

Ionic

Prerequisites

nodejs (latest)

```
sudo apt-get install build-essential
# curl -sL https://deb.nodesource.com/setup_6.x | sudo -E bash -
# sudo apt install nodejs
```

nodeenv

```
sudo apt-get install python-pip
sudo pip install nodeenv
```

Create library

```
LIB=<some-path>
nodeenv --node=0.10.33 $LIB
. $LIB/bin/activate
```

```
npm install -g ionic
# npm install -g bower
```

update

```
npm update -g ionic cordova
```

Update libs

```
. $LIB/bin/activate
npm install -g ionic

cd <project>
ionic lib update
```

Android

It is important to install native gradle or use gradle shipped with android studio (see PATH below)

- ANDROID_HOME=/opt/android-sdk
- PATH=/opt/android-sdk/platform-tools:/opt/android-studio/gradle/gradle-3.2/bin/

example

```
nodeenv lib
. lib/bin/activate
npm install -g ionic cordova
ionic start test1 blank
cd test1
ionic cordova plugin

cd ..
git clone https://github.com/Cloudoki/ImageDetectionCordovaPlugin.git
# create package.json in ImageDetectionCordovaPlugin
cd test1
cordova plugin add ../ImageDetectionCordovaPlugin
ionic cordova platform add android
ionic cordova build android
ionic cordova run android --device
```

example2 with cordova-plugin-camera-preview.git → GianoDroidIonic

```
ionic start MyCameraApp blank --id org.csgalileo.mycameraapp
git clone --depth=1
https://github.com/cordova-plugin-camera-preview/cordova-plugin-camera-preview.git
cd MyCameraApp
ionic cordova platform add android

ionic cordova plugin add ../cordova-plugin-camera-preview
npm install @ionic-native/camera-preview --save

ionic cordova plugin add cordova.plugins.diagnostic
npm install @ionic-native/diagnostic --save

ionic cordova build android
ionic cordova run android --device
```

Add java sources and native C++ support

Add opencv support http://wiki.csgalileo.org/projects:giano:android#simpler_mode

In <project root> add: CMakeLists.txt

```
cmake_minimum_required(VERSION 3.4.1)
set(CMAKE_VERBOSE_MAKEFILE on)
find_library(log-lib log)

set(OpenCV_DIR "android-opencv/opencv/src/sdk/native/jni")
```

```
find_package(OpenCV REQUIRED)
message(STATUS "OpenCV libraries: ${OpenCV_LIBS}")
```

and add to *build-extras.gradle* (this file is copied with hook <hook src="scripts/gradle.sh" type="after_platform_add" />)

```
android {
    externalNativeBuild {
        cmake {
            path "../..CMakeLists.txt"
        }
    }
    packagingOptions {
        // edit also abiFilters and clean project before make apk
        //exclude 'lib/armv7a/libopencv_java3.so'
        exclude 'lib/mips/libopencv_java3.so'
        exclude 'lib/mips64/libopencv_java3.so'
        exclude 'lib/armv7/libopencv_java3.so'
        exclude 'lib/x86/libopencv_java3.so'
        exclude 'lib/x86_64/libopencv_java3.so'
    }
}

dependencies {
    compile project(':android-opencv:opencv')
}

ext.postBuildExtras = {
    logger.quiet('adding java sources')
    android {
        sourceSets {
            main.java.srcDirs += '../..src/android/java'
            main.res.srcDirs += '../..src/android/res'
        }
    }
}
```

from build.gradle of android-opencv comment classpath from dependancies

```
dependencies {
    //classpath 'com.android.tools.build:gradle:2.3.3'

    // NOTE: Do not place your application dependencies here; they
    belong
    // in the individual module build.gradle files
}
```

with <hook src="scripts/gradle.sh" type="after_platform_add" /> apply patch to change settings.gradle in

```
include ":"
include ":CordovaLib"
include ":android-opencv:opencv"
project(":android-opencv:opencv").projectDir = new File("../..../android-opencv/opencv")
```

with `<hook src="scripts/config.xml.py" type="before_compile" />` patch **res/xml/config.xml** to export plugin interface, for example

[res/xml/config.xml](#)

```
<feature name="TestPlugin">
  <param name="android-package"
value="org.csgalileo.giano.TestPlugin"></param>
  <param name="onload" value="true"></param>
</feature>
```

define typescript interface

[src/app/test.module.ts](#)

```
import { Injectable } from '@angular/core';

declare var cordova:any;

@Injectable()
export class TestPlugin {
  public testCall() {
    cordova.exec(
      function(data) { console.log("test-plugin
getDate="+data); },
      function(error) { console.log("test-plugin:"+error); },
      "TestPlugin",
      "getDate",
      []);
  }
}
```

Android Studio

After project creation via cli (see up) import project (Eclipse, ADT, gradle) from folder `<project-root>/platforms/android` (choose to update gradle).

Now it is possible to develop JAVA code in Android Studio but HTML/TS has to be changed in `<project-root>/src` and synced with 'ionic build'

From:

<https://wiki.csgalileo.org/> - **Galileo Labs**

Permanent link:

<https://wiki.csgalileo.org/tips/ionic>

Last update: **2017/08/07 13:54**

