ITIS/ITCS 4180/5180 Mobile Application Development
In Class Assignment 5

Basic Instructions:

1. In every file submitted you MUST place the following comments:
   a) Assignment #.
   b) Full name of all students in your group.
2. This is a group assignment and will follow the below rules:
   a) You will be assigned to a group of 4-5 students.
   b) **Group dynamics:** In each group, only one student is supposed to be sitting in front of a computer and performing the coding exercise. This student will be assigned by the instructor. Other group members should sit beside or behind the coding student and assist the coding student.
   c) Only the coding student is allowed to type, enter, paste code into the project, and the other team members can only assist the student **verbally**. The team members are not allowed to share files with the coding student. All the team members should be actively engaged in helping and guiding the coding student.
   d) The team members could request changing the coding student, however **10 points** will be deducted from the overall team grade for this change.
   e) The instructor could decide on swapping the coding student at anytime during the lecture. So all group members should be actively engaged in assignment.
   f) The team will loose **5 points** for every violation of this group assignment rules.
3. Your assignment will be graded for functional requirements and efficiency of your submitted solution. You will lose points if your code is not efficient, does unnecessary processing or blocks the UI thread.
4. Please download the support files provided with this assignment and use them when implementing your project.
5. Submission details:
   a) Only one member of the group is required to submit the assignment on behalf of all the other group members.
   b) Export your Android project and create a zip file which includes all the project folder and any required libraries.
   c) The file name is very important and should follow the following format: **Group#_InClass05.zip.** You should submit the assignment through Canvas: Submit the zip file.
6. **Failure to follow the above instructions will result in point deductions.**

In Class Assignment 05 (100 Points)

In this assignment we are going to make a Photo Gallery application, which will use a URL that retrieves a text file containing a dictionary of keywords and image URLs related to the associated keyword. The application consists of a single activity that enables the user to download and view online photos. The photos related to the keyword for this app should be retrieved from the below URL:

http://dev.theappsdr.com/apis/photos/index.php?keyword=uncc

The retrieved text contains several lines of pair (keyword, URL). Here, “keyword” is the search keyword. For example, “UNCC” have five URLs associated with it. Other keywords may have more or less. The API returns a semicolon separated text, which includes the keyword and the urls associated with this keyword as follows:

```
UNCC;http://cdn1.theodysseyonline.com/files/
2016/01/09/635879466452203457-1500514842_UNC-Charlotte-043.jpg;https://s-media-cache-
ak0.pinimg.com/736x/48/39/d5/4839d5edc171cf3e11ff9183e624b031.jpg;http://
www.bestvalueschools.com/wp-content/uploads/2014/09/University-of-North-Carolina-
Charlotte-affordable-Online-degree.jpg;https://old.northcarolina.edu/campus_profiles/
img/profiles/uncc/uncc-3.jpg;https://c2.staticflickr.com/
4/3933/15296234239_4f11d889a8_c_d.jpg
```

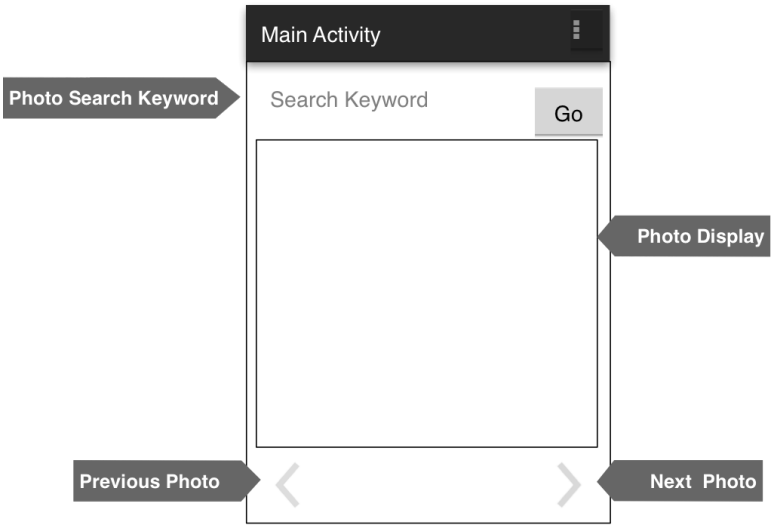
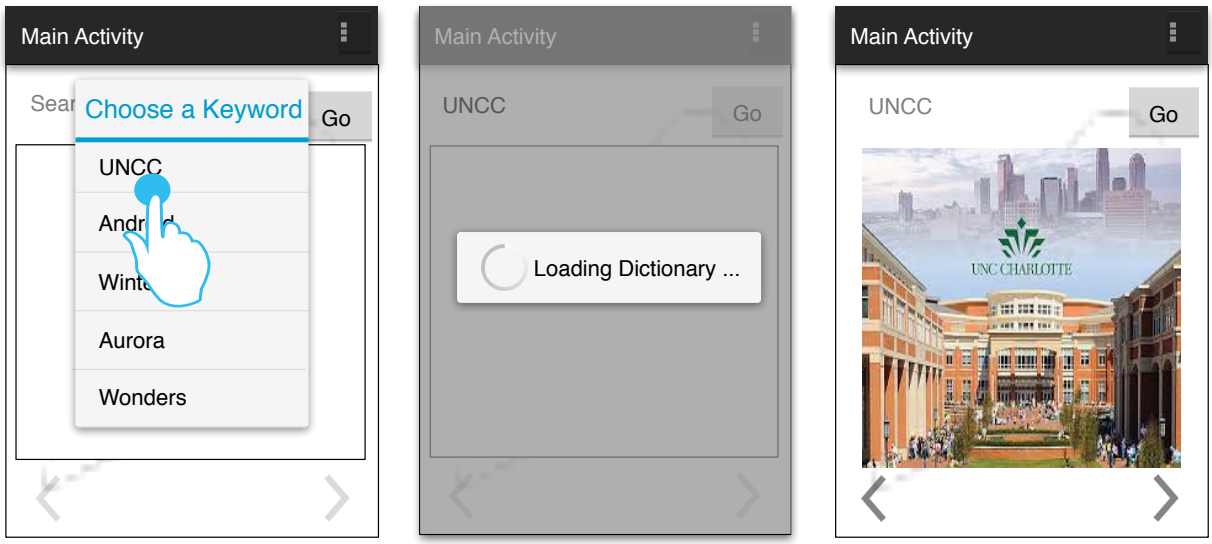
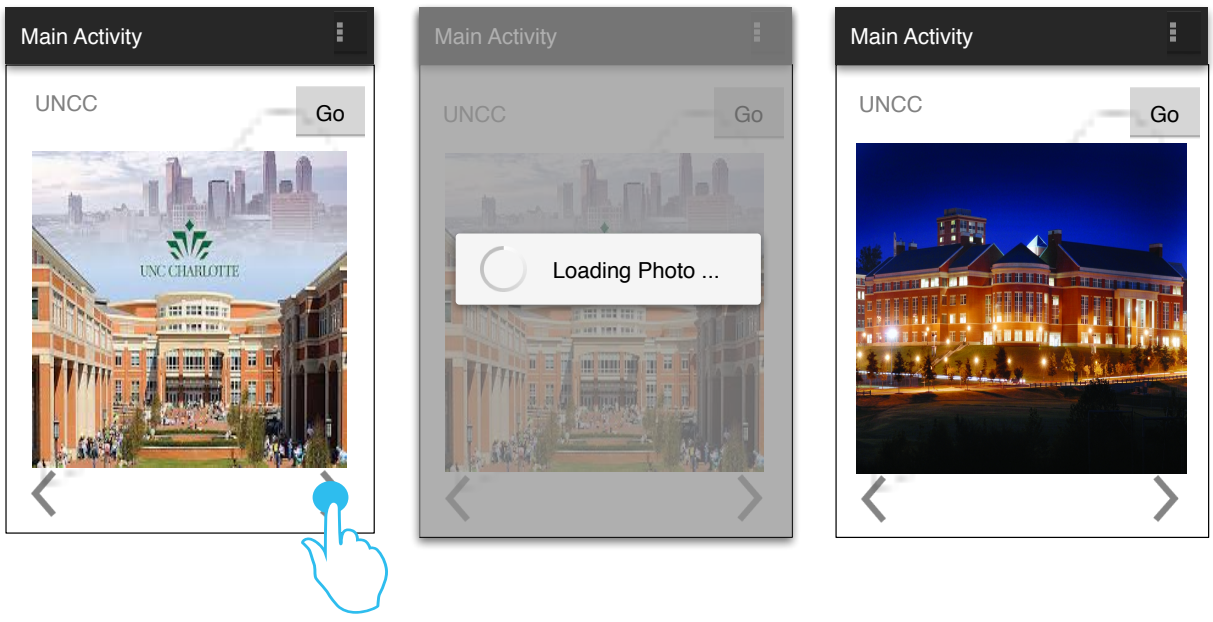


Figure 1, Application Wireframe



(a) Select Keyword (b) Loading Dictionary (c) Retrieve First Image

Figure 2, App Wireframe



(a) Next Image (b) Loading Photo (c) Show Next Photo

Figure 3, App Wireframe

Loading Images based on Keywords

The interface should be created to match the user interface (UI) presented in Figure 1. You will be using layout files, and strings.xml to create the user interface. Perform the following tasks:

1. Clicking on the “Go” Button should display a list of keywords as shown in Figure 2(a). You can either Alert Dialog or Spinner to implement it. The list of keywords are: UNCC, Android, Winter, Aurora, and Wonders.
2. Clicking on a Keyword should do the following:
   a) The TextView will hold the search keyword clicked by the user.
   b) Use AsyncTask/Thread to connect to the provided URL and retrieve the content of the api provided.
3. When the api data is retrieved:
   a) Use another AsyncTask/Thread to retrieve the first image associated with the keyword and display it in the ImageView.
   b) The AsyncTask/Thread class should be in a separate file/class not inner to the main thread. i.e. You should manage passing parameters to the class and then pass back the result image downloaded to the UI.
   c) While the image is being retrieved, you should display a Progress Dialog as indicated in Figure 3(b).
4. The Next and Previous photo icons should be disabled when the application is launched, and enabled after the first photo is displayed. The buttons will also remain disabled in the case there is only 1 image or there are no images corresponding to a keyword. Use icons provided in Support Files for setting the image icons (**next.png, prev.png**)
5. **Do not** store the photos, simply download and display the retrieved photos. Also do not attempt to download all the URLs receive, and your app should only download and display a single photo at any given time.
6. Upon clicking the “Next Photo” icon, you should download the next photo in list of URLs retrieved (following the of order of appearance in the URL list retrieved). You should call the AsyncTask/ Thread used to download the next photo.
   a) If the currently displayed photo happens to be the last photo, you should download and retrieve the photo at index 0 (the first photo) when the Next icon is pressed.
7. Clicking the “Previous Photo” icon, you should download the previous photo. If the currently displayed photo happens to be the first photo, you should download and retrieve the photo at the last index position (last photo).
8. If the api returns an empty list of urls, then clear the currently displayed ImageView and display an error Toast message “No Images Found”.
9. Your application should download the requested photo only if there is an established internet connection. If there is no internet connection you should display a Toast message indicting that there is no internet connection and do not attempt to send the HTTP request.