Mobile Coursework Report

Giovanni Paolini
40276003@napier.ac.uk
Edinburgh Napier University - Mobile Applications Development (SET08114)

1 Introduction

This is a report about the Mobile Applications Development coursework. The scope of this application is to provide a user-friendly tool to make quick calculations. The app is similar in design to the default calculator provided by Android devices, in particular the calculator provided with the phone Huawei Honor 7.

Keywords – Coursework, Napier , 40276003, Calculator

2 Software Design

Screen Design The application is designed to fit dynamically to each screen size. Buttons and textboxes will adjust their size depending on the user's device.

Data Entry All the inputs from user are given through buttons, how to input data is very straightforward and most of users will get used to it easily.

Multiple Inputs The design allows the user to chain various calculations, until he gets the desired results.

Data Usage The application uses low memory to run and little CPU usage. It does not have any background activity.

Hardware The application will run on most of devices. Hardware requirement is low and even the older devices will handle it easily.

3 Design Methodology

The design methodology used is "Waterfall". The development of the application followed 4 sequential steps:

3.1 Analysis:
What should the application do and how to achieve the user needs

3.2 Design
How the application will be displayed on the user's screen and how to make the input user-friendly

3.3 Implementation
Actual writing of the code following the design

3.4 Testing
Try the application on different devices to check for compatibility problems, display issues and bugs. The devices tested are: **Huawei Honor 7,Huawei Honor 9, Xiaomi Redmi Note 3, Asus Zenfone 2.**
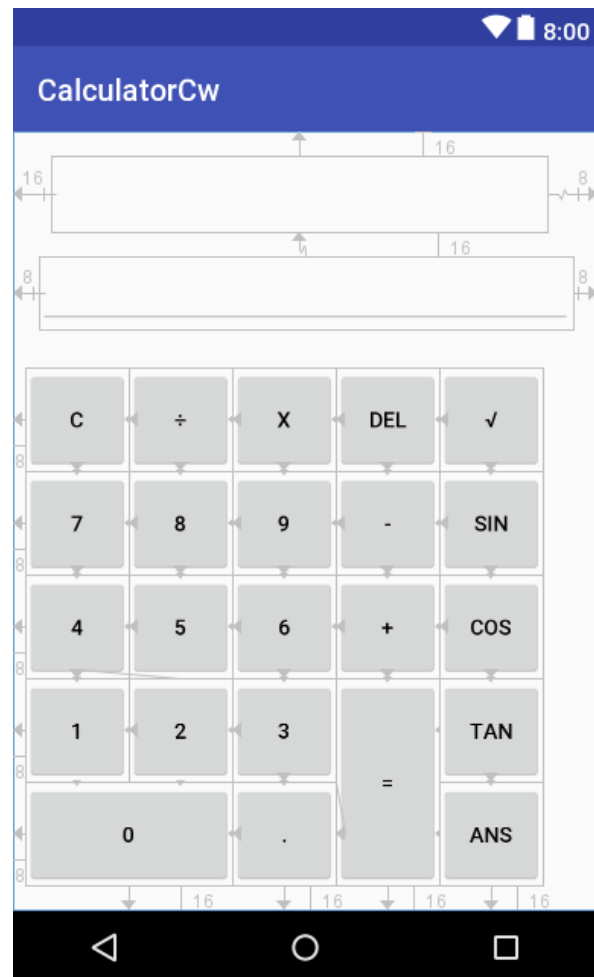

Figure 1: Application from the design menu, with all the constraints shown.

4 Application Implementation

The application was coded using the developing environment **Android Studio**. Each button has its own function and will behave differently on the current status of the operation (I.e. More than 2 inputs by the user).

Num Buttons: Used to input the desired digit.

DEL Button: Delete the last digit inputted.

C Button: Reset the calculation.

Operator Buttons: Choose the desired operation to make.

Dot button: Used to input decimal numbers.

Sqrt,Sin,Cos,Tan Buttons: These can only use the current value in the lower textbox, sequential operation can be obtained using the ans button after another operation.

ANS Button: Input the last operation's result.

Result button: Ends the input stream and gives the final result.

TextBoxs: The top textbox is programmed to show the data inserted in the application, the bottom one shows the current user input.

Multiple input handling: All the input buttons will check the status of the global Boolean variable newOperation to determine if the user got the desired result or wants to input more numbers. Then the function multipleInput() will handle the sequential operations.

5 Critical Evaluation

The application is good for a basic use, as it does not provide complex math functions. It provides all the tools that the default Android calculator provides in non-scientific mode plus the basic scientific functions. The design is essential and clean. Different tests were carried out by other users, which found the application free from errors, easy to use, and met their expectations for what a non-scientific calculator should achieve.

5.1 Possible Improvements

The calculator could be improved in many ways. From the feedback gathered the main function the users want is the possibility to switch to a scientific-calculator mode through a menu, button or screen rotation. Better layout could be achieved using custom buttons and background.
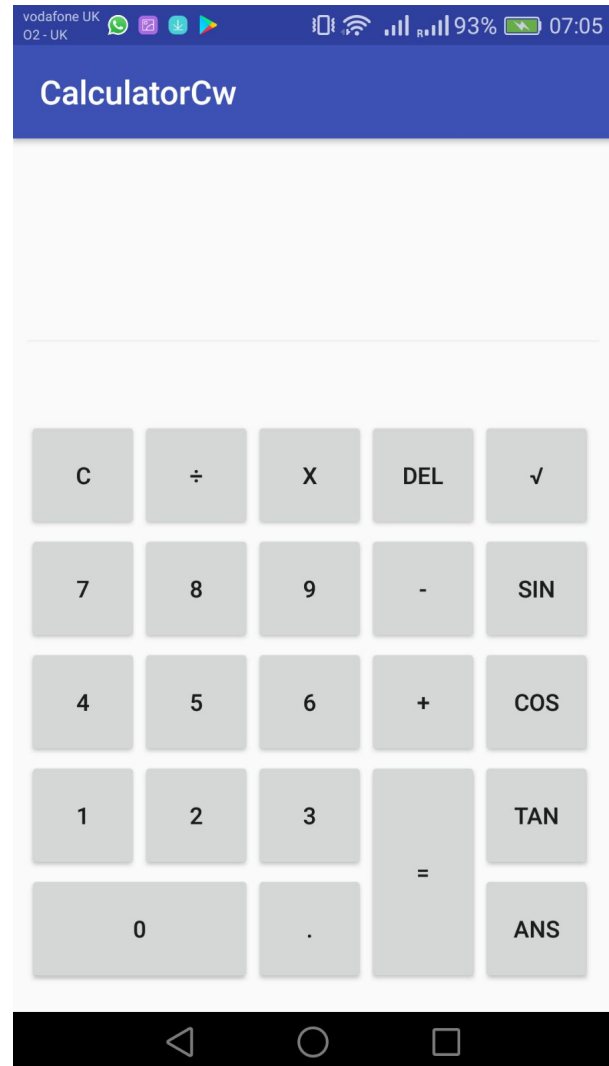

Figure 2: Application live on Huawei Honor 7 device.

6 Personal Evaluation

The project helped me improve in Mobile development and pushed me to learn new functions and more efficient ways to code. The part that took more time was the initial design, at first I didn't think at all the parts that I needed to handle. While coding I realized that it was more efficient to determine which operator the user chooses inside the Operator's button function, keep track of it with a status variable and then switch the operation on the results button. Other issues include different bugs in the *textbox display*, such as the operational button were replacing the previous input in the top textbox instead of attaching the sign and so on. From the information gathered after the tests made by different users I decided to add an **ANS Button**, the possibility to make multiple sequential operations and the basic scientific functions **Square root, Sin, Cos ,Tan**. The layout was not stretching properly on some devices so I had to fix it multiple times until I got a good display of everything. I feel like I could have improved the app a lot with further researching and understanding of the programming environment. I got lots of error along the way and this stopped me from trying to achieve harder tasks, but I am sure that with further practices in labs I will manage to have the necessary knowledge to achieve this.