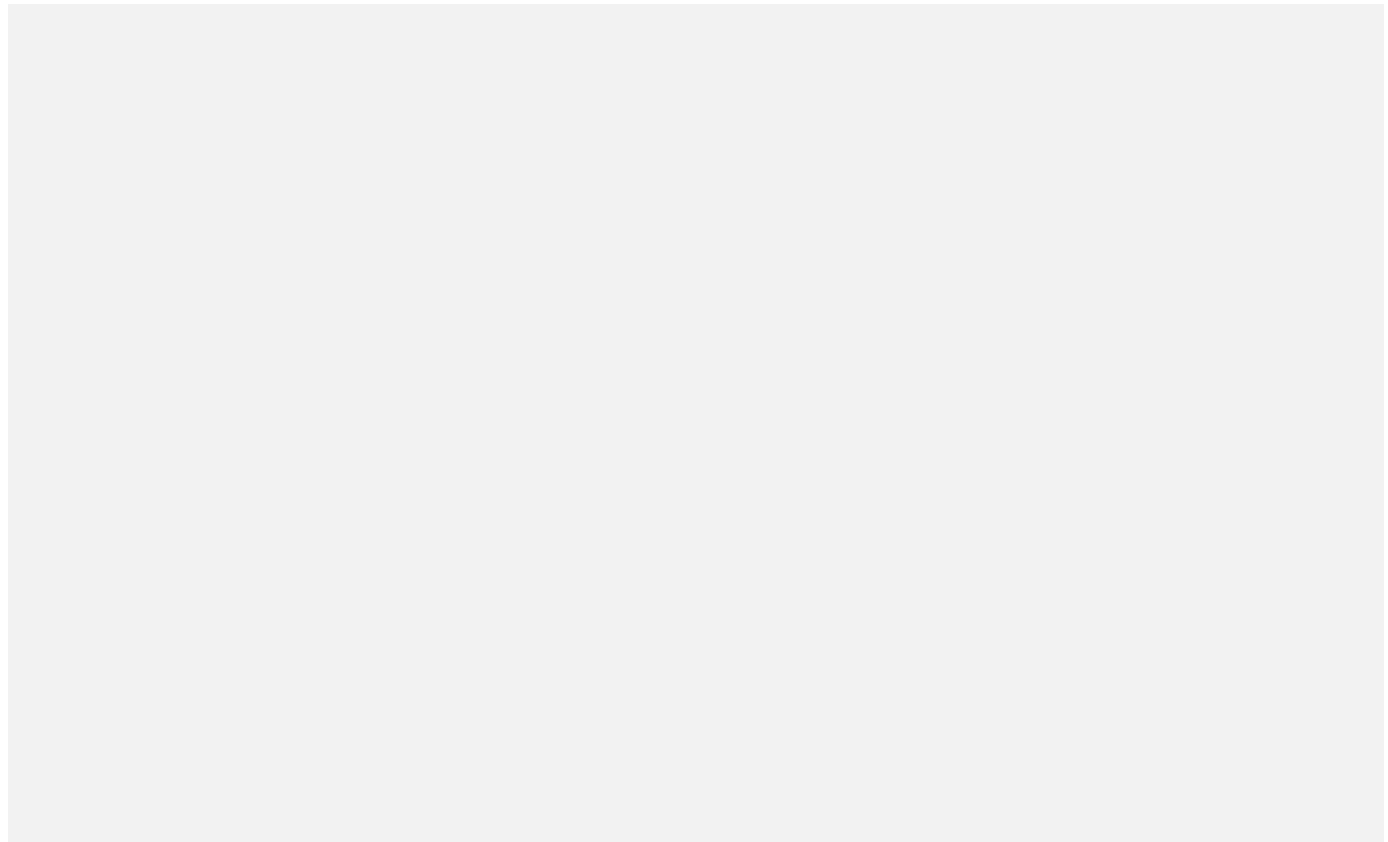


# Creating mashups with Google maps and ConfiForms

Using ConfiForms PlainView together with some scripting to show ConfiForms data on Google map



Showing on the map (2 markers)

|                  |                                              |
|------------------|----------------------------------------------|
| Latitude         | 40.730610                                    |
| Longitude        | -73.935242                                   |
| Delivery Address | brooklyn Bridge, New York, NY, United States |

One marker is added with lat/long and another marker added using the address (which is then decoded to lat/long and shown)

Storage format for the solution

```
<p>Using ConfiForms PlainView together with some scripting to show ConfiForms data on Google map</p>
<ac:structured-macro ac:macro-id="870652ad-599d-4d1f-bdc8-bd6c4ba4e7a8" ac:name="html" ac:schema-version="1">
  <ac:plain-text-body><![CDATA[<script
    src="https://maps.googleapis.com/maps/api/js?key=AIzaSyAnRA1CvBUdFF22IYWlK_dKmW71Vl-Mz7M&v=3.
exp&sensor=false"></script>
<style type="text/css">
.minimap {
  min-height: 420px;
  margin-top: 10px;
  background-color: #f2f2f2;
  /*-webkit-box-shadow: 0 2px 12px rgba(0,0,0,.5);*/
  /*-moz-box-shadow: 0 2px 12px rgba(0,0,0,.5);*/
  /*box-shadow: 0 1px 2px #000;*/
}
</style>
<div id='mapOnWorkItemDiv' class="minimap"></div>
```

```

<script>
// map is global... mmm...
var map;
function geoCodeAddress(latlng, onFind) {
    var geocoder = new google.maps.Geocoder();
    geocoder.geocode({'latLng': latlng}, function(results, status) {
        if (status == google.maps.GeocoderStatus.OK) {
            if (results[1]) {
                onFind(results[1].formatted_address);
            }
        }
    });
}
function addEndLocationMarker(location) {
    if (location != null && location != '') {
        var geocoder = new google.maps.Geocoder();
        geocoder.geocode({'address': location}, function (results, status) {
            if (status == google.maps.GeocoderStatus.OK) {
                var elem = $('#endAddressChoice');
                if (results.length == 1) {
                    if (elem) {
                        elem.hide();
                    }
                    addMarkerOnMap(results[0].geometry.location);
                } else {
                    if (elem) {
                        var rows = '<div class="alert alert-info" id="endAddressChoiceTab" style="display: none">';
                        for (var i = 0; i < results.length; i++) {
                            rows = rows + '<p style="font-size: smaller"> <a href="#" onclick="
removeAddressChoiceAndSetMarker(' + '\'' + results[i].formatted_address + '\'' + ');addMarkerOnMap(new google.
maps.LatLng(' +
                                results[i].geometry.location.lat() + ',' + results[i].geometry.location.lng() + '));return
false;">' + results[i].formatted_address + '</a></p>';
                        }
                        elem.html(rows + '</div>');
                        $('#endAddressChoiceTab').fadeIn()
                    }
                }
            } else {
                if ($('#endAddrLatitude')) {
                    $('#endAddrLatitude').val('');
                }
                if ($('#endAddrLongitude')) {
                    $('#endAddrLongitude').val('');
                }
            }
        });
    }
}
function initMapOnWorkItem(location) {
    if (location != null && location != '') {
        // $('#mapOnWorkItemDiv').hide();
        var geocoder = new google.maps.Geocoder();
        geocoder.geocode({'address': location}, function (results, status) {
            if (status == google.maps.GeocoderStatus.OK) {
                var elem = $('#addressChoice');
                if (results.length == 1) {
                    if (elem) {
                        elem.hide();
                    }
                    showLocationOnMapDiv(results[0].geometry.location);
                } else {
                    if (elem) {
                        var rows = '<div class="alert alert-info" id="addressChoiceTab" style="display: none">';
                        for (var i = 0; i < results.length; i++) {
                            rows = rows + '<p style="font-size: smaller"> <a href="#" onclick="
removeAddressChoiceAndSetAddress(' + '\'' + results[i].formatted_address + '\'' + ');showLocationOnMapDiv(new
google.maps.LatLng(' +
                                results[i].geometry.location.lat() + ',' + results[i].geometry.location.lng() + '));return
false;">' + results[i].formatted_address + '</a></p>';
                        }
                    }
                }
            }
        });
    }
}

```

```

        elem.html(rows + '</div>');
        $('#addressChoiceTab').fadeIn()
    }
}
} else {
    $('#mapOnWorkItemDiv').css('min-height', '160px').css('border', '0').html('<div class="alert alert-
danger" style="height: 100%">We could not locate the address specified :(</div>').show();
    if ($('#addrLatitude')) {
        $('#addrLatitude').val('');
    }
    if ($('#addrLongitude')) {
        $('#addrLongitude').val('');
    }
}
});
}
}
function setAddressBack(address, elem, tabToHide) {
    var addressElem = $(elem);
    if (addressElem) {
        addressElem.val(address);
    }
    $(tabToHide).hide();
}
function removeAddressChoiceAndSetMarker(address) {
    setAddressBack(address, '#endInputAddress', '#endAddressChoiceTab');
}
function removeAddressChoiceAndSetAddress(address) {
    setAddressBack(address, '#inputAddress', '#addressChoiceTab');
}
function addMarkerOnMap(loc) {
    if ($('#endAddrLatitude')) {
        $('#endAddrLatitude').val(loc.lat());
    }
    if ($('#endAddrLongitude')) {
        $('#endAddrLongitude').val(loc.lng());
    }
    addMarker(loc);
}
function showLocationOnMapDiv(loc) {
    var mapOptions = {
        zoom: 13,
        center: loc,
        mapTypeId: google.maps.MapTypeId.ROADMAP
    }
    $('#mapOnWorkItemDiv').css('height', '420px').css('border', '#555555 1px solid').fadeIn();
    if ($('#addrLatitude')) {
        $('#addrLatitude').val(loc.lat());
    }
    if ($('#addrLongitude')) {
        $('#addrLongitude').val(loc.lng());
    }
    map = new google.maps.Map(document.getElementById('mapOnWorkItemDiv'), mapOptions);
    map.setCenter(loc);
    addMarker(loc);
}
function addMarker(loc) {
    var marker = new google.maps.Marker({
        map: map,
        animation: google.maps.Animation.DROP,
        position: loc
    });
    tryToDisplayRoute();
}
function tryToDisplayRoute() {
    if ($('#addrLatitude') && $('#addrLatitude').val() != '' &&
        $('#addrLongitude') && $('#addrLongitude').val() != '' &&
        $('#endAddrLatitude') && $('#endAddrLatitude').val() != '' &&
        $('#endAddrLongitude') && $('#endAddrLongitude').val() != '') {
        var start = new google.maps.LatLng($('#addrLatitude').val(), $('#addrLongitude').val());
        var end = new google.maps.LatLng($('#endAddrLatitude').val(), $('#endAddrLongitude').val());
    }
}

```

```

        displayRoute(start, end);
    }
}
function displayRoute(startLoc, endLoc) {
    var directionsService = new google.maps.DirectionsService();
    var directionsDisplay = new google.maps.DirectionsRenderer();// also, constructor can get
"DirectionsRendererOptions" object
    directionsDisplay.setMap(map); // map should be already initialized.
    var request = {
        origin : startLoc,
        destination : endLoc,
        travelMode : google.maps.TravelMode.DRIVING
    };
    directionsService.route(request, function(response, status) {
        if (status == google.maps.DirectionsStatus.OK) {
            directionsDisplay.setDirections(response);
        }
    });
}
}

</script>

]]></ac:plain-text-body>
</ac:structured-macro>
<ac:structured-macro ac:macro-id="930f8839-77cc-45a9-85f1-523c58a58ff1" ac:name="confiorm" ac:schema-version="
1">
    <ac:parameter ac:name="formName">f1</ac:parameter>
    <ac:rich-text-body>
        <ac:structured-macro ac:macro-id="7724a15a-839b-4c22-9d2d-0e44351c306e" ac:name="confiorm-entry-register"
ac:schema-version="1">
            <ac:rich-text-body>
                <p> </p>
            </ac:rich-text-body>
        </ac:structured-macro>
        <p>
            <ac:structured-macro ac:macro-id="acf36c67-fde7-4ef5-82d2-facb9bf9e6bd" ac:name="confiorm-field-
definition" ac:schema-version="1">
                <ac:parameter ac:name="fieldName">lat</ac:parameter>
                <ac:parameter ac:name="fieldLabel">Latitude</ac:parameter>
                <ac:parameter ac:name="type">text</ac:parameter>
                <ac:parameter ac:name="required">>true</ac:parameter>
            </ac:structured-macro>
        </p>
        <p>
            <ac:structured-macro ac:macro-id="ce927da0-2672-458d-93d3-577dbb18cbca" ac:name="confiorm-field-
definition" ac:schema-version="1">
                <ac:parameter ac:name="fieldName">long</ac:parameter>
                <ac:parameter ac:name="fieldLabel">Longitude</ac:parameter>
                <ac:parameter ac:name="type">text</ac:parameter>
                <ac:parameter ac:name="required">>true</ac:parameter>
            </ac:structured-macro>
        </p>
        <p>
            <ac:structured-macro ac:macro-id="c0ad1438-2866-4de4-9013-0807e1d8a79a" ac:name="confiorm-field-
definition" ac:schema-version="1">
                <ac:parameter ac:name="fieldName">endAddress</ac:parameter>
                <ac:parameter ac:name="fieldLabel">Delivery Address</ac:parameter>
                <ac:parameter ac:name="type">text</ac:parameter>
                <ac:parameter ac:name="required">>true</ac:parameter>
            </ac:structured-macro>
        </p>
    </ac:rich-text-body>
</ac:structured-macro>
<p>Showing on the map (2 markers)</p>
<ac:structured-macro ac:macro-id="54e6f539-371d-445b-8a68-388ca746c378" ac:name="confiorm-card" ac:schema-
version="1">
    <ac:parameter ac:name="formName">f1</ac:parameter>
    <ac:rich-text-body>
        <p>
            <ac:structured-macro ac:macro-id="d16ab35d-172a-430d-9824-995d1c8a6c8b" ac:name="confiorm-field" ac:
schema-version="1">

```

```

        <ac:parameter ac:name="fieldName">lat</ac:parameter>
    </ac:structured-macro>
</p>
<p>
    <ac:structured-macro ac:macro-id="e1bbf1ca-095e-4db8-9588-c8c5648f475e" ac:name="conform-field" ac:
schema-version="1">
        <ac:parameter ac:name="fieldName">long</ac:parameter>
    </ac:structured-macro>
</p>
<p>
    <ac:structured-macro ac:macro-id="9f186681-2e60-43b4-8c42-2b888d9679b6" ac:name="conform-field" ac:
schema-version="1">
        <ac:parameter ac:name="fieldName">endAddress</ac:parameter>
    </ac:structured-macro>
</p>
</ac:rich-text-body>
</ac:structured-macro>
<p>One marker is added with lat/long and another marker added using the address (which is then decoded to lat
/long and shown)<ac:structured-macro ac:macro-id="6787b68b-d525-4ca5-b46b-b1e014674b07" ac:name="conform-
plain" ac:schema-version="1">
    <ac:parameter ac:name="formName">f1</ac:parameter>
    <ac:parameter ac:name="atlassian-macro-output-type">INLINE</ac:parameter>
    <ac:plain-text-body><![CDATA[<script>
        showLocationOnMapDiv(new google.maps.LatLng([entry.lat], [entry.long], true));
        addEndLocationMarker(' [entry.endAddress] ');
    </script>]]></ac:plain-text-body>
</ac:structured-macro>
</p>

```