CS 407, Spring 2012: iOS Assignment

iOS Programming Assignment: Designing a new Reader

Due: Feburary 20, 2012 (Monday) at 5pm. Submission instructions will be announced later.

The goal of this project is to build a simple reader from a RSS feed or a JSON API. You will be required to list the contents provided by the feed/API using a table view, and when a user taps on a given entry, you must then present a new view that shows more detailed information about the object.

Example Data Sources:

- wisc.edu news feed: http://www.news.wisc.edu/feeds/list/4 (table view: list news entries; detailed view: load a web page with news article)
- Twitter public timeline: https://dev.twitter.com/docs/api/1/get/statuses/public_timeline (table view: list public tweets; detailed view: show tweet, time, users who retweeted, etc)
- NY Times' Bestseller List (requires API key): http://developer.nytimes.com/docs/read/best_sellers_api (table view: list top books; detailed view: show picture of book, number of weeks featured, etc)
- Rotten Tomatoes' Movie Ratings: http://developer.rottentomatoes.com/ (table view: list top movies; detailed view: show movie poster, average rating, etc)
- Any blog (engadget, gizmodo, etc)

You don't have to use these exact examples, but find an appropriate data source where you can feature a basic list, pass information about a given entry (i.e., via a class you create), and show more detailed information in the next view.

Resources:

- NSJSONSerialization Class Reference: https://developer.apple.com/library/ios/#documentation/Foundation/Reference/NSJSONSerialization_Class/Reference/Reference.html
- NSXMLParser Class Reference: https://developer.apple.com/library/ios/#documentation/Cocoa/Reference/Foundation/Classes/NSXMLParser_Class/Reference/Reference.html
- UIWebView Class Reference: https://developer.apple.com/library/ios/#documentation/UIKit/Reference/UIWebView_Class/Reference/Reference.html

Hints:

- Use a navigation controller to facilitate transitions between views. Use the sample project you created in the lab as a template.
- RSSParser usage:

```
RSSParser *rssParser = [[RSSParser alloc]
initWithRSSFeed:@"http://www.engadget.com/rss.xml"];
```

  This will return nil if it is unable to download the file (i.e., no network connection). You can then access an array of entries by calling rssParser.articleList. If articleList is nil, then there was an error parsing the feed. If you plan on using the pubDate property, make sure it isn't nil first. Most RSS feeds conform to one date format. If you find that your feed is different, you must modify the library code. A message will be logged to the console if the date value for that article is set to nil.
- The Article class gives a general framework on how to store your parsed data. You must modify the RSSParser class if you do not use the Article class.

Bonus:

- Incorporate the user's location into the results displayed (i.e., show nearby tweets if using Twitter's feed).
- Allow the user to email a link to what the user is currently viewing.
- Feeds that have articles from multiple days can be grouped into multiple sections and have the date as the header for each section.