**SCREENSHOT PLUGIN**

**Objective:** The Screenshot plug-in allows your application to take screenshots of the current screen and save them into the phone.

**Getting Started :**

Before you start to look at the functionality of twitter plug-in make sure that you have following softwares installed:

* android-sdk
* phonegap-2.8.1
* sencha touch 2.0.1.1.

**Part 1: Creating plugin In Eclipse:**

**Steps for screenshot plugin functionality**

1. Create a phone gap project in the eclipse of the android sdk.For that we must have android sdk installed.

You can download the android sdk (ADT BUNDLE) form the following link:

<http://developer.android.com/sdk/index.html>

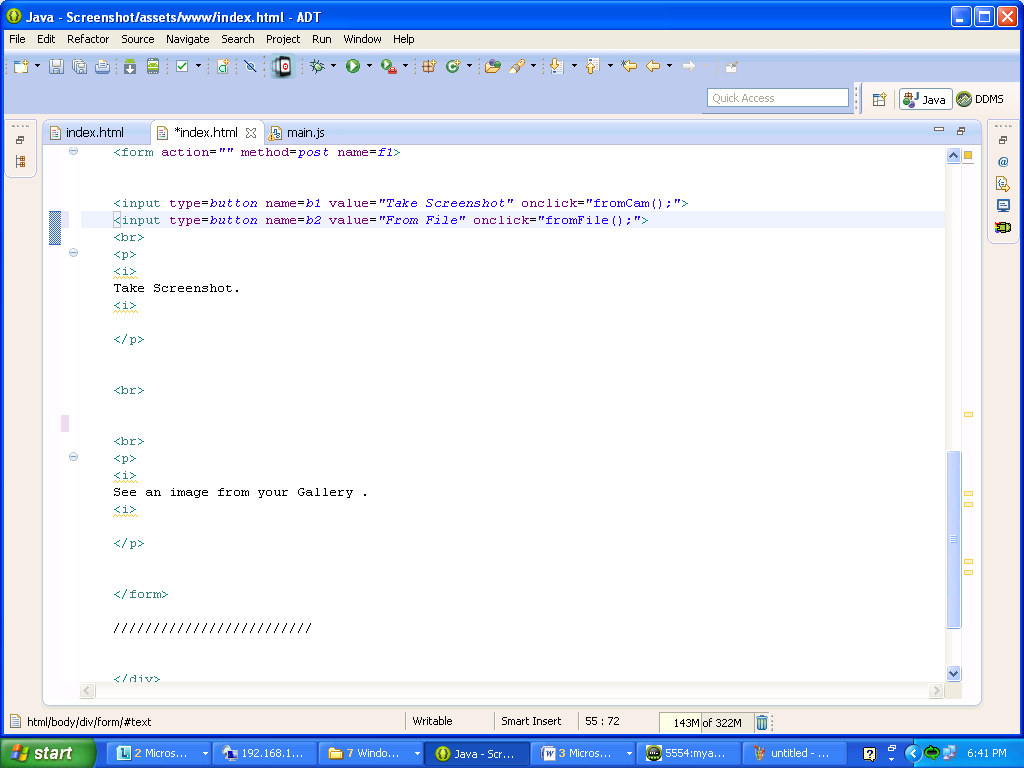
1. **To install the plugin, move www/screenshot.js to your project's www folder and include a reference to it in your html file after phonegap.js.**  
     
       <script type="text/javascript" charset="utf-8" src="cordova.js"></script>  
       <script type="text/javascript" charset="utf-8" src="screenshot.js"></script>
2. Create a package(directory) within your project called “com.phonegap.plugin.screenshot" and move Screenshot.java into it.
3. In the assets folder there exist a folder named as www, there is a js file there(screenshot.js) with the help of this file we will call the above two methods.
4. In your res/xml/plugins.xml file add the following line:

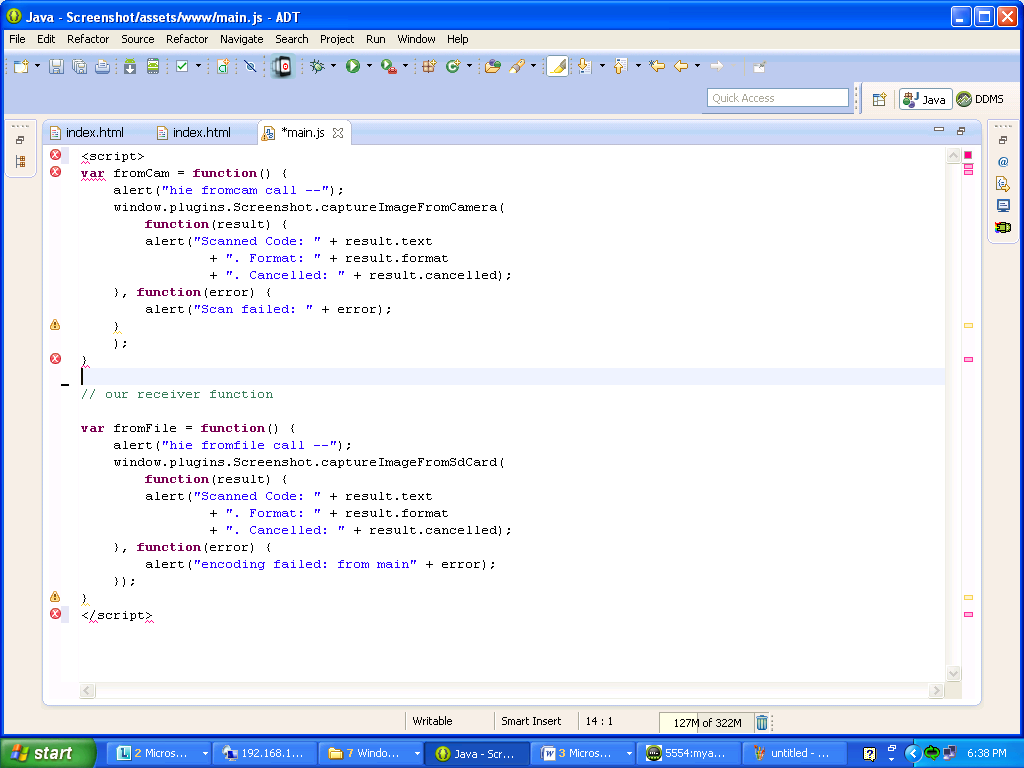
    <plugin name="Screenshot" value="com.phonegap.plugins.screenshot.ScreenShot"/>

6. **## Using the plugin ##**

You can call the methods in Screenshot.java using following available methods from script tag of your index.html  
  
The plugin creates the object `window.plugins.screenshot`. To use, call one of the following, available methods:

**Index.html:**



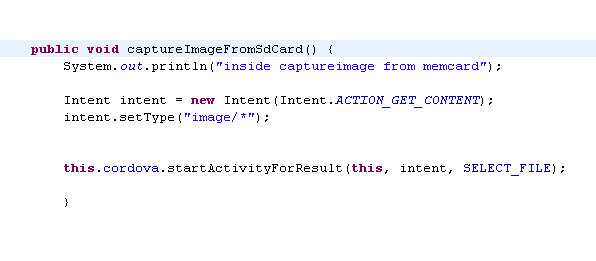
You can call the methods in Screenshot.java using following available methods from script tag of your index.html  
  
The plugin creates the object `window.plugins.screenshot`. To use, call one of the following, available methods:   


**Methods:**

1. caputreImageFromCamera() :



1. caputreImageFromSdCard()



Edit your AndroidManifest.xml and add the following permission:

<uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE" />

In addition you have to edit your res/xml/plugins.xml file to let Cordova know about the plugin:

<plugin name="Screenshot" value="org.apache.cordova.Screenshot"/>

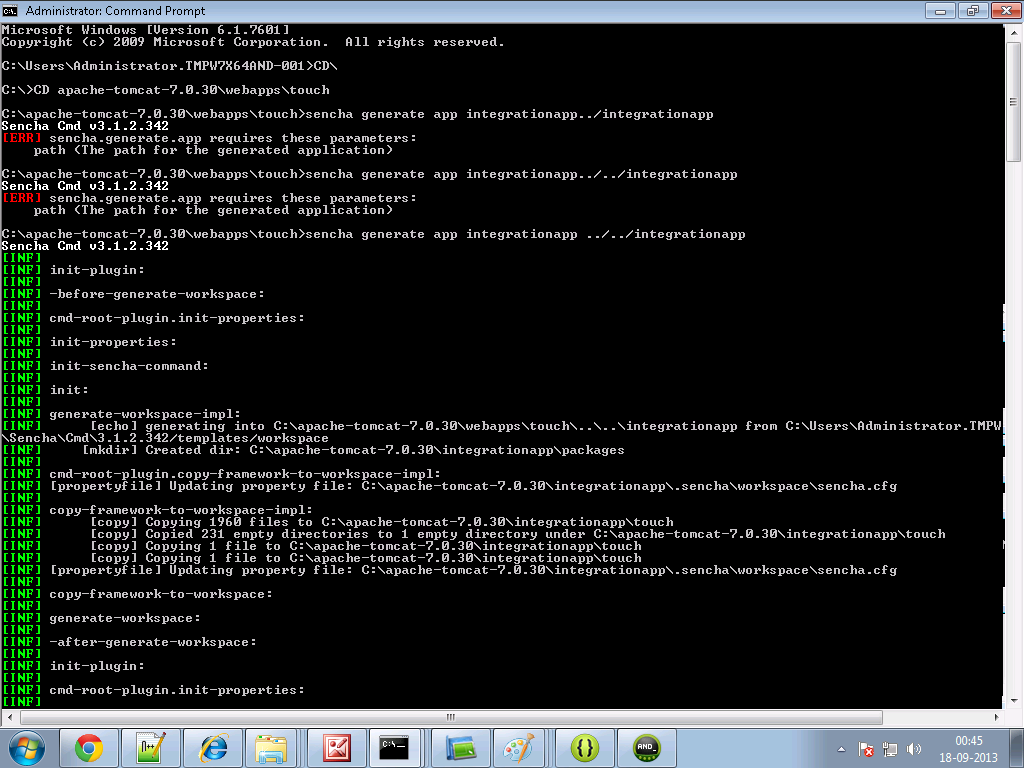
**Part2.**

**Integrating Screenshot Plugin with sencha**

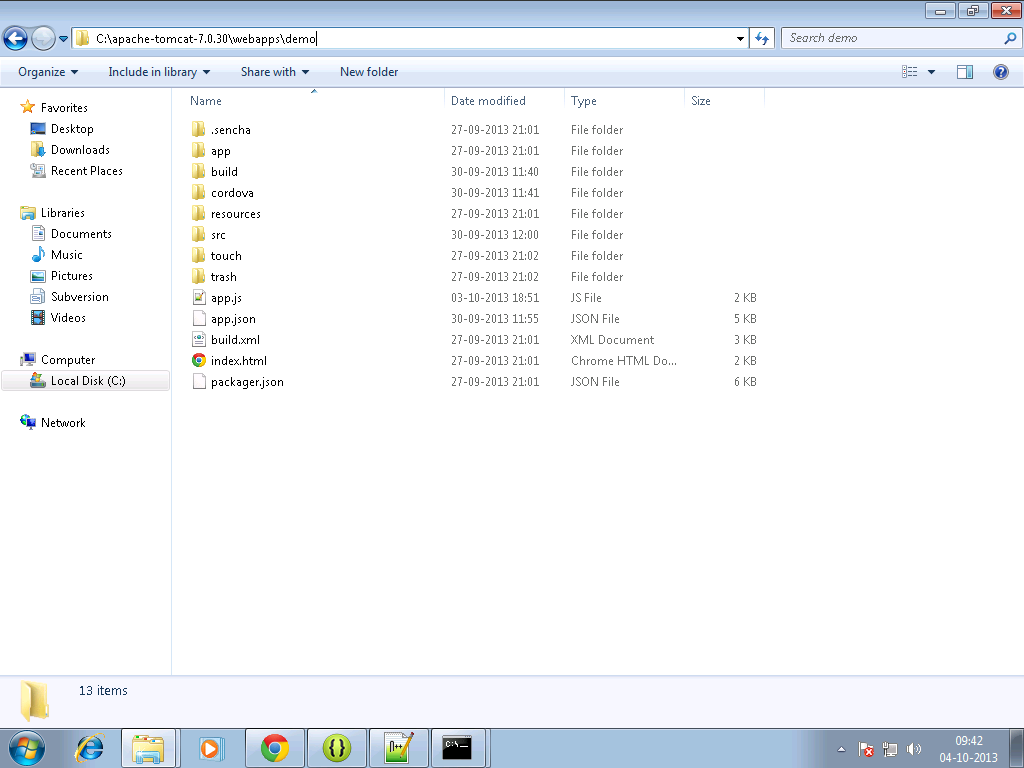
**Steps for screenshot plugin functionality**

1. **create a new application with the help of following command in cmd:**

**C:\apache-tomcat-7.0.30\webapps\touch>** **sencha generate app <appname> ../<appname>**



**2. Create a folder and name it as cordova in your sencha <appname> directory. For example**



**3. Copy the cordova.js ,cordova.js\_android, cordova.js\_ios and cordova.js\_plugins in them.**

**4. Make an entry in the js tag of app.json file residing in your<appname> directory as follows:**

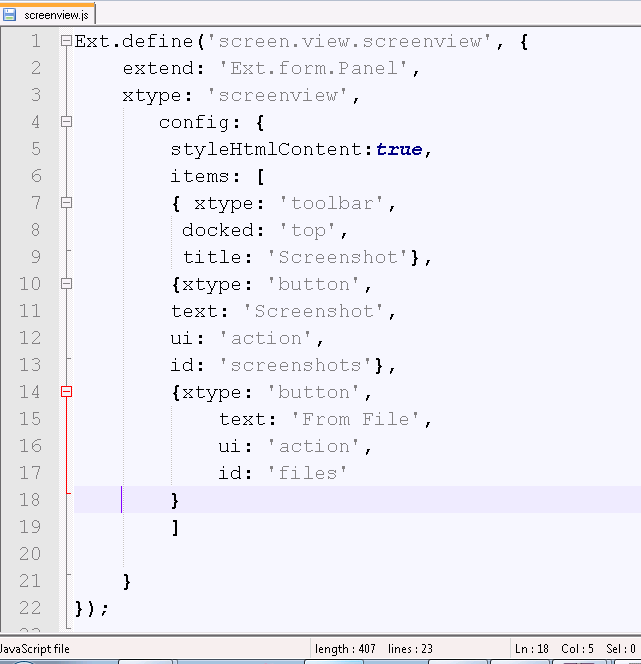
js": [  
  {  
            "path": "cordova/cordova.js",  
            "update": "delta"  
        },

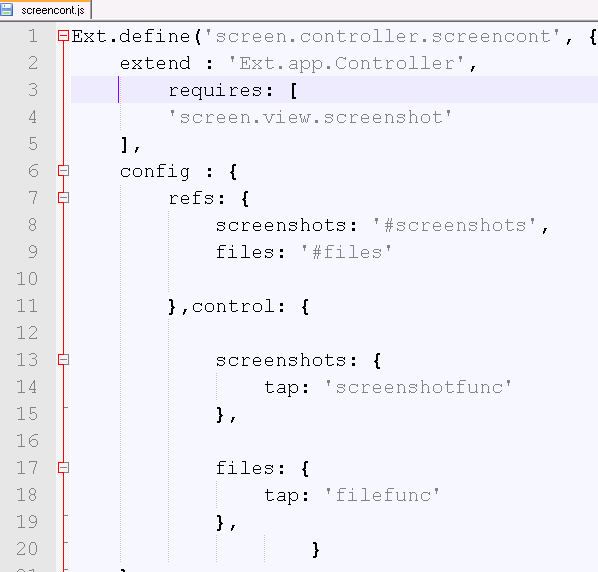
{  
            "path": "cordova/twitter.js",  
            "update": "delta"

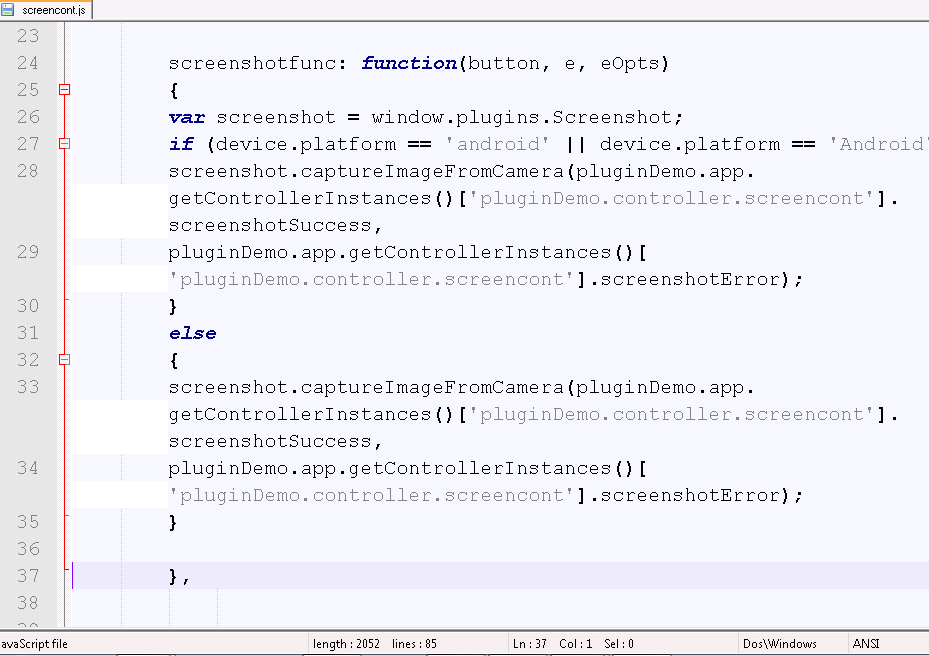
]

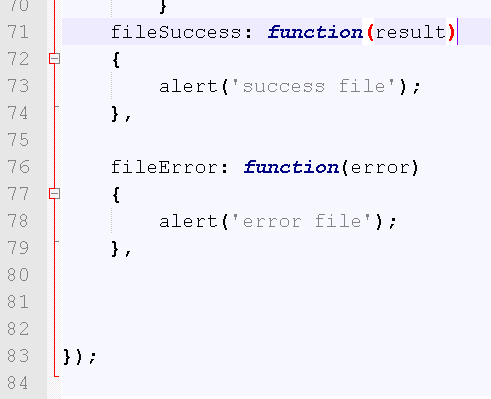
**5. Write the sencha code for view and controller in your sencha <appname>.**

**screenview.js**

****

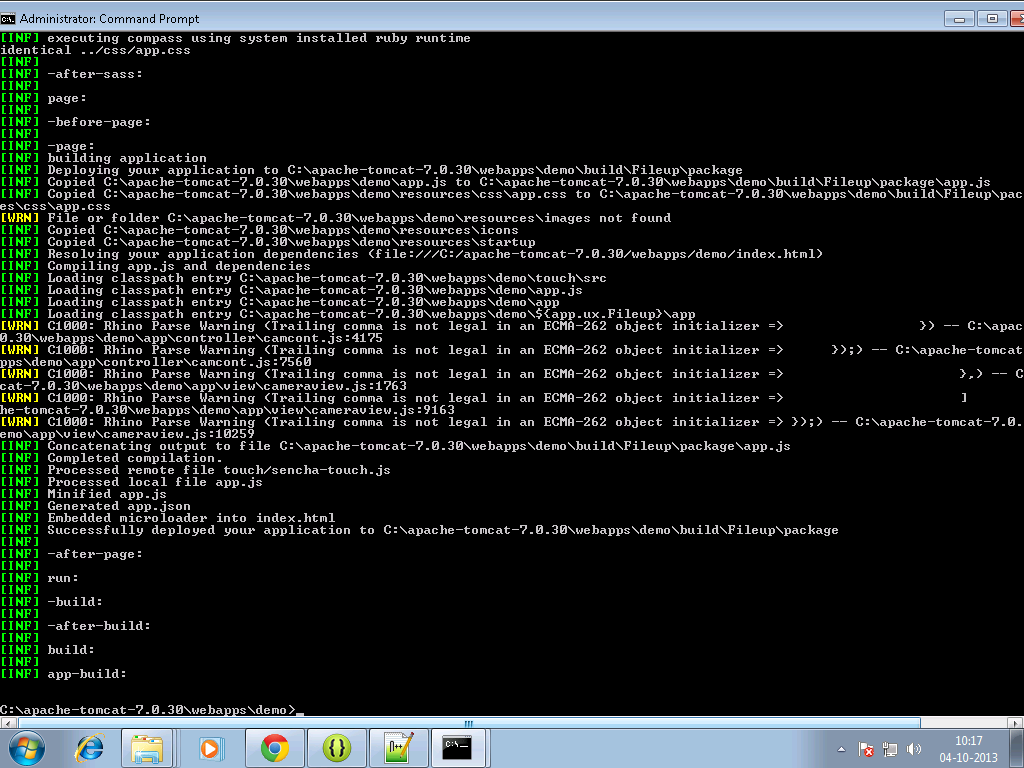






**6.** **In command prompt go to your sencha <appname> directory first, then Build the package of your sencha app by typing the following command :**

**Sencha app build package**

****

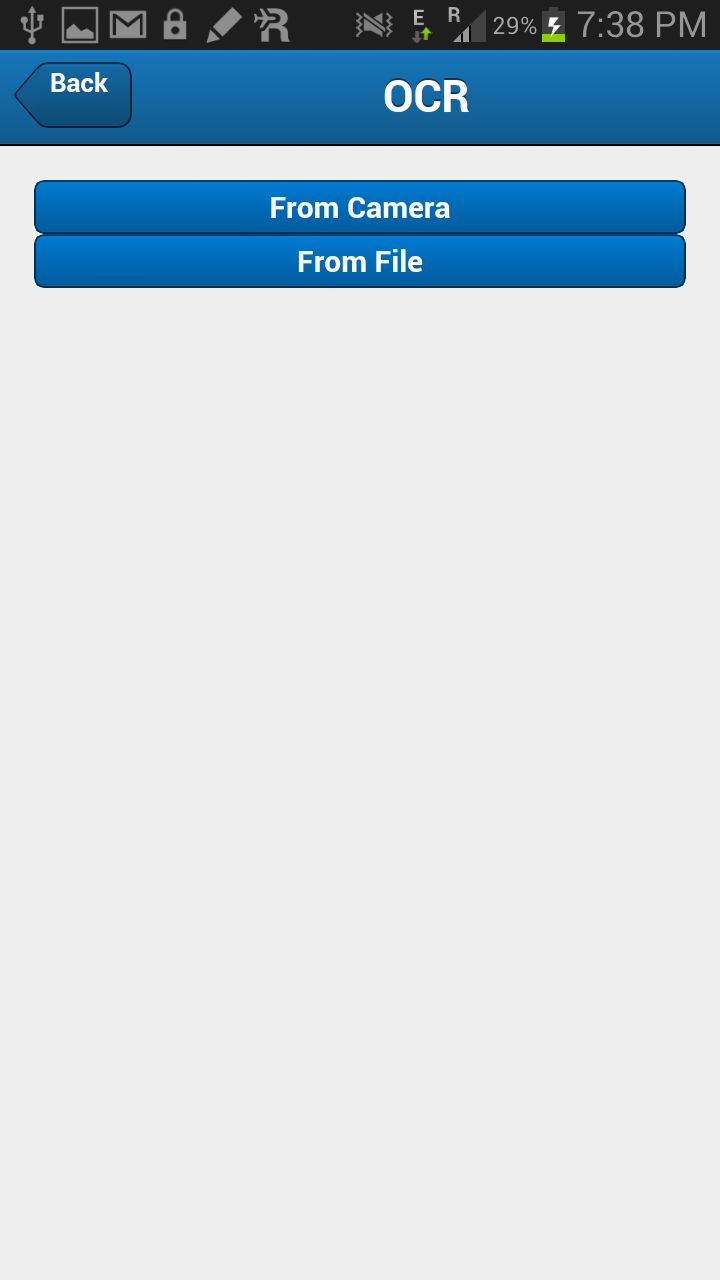
**7. Move to your sencha<app> directory, there a build folder will get created.**

**8. Copy files from package folder under <sencha appname>/build/ sencha < appname> to your www folder of android project for example:**

**C:\temp\TempAndroid\assets\www**

**9. build it - > apk will be in bin folder**

**Output on device:**

****