Android Development Setup Instructions

Introduction

When writing code for Android we chose to utilize the Kivy development kit for python. This choice originated from the cross platform nature of python, the automated Android APK build tool that kivy provides, and also the optional virtual machine which is already setup with the development tools.

Installation Instructions:

1. Download 'Oracle Virtual Box' and install the program.

2. From the Kivy website, download and extract the Virtual Machine Image (.7 compressed).

3. Add the extracted VM Image to VirtualBox, change any settings if wanted, and start it.

4. Building your hello-world .APK program.

4A. Run: "sudo pip install -U buildozer" (update buildozer)

4B. Run "rm -rf ~/.buildozer/android/packages" (remove any previous build remnants)

4C. Run: "buildozer init" (setup new build .spec file)

\_\_version\_\_ = '1.2.0'

from kivy.app import App

from kivy.uix.scatter import Scatter

from kivy.uix.label import Label

from kivy.uix.floatlayout import FloatLayout

class TutorialApp(App):

def build(self):

f = FloatLayout()

s = Scatter()

l = Label(text="Hello!", font\_size=150)

f.add\_widget(s)

s.add\_widget(l)

return f

if \_\_name\_\_ == "\_\_main\_\_":

TutorialApp().run()

4D. Create a "main.py" at $HOME with code:

4E. From the terminal window at $HOME Run: "buildozer android debug"

5. If the above returns an error related to "distribute.sh" buildozer needs to be updated/cleared.

5A. Run: "rm -Rf .buildozer/" (clears out user application directory)

5B. If an error regarding: "Adji" not found. Run "rm -Rf $HOME/.android"

6. The Android APK file will be created in "./bin/"

7. Copy the file to the target Android device.

8. On the device navigate to, and install the program.

Appendix: Recon Instruments Snow2 Specific Installation Instructions.

Introduction

Because of the minimal interface the use cases of the Recon Instruments devices require, as described on their developer information section of their website, the Android 'adb' utility needs to be used to both copy and install the APK to the device. The following instructions will install the 'adb' utility and use it to copy and install the APK file to the device.

(Note although the 'adb' utility is not needed to copy to the internal storage of these devices, it is required to install, and the installation process copies and installs in one step.)

Installation Instructions:

Based off of: https://androidonlinux.wordpress.com/2013/05/12/setting-up-adb-on-linux/

1. Run: "sudo apt-get update"

2. Verify that: "java -version" returns the correct result (should already be installed).

3. Download Android SDK from "http://developer.android.com/sdk/index.html"

Via the command: "wget http://dl.google.com/android/android-sdk\_r24.4.1-linux.tgz"

4. Extract the download with: "tar -xvf android-sdk\_r24.4.1-linux.tgz"

5. Navigate to "./android-sdk-linux/tools/" and run the "./android" executable.

6. A GUI will appear and allow the selection of Android SDK tools to install.

For the Recon Instruments Snow2 will need "Android 4.1.2 (API 16)" in addition to the default selections. Others can be added if you have other Android platforms with other versions. When all required SDK elements are selected, click "Install # packages..." Install the default selected items after removing the .tgz downloaded earlier.

7A. From the "Devices" tab on the VM window, select the SNOW2 USB device to insert into the VM.

7B. Verify the addition by running: "lsusb" and the new USB ID should match the one inserted.

8. Add the usb device to the Android tool's list of devices.

8A. Run: sudo bash -c 'echo SUBSYSTEM=="usb", ATTR{idVendor}=="2523", ATTR{idProduct}=="xxxx", MODE="0666", GROUP=="plugdev" >> /etc/udev/rules.d/51-android.rules'

8B. Run: sudo bash -c 'echo 0x2523 >> $HOME/.android/adb\_usb.ini'

8C. Run: sudo $HOME/android-sdk-linux/tools/android update adb

8D. Run: sudo ./android-sdk-linux/platform-tools/adb kill-server

8E. Run: sudo ./android-sdk-linux/platform-tools/adb start-server

8F. Verify: cat $HOME/.android/adb\_usb.ini

9. Install the .APK to the Snow2 HUD.

9A. Run:

#BUILD server (linux mint) installation:

wget http://dl.google.com/android/android-sdk\_r24.4.1-linux.tgz &

sudo apt-get update

sudo apt-get -y install python-pip

sudo pip install -U buildozer

sudo add-apt-repository ppa:kivy-team/kivy

sudo apt-get -y install python-kivy

#sudo apt-get -y install python3-kivy

sudo apt-get -y install build-essential

sudo apt-get -y install zliblg-dev

sudo apt-get -y install zlib1g-dev

sudo apt-get -y install git

sudo apt-get -y install cython

sudo apt-get -y install default-jdk

#install any missing kivy dependancies

sudo apt-get -yf install

#wget ...android-SDK ...should be done.

tar -xvf android-sdk\_r24.4.1-linux.tgz

#launching GUI to install SNOW2 specific or other Android API versions

sudo $HOME/android-sdk-linux/tools/android &

#now select a crap tone of API and tool versions to install

#Now performing the dummy buildozer app creation:

echo \_\_version\_\_ = \'1.2.0\' Aidl> main.py

echo from kivy.app import App > main.py

echo from kivy.uix.scatter import Scatter > main.py

echo from kivy.uix.label import Label > main.py

echo from kivy.uix.floatlayout import FloatLayout > main.py

echo> main.py

echo class TutorialApp(App): > main.py

echo def build(self): > main.py

echo f = FloatLayout()> main.py

echo s = Scatter()> main.py

echo l = Label(text=\"Hello!\", font\_size=150) > main.py

echo f.add\_widget(s) > main.py

echo s.add\_widget(l) > main.py

echo return f > main.py

echo> main.py

echo if \_\_name\_\_ == "\_\_main\_\_": > main.py

echo TutorialApp().run()> main.py

sudo rm -Rf $HOME/.android

buildozer init

buildozer android debug