

**DREAMING APPS**



**Disusun Oleh :**

**Faizah Via Fadhillah (312210460)**

**Dosen Pembimbing :**

**Donny Maulana, S.Kom., M.M.S.I.**

**FAKULTAS TEKNIK**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**UNIVERSITAS PELITA BANGSA**

**2024**

## A. Splash Launcher

### **MainLauncherSplashLogo.java:**

#### ❖ Kelas MainLauncherSplashLogo:

- Mewarisi AppCompatActivity
- Mengatur layout aktivitas ke activity\_launcher\_splash\_logo.xml
- Menyembunyikan status bar untuk tampilan layar penuh
- Menunda perpindahan ke aktivitas MainHomePage selama 3 detik
- Setelah 3 detik, membuka MainHomePage dan menutup Splash

### **activity\_launcher\_splash\_logo.xml:**

#### ❖ Layout MotionLayout:

- Memiliki dua elemen utama:
  - ConstraintLayout backgroundLayout sebagai latar belakang
  - ImageView blueLogo untuk menampilkan logo aplikasi
- Atribut elemen:
  - backgroundLayout:
    - Ukuran 200dp x 200dp
    - Warna latar belakang ditentukan di file colors.xml
  - blueLogo:
    - Ukuran 150dp x 150dp
    - Menampilkan gambar dari file drawable/dreaming
    - Diposisikan di tengah layer

### **Kesimpulan :**

- Kode ini menampilkan splash screen sederhana dengan logo aplikasi.
- Splash screen ini akan muncul selama 3 detik sebelum beralih ke aktivitas utama.

## **B. Home Page**

### **MainHomePage.java:**

#### ❖ Kelas MainHomePage:

- Mewarisi AppCompatActivity
- Mengatur layout aktivitas ke activity\_home.xml
- Menyembunyikan status bar untuk tampilan layar penuh
- Menginisialisasi tombol untuk setiap kartu menu
- Menyetel listener klik untuk setiap tombol untuk berpindah ke aktivitas yang sesuai

### **activity\_home.xml:**

#### ❖ Layout RelativeLayout:

- TextView dengan id "textView": Menampilkan judul "Menu Program" dengan ukuran besar dan warna putih
- LinearLayout utama: Mengatur tata letak untuk kartu secara vertikal dengan margin
- Tiga LinearLayout horizontal untuk kartu:
  - Setiap LinearLayout berisi tiga CardView untuk menampilkan opsi menu
- CardView:

- Memiliki layout LinearLayout dengan background putih
- Berisi ImageView untuk ikon dan TextView untuk label fitur
- Memiliki sudut tumpul dan efek bayangan

### **Kesimpulan:**

- Kode ini membuat halaman beranda yang rapi dan mudah digunakan dengan tampilan kartu yang menarik.
- Setiap kartu memiliki ikon dan teks yang jelas untuk membantu pengguna memahami fitur yang tersedia.

## **C. Hello**

### **MainHello.java:**

#### ❖ Kelas MainHello:

- Mewarisi AppCompatActivity
- Dalam metode onCreate():
  - Memanggil metode super.onCreate() untuk inisialisasi dasar
  - Mengatur layout aktivitas ke activity\_main.xml

### **activity\_main.xml:**

#### ❖ Layout ConstraintLayout:

- TextView:
  - Menampilkan teks "Apa kabar?" yang diambil dari string resource "@string/apakabar001"
  - Memiliki warna teks biru yang diambil dari color resource "@color/blue"

- Diposisikan di tengah layar dengan constraint ke semua sisi layar

### **Kesimpulan:**

- Kode ini membuat aktivitas dasar yang menampilkan teks sederhana.
- ConstraintLayout digunakan untuk mengatur posisi elemen UI dengan mudah.

## **D. Count**

### **MainCount.java:**

- Kelas MainCount:
  - Mewarisi AppCompatActivity
  - Memiliki variabel mShowCount (TextView) dan currentCount (int)
- Metode onCreate:
  - Mengatur layout aktivitas ke activity\_count.xml
  - Menghubungkan TextView mShowCount dengan elemen di layout
- Metode countUp:
  - Meningkatkan nilai currentCount
  - Menampilkan nilai currentCount di TextView mShowCount
- Metode showToast:
  - Menampilkan pesan Toast singkat

### **activity\_count.xml:**

- Layout ConstraintLayout:
  - Tiga elemen utama:
    - Button button\_toast untuk menampilkan Toast
    - Button button\_count untuk menjalankan fungsi countUp
    - TextView show\_count untuk menampilkan nilai perhitungan
- Atribut elemen:
  - Button:
    - Warna latar belakang colorPrimary
    - Teks sesuai string yang didefinisikan di strings.xml
    - Penanganan klik dengan onClick
  - TextView:
    - Warna latar belakang kuning (#FFFF00)
    - Teks awal "0"
    - Ukuran teks besar (160sp)
    - Teks berwarna colorPrimary
    - Teks tebal

### **Kesimpulan:**

- Kode ini menampilkan aplikasi Android dengan dua tombol dan sebuah TextView.
- Tombol button\_toast menampilkan pesan Toast.

- Tombol `button_count` menghitung angka dan menampilkannya di `TextView`.

## E. News

### **ScrollingIcecold.java:**

#### ❖ Kelas `ScrollingIcecold`:

- Mewarisi `AppCompatActivity`
- Dalam metode `onCreate()`:
  - Memanggil metode `super.onCreate()` untuk inisialisasi dasar
  - Mengatur layout aktivitas ke `activity_scrollicecold.xml`

### **activity\_scrollicecold.xml:**

#### ❖ Layout `RelativeLayout`:

- `TextView` dengan id `"article_heading"`:
  - Menampilkan judul artikel dengan gaya teks besar dan tebal
  - Memiliki background warna `primary` dan teks warna putih
- `ScrollView`:
  - Memungkinkan pengguna menggulir konten yang lebih panjang
  - Berisi `LinearLayout` untuk menyusun elemen UI
- `LinearLayout`:
  - Mengatur elemen UI secara vertikal
  - Berisi `TextView` untuk subjudul dan teks artikel

- TextView dengan id "article\_subheading":
  - Menampilkan subjudul artikel
- TextView dengan id "article":
  - Menampilkan isi artikel dengan link web otomatis dan jarak antar baris yang diatur

### **Kesimpulan:**

- Kode ini membuat aktivitas untuk menampilkan artikel dengan tampilan yang rapi dan mudah dibaca.
- ScrollView memungkinkan pengguna untuk membaca artikel yang panjang tanpa terpotong.

## **F. Alarm**

Laporan ini membahas dua kelas utama dalam proyek Android yang berfungsi sebagai pengatur alarm. Dua kelas tersebut adalah AlarmActivity dan AlarmReceiver. Program ini juga memiliki layout XML yang disebut activity\_sleepalarm.xml.

### **1. AlarmActivity**

AlarmActivity bertanggung jawab untuk menangani aktivitas utama aplikasi dan mengatur alarm. Berikut adalah beberapa poin penting dari kelas ini:

- Inisialisasi Komponen:
  - TimePicker, AlarmManager, dan PendingIntent diinisialisasi dalam metode onCreate().
- Toggle Button untuk Alarm:
  - Metode OnToggleClicked(View view) diimplementasikan untuk menangani perubahan status toggle button. Jika toggle



diaktifkan, alarm diatur menggunakan `AlarmManager` untuk memulai `AlarmReceiver`. Jika toggle dinonaktifkan, alarm dibatalkan.

- **Penanganan Waktu:**
  - Waktu alarm diatur menggunakan `Calendar` dengan mengambil input dari `TimePicker`. `PendingIntent` digunakan untuk menjadwalkan pemutaran `AlarmReceiver` pada waktu yang ditentukan.

## **2. AlarmReceiver**

`AlarmReceiver` adalah penerima siaran (broadcast receiver) yang menanggapi pemutaran alarm yang telah dijadwalkan. Berikut adalah beberapa poin kunci dari kelas ini:

- **Tindakan pada Penerimaan Alarm:**
  - Pada penerimaan alarm, kelas ini memberikan efek suara menggunakan `RingtoneManager` dan vibrasi menggunakan `Vibrator`.
- **Tampilkan Pesan Toast:**
  - Menampilkan pesan toast untuk memberi tahu pengguna bahwa alarm telah aktif.

## **3. Layout XML (activity\_sleepalarm.xml)**

- **TimePicker:**
  - Digunakan untuk memilih waktu alarm dengan tata letak yang diatur di tengah.
- **ToggleButton:**
  - Digunakan untuk mengaktifkan atau menonaktifkan alarm dengan memberikan umpan balik visual.

## **G. Fibonacci**

### **MainActivity.java:**

- Menampung logika utama aplikasi.
- Mengelola tampilan dan interaksi pengguna.
- Menghitung nilai Fibonacci.

### **activity\_toast.xml:**

- Mendefinisikan layout antarmuka pengguna, termasuk:
  - TextView untuk menampilkan nilai Fibonacci saat ini.
  - EditText untuk memasukkan batas Fibonacci.
  - Tombol untuk menghitung, menampilkan pesan tentang deret Fibonacci, dan mereset nilai.

### **Fitur:**

- Pengguna memasukkan batas Fibonacci yang diinginkan pada EditText.
- Tombol "Count" menghitung nilai Fibonacci berikutnya hingga mencapai batas.
- Tombol "Show Fibonacci" menampilkan pesan tentang deret Fibonacci.
- Tombol "Reset" mengembalikan nilai Fibonacci ke 0 dan mengosongkan input batas.
- Nilai Fibonacci ditampilkan dengan warna yang bergantian untuk setiap nilai genap dan ganjil.

### **Kesimpulan:**

Aplikasi ini berhasil menampilkan deret Fibonacci hingga batas tertentu dan memberikan opsi untuk menghitung, menampilkan pesan, dan mereset nilai. Penggunaan warna yang bergantian membantu membedakan nilai genap dan ganjil dalam deret.

## **H. Chat**

Laporan ini membahas dua kelas utama dalam proyek Android yang menggunakan Android Studio. Dua kelas tersebut adalah MainActivityOne dan SecondActivity. Program ini juga menggunakan dua layout XML, yaitu activity\_satu.xml dan activity\_dua.xml.

### **1. MainActivityOne**

MainActivityOne adalah aktivitas pertama dalam aplikasi. Berikut adalah beberapa poin penting dari kelas ini:

- Inisialisasi Tampilan:
  - Komponen antarmuka pengguna diinisialisasi dalam metode onCreate(), termasuk EditText, TextView untuk header dan balasan, dan beberapa variabel lainnya.
- Tombol Klik Aksi:
  - Terdapat metode LaunchSecondActivity(View view) yang dipanggil saat tombol diklik. Metode ini membuat objek Intent untuk memulai SecondActivity dan mengirim pesan tambahan.
- Penanganan Balasan:
  - Metode onActivityResult() digunakan untuk menangani hasil dari SecondActivity. Jika hasilnya ok, maka balasan yang diterima ditampilkan di tampilan.

### **2. SecondActivity**

SecondActivity adalah aktivitas kedua yang dimulai sebagai respons terhadap pemanggilan dari MainActivityOne. Berikut adalah beberapa poin kunci dari kelas ini:

- Menerima Pesan:

- Pesan yang dikirim dari MainActivityOne diterima melalui objek Intent pada metode onCreate().
- Tombol Klik Aksi:
  - Metode returnReply(View view) dipanggil saat tombol "Kirim Balasan" diklik. Balasan yang dimasukkan oleh pengguna dikirim kembali ke MainActivityOne sebagai hasil.
- Menutup Aktivitas:
  - Aktivitas ini ditutup setelah mengirim balasan dengan memanggil finish().

### **3. Layout XML**

#### **activity\_satu.xml:**

- Mendefinisikan tata letak antarmuka untuk MainActivityOne.
- Berisi elemen-elemen seperti EditText untuk memasukkan pesan, TextView untuk menampilkan balasan, dan tombol untuk memulai aktivitas kedua.

#### **activity\_dua.xml:**

- Mendefinisikan tata letak antarmuka untuk SecondActivity.
- Terdapat EditText untuk memasukkan balasan dan tombol untuk mengirim balasan kembali ke aktivitas pertama.

## **I. Maps**

#### **MapActivity.java:**

- Menampilkan peta menggunakan Intent dan Uri.
- Menentukan koordinat lokasi awal (37.7749, -122.4194).
- Memanggil metode showMap untuk menampilkan peta di aplikasi peta yang tersedia.

#### **activity\_map.xml:**

- Mendefinisikan layout tampilan MapActivity.
- Berisi RelativeLayout sebagai layout utama.
- Tidak ada elemen visual lain yang didefinisikan.

**Fungsi:**

- Menampilkan peta pada layar.
- Menggunakan koordinat lokasi yang ditentukan dalam kode.
- Memanfaatkan aplikasi peta eksternal untuk menampilkan peta.

**Kesimpulan:**

MapActivity berfungsi untuk menampilkan peta dari aplikasi peta eksternal menggunakan koordinat lokasi yang ditentukan. Namun, tampilannya masih kosong dan perlu ditambahkan elemen visual untuk melengkapi fungsionalitasnya.

## **J. Movie**

### **1. Tab Activity**

- Tab Activity digunakan untuk mengorganisir konten aplikasi dalam tab yang dapat diakses pengguna.
- Menyediakan antarmuka pengguna yang bersih dan terstruktur dengan kategori atau fitur yang dapat dipilih melalui tab.

**Implementasi:**

- Dalam contoh kode, TabActivity menggunakan TabLayout dan ViewPager untuk menampilkan kategori film seperti Action, Fantasy, dan Romantic.
- Menggunakan adapter Halaman untuk mengelola konten dalam ViewPager.

## **2. Fragment**

- Fragment adalah komponen independen yang dapat digunakan untuk membuat antarmuka pengguna modular dan dapat digunakan kembali.
- Membantu dalam memecah tampilan dan logika menjadi bagian-bagian yang lebih kecil dan dapat dikelola.

### **Implementasi:**

- Tabfragment1Activity adalah contoh fragment yang menampilkan daftar film dalam kategori "Action".
- Setiap film direpresentasikan oleh elemen tampilan seperti gambar, judul, dan deskripsi.
- Menggunakan OnClickListener untuk memulai aktivitas Exoplayer1Activity ketika gambar film diklik.

## **3. ExoPlayer**

- ExoPlayer adalah pustaka Android untuk memutar audio dan video dengan fitur canggih.
- Memberikan kontrol yang baik atas pemutaran, buffering, dan fitur-fitur lainnya.

### **Implementasi:**

- Exoplayer1Activity menggunakan ExoPlayer untuk memutar video.
- Menggunakan StyledPlayerView sebagai antarmuka pemutar video.
- URI video diperoleh dari informasi yang dikirimkan melalui Intent atau secara statis dalam metode getVideoUri.

## **4. Kesimpulan**

Penggunaan Tab Activity, Fragment, dan ExoPlayer dapat meningkatkan pengalaman pengguna dengan menyediakan antarmuka yang terstruktur, modular, dan memungkinkan pemutaran video yang dikelola dengan baik.

Tab Activity membantu dalam mengatur konten, sedangkan Fragment memfasilitasi modularitas dan pengelolaan tampilan.

ExoPlayer memberikan kontrol yang kuat untuk memutar audio dan video dalam aplikasi Android.