Archived:Using Google Maps for JavaScript with Symbian Location API

Archived: This article is **archived** because it is not considered relevant for third-party developers creating commercial solutions today. If you think this article is still relevant, let us know by adding the template {{ReviewForRemovalFromArchive|user=~~~~|*write your reason here*}}.

Here is a complete application with source code on how to display Google Maps for JavaScript with Location API on Symbian. The application was tested on N95 and S60 3rd Edition FP1 Emulator: Media:LBSSample.zip

This article needs to be updated: If you found this article useful, please fix the problems below then delete the {{ArticleNeedsUpdate}} template from the article to remove this warning.

Reasons: hamishwillee (26 Jan 2012)
The article is a bit of a mess, having been updated a number of times to make statements about google licencing of their key without the old status being removed. There is also an updated version for download, but its not clear at what point it was updated and whether the google APIs will now work. This should be cleaned up, and the comments in the talk page that have been addressed removed

The details and on how to make this application is as detailed below

Pre-requisites

Before using this example, compile the HttpClientEngine] example.

To compile the application you will need to first replace:

```
#include <LBSSample_0xE1038E0E.rsg>
```

with

```
#include <LBSSample.rsg>
```

Signup for Google Maps key

Note: Usage of this code with the free Google Maps API Key breaks Google's Terms and Conditions (section 10.8). You should purchase an Enterprise License if you wish to use the Google Maps API as shown in this example.

This is quite simple, just go to the following site and register providing a URL but only after login with a gmail account: http://code.google.com/apis/maps/signup.html

The sample I have used uses a key generated by myself so if any error occurs it might be due to that key. So go to the above site and register your own key.

Complete help documentation is provided in the above site.

Also documentation regarding usage of Google maps in mobile is provided in How to use Google Static Maps data in mobile applications

TAMHANs notes on legality

After having briefly looked atr the contract, I see the following passage in the FAQ:

```
>Can I use Google Maps in my non-Web application?
```


(C) Copyright Nokia 2014. All rights reserved.

```
Yes, the Google Maps APIs can now be used in Desktop applications, provided that they
adhere to the other restrictions of the Terms of Service. Note that in order for a
desktop application to be deemed "publicly accessible", there must be a publicly
accessible webpage from which it can be downloaded. See Section 7.1c of the Terms of
Service for more information.
```

In the end, I think that the legal situation is open ended. Passage 7.8 explicitly contains a dereference clause, which is why it IMHO is not applicable to this! Sending an email to google definitely puts you on the safe side!

PLEASE NOTE, Google Static Maps API no longer requires a Maps API key

For updated information on Google Static Maps API, please visit http://code.google.com/apis/maps/documentation/staticmaps/

Now coding

For making the application and testing it I made use of Carbide.C++ and S60 3rd Edition FP1 emulator Started of using the 'hello world ' template provided by Carbide.C++

Following are the beakdown of different components with main code. The full code can be downloaded from the source zip attached to this page.

Retrieve Current latitude and Longitude

The current latitude and longitude is retrieved using the following code

```
void CLBSSampleAppView::GetGPSInfoL()
     {
     // create CGpsPositionRequest object and put it into cleanup stack;
     // pass application name as argument
     CGpsPositionRequest* request = CGpsPositionRequest::NewL(
         _L("LBS_0xEA962394"));

     CleanupStack::PushL( request );
     // get current location (this operation can be long up to 30 seconds);
     // progress dialog is shown to user during this time
     TBool result = request->FetchCurrentPostionL(latitude, longitude);

     // delete request object
     CleanupStack::PopAndDestroy(request);

         // process result here
         if (result)
             {
             // success, use latitude and longitude coordinates
             }
         else
             {
             // failed getting current position, show error message to user
         }

     }
```

For more details on how to retrieve GPS latitude and longitude can be found from Archived:GPS API in Symbian 3rd Edition

Setup folder to download Map

 (C) Copyright Nokia 2014. All rights reserved.

Setup a file in the filesystem of the phone where the google map image is to be saved.The google map to be downloaded is of jpeg format. You may download other format such as PNG and Gif. If so change the code accordingly

```
void CLBSSampleAppView::SetupFileDownload()
 {
 _LIT(KXMLFilePath, "c:\\Data\\Images\\");
  TFileName iCurrentFileName;
  iCurrentFileName.Append(KXMLFilePath);
 iCurrentFileName.Append(_L("LbsSample.jpg"));
 TInt err=iRLbsImage.Open(CCoeEnv::Static()->FsSession(),iCurrentFileName,EFileWrite);

 if (err==KErrNotFound) // file does not exist - create it
 {

 err=iRLbsImage.Create(CCoeEnv::Static()->FsSession(),iCurrentFileName,EFileWrite);
 }

 }
```

Download Map of the corresponding location

To download the corresponding MAP of the location (Latitude and longitude) obtained from GetGPSInfoL, you have to send an HTTP request to google maps with the latitude & longitude values.

Inorder to send the HTTP request and also to parse the data obtained from the server, I made use of S60 Platform: HTTP Client API Example v2.1 provided in the document section of the Nokia Developer website. This is a quite helpful document on how to use http Get & Post requests. For the LBSSample application only Get method is used.

```
//Prefix of google URL to retrieve Map
// %S, %S will be replaced by latitude and longitude values respectively by the values
retrieved from the method GetGPSInfoL
_LIT(KGoogleMapURL,"http://maps.google.com/staticmap?
center=%S,%S&format=jpg&zoom=8&size=240x320&key=");

// This is my key, Signup in Google maps and create your own key
_LIT(KGoogleMapKey,"ABQIAAAA7mTDJNgGP953Yt-
jOVCpaRSSyDKWhj1AwFHP2_2SAaYvELj_yhTyAVWHzDWGNg0x2EqtccgujgzKwA");

// ---------------------------------------------------------------------------
// CLBSSampleAppView::SendHTTPRequestL()
// send Http request with the latitude and longitude values to google maps
// ---------------------------------------------------------------------------
//

void CLBSSampleAppView::SendHTTPRequestL()
{

const TInt KMaxFolatLength = 8;
const TInt KDecimalPos = 5;

TRealFormat format( KMaxFolatLength, KDecimalPos );
format.iType = KRealFormatFixed | KDoNotUseTriads;

  TBuf8<KDefaultBufferSize> uri8;

  uri8.Append( KGoogleMapURL );
```


(C) Copyright Nokia 2014. All rights reserved.

```
  uri8.Append( KGoogleMapKey );



  HBufC8* longitudeDes = HBufC8::NewLC(257);
  longitudeDes->Des().Num(longitude,format);

  HBufC8* latitudeDes = HBufC8::NewLC(257);
  latitudeDes->Des().Num(latitude,format);


  TBuf8<KDefaultBufferSize> TempBuffer;
  TempBuffer.Format(uri8,latitudeDes,longitudeDes);

  CleanupStack::PopAndDestroy(2);

  // Start transaction
  TRAPD(err, iEngine->IssueHTTPGetL(TempBuffer));
  // TODO: Error handling
  if (err)
      {
      }
}
```

The general URL is as follows as in How to use Google Static Maps data in mobile applications

```
http://maps.google.com/staticmap?
center=41.867878,12.471516&format=png32&zoom=8&size=240x320&key=<API_KEY>
```

```
And I had changed it using my key. It is recommended that you should use your own key.
```

Display the Downloaded map

After the image is download then you have to display it for this you have to use imageconversion API. A nice tutorial on how to read the image file in the filesystem of the phone is provided in thie Wiki PageHow to read images to Symbian bitmap . If you have downloaded other formats of Google maps like PNG or Gif, then also you can make use of the same code.

The following code gives the call to the image reader.

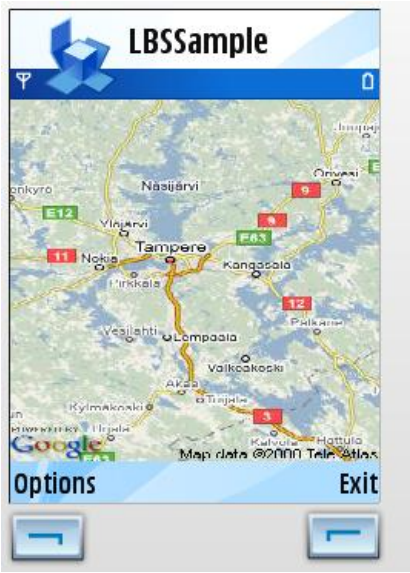
```
_LIT(KTxtFileName, "c:\\Data\\Images\\LbsSample.jpg");
void CLBSSampleAppView::SetImage()
 {
 iJPG_Reader = new(ELeave)CImageReader(*this);
 iJPG_Reader->ConstructL(KTxtFileName);

 }
```

To Display the image on the screen made use of the following code in the Draw Method

```
 if(iJPG_Reader)
   {
    if(iJPG_Reader->Bitmap())
    {
     if(iJPG_Reader->Bitmap()->Handle())
```

 (C) Copyright Nokia 2014. All rights reserved.

```
      {
       gc.DrawBitmap(aRect, iJPG_Reader->Bitmap());
      }
     }
    }
```



The full source code and application can be downloaded from Media:LBSSample.zip

To test the application, you have to import the project to your carbide. After that build the application. Run it on the emulator. Select "Options" --> Select 'GPS Map'. After that select the provider, wait few second here you will see the MAP. In the emulator you will see the MAP of a place in Finland because the emulator returns dummy latitude and longitudinal values of a place in Finland.

One can make changes to this application and make a fine application where a marked postion to be displayed after intervals.

corrected code Media:LBSSample corrected.zip

 (C) Copyright Nokia 2014. All rights reserved.