

---

## Worksheet Responsi 07: Aplikasi Catatan Sederhana

Mata Kuliah	Bahasa Pemrograman II (Praktikum)
Kode MK	S1085
Topik	RESPONSI (Integrasi Materi Minggu 1-6)
Pertemuan	7 (Tujuh)
Dosen Pengampu	Afrig Aminuddin, S.Kom., M.Eng., Ph.D.
Alokasi Waktu	100 Menit
Nama Mahasiswa	Dorasti Riski Margaretha Simanjuntak
NIM	24.12.3156

---

### A. Tujuan Pembelajaran

Setelah menyelesaikan responsi ini, mahasiswa dinilai kemampuannya dalam:

- Mengintegrasikan Activity, Fragment, Intent (dengan data), dan Kontrol Input.
  - Mengimplementasikan RecyclerView lengkap dengan Adapter, ViewHolder, dan Data Class.
  - Menerapkan komponen Material Design (BottomNavigationView, FloatingActionButton) dan Drawable (Vector, Shape).
  - Mengelola alur aplikasi multi-halaman yang fungsional.
- 

### B. Skenario Soal

Anda ditugaskan membuat aplikasi pencatatan sederhana bernama **"MyNotes"** (atau "Catatanku").

Aplikasi ini memiliki 3 fungsionalitas utama:

1. **Menampilkan Daftar Catatan:** Halaman utama menampilkan semua catatan yang ada dalam RecyclerView.
2. **Menambah Catatan Baru:** Pengguna dapat berpindah ke halaman baru untuk menambah data catatan (Judul, Isi, Prioritas).
3. **Melihat Detail Catatan:** Pengguna dapat mengklik salah satu item di daftar untuk melihat detail lengkap catatan tersebut di halaman terpisah.

**PENTING:** Anda **tidak perlu** menggunakan database. Gunakan sebuah object Kotlin (singleton) untuk menyimpan ArrayList data catatan di memori secara statis, agar data tidak hilang saat berpindah Activity/Fragment.

---

### C. Kriteria Fungsionalitas (Wajib)

1. **Struktur Aplikasi (MainActivity)**
  - MainActivity adalah *host* (induk) utama.

- Gunakan BottomNavigationView (Week 6) dengan 2 menu: "Home" dan "About".
  - Gunakan FragmentContainerView untuk menampung Fragment.
  - Saat menu "Home" diklik, tampilkan HomeFragment.
  - Saat menu "About" diklik, tampilkan AboutFragment.
2. **HomeFragment (Halaman Daftar Catatan)**
- Tampilkan daftar catatan menggunakan RecyclerView (Week 5).
  - Tampilkan FloatingActionButton (FAB) (Week 6) di pojok kanan bawah.
  - Saat **FAB diklik**: Gunakan Explicit Intent untuk memulai AddNoteActivity (Week 3).
  - Saat **item di RecyclerView diklik**: Gunakan Explicit Intent untuk memulai DetailActivity DAN kirim data (Judul, Isi, Prioritas) dari item yang diklik menggunakan intent.putExtra() (Week 3).
3. **AboutFragment (Halaman Tentang)**
- Tampilkan TextView statis yang berisi Nama dan NIM Anda.
  - Gunakan *resource @string/* untuk semua teks (Week 2).
4. **AddNoteActivity (Halaman Tambah Catatan)**
- Gunakan ConstraintLayout atau LinearLayout (Week 2).
  - Memiliki 3 Kontrol Input (Week 4):
    - EditText untuk "Judul Catatan".
    - EditText untuk "Isi Catatan" (gunakan minLines="3").
    - RadioGroup dengan 3 RadioButton untuk "Prioritas" (misal: "Penting", "Normal", "Rendah").
  - Sebuah Button "Simpan".
  - Saat **tombol "Simpan" diklik**:
    - Ambil data dari semua kontrol input.
    - Tambahkan data tersebut ke ArrayList statis.
    - Tutup AddNoteActivity (panggil finish()) untuk kembali ke HomeFragment.
5. **DetailActivity (Halaman Detail Catatan)**
- Ambil data yang dikirim dari HomeFragment menggunakan intent.getStringExtra() (Week 3).
  - Tampilkan data "Judul", "Isi", dan "Prioritas" dalam TextView yang berbeda (Week 2).
6. **Data dan Visual (Wajib)**
- Buat Data Class (misal: Note.kt) untuk menampung data (Judul, Isi, Prioritas) (Week 5).
  - Buat object NoteRepository (atau companion object di MainActivity) untuk menyimpan ArrayList dari Note.
  - Buat NoteAdapter.kt untuk RecyclerView (Week 5).
  - Buat item\_row\_note.xml (layout untuk satu item RecyclerView).
  - Gunakan **Shape Drawable** (Week 6) sebagai android:background untuk item\_row\_note.xml (misal: sudut melengkung dan *stroke*).
  - Gunakan **Vector Drawable** (Week 6) dari *Asset Studio* untuk ikon di BottomNavigationView dan FloatingActionButton.
  - Gunakan strings.xml dan colors.xml untuk semua teks dan warna (Week 2).
-

#### D. Alur Pengerjaan (Disarankan)

1. Buat Project baru. Siapkan strings.xml dan colors.xml.
2. Buat Data Class (Note.kt) dan NoteRepository (untuk ArrayList statis).
3. Buat *resource*: Vector Drawable (Home, Info, Add) dan Shape Drawable (untuk item row).
4. Buat res/menu/bottom\_nav\_menu.xml.
5. Siapkan MainActivity: Edit activity\_main.xml (tambah FragmentContainerView & BottomNavigationView).
6. Buat HomeFragment dan AboutFragment (kosong).
7. Edit MainActivity.kt: Implementasikan setOnItemSelectedListener untuk BottomNavigationView (logika *switch fragment*).
8. Edit fragment\_home.xml: Tambahkan RecyclerView dan FloatingActionButton.
9. Buat item\_row\_note.xml dan terapkan Shape Drawable.
10. Buat NoteAdapter.kt.
11. Buat AddNoteActivity (layout XML dan file .kt). Implementasikan UI Kontrol Input.
12. Buat DetailActivity (layout XML dan file .kt). Implementasikan UI TextView.
13. Edit AddNoteActivity.kt: Tambahkan setOnClickListener untuk tombol "Simpan" (ambil data, simpan ke repo, finish()).
14. Edit HomeFragment.kt:
  - o Implementasikan setOnClickListener untuk FAB (start AddNoteActivity).
  - o Setup RecyclerView (Adapter, LayoutManager).
  - o Implementasikan *click listener* di Adapter (start DetailActivity dengan putExtra).
  - o Panggil adapter.notifyDataSetChanged() di dalam onResume() agar daftar otomatis ter-refresh setelah kembali dari AddNoteActivity.
15. Edit DetailActivity.kt: Ambil data (getStringExtra) dan tampilkan ke TextView.
16. Edit AboutFragment: Tambahkan TextView dengan Nama/NIM dari strings.xml.
17. Uji Coba (Testing).

---

#### E. Kriteria Penilaian (Ringkas)

Penilaian akan didasarkan pada skala 1-5 untuk 5 kriteria berikut:

1. **Struktur & Navigasi:** Fungsionalitas MainActivity, BottomNavigationView, dan Fragment (Home/About).
2. **Fungsi Tambah Data:** Fungsionalitas FAB -> AddNoteActivity (Kontrol Input) -> Logika Simpan ke *repository*.
3. **Fungsi Tampil Data:** Fungsionalitas RecyclerView (Adapter, ViewHolder, Data Class, LayoutManager) di HomeFragment.
4. **Fungsi Detail Data:** Fungsionalitas Intent dengan putExtra (klik item) dan getStringExtra (tampil di DetailActivity).
5. **Best Practice & Visual:** Penggunaan strings.xml, colors.xml, Vector Drawable, dan Shape Drawable.

---

#### F. Instruksi Pengumpulan

- *Commit* dan *Push* project Anda ke repository Git yang ditentukan dengan pesan commit: RESPONS: Selesai Mengerjakan Aplikasi MyNotes.
- Pastikan Anda melakukan *push* sebelum batas waktu berakhir.
- Lengkapi identitas dalam worksheet ini, kemudian kumpulkan ke Google Form yang disediakan (<https://forms.gle/yFY01btC1CCQExk89>).

## Lembar Jawab: Screenshot:

```
HomeFragment.kt  </> fragment_home.xml  AboutFragment.kt x  </> colors.xml  </> strings.xml  </> fragment_about.xml
1  package com.example.a3156_responsi
2
3  import android.os.Bundle
4  import android.view.LayoutInflater
5  import android.view.View
6  import android.view.ViewGroup
7  import android.widget.TextView
8  import androidx.fragment.app.Fragment
9
10 2 Usages
11 </> class AboutFragment : Fragment() {
12
13     override fun onCreateView(
14         inflater: LayoutInflater, container: ViewGroup?,
15         savedInstanceState: Bundle?
16     ): View? {
17         // Inflate the layout for this fragment
18         val view = inflater.inflate(resource = R.layout.fragment_about, root = container, attachToRoot = false)
19
20         // Get the TextView and set the text using a string resource
21         val textViewAbout: TextView = view.findViewById(id = R.id.textViewAbout)
22         textViewAbout.text = "note" // Use the string from strings.xml
23
24         return view
25     }
26 }
```

```
package com.example.a3156_responsi

import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import android.widget.Spinner
import android.widget.AdapterView
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity

class AddNoteActivity : AppCompatActivity() {

    private lateinit var edtTitle: EditText
    private lateinit var edtContent: EditText
    private lateinit var spinnerPriority: Spinner
    private lateinit var btnSaveNote: Button

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

        edtTitle = findViewById(R.id.edtTitle)
        edtContent = findViewById(R.id.edtContent)
        spinnerPriority = findViewById(R.id.spinnerPriority)
        btnSaveNote = findViewById(R.id.btnSaveNote)

        // Setup Spinner untuk prioritas
        val priorities = listOf("Tinggi", "Sedang", "Rendah")
        val adapter = ArrayAdapter(this,
```

```

        android.R.layout.simple_spinner_item, priorities)

        adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item)

        spinnerPriority.adapter = adapter

        btnSaveNote.setOnClickListener {
            val title = edtTitle.text.toString()
            val content = edtContent.text.toString()
            val priority = spinnerPriority.selectedItem.toString()

            if (title.isNotEmpty() && content.isNotEmpty()) {
                // Menambah catatan baru ke repository
                val newNote = Note(title, content, priority)
                NoteRepository.addNote(newNote)

                // Kembali ke halaman utama
                finish()
            } else {
                Toast.makeText(this, "Isi semua data catatan",
                    Toast.LENGTH_SHORT).show()
            }
        }
    }
}

```

```

package com.example.a3156_responsi

import android.os.Bundle
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity

class DetailNoteActivity : AppCompatActivity() {

    private lateinit var tvDetailTitle: TextView
    private lateinit var tvDetailContent: TextView
    private lateinit var tvDetailPriority: TextView

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

        // Inisialisasi TextView
        tvDetailTitle = findViewById(R.id.tvDetailTitle)
        tvDetailContent = findViewById(R.id.tvDetailContent)
        tvDetailPriority = findViewById(R.id.tvDetailPriority)

        // Mengambil index catatan yang dipilih melalui Intent
        val noteIndex = intent.getIntExtra("NOTE_INDEX", -1)

        if (noteIndex != -1) {
            // Mengambil catatan berdasarkan index
            val note = NoteRepository.getNoteAt(noteIndex)
            if (note != null) {
                // Menampilkan detail catatan jika note ditemukan
                tvDetailTitle.text = note.title
                tvDetailContent.text = note.content
                tvDetailPriority.text = note.priority
            } else {
                // Jika catatan tidak ditemukan, tampilkan pesan error
                tvDetailTitle.text = "Catatan tidak ditemukan"
            }
        }
    }
}

```

```

        tvDetailContent.text = "Catatan tidak tersedia"
        tvDetailPriority.text = "N/A"
    }
} else {
    // Jika index tidak valid, tampilkan pesan error
    tvDetailTitle.text = "Catatan tidak ditemukan"
    tvDetailContent.text = "Catatan tidak tersedia"
    tvDetailPriority.text = "N/A"
}
}
}
}

```

```

package com.example.a3156_responsi

import android.content.Intent
import android.os.Bundle
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import androidx.fragment.app.Fragment
import androidx.recyclerview.widget.LinearLayoutManager
import androidx.recyclerview.widget.RecyclerView
import com.google.android.material.floatingactionbutton.FloatingActionButton

class HomeFragment : Fragment() {

    private lateinit var recyclerView: RecyclerView
    private lateinit var noteAdapter: NoteAdapter
    private lateinit var fabAddNote: FloatingActionButton

    override fun onCreateView(
        inflater: LayoutInflater, container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View? {
        // Inflating the layout for this fragment
        val view = inflater.inflate(R.layout.fragment_home, container,
false)

        recyclerView = view.findViewById(R.id.recyclerViewNotes)
        fabAddNote = view.findViewById(R.id.fabAddNote)

        // Set up RecyclerView and Adapter
        recyclerView.layoutManager = LinearLayoutManager(requireContext())
        noteAdapter = NoteAdapter(NoteRepository.getNotes()) { position ->
            val intent = Intent(requireContext(),
DetailNoteActivity::class.java)
            val note = NoteRepository.getNoteAt(position)
            intent.putExtra("NOTE_TITLE", note?.title)
            intent.putExtra("NOTE_CONTENT", note?.content)
            intent.putExtra("NOTE_PRIORITY", note?.priority)
            startActivity(intent)
        }
        recyclerView.adapter = noteAdapter

        // FloatingActionButton click listener to add new note
        fabAddNote.setOnClickListener {
            val intent = Intent(requireContext(),
AddNoteActivity::class.java)
            startActivity(intent)
        }
    }
}

```

```

        return view
    }
}

```

```

package com.example.a3156_responsi

import android.os.Bundle
import androidx.appcompat.app.AppCompatActivity
import androidx.fragment.app.Fragment
import com.google.android.material.bottomnavigation.BottomNavigationView

class MainActivity : AppCompatActivity() {

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

        val bottomNav: BottomNavigationView =
            findViewById(R.id.bottomNavigationView)

        // Set default fragment to HomeFragment
        loadFragment(HomeFragment())

        bottomNav.setOnItemSelectedListener { item ->
            when (item.itemId) {
                R.id.menu_home -> {
                    loadFragment(HomeFragment())
                    true
                }
                R.id.menu_about -> {
                    loadFragment(AboutFragment())
                    true
                }
                else -> false
            }
        }

        // Helper function to load fragments
        private fun loadFragment(fragment: Fragment) {
            supportFragmentManager.beginTransaction()
                .replace(R.id.fragmentContainerView, fragment)
                .commit()
        }
    }
}

```

```

package com.example.a3156_responsi

data class Note(
    val title: String,
    val content: String,
    val priority: String
)

```

```

note.xml  AddNoteActivity.kt  </> activity_detail_note.xml  DetailNoteActivity.kt  NoteRepository.kt  NoteAdapter.kt  ×  AndroidManifest.xml
7  import androidx.recyclerview.widget.RecyclerView
8  import com.example.a3156_responsi.R // Ganti dengan package name aplikasi Anda
9
10  1 Usage
11  class NoteAdapter(
12      private val notes: List<Note>,
13      private val onClick: (Int) -> Unit
14  ) : RecyclerView.Adapter<NoteAdapter.NoteViewHolder>() {
15
16      override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): NoteViewHolder {
17          val view = LayoutInflater.from(context = parent.context).inflate(resource = R.layout.item_note, root = parent, attachToRoot = false)
18          return NoteViewHolder(itemView = view)
19      }
20
21      override fun onBindViewHolder(holder: NoteViewHolder, position: Int) {
22          val note = notes[position]
23          holder.title.text = note.title
24          holder.content.text = note.content
25          holder.priority.text = note.priority
26
27          holder.itemView.setOnClickListener { onClick(position) }
28      }
29
30      override fun getItemCount(): Int = notes.size
31
32      4 Usages
33      class NoteViewHolder(itemView: View) : RecyclerView.ViewHolder(itemView) {
34          1 Usage
35          val title: TextView = itemView.findViewById(id = R.id.tvTitle)
36          1 Usage
37          val content: TextView = itemView.findViewById(id = R.id.tvContent)
38          1 Usage
39          val priority: TextView = itemView.findViewById(id = R.id.tvPriority)
40      }
41  }

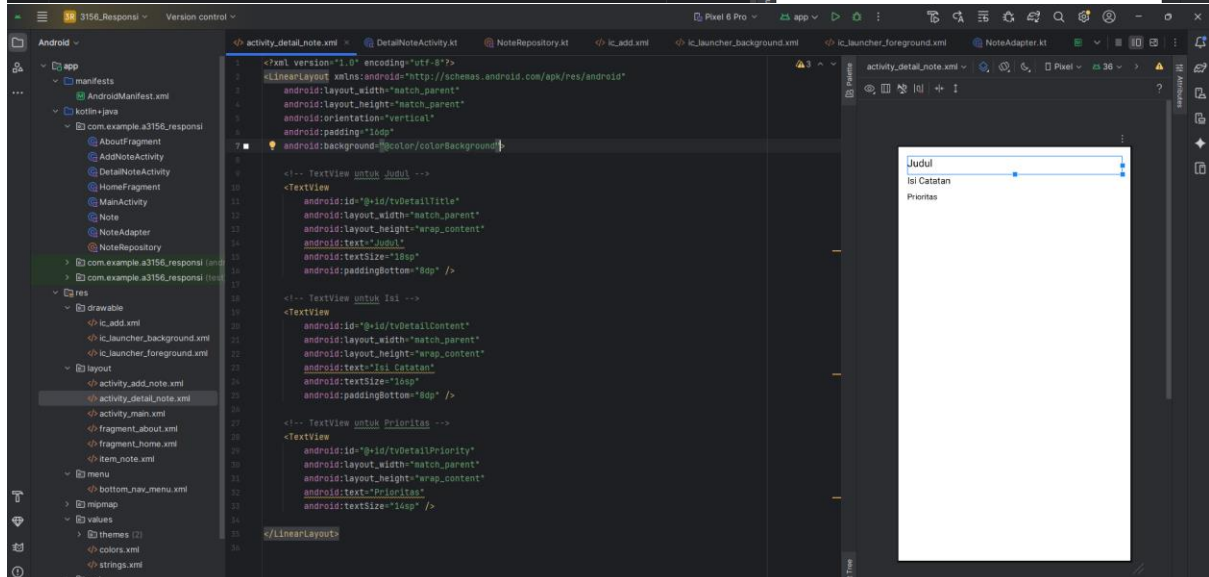
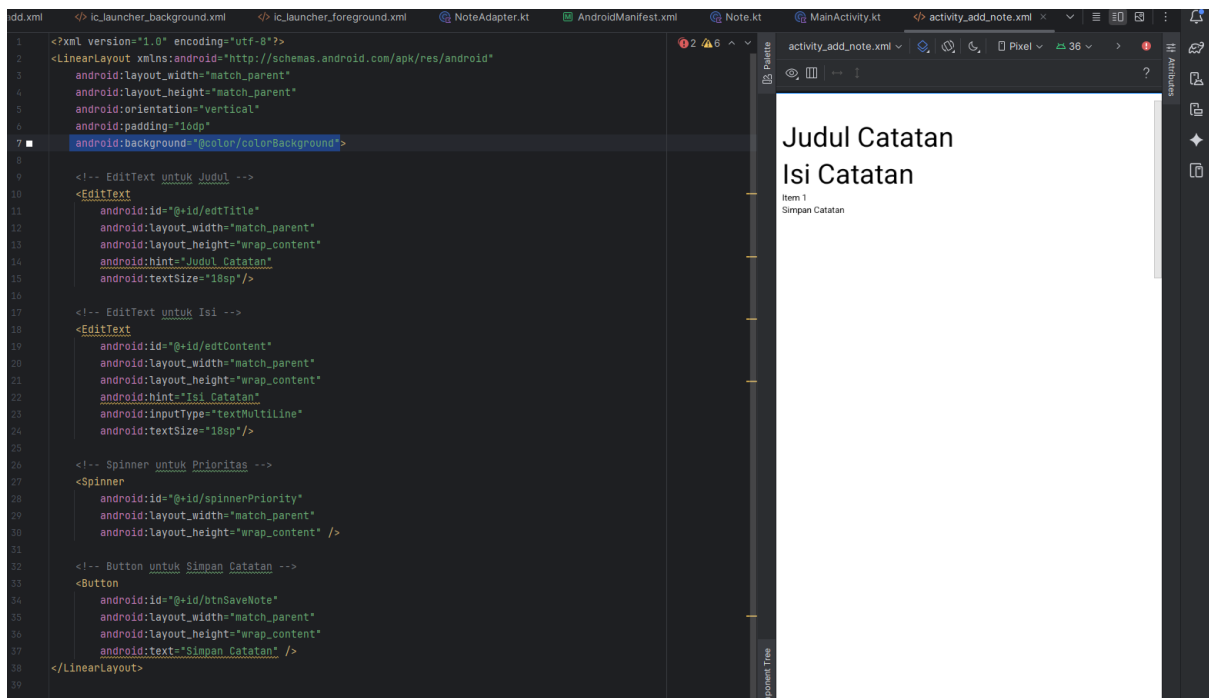
```

```

note.xml  AddNoteActivity.kt  </> activity_detail_note.xml  DetailNoteActivity.kt  NoteRepository.kt  ×
1  package com.example.a3156_responsi
2
3  2 Usages
4  object NoteRepository {
5      4 Usages
6      private val notes = mutableListOf<Note>()
7
8      // Menambahkan catatan baru
9      1 Usage
10     fun addNote(note: Note) {
11         notes.add(note)
12     }
13
14     // Mendapatkan semua catatan
15     fun getNotes(): List<Note> {
16         return notes
17     }
18
19     // Mengambil catatan berdasarkan index
20     1 Usage
21     fun getNoteAt(index: Int): Note? {
22         return if (index >= 0 && index < notes.size) {
23             notes[index] // Mengembalikan catatan sesuai dengan index
24         } else {
25             null // Mengembalikan null jika index tidak valid
26         }
27     }
28 }

```



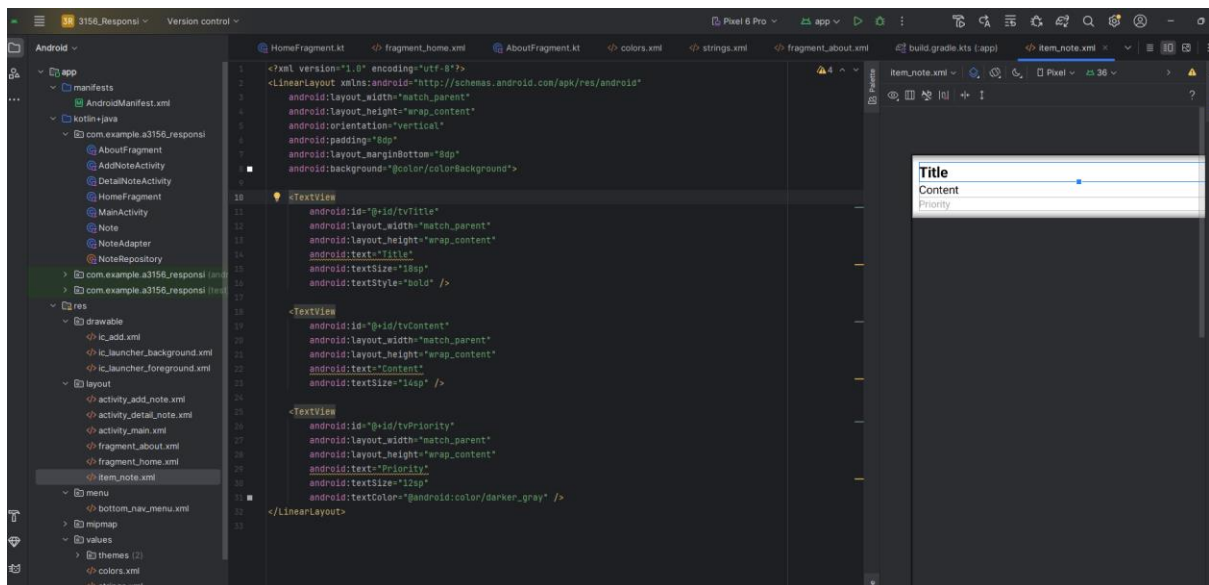
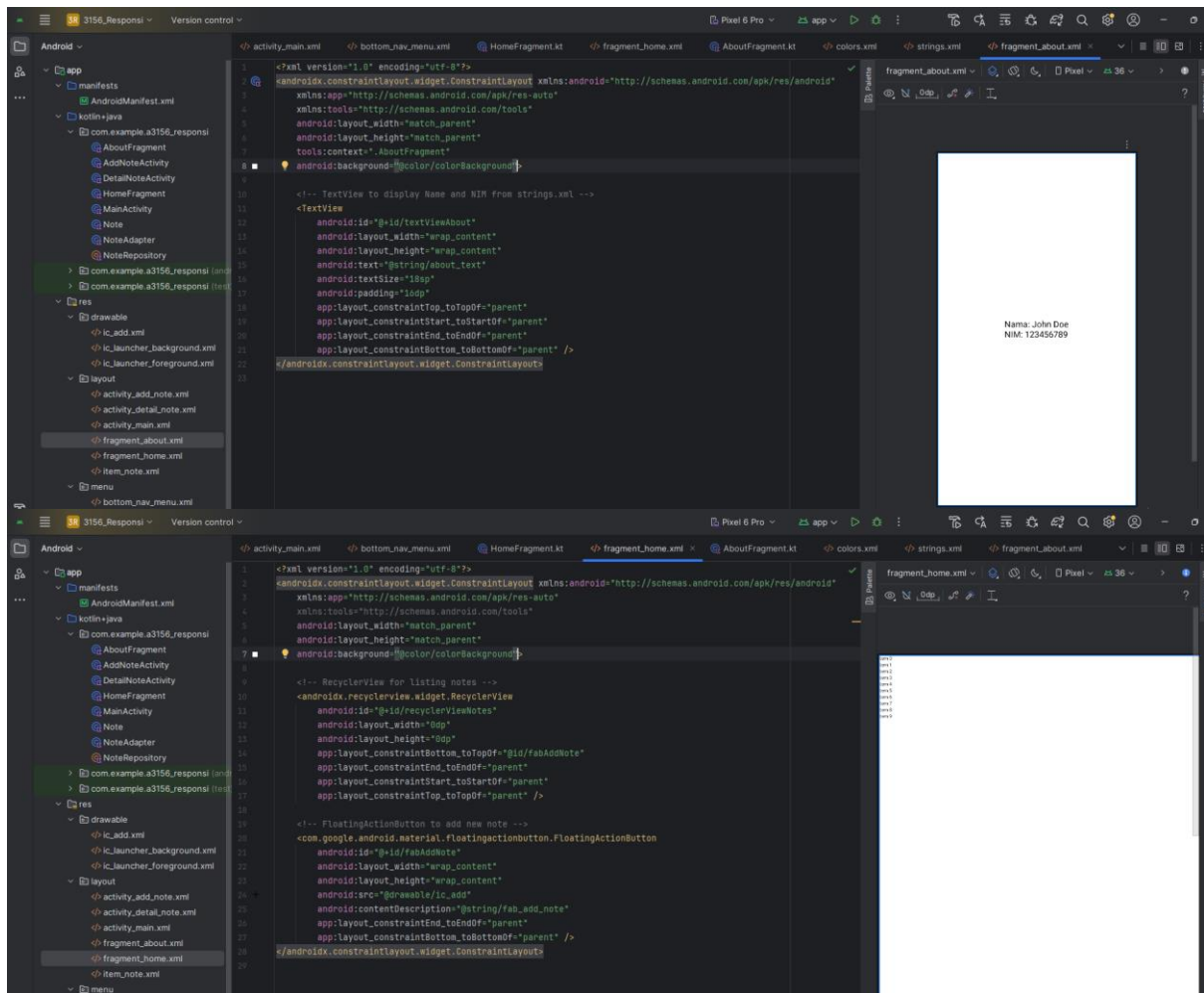


</> activity\_main.xml
×
</> bottom\_nav\_menu.xml
HomeFragment.kt
</> fragment\_ho

```

1      <?xml version="1.0" encoding="utf-8"?>
2      <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://s
3          xmlns:app="http://schemas.android.com/apk/res-auto"
4          xmlns:tools="http://schemas.android.com/tools"
5          android:layout_width="match_parent"
6          android:layout_height="match_parent"
7          tools:context=".MainActivity"
8          android:background="@color/colorBackground">
9
10         <!-- Fragment container for displaying HomeFragment or AboutFragment -
11         <androidx.fragment.app.FragmentContainerView
12             android:id="@+id/fragmentContainerView"
13             android:name="androidx.fragment.app.FragmentContainerView"
14             android:layout_width="0dp"
15             android:layout_height="0dp"
16             app:layout_constraintBottom_toBottomOf="parent"
17             app:layout_constraintEnd_toEndOf="parent"
18             app:layout_constraintStart_toStartOf="parent"
19             app:layout_constraintTop_toTopOf="parent" />
20
21         <!-- BottomNavigationView for navigating between Home and About fragme
22         <com.google.android.material.bottomnavigation.BottomNavigationView
23             android:id="@+id/bottomNavigationView"
24             android:layout_width="0dp"
25             android:layout_height="wrap_content"
26             app:menu="@menu/bottom_nav_menu"
27             app:layout_constraintBottom_toBottomOf="parent"
28             app:layout_constraintEnd_toEndOf="parent"
29             app:layout_constraintStart_toStartOf="parent" />
30
31     </androidx.constraintlayout.widget.ConstraintLayout>
32

```



Link GitHub:

[https://github.com/drsmjt-56/responsi\\_3156.git](https://github.com/drsmjt-56/responsi_3156.git)