

App Development for Smart Devices

CS 441/541

Assignment #2: Contacts List

Published: Thursday, Feb 15 2018
Deadline: 11:59 PM, Friday, Mar 2, 2018

In this assignment, you will be building an app to practice using fragments and multiple activities in an application.

All projects are required to be submitted with (up to) a one-page report that specifies what was implemented, and until what point. You also need to justify your design decisions. Graduate students need to submit additional report of the graduate student requirements.

1. 50 points maximum

Write a contacts app in portrait view

- a. There should be a scrollable list view of all contacts. (5 points);
- b. There should be a check box in front of each name. The user can check multiple names at a time and press “delete” button to delete the checked contacts (5 points).
- c. When user clicks the contact name, it should display the “contact profile” (5 points).
- d. When user clicks the add button, the app should jump to a new activity “Contact Details” (5 points);
- e. In “Contact Details” activity, the user can input the name, phone number, and relationship with contacts that exist in the contacts list (5 points). User presses “Add Person” button to confirm input information.
- f. The “relationship” describes the link between the different contact people. The list should display the whole list of contacts. And the checked name should be moved to the top of the contacts list (10 points).
- g. When a user clicks the contact name under “relationship” in “Contact Profile”, it should jump to the contact profile of the clicked name by creating a new activity (5 points).

- h. When a person Emily is selected as related to person Bob – Bob should show up as a relation in Emily’s “Contact Profile” as well. (10 points).

The image shows three UI panels for a contact management application. The first panel, titled "Contacts", is a list of seven contacts: Contact Name 1, Contact Name 2, Contact Name 3, Contact Name 4, Contact Name 5, Contact Name 6, and Contact Name 7. Each contact has a checkbox to its right, with Contact Name 1, 2, 5, and 7 checked. At the bottom are "Add" and "Delete" buttons. The second panel, titled "Contact Details", has fields for "Name:", "Phone Number:", and "Relationship:". The "Relationship:" field contains a list of six contacts: Contact Name 1, Contact Name 2, Contact Name 3, Contact Name 4, Contact Name 5, and Contact Name 6, each with a checkbox. Contact Name 1 and 2 are checked. An "Add Person" button is at the bottom. The third panel, titled "Contact Profile", has fields for "Name:", "Phone Number:", and "Relationship:". The "Name:" field contains "Bill Gates". The "Phone Number:" field contains "(123) 406-789". The "Relationship:" field contains a list of three contacts: Contact Name 1, Contact Name 2, and Contact Name 3.

2) 50 points maximum

The app needs to have a different layout with landscape view.

- Use fragment to display the contact list on the left (5 points);
- The information you input in portrait view or landscape view needs to stay constant when the orientation changes (10 points).
- When user clicks the contact name, the contact profile should be shown on the right side (5 points).
- When user clicks the “Add” button, the input information also needs to show on the right side (5 points).
- The delete button, relationship list are the same as portrait view (10 points).
- The left fragment and right fragment should occupy $\frac{2}{5}$ and $\frac{3}{5}$ of the width of screen, respectively (5 points).
- When a user clicks the contact name under “relationship” in “**Contact Details**”, it should jump to the contact profile of the clicked name. And the contact profile should also display on the right side in the landscape view (10 points).

Contacts	Contact Details												
Contact Name 1 <input checked="" type="checkbox"/>	Name: <input style="width: 100%;" type="text"/>												
Contact Name 2 <input checked="" type="checkbox"/>	Phone Number: <input style="width: 100%;" type="text"/>												
Contact Name 3 <input type="checkbox"/>	Relationship: <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Contact Name 1</td><td><input checked="" type="checkbox"/></td></tr> <tr><td>Contact Name 2</td><td><input checked="" type="checkbox"/></td></tr> <tr><td>Contact Name 3</td><td><input type="checkbox"/></td></tr> <tr><td>Contact Name 4</td><td><input type="checkbox"/></td></tr> <tr><td>Contact Name 5</td><td><input type="checkbox"/></td></tr> <tr><td>Contact Name 6</td><td><input type="checkbox"/></td></tr> </table>	Contact Name 1	<input checked="" type="checkbox"/>	Contact Name 2	<input checked="" type="checkbox"/>	Contact Name 3	<input type="checkbox"/>	Contact Name 4	<input type="checkbox"/>	Contact Name 5	<input type="checkbox"/>	Contact Name 6	<input type="checkbox"/>
Contact Name 1	<input checked="" type="checkbox"/>												
Contact Name 2	<input checked="" type="checkbox"/>												
Contact Name 3	<input type="checkbox"/>												
Contact Name 4	<input type="checkbox"/>												
Contact Name 5	<input type="checkbox"/>												
Contact Name 6	<input type="checkbox"/>												
Contact Name 4 <input checked="" type="checkbox"/>													
Contact Name 5 <input type="checkbox"/>													
Contact Name 6 <input checked="" type="checkbox"/>													
<input type="button" value="Add"/> <input type="button" value="Delete"/>													

Graduate students' additional requirements (50 points maximum):

Add the ability to store a photo with each contact, which is displayed near the upper right corner of the contact details fragment. Make sure that photos are not lost when your app terminates (you are allowed to store photos as a file). When no photo is available, display a generic head icon in the same area. Tapping this item should allow the user to take a photo and automatically assign it to the contact. If a photo is available, tapping this photo should lead to a zoom in animation until the photo occupies the entire screen. Do your own research in the Android Libraries to find the appropriate class for the zoom feature.

Upload your solution source code, apk package, and report in the same zip or tar.gz file named as yourlastname_yourfirstname. The apk file should be copied to the root directory of the submission folder, also keep the original apk in the bin folder.