Department of
Computer Science and Engineering

Title: Design and Development of a Calculator Application

Mobile Application Development

CSE 402

Green University of Bangladesh

1 Objective(s)

- To develop the Calculator application in Android device for performing basic arithmetic operations such as addition, subtraction etc.

2 Problem analysis

In this experiment, we'll learn to make a calculator APP. Let us plan for a simple and basic functions calculator as it is for learning purposes only. To build a calculator application, we need to design the xml and write code for java. Get method, set method needs to be implemented here along with arithmatic operations.Building this calculator will help us understand the layouts, buttons and action handling.

3 Implementation of Code for Addition and Subtraction Logic

```
1  package com.example.myapplication;
2
3  import androidx.appcompat.app.AppCompatActivity;
4
5  import android.os.Bundle;
6  import android.view.View;
7  import android.widget.Button;
8  import android.widget.EditText;
9
10 public class MainActivity extends AppCompatActivity {
11     Button button0, button1, button2, button3, button4, button5, button6,
12             button7, button8, button9, buttonadd, buttonSub, buttonDiv,
13             buttonMul, buttonPoint, buttonReset, buttonEqual;
14     EditText eT;
15     float ValueOne, ValueTwo, result;
16     boolean Addition, Subtract, Multiplication, Division;
17
18     @Override
19     protected void onCreate(Bundle savedInstanceState) {
20         super.onCreate(savedInstanceState);
21         setContentView(R.layout.activity_main);
22
23         //Referencing
24         button0 = (Button) findViewById(R.id.btnZero);
25         button1 = (Button) findViewById(R.id.btnOne);
26         button2 = (Button) findViewById(R.id.btnTwo);
27         button3 = (Button) findViewById(R.id.btnThree);
28         button4 = (Button) findViewById(R.id.btnFour);
29         button5 = (Button) findViewById(R.id.btnFive);
30         button6 = (Button) findViewById(R.id.btnSix);
31         button7 = (Button) findViewById(R.id.btnSeven);
32         button8 = (Button) findViewById(R.id.btnEight);
33         button9 = (Button) findViewById(R.id.btnNine);
34         buttonPoint = (Button) findViewById(R.id.btnPoint);
35         buttonadd = (Button) findViewById(R.id.btnadd);
36         buttonSub = (Button) findViewById(R.id.btnSub);
37         buttonMul = (Button) findViewById(R.id.btnMul);
38         buttonDiv = (Button) findViewById(R.id.btnDiv);
39         buttonReset = (Button) findViewById(R.id.btnReset);
40         buttonEqual = (Button) findViewById(R.id.btnEqual);
41
42         eT = (EditText) findViewById(R.id.eT);
43
```

```
44          //Event Listeners
45          button1.setOnClickListener(new View.OnClickListener() {
46              @Override
47              public void onClick(View v) {
48
49              //to set the editText with the button value 1
50              eT.setText(eT.getText() + "1");
51              }
52          });
53          button2.setOnClickListener(new View.OnClickListener() {
54              @Override
55              public void onClick(View v) {
56                  eT.setText(eT.getText() + "2");
57              }
58          });
59
60          button3.setOnClickListener(new View.OnClickListener() {
61              @Override
62              public void onClick(View v) {
63                  eT.setText(eT.getText() + "3");
64              }
65          });
66
67          button4.setOnClickListener(new View.OnClickListener() {
68              @Override
69              public void onClick(View v) {
70                  eT.setText(eT.getText() + "4");
71              }
72          });
73
74          button5.setOnClickListener(new View.OnClickListener() {
75              @Override
76              public void onClick(View v) {
77                  eT.setText(eT.getText() + "5");
78              }
79          });
80
81          button6.setOnClickListener(new View.OnClickListener() {
82              @Override
83              public void onClick(View v) {
84                  eT.setText(eT.getText() + "6");
85              }
86          });
87
88          button7.setOnClickListener(new View.OnClickListener() {
89              @Override
90              public void onClick(View v) {
91                  eT.setText(eT.getText() + "7");
92              }
93          });
94
95          button8.setOnClickListener(new View.OnClickListener() {
96              @Override
97              public void onClick(View v) {
98                  eT.setText(eT.getText() + "8");
99              }
100         });
101
```

```
        button9.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                eT.setText(eT.getText() + "9");
            }
        });
        button0.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                eT.setText(eT.getText() + "0");
            }
        });
        buttonReset.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                eT.setText("");
            }
        });

        buttonPoint.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                eT.setText(eT.getText() + ".");
            }
        });


        buttonadd.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

            //convert string to float and set UI Null to get next input
                ValueOne=Float.parseFloat(eT.getText().toString());
                Addition = true;
                eT.setText(null);
            }
        });
        buttonSub.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                ValueOne=Float.parseFloat(eT.getText().toString());
                Subtract = true;
                eT.setText(null);
            }
        });
        buttonMul.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                ValueOne = Float.parseFloat(eT.getText().toString());
                Multiplication = true;
                eT.setText(null);
            }
        });
        buttonDiv.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                ValueOne = Float.parseFloat(eT.getText().toString());
                Division = true;
```

```
                eT.setText(null);
            }
        });

        buttonEqual.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

            //to get the second input and convert that into float
                ValueTwo = Float.parseFloat(eT.getText().toString());

                if (Addition == true) {
                    result= ValueOne + ValueTwo ;
                    eT.setText(String.valueOf(result));
                    Addition = false;
                }

               if (Subtract == true) {
                    result= ValueOne - ValueTwo ;
                    eT.setText(String.valueOf(result));
                    Subtract = false;
                }
            }
        });

    }
}
```

4 Input/Output

Run the code and observe the output in the virtual device.

5 Discussion & Conclusion

From this experiment, we learnt about get and set methods to get text from and set text into UI. Basic calculating operations with the use of buttons is implemented here such that students can understand how to develop a complete application using event listeners only.

6 Lab Task (Please implement yourself and show the output to the instructor)

1. Design the User Interface of the Calculator App. There should be individual button for each number, operator.
2. Implement necessary logic for Multiplication and Division. Your app should also handle divide by zero exception.

6.1 Problem analysis

Create Buttons for each number (0-9) and operators (+, _, *, /, =). Do the necessary codes to implement the operations so that it shows correct results for every operation. Perform exception handling in case of necessity.

7 Lab Exercise (Submit as a report)

- Implement necessary logic for logarithm, exponential, power of x.

- Implement validation check for no user input when pressing operators.

8 Policy

Copying from internet, classmate, seniors, or from any other source is strongly prohibited. 100% marks will be *deducted* if any such copying is detected.