Assignment #3 – UnderOver in Android using Android Studio

Date assigned: Thursday, January 8, 2015
Date due: Friday, January 9, 2015
Points: 30

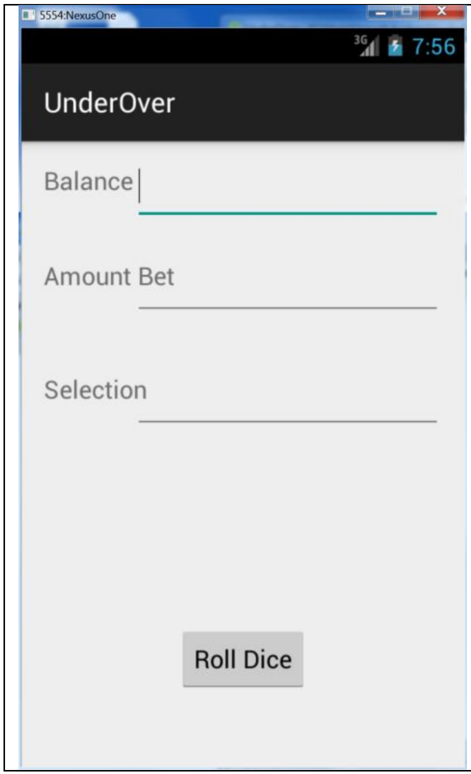
For this assignment, you will write an Android application that plays the game *UnderOver*. If you aren't familiar with the game, here are the basics. UnderOver is played by rolling a pair of dice and using the sum of the pair as the roll's value. If one die shows 4 and the other shows 5, you have rolled a 9. Before rolling the dice, you must place a bet and decide if you think the roll will be under 7, over 7, or exactly 7. The payout for a correct bet is:

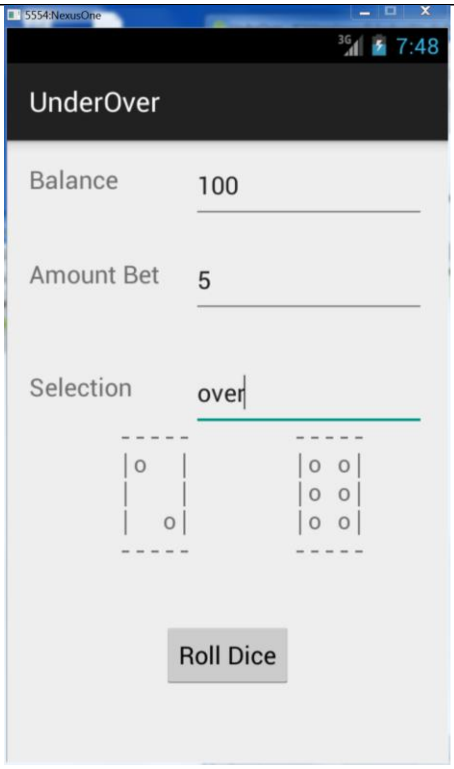
- Under 7 pays 1-1
- Over 7 pays 1-1
- Exactly 7 pays 4-1

So for instance, if you bet $1 on under and win, then you gain a dollar as well as getting your dollar back. If you bet on exactly 7 and win, you make $4.

We will see how reusable your Dice code is from the previous assignment!!! ☺

Here is a screenshot of how your game is to work. For this assignment, I will test your app on a Nexus One 480x800; hdpi using Android 4.2.2.





Balance 100

Amount Bet 5

Selection over

```
 -----        -----
|o   |       |o o|
|    |       |o o|
|   o|       |o o|
 -----        -----
```

Specifics:

1. After displaying the second screen, wait a few seconds and then update the results. So the second screen will become a Balance of 105, and the Amount Bet & Selection will be cleared and the Dice will be cleared.
2. Error checking:
   a. The user cannot enter an Amount Bet more than their Balance and that at least 1 is bet.
   b. The Selection must say **over**, **under**, or **seven**.
   c. If the Balance becomes 0, the game terminates.

For those of you interested in going the extra mile, you can use a drop down menu or radio buttons for the selection if you'd like. If you have other ways you want to fancy the UI up, go for it as long as the functionality works as described.

Goals for Assignment #3:

1. Design using good OO techniques
2. Better understand the Android Studio framework
3. Expand your use of the Java/Android API
4. Work with Android widgets TextView, EditView, Buttons, and callback functions
5. Write efficient Java code

Specifics:

1. Create a folder called your PUNetID, place a copy of UnderOver in your PUNetID folder, and drop PUNetID into CS260-01Drop on grace. Your code is to be written using the development tools specified in the syllabus and using the Java coding standards on the course Web page.

2. If you come to me with a question regarding your solution, I will have you load your project onto a machine in the CS lab. I will not look at your code on your computer or on paper as it just takes me too long to get at the problem. Further, I want you to bring in your lecture notes in case I want you to look up something. Remember, I'm not just a tell you the answer guy. Make sure you understand how to use the developer tools and that you can run your program in Eclipse.

3. If you want help with a compiler error, you must be able to tell me exactly what statement you put in your code that caused the error and be able to isolate the error. If you have typed in a bunch of code and have not tested your code as you've gone along, I'm not going to help you sort out the mess. You've been warned!!

4. IntelliJ keyboard shortcuts https://www.jetbrains.com/idea/help/keyboard-shortcuts-you-cannot-miss.html