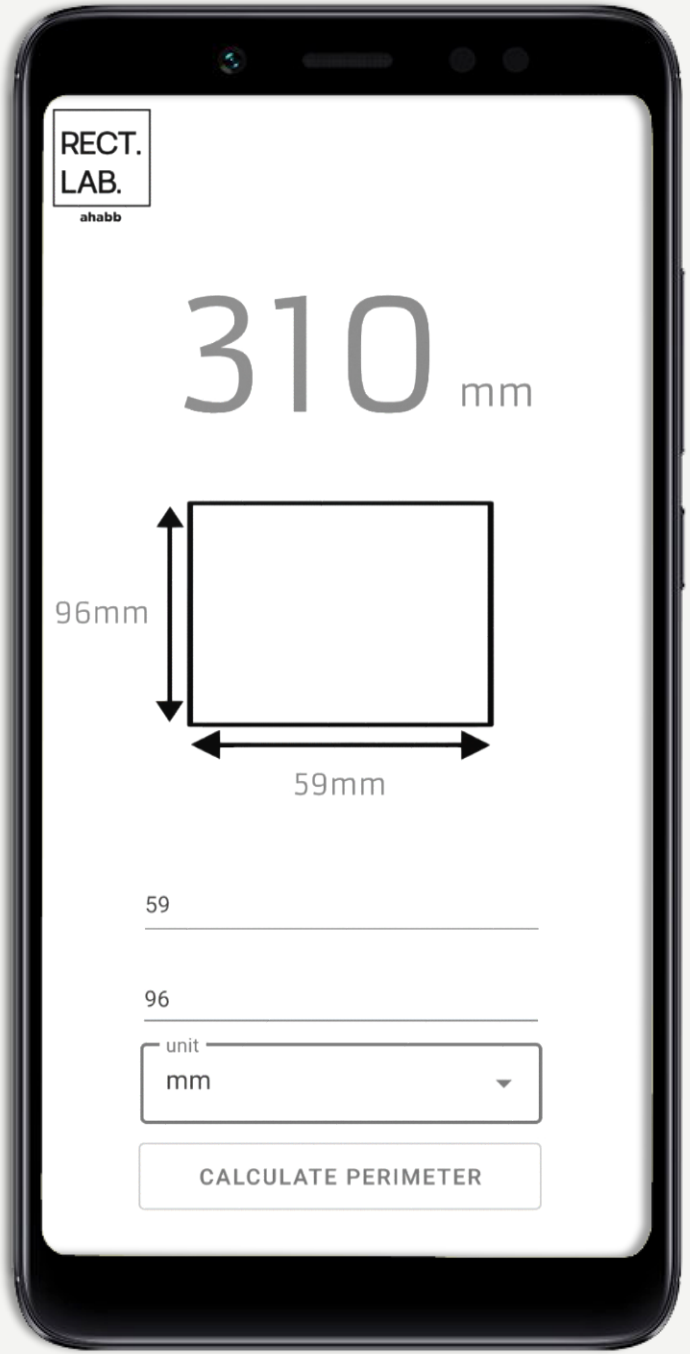
By
M. Ahabb Sheraz

Reg.
2021327

Semester Project
CS121 B



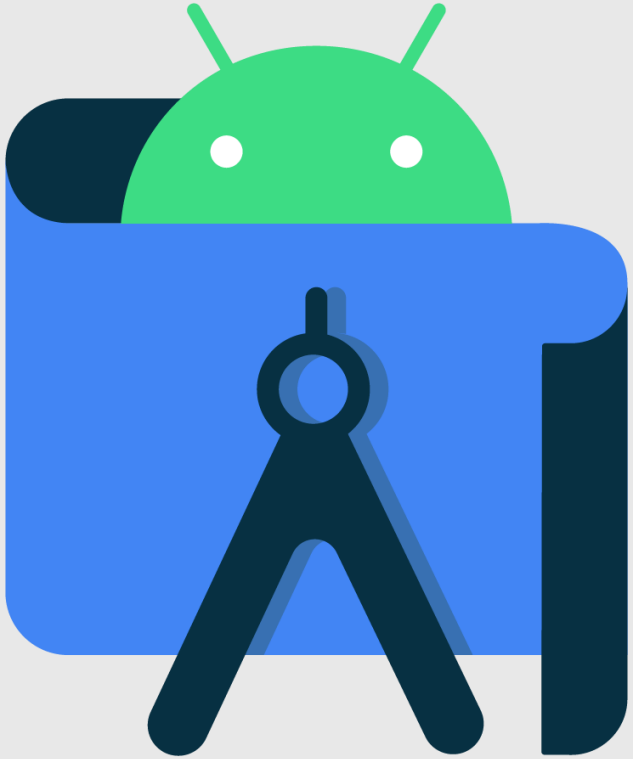
Introducing

RectLab

Purpose

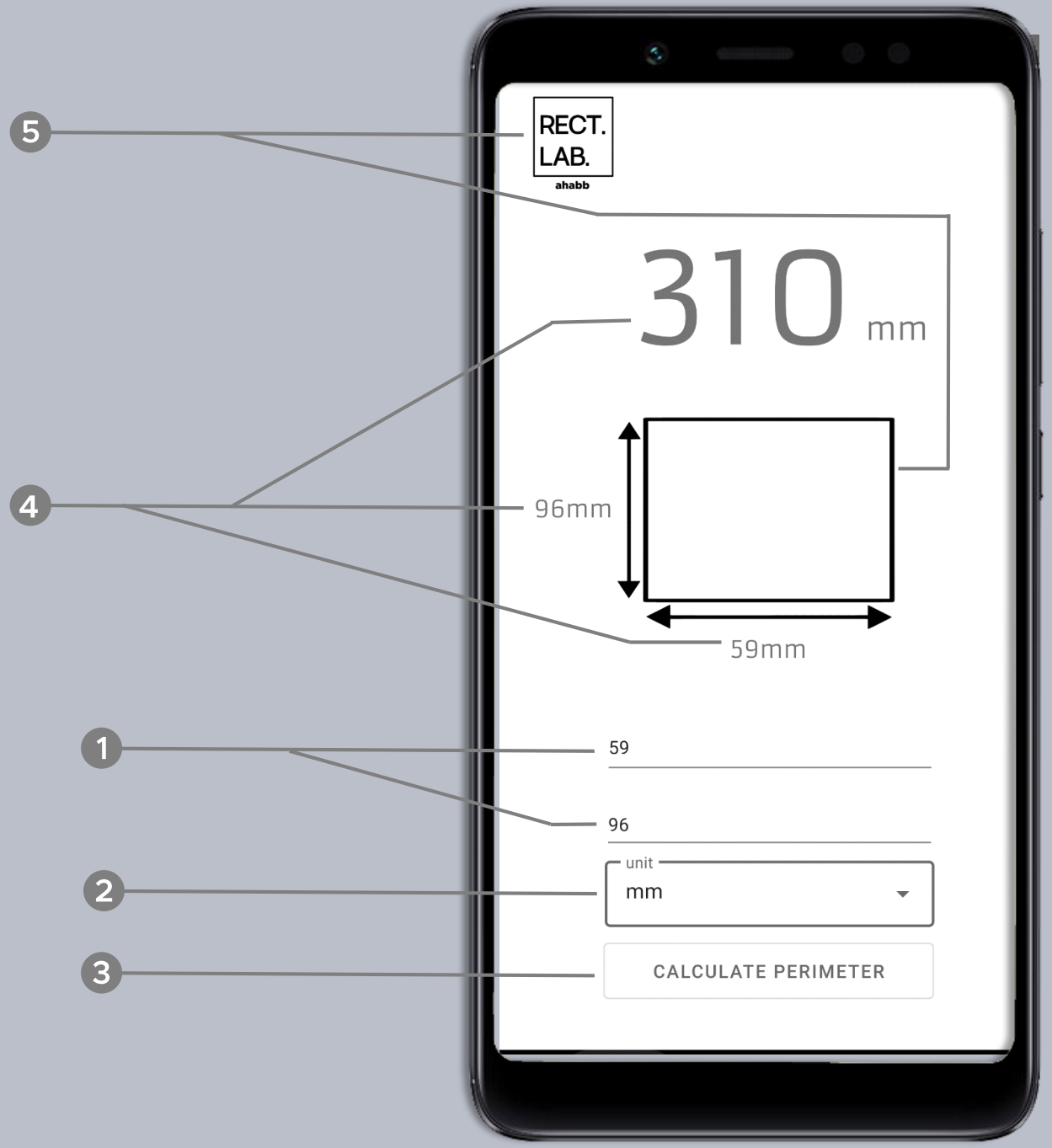
Rectangle Laboratory, RectLab for short, is a simple application that can calculate the perimeter of almost any rectangle. The naming convention was inspired by the much capable MATLAB.

Technologies

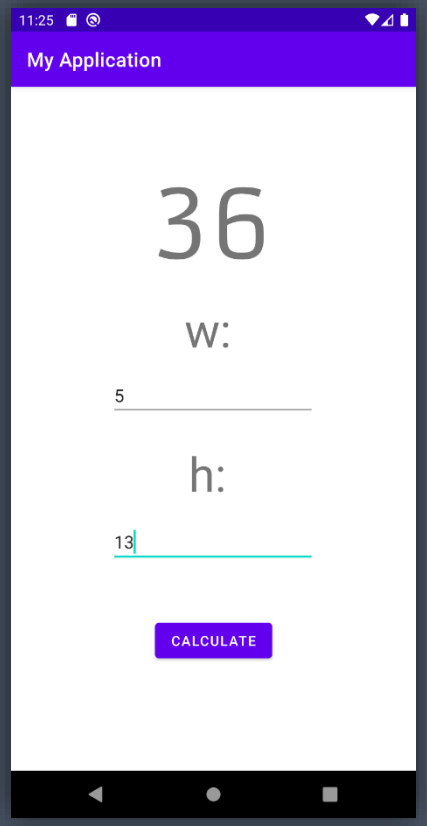


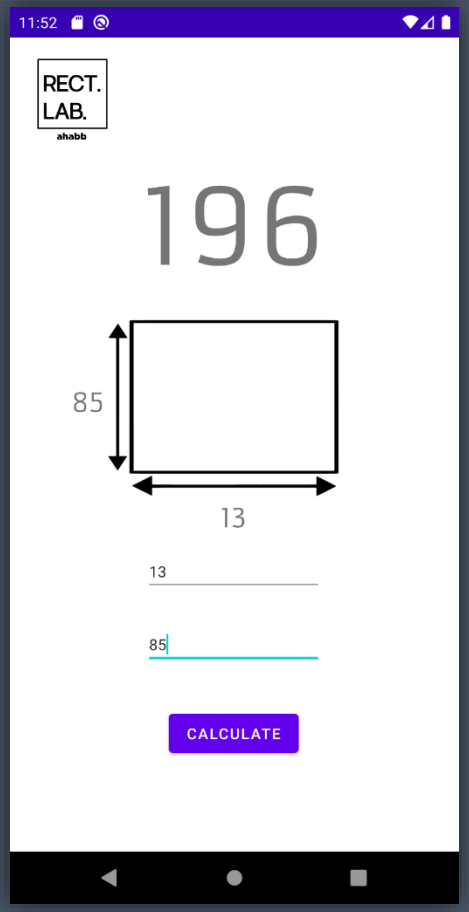
Features

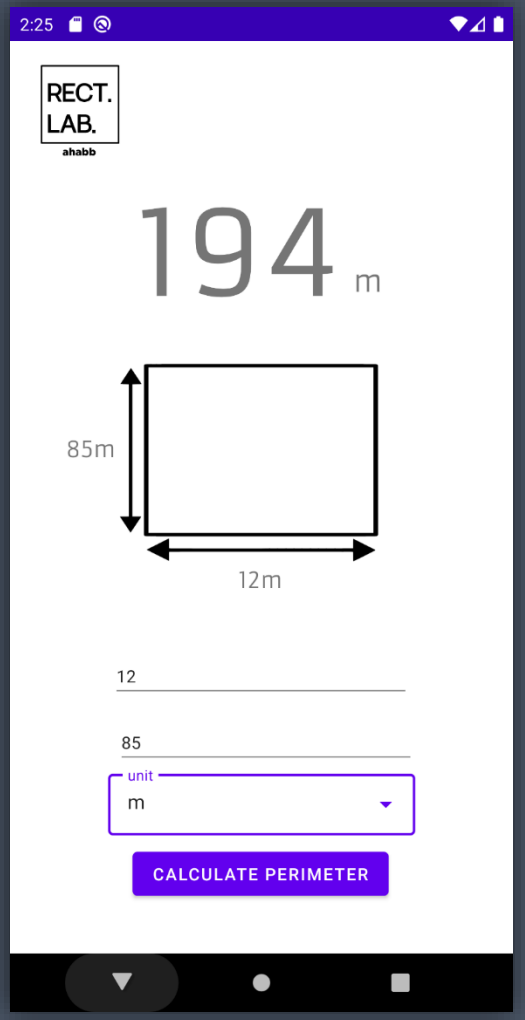
1. Number-only TextFields
2. Exposed Drop-Down Menu
3. Button
4. TextViews
5. Scalable Vector Graphics

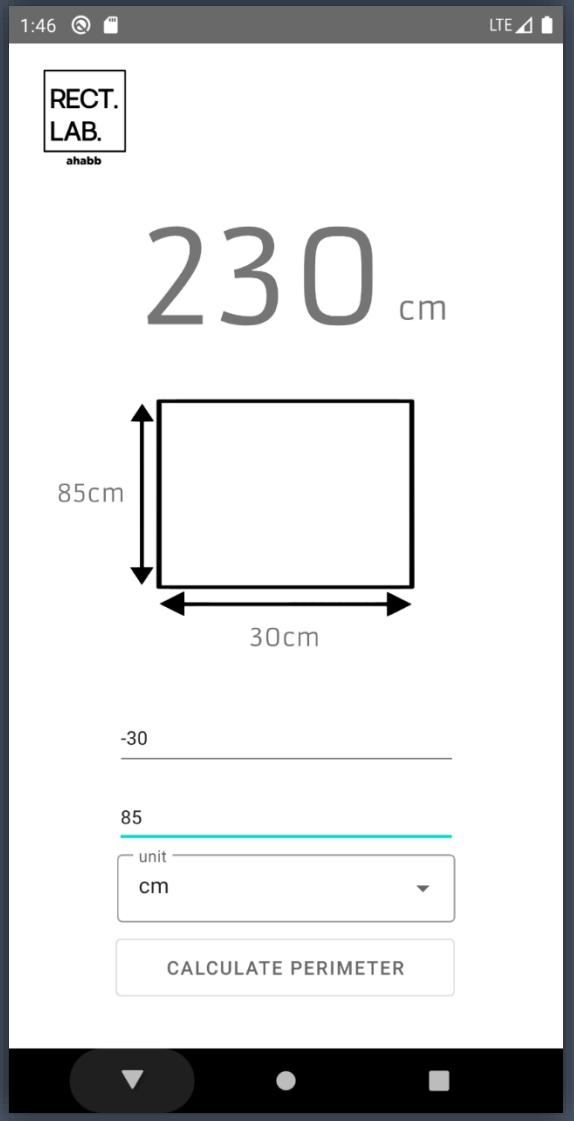


Prototypes and Demos









The Code

```
package com.example.rectlab

import ...

class MainActivity : AppCompatActivity() {
    //private lateinit var binding: ActivityMainBinding
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        val units = resources.getStringArray(R.array.units)
        val adapter = ArrayAdapter( context: this, R.layout.dropdown_item, units)
        val autocompleteTV = findViewById<AutoCompleteTextView>(R.id.autoCompleteTextView12)
        autocompleteTV.setAdapter(adapter)

        val calbutton = findViewById<Button>(R.id.button)
        val randtext = findViewById<TextView>(R.id.peritext)
        val unitext = findViewById<TextView>(R.id.unittext)
        val diswidth = findViewById<TextView>(R.id.w)
        val disheight = findViewById<TextView>(R.id.h)
        val height = findViewById<TextView>(R.id.editheight)
        val width = findViewById<TextView>(R.id.editwidth)


        calbutton.setOnClickListener { it: View!
            val h = height.text.toString().toInt()
            val w = width.text.toString().toInt()
            val h1 = abs(h)
            val w1 = abs(w)
            randtext.text = (2 * (h1 + w1) ).toString()
            diswidth.text = w1.toString() + autocompleteTV.getText().toString()
            disheight.text = h1.toString() + autocompleteTV.getText().toString()
            unitext.text = autocompleteTV.getText().toString()
```

Importing Essential Libraries

activity_main.xml | MainActivity.kt

```
1  package com.example.rectlab
2
3  import androidx.appcompat.app.AppCompatActivity
4  import android.os.Bundle
5  import android.widget.*
6  import androidx.core.content.ContentProviderCompat.requireContext
7  import androidx.core.splashscreen.SplashScreen.Companion.installSplashScreen
8  import kotlin.math.abs
9
```

MainActivity.kt

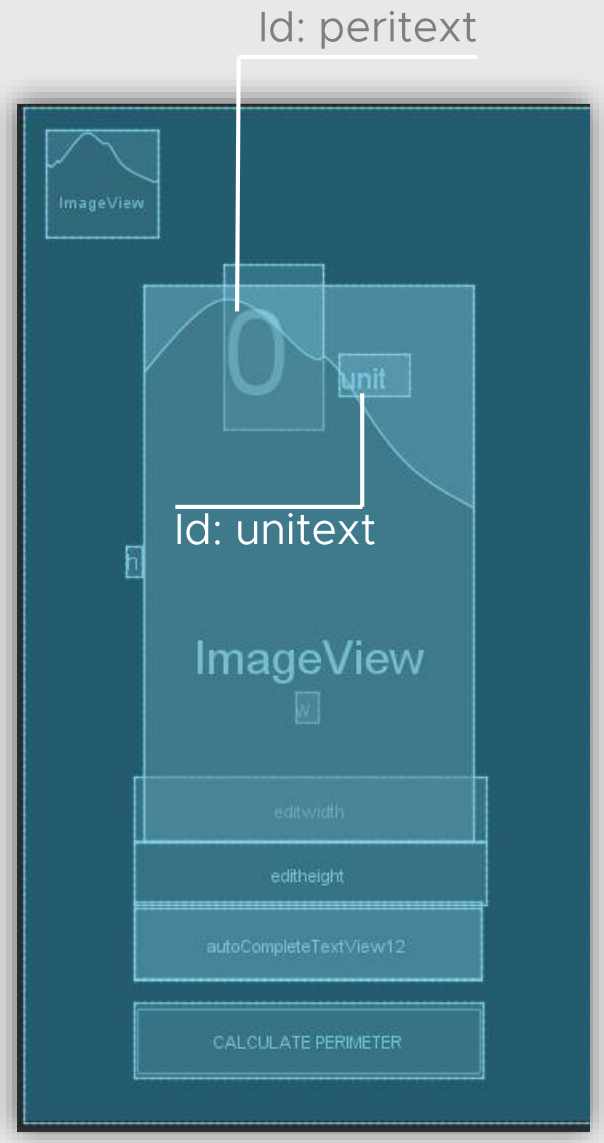
Declaring Variables

```kotlin
class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        //declaring variables in the main function
        val units = resources.getStringArray(R.array.units) //already declared a string array named units in strings.xml
        val adapter = ArrayAdapter( context: this, R.layout.dropdown_item, units)
        val autocompleteTV = findViewById<AutoCompleteTextView>(R.id.autoCompleteTextView12) //declared to change text on drop down menu
        autocompleteTV.setAdapter(adapter)

        val calbutton = findViewById<Button>(R.id.button) //declared a button and connect it to app button using id
        val peritext = findViewById<TextView>(R.id.peritext) //declared a textview and connect it to app textview using id
        val unitext = findViewById<TextView>(R.id.unittext) //declared a textview and connect it to app textview using id
        val diswidth = findViewById<TextView>(R.id.w) //declared a textview and connect it to app textview using id
        val disheight = findViewById<TextView>(R.id.h) //declared a textview and connect it to app textview using id
        val height = findViewById<TextView>(R.id.editheight) //declared a textfield and connect it to app textfield using id
        val width = findViewById<TextView>(R.id.editwidth) //declared a textfield and connect it to app textfield using id
```

MainActivity.kt



Button Call Function

```
30          calbutton.setOnClickListener {  it: View!
31              val h = height.text.toString().toInt() //height = findViewById<TextView>(R.id.editheight)
32              val w = width.text.toString().toInt()  //width = findViewById<TextView>(R.id.editwidth)
33              val h1 = abs(h) //makes sure to take absolute value of height
34              val w1 = abs(w) //makes sure to take absolute value of height
35              peritext.text = (2 * (h1 + w1) ).toString() //calculates perimeter and converts it to string to display it
36              diswidth.text = w1.toString() + autocompleteTV.getText().toString()
37              disheight.text = h1.toString() + autocompleteTV.getText().toString()
38              unitext.text = autocompleteTV.getText().toString()
39              //easter eggs
40              if (w >= 100000 || h >= 100000){
41                  Toast.makeText( context: this,  text: "OH NO! A REKTANGLE!!!", Toast.LENGTH_LONG).show()
42              }
43              if (diswidth.text == disheight.text){
44                  Toast.makeText( context: this,  text: "Rectangle is a Square.", Toast.LENGTH_LONG).show()
45              }
46          }
47      }
48  }
49
```

MainActivity.kt

Exposed Drop Menu Implementation

```kotlin
class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        //declaring variables in the main function
        val units = resources.getStringArray(R.array.units) //already declared a string ar
        val adapter = ArrayAdapter( context: this, R.layout.dropdown_item, units)
        val autocompleteTV = findViewById<AutoCompleteTextView>(R.id.autoCompleteTextView1
        autocompleteTV.setAdapter(adapter)

```

MainActivity.kt

activity_main.xml | MainActivity.kt | strings.xml

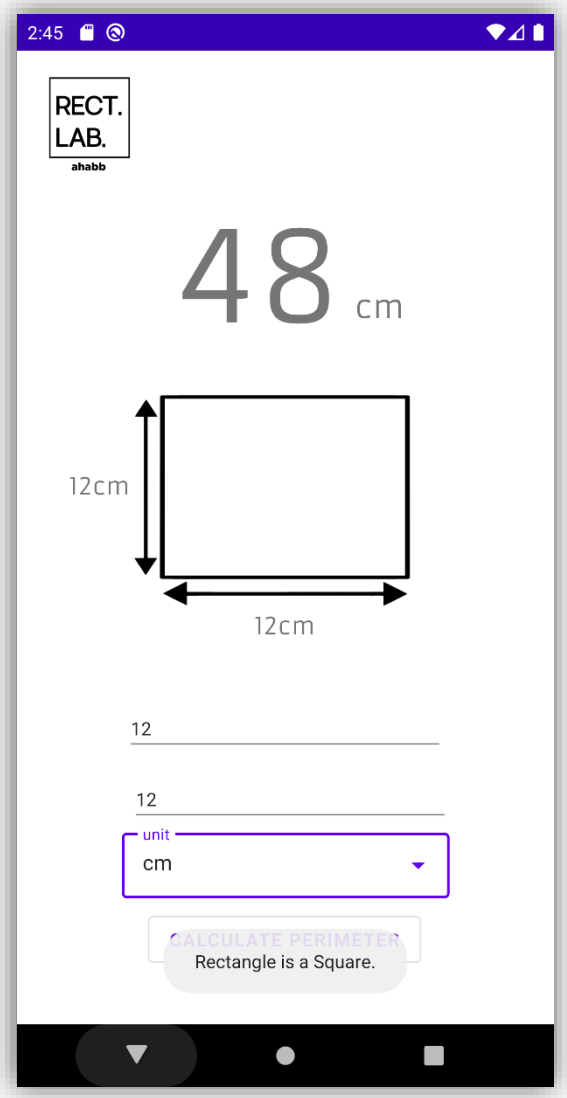
Edit translations for all locales in the translations editor.

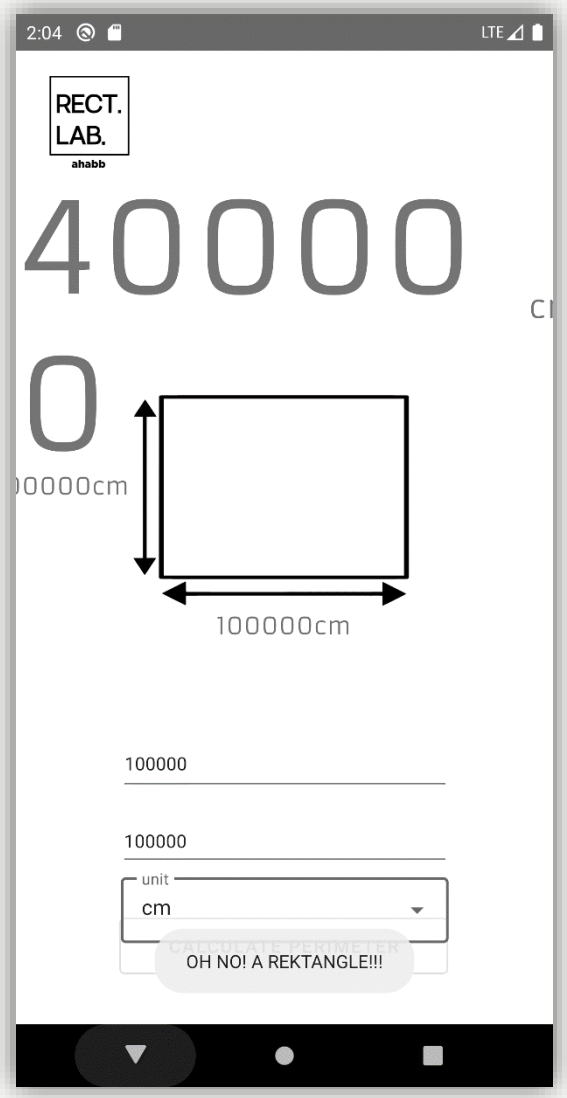
```xml
<resources>
    <string name="app_name">RectLab</string>

    <string-array name="units">
        <item>mm</item>
        <item>cm</item>
        <item>m</item>
        <item>in</item>
    </string-array>
    <string name="title_activity_splashscreen">splashscreen</string>
    <string name="dummy_button">Dummy Button</string>
    <string name="dummy_content">DUMMY\nCONTENT</string>
</resources>
```

strings.xml

Easter Eggs






```
    //easter eggs
    if (w >= 100000 || h >= 100000){
        Toast.makeText( context: this,  text: "OH NO! A REKTANGLE!!!", Toast.LENGTH_LONG).show()
    }
    if (diswidth.text == disheight.text){
        Toast.makeText( context: this,  text: "Rectangle is a Square.", Toast.LENGTH_LONG).show()
    }
}
```

MainActivity.kt

Conclusion

This project introduced me to the world of Android Application Development and gave me the chance to learn a lot of new things. Through this project, I was introduced to Kotlin for Android Studio. As for the project, I am happy with how it turned out to be. In the future, I plan to add more features such as the ability to convert units of both the height and width.

Thank

You!