# Public Information Web Map

## Instructions for configuration tasks

# Contents

## Skills

The Public Information Web Map uses the ArcGIS API for JavaScript. The core capabilities of the application have been exposed in a configuration file written in a web language called JSON. JSON is just like XML except is it formatted differently, with {} instead of <>. If you are unfamiliar with JSON, see Appendix A for a simple JSON explanation. Modify the configuration file (*config/config.js*) to make your own custom Public Information Web Map.

For success with this document and configuring the application, you should have a minimum level of understanding of:

- ArcGIS Online map creation and sharing, including where to get the web map ID
- Modifying JSON values (see Appendix A)
- What you want your application to look like
- How you want users to interact with your application

## Application overview

Explore a sample application, using the interface to become familiar with the application capabilities.

- A basemap selector
- Layers list with authoritative content
- Social content including the ability to change the social settings (keywords, …) and visualize as clusters or trends (heatmap)
- Places panel for adding spatial bookmarks
- Share panel for sharing the configured map
- Ability to locate a place with the search box
- An About page with information about the application

## Application setup

The Public Information Web Map application will need to be published on a web server for users to view. Unzip the downloaded files to a directory on your web server. If you move the application files after unzipping them, be sure to maintain the original directory structure. Contact your IT personnel if you are unsure of how to publish to the web server.

## Create an ArcGIS Online web map

The Public Information Web Map application displays a web map from ArcGIS Online (http://www.arcgis.com). Before attempting to customize this application to suit your purposes, create a web map on ArcGIS Online and set its sharing visibility to *Everyone*. For more information on creating and sharing web maps using ArcGIS Online, please reference About authoring web maps on resources.esri.com.

## Tasks for configuring the Public Information Web Map application

The Public Information Web Map application customization options found in the application configuration file will change the look and functionality of the application components (menus, title,

social media, etc.). Although the application interface displays layer names and symbology from the web map, you will need to edit the ArcGIS Online web map on ArcGIS.com to configure these values.

By using ArcGIS Online and the Public Information Web Map application together, you can deliver an elegant mapping application to your audience. Use this task guide to configure the Public Information Web Map application and discover the customization possibilities.

## Configure the application using the Share menu

### *Task 1: Edit the configuration from within the application*

This task will teach you what configuration options are available from the user interface, with no programming required.

The Sharing capability bundles changes made to the application into the share URL. This can reduce the amount of editing done in the config file for small updates once the application is initially setup.

Using the sample application above or linked off of ArcGIS.com:

1. Make the following changes:
    a. Change the map extent
    b. On the Basemap dropdown, change the base map to something other than default
    c. On the Layers dropdown, check off or on the visibility of one or more layers
    d. On the Social dropdown, click the gear and change the keyword (and any other parameters) for at least one of the social media layers
    e. In the Places menu, add at least one place.
2. Click the Share button and copy the 'Share a link to your map' URL
3. Paste this URL into a new browser and see that the changes you made above are remembered

Note - any values defined in the Share URL take priority over values defined in the config file. When you use the share URL, any bookmarks you have saved will only preserve the spatial extent. Any other application configurations saved with the bookmarks will not persist.

## Configure the application using the config file

The rest of the tasks outlined in this document share a common process to make permanent changes to the application configuration code:

1. Open the config file in a text editor such as notepad or notepad++
2. Identify the value you wish to modify
3. Delete the old value and type your new custom value in its place
4. Save the config file, overwriting the old config values
5. Test the changes to make sure the application functions as you intended

Note – on completing changes to JSON text, always check for errors. Poorly formed JSON will cause the application to crash in certain web browsers, including Internet Explorer. To check your edits:

1. Copy code into JSONLint.com - http://jsonlint.com/
2. Press validate
3. Correct any errors

## Config file description

*config/config.js* – The config file is bundled with the application download and can be found in the config folder. The config file defines configuration values for the application functionality, user interface, and social media layers.

## General application tasks

### *Task 2: Set the map document using mapID*

**`"webmap":`** The **`webmap`** ID is the unique combination of letters and numbers found on the end of the URL of your ArcGIS Online web map. To obtain the **`webmap`** ID, sign in to ArcGIS.com, navigate to your web map resource, open the web map in the ArcGIS Online viewer, and copy the last portion of the URL. If the URL appears like http://www.arcgis.com/home/webmap/viewer.html?webmap=d5e02a0c1f2b4ec399823fdd3c2fdebd, your **`webmap`** ID is **`d5e02a0c1f2b4ec399823fdd3c2fdebd`**.

### *Task 3: Set the proxy page*

Proxy pages are required by the ArcGIS API for JavaScript, which the Public Information Web Map is built on. The needed proxy page is included with this package download and can be found at resources/proxy.ashx. You can use a custom proxy page and custom proxy settings by changing the proxy values in the config file.

**`"proxyURL":`** Leave this unchanged to use the proxy page packaged with the application. If using another proxy page, change the **`proxyURL`** value to point to the desired proxy page.

**`"alwaysUseProxy":`** If the application should use the proxy page for all network requests, set the value to **`true`**, if not leave the value **`false`**.

To learn more about using and customizing the proxy page, visit Using the proxy page on resources.esri.com.

### *Task 4: Set the geometry service*

If your webmap requires a geometry service to display properly, use the **`geometryserviceurl`** option to define the geometry service to use.

**`"geometryserviceurl":`** Set to a custom URL if needed. The default value will be satisfactory for most web maps.

To learn more about using a geometry service, visit About the geometry service on resources.esri.com.

### *Task 5: Add and remove menus from the application*

The Public Information Web Map application lets users switch between Basemap, Layers, and Social menus. Each menu can be removed from the application, if desired.

**"showBasemapMenu":** To add or remove the Basemap menu button from the user interface, set the value to **true** or **false**. When **true**, users can see the Basemap menu and change the basemap. Note – if a custom basemap is used in your web map, it will not appear as an option on the Basemap menu. If users select another basemap while using your application, they will have to reload the application to view the custom basemap again.

**"showLayersMenu":** To add or remove the Layers menu button from the user interface, set the value to **true** or **false**. When **true**, users can interact with the map layers (turn on/off, adjust transparency, view legends, etc.).

**"showSocialMenu":** To add or remove the Social menu button from the user interface, set the value to **true** or **false**. When **true**, users can adjust the social options (turn on/off layers, change search terms and dates, etc.).

**"defaultMenu":** Users can choose between 3 map menus while interacting with the Public Information Web Map template. Set the default open menu by setting the value of **defaultMenu** to **basemap**, **layers**, or **social**. Note – for the defaultMenu option to work correctly, make sure the desired menu is included in the application by setting the corresponding **showBasemapMenu**, **showLayersMenu**, or **showSocialMenu** value to **true**.

**"bingMapsKey":** An alpha-numeric key is required to display base maps by Bing Maps. When the value is a valid key, Bing Maps' base map options are automatically added to the Basemap menu. If needed, generate a key by creating a developer account at http://www.bingmapsportal.com/ and selecting "Create or view keys".

### *Task 6: Add and remove user interface features from the application*

Individual elements of the application can be removed or added using the following values:

**"showSearchBox":** To add or remove the 'Find a place' search box to let users search for a place, set the value to **true** or **false**. If **true**, you must also provide a value for **locatorURL**.

**"locatorURL":** URL of the geocode server used to find places. Users type a place name into the search box and press enter to send a geocode response to this resource. The default geocode server can be referenced for non-commercial use. For more information about using

Esri's World Geocoding Task Service, please see the World Geocoding Task Service information page.

**"showPlaces":** To add or remove the Places menu button from the user interface, set the value to **true** or **false**. When **true**, users can store the current view in a bookmark. Up to 5 bookmarks can be saved at a time. Note – if more than 5 bookmarks are defined in your ArcGIS Online web map, the first 5 will appear in the Places menu. The others will not be loaded in the application.

**"showGeolocation":** To add or remove the option to find the current location of the user, set the value to **true** or **false**. When **true**, a link to My Current Location is displayed at the top of the Places menu. Note – depends on **showPlaces** to be **true**.

**"showShareMenu":** To add or remove the Share menu button from the user interface, set the value to **true** or **false**. When **true**, users can access options for sharing the application by URL, Facebook, Twitter, or embedding in websites.

**"bitly":** Use Bitly to shorten the URL link to the application. Users will copy this shortened URL from the Share menu. Find out more about using Bitly to shorten URLs at http://dev.bitly.com/.

| Option | Value |
|---|---|
| "login" | Bitly login alias |
| "key" | Bitly API key |
| "APIURL" | "http://api.bit.ly/v3/shorten" |

**"showScalebar":** To add or remove the scalebar on the map, set the value to **true** or **false**, respectively.

**"pointSymbol":** URL of the place marker symbol image. Default is a blue push pin.

### *Task 7: Set a custom about page*

An About page is where users can go to get more detailed information about your application.

**"aboutPage":** Set this value to the URL of the page users will access to find out more information about the application. There are three common values:

| Value | Result |
|---|---|
| "" | About link removed from user interface; users won't see a link |
| "http://www.arcgis.com/home/item.html?id={mapID}" | Opens ArcGIS Online details page for the web map; replace **{mapID}** |

| | |
|---|---|
| | with the **mapID** of the ArcGIS Online web map |
| "http://www.mywebsite.com/about.html" | Opens custom web page; replace the sample URL with one to a custom webpage |

## Social media tasks

The social media settings define the initial settings seen by the user in the social media settings panel. Access these settings in the GUI by clicking the gear icon next to one of the social layers.

### *Task 8: Set a default search area for social media layers*

In order to request geo-located social content from social media networks, it is mandatory to indicate a geographic scale. The **socialGeo** parameter defines the point for which a geographic search will begin and the **socialDistance** parameter defines the search area.

Leaving the **socialGeo** parameter empty will set the search area to the map extent center and therefore the end user will always get social content for the current extent they are viewing.

```
"socialGeo": {
            "x": "",
            "y": ""
      },
```

Set the **socialDistance** to define the search distance from the center point set above. Each API defines a limit to the quantity of information returned with each request, so tuning this value to the scale of the map will provide optimal results. Valid values are **local**, **regional**, or **national**.

```
"socialDistance": "regional",
```

### *Task 9: Set the social media parameters for individual social networks*

Flickr, Twitter, YouTube, and Ushahidi support 3rd party requests for information. The Public Information Web Map is built to leverage this ability and display geographically coded social media. Update the following values to configure the social experience and content:

Common options for Flickr, Twitter, YouTube, and Ushahidi are listed in the table below:

| Option | Nested option | Value |
|---|---|---|
| "enabled": | | **true** or **false**; when **true**, social network is searched and results included on map |
| "visible": | | **true** or **false**; when **true**, social network appears in the Social menu list and on the map |
| "uniqueID": | | A unique identifier for each layer; used by the Public Information Map app to keep track of layers |

| | |
|---|---|
| `"title":` | Text to appear in Social menu list |
| `"description":` | Text to appear in social menu list layer description box; appears when users click the icon |
| `"legendIcon":` | The URL to an image which appears next to the Title |
| `"symbol": "URL":` | The URL of the map marker to identify the social results |
| `"width":` | Width of the marker |
| `"height":` | Height of the marker |

**`"Ushahidi":`** For information about using Ushahidi, visit http://www.ushahidi.com/.

**`"apiURL":`** URL of the Ushahidi service that the Public Information Map will query. All results from the Ushahidi resource will be displayed on the map.

**`"Flickr":`** For information about using the Flickr API, visit http://www.flickr.com/services/developer/api/.

**`"showSocialSettings":`** Set this to **`true`** or **`false`**; when value is **`true`**, the icon appears to the right of the Flickr title in the Social menu list. Users are able to adjust the social search settings through interacting with a pop-up menu.

**`"key":`** Set this value to the unique developer key associated with the application. Obtain a key by visiting: http://www.flickr.com/services/developer/api/ and clicking the 'Request an API key' link.

**`"searchTerm":`** Set this value to search Flickr for photos tagged with the same term. For example, if the application highlights earthquakes, set the value to **`"earthquake"`**.

**`"flDateFrom":`** and **`"flDateTo":`** Limit the results of a Flickr search, specify a starting date as this value. The default "" values will search Flickr from one month prior to the use of the application. Use the format *mm-dd-yyyy* if specific dates are required.

**`"Twitter":`** For information about using the Twitter API, visit https://dev.twitter.com/.

**`"showSocialSettings":`** Set this to **`true`** or **`false`**; when value is **`true`**, the icon appears to the right of the Twitter title in the Social menu list. Users are able to adjust the social search settings through interacting with a pop-up menu.

**`"searchTerm":`** Set this value to search Twitter for photos tagged with the same term. For example, if the application highlights earthquakes, set the value to **`"#earthquake"`**.

**"YouTube":** For information about the YouTube API, visit https://developers.google.com/youtube/.

**"showSocialSettings":** Set this to **true** or **false**; when value is **true**, the icon appears to the right of the YouTube title in the Social menu list. Users are able to adjust the social search settings through interacting with a pop-up menu.

**"searchTerm":** Set this value to search YouTube for photos tagged with the same term. For example, if the application highlights earthquakes, set the value to **"earthquake"**.

**"YTRange":** Limit the results of a YouTube search by setting the value to a valid YouTube time frame. Valid values are today, **this_week**, **this_month**, and **all_time**.

**"popupWidth":** and **"popupHeight":** Set the value of the width and height (in pixels) of the social media pop-ups.

**"clusterImage":** When multiple social media points occupy the same place on the map, a cluster image will replace the individual symbols. Set this value to a custom cluster image URL to change the cluster default cluster image.

### *Task 10: Set up social media filtering and flagging features (advanced)*

If your organization requires the application to filter social media content and/or to give users the ability to report inappropriate social media content, the Public Information Web Map template provides a solution. URLs pointing to filtering and flagging services are available in the config file:

**"inappSvcURL":** Service URL to a table of banned social media authors. The authors are banned using the Moderate Social Media workflow shared at arcgis.com.

**"badWordsURL":** Service URL to a table of words which you do not want to appear in your app (expletives, vulgarities, etc.). Publish this table as part of a map service or feature service using the schema below.
*Schema*: field of type text with alias "words" and field with the OID

**"FAI_Email":** Set this value to an email alias the application will send notifications to when inappropriate social content is flagged (see Moderate Social Media workflow).

**"mailServer":** URL to PHP mail server script. The script makes arbitrating flagged content simple by generating and sending an email to **FAI_Email**. The generated email will contain links to arbitrate the flagged social media (confirm as offensive, mark as accidental flag, etc.). When clicked, the link will open a web page which will interact with the feature service to update the underlying database. (See Moderate Social Media workflow.)

To learn more about setting up filtering and/or reporting services to work with your application, see the Moderate Social Media workflow shared at arcgis.com.

Download the code and follow the documentation. Note that references to "your application" are referring to the Public Information Web Map template, which is ready to work with the PHP script from the Moderate Social Media download.

## Configuration file identification tasks

The following tasks are for identification of the remaining elements of the *config.js* file. They are included for informational purposes only. Any modifications to the following values risk disabling the normal functionality of the application.

### *Task 11: Identify the Basemap menu values*

By default, the Basemap menu includes base map options from the ArcGIS Online Basemap menu.

**`"title":`** This value is the title of the base map group on ArcGIS Online.

**`"owner":`** This value is the ArcGIS Online username of the base map group owner.

**`"initial":`** This value is the ID of the ArcGIS Online resource to be used as the default base map when the application is initially loaded.

✖ It is not recommended to use a custom base map group in the Social Media Map template. The information above is presented for document completeness. Please do not alter the original settings. Use the following reference if you need to restore the original values:

```
"basemaps": {
      "title":"ArcGIS Online Basemaps",
      "owner":"esri_en",
      "initial":"6e03e8c26aad4b9c92a87c1063ddb0e3"
      },
```

### *Task 12: Identify the build version*

**`"buildVersion":`** Do not change this value. This is the release number of the Public Information Web Map template that your application uses.

## Appendix A: Basic JSON Explanation

The *config.js* file contains JavaScript Object Notation (JSON) text, which sets the configuration values to customize the application. To edit the values in *config.js*, open the file in a text editor such as notepad or notepad++.

The elementary unit of JSON is a pair of related values known as a key/value pair and written as **"key": "value"**. The configuration options in *config.js* are arranged as key/value pairs. To set the value of a configuration option, replace "value" with the custom option.

*Example 1*

```
"mapID": "8b3d38c0819547faa83f7b7aca80bd76"
```

The *config.js* file also uses another standard feature of JSON: nested elements. A nested element is simply a key/value pair nested inside another key/value pair. To make a nested key/value pair, start with the elementary key/value pair structure and replace its value with another key/value pair nested inside curly brackets – {}. The structure is **"key": { "key": "value" }**.

Commas are used to separate one key/value pair from another.

*Example 2*

```
"socialGeo": { "x": "-11687900.88", "y": "4830562.43" }
```

To make JSON more human-readable, new lines and indents are added at logical break points. An example of a logical break point would be after an opening bracket. Here is the previous example with new lines and indents:

*Example 3*

```
"socialGeo": {
      "x": "-11687900.88",
      "y": "4830562.43"
}
```

With this information, notice that *config.js* begins and ends with curly brackets {} (first and last lines of the file). This indicates that the entire file is a nested JSON structure. To edit the file, make sure that no curly brackets are deleted and that every opening bracket, {, has a closing bracket, }.

JSON accepts numbers, text, and true/false values. Numbers and text values need to be nested in quotation marks, "". True/false values should be written in all lowercase and not nested in quotes. See the sample *config.js* for example. Remember to always end a key/value pair with a comma whenever another key/value pair follows it.

The following resources are recommended if you want to learn more about JSON:

- W3Schools JSON Tutorial - http://www.w3schools.com/json/default.asp
- MSDN Library - http://msdn.microsoft.com/en-us/library/bb299886.aspx
- JSON.org - http://www.json.org/
- Wikipedia.org - http://en.wikipedia.org/wiki/JSON
- JSON Lint - http://www.jsonlint.com/