**Report of ToDoList**

Xin Yang

NetID: xy213

ECE Graduate Student

The ToDoList app realized function of adding task, deleting task through long click or checkbox(additional requirement for graduate students), writing data into txt file and synchronizing database and task list, which can make sure when user close the app and restart, the added tasks won’t be cleaned. Detailed process is stated as followed.

When the app starts with the onCreate function, it will first declare several variables and objects then initialize. Then create a MyDatabaseHelper object and mySimpleCursorAdapet object, separately extended from SQLiteOpenHelper and SimpleCursorAdapter in order to create the database and provide a data adapter to fill the listview. After that, bind SimpleCursorAdapter to listview and refresh the view to fill listview with data from task database. A MyDatabaseHelper class is to help create a database on the first time and create the task table with “\_id”, “title” and ”description” keys.

Then define several functions to realize main function. First is the refreshListView function, which set a new cursor and fill it with task data from database, then change former cursor to this one, which can refresh the content. Then is the insertData function, which uses ContentValues to put data from EditText into databse, then toast if seccessed and refresh listview, reset edittext for user’s next type. deleteData function create a new cursor and move to the current position of listview item, then get the \_id of item and delete from database, then refresh.

Also there’re three listeners, one for add button, one for long press delete on task and another for the delete checkbox of every item. The onclicklistener of add button mainly play two roles, get user’s input and save task to txt file and insert into database. A FileHelper class was built to achieve this, where utilizes file stream to save input task as single file in default directory, and make a toast to show succeed or not. The onItemLongClickListener will pop up a dialog to ask user if they are sure to delete the task, if pressed yes, deleteData function will be called to delete the task, then keep listen for further action. The checkbox listener is designed inside the MySimpleCursor class to get the current item position for delete, and an ArrayList was declared to store the checkbox selection information in case chaos caused by listview refresh, the listener will judge if a checkbox is selected, then delete the located task item directly from database.