**Overview**

This is another example of integrating ESRI into Public Sector Solutions and Omnistudio.  This example is a more basic version that allows for the selection of one polygon on a map without selecting a schedule.  The example is a very simple example and is not intended to be directly implemented.  Instead the example is intended to provide insights on how the accelerator might be used for other use cases.  Certain design decisions were made based on simplicity of reusing objects already in use by the ESRI Connector.  A full design review is required before implementing this into a production environment.

No Salesforce package is provided with this example.  The instructions below will walk through how to add and view the Basic Map example in a Salesforce Org.

**Highlights of the Basic Map**

* Reuse of many of the features of the package accelerator
* Seeding the map with the coordinates of a Contact instead of a static starting point stored in Custom Metadata
* Displaying a map without having to select a scheduled time
* Validation to only allow on drawn polygon on the map
* Reviewing the drawn polygon directly on the Business License Application record

**Inventory of Components**

Omniscript

* Basic Map Omniscript.json - Omniscript that contains the map, calls to a Datraptor to create supporting objects and a confirmation step.  The Omniscript is very simple and only used as a very basic example

Lightning Web Components -

* basicMap - Allows an applicant to review a submitted map on the Community
* basicMapCreate - Embedded inside of the Omniscript to allow an applicant to draw a polygon on a map
* basicMapReviewerApproval - Allows a reviewer to review a submitted map with a polygon.  Reviewers can make updates to the location and approve or reject the drawn polygon.

Visualforce Pages

* Basic\_Map\_Create\_Page - Used by the basicMapCreate LWC to allow applicants to draw a polygon
* Basic\_Map\_Page - Used by the baseMap WC to allow applicants to review submitted polygons
* Basic\_Map\_Reviewer\_Approval\_Page - Used by basicMapReviewerApproval to allow reviewers to review and make updates to submitted polygons.

Apex Classes

* Basic\_Map\_Create\_Ctr - Apex controller for Basic\_Map\_Create\_Page Visualforce page
* Basic\_Map\_Ctr - Apex controller for Basic\_Map\_Page Visualforce page
* Basic\_Map\_Reviewer\_Approval\_Ctr - Apex controller for Basic\_Map\_Reviewer\_Approval\_Page Visualforce page

**Assumptions**

* This example assumes that Public Sector Solutions with Omnistudio has been properly installed
* Ideally the Experience Cloud site used for this example should not have a suffix in the URL.  If the Experience Cloud site does have a suffix then the Experience Cloud user must have access to the Experience Cloud site that does not have a suffix and other setup described below is required.
* Lightning Web Security must be enabled through Setup-> Session Settings
* For preseeding of the map on the Community, the Contact Mailing address is used for the coordinates of the Community user.
  + In order for the preseeding to work properly, the Geocodes for Contact Mailing Address Data Integration Rule must be enabled.  See more [here](https://help.salesforce.com/s/articleView?id=sf.ddc_data_integration_rule_activating.htm&type=5) for details.
* This example is based on the delivered Unmanaged package and uses many of the assets from this package.
* This example works with either Standard Runtime or Managed Package runtime
* Experience Cloud user and profile settings have been setup to allow for use of LPI features and executing Omniscripts

**Installation Instructions**

1. Installed **Unmanaged** Package for the ESRI Accelerator.
   1. <https://login.salesforce.com/packaging/installPackage.apexp?p0=04tDp000000VCdB>
2. Add Apex Classes
   1. Basic\_Map\_Create\_Ctr
   2. Basic\_Map\_Ctr
   3. Basic\_Map\_Reviewer\_Approval\_Ctr
3. Add Visualforce Pages
   1. Basic\_Map\_Create\_Page
   2. Basic\_Map\_Page
   3. Basic\_Map\_Reviewer\_Approval\_Page
4. Add LWCs
   1. basicMap
   2. basicMapCreate
   3. basicMapReviewerApproval
5. Import and Activate the Basic Map Omniscript
6. Add VF Pages (Base\_Map\_Create\_Page and Basic\_Map\_Page) to Site through Workspaces -> Administration -> Pages -> Go to force.com
   1. Note:  If the Experience Cloud site has a suffix then this step also must be performed on the Experience Cloud site that does not have a suffix.
7. Add page to site where Basic Map Omniscript will sit and add the Omniscript
8. On the site add tab to Business License Application page on site with a tab pointing to the basicMap
9. Add a tab to the Business License Application Lightning Page On the tab add the basicMapReviewerApproval LWC
10. Set Custom Metadata
    1. utils
       1. APIKey - Set to your ESRI API Key
       2. SiteName - Name of community
11. Assign permissions to Community User by adding ESRI Community Permission Set
12. Grant Read, Write, Edit on Polygon and Polygon\_Schedule\_\_c objects and all fields

**Troubleshooting**

* If the Experience cloud site you are using has a suffix, make sure that Experience Cloud user has access to the site that does not have a suffix and also ensure that the Visualforce pages have been added to the site without a suffix as shown in step 6 above
* Clearing cache
* Refresh browser
  + This may be required several times the first time you attempt to launch the Omniscript.
* Checking the Browser Console for issues
* Check the Mailing Address of the Community User’s Contact record
* Make sure LWS is enabled under Session Settings
* The included dataraptor assumes a Regulatory Authorization Type of Building Permit.  Verify that there is a Regulatory Authorization type called Building Permit or change the Dataraptor to use a different LicenceTypeId

**Disclaimer**

I am a pre-sales Solutions Engineer for [Salesforce](https://www.salesforce.com/) and I develop solutions for customers to demonstrate the capabilities of the amazing Salesforce platform. *This package represents functionality that I have used for demonstration purposes and the content herein is definitely not ready for actual production use; specifically, it has not been tested extensively nor has it been written with security and access controls in mind.*