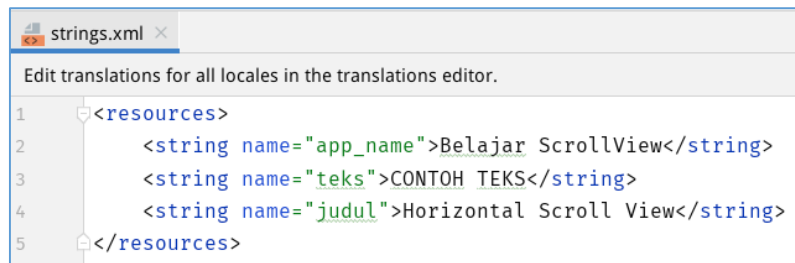


## Pertemuan 7

### Fragment (Belajar Horizontal Scroll View)

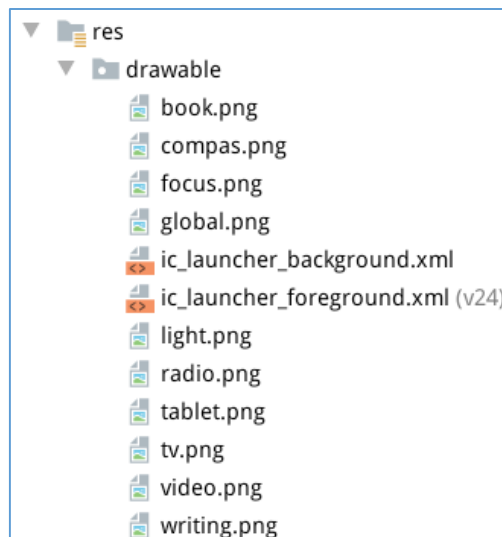
Pada modul ini kita akan mencoba membuat tampilan dengan memanfaatkan fitur scroll. Seperti yang pernah di bahas bahwa ScrollView termasuk kedalam sebuah ViewGroup. Artinya, kita dapat meletakkan sejumlah View atau Object didalam sebuah ScrollView. Kali ini kita akan mencoba menggunakan fitur scroll yang jarang dipakai, yaitu Horizontal ScrollView.

Silahkan buat sebuah projek android baru, beri nama „**BelajarHorizontalScrollView**“. Kemudian buka file **strings.xml** untuk mendaftarkan beberapa teks yang akan kita gunakan nantinya pada aplikasi.



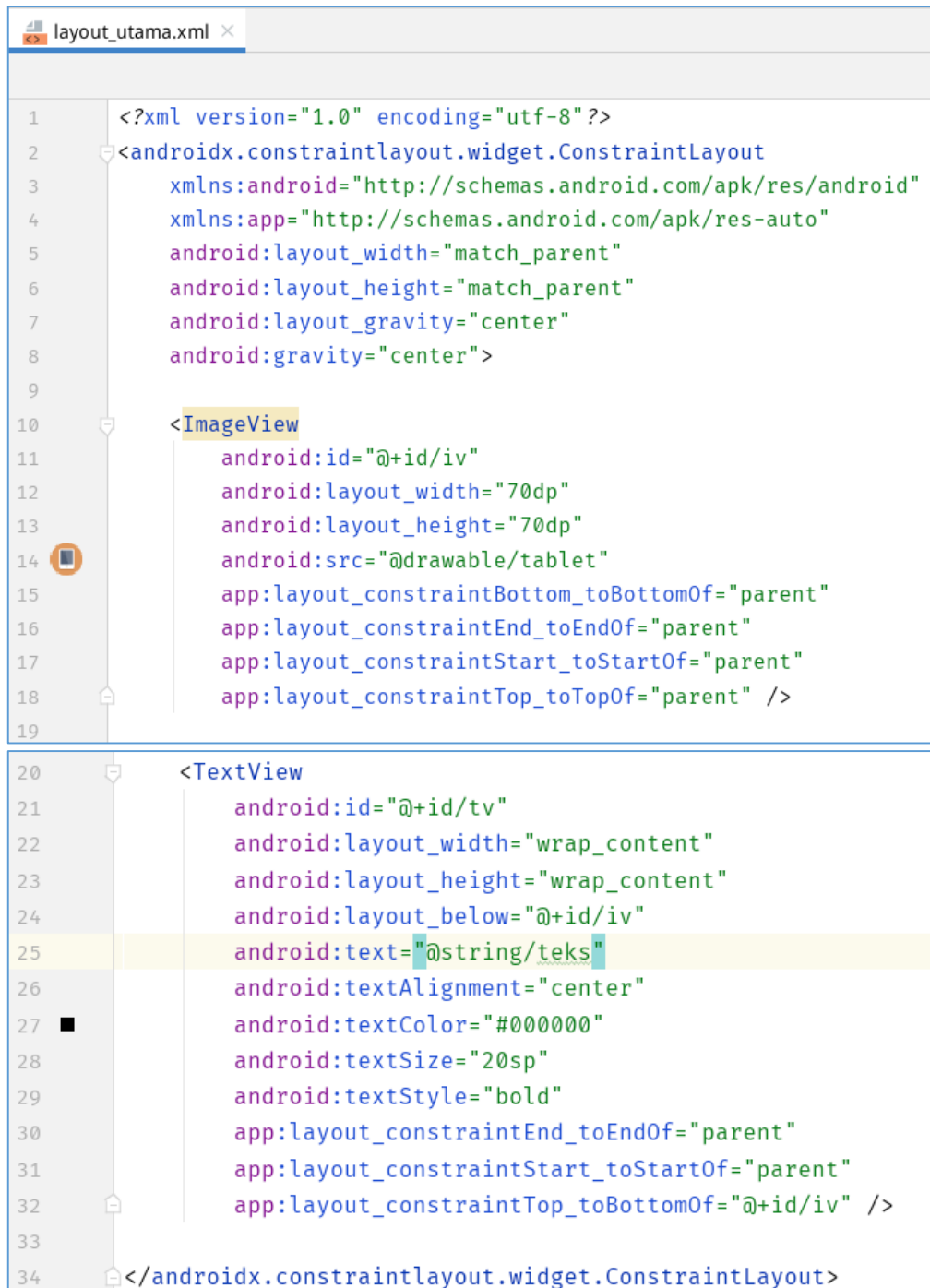
Gambar 1. Halaman string.xml

Selanjutnya anda dapat menggunakan ide bebas tentang gambar apa yang ingin ditampilkan dengan menggunakan **ScrollView**. Silahkan **unduh gambar** nya di internet, kemudian masukkan ke dalam folder **Drawable**. Apapun nama file yang digunakan, maka itu akan berpengaruh di pemanggilan pada file kotlin. Jadi silahkan diperhatikan penamaan gambar, usahakan buat semudah mungkin. Pada contoh ini, terdapat 10 jenis gambar atau *icon* berbeda yang digunakan pada aplikasi.



Gambar 2. Konten gambar di Drawable

Selanjutnya, silahkan tambahkan sebuah layout bernama **"layout\_utama.xml"**. Layout ini akan berfungsi sebagai tampilan dasar untuk gambar yang akan ditampilkan dengan menggunakan ScrollView. Kemudian silahkan ketikkan program xml di bawah ini.



```
1      <?xml version="1.0" encoding="utf-8"?>
2      <androidx.constraintlayout.widget.ConstraintLayout
3          xmlns:android="http://schemas.android.com/apk/res/android"
4          xmlns:app="http://schemas.android.com/apk/res-auto"
5          android:layout_width="match_parent"
6          android:layout_height="match_parent"
7          android:layout_gravity="center"
8          android:gravity="center">
9
10         <ImageView
11             android:id="@+id/iv"
12             android:layout_width="70dp"
13             android:layout_height="70dp"
14             android:src="@drawable/tablet"
15             app:layout_constraintBottom_toBottomOf="parent"
16             app:layout_constraintEnd_toEndOf="parent"
17             app:layout_constraintStart_toStartOf="parent"
18             app:layout_constraintTop_toTopOf="parent" />
19
20         <TextView
21             android:id="@+id/tv"
22             android:layout_width="wrap_content"
23             android:layout_height="wrap_content"
24             android:layout_below="@+id/iv"
25             android:text="@string/teks"
26             android:textAlignment="center"
27             android:textColor="#000000"
28             android:textSize="20sp"
29             android:textStyle="bold"
30             app:layout_constraintEnd_toEndOf="parent"
31             app:layout_constraintStart_toStartOf="parent"
32             app:layout_constraintTop_toBottomOf="@+id/iv" />
33
34     </androidx.constraintlayout.widget.ConstraintLayout>
```

Gambar 3. Halaman layout\_utama.xml

Setelahnya, kita akan menambahkan Horizontal Scroll View kedalam file **activity\_main.xml**. Silahkan buka halaman tersebut, kemudian ketikkan kode xxml di bawah ini.



Gambar 4. Halaman activity\_main.xml

Setelah selesai membuat file activity\_main.xml, maka selanjutnya kita akan menambahkan logika program di file **MainActivity.kt**. Pastikan untuk memanggil nama gambar yang sesuai dengan punya anda masing-masing. Sesuaikan juga teks atau judul dari masing-masing gambar pada variable menu.

```
package com.example.belajarscrollview

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.view.LayoutInflater
import android.view.View
import android.widget.ImageView
import android.widget.LinearLayout
import android.widget.TextView

class MainActivity : AppCompatActivity() {
    private var linearLayout: LinearLayout? = null
    private val menu = arrayOf("Compas", "Bulb", "TV", "Radio", "Tablet", "Book", "Focus", "Global", "Writing", "Video")
    private val gambar = intArrayOf(R.drawable.compas, R.drawable.light, R.drawable.tv, R.drawable.radio,
    R.drawable.tablet, R.drawable.book, R.drawable.focus, R.drawable.global, R.drawable.writing, R.drawable.video)

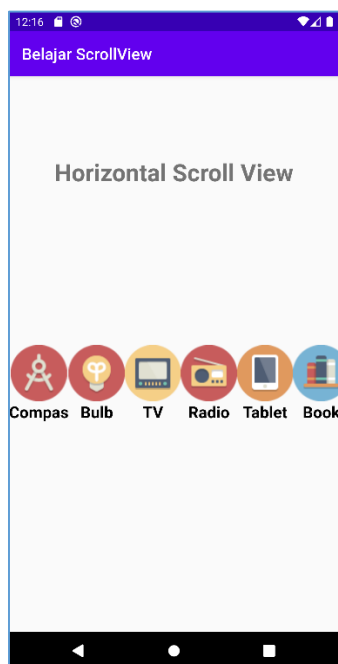
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        linearLayout = findViewById(R.id.linear1)
        val inflater : LayoutInflater = LayoutInflater.from( context: this)

        for (i :Int in menu.indices){
            val view: View = inflater.inflate(R.layout.layout_utama, linearLayout, attachToRoot: false)
            val imageView :ImageView = view.findViewById<ImageView>(R.id.iv)
            imageView.setImageResource(gambar[i])
            val tv :TextView = view.findViewById<TextView>(R.id.tv)
            tv.text = menu[i]
            linearLayout?.addView(view)
        }
    }
}
```

Gambar 5. Halaman MainActivity.kt

Aplikasi siap untuk dijalankan. Silahkan coba jalankan aplikasi tersebut, jika berhasil maka tampilan akan membentuk list gambar yang dapat di scroll ke arah kiri dan kanan seperti pada contoh di bawah ini.



Gambar 6. Tampilan Proyek BelajarHorizontalScrollView

## Fragment (Navigation Drawer)

Pada modul ini kita akan mencoba membuat tampilan menu slide seperti pada aplikasi Gmail pada android. Sebenarnya, android studio sudah menyediakan template drawer yang siap pakai. Akan tetapi agar lebih memahami prosesnya, maka dalam modul kali ini kita akan membuat aplikasi dengan menggunakan Navigation Drawer dengan membuatnya dari awal.

Silahkan buat sebuah projek android studio baru, beri nama „**BelajarNavigationDrawer**“. Kemudian silahkan tambahkan beberapa librari dibawah ini kedalam file **build.gradle**. Silahkan sesuaikan dengan apa yang ada di file anda. Jangan lupa tambahkan **viewBinding**.

```
30 dependencies {
31     implementation fileTree(dir: "libs", include: ["*.jar"])
32     implementation "org.jetbrains.kotlin:kotlin-stdlib:$kotlin_version"
33     implementation 'androidx.core:core-ktx:1.9.0'
34     implementation 'androidx.appcompat:appcompat:1.5.1'
35     implementation 'androidx.constraintlayout:constraintlayout:2.1.4'
36     implementation 'com.google.android.material:material:1.7.0'
37     implementation 'de.hdodenhof:circleimageview:3.1.0' ✓
38     implementation 'androidx.legacy:legacy-support-v4:1.0.0'
39     testImplementation 'junit:junit:4.13.2'
40     androidTestImplementation 'androidx.test.ext:junit:1.1.3'
41     androidTestImplementation 'androidx.test.espresso:espresso-core:3.4.0'
42 }
```

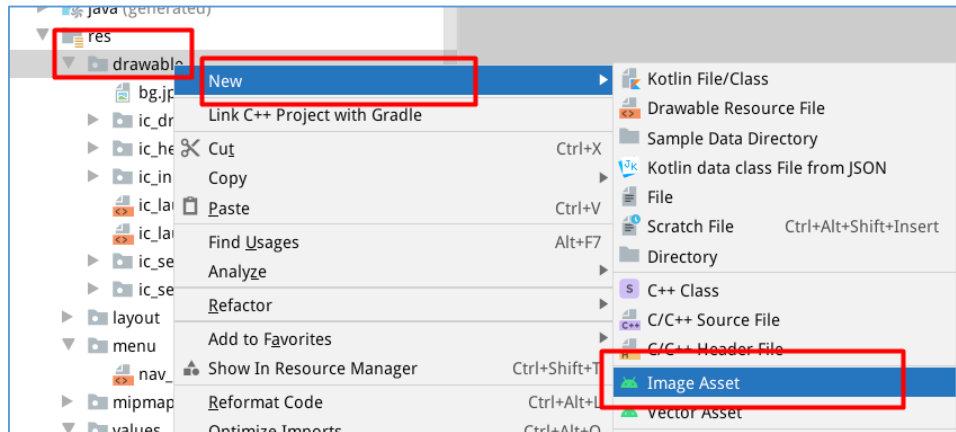
Gambar 7. Halaman build.gradle

Setelah itu, kita akan mendaftarkan teks yang digunakan pada tampilan aplikasi kedalam file strings.xml. Silahkan tambahkan beberapa teks dibawah ini. Silahkan modifikasi nama profile dan prodi sesuai keinginan anda.

```
strings.xml
Edit translations for all locales in the translations editor.
1 <resources>
2     <string name="app_name">Belajar Navigation Drawer</string>
3     <string name="open">Open</string>
4     <string name="close">Close</string>
5     <!-- TODO: Remove or change this placeholder text -->
6     <string name="hello_blank_fragment">Hello blank fragment</string>
7     <string name="halaman_utama">Halaman Utama</string>
8     <string name="halaman_inbox">Halaman Inbox</string>
9     <string name="halaman_draft">Halaman Draft</string>
10    <string name="halaman_send">Halaman Send</string>
11    <string name="nama_profile">Shumaya Resty Ramadhani</string>
12    <string name="prodi">Teknik Informatika</string>
13    <string name="inbox">Inbox</string>
14    <string name="draft">Draft</string>
15    <string name="send">Send</string>
16    <string name="halaman_help">Halaman Help</string>
17    <string name="halaman_setting">Halaman Setting</string>
18    <string name="controls">Controls</string>
19    <string name="setting">Setting</string>
20    <string name="help">Help</string>
21 </resources>
```

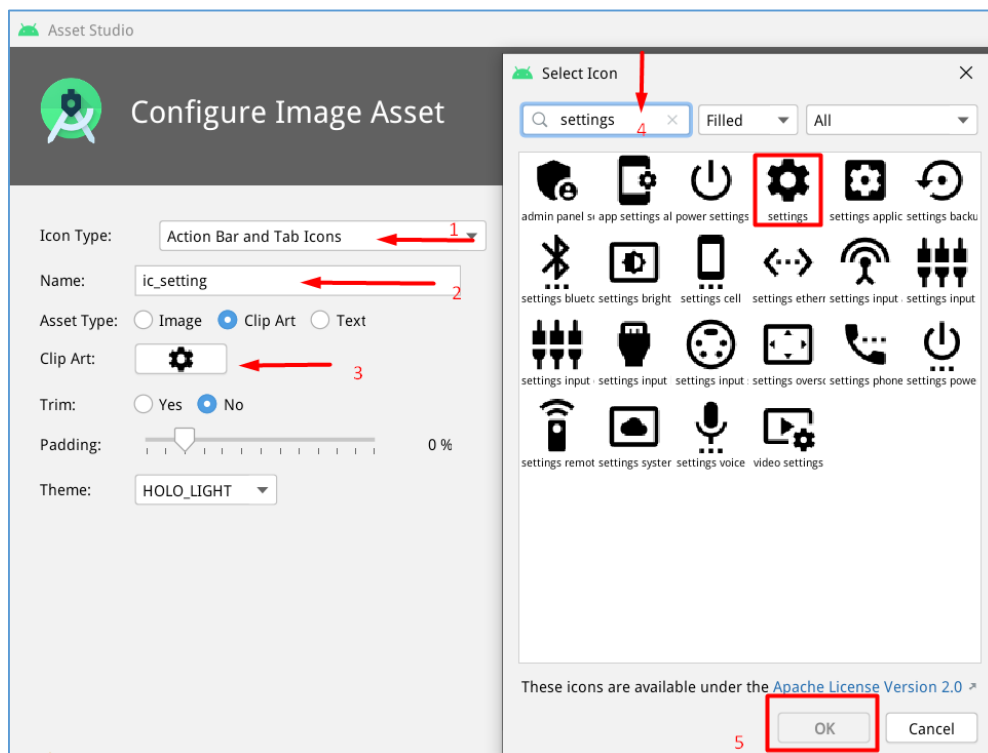
Gambar 8. Halaman strings.xml

Selanjutnya silahkan tambahkan beberapa buah gambar ke dalam folder drawable. Gambar ini akan berperan sebagai icon teks dan juga digunakan untuk header. Anda dapat mengunduh gambar yang akan digunakan sebagai background pada header (pada contoh dibawah, file background bernama bg.jpg). Kemudian untuk menambahkan beberapa icon standar, kita dapat menggunakan **Image Asset**. Lakukan proses sesuai gambar dibawah.



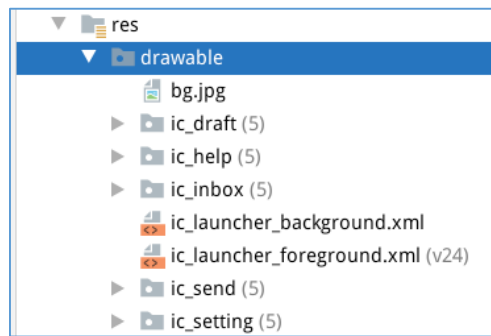
Gambar 9. Menambahkan Image Asset

Kemudian pastikan Icon Type berupa **“Action Bar and Tab Icons”**. Kemudian beri nama icon tersebut sesuai dengan kegunaannya untuk memudahkan identifikasi.



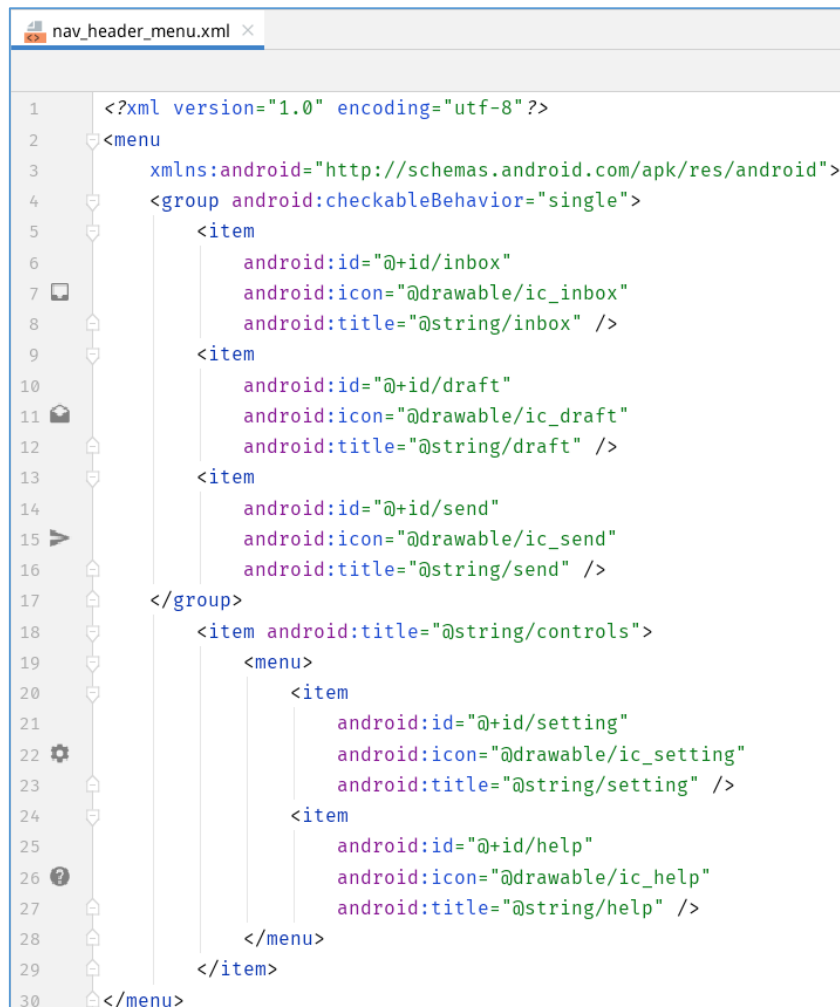
Gambar 10. Halaman Asset Studio

Jika berhasil, maka di dalam folder drawable akan terbentuk sebuah folder sesuai dengan nama yang telah anda set sebelumnya. Di dalam folder tersebut akan tersedia beberapa jenis gambar dengan ukuran yang berbeda.



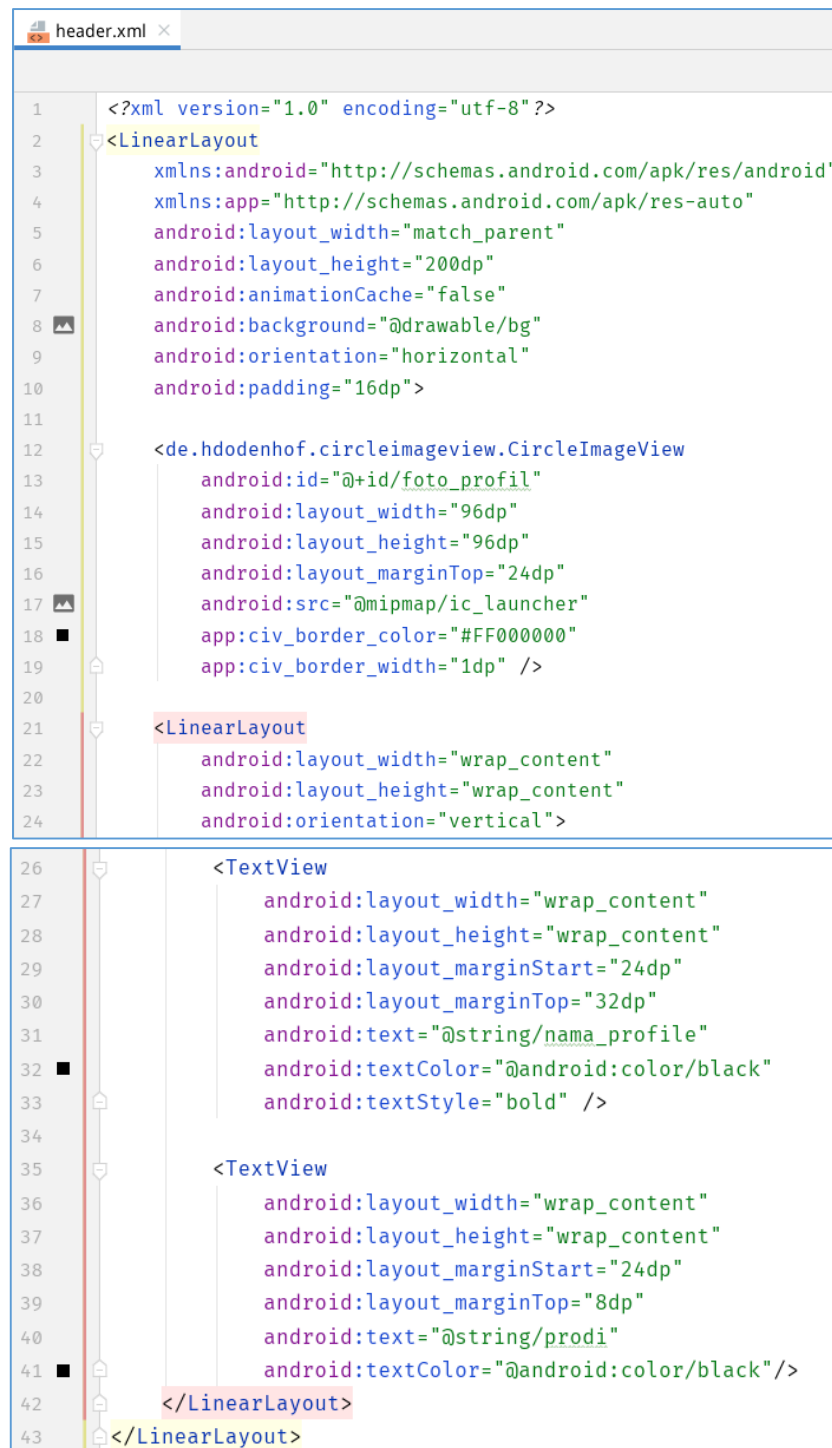
Gambar 11. Hirarki package Drawable

Langkah selanjutnya adalah membuat halaman menu. Silahkan buat sebuah packages baru dengan cara klik **kanan** pada folder **res** → **New** → **Android Resource Directory** → **Directory Name** : **menu** → **Resource type** : **menu**. Kemudian klik **kanan** pada folder **menu** → buat sebuah layout bernama **nav\_header\_menu.xml**. Tambahkan kode xml di bawah ini. Silahkan panggil nama icon sesuai dari yang anda tambahkan di awal. Jika benar, maka anda akan melihat iconnya muncul disamping nomor baris kode seperti pada contoh di bawah.



Gambar 12. Halaman nav\_header\_menu.xml

Kemudian kita akan membuat header dari menu slide. Silahkan tambahkan sebuah layout di dalam folder **layout** bernama **"header.xml"**. Kita akan membuat tampilan foto dengan beberapa informasi.

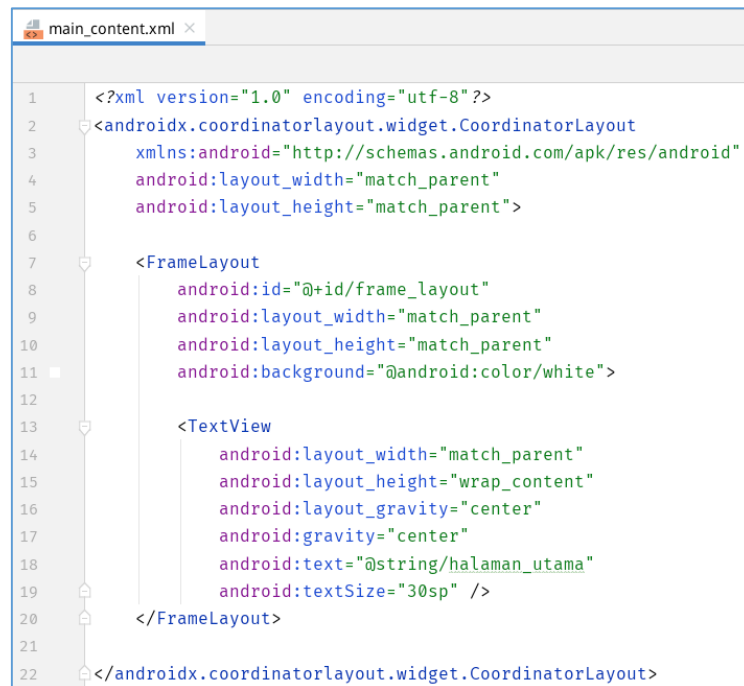


Gambar 13. Halaman header.xml

Setelah selesai, kita akan membuat tampilan halaman utama ketika pertama kali kita menjalankan aplikasi. Buat sebuah halaman layout lagi bernama **"main\_content.xml"** di dalam folder

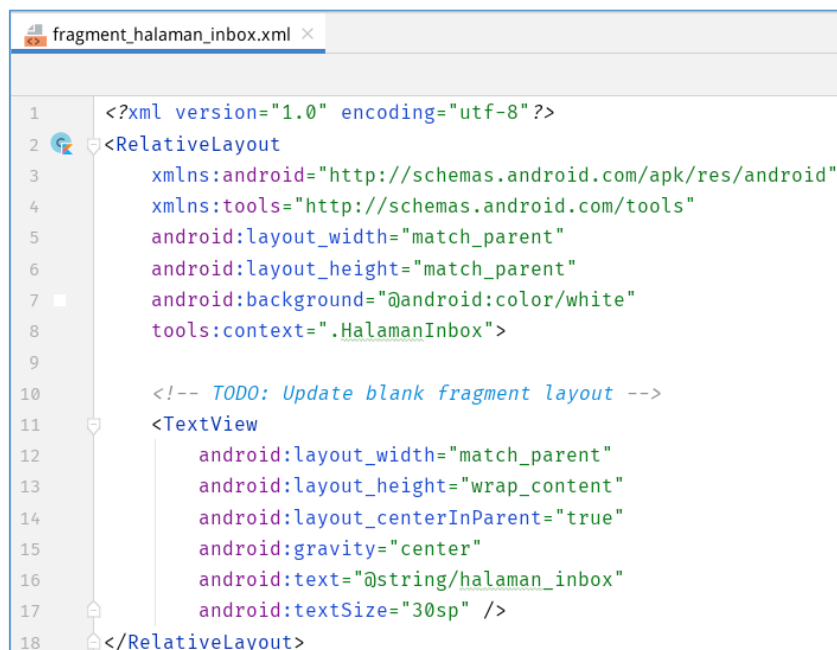


layout. Halaman inilah yang akan berubah sesuai dengan fragment yang kita miliki. Pada halaman ini hanya berisi sebuah **TextView** berisi teks „Halaman Utama“.



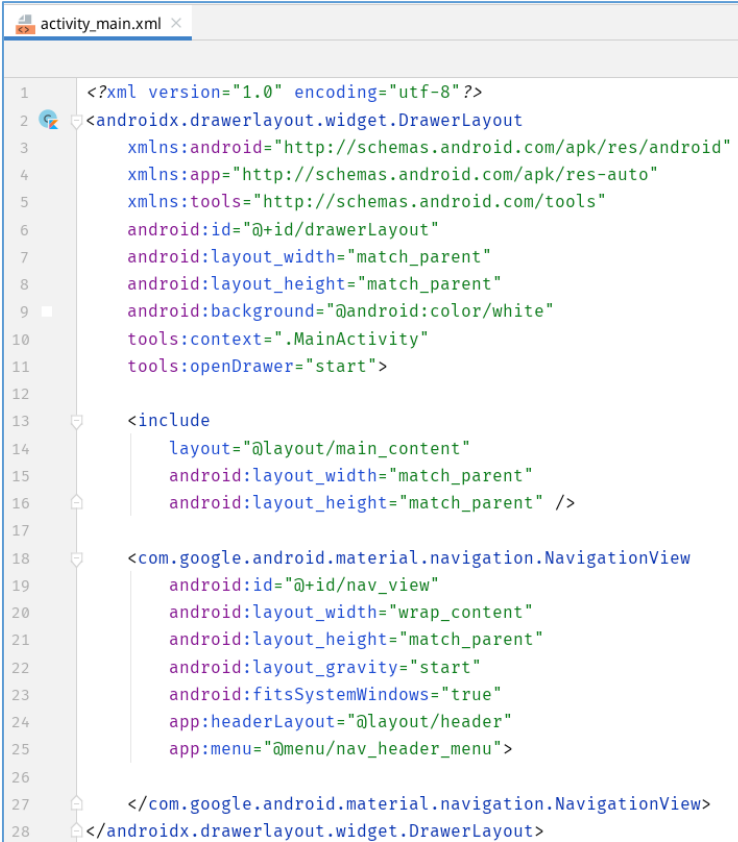
Gambar 14. Halaman main\_content.xml

Kemudian kita akan menambahkan beberapa buah fragment yang akan menjadi halaman untuk setiap menu yang kita miliki. Dalam modul ini, kita membuat sebanyak **lima** buah fragment bernama **HalamanInbox**, **HalamanSend**, **HalamanDraft**, **HalamanSetting**, **HalamanHelp**. Silahkan tambahkan sebuah **TextView** pada setiap fragment sebagai penanda fungsi dari halaman tersebut. Anda dapat menggunakan teks yang telah kita daftarkan pada file strings.xml sebelumnya.



Gambar 15. Halaman fragment

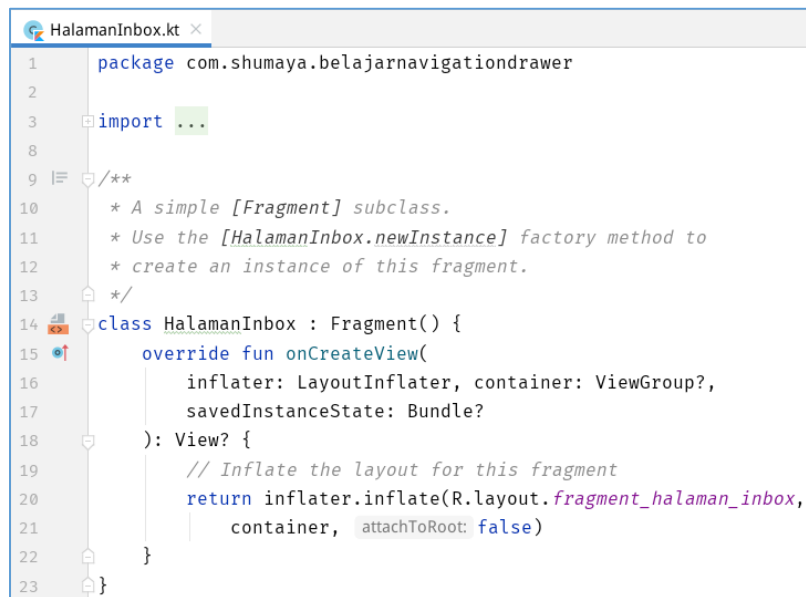
Hal terakhir yang dilakukan untuk menggabungkan seluruh layout yang kita miliki adalah memodifikasi file **activity\_main.xml**. Silahkan buka file tersebut, lalu tambahkan kode xml dibawah ini. Kita akan menggabungkan seluruh layout ke dalam **DrawerLayout**.



```
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.drawerlayout.widget.DrawerLayout
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     xmlns:app="http://schemas.android.com/apk/res-auto"
5     xmlns:tools="http://schemas.android.com/tools"
6     android:id="@+id/drawerLayout"
7     android:layout_width="match_parent"
8     android:layout_height="match_parent"
9     android:background="@android:color/white"
10    tools:context=".MainActivity"
11    tools:openDrawer="start">
12
13    <include
14        layout="@layout/main_content"
15        android:layout_width="match_parent"
16        android:layout_height="match_parent" />
17
18    <com.google.android.material.navigation.NavigationView
19        android:id="@+id/nav_view"
20        android:layout_width="wrap_content"
21        android:layout_height="match_parent"
22        android:layout_gravity="start"
23        android:fitsSystemWindows="true"
24        app:headerLayout="@layout/header"
25        app:menu="@menu/nav_header_menu">
26
27    </com.google.android.material.navigation.NavigationView>
28 </androidx.drawerlayout.widget.DrawerLayout>
```

Gambar 16. Halaman activity\_main.xml

Selanjutnya kita akan masuk ke file kotlin. Silahkan rapikan kode program kotlin pada setiap class fragment yang telah dibuat, yaitu **HalamanInbox**, **HalamanSend**, **HalamanDraft**, **HalamanSetting**, **HalamanHelp**. Silahkan hapus kode program yang tidak digunakan agar lebih rapi.



```
1 package com.shumaya.belajarnavigationdrawer
2
3 import ...
4
5
6
7
8
9 /**
10  * A simple [Fragment] subclass.
11  * Use the [HalamanInbox.newInstance] factory method to
12  * create an instance of this fragment.
13  */
14 class HalamanInbox : Fragment() {
15     override fun onCreateView(
16         inflater: LayoutInflater, container: ViewGroup?,
17         savedInstanceState: Bundle?
18     ): View? {
19         // Inflate the layout for this fragment
20         return inflater.inflate(R.layout.fragment_halaman_inbox,
21             container, attachToRoot: false)
22     }
23 }
```

Gambar 17. HalamanInbox.kt

Selanjutnya, kita akan membuat aksi ketika setiap menu ditekan pada class **MainActivity.kt**. Silahkan buka class tersebut, lalu ketikkan kode program di bawah ini.

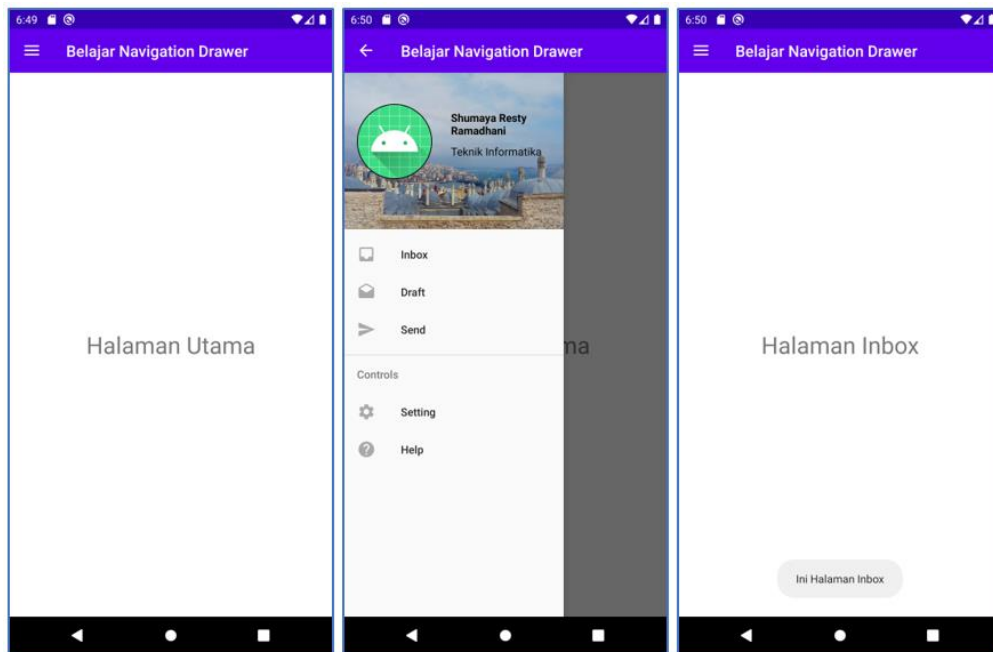
```
MainActivity.kt x
1  package com.shumaya.belajarnavigationdrawer
2
3  import androidx.appcompat.app.AppCompatActivity
4  import android.os.Bundle
5  import android.view.MenuItem
6  import android.widget.Toast
7  import androidx.appcompat.app.ActionBarDrawerToggle
8  import androidx.fragment.app.FragmentTransaction
9  import com.google.android.material.navigation.NavigationView
10 import com.shumaya.belajarnavigationdrawer.databinding.ActivityMainBinding
11
12 class MainActivity : AppCompatActivity(), NavigationView.OnNavigationItemSelectedListener {
13
14     private lateinit var binding: ActivityMainBinding
15
16     lateinit var toggle: ActionBarDrawerToggle
17     lateinit var inboxFragment: HalamanInbox
18     lateinit var draftFragment: HalamanDraft
19     lateinit var sendFragment: HalamanSend
20     lateinit var settingFragment: HalamanSetting
21     lateinit var helpFragment: HalamanHelp
22
23     override fun onCreate(savedInstanceState: Bundle?) {
24         super.onCreate(savedInstanceState)
25         binding = ActivityMainBinding.inflate(layoutInflater)
26         setContentView(binding.root)
27
28         toggle = ActionBarDrawerToggle(activity, this, binding.drawerLayout, "Open", "Close")
29         binding.drawerLayout.addDrawerListener(toggle)
30         toggle.syncState() //sekarang toggle ready utk dipake
31
32         supportActionBar?.setDisplayHomeAsUpEnabled(true)
33         binding.navView.setNavigationItemSelectedListener(this)
```

```
34     }
35
36     override fun onNavigationItemSelected(item: MenuItem): Boolean {
37         when (item.itemId) {
38             R.id.inbox -> {
39                 inboxFragment = HalamanInbox()
40                 supportFragmentManager
41                     .beginTransaction()
42                     .replace(R.id.frame_layout, inboxFragment)
43                     .setTransition(FragmentTransaction.TRANSIT_FRAGMENT_OPEN)
44                     .commit()
45                 Toast.makeText(applicationContext, text: "Ini Halaman Inbox", Toast.LENGTH_SHORT)
46                     .show()
47             }
48             R.id.draft -> {
49                 draftFragment = HalamanDraft()
50                 supportFragmentManager
51                     .beginTransaction()
52                     .replace(R.id.frame_layout, draftFragment)
53                     .setTransition(FragmentTransaction.TRANSIT_FRAGMENT_OPEN)
54                     .commit()
55                 Toast.makeText(applicationContext, text: "Ini Halaman Draft", Toast.LENGTH_SHORT)
56                     .show()
57             }
58             R.id.send -> {
59                 sendFragment = HalamanSend()
60                 supportFragmentManager
61                     .beginTransaction()
62                     .replace(R.id.frame_layout, sendFragment)
63                     .setTransition(FragmentTransaction.TRANSIT_FRAGMENT_OPEN)
64                     .commit()
65                 Toast.makeText(applicationContext, text: "Ini Halaman Send", Toast.LENGTH_SHORT)
66                     .show()
67             }
68         }
69     }
70 }
```

```
67     }
68     R.id.setting -> {
69         settingFragment = HalamanSetting()
70         supportFragmentManager
71             .beginTransaction()
72             .replace(R.id.frame_layout, settingFragment)
73             .setTransition(FragmentTransaction.TRANSIT_FRAGMENT_OPEN)
74             .commit()
75         Toast.makeText(applicationContext, text: "Ini Halaman Setting", Toast.LENGTH_SHORT)
76             .show()
77     }
78     R.id.help -> {
79         helpFragment = HalamanHelp()
80         supportFragmentManager
81             .beginTransaction()
82             .replace(R.id.frame_layout, helpFragment)
83             .setTransition(FragmentTransaction.TRANSIT_FRAGMENT_OPEN)
84             .commit()
85         Toast.makeText(applicationContext, text: "Ini Halaman Help", Toast.LENGTH_SHORT)
86             .show()
87     }
88 }
89 binding.drawerLayout.closeDrawers()
90 return true
91 }
92
93 override fun onOptionsItemSelected(item: MenuItem): Boolean {
94     if (toggle.onOptionsItemSelected(item)) {
95         return true
96     }
97     return super.onOptionsItemSelected(item)
98 }
99 }
```

Gambar 18. Halaman MainActivity.kt

Setelah selesai, silahkan jalankan aplikasi. Jika berhasil, maka akan muncul tampilan seperti pada contoh di bawah ini. Ketika menu ditekan, maka akan terbuka fragment dari menu tersebut sekaligus muncul pop up teks nya.



Gambar 19. Tampilan Aplikasi

### **Tugas Latihan (kumpul akhir praktikum) :**

1. Silahkan buat latihan pada projek baru bernama **LatihanNavDrawer**. Kemudian buat tampilan seperti pada contoh berikut ini. Manfaatkan layout constraint/linear layout agar tampilan gambar tersusun seperti itu.
2. Jika selesai, maka ketika setiap menu di klik maka akan menampilkan halaman putih kosong yng berisi textview sesuai judul menu.

