**Report of Contact List**

Xin Yang

NetID: xy213

ECE Graduate Student

This app realized all of the function required including graduate additional part. User can see the main activity at the very beginning, and when clicked the add button, a contact detail activity will show up and allow user to add contact, also user can give camera permission and take photos to set contact avatar, and when user clicked the relation list, selected item will jump to the first, when added, the name will also be added to corresponding contact’s relation list. When user click the item in lists in both detail or profile activities, the clicked contact’s profile will appear, showing all information related. Also when the phone is rotated to landscape view, the app will adjust its view to fit the screen, and input information can be saved after rotation. Left side shows the main activity and the right part shows contact detail or contact profile according to user’s operation.

User can save photo for contact and see it at profile page, otherwise a default android avatar will be set. When avatar is clicked, user will see the zoomed image.

To fill the list in main activity and relationship list in contact detail, customized Base Adapter classes are used, also a Hashmap inside is utilized to store checkbox selection when orientation changes. For contact profile’s relation list, a simple Array Adapter is used; Except the avatar is saved as image file, all other parameters or variables are stored via SharedPreferences instead of file or database, all complex data types as Map or Bitmap is transferred into bundle or bytes for passing data between different fragments or activities.

However, this app is not perfect when you are in portrait main and rotate screen, then click the add button or go to profile page, then rotate again to portrait, the shown page will still be main activity in that the current activity is fixed as main when first rotate. To realize save current instance while rotation, you’ll have to go into certain activity under portrait mode, and then rotate the screen and back, to make sure the two orientations are under the same activity. This problem can be resolved for sure but due to limited time, I tried my best to achieve this and I’m here to clarify this cons.