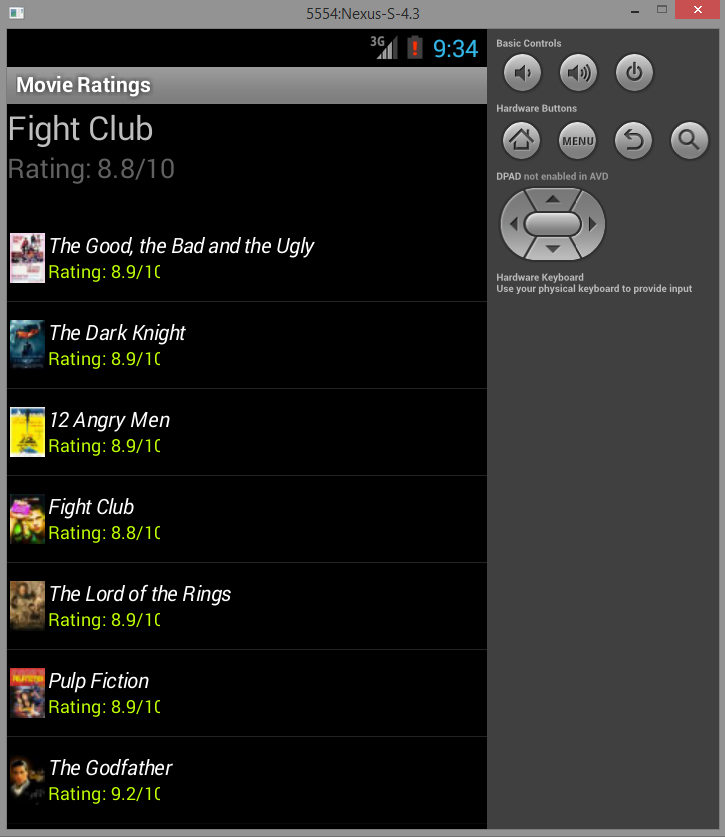
**Submission for Assignment 06**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

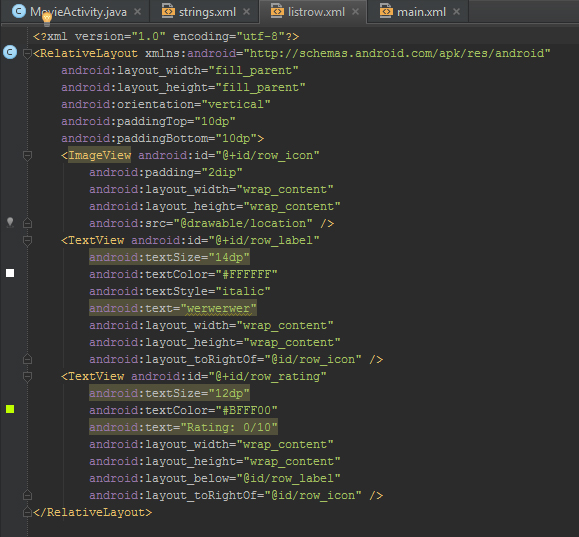
**Task 1**

Fragments are located within an activity, and multiple fragments can be layered over top of each other (for example, if you used a transparent background you could interact with different fragments), whereas an activity opens up a new window. Fragments facilitate modular UI's, and allow the use of a back button within an activity.

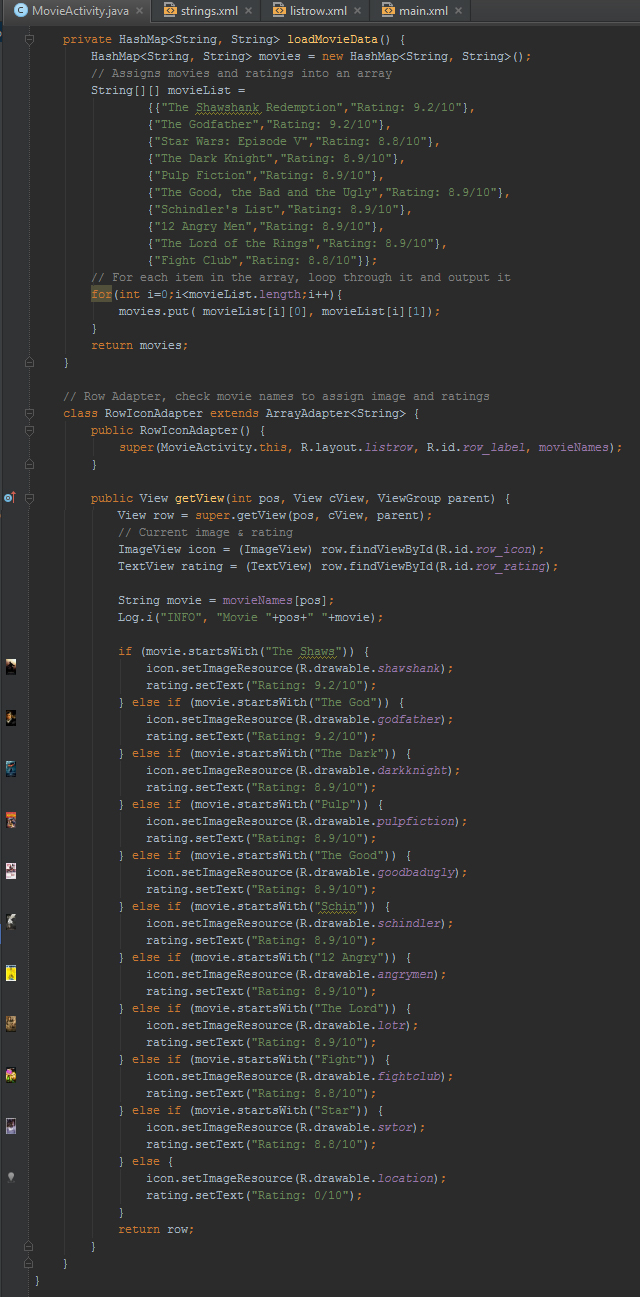
FragmentManager manages messages (FragmentTransactions) between fragments. Examples of these transactions are adding and removing fragments, but also the ability to work with the back button and the backstack. In an activity, the back button would return you to a previous activity, however with fragments, it returns you to a previous fragment state.

**Task 2**

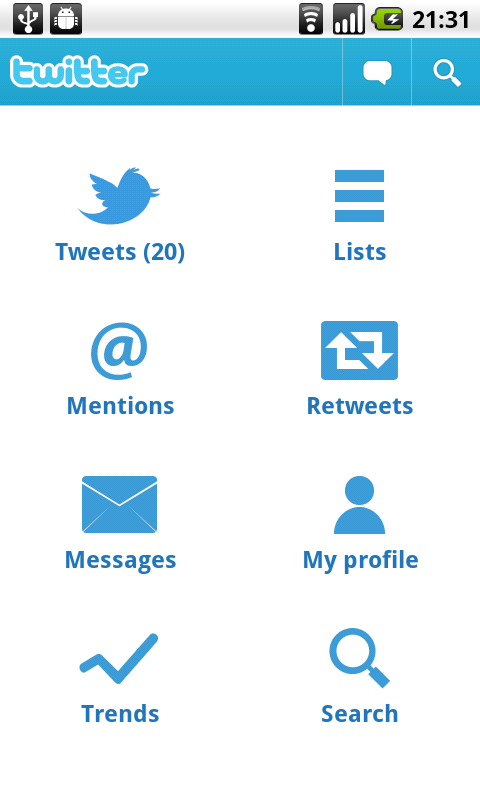
main.xml - Screenshot listrow.xml - Updated to show rating



MovieActivity - Row Adapter code



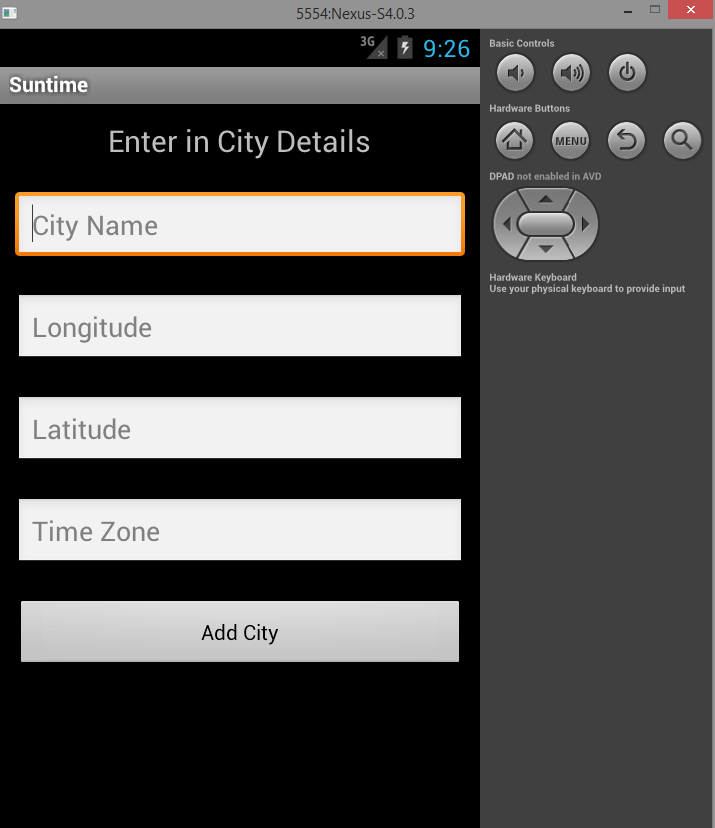
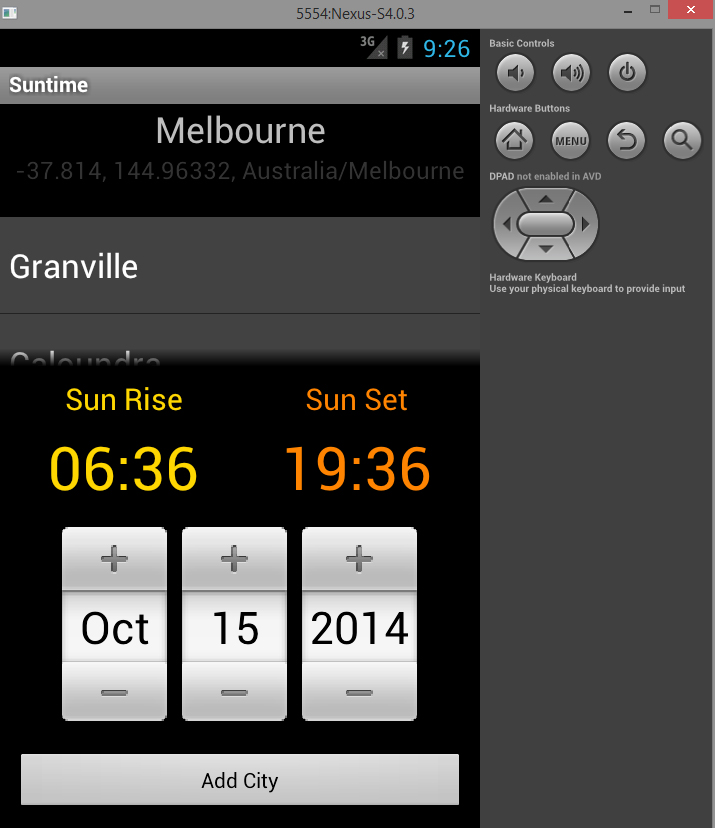
**Task 3**

Android recommends using an **app icon** that help identify brand and to be used as a home button, **view control** that either indicates the current activity someone is on or allows the user to quickly switch between activities, **action buttons** that are typically used for functions that are often used, and finally the **action overflow**, otherwise known as the sliding drawer which stores less used links.

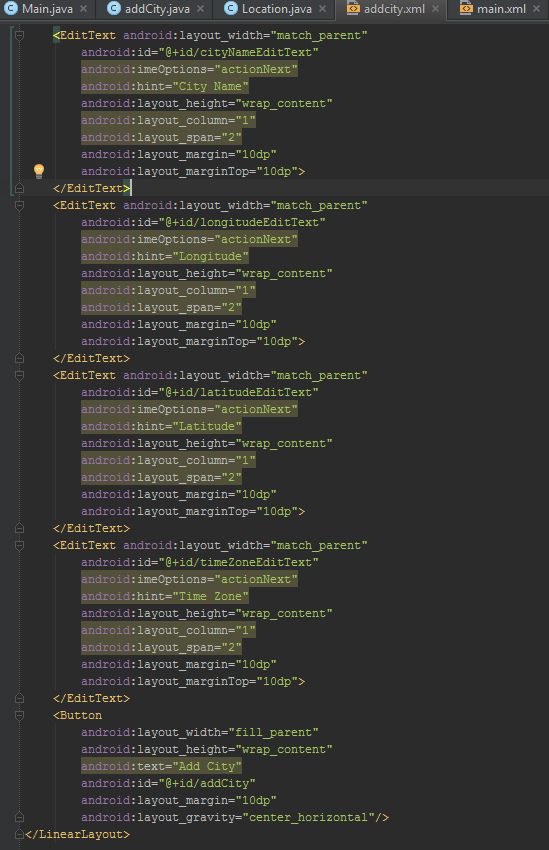
Dashboard patterns are no longer as popular as no one wants to have to click/touch through multiple screens to get to what they want. This can be attributed to a of lack of time and short attention spans . If an interface is intuitive, and once a user is introduced to the initial buttons in the UI, tasks can be simplified and kept purely in the action bar.

**Task 4**

main.xml - Screenshot addcity.xml - Screenshot



addcity.xml - Layout



Main.java - Adding text to file

