# Documentation - Bank & ATM Finder PhoneGap/Cordova Hybrid App

Bank & ATM Finder is a PhoneGap / Cordova Full Hybrid App with [AngularJS](https://angularjs.org/) **+** [Onsen UI](http://onsenui.io/)**.**

**You can use this App template to create full-fledge Bank & ATM Finder App for Android, iOS, Windows Phone, Firefox OS and other platforms that supported by PhoneGap/Cordova.**

# ****HTML Guide****

[Onsen UI](http://onsenui.io/) **is used for the HTML part that is powered by AngularJS. You can check the Elements from Onsen UI website -** <http://onsen.io/guide/components.html>

**Index.html contains the main markup that loads other part templates inside it. i.e all bank branches are individually separated to bank/ folder as well as ATM booths are separated to atm/ directory.**

**You have to add/change the bank branches and atm booths from these directories.**

**All sections are well commented for better understanding –** 

Change the markup or Edit as per your needs.

# CSS Guide

Several CSS Files are included on this template.



First two are for Onsen UI core CSS and the third one is for mapsed plugin that is used for a Map. You don’t have to change these files.

fonts.css is for offline Google Fonts (Oswald and Ubuntu). You can add your own fonts or refer directly the google fonts CDN links.

App.css contains the main styles for the App that are well commented with Table of Contents.

# JS Guide

Several scripts are included on this template.

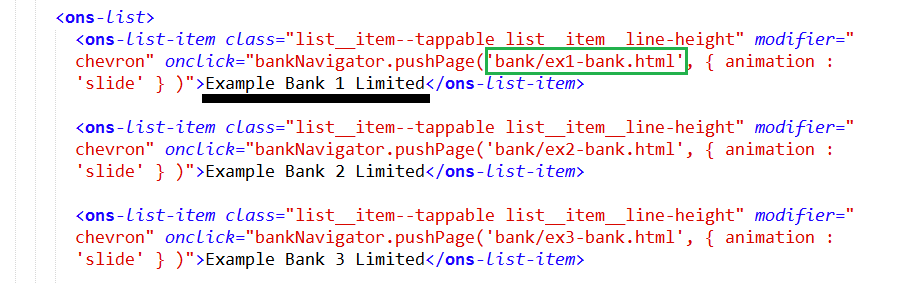


app.js contains your custom scripts if needed. bank-list.js contains all lsit of branches and their information and atm-list.js contains the list of ATM Booths and their information. Other files are Onsen core JS and Map plugin.

## Add Bank Branches

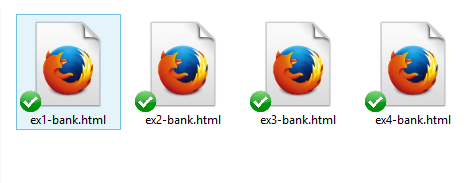
To add new branches on your application –

1. Go to find-bank.html and add your bank to the list -



Give the Bank Name and the URL of bank (part template located at bank/ directory)

1. Go to bank/ directory and create part template file for the bank bu duplicating one of the existing example part template file.



Rename it exactly like the link you placed earlier on find-bank.html (here ex1-bank.html).

Open the file and Rename the Bank name and Controller Name (marked on screenshot below)



1. Open js/bank-list.js and create a scope for the controller created above. Name the scope name same as the controller. That is the scope will only used within this controller.



Do same for other Bank branches. 10 examples are added on the template. You can use those as reference.

## Add Bank Branches

Adding ATM booths are pretty same as adding Bank branches that did above. Just remember to keep the controller name and scope name same. That’s it!

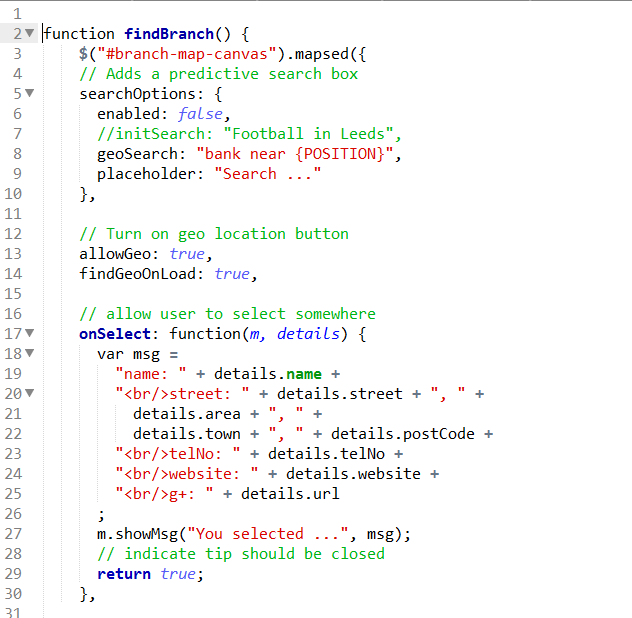
# Changing “Near You” Page

This page will work as it is without any modification

[Mapsed](http://mapsed.apphb.com/) plugin is used to build this page. Open the map.html to change any content.

Two canvas div named <div id="branch-map-canvas"></div> and <div id="atm-map-canvas"></div> showing the maps for branch and ATMs here.

If you want to change any map functionality open js/map.js file and change the options that need –



Use the plugin documentation for further change for this page - <http://mapsed.apphb.com/>

# Cordova App Guide for Android:

If you want to build your app for Android follow the guideline from [PhoneGap/Cordova documentation](http://docs.phonegap.com) for creating initial App for your Operating System. Use your own proper package name while creating the app.

After creating App, replace the www directory with the Template’s www directory you downloaded with this Template Pack.

Use the command-line interface to create app with the latest Cordova - <http://docs.phonegap.com/en/3.5.0/guide_cli_index.md.html#The%20Command-Line%20Interface>

Import project to Eclipse. Use this guide - <http://cordova.apache.org/docs/en/3.5.0/guide_platforms_android_index.md.html>

# Cordova Plugins:

### InAppBrowser

You can use InAppBrowser plugin to open external link with internal/externalbrowsers.

I have used the external links for this plugin. So you need to add this for better User Experience.

Documentation available here - <https://github.com/apache/cordova-plugin-inappbrowser/blob/master/doc/index.md>

### Splashscreen

You can use splashscreen plugin if you want to add slashscreen image to your app.

Documentation available here - <https://github.com/apache/cordova-plugin-splashscreen/blob/master/doc/index.md>

You can use more plugins from repository if needed - <https://github.com/apache/cordova-plugin-inappbrowser/blob/master/doc/index.md>

# Add Google Admob with Google Play Service to your Android App:

You can use Google Admob for your Android version to monetize.

* including google-play-services\_lib in your project. Use this if you never did it earlier - <http://code.tutsplus.com/tutorials/integrating-google-play-services-on-android--cms-19828>
* Then in your main class (will be something like honeGapAppName/platforms/android/src/com/ myname/myApp/myApp.java)
* Under the cordova import (import org.apache.cordova.\*;) add:

import android.view.WindowManager;

import android.widget.LinearLayout;

import com.google.android.gms.ads.AdRequest;

import com.google.android.gms.ads.AdSize;

import com.google.android.gms.ads.AdView;

* Then in your main publc click (something like public class myApp extends CordovaActivity) replace the default code with something like:

private AdView mAdView;

@Override

public void onCreate(Bundle savedInstanceState)

{

super.onCreate(savedInstanceState);

super.init();

super.loadUrl(Config.getStartUrl());

//super.loadUrl("file:///android\_asset/www/index.html")

mAdView = new AdView(this);

mAdView.setAdUnitId("YOUR\_ADD\_UNIT\_CODE");

mAdView.setAdSize(AdSize.BANNER);

LinearLayout layout = super.root;

layout.addView(mAdView);

layout.setHorizontalGravity(android.view.Gravity.CENTER\_HORIZONTAL);

mAdView.loadAd(new AdRequest.Builder().build());

}

@Override

protected void onPause() {

mAdView.pause();

super.onPause();

}

@Override

protected void onResume() {

super.onResume();

mAdView.resume();

}

* You may also have to add something like this to your AndroidManifest.xml:

<activity

android:name="com.google.android.gms.ads.AdActivity"

android:configChanges="keyboard|keyboardHidden|orientation|screenLayout|uiMode|screenSize|smallestScreenSize"/>

<meta-data android:name="com.google.android.gms.version"

android:value="@integer/google\_play\_services\_version"/>

Now run your App on device, you will see the admob ad.

Export your App and publish to Google Play.

If you face any problem or find anything not working after following the documentation, let me know from my profile page. I will try to help you to figure out.

Thanks for purchasing the App Template.