

LAPORAN PRAKTIKUM PENGEMBANGAN APLIKASI BERGERAK

Android Fundamental 1 - WEEK 6



Disusun oleh:

Nama : Alfath Roziq Widhayaka

Nim : L0122012

Kelas : A

PROGRAM STUDI INFORMATIKA

FAKULTAS TEKNOLOGI INFORMASI DAN SAINS DATA

UNIVERSITAS SEBELAS MARET

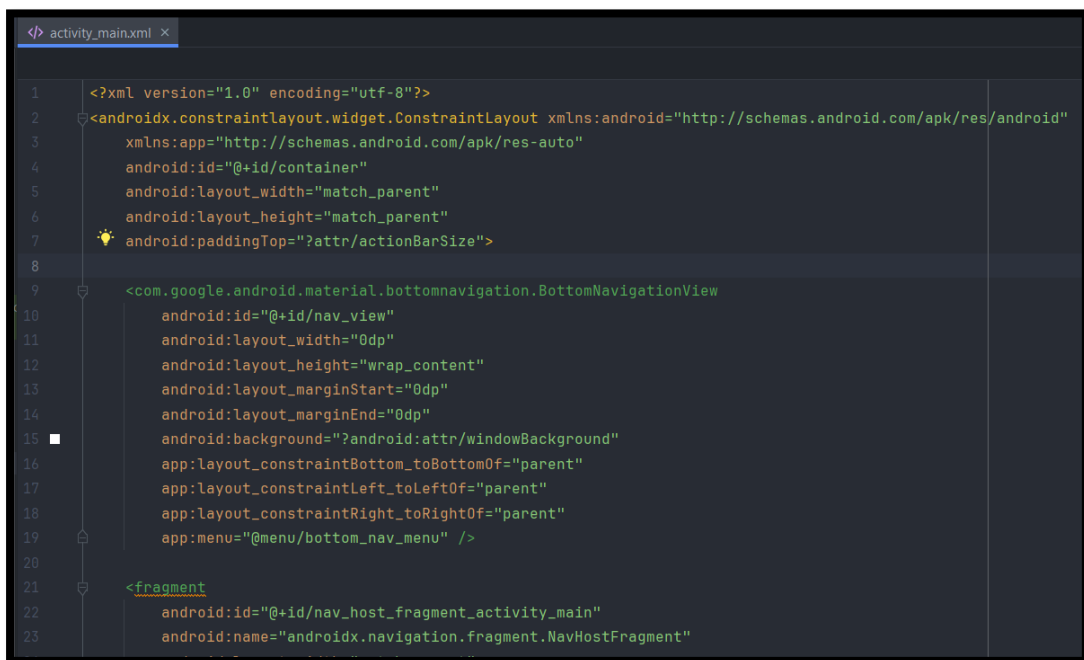
2024

1. Screenshot Source Code

Berikut adalah tugas PAB Week 6 tentang Android Fundamental, yang menitikberatkan pada penerapan **Bottom Navigation** dan **Navigation Component** dalam pembuatan aplikasi. Saya mengimplementasikan materi tersebut dalam sebuah aplikasi Music Player sederhana, yang menawarkan Bottom Navigation dengan opsi Home, Favorite, dan Account. Penerapan Navigation Component dengan bundle digunakan untuk mengarahkan pengguna saat memilih opsi Top 2 Trending. Di bawah ini, saya akan menjelaskan setiap komponen secara detail.

A. Bottom Navigation

1. Membuat activity_main.xml yang berisi Bottom Navigation

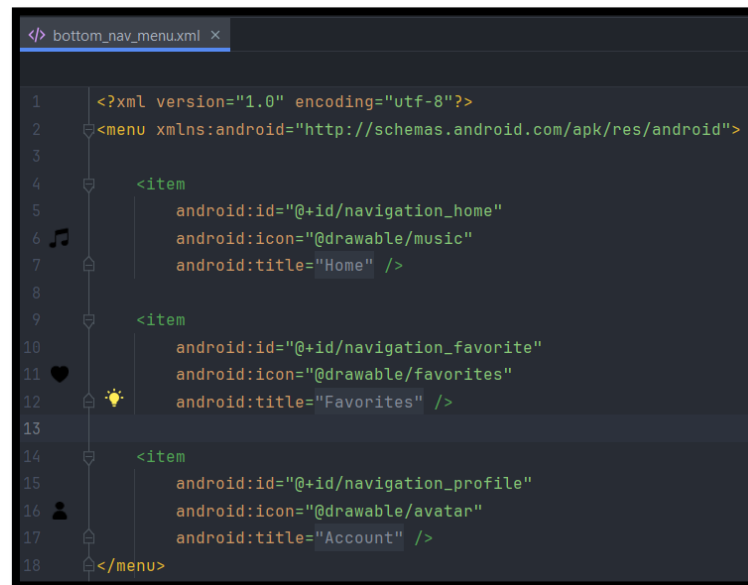


Gambar 1 activity_main.xml

Source code XML diatas terdapat **BottomNavigationView** yang digunakan untuk menyediakan navigasi antara beberapa pilihan dalam aplikasi. BottomNavigationView diberi **ID nav_view**. Kemudian, terdapat **fragment NavHostFragment** yang menampilkan konten utama aplikasi dengan menggantikan fragmen tujuan. Fragment ini menggunakan **ID nav_host_fragment_activity_main** dan diset sebagai default navigation host, sehingga semua navigasi diarahkan ke fragmen ini. Fragmen ini mengisi seluruh ruang kosong yang tersedia di antara ConstraintLayout dan BottomNavigationView.

Dengan struktur ini, user dapat berpindah antara fragmen/pilihan yang diinginkan melalui **BottomNavigationView**, sementara **NavHostFragment** akan menangani penggantian fragmen yang sesuai dengan tujuan yang dipilih, menjaga navigasi dalam satu aktivitas tanpa perlu membuka aktivitas baru untuk setiap fragmen yang ditampilkan.

2. Menyesuaikan logo pada bottom navigation



Gambar 2 bottom_nav_menu.xml

Source code XML di atas adalah representasi dari menu navigasi dalam format XML untuk aplikasi Android. Dalam kode tersebut, terdapat elemen **<menu>** yang mendefinisikan menu navigasi. Setiap **<item>** di dalam **<menu>** mewakili satu opsi navigasi dalam menu tersebut. Setiap **<item>** memiliki atribut **android:id** yang memberikan ID unik untuk opsi navigasi tersebut. Tiga item tersebut merupakan navigasi: **Home**, **Favorite**, dan **Profile**, masing-masing dengan ikon yang beda dengan template dan teks judul yang terkait.

3. Menyesuaikan behaviour dengan Bottom Navigation terkait

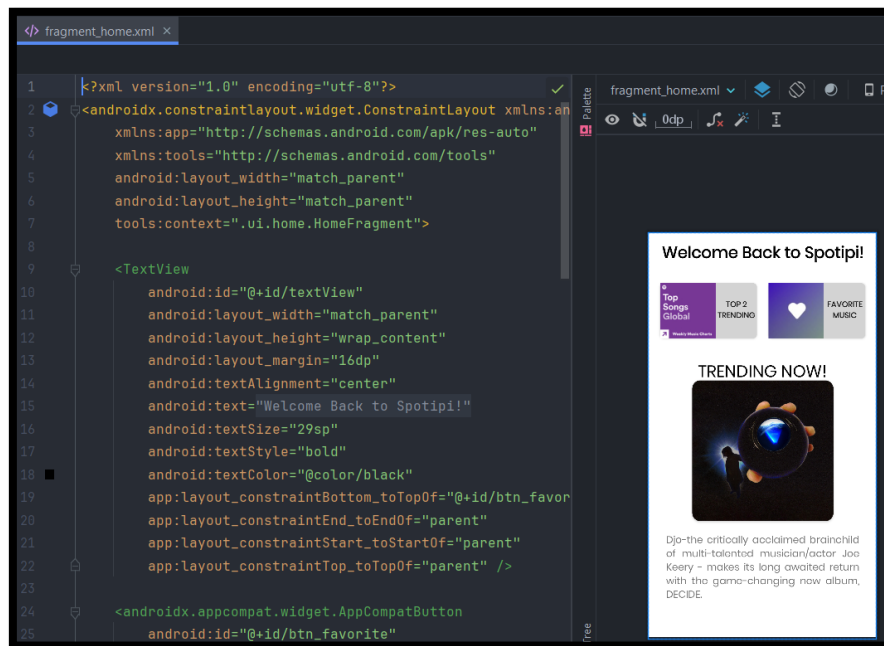


Gambar 3 MainActivity.kt

Tambahkan potongan kode diatas yang digunakan supaya AppBar pada setiap fragment memiliki behaviour yang sesuai dengan Bottom Navigation tadi. Tulis sesuai dengan id yang dideklarasikan tadi pada **bottom_nav_menu.xml**. Setelah menambahkan potongan kode ini maka **Bottom Navigation** sudah bisa digunakan dengan sempurna.

B. Menyesuaikan Isi Konten

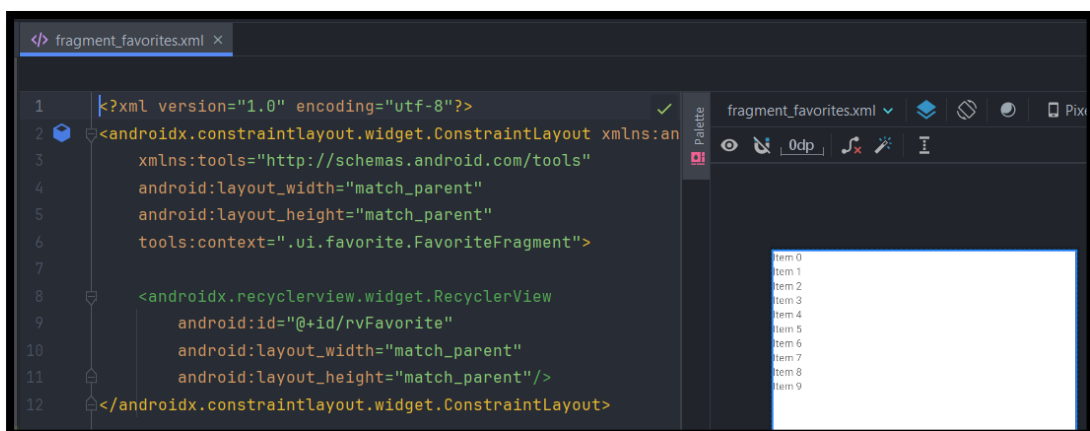
1. Fragment Home



Gambar 4 fragment_home.xml

Source code diatas merupakan tampilan fragment utama atau homepage awal yang menampilkan beberapa elemen UI, termasuk **TextView**, dua button (**AppCompatButton**), sebuah **CardView**, dan **ImageView**. TextView pertama menampilkan teks dengan format tertentu, yang diatur untuk berada di bagian atas layar. Dua button ditempatkan di bawahnya dengan menggunakan constraint agar saling terkait. CardView dan ImageView digunakan untuk menampilkan gambar dalam sebuah card dengan sudut membulat. TextView terakhir menampilkan deskripsi dan ditempatkan di bagian bawah layar. Untuk penjelasan selanjutnya pada Home Fragment ini dijelaskan pada Navigation Component untuk kedua bottom diatas.

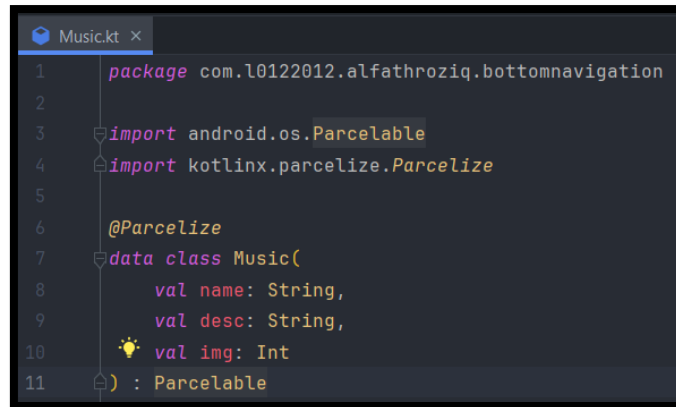
2. Fragment Favorite



Gambar 5 fragment_favorites.xml

Source code diatas merupakan XML yang memiliki **ConstraintLayout** dan di dalamnya terdapat sebuah **RecyclerView** dengan id **rvFavorite**. Lebar dan tinggi

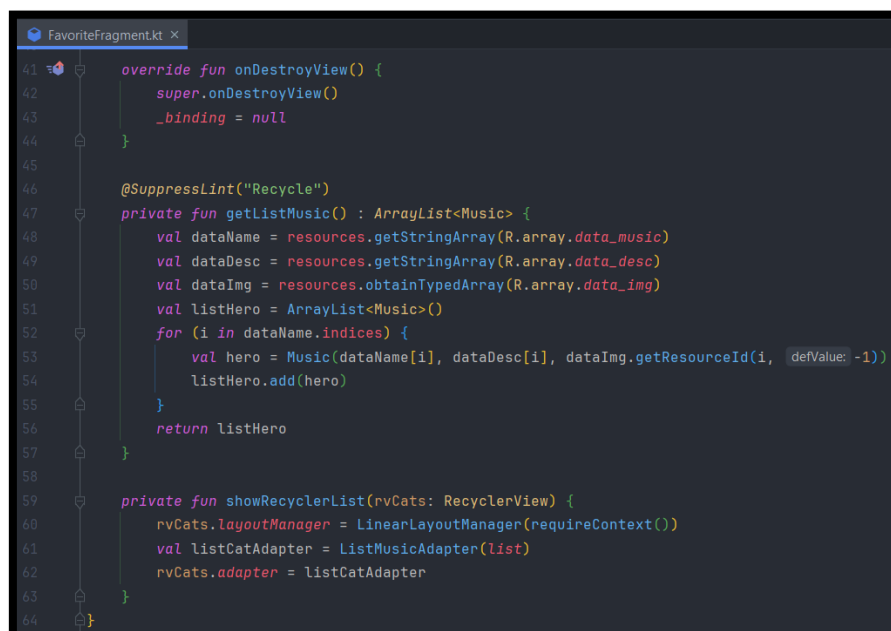
RecyclerView diatur agar sesuai dengan lebar dan tinggi parentnya, sehingga akan menempati seluruh ruang yang tersedia dalam layar. Ini memungkinkan tampilan daftar **list view** yang dapat digulirkan penuh untuk menampilkan data favorit user dalam aplikasi. Source code ini saya tambahkan dikarenakan menerapkan materi week sebelumnya dan menjadi lebih menarik isi kontennya.



```
1 package com.l0122012.alfathroziq.bottomnavigation
2
3 import android.os.Parcelable
4 import kotlinx.parcelize.Parcelize
5
6 @Parcelize
7 data class Music(
8     val name: String,
9     val desc: String,
10     val img: Int
11 ) : Parcelable
```

Gambar 6 Music.kt

Penambahan **Kelas Music** memiliki tiga property yaitu **name** untuk menyimpan nama musik, **desc** untuk deskripsi musik, dan **img** untuk menyimpan referensi gambar musik. Dengan menggunakan **@Parcelize**, Kotlin Android Extensions secara otomatis akan menghasilkan kode serialisasi/deserialisasi untuk kelas ini, sehingga objek Music dapat dengan mudah dilewatkan antar komponen aplikasi Android seperti intent dan Bundle.



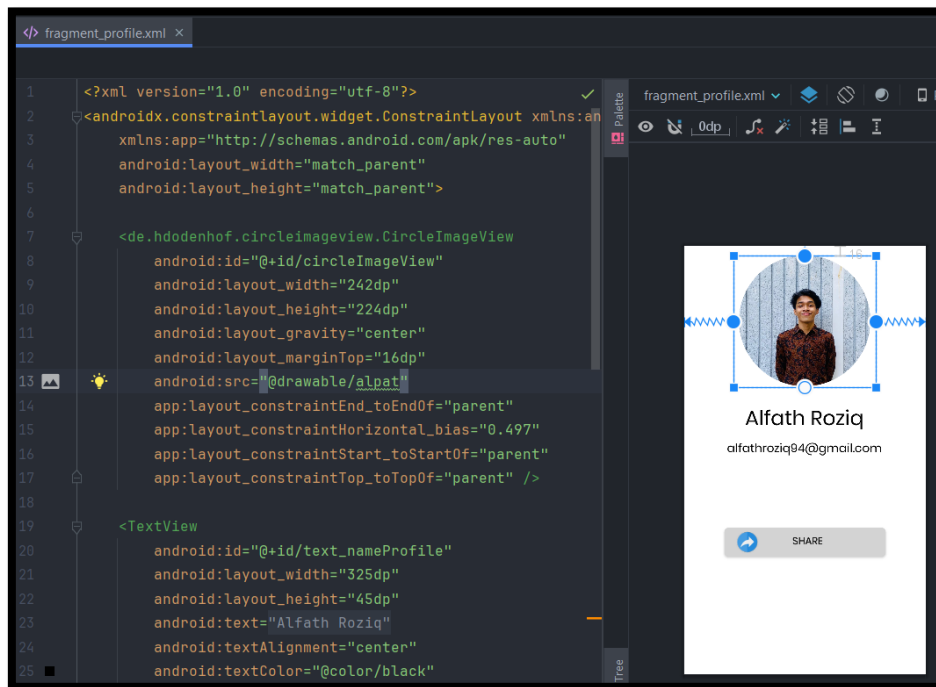
```
41 override fun onDestoryView() {
42     super.onDestoryView()
43     _binding = null
44 }
45
46 @SuppressLint("Recycle")
47 private fun getListMusic() : ArrayList<Music> {
48     val dataName = resources.getStringArray(R.array.data_music)
49     val dataDesc = resources.getStringArray(R.array.data_desc)
50     val dataImg = resources.obtainTypedArray(R.array.data_img)
51     val listHero = ArrayList<Music>()
52     for (i in dataName.indices) {
53         val hero = Music(dataName[i], dataDesc[i], dataImg.getResourceId(i, defValue: -1))
54         listHero.add(hero)
55     }
56     return listHero
57 }
58
59 private fun showRecyclerView(rvCats: RecyclerView) {
60     rvCats.layoutManager = LinearLayoutManager(requireContext())
61     val listCatAdapter = ListMusicAdapter(list)
62     rvCats.adapter = listCatAdapter
63 }
64 }
```

Gambar 7 FavoriteFragment.kt

Potongan source code diatas ditambahkan sebagai penggunaan **fungsi RecyclerView** pada fragment favorite di Aplikasi Music Player ini. **data_name**, **data_desc**, dan **data_img** diambil dari data-data yang telah dideklarasikan pada **strings.xml** dengan jumlah yang sama setiap datanya. Dengan menambahkan potongan kode diatas maka

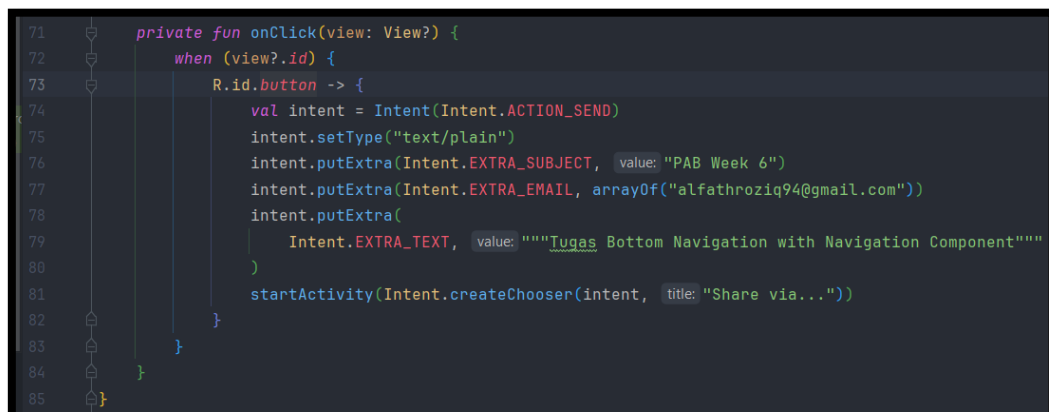
fungsi RecyclerView pada fragment favorite ini dapat berjalan dengan lancar tetapi tidak menggunakan fungsi onClick.

3. Fragment Profile



Gambar 8 fragment_profile.xml

Source code XML diatas merupakan isi konten dari fragment profile, jika user memilih navigasi profile maka akan muncul seperti gambar diatas. Untuk source code diatas hampir sama seperti tugas sebelumnya tentang profil pribadi dengan menambahkan tombol **share**.

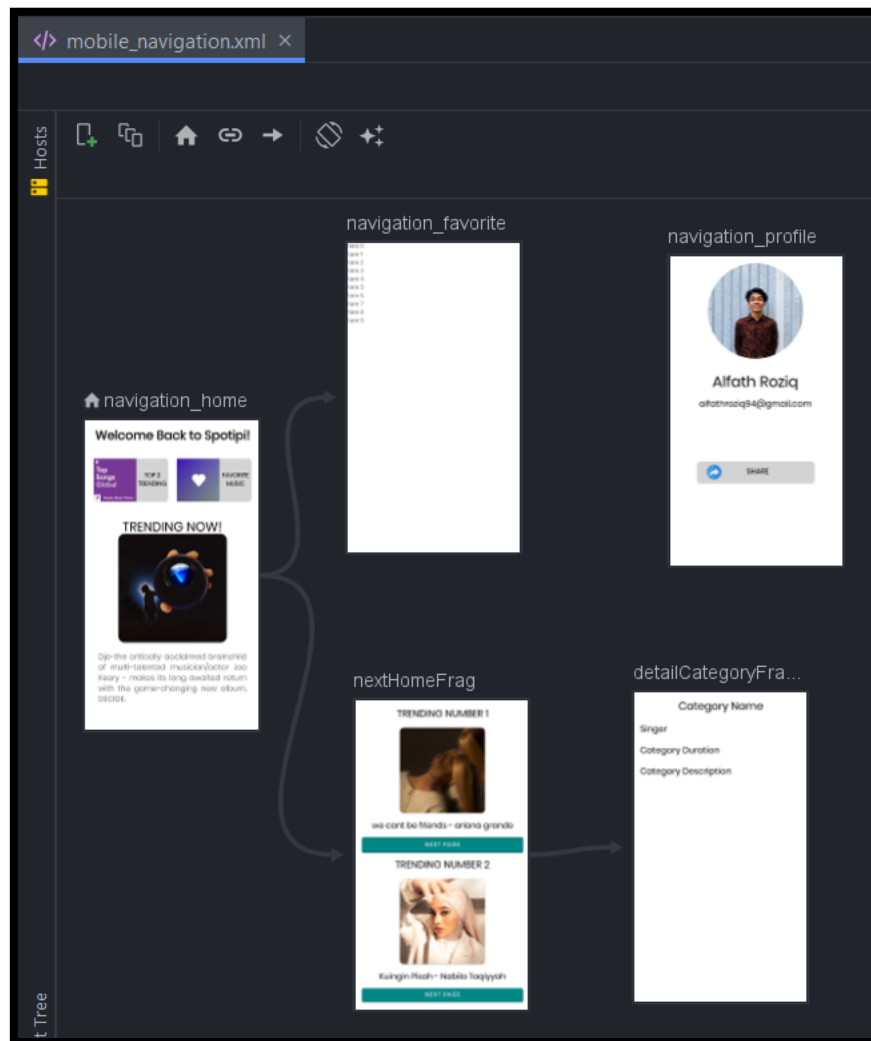


Gambar 9 FragmentProfile.kt

Source code diatas merupakan potongan kode yang ditambahkan pada **FragmentProfile.kt** yang digunakan sebagai fungsi **Intent** pada tombol share tadi. Jika user menekan tombol share pada fragment profile maka akan muncul pilihan aplikasi yang diinginkan user dengan tambahan pesan “Tugas Bottom Navigation with Navigation Component”.

C. Navigation Component & Send Data Using Bundle

1. Membuat Destination Navigation pada Fragment Home



Gambar 10 Destination Navigation

Gambar di atas menunjukkan tujuan atau destinasi pada bottom yang ada pada **fragment home**. Di sini, saya telah menetapkan tujuan pada tombol pertama, yaitu **TOP 2 TRENDING** untuk menuju ke **nextHomeFrag** yang berisi 2 musik yang sedang tren. Tombol kedua, **FAVORITE MUSIC** memiliki tujuan menuju **fragment favorite** yang ada dalam bottom navigation. Kedua Navigation Component ini dapat berjalan dengan baik ketika pengguna menekan kedua tombol tersebut. Untuk destinasi selanjutnya, saya menyambungkan tombol yang ada di **nextHomeFrag** ke **detailCategoryFragment**. Dengan demikian, ketika pengguna menekan tombol di **nextHomeFrag**, tampilan akan dialihkan secara otomatis ke **detailCategoryFragment** yang berisi detail tentang musik yang sedang tren.

```

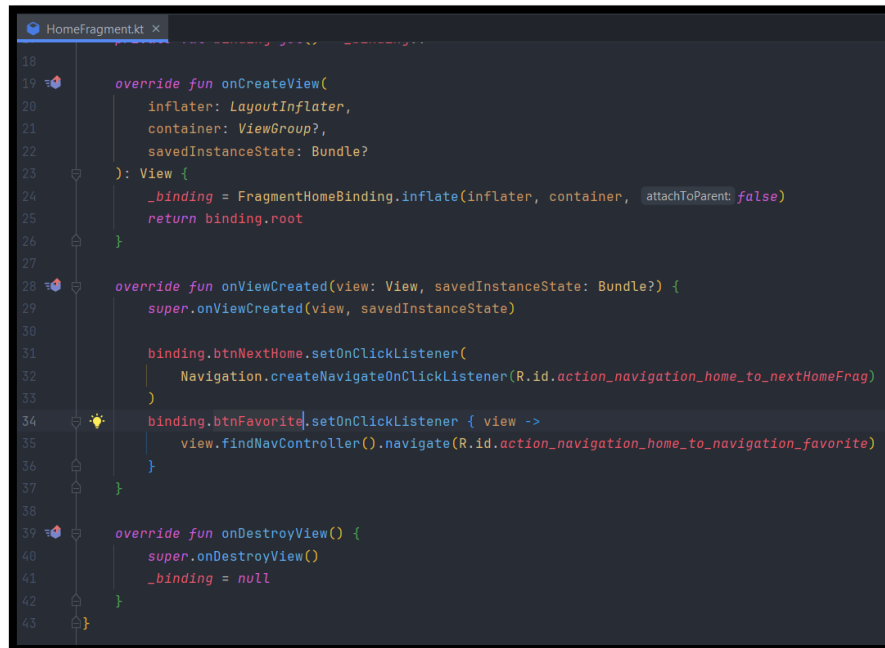
1  <?xml version="1.0" encoding="utf-8"?>
2  <navigation xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:id="@+id/mobile_navigation"
6      app:startDestination="@+id/navigation_home">
7
8      <fragment
9          android:id="@+id/navigation_home"
10         android:name="com.l0122012.alfathroziq.bottomnavigation.ui.home.HomeFragment"
11         android:label="Home"
12         tools:layout="@layout/fragment_home" >
13         <action
14             android:id="@+id/action_navigation_home_to_nextHomeFrag"
15             app:destination="@id/nextHomeFrag" />
16         <action
17             android:id="@+id/action_navigation_home_to_navigation_favorite"
18             app:destination="@id/navigation_favorite" />
19     </fragment>
20
21     <fragment
22         android:id="@+id/navigation_favorite"
23         android:name="com.l0122012.alfathroziq.bottomnavigation.ui.favorite.FavoriteFragment"
24         android:label="Favorites"
25         tools:layout="@layout/fragment_favorites" />
26
27     <fragment
28         android:id="@+id/navigation_profile"
29         android:name="com.l0122012.alfathroziq.bottomnavigation.ui.profile.FragmentProfile"
30         android:label="Account"
31         tools:layout="@layout/fragment_profile" />
32     <fragment
33         android:id="@+id/nextHomeFrag"
34         android:name="com.l0122012.alfathroziq.bottomnavigation.NextHomeFrag"
35         android:label="Trending"
36         tools:layout="@layout/fragment_next_home" >
37         <action
38             android:id="@+id/action_nextHomeFrag_to_detailCategoryFragment"
39             app:destination="@id/detailCategoryFragment" />
40     </fragment>
41     <fragment
42         android:id="@+id/detailCategoryFragment"
43         android:name="com.l0122012.alfathroziq.bottomnavigation.DetailCategoryFragment"
44         android:label="About"
45         tools:layout="@layout/fragment_detail_category" />
46 </navigation>

```

Gambar 11 Source Destination Navigation pada mobile_navigation.xml

Gambar diatas merupakan source code pada mobile_navigation diatas yang digunakan sebagai set destination. Untuk **action** ini nantinya akan otomatis ditambahkan saat user menarik garis dari satu fragment ke fragment lain. Masing-masing action disini memiliki nama tersendiri yang digunakan sebagai penanda.

2. Mengatur Action pada Home Fragment

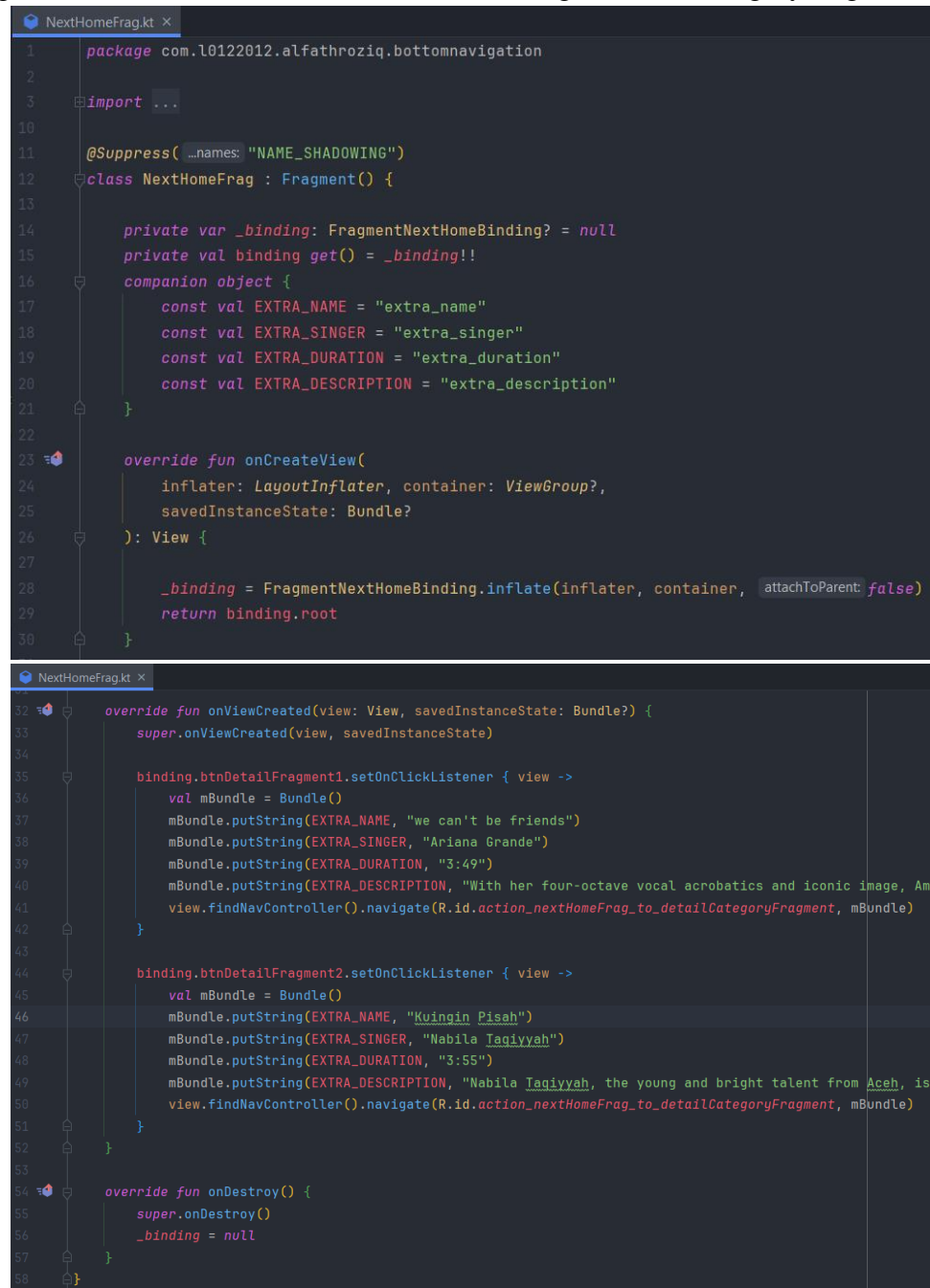


```
18
19 override fun onCreateView(
20     inflater: LayoutInflater,
21     container: ViewGroup?,
22     savedInstanceState: Bundle?
23 ): View {
24     _binding = FragmentHomeBinding.inflate(inflater, container, attachToParent: false)
25     return binding.root
26 }
27
28 override fun onViewCreated(view: View, savedInstanceState: Bundle?) {
29     super.onViewCreated(view, savedInstanceState)
30
31     binding.btnNextHome.setOnClickListener {
32         Navigation.createNavigateOnClickListener(R.id.action_navigation_home_to_nextHomeFrag)
33     }
34     binding.btnFavorite.setOnClickListener { view ->
35         view.findNavController().navigate(R.id.action_navigation_home_to_navigation_favorite)
36     }
37 }
38
39 override fun onDestroyView() {
40     super.onDestroyView()
41     _binding = null
42 }
43 }
```

Gambar 11 Source code HomeFragment.kt

Source code di atas adalah bagian dari Bottom Navigation. **Kelas HomeFragment** adalah sebuah Fragment yang menampilkan tata letak antarmuka pengguna untuk tampilan beranda. Disini menerapkan **metode onViewCreated** sebagai aksi klik tombol **btnFavorite** dan **btnNextHome**. Ketika tombol **btnNextHome** diklik, navigasi dilakukan menggunakan **Navigation.createNavigateOnClickListener** untuk menuju **NextHomeFrag**. Sedangkan ketika tombol **btnFavorite** diklik, navigasi dilakukan menggunakan **view.findNavController().navigate** untuk menuju **navigation_favorite**. Setelah tampilan fragment dihancurkan, metode **onDestroyView** akan dipanggil untuk membersihkan referensi binding.

3. Mengirim Send Action Bundle dari NextHomeFrag ke DetailCategoryFragment



```
1 package com.l0122012.alfathroziq.bottomnavigation
2
3 import ...
4
5
6
7
8
9
10
11 @Suppress( ...names: "NAME_SHADOWING")
12 class NextHomeFrag : Fragment() {
13
14     private var _binding: FragmentNextHomeBinding? = null
15     private val binding get() = _binding!!
16
17     companion object {
18         const val EXTRA_NAME = "extra_name"
19         const val EXTRA_SINGER = "extra_singer"
20         const val EXTRA_DURATION = "extra_duration"
21         const val EXTRA_DESCRIPTION = "extra_description"
22     }
23
24     override fun onCreateView(
25         inflater: LayoutInflater, container: ViewGroup?,
26         savedInstanceState: Bundle?
27     ): View {
28         _binding = FragmentNextHomeBinding.inflate(inflater, container, attachToParent: false)
29         return binding.root
30     }
31
32     override fun onViewCreated(view: View, savedInstanceState: Bundle?) {
33         super.onViewCreated(view, savedInstanceState)
34
35         binding.btnDetailFragment1.setOnClickListener { view ->
36             val mBundle = Bundle()
37             mBundle.putString(EXTRA_NAME, "we can't be friends")
38             mBundle.putString(EXTRA_SINGER, "Ariana Grande")
39             mBundle.putString(EXTRA_DURATION, "3:49")
40             mBundle.putString(EXTRA_DESCRIPTION, "With her four-octave vocal acrobatics and iconic image, Ariana Grande is one of the most talented and influential artists of her generation.")
41             view.findNavController().navigate(R.id.action_nextHomeFrag_to_detailCategoryFragment, mBundle)
42         }
43
44         binding.btnDetailFragment2.setOnClickListener { view ->
45             val mBundle = Bundle()
46             mBundle.putString(EXTRA_NAME, "Kuingin Pisah")
47             mBundle.putString(EXTRA_SINGER, "Nabila Tagiyyah")
48             mBundle.putString(EXTRA_DURATION, "3:55")
49             mBundle.putString(EXTRA_DESCRIPTION, "Nabila Tagiyyah, the young and bright talent from Aceh, is a rising star in the Indonesian music scene.")
50             view.findNavController().navigate(R.id.action_nextHomeFrag_to_detailCategoryFragment, mBundle)
51         }
52     }
53
54     override fun onDestroy() {
55         super.onDestroy()
56         _binding = null
57     }
58 }
```

Gambar 12 Source code NextHomeFrag.kt

Fragment **NextHomeFrag** merupakan bagian dari bottom navigation setup, yang menampilkan dua buah tombol yang, ketika diklik, akan membawa pengguna ke fragment lain yang menampilkan detail dari dua lagu yang berbeda. Setiap tombol memiliki listener yang mengarahkan pengguna ke fragment **DetailCategoryFragment** dan menyertakan beberapa data tambahan, seperti nama lagu, penyanyi, durasi, dan deskripsi lagu, yang dikirim menggunakan obyek **Bundle**. Setiap kali user mengklik salah satu tombol, obyek **Bundle** dibuat dan diisi dengan informasi tentang lagu yang bersangkutan. Kemudian, fragment ini menggunakan **NavController** untuk melakukan navigasi ke fragment **DetailCategoryFragment** sambil mengirimkan obyek **Bundle** yang berisi data tersebut.

```

13     private var _binding: FragmentDetailCategoryBinding? = null
14     private val binding get() = _binding!!
15
16     override fun onCreateView(
17         inflater: LayoutInflater, container: ViewGroup?,
18         savedInstanceState: Bundle?
19     ): View {
20
21         _binding = FragmentDetailCategoryBinding.inflate(inflater, container, attachToParent: false)
22         return binding.root
23     }
24
25     @SuppressWarnings("SetTextI18n")
26     override fun onViewCreated(view: View, savedInstanceState: Bundle?) {
27         super.onViewCreated(view, savedInstanceState)
28
29         val dataName = arguments?.getString(NextHomeFrag.EXTRA_NAME)
30         val dataSinger = arguments?.getString(NextHomeFrag.EXTRA_SINGER)
31         val dataDuration = arguments?.getString(NextHomeFrag.EXTRA_DURATION)
32         val dataDescription = arguments?.getString(NextHomeFrag.EXTRA_DESCRIPTION)
33
34         binding.tvCategoryName.text = dataName
35         binding.tvCategorySinger.text = dataSinger
36         binding.tvCategoryDuration.text = "Duration : $dataDuration"
37         binding.tvCategoryDescription.text = dataDescription
38     }
39
40     override fun onDestroy() {
41         super.onDestroy()
42         _binding = null
43     }

```

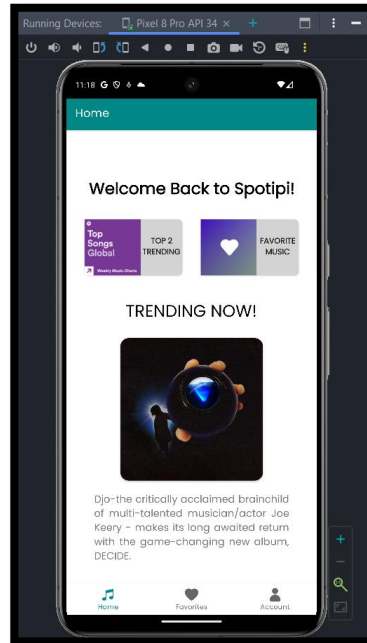
Gambar 13 Source code DetailCategoryFragment.kt

Fragment **DetailCategoryFragment** bertanggung jawab untuk menampilkan detail dari suatu kategori, dalam konteks ini adalah detail dari lagu. Ketika fragment ini dibuat, data mengenai nama lagu, penyanyi, durasi, dan deskripsi lagu diterima melalui obyek **Bundle** yang dikirim dari fragment sebelumnya, yaitu **NextHomeFrag**. Pada metode **onViewCreated**, data-data yang diterima dari bundle diambil menggunakan kunci-kunci yang telah ditentukan sebelumnya dalam kelas **NextHomeFrag**. Data tersebut kemudian ditampilkan di dalam tampilan **TextView** yang ada dalam layout fragment menggunakan metode **setText()**.

2. Screenshot Hasil Emulator

Berikut adalah hasil pada emulator aplikasi Music Player sederhana. Disini saya akan menampilkan per tampilannya.

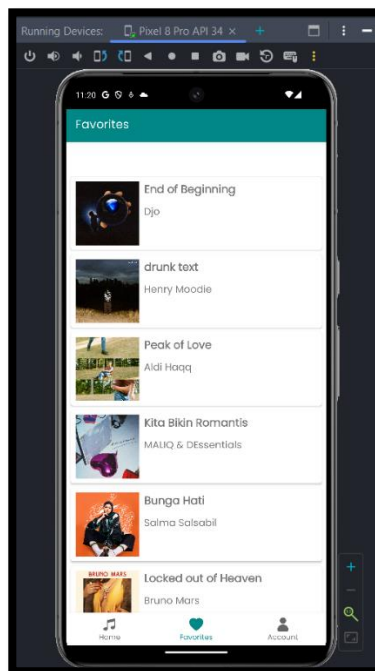
A. Home



Gambar 14 hasil tampilan home

Gambar diatas merupakan screenshot hasil dari emulator tampilan **Home**.

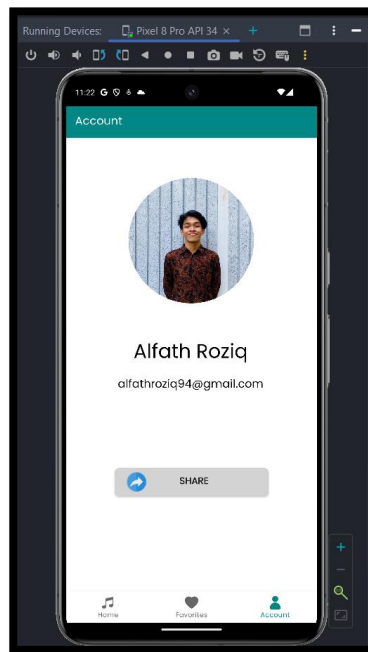
B. Favorite



Gambar 15 hasil tampilan favorite

Gambar diatas merupakan screenshot hasil dari emulator tampilan **Favorite**.

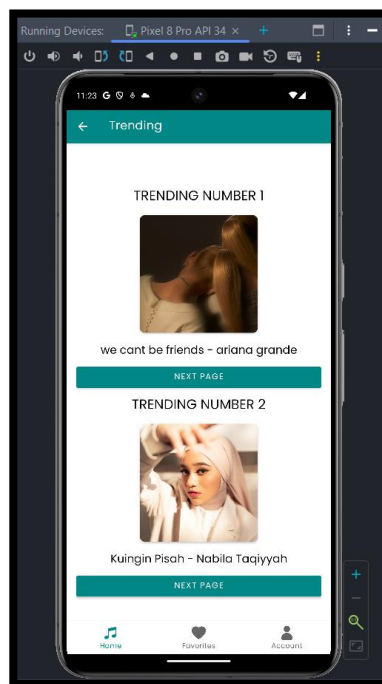
C. Account



Gambar 16 hasil tampilan account

Gambar diatas merupakan screenshot hasil dari emulator tampilan **Account**.

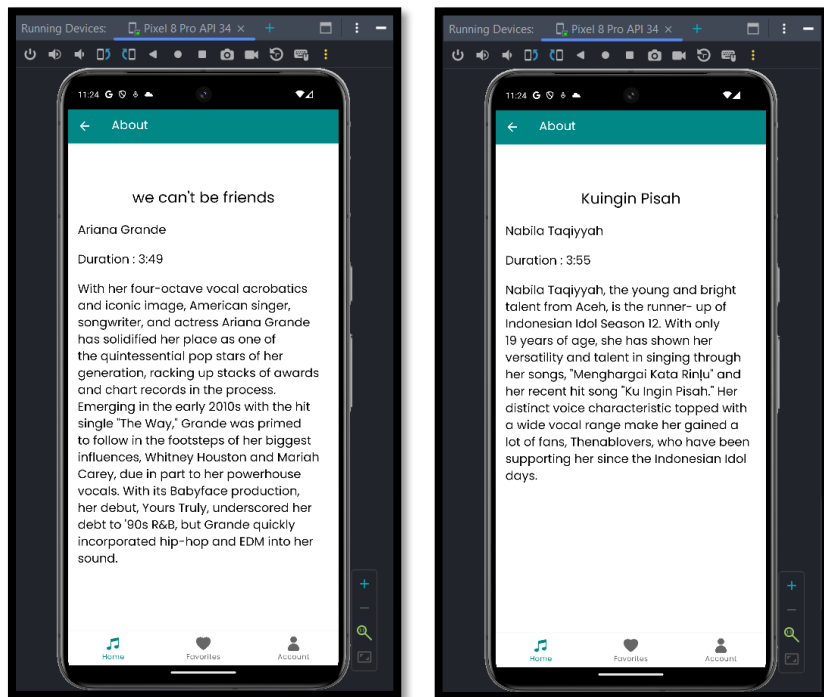
D. Destination Home → NextHomeFrag



Gambar 17 hasil tampilan nexthomefrag (trending)

Gambar diatas merupakan screenshot hasil dari emulator tampilan **Trending** yang mengirimkan destination home ke **trending**.

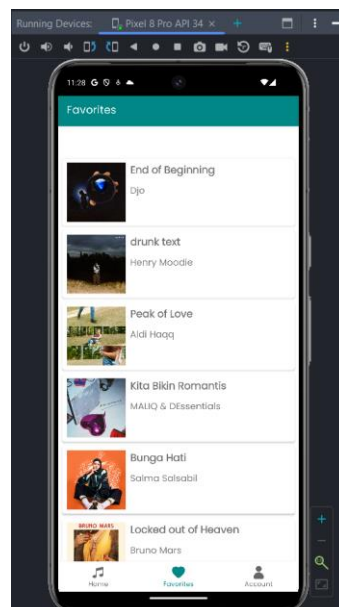
E. Destination NextHomeFrag Bundle → DetailCategoryFragment



Gambar 18,19 hasil tampilan detailcategoryfragment

Gambar diatas merupakan screenshot hasil dari emulator tampilan **About** yang mengirimkan destination home ke trending dengan membawa bundle dari **nextHomeFrag**.

F. Destination Home → Favorite



Gambar 20 hasil tampilan nexthomefrag (trending)

Gambar diatas merupakan screenshot hasil dari emulator tampilan **Favorite** yang mengirimkan destination home ke **Favorite**.

3. KESIMPULAN

Praktikum PAB Week 6 mempelajari penerapan **Bottom Navigation** dan **Navigation Component** dalam pengembangan aplikasi Android. Dalam praktikum ini, saya berhasil mengimplementasikan struktur UI dengan menggunakan **Bottom Navigation** yang memungkinkan user untuk dengan mudah berpindah antara berbagai bagian dalam aplikasi. Penggunaan **Navigation Component** membantu dalam mengatur navigasi antar fragmen dengan lebih efisien dan terstruktur, serta memungkinkan pengiriman data antar fragmen menggunakan **Bundle**. Dengan demikian, aplikasi **Music Player sederhana** yang saya buat dan kembangkan dalam praktikum ini menjadi lebih dinamis dan interaktif, memberikan pengguna pengalaman yang menyenangkan dan mudah dipahami.