



## CS 442: Mobile Applications Development

### Assignment 1 – Tip/Split Calculator

Your assignment is to create a Tip and Split Calculator app as described below.

#### App Requirements

- The app will assist you in calculating your total dining cost with a selected tip added, and will allow you to split that total evenly for each person dining, specifying the amount owed by each person.
- Inputs:
  - User inputs are (1) the original bill total, (2) the desired tip percentage, and (3) the number of people to split the total with tip.
    1. The original bill total input is a positive whole or decimal number (no negative values allowed).
    2. The desired tip percentage is chosen via 4 radio buttons, with the values 12%, 15%, 18% and 20%.
    3. The number of people to split the bill is a positive whole number (no zero, negative or decimal values).
- After entering the bill total, selecting a tip percentage radio button should result in the tip amount and total with tip included to be calculated, and their data fields on the UI to be filled in (*these values should all be displayed out to 2 decimal places*).
  - *If the bill total field is empty, selecting a tip percentage should do nothing (and the selected tip percentage radio button should then be automatically un-checked).*
- After entering the Number of people value and tapping the “GO” button, the total per person and “overage” values should be calculated and their data fields on the UI should be filled in (*these values should be displayed out to 2 decimal places*).
  - *NOTE: the calculated amount per person should always be rounded UP to the nearest cent. For example, a calculated amount per person of 28.394 should be rounded UP to 28.40; a calculated amount per person of 15.999 should be rounded UP to 16.00. If you do not do this, the amount per person multiplied by the number of people can be less than the total with tip.*
- *The overage value is the difference between the amount per person multiplied by the number of people, and the bill total. For example: If the bill total was \$141.97, the amount per person with 5 people would be \$28.40. \$28.40 multiplied by the 5 people results in \$142.00 – 3 cents greater than the total with tip. The overage here is \$0.03 – this is expected behavior.*
- Pressing the “Clear” button will clear all fields – the Bill Total, Tip Amount, Total with Tip, Number of People, Total per Person, Overage, and any selected radio button should be unselected.
- The Constraint Layout must be used.
- You *will* need a separate Landscape layout for this assignment.

- The app must look and act consistently in either portrait or landscape orientation. No data content should be lost upon rotation (i.e., the Tip Amount, the Total with Tip, the Total per Person, and the Overage values should be maintained upon rotation).
- Concepts and practices discussed in class and presented in course materials must be followed.
- Please review the Academic Integrity and Plagiarism section of the course syllabus to ensure your complete understanding of the information there to avoid any violations. All work must be your own.

### Layout Design

- The below are examples of how the app should look (in both orientations):

- All 2 user input data fields should have a pale green background
- Tip percent values are in a Radio Group of Radio Buttons. Selection of a tip percentage automatically calculates the Tip Amount and Total with Tip values.
- Output fields (read-only) should have a pale grey background
- Label fields do not have a background color.
- Tapping the GO button will calculate the Total per Person and Overage values.
- Tapping the CLEAR button will clear all text fields and radio buttons

- A significant goal of this assignment is to learn how to create layouts that incorporate a variety of Views. As such, you need to match the layout diagrams shown above as closely as possible.

### Assignment Assistance

Your course TA is available to assist you with your assignment if needed. Questions on assignment requirements and course concepts can be sent to the instructor.

### Submissions & Grading

- Submissions must consist of your zipped project folder. For submission - before zipping your project file you must select menu option **Build → Clean Project** to insure your zipped submission is small enough to submit properly.
- Submissions must reflect the concepts and practices we cover in class, and the requirements specified in this document.
- Please review the Academic Integrity and Plagiarism section of the syllabus before, during, and after working on this assignment. All work must be your own.
- Late submissions will be penalized by 10% per week late. (i.e., from one second late to 1 week late: 10% penalty, from one week late to 2 weeks late: 20% penalty, etc.). **NO SUBMISSIONS CAN BE MADE FOR THIS ASSIGNMENT BEYOND 2 WEEKS LATE.**
- The following are the key points that will be examined in Project when graded:
  - 45% UI Design and Appearance (in both Project and Landscape orientation)
  - 25% Input operation
  - 30% Calculations & Results Displays
- Grading will be performed with the following SDK details:
  - Project Compile & Target SDK Version: 30
  - Project Minimum SDK Version: 25

- Grading will be performed on emulator devices with the following characteristics:

Resolution	Details	Example Emulators to Use
1080 x 1920	With Playstore	Pixel, Pixel 2, Nexus 5, Nexus 5X
1080 x 2220 or 2280	With Playstore	Pixel 3a, Pixel 4

*If you do not understand anything in this handout, please ask.*

*Otherwise the assumption is that you understand the content.*

***Unsure? Ask!***