Visulate Requirements

– Mobile Browser UI

Peter Goldthorp, April 2012

# Overview

Visulate (http://visulate.com/) is a browser based application that allows users to find and view real estate properties.  It is optimized for the iPad and PCs with a screen resolution of 1024x768 or greater.  We need to develop a version for smart phones with a 320x480 screen resolution.

# Requirements

Analysis of the Google Analytics data for Visulate showed the site received 133,956 unique visits for the period March 28 to April 27, 2012. 17,875 of these were from mobile devices. 6,140 of these were from iPad, 6,000 were iPhone and 5,100 were Android devices.

The current screen layout is optimized for the iPad. Need to provide support for:

1. iPhone and Android devices with320x480 screen resolution

## Functional Requirements

Visulate provides facilities for property management and property research. The property management screens are a series of data entry screens to record information about tenants and tenancy agreements. The property research facilities are interactive reports that allow users to view information on properties and companies in Florida.

Google has indexed over 1 million of the property research pages on the Visulate site. Most of the traffic that Visulate serves is for property research pages.

# Use Cases

1. Google Search Result
   1. Customer searches for an address or company in Central Florida
   2. Google displays Visulate pages in the search result. Examples:
      1. <http://visulate.com/rental/visulate_search.php?CORP_ID=M09000004032>
      2. <http://visulate.com/property/70444>
   3. Visulate displays details of nearby properties for sale or rent/and or Google advertisement.
2. Latest Listings
   1. Customer visits the Visulate home page - <http://visulate.com/>
   2. Visulate displays an html table with photographs of properties that have been recently listed for sale.
   3. Customer clicks on one of the pictures
   4. Visulate displays details of the property and other properties that are available for sale nearby. Example:
      1. http://visulate.com/rental/visulate\_search.php?REPORT\_CODE=PROPERTY\_DETAILS&PROP\_ID=175016
   5. Customer clicks on the “Contact Sue” links for additional details. This results in a sales lead.
3. Drill-down Listing Search
   1. Customer clicks on the “Homes for Sale” image
   2. Visulate shows a page with a breakdown by county of properties for sale as a list and also as a map:
      1. <http://visulate.com/rental/visulate_search.php?REPORT_CODE=LISTINGS&qtype=RESIDENTIAL&state=FL>
   3. They click on one of the counties. Visulate displays an html table with a breakdown of properties for sale by zipcode and price:
      1. <http://visulate.com/rental/visulate_search.php?REPORT_CODE=LISTINGS&qtype=RESIDENTIAL&state=FL&county=ORANGE>
   4. Customer clicks on their desired location and price. Visulate displays an html table with photos of properties that meet the search criteria. It also shows the zipcode/price table to allow them to change their search. Example:
      1. <http://visulate.com/rental/visulate_search.php?REPORT_CODE=LISTINGS&qtype=RESIDENTIAL&state=FL&county=ORANGE&ZCODE=32712&MAX=B_MEDIAN>
   5. Customer clicks on a property in the table. Visulate shows the property details along with links to all the other properties from the search.
4. Geo Location Based Search (note: this is currently a work in progress. The AJAX call sometimes fails)
   1. Customer requests a map based search (http://visulate.com/rental/html/geo\_search.html)
   2. Visulate makes an html5 call to get their current location
   3. It plots this on a map then makes an AJAX call to find details of nearby properties and listings.
   4. It plots these on the map as clickable map pins
   5. Customer clicks on a pin and is taken to a property page.

## Current Behavior

Visulate has a 3 level menu structure. The full screen version displays these as tabs, sub-tabs and context menus. Menus are presented as unordered lists <ul> and formatted using CSS. Each menu is contained within a <div> element. Active menu selections are identified by setting their class to “active”. Example:

<div id="l1\_tab">

<ul>

<li><a href="#" class="active">Property</a></li>

<li><a href="2-1.html">Tenant</a></li>

<li><a href="3-1.html">Payment</a></li>

</ul>

</div>

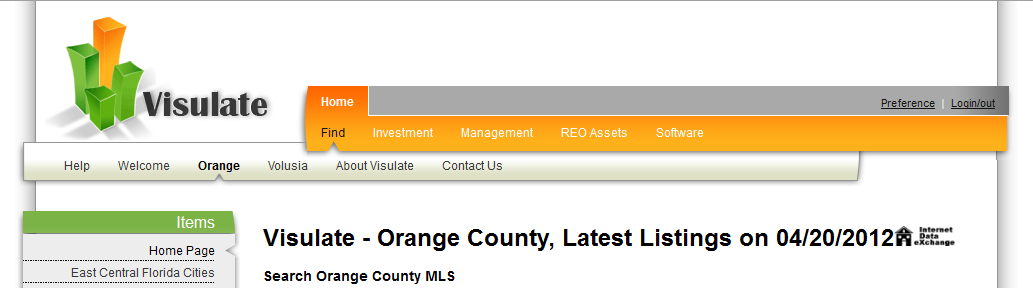
CSS is used to divide the page into 3 regions:

* Logos and tabs (Level 1 and 2 menus)
* Context Menu (Level 3 menus)
* Data Area

The full screen version displays logos and tabs at the top of the screen. The context menu appears under this on the left of the screen. The data area appears under the menu on the right of the screen. Most of the layout is achieved using CSS. Some elements are arranged in html tables. Layout tables are assigned to the "layouttable" class. Tables that display data are assigned to the "datatable" class. The stylesheets it uses to do this are:

* http://visulate.com/rental/css/vis2008layout.css
* <http://visulate.com/rental/css/vis2008tables.css>

Some pages are broken down into sub-pages. The full screen pages display these under the L2 menu. The home page shows an example. It has sub-pages titled Welcome, Orange, Volusia, About Visulate and Contact Us

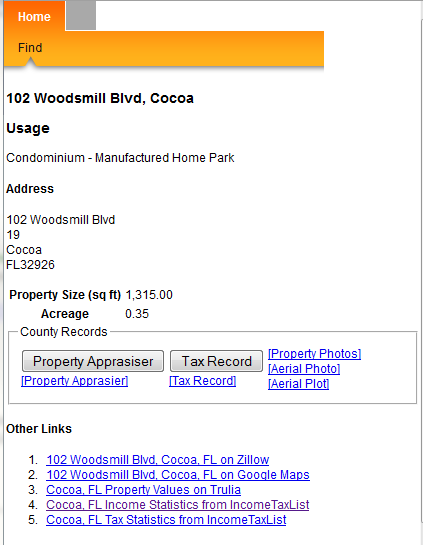


## Mobile Framework

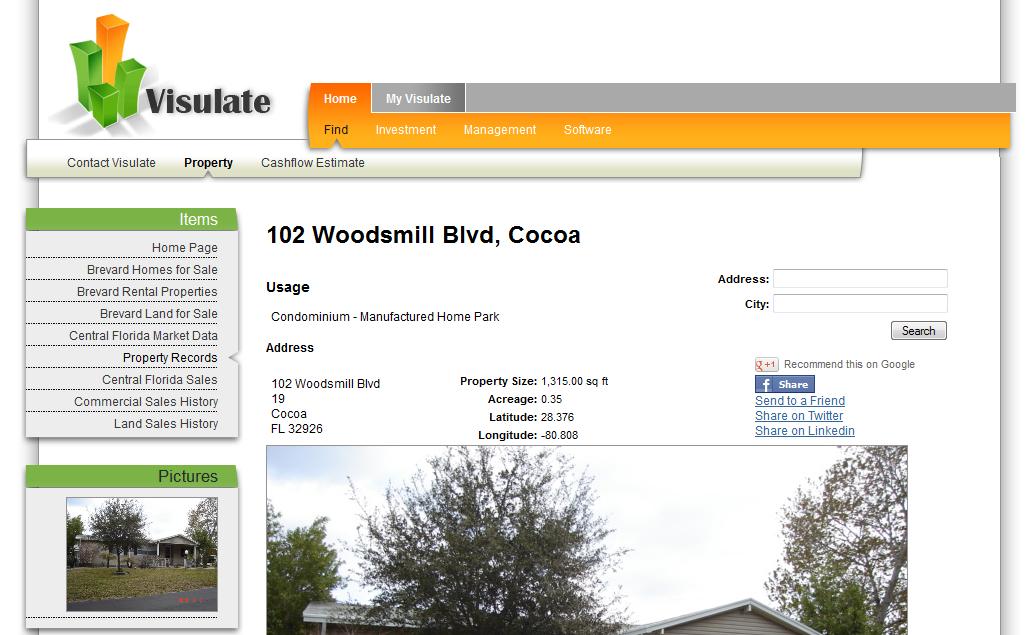
Visulate employs a MVC architecture. Smarty templates are used to control the UI. The PHP code that processes requests includes a function that can identify mobile devices and use a mobile version of the smarty template. This means the mobile screens can be implemented without affecting the full screen templates. The code to do this is already available. We developed a version of the templates for Blackberry phones a few years ago. We don’t use them at the moment but the templates are still available. It had a very basic stylesheet:

* http://visulate.com/rental/css/ mobile-layout.css

Here’s an example of the layout it produced:



Here’s a full screen version of the same page:



# Design

We need a design that can handle the use cases described above. jQuery Mobile looks like it might be worth investigating: http://jquerymobile.com/