Mini Project 2: Directory

Due: Mar. 12, 23:59 PDT

Overview

Build a directory app. Topics covered include RecyclerView, Action Bar & Menu, Dialog, and Intent. Fig. 1 includes screenshots that show the functionalities.

Set the project name as “Project2YourName”, and your app name as “My Directory”. Feel free to use the code in exercise 2 (RecyclerView) and 3 (Intent).

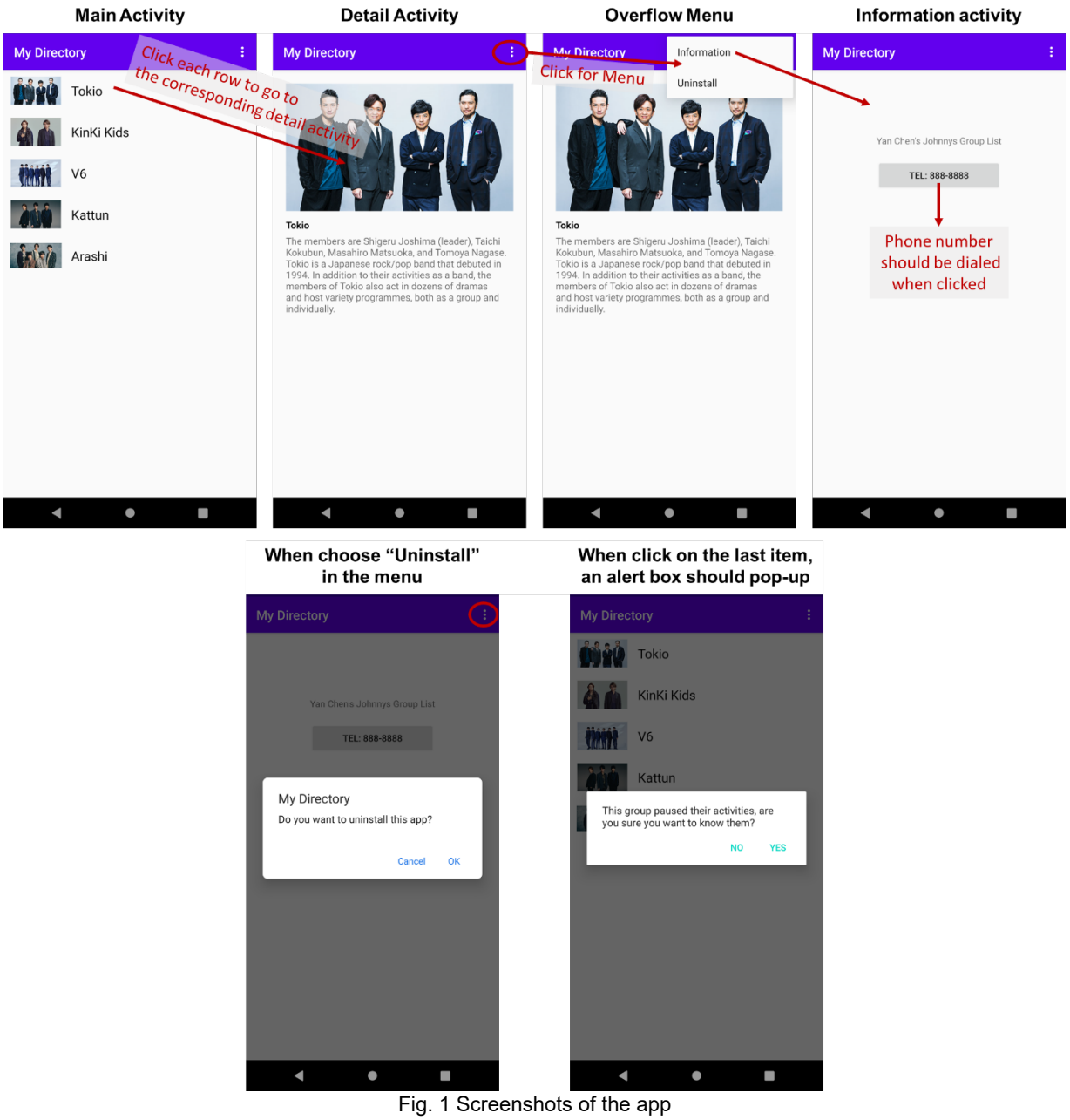

Fig. 1 Screenshots of the app

Activities

The app has 3 activities. That is, 3 Activity classes and 3 UI files.

1. Main Activity

The Main Activity (launch page) should be a list of items of your choice. You need at least 5 different items. Each row in the list should have:

- A thumbnail picture of the item
- The name of the item

This should be implemented as a RecylerView. When any part of a row is clicked, it should launch the Detail Activity for the corresponding item.

2. Detail Activity

The Detail Activity shows details on the item chosen in the list. The activity should show:

- The name of the item
- A large image of the item
- A short description of the item

The same Detail Activity should be reused for each item. That is, there is only one UI file for the Detail Activity, the actual content will be passed in from Main Activity based on which item being clicked.

3. Information Activity

The information activity should have:

- The name of the directory (please include your name, e.g., Yan Chen's Celebrity Directory)
- A phone number (any phone number, or just 888-8888) represented as Button or a TextView. When clicked, the phone number should be dialed. Use the Intent.ACTION_DIAL intent for this.

This activity should be triggered by information option of the Menu Items.

Menu Items

The ActionBar should remain persistent throughout the app (appear on all activities) with 2 options:

- Information. This should launch the Information Activity described above.
- Uninstall. This should call an intent to uninstall the app. Refer to project 1 for this if needed.

The menu items listed above should appear in the overflow menu, NOT as buttons in the action bar. Recall that only frequent, important, or typical actions are supposed to be buttons in the ActionBar.

Dialog Box

When the user clicks on the last item in the Main Activity, an alert box should pop-up, warning the user and asking the user if they want to proceed. If the user clicks "Yes", then the app should proceed as normal. If the user clicks "No", then the app should remain on the Main Activity.

Additional Notes & Tips

Register Activities

Since you have different activities, you need to register all of them in the manifest file as child elements of application tag. (MainActivity should already automatically registered)

```
<application ...>
    <activity android:name=".InfoActivity"/>
    <activity android:name=".DetailActivity" />
    <activity android:name=".MainActivity">
        ...
    </activity>
</application>
```

About the Data (Items)

- You can pick any kind of items you want to display. For example, a list of animals, a list of celebrities, a list of anime, a list of food, a list of sweets, etc.
- You can specify any details. For each item, you may want a description that is not long, so that you don't need to worry about if it will fit the screen or if you need to add a scroller (of course you can explore this if you want). Don't spend too much time thinking of these details. Feel free to be creative or boring. You can even just put "A description of XXX" for the description part.
- You can just hardcode the data. You may want to create a class for your items so it's easier to manipulate. For example, define a class called Animal with name, description and the image, then hardcode an ArrayList of animal objects. Refer to lesson 6 page 19 - 21 if needed.

```
ArrayList<Animal> data = new ArrayList<>();
data.add(new Animal("Panda", "Description of Panda"),
                    R.drawable.panda));
```

About the Images

- The simplest way may be to store the pictures in res/drawable folder. You can find the folder by right click the drawable folder in the project file list and choose "Show in Explorer". Place your .png or .jpg files under the drawable folder. Then set the image to a View use setImageResource:

```
image.setImageResource(R.drawable.filename);
```

- Although you need set all images/text in Java code (either in the adapter for recycler or pass in through another activity), you can set one of the items in the UI file to help you place the items. Here is the code for setting an image in UI file (if you have saved the image in drawable folder):

```
app:srcCompat="@drawable/filename"
```

- Your rows should align correctly, which means each row should have the same height, and the images should be in the same size. A simple way may be choosing images of the same size or change them to the same size before putting them into the folder.
- You can use the same image for both thumbnails and full images. Or different. Again, the content details are up to you.
- Try to use relatively small images (below 500kb). You may run into memory issues otherwise.

Functionality Requirements

Your project will be graded based on if your app meets all requirements.

Requirement	Points
The app has all three required activities	0.5
App is well-polished (views should align properly, etc.)	0.5
Main Activity implemented using RecyclerView	1
The warning dialog when clicking the last element works	0.5
Details Activity gets the data passed from the Main Activity	0.5
Information Activity has a phone number that will be dialed once clicked	0.5
Your name is on the Information Activity	0.5
The action bar menu is throughout the app	0.5
The information option on the overflow menu works	0.25
The uninstall option on the overflow menu works	0.25
Total	**5**

Submission

- Push your project to a Bitbucket repository (name it “project2”) by the due date.
- Invite and share your Bitbucket repository the grader (edmond.lin@sjsu.edu) and the instructor (yan.chen01@sjsu.edu).
- Submit “Mini Project 2 - Directory” assignment on Canvas using the template provided (see description of the assignment).
- Only your last submission before deadline will be graded based on the criteria in Functionality Requirement.
- (Optional) Post a discussion on Canvas to share any suggestions/tricks/hints, etc. for finishing this project. For example, how to set a onItemClick listener in recycler view, how to set an object that can be passed from one activity to another, how to make the ActionBar throughout all activities, etc. Check other posts before posting to avoid any duplicates.