Homework 9
Android App for eBay Search

1. Objectives

- ➢ Become familiar with Android Studio, Android App development and Facebook SDK for Android.
- ➢ Build a good-looking Android app using the Android SDK.
- ➢ Add social networking features using the Facebook SDK.

2. Background

2.1 Android Studio

Android Studio is the official IDE for Android application development, based on IntelliJ IDEA (https://www.jetbrains.com/idea/). On top of the capabilities you expect from IntelliJ, Android Studio offers:

- Flexible Gradle-based build system
- Build variants and multiple apk file generation
- Code templates to help you build common app features
- Rich layout editor with support for drag and drop theme editing
- Lint tools to catch performance, usability, version compatibility, and other problems
- ProGuard and app-signing capabilities
- Built-in support for Google Cloud Platform, making it easy to integrate Google Cloud Messaging and App Engine

The home page of the Android Studio is located at:
http://developer.android.com/tools/studio/index.html.

2.2. Android

Android is a mobile operating system initially developed by Android Inc., a firm purchased by Google in 2005. Android is based upon a modified version of the Linux kernel. Android dominates the world mobile operating system market share by a large margin.

The Android operating system software stack consists of Java applications running on a Java based object oriented application framework on top of Java core libraries running on the Dalvik virtual machine featuring JIT compilation (for versions <= Android 4.4 KitKat) and the ART runtime from Android 5.0. The Official Android home page is located at: http://www.android.com/. The Official Android Developer home page is located at: http://developer.android.com/index.html
You can start learning android at https://developer.android.com/training/index.html.

2.3 Facebook

Facebook is a social networking service launched in February 2004, owned and operated by Facebook Inc. Users can add friends and send them messages, and update their

personal profiles to notify friends about themselves and what they are doing. Users can additionally post news feeds to their profiles, and these feeds may include images, besides text messages.
The Facebook homepage is available at: http://www.facebook.com. Facebook provides developers with an application-programming interface, called the Facebook Platform.

3. Prerequisites

This homework requires the use of the following components:

1. Download and install Android Studio. You may use any other IDE other than Android Studio such as Eclipse, but you will be on your own if problems spring up. First you need to install Java on your local machine. You can download JDK 7 from - http://www.oracle.com/technetwork/java/javase/downloads/index.html. For windows users, after installing the JDK, you need to add environment variables for JDK.

- **Properties -> Advanced -> Environment Variables -> System variables -> New Variable**
  Name: **JAVA_HOME**, Variable Value: **<Full path to the JDK>**
- Typically, this full path looks something like **C:\Program Files\Java\jdk1.7.0**. Then modify the PATH variable as follows on Microsoft Windows: **C:\WINDOWS\system32;C:\WINDOWS;C:\Program Files\Java\jdk1.7.0\bin**
  This path may vary depending on your installation.
- **Note:** The PATH environment variable is a series of directories separated by semicolons (;) and is not case-sensitive. Microsoft Windows looks for programs in the PATH directories in order, from left to right. You should only have one bin directory for a JDK in the path at a time. Those following the first instance are ignored. If you are not sure where to add the path, add it to the right of the value of the PATH variable. The new path takes effect in each new command window you open after setting the PATH variable.
- Reboot your computer and type "**java –version**" in the terminal to see whether your JDK has been installed correctly.

Set up the Android Studio environment so that you can run any sample android app on your phone/tablet/virtual device from it. Then you can start with this homework app. You will need to enable "Developer Options" and "USB debugging" is you are using an actual device. There are endless resources a simple search away on how to setup your Android Studio.

2. You also need to create a Facebook Developer application as you did for your homework 8. Go to https://developers.facebook.com/apps/ and follow the steps:

- Create a new **Android** application
- **Download SDK**: Download the latest Facebook Android SDK 4.0.1

- **Install Facebook** on the emulator in case you are using an emulator in lieu of a device
- **Import SDK** to your Android Studio Project
  Incase you face issues with this step, you may look at
  - http://stackoverflow.com/questions/24466921/android-studio-0-8-1-how-to-use-facebook-sdk
  - https://trinitytuts.com/integrating-facebook-sdk-application-android-studio/
- Specify **App Info**
- **Key Hashes**: Specify Android key hash for the development environment using the commands mentioned
- Track App Installs and App Opens: Not required
- Next Steps: Utilize **Login** and **Share** tutorials to achieve the functionality required for the exercise.

Note: In your Facebook application settings, you should go to the "Status & Review" section and choose "Yes" for the question "*Do you want to make this app and all its live features available to the general public*?" as you did for homework 8.

4. Description of the Exercise

In this exercise, you will implement an Android Mobile application (hereby the "app") that ports the Homework 8 web interface to a mobile interface. The app does the following sequence of actions:

4.1. Initial Form

Create an Android Activity that takes the "Keyword"(**mandatory**), "Price From", "Price To", "Sort By" as input. The Keyword, "Price From" and "Price To" fields are text fields (*called EditText in Android*), while the "Sort By" field is a dropdown menu selector (*called Spinner in Android*). For the Sort By field four values should be populated: "Best Match" (**default**), "Price: highest first", "Price + Shipping: highest first", and "Price + Shipping: lowest first". The form should also contain Clear and Search Buttons. Sample screenshot is shown in Figure 1.

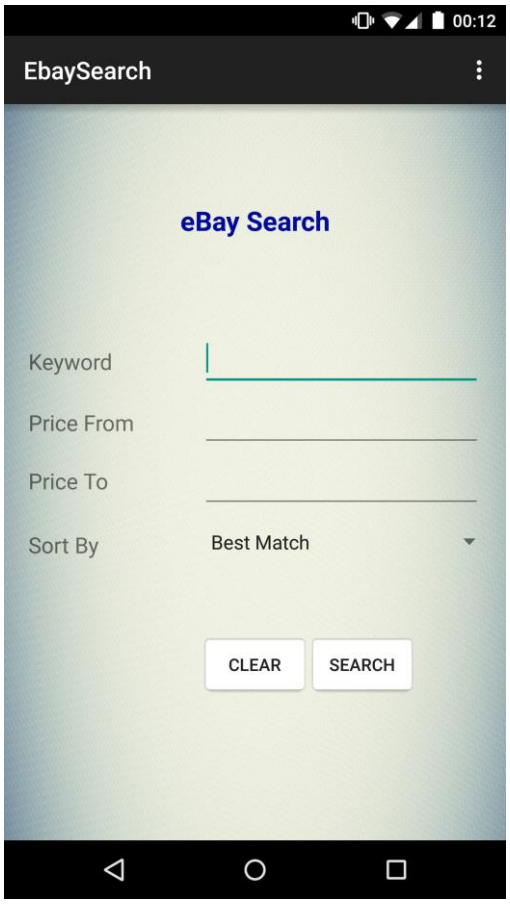

Figure 1: The Form

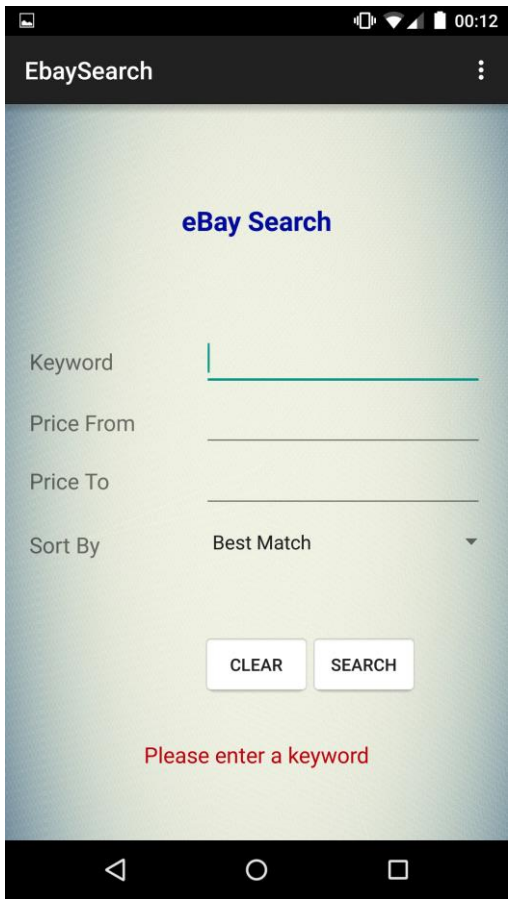

Figure 2: Error handling in the form

On clicking on the "Clear" button, the fields should be reset such that the text fields are empty and the Sort By spinner must have the default value. Also any previous error messages must be cleared.

When the "Search" button is tapped, the search form is validated first. If the values provided in the form are valid, the app makes a request to the PHP script located on your AWS account (similar to hw8) and retrieve the JSON result. Another activity should be displayed to show the results, which is a scrollable list of items described below.

4.2. Result Display

This should have a **scrollable** list detailing the items for the keyword. For each item, its image (using galleryURL or pictureURLSuperSize, whichever is available and fits the size), title, and price with shipping cost must be shown. Price with shipping cost must be displayed similar to HW8 such as "Price: $45 (FREE Shipping)" or, "Price: $45 (+$5 for Shipping)". If any of the field is unavailable, you should show "N/A" instead. This activity should look similar to Figure 3 below.

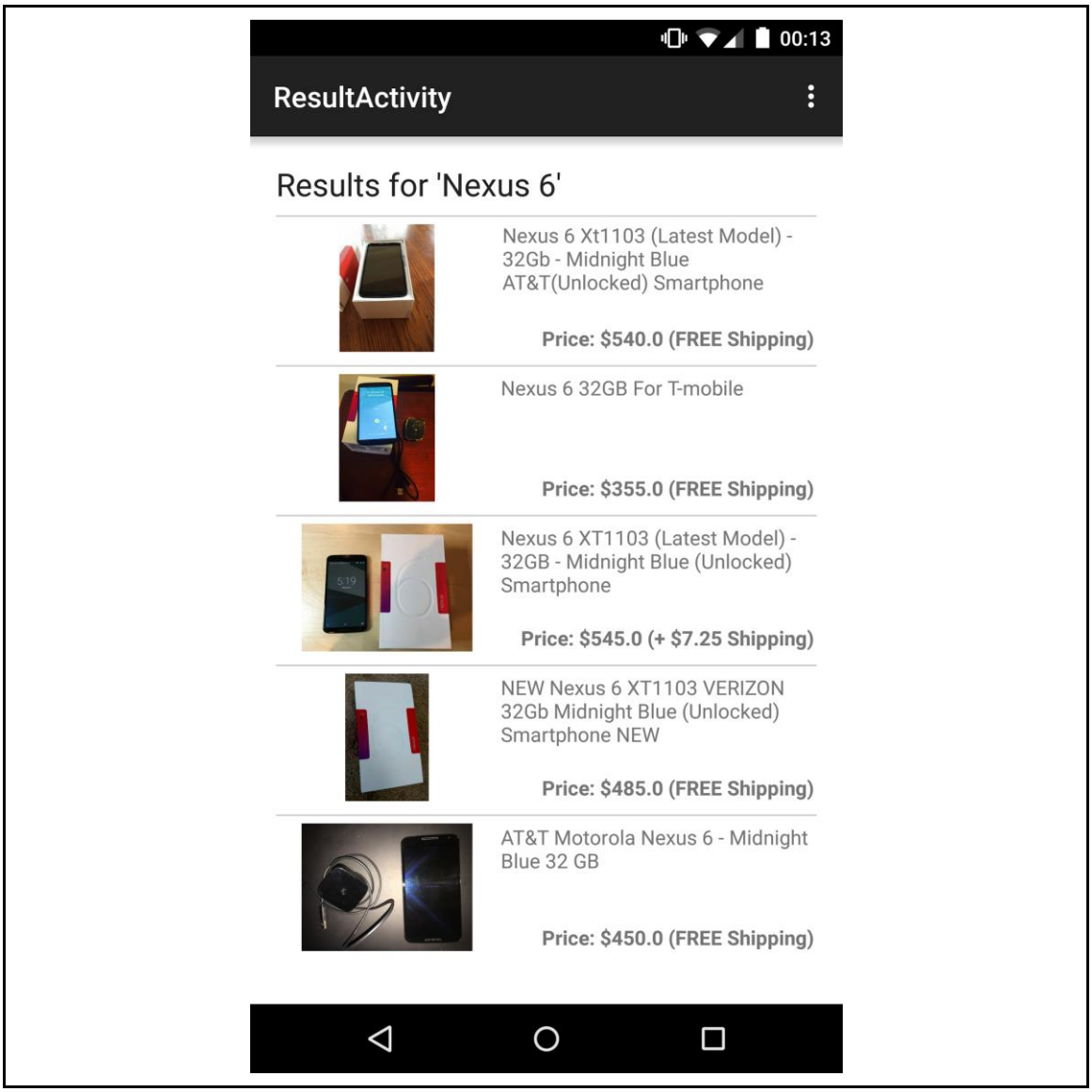

Figure 3: Results.

The image above does not show scrolling since all items fit on a large screen, but you must make the entire activity scrollable. **Please note that only 5 results would be displayed for a keyword so you can hardcode ResultsPerPage for the eBay API call as 5.**

Tapping on the title of an item must invoke the "Item Details" activity described next.
Tapping on the image of an item must redirect the user to the item's eBay page in the mobile browser (NOT in the app).

4.3. Item Details Display

In the third and final activity, you must show the details of the one particular item that was clicked/tapped in the Result Activity. You need to show the image of the item at the top; preferably using pictureURLSuperSize, but galleryURL must be used in case the former

is unavailable/empty. The title, price (including shipping) and location should be shown below the image. Figure 4 shows the Item Details activity.
Below this, you must show a "top rated" badge to the left edge, if applicable, similar to hw8. At the center(vertically), there must be a "Buy Now"/"Buy It Now" button which links to the actual ebay page of the item in the browser. To the right edge, there must be the "Facebook" button which posts details to Facebook.

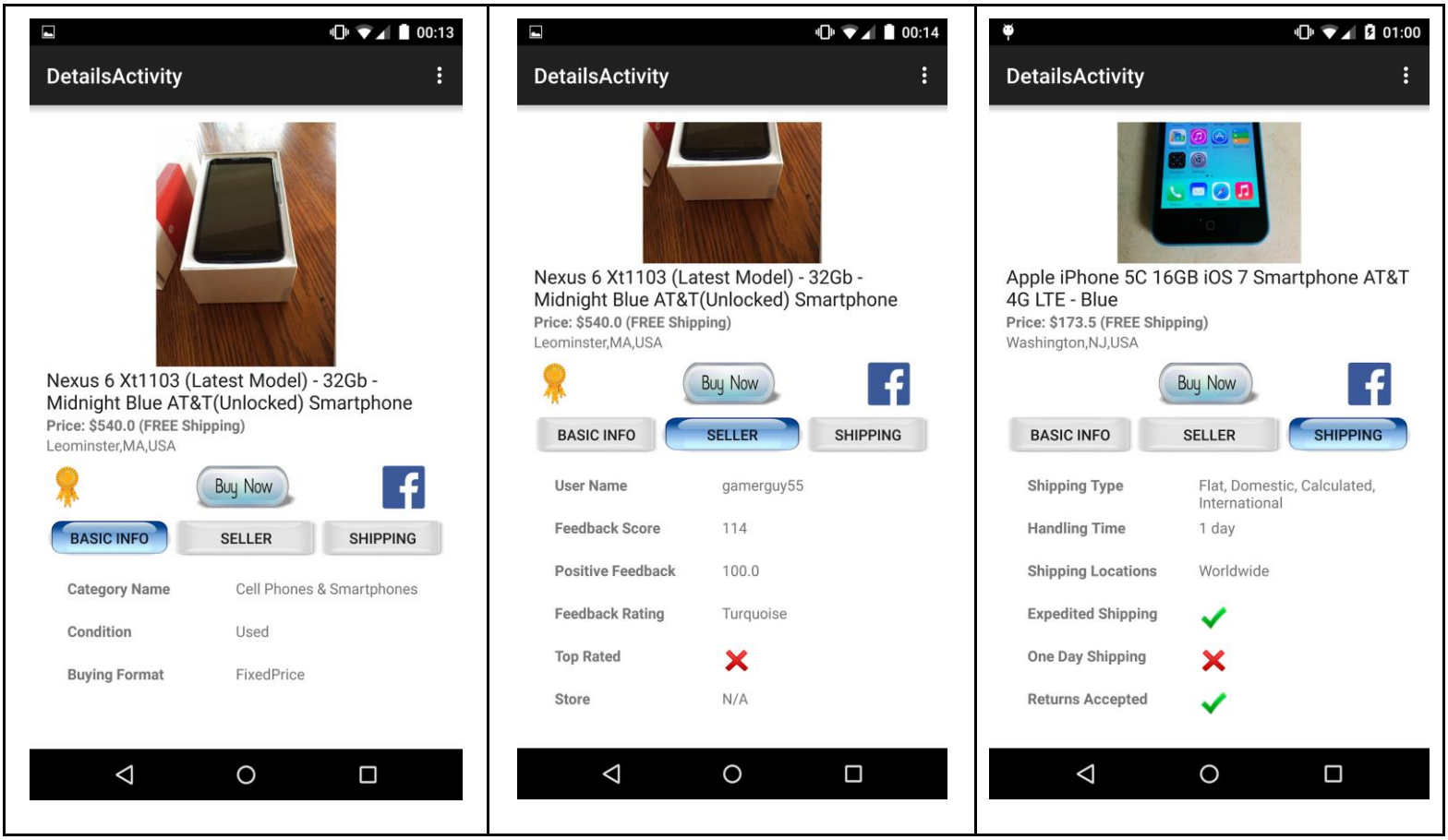

Figure 4: Details Activity: the three tabs. Third image is for an iphone item. Notice the top rated badge, shipping type and store fields.

Below the badge/buy now/FB buttons which are aligned horizontally, you must show the details of the item similar to hw8. There is no bootstrap to help you with this, so you should use Android views to create the tabs, handle tab functionality, highlight the current tab button, display the current tab details and hide the non-current tabs. The entire Activity, from the main image at the top to the tabs(inclusive) must be scrollable so that one can see the full details.
You are free to use any appropriate image for the "check yes", "check no", "tabs" (highlighted and un-highlighted), and "buy now" buttons. You may use the same Facebook and badge images as in hw8. Make sure the Shipping Type field is separated by space or commas (3rd image in Figure 4). For any unavailable fields, use "N/A" as shown in the middle image for "Store".

4.4. Facebook

Clicking on the facebook button must allow users to login and post item details on their own Facebook pages. The pop up window that first appears should also have an option to cancel the action, along with posting on Facebook. The information posted to Facebook must contain item title, price with shipping cost, location, image (galleryURL) and link to the eBay page for the item (viewItemURL).

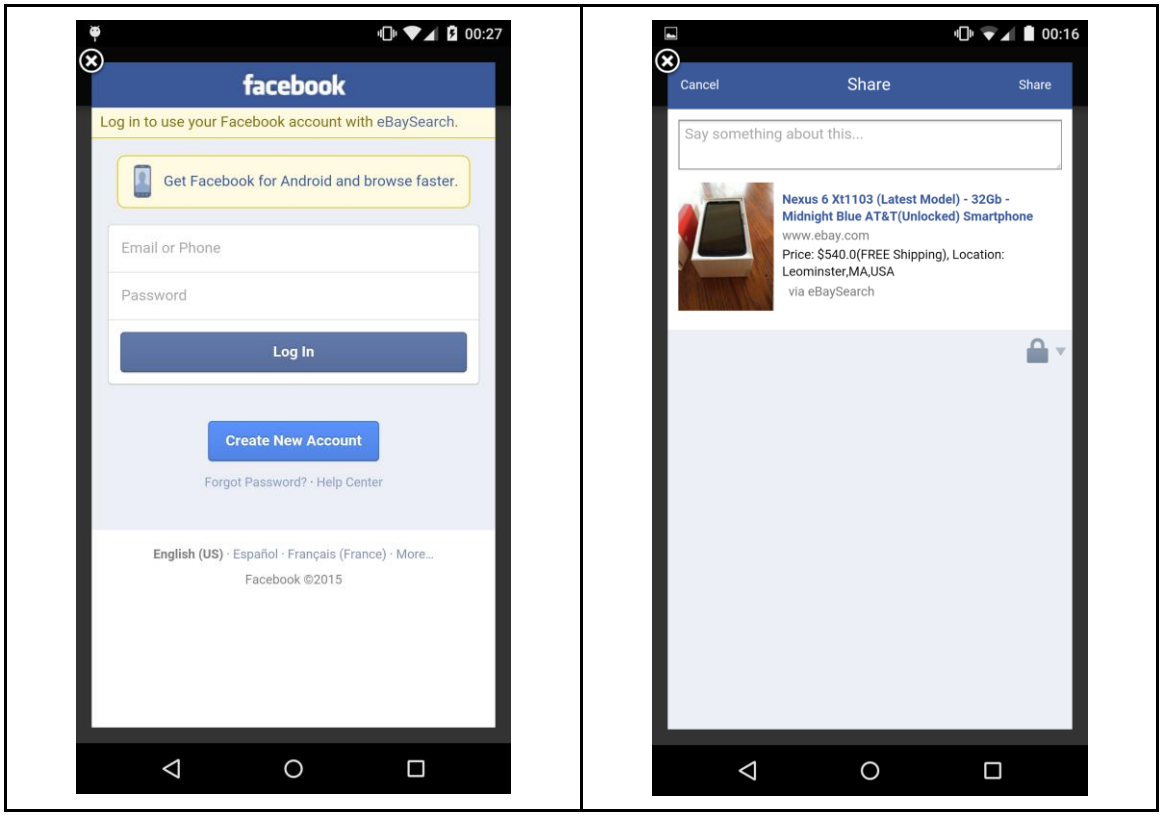

Figure 5: Facebook Post

On successful posting (or failure), appropriate short "**Toast**" messages should be shown at the bottom 'Posted Story, ID: XXXXX..' or 'Post Cancelled' for cancelled post.

4.5. Error handling:

In the first activity, all errors must be shown at the bottom as shown in figure 2.

a. **Keyword Empty Field:** The keyword must not be empty.
b. **Price From Field:** Must be a positive integer or decimal number.
c. **Price To Field:** Must be a positive integer or decimal number.
d. **Price From and Price To Field:** Price To must not be less than Price From.
e. **Invalid Input:** If there were no matches found by the eBay API , then an error message "No Results Found" should be shown at the bottom. You may use different text style/font/size/color for this error, but that is **not** mandatory.

5. Implementation Hints

- Facebook offers multiple APIs for you to upload or request data. In this HW implementation, you need to use 2 Facebook APIs (Login and Share Dialog). You may also use any other facebook API if you desire so.
- You need to use **AsyncTask** class to fetch the JSON data/item images in background in order to avoid any "YourApp is not responding" error.
- Using **Relative Layouts** as containers for views will be very helpful, since you can hide/unhide an entire layout view which will hide/unhide all its children views. It is also good to position elements relative to their parent views.
- Android "Toast" messages can be very helpful in debugging an app.

6. Materials to submit

Unlike other exercises, for this homework you will have to demo your submission **in person** during a special grading session. Details and logistics for the demo will be provided in class, the class website, and Piazza.

You must also **ZIP** your Android **source** directory and **SUBMIT** the resulting ZIP file. Make sure that the source path does not include the .apk binary file.