Course Project: A Calculator Application (15%)

Introduction

The calculator is one of the essential applications in any mobile device. In this project assignment, you will design, develop, and test a simple calculator.

Objectives

After completing this project 2, you will be able to:

- Write design specs for a mobile app
- Identify design challenges
- Prepare a test plan
- Build a calculator application
- Deliver on-time

App Description:

In this project, you will develop a Calculator App. The **minimum** calculator requirements are:

- supports addition, multiplication, subtraction and division.
- supports memory call, memory add, memory subtract and memory recall features.
- a separate Equal button.
- supports a decimal point (4-digit precision) and negative sign.
- You should identify the maximum number of digits that can be displayed in the TextView for results.

A full list of the requirements is outlined in the marking rubric table below.

Project Submission

Instructions

1. Submit two incremental versions of the project PDF report to LMS.
   a. The firs milestone would be the full layout of the calculator design in XML. You should copy the XML code in a Word file, add screenshots of the layout, and list the components used in the layout. Following that, convert the file into PDF and submit it on LMS.
   b. The second milestone is the basic function of the calculator of all 4 mathematical operation (addition, subtraction, multiplication, division). You should copy the Java/Kotlin code in a Word file, add screenshots of the layout, and list the testcase scenarios. Following that, convert the file into PDF and submit it on LMS.
2. Upload the final project report as a PDF along with the project workspace to LMS.
3. Demonstrate your app to your course instructor

Marking Rubric

Your mark will be determined by the level of functionality you achieve for your Simple Calculator Application created in Android, as outlined in the following marking rubric table.

Component	Marks	Description
Presentation/Documentations	5	Code must be clearly commented and documented. Full mark will be given for adequate commenting and professional presentation of deliverables.
Basics functionality	45	All bare basic requirements of a calculator are satisfied and basic functionality is working perfectly. Requirements for basic functionality include: • Support for "running" calculations, as usual: 1+33-18*3, showing the intermediate result and history of operations every time you hit the next operator. • Display text field and numeric buttons, "+/-", and decimal point • Basic operational buttons, including equal, clear, and backspace.
Usable calculator	35	Basic and complete calculator. In addition to above, includes: • Simple memory (store, recall, clear, add-to), the usual. • Excellent design, including clear feedback on button pushes, and status of operation. Could include sound, flashing, highlighting, color, etc. • Adapts/scales reasonably to changes in device screen orientation.
User Interface quality	15	Credit will be given for the inclusion of one or more extra features, such as: • Flash window that shows your name and a welcome message as it starts. • Full scientific calculator • Allow switch between decimal, hex, and octal, and binary. • Allow switch between normal, fixed point, and scientific notation. • "Settings" screen: can set decimals places, coloring options for negative values, etc. • Your own idea for a feature that is useful and interesting...
TOTAL	**100**	