Integrating Google Maps

We are all familiar with using **Google Maps** from our desktops to show us the locations of places that we are interested in. You can integrate this capability into your mobile apps with a bit of JavaScript coding using a ViziApps ready-to-use library.

This document describes how ViziApps gives you direct access to render locations onto Google Maps. You can see how to search Google Places to find locations in the companion document, **Searching with Google Places**.

Load the **template app** called **Google Places and Maps** to see the following examples in action.

Create a Field for the Map

You must create a field on some page in your app, so the map can be shown. An image field is the easiest way to do this: place the image field onto a page, then move and resize it to exactly the configuration you prefer. You can have multiple maps shown in your app, each rendered into a different ViziApps field. You will pass the name of this field, e.g. *imageFieldForMap*, when you want to render points on a map.

Place an image field into your app design:		The image field renders the Google Map when your app runs:	

Setup the Library

Install the ViziApps JavaScript library into your app's HTML Header. Use the document event GooglePlacesLoaded to guarantee that calls are only made after the library is installed and ready.

```
<script
src="https://s3.amazonaws.com/viziapps/apps/google_places_1.0.2.js"></script>

<script>
var goomaps;
$(document).on( 'GooglePlacesLoaded', function()
{
    // create an instance of the google maps api:
    goomaps = new v.googlePlaces();
});
```

```
function showCities()
{
    if ( !goomaps ) return;
    // make calls to the google maps library, such as:
    var pts = [
        { name: 'Boston', lat: 42.361592, lng: -71.060419 },
        { name: 'Chicago', lat: 41.878227, lng: -87.649415 },
        { name: 'Dallas', lat: 32.786036, lng: -96.803094 },
        { name: 'Seattle', lat: 47.606979, lng: -122.319019 }
    ];
    gotoPage( 'pageWithMapShowing' );
    goomaps.showpoints( 'imageFieldForMap', pts );
}
</script>
```

Google Places Search Results

If you have search results from Google Places, you can render these on a map with a single call:

```
goomaps.showmap( 'imageFieldForMap' );
```

InfoWindow Callback

When you tap on a pin in a map, a popup shows some information about the pin - this is called an InfoWindow. The main (bold) text will show a **.name** or **.title** property, and its subtext will show a **.address** or **.subtitle** property. The InfoWindow also contains an icon that the user can tap to explore the place further.

In order for your app to take an action after a user taps on an InfoWindow, for example change to a detailed view of the place, you need to provide a callback function when you call **.showpoints()** or **.showmap()**.

```
// callback function after a map pin has been tapped
function pinTapped( placeObject ) {
    alert( 'You picked: ' + placeObject.name );
}

// showmap automatically uses the most recent search results
goomaps.showmap( 'imageFieldForMap', pinTapped );
```

Satellite View

A Google Map will render as a roadmap by default. You can change the background to a satellite view with the **.maptype()** method.

```
// change to satellite view
goomap.maptype( 'satellite' );
// change to roadmap view
goomap.maptype( 'roadmap' );
```

Map Options

InfoWindow Icon

You can change the icon that the user taps in the InfoWindow by setting the **.infoicon** property to any URL, however this should be a small PNG image.

Map User Controls

You can enable or disable the controls that appear on the map by changing these properties to true or false:

.controls.zoom	Controls the zoom level with +/- buttons
.controls.streetView	Shows the Street View pegman on the map
.controls.mapType	Shows a control that switches between satellite and roadmap views

From:
https://viziapps.com/dokuwiki/ - **ViziApps Help Wiki**

Permanent link:
https://viziapps.com/dokuwiki/integrating_google_maps

Last update: **2015/01/14 12:04**