

**UTS**

**Mata Kuliah Pemrograman Mobile 1**

**DOSEN PENGAMPU: Nova Agustina, M.Kom.**



**Disusun Oleh:**

**Adrian Rusmana**

**23552011228**

**TIF-RP23 CNS C**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**UNIVERSITAS TEKNOLOGI BANDUNG**

## **Essay**

1. Apa fungsi getExtra pada intent?
2. Bagaimana cara menerima data dari Activity pertama ke Activity kedua menggunakan intent?
3. Error apa yang terjadi jika file java salah menginisialisasi findViewById atau objek pada xml belum diinisialisasi?
4. Buat sebuah contoh program untuk menampilkan pesan error NullPointerException! Screenshot logcat-nya!
5. Kumpulkan dalam bentuk pdf di Elearning. Soal essay digabung dengan soal studi kasus cek point 7)

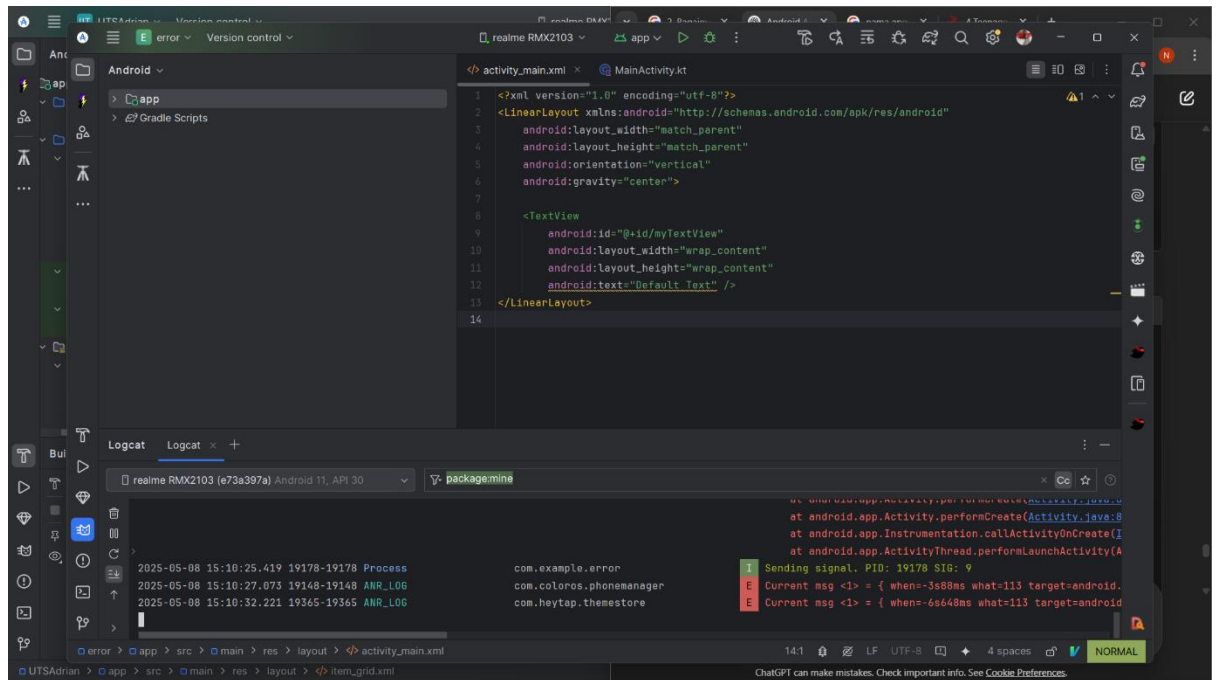
## **Studi Kasus**

1. Buatlah sebuah program sederhana yang terdiri dari 4 Activity menggunakan Android Native (Java + XML) yang terdiri dari:
  - a SplashScreen Activity
  - b Login Activity
  - c Register Activity
  - d Gallery Activity
2. Ketentuan: Silahkan membuat splashscreen dengan baik.
3. Pada Register Activity, minimal terdapat objek: TextView, EditText, Button, ImageView!
4. Tampilkan event Log, Toast dan Toast pada saat Button Register di klik.
5. Gallery Activity terdapat data yang ditampilkan dalam gridview
6. Upload project di Github.
7. **Jelaskan fungsi setiap baris source code pada file kotlin dan submit dalam bentuk pdf pada Elearning**
8. **Link Github harus tercantum pada pdf (point 7)**

Jawaban

Essay

1. `getExtra` pada intent berfungsi untuk mengambil data tambahan (extra data) yang dikirimkan melalui intent. Data ini biasanya disimpan dalam bentuk Bundle dan dapat diakses melalui kunci (key) tertentu.
2. Di Activity pertama kirim data dengan Intent menggunakan `putExtra` kemudian di activity kedua diterima menggunakan `getStringExtra`.
3. Akan terjadi error `NullPointerException`
- 4.



Studi kasus

Penjelasan file Kotlin

1. `GalleryActivity.kt`  
**`package com.example.utsadrian`**  
Menentukan package tempat file ini berada

**`import android.os.Bundle`**  
**`import android.widget.GridView`**  
**`import androidx.appcompat.app.AppCompatActivity`**

Mengimpor class yang diperlukan: Bundle, GridView, dan AppCompatActivity.

**`class GalleryActivity : AppCompatActivity() {`**

Membuat class `GalleryActivity` yang merupakan activity untuk menampilkan galeri gambar.

```
private val parts = arrayOf(
    R.drawable.download,
    R.drawable.rafael,
    R.drawable.michelangelo,
    R.drawable.donatello,
    R.drawable.reonardo
)
```

Array berisi ID gambar (res/drawable) yang akan ditampilkan di GridView.

```
override fun onCreate(savedInstanceState: Bundle?) {
```

Fungsi yang dipanggil saat activity pertama kali dibuat.

```
super.onCreate(savedInstanceState)
```

Memanggil method onCreate() dari superclass (AppCompatActivity).

```
setContentView(R.layout.gallery_activity)
```

Menetapkan layout XML yang akan digunakan di activity ini.

```
val gridView = findViewById<GridView>(R.id.gridView)
```

Menghubungkan variabel dengan GridView dari layout XML.

```
val adapter = ImageAdapter(this, parts)
```

Membuat objek adapter kustom (ImageAdapter) untuk menampilkan gambar

```
gridView.adapter = adapter
```

Menyambungkan adapter ke GridView.

## 2. ImageAdapter.kt

```
package com.example.utsadrian
```

Lokasi file

```
import android.content.Context
```

```
import android.view.LayoutInflater
```

```
import android.view.View
```

```
import android.view.ViewGroup
```

```
import android.widget.BaseAdapter
```

```
import android.widget.ImageView
```

Mengimpor komponen Android untuk adapter dan UI.

```
class ImageAdapter(private val context: Context, private val images: Array<Int>) :
    BaseAdapter() {
```

Membuat adapter kustom untuk GridView, menerima konteks dan array gambar.

```
override fun getCount(): Int = images.size
```

Mengembalikan jumlah item yang akan ditampilkan.

***override fun getItem(position: Int): Any = images[position]***

Mengembalikan item pada posisi tertentu.

***override fun getItemId(position: Int): Long = position.toLong()***

Mengembalikan ID item (berdasarkan posisi).

***override fun getView(position: Int, convertView: View?, parent: ViewGroup?): View {***

Method utama untuk membuat tampilan setiap item GridView.

***val view = convertView ?: LayoutInflater.from(context).inflate(R.layout.item\_grid, parent, false)***

Menggunakan view yang sudah ada jika tersedia, jika tidak maka inflate dari layout item\_grid.xml.

***val imageView = view.findViewById<ImageView>(R.id.imageView)***

Menemukan ImageView dalam item

***imageView.setImageResource(images[position])***

Menetapkan gambar dari array ke ImageView

***return view***

Mengembalikan view untuk ditampilkan.

### 3. LoginActivity.kt

***package com.example.utsadrian***

Lokasi file

***import android.content.Intent***

***import android.os.Bundle***

***import android.widget.\****

***import androidx.appcompat.app.AppCompatActivity***

Impor class untuk intent, UI, dan activity.

***class LoginActivity : AppCompatActivity() {***

activity login

***override fun onCreate(savedInstanceState: Bundle?) {***

***super.onCreate(savedInstanceState)***

***setContentView(R.layout.logina\_ctivity)***

Memuat layout login saat activity dimulai.

***val btnLogin = findViewById<Button>(R.id.btnLogin)***

***val tvRegister = findViewById<TextView>(R.id.tvRegister)***

Menghubungkan objek layout ke variabel Kotlin.

```
btnLogin.setOnClickListener {  
    startActivity(Intent(this, GalleryActivity::class.java))  
}
```

Saat tombol login ditekan, buka GalleryActivity.

```
tvRegister.setOnClickListener {  
    startActivity(Intent(this, RegisterActivity::class.java))  
}
```

Saat teks "register" ditekan, buka RegisterActivity.

#### 4. RegisterActivity.kt

```
package com.example.utsadrian
```

Lokasi file

```
import android.content.Intent  
import android.os.Bundle  
import android.util.Log  
import android.widget.*  
import androidx.appcompat.app.AppCompatActivity
```

Impor class yang dibutuhkan

```
class RegisterActivity : AppCompatActivity() {
```

Activity untuk registrasi pengguna.

```
    override fun onCreate(savedInstanceState: Bundle?) {  
        super.onCreate(savedInstanceState)  
        setContentView(R.layout.register_activity)
```

Menetapkan layout register.

```
        val etNama = findViewById<EditText>(R.id.etNama)  
        val btnRegister = findViewById<Button>(R.id.btnRegister)
```

Ambil referensi dari EditText dan Button

```
        btnRegister.setOnClickListener {  
            val nama = etNama.text.toString()
```

Ambil teks dari input nama saat tombol register ditekan.

```
            Log.d("Register", "Nama: $nama")  
            Toast.makeText(this, "Berhasil daftar sebagai $nama",  
                Toast.LENGTH_SHORT).show()
```

Tampilkan log dan toast untuk konfirmasi.

```
            val intent = Intent(this, GalleryActivity::class.java)  
            startActivity(intent)  
            finish()
```

Pindah ke GalleryActivity dan tutup activity ini.

## 5. SplashScreenActivity.kt

```
package com.example.utsadrian
```

Lokasi file

```
import android.content.Intent
```

```
import android.os.Bundle
```

```
import android.os.Handler
```

```
import androidx.appcompat.app.AppCompatActivity
```

Impor class untuk splash screen dan intent.

```
class SplashScreenActivity : AppCompatActivity() {
```

Activity pertama yang muncul saat aplikasi dibuka.

```
    override fun onCreate(savedInstanceState: Bundle?) {
```

```
        super.onCreate(savedInstanceState)
```

```
        setContentView(R.layout.activity_splash_screen)
```

Menampilkan layout splash screen.

```
        Handler().postDelayed({
```

```
            startActivity(Intent(this, LoginActivity::class.java))
```

```
            finish()
```

```
        }, 3000)
```

Menunda 3 detik, lalu berpindah ke LoginActivity dan menutup splash screen.

Link github : <https://github.com/adrianrsmna/uts>