Launch again, the emulator will begin. This may be quite slow to start, but you will not need to close the emulator until you are finished.

3.3.3 Deploying the app

In Android Studio, choose ***Run*** > ***Run App***. It may ask you where do you want to deploy your app to if you have more than one emulator running or if your phone is also plugged in. Choose the emulator.

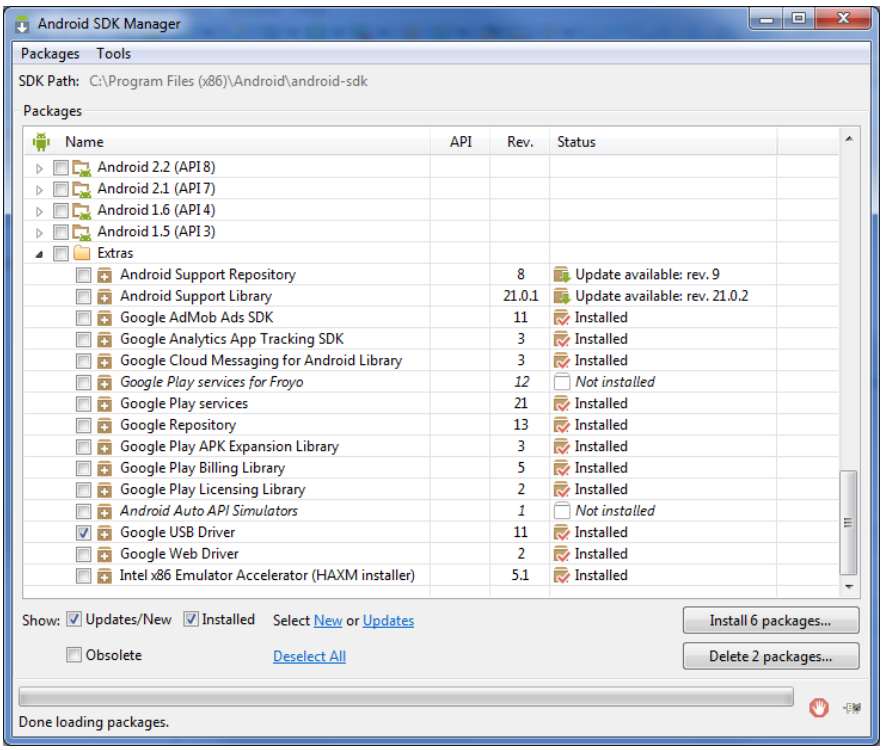
If you modify your app, you can deploy it again to the emulator without having to restart the emulator. Just choose ***Run*** > ***Run App*** again.

3.4 Running your app on a physical device

3.4.1 Setting up your computer

You will need an Android device and a USB cable to connect your device to your computer.

Install ***Google USB Driver*** from the ***Android SDK Manager***. This will be at the near the bottom of the list, under the ***Extras*** folder. The figure shown below has the Google USB driver checked. Note that some Android devices do not need the Google USB driver for development. The Galaxy Nexus for example, uses the OEM driver. Consult your manufacturer documentation if in doubt.



3.4.2 Setting up your Android device

Devices running Android 4.2+ have developer mode disabled by default. To enable it on your device, go to ***Settings*** > ***About Phone*** > ***Build Number***. On some devices this may be under ***Settings*** > ***About Phone*** > ***Software Information*** > ***Build Number***.

Tap on Build Number **7 times**. There should be a pop up notification saying “you are x steps away from being a developer” which counts down after two taps.

If you were successful, the ***Developer Options*** menu should be visible under ***Settings***.

3.4.3 Deploying your app

Connect your device to your laptop using the USB cable. On your device, you may need to switch to ***MTP mode***.
In Android Studio, right click on your project and select ***Run*** > ***Run App***. A window prompt will appear asking you where you want to deploy it to. Your physical device should appear in the list. Select your device then click deploy.

If this is the first time you have deployed to your phone from this computer, your phone will ask you if you want to allow apps to be installed from this source. You can also save this source so it won’t ask you again next time. Accept the connection to continue installation.

The app will run when automatically when it has been installed.

4. Tasks

4.1 Modifying your app

4.1.1 Change the text

You will notice that the app shows “Hello World” already. Look through the app files generated and find where and how the text has been displayed.
Then add in another line of text, showing your name and student number. ***DO NOT*** hardcode your text in the content_main.xml file or the Java source file - use the **string resources**. As a challenge, try to put it in your text **in the center of the screen**.

Hint - Start from MainActivity.java and look at what layout it references. Then look in the layout and look at what string resource it uses.

4.1.2 Changing the app icon

The app has launch icon by default. This is stored in the ***res/drawable*** folder, which is where graphics such as icons and images can be stored.
There are several drawable folders corresponding to different resolution devices: Low DPI, Mid DPI, High DPI and Extra High DPI. Android chooses the files most appropriate to the device that is running the app. By default, Android Studio has not created these folders but you should use these in your apps.

If the file is missing from a particular folder, it will choose the image from the next best available resolution. For example, a High resolution device will try to source its images from the ***res/drawable-hdpi*** folder. If it is cannot find a particular image, it will look in the ***res/drawable-xhdpi*** folder and scale down the resolution to fit. If not successful, it will then look in the ***res/drawable-mdpi*** folder and scale up the resolution. This is not particularly beautiful so you should ensure that you provide a low/med/high/extra high resolution image for your app.

Change your apps icon to be the **cuhk.png** but do not rename the file. For this exercise, you do not need to create logos of different resolutions.

5. Submission

To submission your assignment, compress the whole project folder into a ZIP file and name it in the following format: **<your_student_id>_assgn0** (e.g. 1234567689_assgn0). Submit it in the CUHK eLearning System online:
https://elearn.cuhk.edu.hk/

6. Resources

- Google Android Training
  https://developer.android.com/training/index.html
- Android Style Resources
  https://developer.android.com/design/downloads/index.html
- Android API Reference
  http://developer.android.com/reference/packages.html
- Where to go if you have a question (apart from Google):
  http://stackoverflow.com/questions/tagged/android