Prototyping integration solutions using mashup tools

Assignment 5: Due before class starts, April 7

Learning objectives:

- Explore Mashup tools and simple data integration problems using Yahoo Pipes (http://pipes.yahoo.com/pipes/) and a prototype application dealing with reading libraries.
- Know how to register and setup developer access to public Web services.
- Know how to setup a data file from a remote Web server.
- Demonstrate a working understanding of simple APIs and the XML format of data files and RSS results.
- Demonstrate proficiency with a mashup tool (Yahoo pipes).
- Apply design principles for the pipes-and-filters integration design pattern and learn its limitations.

Instructions:

Part of this exercise is to learn how to identify, research, and keep a record of resources required when developing an integration prototype and testing results. Keep notes as you go along and include details in your written results. How well you do document your work will have significant impact on your grade. The notes should be sufficient for someone else to replicate your results.

1. [25 points] Create an inventory for your (fictitious) personal library.
    a. Using Google Books create a bookshelf that includes all the books in your library.
    b. Your library should include at least 25 books. Ten (no more, no less) of your books should be from the New York Times hardback fiction bestseller list as of April 1, 2014.
    c. Export your bookshelf from Google Books as XML.
    d. Save it on your personal web site on http://www.personal.utulsa.edu.
    e. Submit a write-up of your efforts with
        i. The URL for your bookshelf on Google Books.
        ii. The URL for your XML file on http://www.personal.utulsa.edu.
        iii. A screenshot from Internet Explorer of the top of your XML file.
        iv. A detailed and complete checklist (sufficient for someone else to duplicate what you did) of all steps necessary to setup your catalog. This should be a bulleted list. Include any instructions on setting up your personal web site.

2. [25 points] Create a mashup application on Yahoo Pipes that allows a user to search your library catalog.
    a. Implement a simple search that allows the user to enter an author and/or title to search for books in your catalog. Return results on the Yahoo Pipes interface where the user can download the results.
    b. Submit
        i. The URL of your pipe on Yahoo Pipes.
        ii. Instructions on how to access and test your pipe. This should be a bulleted list.
        iii. Screen shots showing your search working (show 2 different queries with results).

3. [50 points] Create a mashup application on Yahoo Pipes that shows current (as of April 1, 2014) hardback fiction bestsellers (according to the New York Times) available in your library's inventory. Offer the list as an RSS feed.
    a. Submit
        i. The URL of your published pipe on Yahoo Pipes. Make sure to use Yahoo Pipes' features to keep your TU site private.
        ii. The URL of your RSS feed result.
        iii. **Complete instructions** on how to access and test your pipe.
        iv. Screen shots your pipe working (show the RSS feed in the Internet Explorer browser).
        v. A detailed and complete checklist (sufficient for someone else to duplicate what you did from start to finish) of all steps necessary to setup your mashup. This should be a bulleted list. This is NOT to focus on your design, but instead on the administrative steps required to set up accounts, etc.

4. [5 points extra credit] Modify your results in problem 3 to also allow the user to input any date for the bestseller matching search before your pipe starts.

5. [30 points super bonus extra credit] Implement problem 3 using Huginn (https://github.com/cantino/huginn) instead of Yahoo Pipes (note this will require you to install Huginn on your own machine or one in the classroom). Demonstrate the implementation in class or show why it cannot be done with Huginn. You may for this problem ONLY join with a team and divide the extra credit points among the team members (for example, 3 team members would get 10 points each or if the team determined that there should be some other division, some fraction of 30 each that totaled 30.)