Developing a Web Mapping Application for Collaborative Dataset Management

Travis Bock
Innovate! & Penn State
Travis Bock

Geospatial Developer, Innovate, Inc.

Contractor for Local, Territorial and Federal Governments and private organizations

Build End-user solutions to collect/manage geospatial data and help answer questions.
Intro

GIS should be an integral part to an organization's day to day activities

Better data organization

Increase collaboration

This approach can apply to many different databases

For this application I focused on Geographic Response Planning for the EPA
Building Blocks

- HTML5
- CSS3
- JavaScript
  - ArcGIS JS API 3.x
  - AngularJS 1.x
  - Angular Material 1.x
- ArcGIS Server/Online
What is a Geographic Response Plan?

Plan covering loosely defined geographic area

Used for early stage response to oil, chemical or any spill that could cause serious harm to sensitive sites.

Comprised of Sensitive Sites (points), Containment Booms (lines), Incident Action Plans (polygons) and various related tables.
Sensitive Sites with Strategies and Booms

- Environmental
- Economical
- Historical
- Cultural
- Archeological
Incident Action Plan

- Early response coordination
- Fire, Police and Medical resources
- Spill response resources
- Local, Regional and National contacts
Who creates these plans?

- Federal, State, Territory and Local Emergency Response
- First Responders
- Environmental Scientist
- Local/Regional Experts
- Various subject matter experts
Current tools

- Email
- Microsoft Word
- ESRI Web AppBuilder
Current database

Single flat table
Attributes are repeated
Relationship is 1:7 instead of 1:Many
Missing lots of information
No storage of Incident Action Plan in data model
What’s wrong with that?

- Spatially enabled in a disconnected manner
- Lack of access controls
- Not centralized (many versions could exist)
- Difficult to collaborate
- Poor quality data model

How can we do better?

- User requirements gathering
- New data model
- Create custom user interface
New Sensitive Site Data Model

Actual 1:Many relationships

All relevant information is stored in a single place
New Incident Action Plan Data Model

Provides a single location to store plan information

Can be reproduced for many areas without much effort.
What will this look like

Start from scratch?

Use graphic designer?

This UI is Awesome!
Graphic Designers!

User requirements

Data structure

Create static mockup to present to end users for feedback

Starting point for developing UI
How to build

Ajax
Client-side
HTML
CSS
JavaScript
API's
Frameworks
But what about ESRI Web AppBuilder?

- Built using Ajax technology
- Custom Widgets
- Good for visualization
- Bad for data entry
User Interface Frameworks

- Faster than starting from scratch
- Angular Material
- Dojo
- Bootstrap
- Other
Building Blocks

- HTML5
- CSS3
- JavaScript
- ArcGIS JS API 3.x
- AngularJS 1.x
- Angular Material 1.x
- ArcGIS Server/Online
Software as a Service

ArcGIS Online
Still need that server!

ArcGIS Server

Others?

JSON (ESRI JSON vs GeoJSON)
Cloud Based ArcGIS Server

Amazon Web Services

Microsoft Azure
End Product
General

Name:
Tumon Area Release IAP

Short Name:
Tumon Incident IAP

Executive Summary:
TB added

Cover Image:
Factory Pattern with esriLoader.require()

```javascript
angular.module('GRPApp').factory('relatedFeatureFactory',
  function ($q, esriLoader, $mdToast, $filter, $mdDialog, $state, esriAuth) {
    'use strict';
    var relatedFactory = function (feature, foreignKeyField, foreignKey) {
      var deferred = $q.defer(),
          Query,
          Graphic,
          arcgisService;

      esriLoader.require(['esri/tasks/query', 'esri/graphic'],
        function (_Query, _Graphic) {
          Query = _Query;
          Graphic = _Graphic;
          deferred.resolve();
        });
    }
  });
```
Factory Pattern with `esriLoader.require()` cont’d

```javascript
function isLoaded() {
    return deferred.promise;
}
```

```javascript
get: function (globalId, objectId) {
    isLoaded().then(function () {
        var query = new Query(),
```
Using field metadata in template

```javascript
function initFields(obj) {
    angular.forEach(obj.layer.fields, function (field) {
        obj.fields[field.name] = field;
    });
}
```

```html
<md-input-container>
    <label>{{coastalSites.site.fields.Site_ID.alias}}</label>
    <input type="text" ng-model="coastalSites.site.attributes.Site_ID"
        maxlength="{{coastalSites.site.fields.Site_ID.length}}"/>
</md-input-container>
```
Conclusion

Web based spatially enabled tools will increase planning participation and collaboration.

Collaboration increases the quality of information in the plan.

Always available applications will increase the likelihood of having current information.

Web base client side applications backed by cloud based services insures data will always be available.
Questions?