

PROGRAMMING ASSIGNMENT #1

CIS 436 – MOBILE APPLICATION DEVELOPMENT – SUMMER 2017
UNIVERSITY OF MICHIGAN - DEARBORN
PROF. JOHN P. BAUGH

Due: Wednesday, June 14, 2017 at 11:59 p.m.

Points: _____ / 100

OBJECTIVES

- To create a basic user interface
- To create event listeners and handlers for UI components
- To create a basic test plan

INSTRUCTIONS

You are charged with the responsibility of creating a basic app, giving the user the ability to *play a slot machine*. It is assumed you will use good software engineering practice throughout, including the following:

- Use comments (useful comments), including your name, instructor's name, class you are in, and which project this is at the top of your Main Activity Java file
- Give appropriate ids to your widgets
 - Button1, button2, etc. will result in loss of points

Amount: Bank

You should ensure your program and assignment adhere to the following (In fact you could use the following, keeping the above instructions in mind also, as a check list):

- You will allow the user to enter any numeric value from **100 to 500 (inclusive)** into the editable text field
 - You must make sure only the **numeric** soft keyboard appears when the user focuses on the editable text field
 - You must also make sure that a Toast displays indicating an error if the values are not within the range 100 to 500 (inclusive.)
- Once the user enters the numeric value, they should be able to click a button labeled **Set Value**, which will cause both the button and the editable text field to become deactivated (i.e, you cannot type in the text field and/or click the button)
- There should also be a **New Game** button, which will reset the bank to 0, and disable the slot machine controls until the user selects a new amount and presses **Set Value**
- Playing the Game:
 - Each time the user presses the **Pull the Lever** button, \$5 is removed from his/her account
 - Three TextView widgets display a number from 1 through 5 (inclusive)
 - Hint: See the Java Class Framework's **Random** class
 - <https://docs.oracle.com/javase/7/docs/api/java/util/Random.html>
 - If **none** of the numbers match, the user doesn't get anything
 - If **two** of the numbers match, the user gets \$10
 - If **three** of the numbers match:
 - If the number that matches is less than 5, the user gets \$40
 - If the number is 5 through 8 (inclusive), the user gets \$100
 - If the number is 9 in all three slots, the user gets \$1000
- If the user's bank ever equals or exceeds \$1000, a Toast should pop up and tell the user they have cleared out the slot machine, and the program should then reset (the same as if the user had clicked **New Game**)
- If the user's bank ever equals 0, the game should display a Toast that tells the user they've lost all their money, and the game should reset (as if the user had clicked **New Game**)
- You must create a basic test plan for **each** of the above features and fill it in
 - Use the example text plan on Canvas
 - Use Microsoft Excel (preferred), or Apache OpenOffice Calc, or LibreOffice Calc for the workbook with the Test Plan
 - If you use OpenOffice Calc or LibreOffice Calc, please generate a PDF and turn that in – do not turn in the native formats of these open source programs

DELIVERABLES

Zip your entire Android project, with the **test plan stored at the top level of the project**

Upload the entire zip file to Canvas on or before the due date

