

TK Teknologi Perangkat Lunak - Mobile Development

Lab 6

Disusun Oleh:

Usamah Nashirul Haq - 1606917954

7 Desember 2020

<https://github.com/usamah1707/learn-tktpl-1606917954/tree/lab-6>

A. Deskripsi Aplikasi

Aplikasi yang dibuat pada Lab 6 kali ini adalah aplikasi *counter* sederhana. Saya menggunakan hanya menggunakan satu *activity* pada aplikasi di lab kali ini. Cara bekerja aplikasi pada umumnya hanyalah melakukan penjumlahan, pengurangan, dan perkalian terhadap angka sebelumnya (*increment*, *decrement* dan *multiply*). Selanjutnya ada fitur untuk mereset jumlah angka agar kembali menjadi 0.



B. Metode

Aplikasi ini dibuat dengan menggunakan sebuah desain xml sederhana dan sebuah *main activity* yang sederhana pula. Yang spesial dari aplikasi ini adalah fitur yang menjadi *logic* dari aplikasi ini dibuat dalam bahasa native C++. Berikut adalah potongan kodenya.

```
class MainActivity : AppCompatActivity() {
    private lateinit var binding: ActivityMainBinding
    var indeks: Long = 0L
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = DataBindingUtil.setContentView(this,
            R.layout.activity_main)

        binding.addButton.setOnClickListener {
            indeks = Increment(indeks)
            binding.tvCounter.text = indeks.toString()
        }

        binding.minusButton.setOnClickListener {
            indeks = Decrement(indeks)
            binding.tvCounter.text = indeks.toString()
        }

        binding.multiplyButton.setOnClickListener {
            indeks = MultiplyBy2(indeks)
            binding.tvCounter.text = indeks.toString()
        }

        binding.resetButton.setOnClickListener {
            indeks = 0L
            binding.tvCounter.text = indeks.toString()
        }
    }

    external fun Increment(x: Long): Long
    external fun Decrement(x: Long): Long
    external fun MultiplyBy2(x: Long): Long
    companion object {
        // Used to load the 'native-lib' library on application startup.
        init {
            System.loadLibrary("native-lib")
        }
    }
}
```

```

#include <jni.h>
#include <string>

extern "C"
JNIEXPORT jlong JNICALL
Java_id_ac_ui_cs_mobileprogramming_usamahnashirulhaq_helloworld_MainActivity
_Increment(
    JNIEnv *env, jobject thiz, jlong x) {
    return ++x;
}

extern "C"
JNIEXPORT jlong JNICALL
Java_id_ac_ui_cs_mobileprogramming_usamahnashirulhaq_helloworld_MainActivity
_Decrement(
    JNIEnv *env, jobject thiz, jlong x) {
    return --x;
}

extern "C"
JNIEXPORT jlong JNICALL
Java_id_ac_ui_cs_mobileprogramming_usamahnashirulhaq_helloworld_MainActivity
_MultiplyBy2(
    JNIEnv *env, jobject thiz, jlong x) {
    return x*2;
}

```

Pemanfaatan native C++ membuat penggunaan memori yang digunakan akan lebih kecil, karena C++ tidak memiliki terlalu banyak *junk* atau sampah. C++ adalah superset dari C dan seharusnya mengkompilasi hampir semua program C, sehingga kita juga dapat menggunakan kembali perangkat lunak C di dalam Android dengan fungsi external.