CMSC436: Fall 2013 – Week 8 Lab (Part I)

Objectives:

Familiarize yourself with Alarms and Networking. Create two applications using Alarms and Networking.

Once you've completed this lab you should have a better understanding of Alarm and Networking. You should know how to use and create alarms using the AlarmManager Service, and how to use Networking support classes to send HTTP POST/GET requests to other services on the Internet.

Overview:

In total, this lab has two parts. We are handing part 1 today, and will hand out part 2 at our next class meeting. This part of the lab focuses primarily on Networking. The next part will also focus on Alarms.

Part 1: Networking

In this part, you will modify the Threads/AsyncTask and Handlers lab to use networking support classes to obtain live stock quote values and historical information from Yahoo! Finance. The URL for this service is http://finance.yahoo.com/d/quotes.csv

Documentation for this service can be found at the following URLs:

http://www.jarloo.com/yahoo_finance/

To use the API you create a URL that specifies: 1) the companies about which you want data, and 2) the kinds of data you want.

The return values are formatted as comma-separated values (CSV) files, containing the values you requested. You'll need to parse that data so that it can be displayed.

You will be given a skeleton that provides nearly all the code you need for this application. You will need to modify the skeleton code however in at least the following ways.

The current skeleton requests only one value - the current realtime ask price. Modify the skeleton so that it requests not only the current realtime ask price, but also the 52-week high, and the 52-week low prices as well. To create the right URL, you will need to look up the codes for these data in the documentation listed above. Once you receive this data, you will also need to modify the data structures, user interface, and saving and loading routines to work with these new data values.

On each data refresh, the skeleton code currently requests data for one company at a time. That is, if the user has entered five stock symbols, then the application currently makes five separate network requests for data; one for each company. This is slow and unnecessary because the API allows you to request data for multiple companies at one time. You must modify the code so that it requests the data for all five companies at one time.

Implementation Notes:

1) Download the application skeleton files from the Lectures & Labs web page and import them into your IDE.

2) Modify the code so that it requests the realtime ask price, the 52-week low and the 52-week high.

3) Modify the QuoteItem data structure so that it stores and retrieves all three pieces of data.

4) Modify the ListView layout and QuoteListAdapter so that all three pieces of data are displayed in the ListView.

5) Modify the code so that when the data arrives, your application parses it to extract all three values.

6) Modify the code so that it requests the data for all five companies all at one time. There are different ways to achieve this, but you might try to:
    a. Modify getQuote() and doInBackground() to work with multiple companies.
    b. Write some helper functions to parse data for a specific company from the data of many companies.
7) Modify the code so that the values of the QuoteItems are correctly saved in onPause().