**Calabash-Cucumber (Android) setup on Windows**

**Cucumber setup : -**

Refer cucumber setup document for installing cucumber on windows

**Calabash Android setup : -**

1. Install calabash-android gem (***gem install calabash-android***)
2. Check if all of its dependencies have been installed correctly

**Sample Calabash Android automation : -**

1. Create a directory *calabash\_cucumber*
2. Open command prompt , go to *calabash\_cucumber* directory and type

***calabash-android gen***

it will create a directory, with the following structure :

*features*

*|\_support*

*| |\_app\_installation\_hooks.rb*

*| |\_app\_life\_cycle\_hooks.rb*

*| |\_env.rb*

*| |\_hooks.rb*

*|\_step\_definitions*

*| |\_calabash\_steps.rb*

*|\_my\_first.feature*

1. Download an android app (.apk file) and save it under *calabash\_cucumber directory (* the apk file must be a signed apk file; I have used **Teddy Hyde** - *Teddy Hyde (Alpha).apk*)
2. Open an Android device or emulator (I have used AVD)
3. To debug (inspect) objects :
4. Type the command ***calabash-android console "Teddy Hyde (Alpha).apk"***
5. It should open the ruby console (irb), if all settings are correct
6. Type ***reinstall\_apps*** and the app should be installed in AVD
7. And then type ***start\_test\_server\_in\_background*** and the app should be launched
8. Now you can inspect objects in the screen

Like, type ***query("Button")*** and it will list its properties like this

*[*

*[0] {*

*"id" => "button",*

*"enabled" => true,*

*"contentDescription" => nil,*

*"text" => "Login",*

*"visible" => true,*

*"tag" => nil,*

*"description" => "android.widget.Button{b2d32a48 VFED..C. ......I. 108,196-211,254 #7f040014 app:id/button}",*

*"class" => "android.widget.Button",*

*"rect" => {*

*"center\_y" => 298,*

*"center\_x" => 159,*

*"height" => 58,*

*"y" => 269,*

*"width" => 103,*

*"x" => 108*

*}*

*}*

*]*

1. Refer to query document for native app debugging
2. Type the command

***calabash-android run "Teddy Hyde (Alpha).apk"***

and it will reinstall the app and run the cucumber test scripts against the app

1. If error occurs, then the apk may not be a signed one

Type ***calabash-android resign <filename.apk>*** and again run it (and still if error occurs, then add ***<uses-sdk android:targetSdkVersion="SDK\_VERSION" />*** and ***<uses-permission android:name="android.permission.INTERNET" />*** to the native app ***AndroidManifest.xml*** , where SDK\_VERSION is the version of the Android SDK you are using)