**Easy embedding Google Map**

ပထမဦးဆံုး ေျပာခ်င္တာက ကၽြန္ေတာ္႔ရဲ႕ ဒီေဆာင္းပါးေလးဟာ ကၽြန္ေတာ္႔လို တကယ္႔ကို အေျခခံသမားေတြ အတြက္ပဲ ရည္ရြယ္ပါတယ္။ နားလည္ တတ္ကၽြမ္းၿပီးသား ဆရာမ်ားအေနနဲ႔ မျပည္႔စံု မတိက်မႈေတြ၊ အားနည္းခ်က္ေတြကို သီးခံေပးၿပီး ဒီထက္ ပိုေကာင္းတဲ႔ နည္းလမ္းမ်ားကို ညႊန္ျပေပးၾကပါလို႔ ေမတၱာရပ္ခံပါတယ္။

လိုအပ္ခ်က္ တစ္ခုေၾကာင္႔ ေျမပံုကို Software ေတြထဲမွာ embedded လုပ္ၿပီး သံုးဖို႔ အလြယ္ဆံုး နည္းကို ကၽြန္ေတာ္ လိုက္ရွာေနခဲ႔ပါတယ္။ Google နဲ႔ အျခား Map API ေတြ အမ်ားႀကီး ရွိမယ္ ထင္ပါတယ္။ ဒါေပမယ္႔ ကၽြန္ေတာ္႔ဖုန္းမွာက Google playstore service အဆင္မေျပတာနဲ႔ ဖုန္းအားလံုးအတြက္ လိုအပ္ခ်က္ အနည္းဆံုး နည္းလမ္းကို ရွာခဲ႔ရတာပါ။

ကၽြန္ေတာ္ သံုးခဲ႔တဲ႔ နည္းလမ္းကေတာ႔ Android နဲ႔ Windows app ေတြမွာ WebView ကိုသံုးၿပီး အဲဒီေနရာမွာ ကိုယ္လိုခ်င္တဲ႔ ေျမပံုေပၚေစတဲ႔ URL ကို load လုပ္ၿပီး ေဖာ္ျပေစတာပဲ ျဖစ္ပါတယ္။

**Simple Map without Marker**

ပထမဆံုး သံုးခဲ႔တဲ႔ နည္းကေတာ႔ ေအာက္ပါ အတိုင္းပဲ ျဖစ္ပါတယ္ - <https://www.google.com.mm/maps/@22.0311769,96.4686856,21z?hl=en%22>

ဆိုတဲ႔ URL ကို latitude, longitude နဲ႔ zoom ေတြ ကိုယ္လိုခ်င္တဲ႔ တန္ဖိုးေတြနဲ႔ အစားထိုးၿပီး သံုးတဲ႔ နည္းပါပဲ။

အဲဒီမွာ 22.0311769 ဆိုတာက latitude တန္ဖိုး၊ 96.4686856 ကေတာ႔ longitude နဲ႔ 21 ကေတာ႔ Zoom တန္ဖိုးေတြ ျဖစ္ပါတယ္။ အဲဒီတန္ဖိုးေတြကို သက္ဆိုင္ရာ programming ေတြထဲမွာ String variable ေတြနဲ႔ ကုဒ္မွာ အစားထိုးၿပီး URL ကို စိတ္ႀကိဳက္ ျပင္ဆင္ႏိုင္ပါတယ္။ နမူနာကုဒ္ေတြကိုေတာ႔ Android နဲ႔ ေရးျပထားပါတယ္။

**ဥပမာ-**

String lat=22.0311769;

String lng=96.4686856;

String zoom=25;

String URL = “<https://www.google.com.mm/maps/@”+lat+”,”+lng+”,”+zoom+”z?hl=en%22>” ;

Latitute နဲ႔ Longitude တန္ဖိုးေတြကို Android မွာ GPS ကေနတဆင္႔ ရယူႏိုင္ပါေသးတယ္။ အဲဒီ URL ကို Webview မွာ load လုပ္ျခင္းျဖင္႔ ေျမပံုေပၚလာမွာ ျဖစ္ပါတယ္။

wv1=(WebView)findViewById(R.id.*webView2*);  
wv1.getSettings().setLoadsImagesAutomatically(**true**);

wv1.getSettings().setJavaScriptEnabled(**true**);

wv1.setScrollBarStyle(View.*SCROLLBARS\_INSIDE\_OVERLAY*);

wv1.loadUrl(url);

UI မွာ ကိုယ္ဖန္တီးထားတဲ႔ WebView ေပၚမွာ ေျမပံုကို ျပေစခ်င္ရင္ ေနာက္ထပ္ ကုဒ္ေတြ ထပ္ထည္႔ေပးရပါေသးတယ္။

wv1.setWebViewClient(**new** MyBrowser()); ဆိုတာကို ထည္႔ေပးရၿပီး MyBrowser() class တစ္ခုလည္း ထည္႔ေပးရပါဦးမယ္။

**private** **class** MyBrowser **extends** WebViewClient {

@Override

**public** **boolean** shouldOverrideUrlLoading(WebView view, String url) {

view.loadUrl(url);

**return** **true**;

}

}

ဒါေပမယ္႔ ဒီနည္းမွာ အားနည္းခ်က္က Markers ေတြ တင္ၿပီး ေနရာကို ညႊန္ျပျခင္း မျပဳႏိုင္တဲ႔အတြက္ ေနာက္တစ္နည္းကို စဥ္းစားခဲ႔ရပါတယ္။

**Simple Map with Markers**

ေနာက္ဆံုး ေအာက္ပါ လိပ္စာက Google Developers ကနည္းကို အသံုးျပဳဖို႔ ဆံုးျဖတ္လိုက္ပါတယ္။

<https://developers.google.com/maps/documentation/javascript/examples/marker-simple>

ေအာက္က ကုဒ္ေတြကေတာ႔ အထက္ပါလင္႔က နမူနာကို အေျခခံၿပီး မိတ္ေဆြ တစ္ေယာက္ရဲ႕ အကူအညီနဲ႔ ေရးခဲ႔ပါတယ္။ map2.php လို႔ အမည္ေပးလိုက္ပါတယ္။

<!DOCTYPE html>

<html>

<head>

<style>

#map {

width: 500px;

height: 400px;

}

</style>

</head>

<body>

<div id="map"></div>

<?php

$name=addslashes($\_GET['name']);

$address=$\_GET['address'];

$lat=$\_GET['lat'];

$lng=$\_GET['lng'];

?>

<script>

function initMap() {

var myLatLng = {lat: <?php echo $lat; ?> , lng: <?php echo $lng; ?>};

var mapDiv = document.getElementById('map');

var map = new google.maps.Map(mapDiv, {

center: myLatLng,

zoom: 15

});

var marker = new google.maps.Marker({

position: myLatLng,

map: map,

title: '<?php echo $name; ?>'

});

}

</script>

<script src="https://maps.googleapis.com/maps/api/js?callback=initMap"

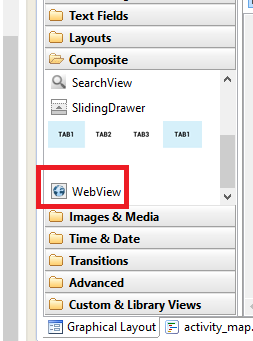
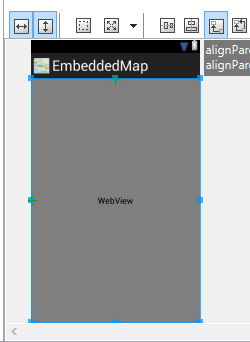
async defer></script>

</body>

</html>

အဲဒီ PHP ဖိုင္ကို အင္တာနက္မွ web location တစ္ေနရာမွာ သြားသိမ္းထားလိုက္ပါတယ္။ <http://mmgreenhackers.com/esdb/map2.php> ဆိုတဲ႔ နမူနာေနရာေလးမွာပါ။

အဲဒီေျမပံုကို android က webview တစ္ခုေပၚမွာ ျပသေပးဖို႔ UI မွာ webview တစ္ခုကို ထည္႔ပါတယ္။ webView1 ဆိုၾကပါစို႔။ UI ကို ေအာက္ပါအတိုင္း ရိုးရိုးရွင္းရွင္းပဲ တည္ေဆာက္ပါမယ္။

**activity\_map.xml**

<RelativeLayout xmlns:android=*"http://schemas.android.com/apk/res/android"*

xmlns:tools=*"http://schemas.android.com/tools"*

android:layout\_width=*"match\_parent"*

android:layout\_height=*"match\_parent"* >

<WebView

android:id=*"@+id/webView1"*

android:layout\_width=*"match\_parent"*

android:layout\_height=*"match\_parent"*

android:layout\_alignParentLeft=*"true"*

android:layout\_alignParentTop=*"true"* />

</RelativeLayout>

အင္တာနက္သံုးဖို႔ permission ထည္႔ပါမယ္။ ေအာက္က ႏွစ္ခုကေတာ႔ GPS သံုးဖို႔ လိုရမယ္ရ ထည္႔ထားတာပါ။

<uses-permission android:name=*"android.permission.INTERNET"* />

<uses-permission android:name=*"android.permission.ACCESS\_FINE\_LOCATION"* />

<uses-permission android:name=*"android.permission.ACCESS\_COARSE\_LOCATION"* />

ေအာက္ပါကုဒ္ေတြကို သံုးပါမယ္။ အေသးစိတ္ကို project တစ္ခုလံုး download လုပ္ၿပီး ေလ႔လာၾကည္႔ေစလိုပါတယ္။

**public** **void** showMap2(String name, String address, String lat, String lng)

{

String url =<http://mmgreenhackers.com/esdb/map2.php?name=>

+name+"&address="+address+"&lat="+lat+"&lng="+lng;

wv1=(WebView)findViewById(R.id.*webView1*);

wv1.setWebViewClient(**new** MyBrowser());

wv1.getSettings().setLoadsImagesAutomatically(**true**);

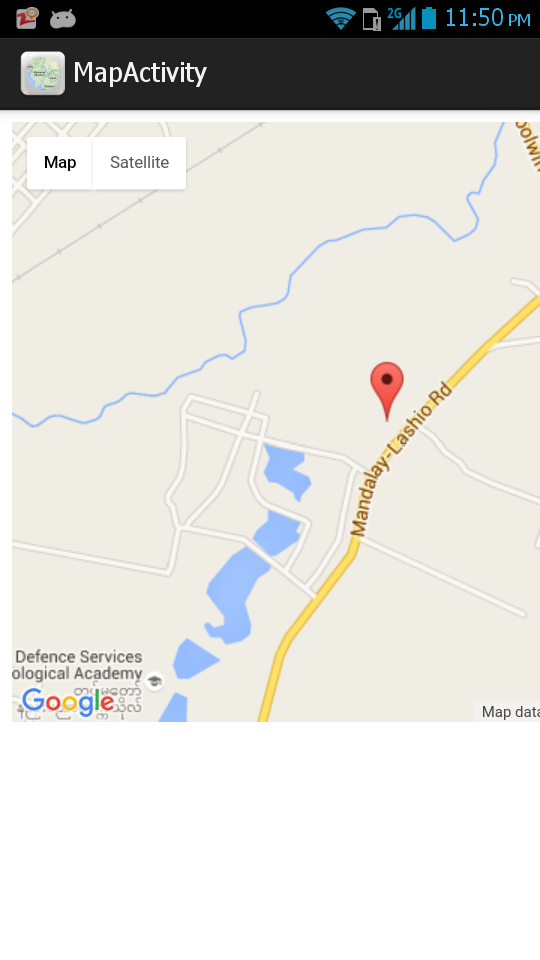
wv1.getSettings().setJavaScriptEnabled(**true**);

wv1.setScrollBarStyle(View.*SCROLLBARS\_INSIDE\_OVERLAY*);

wv1.loadUrl(url);

}

ေအာက္ပါအတိုင္း ေျမပံုကို Marker နဲ႔ တကြ ျမင္ရမွာ ျဖစ္ပါတယ္ ခင္ဗ်ာ..



ကုဒ္ေတြကို ေအာက္ပါ လိပ္စာမွာ MIT opensource license နဲ႔ ရယူအသံုးျပဳႏိုင္ပါတယ္ ခင္ဗ်ာ။

<https://github.com/AungWinnHtut/EmbeddedMap>

Dr. ေအာင္ဝင္းထြဋ္

http://www.engineer4myanmar.com