Project 2 – Zoo Directory

Overview

In this homework assignment, you will create an app for a fictional zoo. The app provides a listing of animals, detailed information of each animal, and basic information about the zoo.

This homework will give familiarity with ListView, adapters, menus, dialogs, and intents.

Requirements

The zoo app has the following requirements:

Activities

The app has 3 activities:

1. Animal listing activity

Your zoo should have at least 5 animals (you can pick them). Each row in the list should have:

- a thumbnail picture of the animal
- the name of the animal

This should be implemented as a ListView. When any part of a row is clicked, it should launch the details activity for the corresponding animal (described next).

2. Animal detail activity

The animal detail activity shows details on the animal chosen in the list. The activity should show:

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

The **same** detail activity code should be used for each animal, but with different content, obviously

3. Zoo information activity

The zoo information activity should have:

- The name of the zoo
- A phone number, represented as Button or a TextView. When clicked, the phone number of the zoo (888-888-8888) should be dialed. You must use the `Intent.ACTION_CALL` intent for this.
    - *Note: you can either set targetSDKVersion to 22 in build.gradle file or keep it as 23 (default) and implement run-time permission request mechanism.*

This activity should be triggered by one of the menu items (see the "Menu Items" section below).

Menu Items

The ActionBar should remain persistent throughout the app. The overflow menu should have two iteams:

- Information. This should launch the zoo information activity (described above).
- Uninstall. This should call an intent to uninstall the app. You can use the `Intent.ACTION_DELETE` intent for this. (Refer to this page)

The menu items listed above should appear in the overflow menu and not as buttons in the action bar. This is because only frequent, important, or typical actions are supposed to appear as buttons in the ActionBar.

Dialog Box

When the user clicks on the last animal in the animal listing activity, an alert box should pop-up, warning the user that the animal is very scary 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 same activity.

Favorite Toggle Icon

In the Animal Listing Activity, add a “Favorite” icon towards the right side of each row. Initially by default each row is a non-favorite. When user clicks on the icon, it changes to mark that row as user’s favorite. Clicking again will turn the icon back to non-favorite state. This is similar to Gmail star feature. See picture attachment in the end.

Notes

- You can choose the specific details of the zoo (the name of the zoo, the animals, the descriptions, etc.). Don't spend too much time thinking of these details. Feel free to be creative or boring. You can reuse the animals in the sample code shown in class.
- You can just hardcode the list of animals in code, similar to what we saw in class. You don't have to use SQLite. You can just store the pictures in the assets or drawables folder.
- You can store separate images for the thumbnails and full images.
- Try to use relatively small images (below 500kb). You may run into memory issues otherwise.
- You can support both landscape mode and portrait modes, in which you need to make sure they look reasonable in both cases. Alternatively, you can fix your activities in portrait mode. Search the web for fixed mode solutions.

Scoring

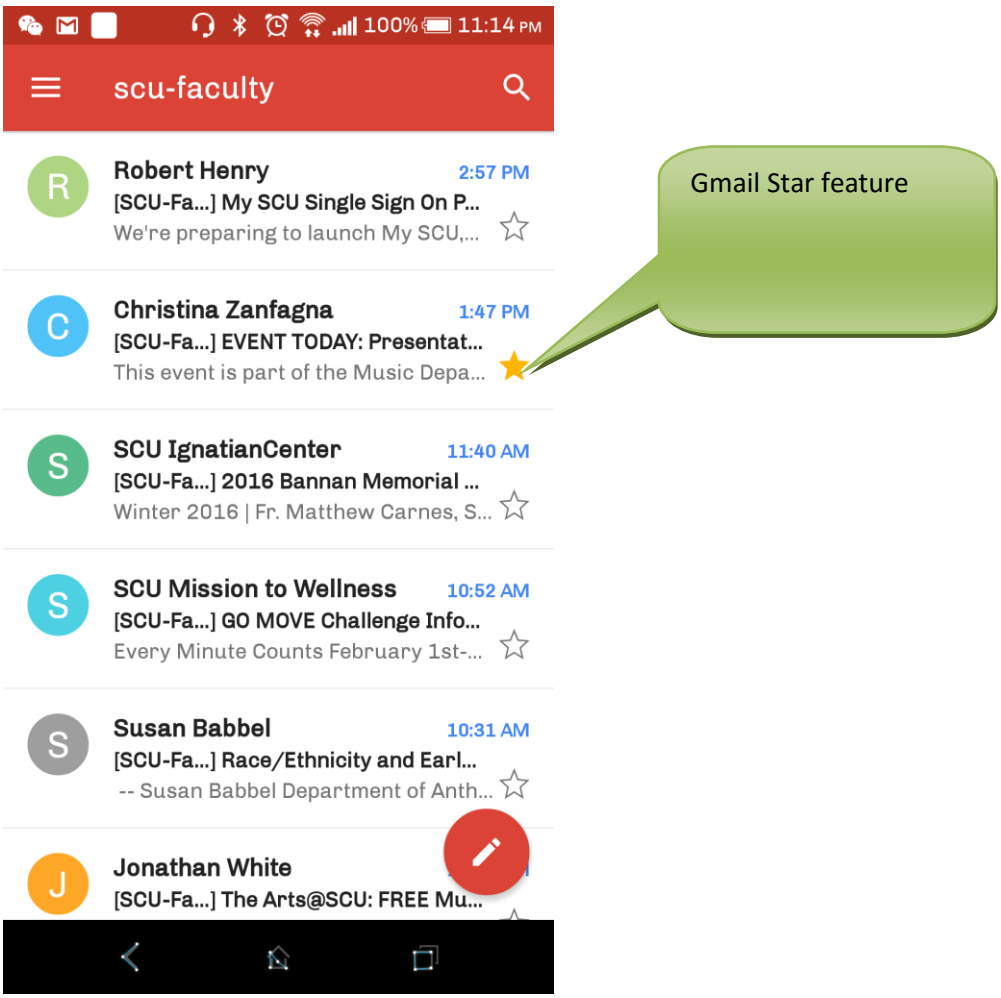
Total point is 10.

- Functionalities (8 pt)
  - Implement 3 functional activities
  - Implement Menu items, dialog and favorite toggle icon
  - No crashes or visual defects
- Layout & UI (1 pt)
  - Aesthetically pleasing with customized launch icon
  - Render well in both landscape mode and portrait mode; Or has fixed portrait mode
- Programming & coding (1 pt)
  - Neat & clean programming with clear naming
  - Correct usage framework methods

Submission

- Make sure your project builds on the latest Android Studio and runs on emulator before submission
- Include a README.txt file at top level of project directory. Teacher/grader will read this file first before building and testing your program.
  - List any special studio settings, project settings or demo/testing instructions if necessary.
  - Where are image files stored? Are you using separate icon images and full-view images?
  - Do you support both viewing modes or have fixed portrait modes?
- Rename your project directory as *lastname.firstname.pa2* (e.g., sun.jun.pa2) Recall that project directory can be located by switching to “Project” view in Android Studio.
- Zip the whole project directory as a ZIP file with naming as *lastname.firstname.pa2.zip*
  - *Note: Wrong file/directory name will deduct up to 1 point*
- Submit the zip file via Camino/Canvas

Appendix


100% 11:14 PM
scu-faculty
Robert Henry 2:57 PM
[SCU-Fa...] My SCU Single Sign On P...
We're preparing to launch My SCU,...
Christina Zanfagna 1:47 PM
[SCU-Fa...] EVENT TODAY: Presentat...
This event is part of the Music Depa...
SCU IgnatianCenter 11:40 AM
[SCU-Fa...] 2016 Bannan Memorial ...
Winter 2016 | Fr. Matthew Carnes, S...
SCU Mission to Wellness 10:52 AM
[SCU-Fa...] GO MOVE Challenge Info...
Every Minute Counts February 1st-...
Susan Babbel 10:31 AM
[SCU-Fa...] Race/Ethnicity and Earl...
-- Susan Babbel Department of Anth...
Jonathan White
[SCU-Fa...] The Arts@SCU: FREE Mu...
Gmail Star feature