Weston Fiala

EE 590A Spring 2019 – Final Report

6/9/2019

The custom dice roller app contains three activities, and multiple supporting fragments and classes. In the following paragraphs I will give a high-level view of the different activities and major fragments.

The most important activity is MainActivity. MainActivity controls the bulk of the functionality and encases the communication between its containing fragments. It includes the actual rolling code, and a history of all previous rolls. SettingsActivity includes some simple settings that the user can interact with. AboutActivity has versioning information and legal information. The about and settings activities can be navigated to through a menu in the main activity.

In MainActivity, the supporting fragments are laid out as follows. At the top, a PageViewModel holds the data that needs to be preserved so long as the app is in memory. It includes many LiveData elements that different fragments in the activity observe and update. Below that is a ViewPager that handles the different tabs that can be swiped between, RollerFragment and HistoryFragment. RollerFragment contains a set of buttons that the user can click to begin a roll. When a button is clicked a dialog will show up that displays several dice images corresponding to the number of dice the user specified to be rolled. The dialog tells the user to shake their phone, and then to hold still. The fragment reads the accelerometer to determine if a user is shaking the phone and when they are holding still. While the user shakes the dice bounce around the screen. At the completion of the roll, a new dialog shows up that displays the roll results. At the same time as the results dialog showing up, a HistoryStamp is created and saved into the PageViewModel. The HistoryFragment consumes this HistoryStamp and displays the information about the roll. All new rolls are displayed at the top because newer rolls are more important to the user.

As described above, the only sensor that is used in this project is the accelerometer. When the shake dialog is open, the accelerometer readings are used to move the dice around as the user shakes, and to determine when the user is shaking. The dialog will remain open so long as the user keeps shaking, but once they stop shaking for a short time, the dice will begin slowing down and eventually stopping. At this point, the accelerometer readings are ignored until the next dice is rolled.

In addition to what is described above, I have published the app to the play store. I plan on finishing out the features that I did not get to during this project because they are the ones that will set my app apart from other apps. Check it out here: https://play.google.com/store/apps/details?id=com.fialasfiasco.customdiceroller