

# Homework 8: Responsive Web Design

Stock Search using Markit on Demand & Facebook Mashup  
(Bootstrap/JQuery/AJAX/JSON/Cloud Exercise)

Total Points: 10

## Basic Search Form (1 point in total)

- The initial webpage with all form fields as described. Clear Button, Search Button and the Markit on demand logo with a link to the Markit homepage **(0.25 points)**
- The initial carousel with the default favorite list and all the three features: Auto Refresh, Refresh and Display Stock Information (disabled initially) **(0.25 points)**
- HTML5 and JQuery validation for the search input **(0.5 points)** both use cases given in the homework description.

## Design (0.5 point in total)

- All the components (text, font, borders, tooltips and bootstrap components) were properly placed. – **0.5 points**

## Result Display (8.5 points in total)

### Favorite List (1 point in total)

- The table with all valid data and the styles. – **0.25 point**
- Automatic Refresh should refresh data every 5 seconds. – **0.5 point**
- Refresh, navigate to stock information should work as per the description. – **0.25 points**

### Current Stock (4.5 points in total)

- The table with the current stock information – **1 point**
- The layout and style should be mimicked as close as possible and responsive. – **0.75 point**
- The Yahoo chart with correct values and size should be placed properly in the tab – **1 point**
- Favorite button – **0.5 point**
  - Should add or remove favorite based on the description, changing the CSS of the button (color of the star) – **0.25 points**
  - Should also refresh the entry in the favorite list – **0.25 points**
- Facebook button – **1.25 point**
  - Authorize the user to facebook if the user is not already logged in. - **0.25 points**
  - Post the Stock Information to a user's News Feed – **1 point**

- Display the correct message as Company Name and Price. Together with this you need to give the description which should include the price and change. – **0.25 points**
- Stock image from Yahoo charts – **0.25 points**
- Working link to Markit on Demand – **0.25 points**
- Successful/Failed Message after posting. – **0.25 points**

### Historical Charts (2 point in total)

- Should display the proper chart with all the needed buttons
- No extra buttons and features should be visible other than what was depicted in the description.

### News Feed (1 point in total)

- All layout and style should be mimicked as close as possible and responsive – **0.5 points**
- All the data should be properly loaded. – **0.5 points**

### Use of GAE/AWS

- PHP script must be deployed on GAE/AWS. Implementing the assignment by using the Apache + PHP on **cs-server.usc.edu** instead of GAE/AWS will result in a **penalty of 4 points**. To allow the graders to perform this test, an additional link should be added to the list of homework, with a format similar to the following:
- [http://example.appspot.com/?\[listing\\_the\\_parameters\\_and\\_sample\\_values\]](http://example.appspot.com/?[listing_the_parameters_and_sample_values])  
Or  
[http://example.elasticbeanstalk.com/?\[listing\\_the\\_parameters\\_and\\_sample\\_values\]](http://example.elasticbeanstalk.com/?[listing_the_parameters_and_sample_values])

This link would guide graders to a page where it displays the JSON data wrapped in your PHP file running on the Google App Engine/Amazon AWS.

- Implementing API access to MarkitOnDemand using JavaScript instead of PHP (cross-domain using the “hack”, JSONP or CORS) will result in a **penalty of 5 points**.

### Additional Requirements

- The program must work in Firefox and Chrome
- All the results should be displayed in the same window as the search form.
- The page must be responsive. If not you will have a penalty
- The window should not reload for any kind of data request. All transactions are asynchronous.