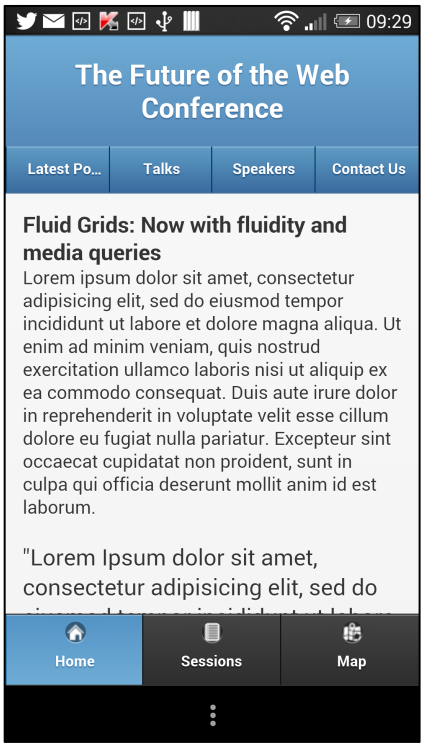
Worksheet D: Using PhoneGap and running on the Android emulator/device

Chris Loftus (cwl@aber.ac.uk)
June 2014

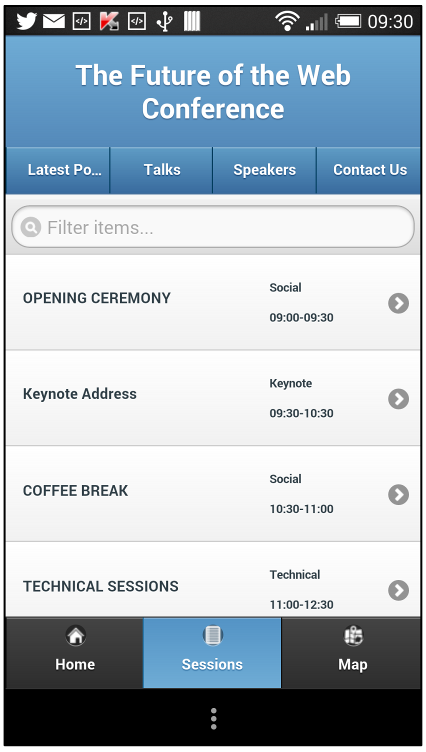
In this worksheet you will take the *with-jquery-mobile-solution* and package it so that it can be turned into a PhoneGap app. This will involve uploading to the PhoneGap Build website for building and packaging and then downloading the Android package which is then deploy on the Android emulator (or your Android phone). The solution to this exercise can be found in the folder: *worksheets/with-phonegap-solution*. A downloaded Android package file for the Conference PhoneGap app called *ConferenceBrowser-debug.apk* can be found in the worksheets folder.

We want to produce mobile screens that look like the following:



Fluid Grids: Now with fluidity and media queries
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

"Lorem Ipsum dolor sit amet, consectetur adipisicing elit, sed do



OPENING CEREMONY	Social	09:00-09:30
Keynote Address	Keynote	09:30-10:30
COFFEE BREAK	Social	10:30-11:00
TECHNICAL SESSIONS	Technical	11:00-12:30

Software Alliance Wales is a European funded project led by Swansea University in partnership with the University of Bangor, Glamorgan, Aberystwyth and the University of Wales Trinity Saint David.

Step One: Packaging the mobile app for PhoneGap

1. Use you folder navigator tool (e.g. Windows Explorer or Mac's Finder) to **copy** the *with-jquery-mobile-solution* and **giving** it the name *with-phonegap*.
2. From RubyMine, **open** the folder *with-phonegap* and then open the file *index.html*. A**dd** the following script element just before the *Controller.js* script element:

   ```
   <script src="phonegap.js" type="text/javascript"></script>
   ```

   This file will be added automatically by PhoneGap Build when you upload to their site.

3. **Create** the file *config.xml* within the *with-phonegap* folder. You can do this by using RubyMine: *File->New File*. This will open an empty file. This file will contain instructions that PhoneGap Build will use when creating build targets for Android, iOS etc
4. Within the empty *config.xml* file and **add** the following XML (please copy and paste from the electronic version of this worksheet). Ignore any XML errors regarding the URIs. The comments in the XML explain the purpose of each configuration statement. These are just a selection and many more are possible: see https://build.phonegap.com/docs/config-xml for a complete description:

   ```
   <?xml version="1.0" encoding="UTF-8"?>
   <!--
       This is the root element containing XML schema namespace settings, a unique
       id for this application (use reverse domain name to ensure unique) and a
       version attribute to identify the version of this app on PhoneGap Build. The
       versionCode attribute is optional and is Android specific: it's the version number
       shown on the Google Play store.
   -->
   <widget xmlns="http://www.w3.org/ns/widgets"
       xmlns:gap="http://phonegap.com/ns/1.0"
       id="uk.ac.aber.dcs.cwl.conference"
       versionCode="1"
       version="1.0.0">
     <!--
       The name and description are used in the PhoneGap Build website
     -->
     <name>Conference Browser</name>
     <description>Browse conference sessions and discover venues</description>

     <!--
        Identifies the author and their contact details. The email is displayed on the
        PhoneGap Build website
     -->
     <author href="http://users.aber.ac.uk/cwl"
          email="cwl@aber.ac.uk">
       Chris Loftus
     </author>

     <!--
       We can specify the icons to use for different platforms. In this case we
   ```

Software Alliance Wales is a European funded project led by Swansea University in partnership with the University of Bangor, Glamorgan, Aberystwyth and the University of Wales Trinity Saint David.

```
    have only included the Android icons (for the launcher screen) and at
    four levels of resolution so that the correct version will be used depending
    on the device screen. Note that PhoneGap will also look for a default icon
    in the project's top level folder called icon.png.
-->
<icon src="icons/android/ldpi.png" gap:platform="android" gap:density="ldpi" />
<icon src="icons/android/mdpi.png" gap:platform="android" gap:density="mdpi" />
<icon src="icons/android/hdpi.png" gap:platform="android" gap:density="hdpi" />
<icon src="icons/android/xhdpi.png" gap:platform="android" gap:density="xhdpi" />

<!--
   An example of a preference setting. This preference will prevent landscape mode.
   By default both orientations are allowed.
-->
<preference name="orientation" value="portrait"/>

<!--
    Use the following if you want a launch splash screen to appear briefly
 -->
<!--
<gap:splash src="icons/splash/splash2x.png" width="640" height="960"/>
<gap:splash src="icons/splash/splash.png" width="320" height="480"/>
-->

<!--
    For each kind of mobile service required we add a plugin element. PhoneGap (Cordova)
    provides an API that enables access to many kinds of services: see
       http://docs.phonegap.com/en/2.3.0/index.html
       Here are some examples (although in our app we don't use any)
  -->
 <!--
<plugin name="Geolocation" value="org.apache.cordova.GeoBroker"/>
<plugin name="Device" value="org.apache.cordova.Device" />
<plugin name="Capture" value="org.apache.cordova.Capture"/>
<plugin name="Storage" value="org.apache.cordova.Storage" />
-->

</widget>
```

Change the widget element's id attribute value to your reverse domain name, although it can be anything unique. Also **change** your author details.

5. **Create** a folder called *icons* within the *with-phonegap* folder.
6. **Copy** the folder *worksheets/icons/android* to your *icons* folder. You can use copy and paste for this. This contains the icon files for varying resolution screens (in this case just for Android).
7. **Copy** the file *worksheets/icons/android/mdpi.png* to the *with-phonegap* folder and **rename** it *icon.png*. This is the default icon to use on a device if nothing more specific can be found.

8. No other changes in this simple version of the app are required. In more advanced apps we would really need JavaScript that detects the device the app is running on so that we can, for example, add a back button (e.g. for iOS devices where an app-defined back button is required).

   Also, if we decide to create an app that can run in raw HTML form as well as in a packaged form then we will need further JavaScript that determines whether the app is running as a PhoneGap app or a raw HTML app. Why? The device libraries provided by PhoneGap are different from those provided by browsers. Typically, your JavaScript will check to see if PhoneGap is enabled, and if so will use its library functions, else it will check to see if the browser supports the required device library function, and if so calls it, otherwise a user-friendly message is displayed to the user saying the function is not supported, or the user interface feature requiring that function is disabled.

9. Using your folder navigator tool, **right click** on the *with-phonegap* folder and **select compress** or similar to create a zip file.

Step Two: Uploading the app to PhoneGap Build

The setup instructions distributed prior to the workshop requested that you create an account on the PhoneGap Build website.

If you haven't done that, don't worry you can do it now (steps 10-12):

10. On your browser **navigate** to http://build.phonegap.com/
11. **Select** the *Register* link and then select the *Free plan*.
12. **Create** an Adobe account. Note that although the free plan only allows you to upload one private app, you can delete that app by clicking on the app's settings and the clicking the *Delete the app* button. That way you can upload something different later on.

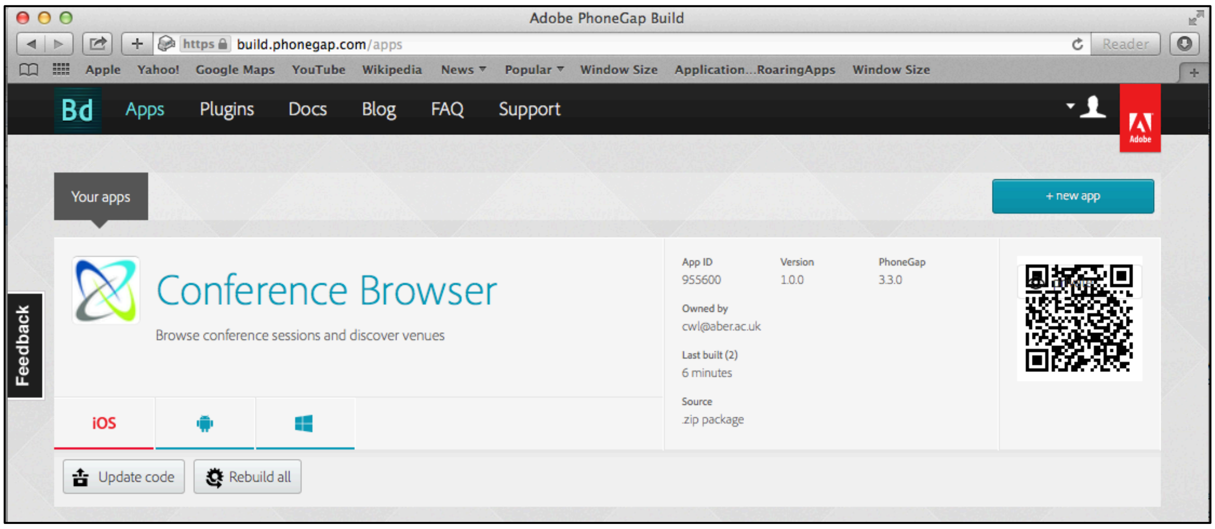
We now sign in and upload and package our code.

13. **Sign into** your PhoneGap Build account.
14. **Click** on the *+ new app* button.
15. Use the Upload a zip file to **upload** your zipped conference app. Click the *Ready to Build* button. PhoneGap Build will attempt to build native app wrappers for iOS, Android, and Windows Phone. This build may take a little time. Note that iOS requires a vendor provided signing key and so will fail. PhoneGap Build will add a default debug signing key for Android allowing us to run the app in debug mode on an emulator or Android device. If you want to upload your Android app to Google Play then you will need to sign it with a self-signed certificate (see the PhoneGap Build documentation).

There are three ways to install your app. Select the option most suitable for you:

1. Installing the app on your android device using a QR Code reader
2. Running the app in an Android emulator/device (if you don't have an Android device)
3. Installing onto an android device from the command line

Software Alliance Wales is a European funded project led by Swansea University in partnership with the University of Bangor, Glamorgan, Aberystwyth and the University of Wales Trinity Saint David.



Conference Browser

Browse conference sessions and discover venues

App ID 955600

Version 1.0.0

PhoneGap 3.3.0

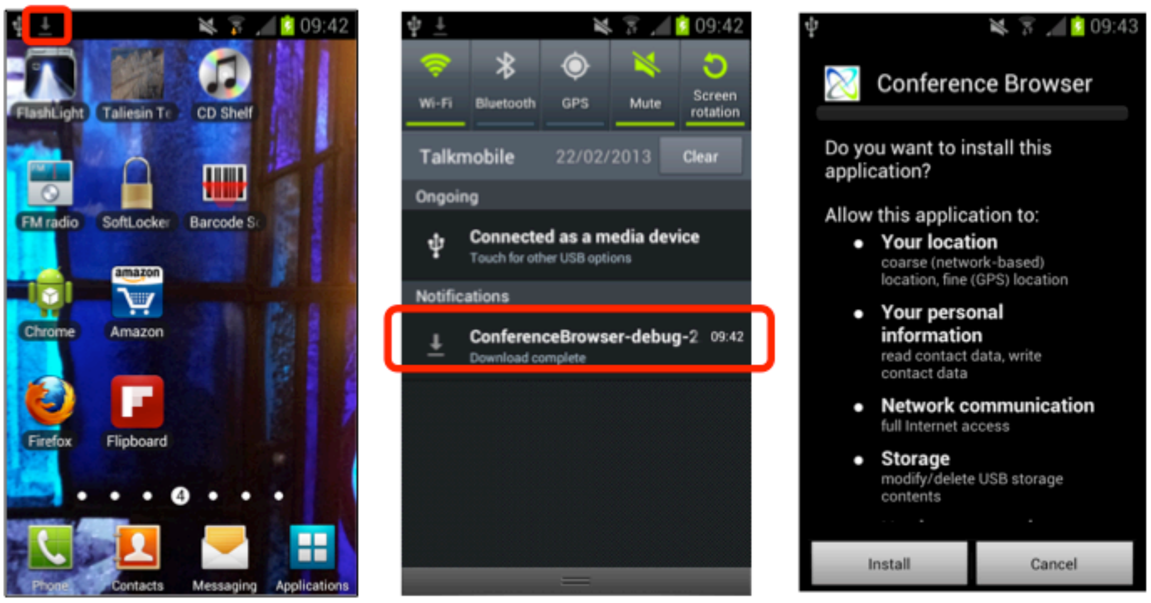
Owned by cwl@aber.ac.uk

Last built (2) 6 minutes

Source .zip package

Step Three (option 1): Installing the app on your android device using a QR Code reader:

This is the easiest approach. The PhoneGap Build apps page displays a QR Code next to each of your apps. If you have a Barcode reader app on your Android device you should be able to use it to download the ConferenceBrowser app. Make sure that your device settings are correct: **tap** *Settings*, then **tap** *Security* and then **select** *Unknown Sources*. Capture the QR Code, then open the notifications bar, tap on the downloaded app and install or reinstall the app:



Conference Browser

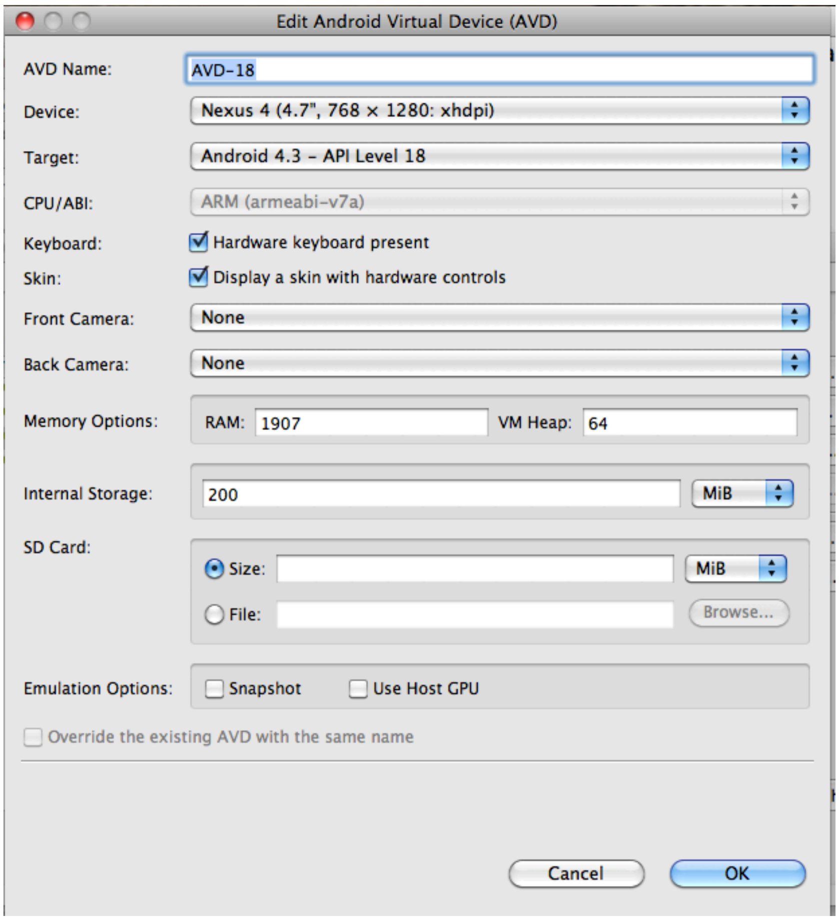
Do you want to install this application?

Allow this application to:

- Your location
  coarse (network-based) location, fine (GPS) location
- Your personal information
  read contact data, write contact data
- Network communication
  full Internet access
- Storage
  modify/delete USB storage contents

Step Three (option 2): Running the app in an Android emulator/device

16. Once built, **click on** the Android tab to **download** the APK (Android Package) file to a folder of your choice.
17. The first step is to create a virtual device for the emulator. **Navigate** to the Android installation and its *tools* subfolder. On our Macs this can be found in */Applications/android-sdk-macosx/tools*. Double click on the *android* file.
18. From the SDK Manager's Tools menu **select** *Manage AVDs…*
19. If there are no AVDs listed[1], then you will need to **create** one via the *New* button. **Fill in** the fields as shown below:



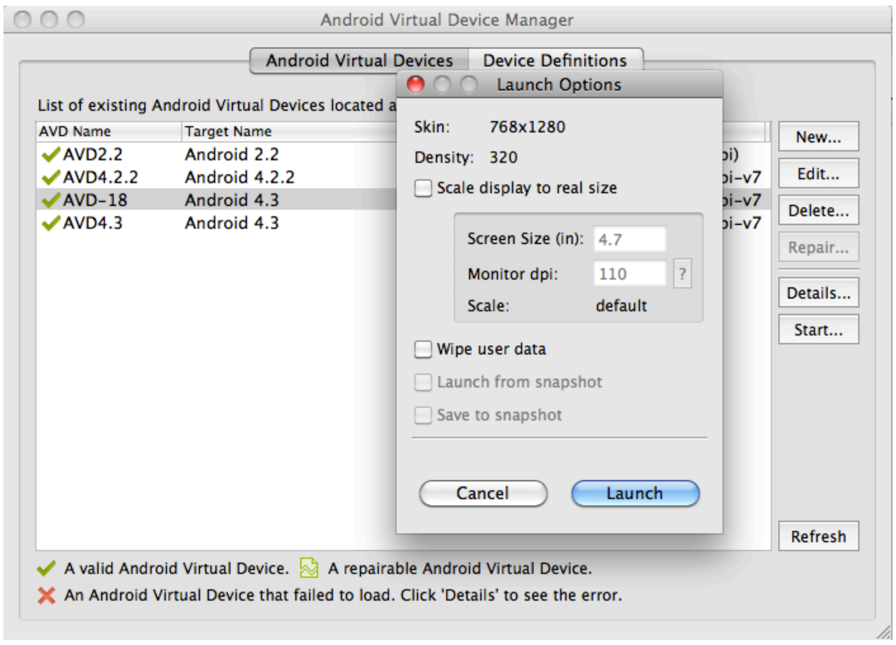
AVD Name:	AVD-18
Device:	Nexus 4 (4.7", 768 × 1280: xhdpi)
Target:	Android 4.3 – API Level 18
CPU/ABI:	ARM (armeabi-v7a)
Keyboard:	☑ Hardware keyboard present
Skin:	☑ Display a skin with hardware controls
Front Camera:	None
Back Camera:	None
Memory Options:	RAM: 1907 VM Heap: 64
Internal Storage:	200 MiB
SD Card:	● Size: MiB ○ File:
Emulation Options:	☐ Snapshot ☐ Use Host GPU

☐ Override the existing AVD with the same name

You will use the AVD name in a moment. Click OK.

20. You can start the emulator from the AVD Manager by **clicking** on Start… and then Launch:

[1] Some might be listed if you are using one of our computers. If some do exist then write down the name of an AVD. You will need to refer to it in a moment. If the AVD listed is for an old version of Android, created a new AVD anyway. That way, you can compare and contrast the application working on both an old version of Android and the latest version of Android.



Android Virtual Device Manager

Android Virtual Devices | Device Definitions

List of existing Android Virtual Devices located a

AVD Name	Target Name
✓ AVD2.2	Android 2.2
✓ AVD4.2.2	Android 4.2.2
✓ AVD-18	Android 4.3
✓ AVD4.3	Android 4.3

Launch Options

Skin: 768x1280

Density: 320

☐ Scale display to real size

Screen Size (in): 4.7

Monitor dpi: 110

Scale: default

☐ Wipe user data

☐ Launch from snapshot

☐ Save to snapshot

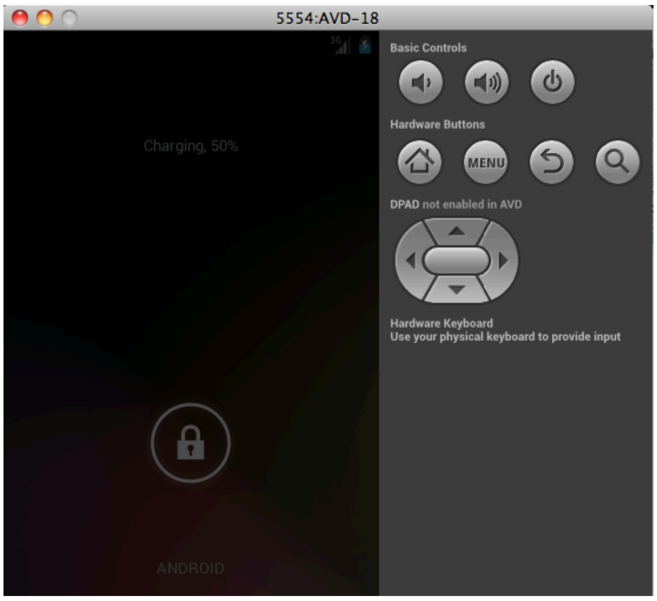
Cancel | Launch

New... | Edit... | Delete... | Repair... | Details... | Start... | Refresh

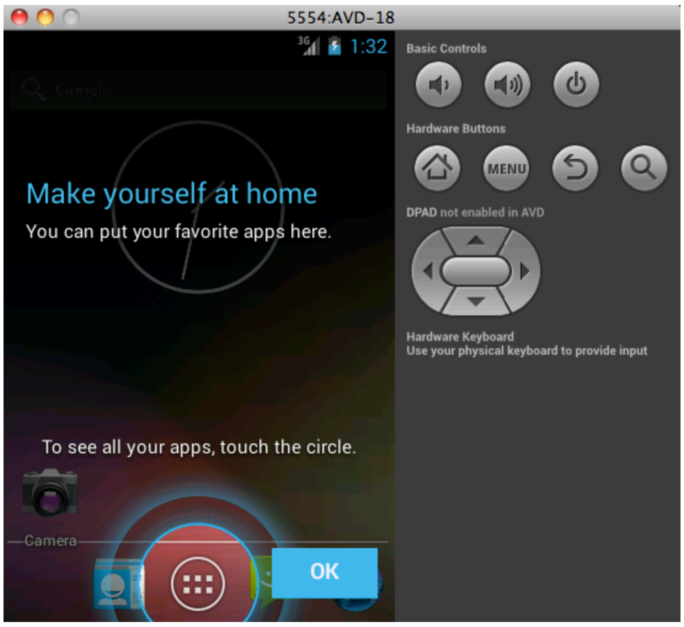
✓ A valid Android Virtual Device. A repairable Android Virtual Device.

✗ An Android Virtual Device that failed to load. Click 'Details' to see the error.

21. The emulator takes quite a long time to start, and especially on Windows machines. You will see the word *android* appear on the emulator's screen for some time (several minutes). Eventually, you may see the following (a lock screen appears for some emulators but not for others):



22. **Click** on the lock and **drag it** to the unlock icon (this will appear). The screen will now be unlocked (**click** the OK button):


5554:AVD-18
1:32
Make yourself at home
You can put your favorite apps here.
To see all your apps, touch the circle.
Camera
OK
Basic Controls
Hardware Buttons
MENU
DPAD not enabled in AVD
Hardware Keyboard
Use your physical keyboard to provide input

23. We now need to install the APK file you downloaded from Phonegap Build website. If you are using a computer where you have set the system PATH variable then you can simply open a terminal/command line window in the folder that contains the APK file.
    a. The first time you use the emulator you need to start the adb (Android debugger bridge) server:

       ```
       ./adb start-server
       ```

    b. ./adb devices will list the emulator and any other emulators or devices connected to the computer.
    c. Then deploy your apk file to the emulator with:

       ```
       ./adb install ConferenceBrowser-debug.apk
       ```

    d. To reinstall the app after making changes, you first need to uninstall. For that **type** the command:

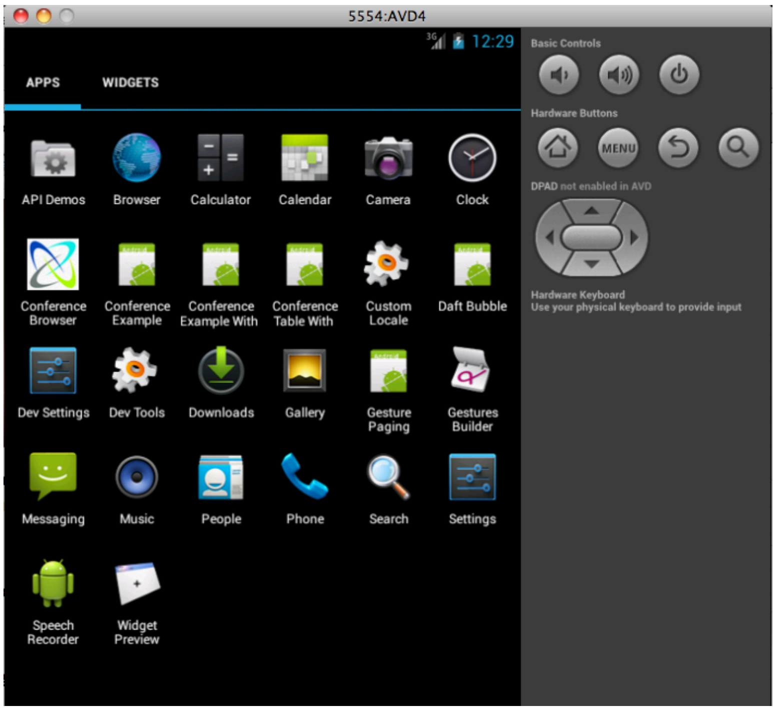
       ```
       ./adb uninstall uk.ac.aber.dcs.cwl.conference
       ```

       Use the package name you used for the widget id attribute in your config.xml file.

24. If you are using one of our Macs or have not set the system PATH you can open a command line window/terminal in the Android SDK installation's *platform-tools* folder (/Applications/*android-sdk-macosxplatform-tools)* and then type the above commands, but make sure you provide the full path to the apk file to the install command, or else copy the apk file to that folder. Ask us if you need help. On our Macs the path will probably be */Users/Guest/Downloads/ConferenceBrowser-debug.apk*.

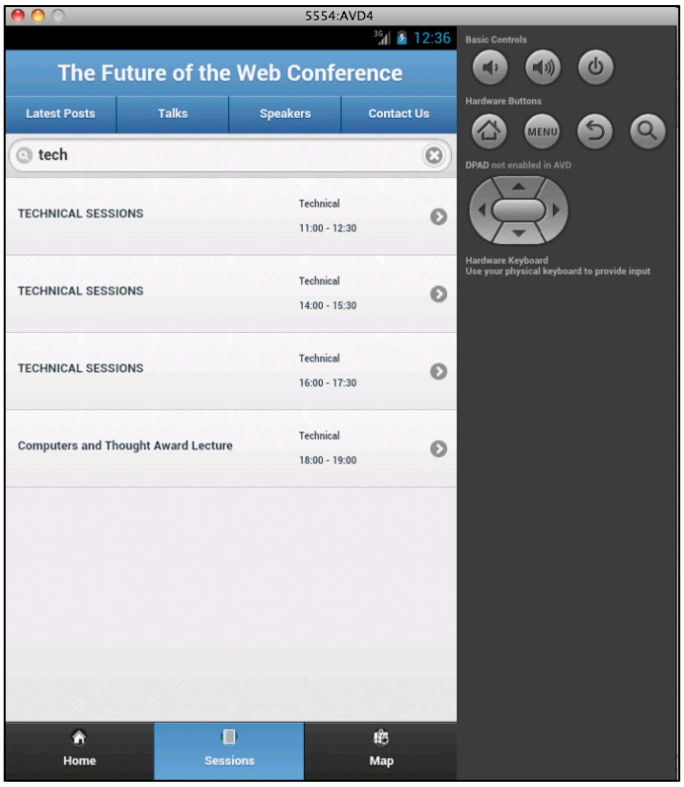
25. To see installed app **click** on the launcher screen button. The Conference Browser app icon should be listed:

Software Alliance Wales is a European funded project led by Swansea University in partnership with the University of Bangor, Glamorgan, Aberystwyth and the University of Wales Trinity Saint David.



Sometimes you will be prompted for other information when you click on the Launcher screen button. Normally, clicking the OK button is all you need to do, but occasionally the Launcher screen fails. If this happens then try running the emulator again.

26. **Click** on the Conference Browser app. It may take a little while to appear. Try clicking the tabs, scrolling etc. If you enter text into the filter field then you have to hit the enter key to get the filter to work:



The Future of the Web Conference

Latest Posts | Talks | Speakers | Contact Us

tech

TECHNICAL SESSIONS	Technical	11:00 - 12:30
TECHNICAL SESSIONS	Technical	14:00 - 15:30
TECHNICAL SESSIONS	Technical	16:00 - 17:30
Computers and Thought Award Lecture	Technical	18:00 - 19:00

Software Alliance Wales is a European funded project led by Swansea University in partnership with the University of Bangor, Glamorgan, Aberystwyth and the University of Wales Trinity Saint David.

Step Three (option 3): Installing onto an android device from the command line (this will not work from our Macs):

a. Installing to an android device from the command line tends to be easier for Macs than Windows; Windows may not have the required driver software for your device, although you may be able to obtain it online or install the Goggle USB Driver package via the SDK Manager tool.[2]

b. First enable your device to allow operation in debug mode when connected to the computer via a USB cable. To do this, **enter** the *Settings* app and then **select** *Developer options*. Then **select** *USB debugging*. On Android 4.2+ (Jelly Bean) things are more hidden:
    i. Go to the settings menu, and scroll down to "About phone." Tap it.
    ii. Scroll down to the bottom again, where you see "Build number." (for HTC One see iv below).
    iii. Tap it seven (7) times. After the third tap, you'll see a playful dialog that says you're four taps away from being a developer. (If only it were that simple, eh?) Keep on tapping, and *poof*, you've got the developer settings back.
    iv. On HTC One you can get to the Build Number from: About->Software information->More!
    v. You will now find Developer options directly under Settings.

c. After plugging your device into the computer via a data cable, list the devices and emulators running by **typing**:

```
./adb devices
```

   If Android recognizes you device it will list its identifier, otherwise there was a problem with the device driver.

d. You can now install to the device by **typing**:

```
./adb -s 00199dc367307e install ConferenceBrowser-debug.apk
```

   The –s option takes the device's identifier. This option can also be used with uninstall.

[2] http://www.howtogeek.com/125769/how-to-install-and-use-abd-the-android-debug-bridge-utility/ has some further help on how to obtain device drivers.

Software Alliance Wales is a European funded project led by Swansea University in partnership with the University of Bangor, Glamorgan, Aberystwyth and the University of Wales Trinity Saint David.