

Object Oriented Design
IS314 -- Exercise 1 – Fall 2011
Due October 14, 2011

Exercise 1 Overview: Organizations are becoming increasingly dependent on a savvy public where people are able to build value into systems through processes of secondary design. Without the contributions of authors and editors Wikipedia is simply a technology. Flickr is little more than a website without people posting and tagging images. Such online communities rely on the loyalty and cognitive surplus of a community that shares their time, energy, and expertise in secondary design and development processes in the creation of value for an organization.

Exercise 1 Requirements: Build an interface that allows users to submit and rank articles. This type of system is exemplified in online communities like Digg and Reddit (I am not responsible for the content of Digg or Reddit. I am simply using them as an example of functionality, not content). An obvious application of this process at UWEC is on the front page. Pretending that the issues of politics, control, and general uncertainty were overcome at UWEC, we could move forward with the design and development of such a system. There are definitely some limitations with this assignment, largely stemming from the fact that we are not using a database yet. However, following an agile approach, we are going to build a working prototype of the system that will mock-up the system so that potential people using the system can comment. For assignment 1, the system must contain the following:

- 1) A login and password form that will allow people to enter the system
 - a. The login and password checking will be done using the IST data hash class library. This will be provided. We will only create one login and password but it will be stored using an MD5 hash directly in the code.
 - b. We will need a 'create and verify' method for creating and verifying the user.
 - c. We will also create a user class/object that will store relevant information of the user.
 - d. Users will only be able to attempt three logins before the system kicks them out.

- 2) A form that will allow people to view top posted articles
 - a. For now, this will only show three articles. The articles will have the following characteristics:
 - i. Link to the html of the actual article
 - ii. Time the article was submitted
 - iii. Category to which the article was submitted
 - iv. The count of 'votes' that article has received (+ or -)
 - v. All of the information (i.e. properties) will be retrieved from an 'article class/object'
 - b. This form will also allow people to 'digg' or vote-up articles to the top of the page.
 - i. If a user clicks the up or down arrow, the arrows should become hidden and count total should increase by one.
 - ii. Clicking the up or down arrows will have to update the articles property as well.
 - c. The form will also include a 'newly submitted' section. This will behave similarly to the 'tops' section with a newly submitted class/object.
 - i. It will have its own class/object and the properties of the submitted article.
 - d. For this assignment, I don't see the top and new sections being terribly different. We will have differences once we bring the database into the picture and start ranking multiple articles.

3) A form that will allow people to submit articles

- a. For assignment 1, this form will override the properties of the newly submitted class/object. I do this so that I can return to the view form to see the newly submitted article.

Grading notes for this assignment include the following:

- An overall clean submission that I'm able to work with without getting frustrated (1 point)
 - Building the appropriate interfaces/forms as described above (1 point)
 - Using the IST data hash library for handling the user login component of the system (1.5 points)
 - Writing your classes to a class library and then subsequently consuming your library/classes (2 points)
 - Using a fellow classmates class library for doing the same system behaviors (1 point)
 - Following file, folder, variable, and constants naming conventions (1 point)
 - Code Documentation (.5 points)
 - Listing with your comments, where the following occurred: instantiation and inheritance (1 point)
 - Including a comment on your experiences and difficulties and one tech tip (1 point)
 - Submitting complete project and avoiding late deduction (10% per day)
-