Team member list: Charu Mishra & Puja Soni
Device Name: Slowking
Project/App Title: LifeBudsBucketList

Basic instructions on usage:

This Android App is a simple bucket list app that allows you to enter and keep track of any activities you need/want to do.

You can add new items to the list by clicking the "+" button in the lower right corner of the main screen - directing you to a screen where you can enter your activity's name, description, latitude and longitude, as well as pick a due date.

Each item in the list has a checkbox next to it, allowing you to keep track of which activities you've completed so far. Simply check the box when you complete an activity.

The list is sorted such that incomplete tasks are "first" (at the top of the list) and complete tasks are at the bottom. Within these 2 groups, the tasks are then organized by due date. This keeps the task that is incomplete and due soonest at the top of the list for you.

You can also edit already created items by clicking on the task in the main screen. This will bring up an edit item screen with the corresponding data for that task. Any changes made here will be reflected back in the list after the you click the "save" button.

Lessons learned:

We learned a lot from this project, especially since one of us had never done android development before. One major lesson we learned about was about how to use intents to send data back and forth between activities, which we needed to do for the edit and add item activities. We learned how to create an intent and use the putExtra and getExtra methods. An obstacle we had with this though was that the data needed to be in a string format, but we created our own taskDate class for all of the date information. To solve this issue, first we tried using serializable to send the date object to the next activity but our app crashed with this. We ended up fixing the problem by sending the data necessary to create a taskDate object (year, month, and day) separately/directly and from this created the object in the BucketListActivity. Sending the data and updating it for the edit item activity was probably the most difficult lesson to learn because we needed to figure out how to make each item in the recycler view clickable so that when edited, the updates would be sent back to that item and show up on the main screen. First we put a rowview in the linear layout for bucket item, but with this we had to click a lot of times for the screen to register it was a click and show the edit screen. We fixed this by putting the rowview in the relative layout instead and it solved that issue.

Also, we learned how to manipulate the xml files to make the view the way that we wanted it to be. When we first started coding for the app, everything was in a relative layout and so we ran into the issue that the due date would show up next to the activity name rather than underneath it. We fixed this by adding a linear layout. Later we had another issue that save button for the add item / edit item screens wouldn't show up because it didn't fit into the page. We fixed this by learning about ScrollView and adding it to the xml files. This created a scroll bar on the screen so then we were able to scroll on the screen and see/use the save button. We

learned a lot about how to change the xml files to adjust the layout/appearance of the app by running into little issues like this and solving them.