

LAPORAN PRAKTIKUM
PENGEMBANGAN APLIKASI BERGERAK
PRAKTIKUM 10 – STORAGE



L0122034
BINTANG HARIDA RAMADHAN

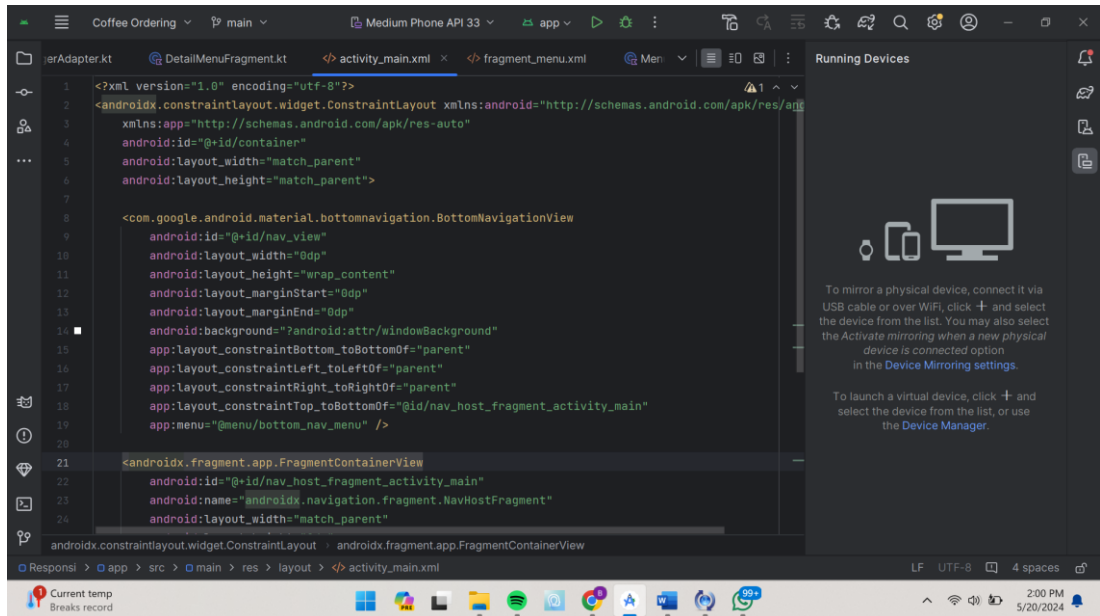
PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS DATA
UNIVERSITAS SEBELAS MARET

2024

BAB I

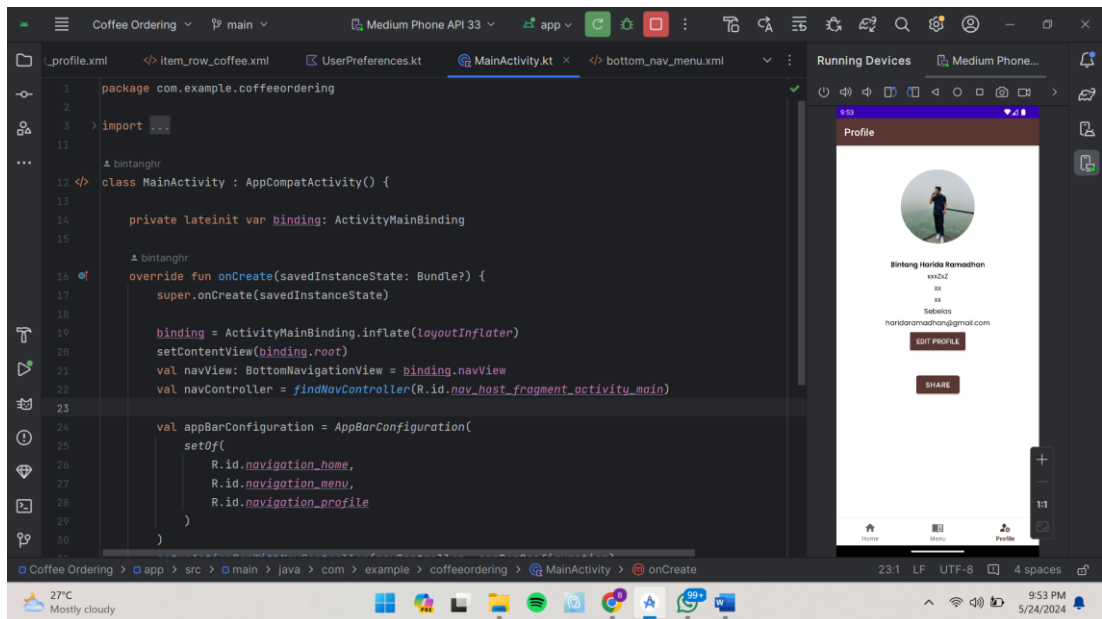
SOURCE CODE

1. File “activity_main.xml”



Source code di atas mendefinisikan layout utama untuk sebuah activity menggunakan ConstraintLayout sebagai root layout. Di dalamnya terdapat dua elemen utama, yaitu BottomNavigationView dan NavHostFragment. BottomNavigationView dengan ID nav_view terletak di bagian bawah layar dan digunakan untuk navigasi antar fragment dalam aplikasi, dengan menu yang ditentukan di file bottom_nav_menu.xml. NavHostFragment dengan ID nav_host_fragment_activity_main berfungsi sebagai container untuk menampilkan fragment sesuai dengan navigasi yang ditentukan dalam file mobile_navigation.xml. NavHostFragment ini diatur sebagai default navigation host menggunakan atribut app.defaultNavHost="true", dan posisinya diatur dengan constraint agar berada di atas BottomNavigationView serta memenuhi sisa ruang yang tersedia di layar.

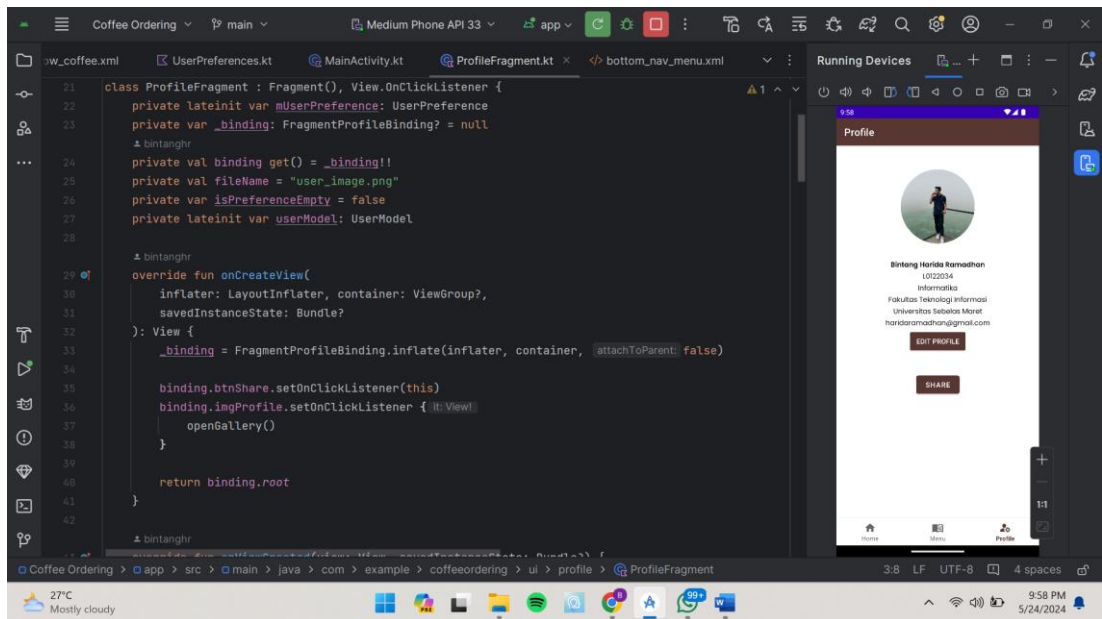
2. File “MainActivity.kt”



MainActivity adalah aktifitas utama dari aplikasi dengan menampilkan navigasi dasar menggunakan BottomNavigationView. Pada metode onCreate, ActivityMainBinding diinisialisasi untuk menghubungkan layout dengan aktivitas ini. Layout diatur dengan memanggil setContentView pada root view dari binding. BottomNavigationView (navView) diambil dari binding dan dihubungkan dengan NavController, yang dikelola oleh NavHostFragment yang terdapat dalam layout utama (activity_main).

Sebuah AppBarConfiguration dibuat untuk menentukan fragment mana saja yang harus dianggap sebagai destinasi tingkat atas, dalam hal ini termasuk R.id.navigation_home, R.id.navigation_menu, dan R.id.navigation_profile. Metode setupActionBarWithNavController digunakan untuk menyelaraskan ActionBar dengan NavController, dan navView.setupWithNavController menghubungkan BottomNavigationView dengan NavController, memungkinkan navigasi diatur sesuai dengan destinasi yang ditentukan. Fungsi onSupportNavigateUp dioverride untuk mendukung navigasi naik ketika tombol naik di toolbar ditekan, menggunakan NavController untuk menangani navigasi.

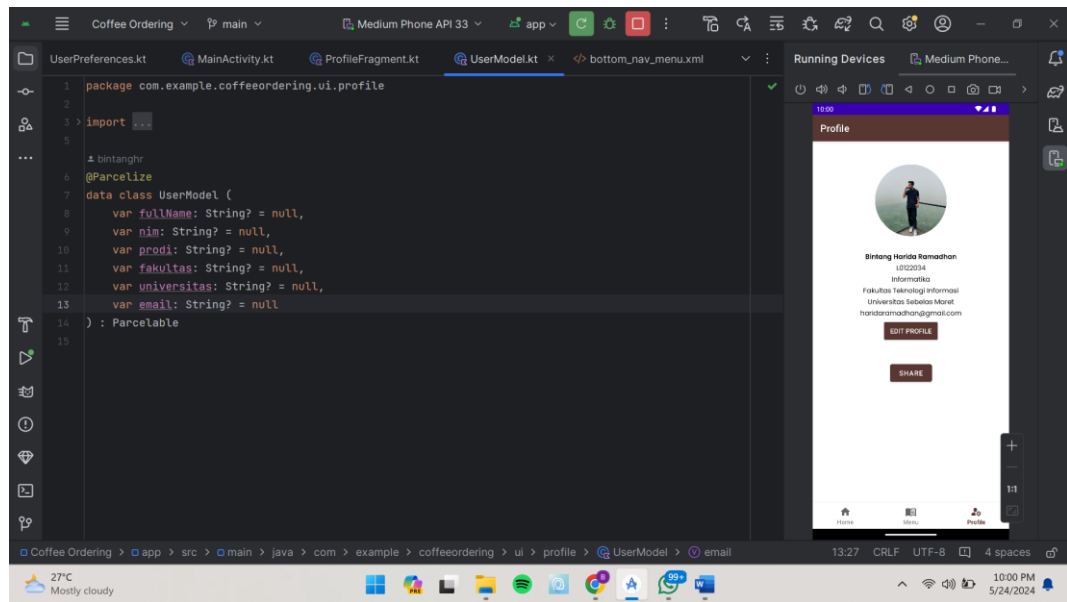
3. File “ProfileFragment.kt”



Source Code di atas merupakan implementasi dari ProfileFragment yang mengelola tampilan dan interaksi fragment profil. Fragment ini menggunakan View Binding untuk mengakses tampilan UI dan menyimpan preferensi pengguna menggunakan UserPreference. Pada onCreateView, binding diinisialisasi dan pengaturan listener untuk tombol serta gambar profil dilakukan. Metode showExistingPreference mengambil data pengguna yang tersimpan dan menampilkannya di UI dengan bantuan metode populateView.

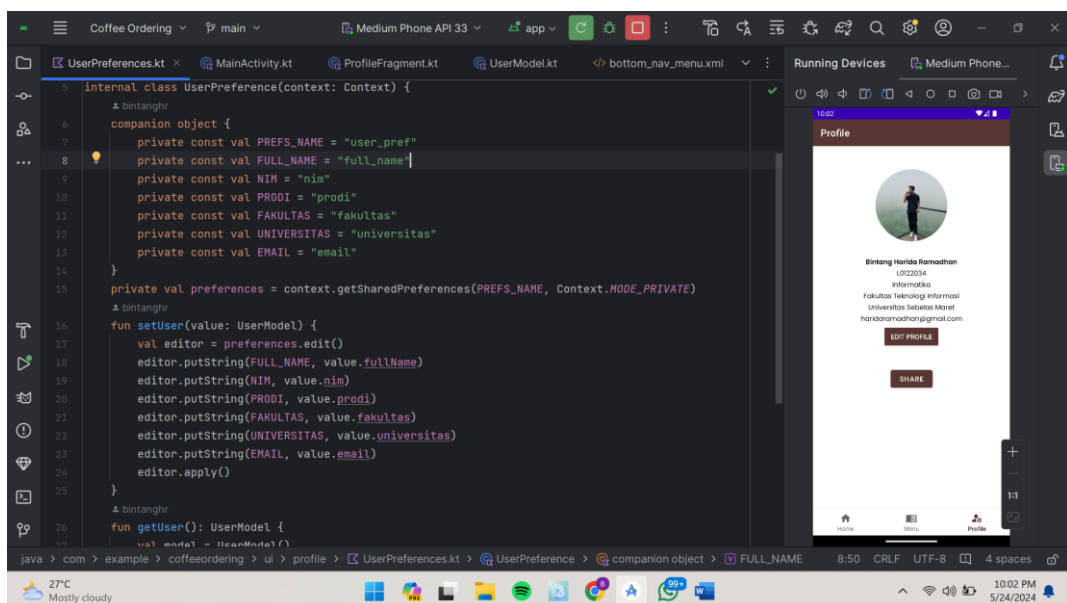
Fragment juga memiliki fungsi untuk membuka galeri dan mengambil gambar profil pengguna yang baru, menyimpannya di penyimpanan internal, dan memuatnya kembali saat diperlukan. Ketika tombol "Share" ditekan, aplikasi akan mempersiapkan dan memulai intent berbagi data pengguna melalui email. Tombol "Save" digunakan untuk menavigasi ke FormUserPreferenceFragment untuk menambahkan atau mengedit preferensi pengguna berdasarkan kondisi apakah preferensi pengguna sudah ada atau belum. Fragment menggunakan NavController untuk mengelola navigasi antar fragment dalam aplikasi.

4. File "UserModel.kt"



Source Code di atas mendefinisikan data class, yang digunakan untuk menyimpan informasi profil pengguna dalam aplikasi. Kelas ini diimplementasikan menggunakan anotasi `@Parcelize` dari library `kotlin-parcelize`, yang secara otomatis menghasilkan metode untuk membuat objek `Parcelable`. `Parcelable` adalah antarmuka Android yang digunakan untuk mengemas objek kompleks menjadi bentuk yang lebih sederhana sehingga dapat dipindahkan antar komponen aplikasi, seperti fragment dan activity. `UserModel` memiliki beberapa properti: `fullName`, `nim`, `prodi`, `fakultas`, `universitas`, dan `email`, yang semuanya bersifat nullable (`String?`) dan memiliki nilai default `null`.

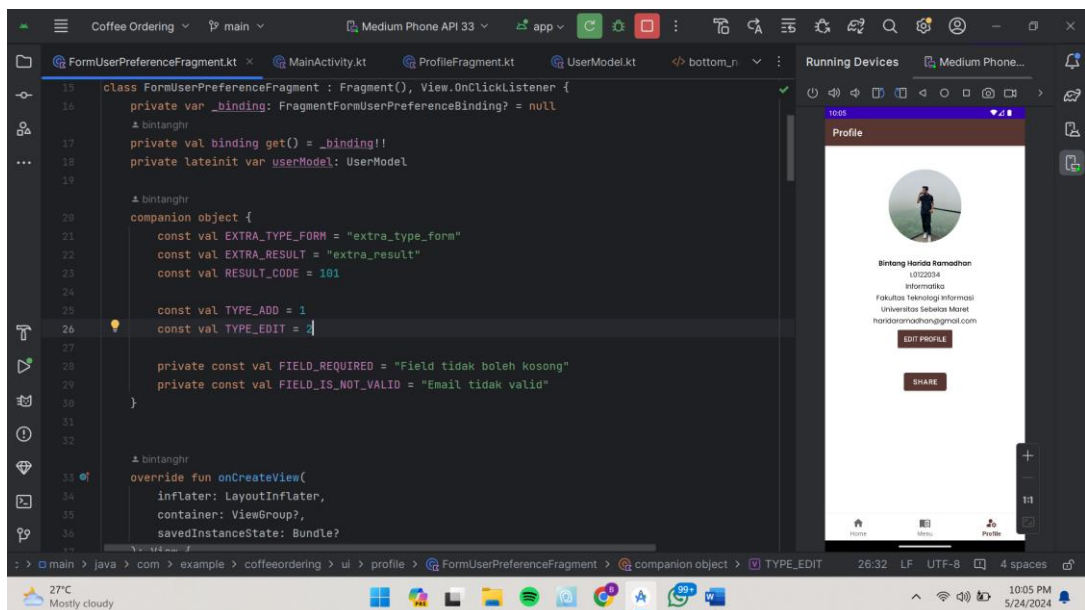
5. File “UserPreferences.kt”



Source Code di atas mendefinisikan kelas `UserPreference` yang digunakan untuk menyimpan data pengguna ke `SharedPreferences` dan mengambil data pengguna dari `SharedPreferences` di Android. `SharedPreferences` adalah mekanisme penyimpanan data sederhana yang digunakan untuk menyimpan pasangan kunci-nilai. `UserPreference` memiliki konstanta untuk nama file `SharedPreferences` (`PREFS_NAME`) dan kunci untuk setiap atribut pengguna seperti `FULL_NAME`, `NIM`, `PRODI`, `FAKULTAS`, `UNIVERSITAS`, dan `EMAIL`.

Kelas ini memiliki dua fungsi utama: `setUser` dan `getUser`. Fungsi `setUser` mengambil objek `UserModel` dan menyimpan setiap atributnya ke `SharedPreferences` menggunakan editor. Fungsi `getUser` mengambil data dari `SharedPreferences` dan mengembalikannya sebagai objek `UserModel`. Dengan cara ini, `UserPreference` memungkinkan penyimpanan dan pengambilan data pengguna secara persisten dalam aplikasi, memudahkan pengelolaan data pengguna di berbagai bagian aplikasi.

6. File “FormUserPreferenceFragment.kt”



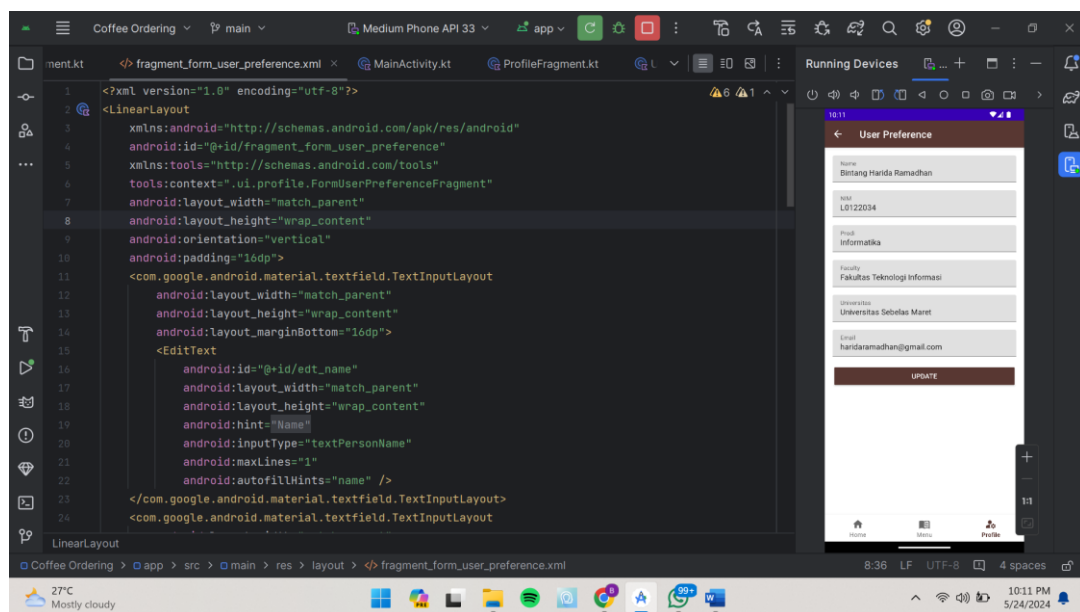
Source Code mendefinisikan sebuah fragmen yang bertanggung jawab untuk menampilkan formulir preferensi pengguna. Saat fragment dibuat, Fragment ini mengambil data yang diperlukan dari argumen yang diteruskan, seperti tipe formulir (tambah atau edit) dan data pengguna jika dalam mode edit. Kemudian, fragmen menyesuaikan tampilan dan perilakunya berdasarkan tipe formulir tersebut, seperti menampilkan judul dan teks tombol yang sesuai.

Metode onCreateView digunakan untuk menginisialisasi tampilan fragment, sedangkan onCreateView dipanggil setelah tampilan telah dibuat, di mana fragmen menetapkan teks dan tindakan tombol berdasarkan argumen yang diterima. Metode showPreferenceInForm digunakan untuk menampilkan data pengguna saat mode edit.

Ketika pengguna mengklik tombol simpan, metode onClick memvalidasi input pengguna dan menyimpan data ke SharedPreferences menggunakan kelas UserPreference. Setelah itu, hasilnya dikirim kembali ke fragmen sebelumnya, dan fragmen ini menavigasi kembali ke halaman sebelumnya dengan menggunakan popBackStack.

Kelas UserPreference digunakan untuk menyimpan dan mengambil data pengguna dari SharedPreferences, sehingga data pengguna dapat diakses dan disimpan secara lokal.

7. File “fragment_form_user_preference.xml”



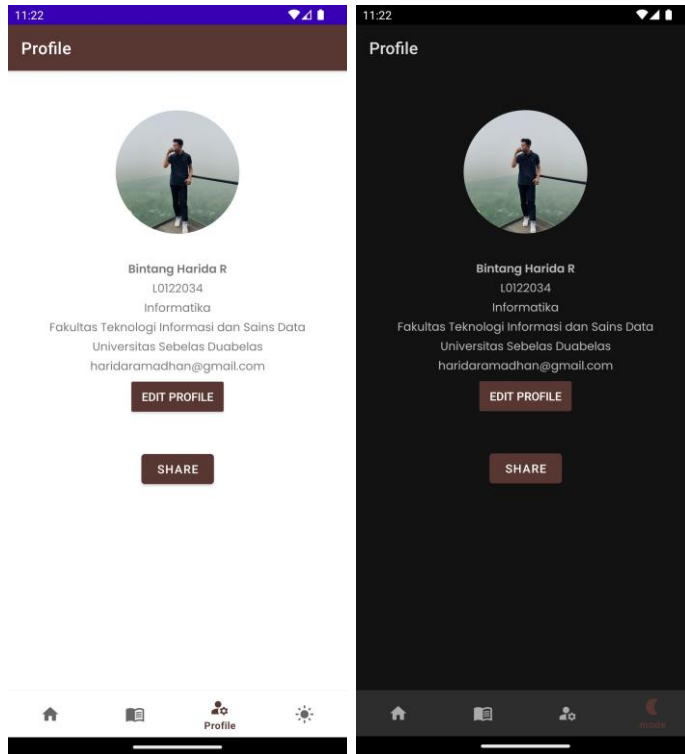
Source Code XML tersebut mendefinisikan layout untuk tampilan formulir preferensi pengguna dalam sebuah LinearLayout. Setiap TextInputLayout digunakan untuk mengelompokkan label dan input teks, dengan setiap EditText yang mewakili bidang input. Bidang input ini memiliki hint untuk memberi petunjuk kepada pengguna tentang jenis data yang diharapkan. Di bagian bawah, terdapat tombol "Save" yang digunakan untuk menyimpan data yang dimasukkan oleh pengguna. Dengan tata letak ini, pengguna dapat memasukkan data seperti nama, NIM, program studi, fakultas, universitas, dan alamat email mereka.

BAB II

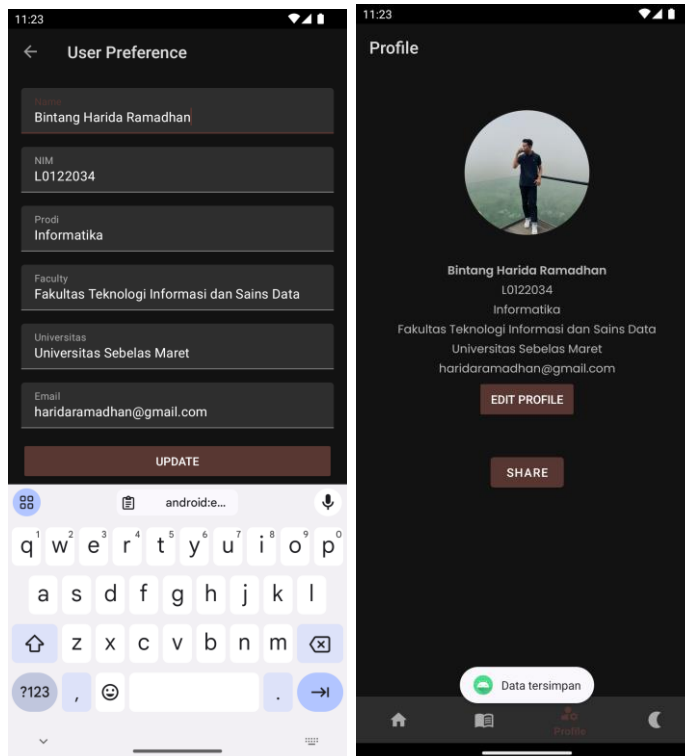
USER INTERFACE

1. Profile Fragment

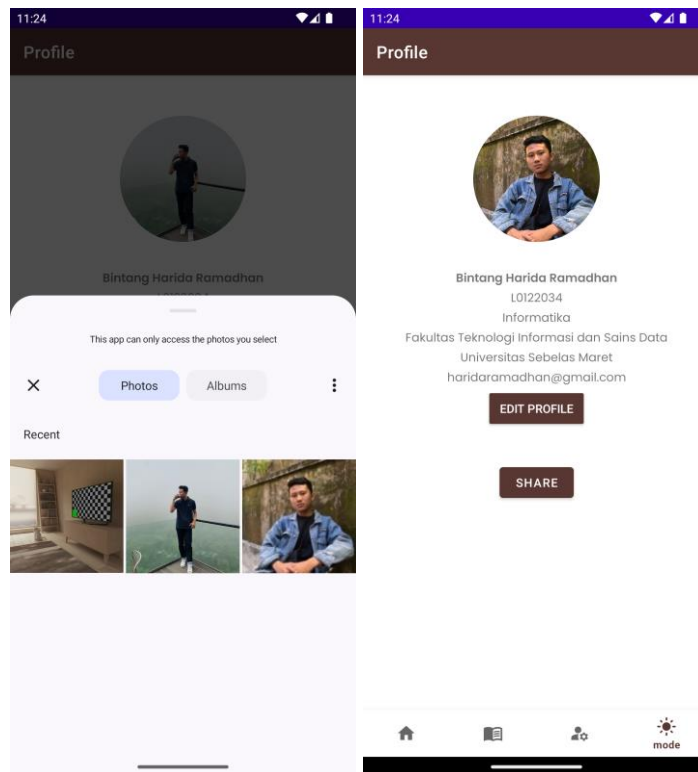
- Sebelum Edit



- Edit Data Diri



- Edit Foto



Pada fragment ini ditampilkan foto dan data diri saya. Data diri dan foto dapat diubah dengan cara menekan tombol “Edit Profile” untuk mengedit data diri dan menekan foto profile untuk mengedit. Selain itu juga terdapat button Share yang akan melakukan intent ke default sharing application pada android.

BAB III

KESIMPULAN

Untuk dapat menyimpan data diri dari aplikasi ini diperlukan beberapa komponen yang saling berkaitan. Pertama, terdapat model data UserModel yang mencakup berbagai properti seperti nama lengkap, NIM, prodi, fakultas, universitas, dan alamat email. Kelas UserPreference digunakan untuk menyimpan dan mengambil preferensi pengguna menggunakan Shared Preferences, memungkinkan aplikasi untuk menyimpan data profil secara lokal di perangkat.

Fragment ProfileFragment bertanggung jawab untuk menampilkan profil pengguna, menampilkan data dari UserModel, serta memungkinkan pengguna untuk membagikan profil mereka melalui email. Pengguna juga dapat memilih gambar profil dari galeri melalui fragment ini. Sementara itu, FormUserPreferenceFragment mengelola formulir untuk memasukkan atau mengedit preferensi pengguna. Ini memungkinkan pengguna untuk memasukkan atau mengedit data seperti nama, NIM, dan lainnya, dengan data yang dimasukkan disimpan dalam UserPreference.