ITIS/ITCS 4180/5180 Mobile Application Development
In Class Assignment 7

Basic Instructions:

1. In every file submitted you MUST place the following comments:
   a. Assignment #.
   b. File Name.
   c. Full name of all students in your group.
2. Each group should submit only one assignment. Only the group leader is supposed to submit the assignment on behalf of all the other group members.
3. Please download the support files provided with this assignment and use them when implementing your project.
4. Export your project as follows:
   a. From eclipse, choose *“Export…”* from the File menu.
   b. From the Export window, choose *General* then *File System*. Click *Next*.
   c. Make sure that your project for this assignment is selected. Make sure that all of its subfolders are also selected.
   d. Choose the location you want to save the exported project directory to. For example, your *Desktop* or *Documents* folder.
   e. When exporting make sure you select *Create directory structure for files*.
   f. Click Finish, and then go to the directory you exported the project to. Make sure the exported directory contains all necessary files, such as the .java and resource files.
5. Submission details:
   a. When you submit the assignment, compress your exported project into a single zip file. The format of compressed file name is InClassAssignment#.zip
   b. You should submit the assignment through Moodle: Submit the zip file.
6. **Failure to follow the above instructions will result in point deductions.**

In Class Assignment 7 (100 Points)

In this assignment you will develop “BBC News App”. The App displays news feeds from BBC in different categories and lets you manage (add, update, delete) news items to a History list. In this assignment, you will parse XML news feeds using threads, display them in ListViews and learn to add, retrieve, update and delete items to a SQLite database.

Important App Requirements:

1. The required Android Virtual Device (AVD) should have **minimum SDK version set to 14 and target SDK at least 19**. The app should display correctly on 3.2” QVGA (ADP2) (320x480: mdpi). **Your assignment will not be graded if it does not meet these requirements, and you will not be granted any points on your submission.**
2. All strings should be read from your strings.xml, all dimensions from dimens.xml, and all images from drawable-ldpi. The string values used for the text labels, and button labels should be read from the `strings.xml` file and should not be hardwired in the layout file.
3. All image downloading should be performed using Threads (or AsyncTask) and your code should not block the main thread.

Part A: Main Activity (5 Points)

The launching activity is the Main Activity. Figure 1, shows the UI for the Main Activity. The requirements are as follows:

1. The activity displays a list of the news categories provided by BBC News feed website at http://www.bbc.com/news/10628494, along with a “Reading List” category.
2. The categories that should be included in the list are Top Stories, World, UK, Business, Politics, Health, Education & Family, Science & Environment, Technology, Entrainment & Arts, and My History. The list should be created using a simple ListView UI component.
3. Clicking on any category should start the “**NewsActivity**”, passing it only the url of the selected category. If the “My History” is selected pass a null url.

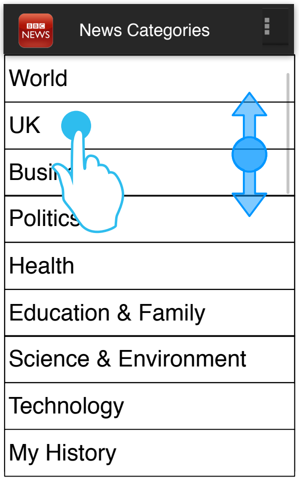


Figure 1, Main Activity

Part B: Loading and Parsing News Items (News Activity) (80 Points)

News Activity should display brief news feed of the selected category. For all the categories except the My History category, the url to the xml rss feed for the selected will be provided by the Main Activity (in the extras). Each news item is included under the item element, which includes the news title, description, and pubDate. In addition, each news item includes both a small and large thumbnail urls. The requirements are as follows:

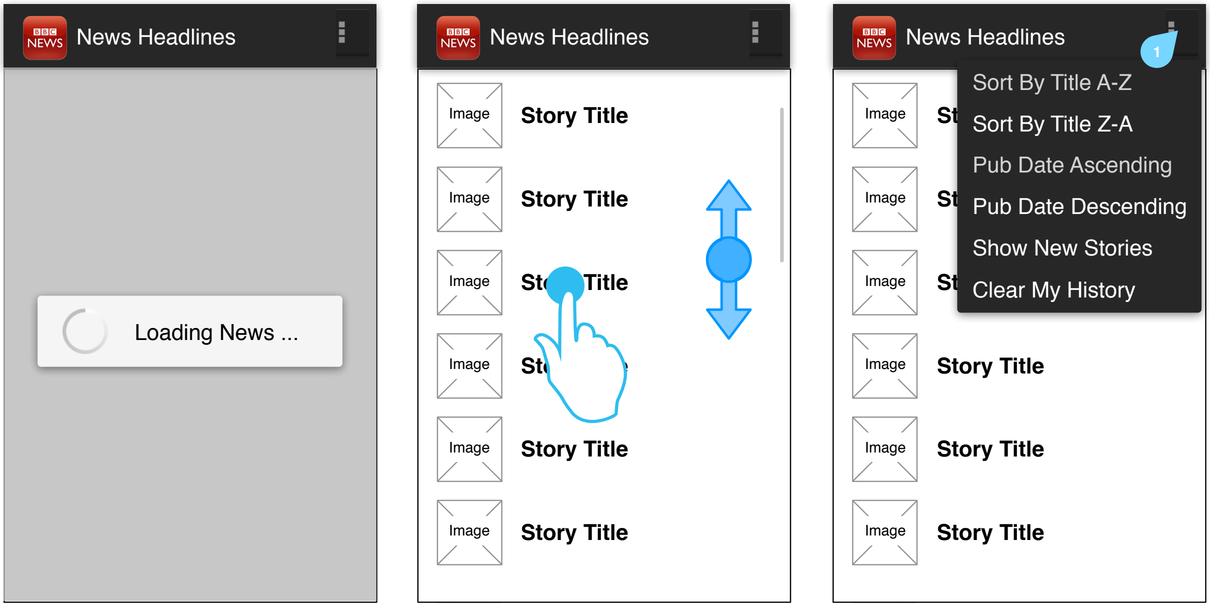
1. Use thread or AsyncTask to establish a connection to BBC server using the url passed by the previous activity. All the XML parsing and network connections should be performed by the child thread.
2. Display a Progress Dialog during the download and/or parsing of the XML and dismiss it once these operations are completed as indicated in Figure 2(a).
3. Parse the XML stream using either a XMLPullParser or SAXParser and return the result to the News List.
4. For each news item retrieve the title, description, pubDate, link, the url for both the small and large images provided.
5. The news items should be displayed in a list, this list should be created using a customized ListView. Each item in the ListView should contain an ImageView and a TextView. See Figure 2(b).
6. The activity should display all the news items retrieved from the XML feed. In the new items list, for each news item display it's small thumbnail image and the news title. See Figure 2(b). A separate thread or AsyncTask should be used to retrieve the news thumbnail. You can use the Picasso library http://square.github.io/picasso/ .
7. The news item row should be clickable to provide the following functionality:
    a) If the news item is not already stored in the SQLite History table, then add the news item to the History table. Otherwise, if the news item is already stored in the SQLite History table, simply increment the view count stored in the SQLite History table for this news item.
    b) Start the **DetailedNewsActivity** which provides more details about the current news item.
8. If the category selected in the Main Activity is "My History", then the news list should be retrieved from the History table and should be presented as shown in Figure 2.
    a) The news items displayed in the ListView should be sorted in descending order based on view count.
    b) The History table should store all the required news information such as the title, link url, thumbnail image url, pub date, large image url, and description. Use the link url as the primary key for the History Table. In addition, the table should maintain the news item view count, which is required as a sorting key.
9. Managing the Menu Items
    a) Clicking the menu item "Sort By Title A-Z" should sort and redisplay the displayed news list in ascending order based on the title.
    b) Clicking the menu item "Sort By Title Z-A" should sort and redisplay the displayed news list in descending order based on the title.
    c) Clicking the menu item "Pub Date Ascending" should sort and redisplay the displayed news list in ascending order based on the pubDate value. You can use the SimpleDateFormat in Android to format the retrieved dates. http://developer.android.com/reference/java/text/SimpleDateFormat.html
    d) Clicking the menu item "Pub Date Descending" should sort and redisplay the displayed news list in descending order based on the pubDate value.

You can use the SimpleDateFormat in Android to format the retrieved dates. http://developer.android.com/reference/java/text/SimpleDateFormat.html

e) Clicking the menu item “Show New Stores” should filter the displayed news items and only display the new items that have not been viewed by the user. In order to perform this task you are required to compare the news items with the news items stored in the History Table. Refresh the list to only show the news stories that have not been viewed by the user.

f) Clicking the menu item “Clear My History” should delete the contents of the History table. If the history table is currently being displayed then the list view should be refreshed to indicate this change.

10. Note: In order to easily view the xml feed, you can download it as an RSS XML file and view it in Firefox browser or you can use the eclipse xml viewer. Or you can use the XML reader at http://codebeautify.org/xmlviewer



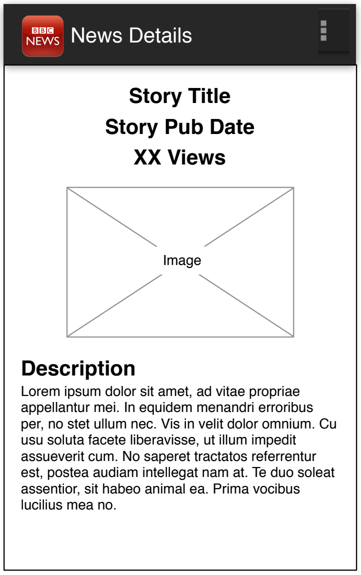
(a) Progress Dialog (b) News List (c) Menu Items

Figure 2, News Activity Wireframe.

Part C: Detailed Activity (15 Points)

Detailed Activity shows details about a selected news item. The user can view more details about the news and then return back to the list. The requirements are as follows:

1. The activity's wireframe is shown in Figure 3. The news details should include the title, publication date, large image, and description. Pressing the back key should return to the News Activity.
2. Notice under the Story Pub Date, the activity should display the number of views that this news item has been viewed by the user.
3. The image should be retrieved using a background thread (Thread or AsyncTask).



Lorem ipsum dolor sit amet, ad vitae propriae appellantur mei. In equidem menandri erroribus per, no stet ullum nec. Vis in velit dolor omnium. Cu usu soluta facete liberavisse, ut illum impedit assueverit cum. No saperet tractatos referrentur est, postea audiam intellegat nam at. Te duo soleat assentior, sit habeo animal ea. Prima vocibus lucilius mea no.

Figure 3, Detailed Activity