

Final Year Project

Background Research

02/11/2015

**Server/Client Progress** - *Which OS should I choose and why?*

**Raspbian vs Ubuntu**

Originally I decided that because I have experience using Ubuntu and working with the terminal that I would download the latest image of Ubuntu for my raspberry pi. However, after some research I found the OS called raspbian which was developed back in 2012 around the time the first pi was released. Commonly referred to as ‘distro’, raspbian is the most common OS used for the pi. I feel it would benefit me if I choose this OS because it would give me a chance to learn a new OS. Also if I were to run into any problems with setting up the OS/Server, I would have a better chance finding the solution online due to the OS being the most common.

I read [[1]](#endnote-1)online that raspbian comes with APT (Advance Packaging Tool) and raspo-config. These tools allow me to install the latest software from raspbian repositories and also simplify the managing of rapberry pi configurations such as setting up ssh and enabling raspberry pi cameras. So I decided to download the latest [[2]](#endnote-2)version of raspbian which was named Jessie.

The only main difference between version Jessie and the previous version (whezzy) is the OS now boots straight to desktop instead of the command line. The update has also reported the OS to be more stable.

**Setup Process**

Since it was my first time setting up and using the pi, I decided to have a look for some online help.

I came across a [[3]](#endnote-3)tutorial which showed me how to install Jessie onto an SD card through the terminal of my Ubuntu OS.

Steps Involved:

* I had to locate the image which I earlier downloaded and change directories where the location of the image was. (cd /image location/)
* Next I had to view the list of devices which were connected to my machine. Once I could locate the SD card, I could then unmount the card and use the DD tool write the image to it.  
  (ddb bs=1M if=image/name of=location of card) - bite size set to 1 and input/output file set.

Now that my OS was set up, I was able to attempt to SSH to the pi and begin with the server/client connection.



**SSH Progress**

**Gstreamer (framework)**

Upon researching media streaming, I came across a framework called [[4]](#endnote-4)Gstreamer. With the gstreamer framework, the developer is able to build any type of streaming multimedia application.

The thing that interests me the most about gstreamer is that it handles pluggable components which can be matched into pipelines. These pipelines can then be edited with a GUI editor (application - record/move cam). This means developers can write their own video editing application.

Gstreamer basically handles the connection between device plugins.

**Android vs Apple**

There are a couple of reasons as to why I picked android over apple. The main reason is because I have previous experience working with android development and enjoyed working with java. I also tried learning python before and found it slightly more difficult. I own an android phone which also means I can test my application when needed.

*(Eclipse vs Android studio)* – *Which one and why?*

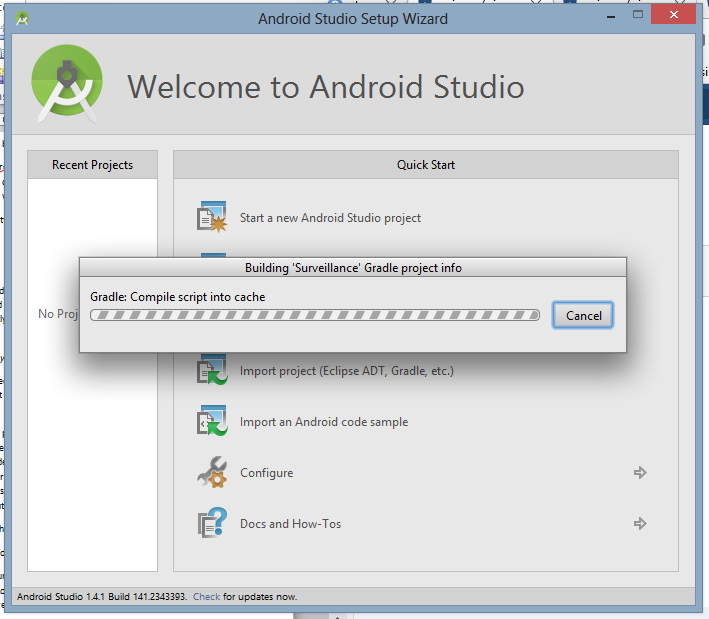
I originally planned to have developed my project using Eclipse with androids sdk tools because of previous experience. However, I was told about androids new development kit and felt it may be of use to my project.

We have no more need for work spaces or sub projects like used in eclipse instead android studio has *modules* and *library modules*. These modules are very useful features to the kit as it allows developers to run, test and debug functions independently. A file called gardle build is generated oupon creating a project. This file contains information about the project such as the android version range of the app or even the apps dependencies when used by a device. The project also includes Junit testing which I plan to be using throughout the application.

After downloading the software and finishing the installation, I open up a new project.

The process in creating your own project is as follows:

* Select the form factors your app will run on (What machine and minimum SDK requirement)
* Adding an activity to the module (eg. Login activity)
* Lastly, choosing the activity name if already not picked



1. http://www.element14.com/community/polls/2103 [↑](#endnote-ref-1)
2. https://www.raspberrypi.org/downloads/raspbian/ [↑](#endnote-ref-2)
3. https://www.youtube.com/watch?v=pqt [↑](#endnote-ref-3)
4. http://gstreamer.freedesktop.org/ [↑](#endnote-ref-4)