**Assignment Instructions**

Homework 08

Read this entire homework description once through before proceeding.

**Required Reading**

* AngularJS
  + Tutorial 1 Video: [http://youtu.be/uFTFsKmkQnQ](http://youtu.be/uFTFsKmkQnQ" \t "_blank)
  + Tutorial 2 Video: [http://youtu.be/WuiHuZq\_cg4](http://youtu.be/WuiHuZq_cg4" \t "_blank)
  + Read steps 1, 2, 4-6 from the Tutorial: [http://docs.angularjs.org/tutorial](http://docs.angularjs.org/tutorial" \t "_blank)
* Homework 06 Solution: [https://c9.io/ajmudrakru/twitter-search-demo-ajax/](https://c9.io/ajmudrakru/twitter-search-demo-ajax/" \t "_blank)

**Task**

Use the Homework 6 solution, or your own code for Homework 6 if you prefer, to produce a version of the Twitter Search that uses AngularJS instead of jQuery.

In particular, you should create AngularJS templates for areas of the page that change and display results and history, instead of using jQuery append, appendTo, show/hide, etc.

Ensure that the previous functionality remains intact, as expected in Homework 6.

**Restrictions**

You may use jQuery’s $.ajax, but you SHOULD NOT USE ANY other jQuery functionality.

**API Information**

**How to Use twitter.php**

The twitter.php file is meant to act as a server-side facilitator to access the Twitter REST API while avoiding the cross domain access policy in browsers.  Since twitter.php will be running on the same server as your project, you will be able to make AJAX calls to it, instead of Twitter directly.  Plus, it also hides your consumer key and secret.

The “twitter.php” file is an alias for the URL “https://api.twitter.com/1.1”. Therefore, any path and query string you put after the “twitter.php” file in your AJAX call will be passed through.

For example:

/twitter.php/search/tweets.json

is equivalent to:

https://api.twitter.com/1.1/search/tweets.json

**Twitter API itself:**

See the Twitter API documentation at: [https://dev.twitter.com/docs/api/1.1](https://dev.twitter.com/docs/api/1.1" \t "_blank)

**Handing in Your Work**

* Provide, in the Sakai assignment submission area, a URL to your Cloud9 workspace containing the homework content.

**Grading Criteria**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Beginning** | **Developing** | **Accomplished** | **Exemplary** | **Total** |
| Functionality | 3 - 3.4  Lacking much of the critical functionality.  Errors are common and may cause features to fail or appear unimplemented.  Functionality is not apparent to the user when performing behaviors. | 3.5 - 4.4  Some important functions work as described.  Behavior-affecting errors may be present.  Some functionality may seem vague to the user when actions are performed. | 4.5 - 5.4  Critical functions work described.  Some functionality may produce minor errors that do not significantly impact behavior..  Some behaviors might be slightly unclear when performed by the user. | 5.5 - 6  All functionality is as described.  Behaviors make intuitive sense when performed by the user.  Each feature is implemented within the described restrictions. | 6 |
| Design / Usability | 1  Design lacks neatness, appropriate alignment, or contains a significant number of color choices that reduce usability.  Elements are generally lacking any typical behavioral cues. | 1.1-1.4  Design is somewhat neat, has poor in many places alignments, and may have poor color choices that reduce readability or usability.  Many elements may lack typical behavioral cues. | 1.5 - 1.7  Design is mostly neat, but may have poor alignments.  Most elements have typical behavioral cues.  Most critical elements fit within a normal page size. | 1.8 - 2  Design is clean and concise.  Elements used are all clearly marked for behavioral cues (e.g. buttons or links to perform actions, dropdown cues).  All elements are laid out neatly within in a normal page size. | 2 |
| Robustness | 1  Lacks appropriate form validation, allowing errors to be submitted with the form.  Error messages may not be apparent to the user. | 1.1-1.4  Some form functionality is properly validated, and thus some validation errors may be missed and allow invalid submissions.  Most error messages may be repetitive or inconcise. | 1.5 - 1.7  Most critical form functionality is properly validated and do not allow submission.  Most error messages are clear and indicate the issue to the user. | 1.8 - 2  Forms are fully and properly validated and do not allow any data submissions with errors.  Error messages are reported clearly and concisely to the user, and with no severe interruption to user flow. | 2 |