

Subject

Project Documentation For MOBI3002

Author: Mason Fraser
Date: 02 Oct 2019

Subject

Table of Contents

| | | |
|------|-----------------------------------|---|
| 1. | Project Overview | 3 |
| 2. | Project Requirements..... | 3 |
| 2.1. | Requirements | 3 |
| 3. | Design Plans | 4 |
| 3.1. | Screen Sizes | 4 |
| 3.2. | API Levels | 4 |
| 3.3. | Land Scape and Portrait | 5 |
| 3.4. | Graphic Images for functions..... | 6 |
| 3.5. | Custom Launcher | 6 |

Subject

1. Project Overview

Make Your Calculator Application support the 4 different Android screen densities....(for example, hdpi, mdpi, xhdpi, xxhdpi).

2. Project Requirements

Make a functional, customized calculator.

2.1. Requirements

Mix up the testing on emulators as follows:

test on 3 different screen sizes

2 different API-levels (23, 22, 21, ...any 2 different)

at least one is Landscape, at least one is Portrait.

You decide exactly which sizes and API levels to use, but it needs to be detectable per the next section.

...with the following features...

*Use graphics for your +, -, *, / buttons such that they are the proper size for the format.*

Create a graphic (to display with the calculator) with your name that changes density to match the screen (similar to my example).

Create your own mipmap/launcher.png file to match the different screen densities.

Subject

3. Design Plans

Here we in the following paragraphs I will describe how the application meets the above requirements.

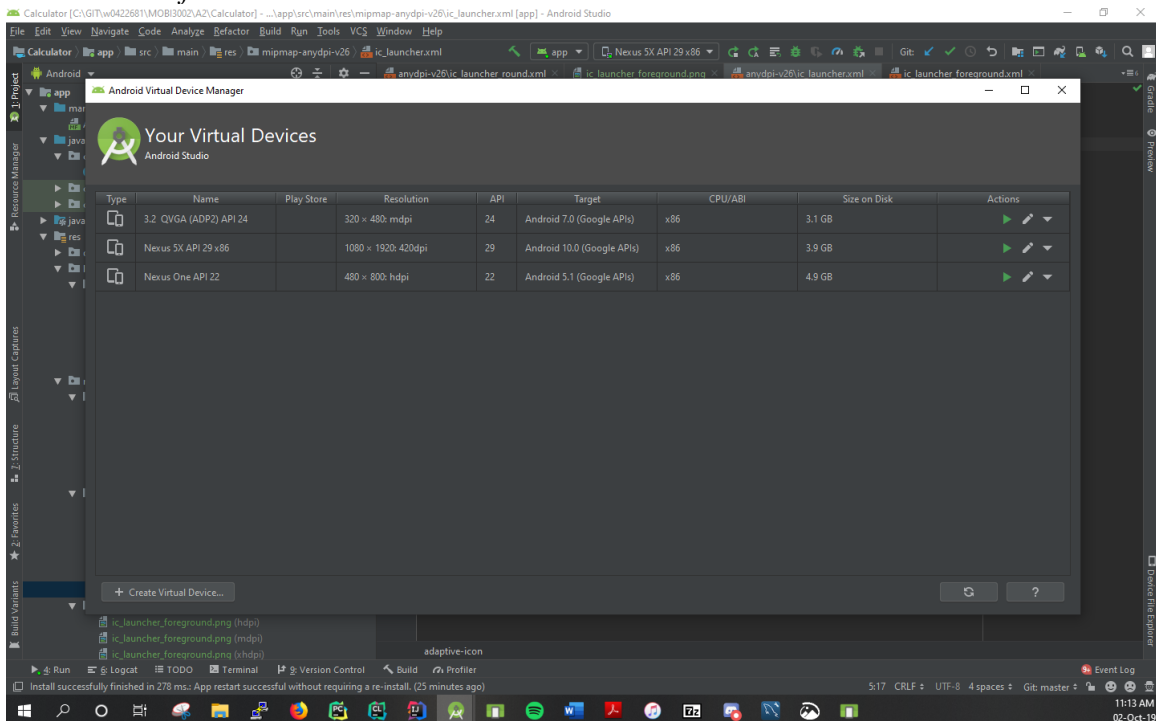
3.1. Screen Sizes

The application works with various screen sizes, all of different pixel density.

| Resolution |
|---------------------|
| 320 × 480: mdpi |
| 1080 × 1920: 420dpi |
| 480 × 800: hdpi |

3.2. API Levels

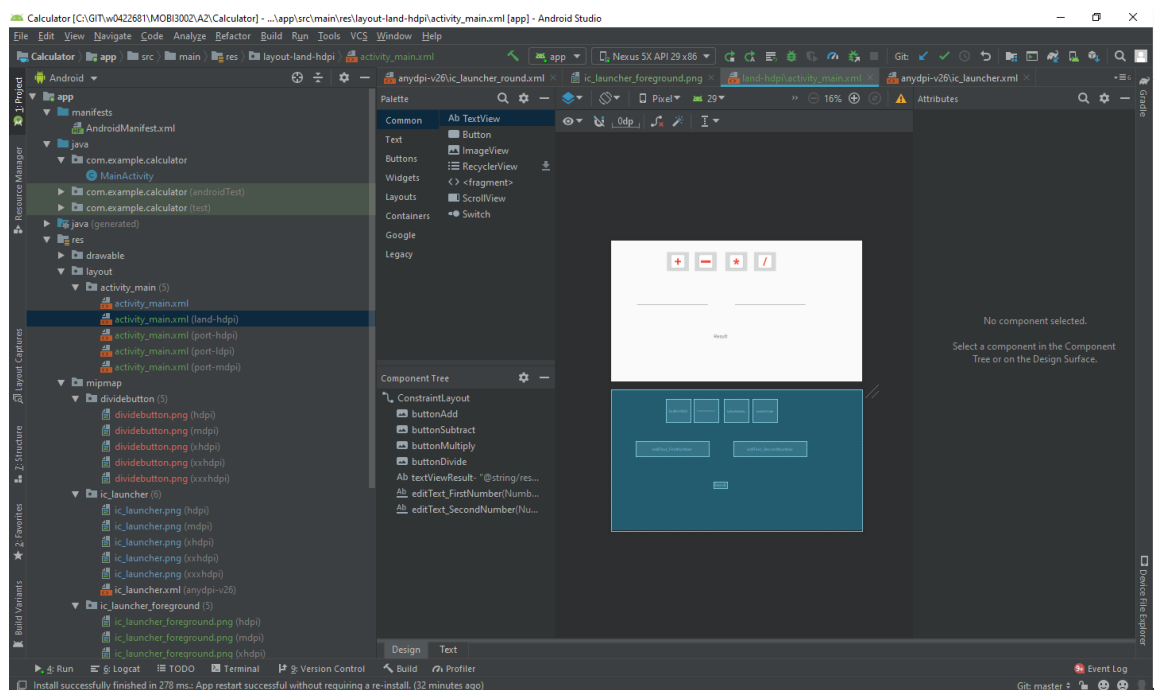
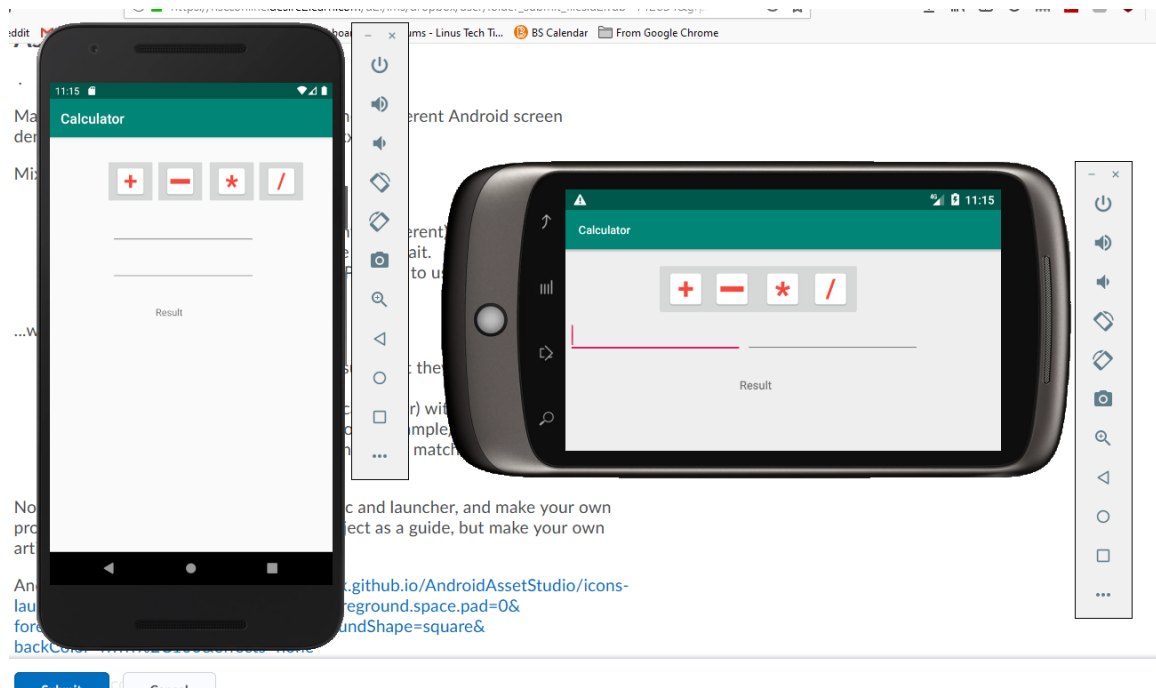
I tested the software on as new as API 29 and as old as API 22.



Subject

3.3. Land Scape and Portrait

Here the app works in landscape and portrait.



Subject

3.4. Graphic Images for functions

As opposed to using ascii, we have actual images for buttons.



```
9      import android.widget.TextView;
10
11      import java.nio.DoubleBuffer;
12
13
14      public class MainActivity extends AppCompatActivity {
15          @Override
16          protected void onCreate(Bundle savedInstanceState) {
17              super.onCreate(savedInstanceState);
18              setContentView(R.layout.activity_main);
19
20              final EditText firstValue = findViewById(R.id.editText_FirstNumber);
21              final EditText secondValue = findViewById(R.id.editText_SecondNumber);
22              ImageButton buttonAdd = findViewById(R.id.buttonAdd);
23              ImageButton buttonSubtract = findViewById(R.id.buttonSubtract);
24              ImageButton buttonMultiply = findViewById(R.id.buttonMultiply);
25              ImageButton buttonDivide = findViewById(R.id.buttonDivide);
26
27              final TextView result = findViewById(R.id.textViewResult);
28
29              buttonAdd.setOnClickListener((v) -> {
30                  int value1 = Integer.parseInt(firstValue.getText().toString());
31                  int value2 = Integer.parseInt(secondValue.getText().toString());
32                  int answer = value1 + value2;
33                  result.setText(Integer.toString(answer));
34              });
35
36              buttonSubtract.setOnClickListener((v) -> {
37                  int value1 = Integer.parseInt(firstValue.getText().toString());
38                  int value2 = Integer.parseInt(secondValue.getText().toString());
39                  int answer = value1 - value2;
40                  result.setText(Integer.toString(answer));
41              });
42
43              buttonMultiply.setOnClickListener((v) -> {
44                  int value1 = Integer.parseInt(firstValue.getText().toString());
45                  int value2 = Integer.parseInt(secondValue.getText().toString());
46                  int answer = value1 * value2;
47                  result.setText(Integer.toString(answer));
48              });
49
50              buttonDivide.setOnClickListener((v) -> {
51                  int value1 = Integer.parseInt(firstValue.getText().toString());
52                  int value2 = Integer.parseInt(secondValue.getText().toString());
53                  int answer = value1 / value2;
54                  result.setText(Integer.toString(answer));
55              });
56
57          }
58      }
```

3.5. Custom Launcher

The app has its own launcher image, Royalty Free!

Subject

