

**PERANCANGAN APLIKASI FAMCARE BERBASIS ANDROID UNTUK
PENCARIAN DAN PENYEWAAN PENGASUH**

Laporan ini Dibuat untuk Memenuhi Ujian Tengah Semester
Mata Kuliah Praktikum Pemrograman Mobile



Disusun oleh:

Aji Wijayatama	NIM	2110512135
Stefanny	NIM	2110512137
Ni Wayan Giri Puspawati	NIM	2110512143
Novi Nur Rahmawati	NIM	2110512161
Siti Aisyah Nur Alim	NIM	2110512166

Kelas: B

Dosen Pengampu Mata Kuliah:

Andhika Octa Indarso, S.Kom., MMSI.

**PROGRAM STUDI S1 SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER
UNIVERSITAS PEMBANGUNAN NASIONAL "VETERAN" JAKARTA
JAKARTA
2024**

KATA PENGANTAR

Puji syukur kehadirat Tuhan Yang Maha Esa atas karuniaNya sehingga kami dapat menyelesaikan proyek ujian akhir semester yang berjudul "Perancangan Aplikasi FamCare Berbasis Android untuk Pencarian dan Penyewaan Pengasuh". Adapun tujuan dari penulisan laporan ini adalah untuk memenuhi ujian akhir semester pada mata kuliah Pemrograman Mobile.

Kami mengucapkan terima kasih kepada Bapak Andhika Octa Indarso, S.Kom., MMSI., selaku dosen pada mata kuliah Pemrograman Mobile yang telah mengajarkan kami dan memberikan tugas ini sehingga dapat menambah pengetahuan dan wawasan sesuai dengan bidang studi yang kami tekuni. Bimbingan dan ilmu yang telah beliau berikan sangat berarti bagi kami dalam menyelesaikan proyek ini.

Kami juga mengucapkan terima kasih kepada semua pihak yang telah berkontribusi dan memberikan dukungan, sehingga tugas ini dapat terselesaikan dengan baik. Tanpa bantuan dan dukungan dari berbagai pihak, penyusunan proyek ini tidak akan berjalan dengan lancar.

Kami menyadari bahwa tugas ini masih jauh dari kata sempurna. Oleh karena itu, kritik dan saran yang membangun akan kami nantikan demi kesempurnaan tugas ini di masa yang akan datang. Kami berharap laporan ini dapat bermanfaat bagi pembaca dan memberikan kontribusi positif bagi pengembangan aplikasi mobile di bidang pencarian dan penyewaan pengasuh. Akhir kata, kami ucapan terima kasih.

Jakarta, 24 Juni 2024

Penulis

DAFTAR ISI

HALAMAN COVER	i
KATA PENGANTAR	ii
DAFTAR ISI	iii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan	2
1.4 Manfaat	2
1.5 Ruang Lingkup	2
1.6 Peran	3
BAB II PEMBAHASAN	4
2.1 Workflow Aplikasi	4
2.2 Implementasi Kode	6
2.2.1 Splash Screen	6
2.2.2 Onboarding	7
2.2.3 Register	12
2.2.4 Login	19
2.2.5 Discover	26
2.2.6 Booking, History, dan Chat	65
2.2.7 Facilities	124
2.2.8 Profile	147
2.3 Tampilan	193
2.3.1 Splash Screen	193
2.3.2 OnBoarding	193
2.3.3 Register	193
2.3.4 Log In	194
2.3.5 Discover	194
2.3.6 Detail Post	195
2.3.7 Booking	195
2.3.8 History dan Chat	196
2.3.9 Facilities	197
2.3.10 Profile	197
BAB III PENUTUP	200
3.1 Kesimpulan	200
3.2 Saran	200
DAFTAR PUSTAKA	202
LAMPIRAN	203

BAB I

PENDAHULUAN

1.1 Latar Belakang

Di era modern seperti saat ini, dengan gaya hidup yang semakin sibuk dan mobilitas yang tinggi, banyak individu menemui tantangan dalam merawat anggota keluarga yang membutuhkan perhatian khusus seperti bayi, anak-anak, dan lansia. Menurut sebuah studi oleh Kementerian Pemberdayaan Perempuan dan Perlindungan Anak Republik Indonesia (2020), peningkatan mobilitas dan kesibukan dalam kehidupan modern telah menyebabkan berkurangnya waktu yang tersedia bagi keluarga untuk merawat anggota yang membutuhkan perhatian khusus. Ketersediaan waktu yang terbatas seringkali menyulitkan dalam mencari pengasuh yang terpercaya dan berkualitas.

Selain itu, kekhawatiran akan keselamatan dan kenyamanan anggota keluarga yang ditinggalkan seringkali menjadi beban pikiran orang tua dan kerap menjadi faktor yang menghambat produktivitas. Penelitian oleh Amelia dan Alini (2020) menunjukkan bahwa ketidakpastian tentang keselamatan dan kesejahteraan anak-anak atau lansia yang ditinggalkan dapat menyebabkan stres dan menurunkan produktivitas orang tua di tempat kerja. Padatnya jadwal seorang individu juga dapat menyebabkan orang terdekat seperti anak dan orang tua mempunyai kemungkinan besar mengalami kesepian. Keadaan tersebut dapat menyebabkan konflik berkelanjutan yang berdampak buruk pada hubungan antar anggota keluarga.

Berdasarkan Data Susenas (Survei Sosial Ekonomi Nasional) tahun 2020, masih terdapat 3,64 persen pengasuhan tidak layak di Indonesia dan sebanyak 15 provinsi berada di atas angka tersebut. Menurut Ilmi & Qhoiriah (2016), di sisi pengguna juga muncul permasalahan di mana beberapa keluarga tidak mempunyai waktu untuk mencari asisten rumah tangga dengan datang langsung ke penyedia jasa asisten rumah tangga. Hal tersebut dialami bagi orang-orang yang memiliki aktivitas tinggi termasuk wanita karir. Berdasarkan permasalahan tersebut, maka perlu adanya suatu aplikasi berbasis android yang memanfaatkan internet dalam mencari pengasuh serta mengetahui detail profil calon pengasuh. Namun, mencari informasi tentang pengasuh anak maupun orang tua sulit untuk dilakukan karena jumlah agensi pengasuh cukup sedikit dan lokasi agensi tersebut tidak mudah ditemukan. Oleh karena itu, FamCare hadir untuk memberikan kemudahan dalam pencarian pengasuh. FamCare menyediakan berbagai informasi seperti profil pengasuh, kemampuan, biaya jasa, serta lokasi pengasuh tersebut.

Aplikasi ini dapat diakses melalui platform android sehingga memudahkan akses pengguna untuk mencari informasi tentang pengasuh. Selain itu, aplikasi ini dirancang dengan bahasa pemrograman Kotlin, yang merupakan bahasa pemrograman native yang didukung oleh Google, lebih simpel, open source, dan mudah diimplementasikan. Tampilan antarmuka yang menarik juga mendukung proses pencarian pengasuh menjadi lebih interaktif. Selain itu pengguna juga dapat berinteraksi langsung dengan pengasuh melalui fitur chat. Dengan demikian, aplikasi FamCare dapat menjadi solusi efektif dalam mencari informasi pengasuh.

1.2 Rumusan Masalah

Berikut adalah rumusan masalah dari pembuatan aplikasi FamCare :

- 1) Bagaimana merancang antarmuka pengguna (UI) yang responsif dan user-friendly menggunakan bahasa pemrograman Kotlin?
- 2) Bagaimana mengimplementasikan algoritma pencarian dan filter pengasuh yang efisien menggunakan Firebase Firestore untuk menyimpan dan mengambil data?
- 3) Apa saja fitur inti yang perlu dikembangkan dalam aplikasi FamCare untuk memastikan fungsionalitas yang lengkap dan optimal?

1.3 Tujuan

Tujuan dari pembuatan aplikasi FamCare adalah sebagai berikut:

- 1) Merancang antarmuka pengguna (UI) yang responsif dan user-friendly menggunakan bahasa pemrograman Kotlin.
- 2) Mengimplementasikan algoritma pencarian dan filter pengasuh yang efisien menggunakan Firebase Firestore untuk menyimpan dan mengambil data.
- 3) Mengidentifikasi dan mengembangkan fitur inti yang memastikan fungsionalitas aplikasi FamCare yang lengkap dan optimal.

1.4 Manfaat

Manfaat yang dapat diperoleh oleh pengguna aplikasi FamCare adalah sebagai berikut:

- 1) Meningkatkan pengalaman pengguna melalui antarmuka yang mudah digunakan dan responsif.
- 2) Meningkatkan efisiensi pencarian dan pemfilteran pengasuh dengan memanfaatkan Firebase Firestore.
- 3) Menyediakan aplikasi FamCare dengan fitur yang lengkap dan optimal untuk kebutuhan pengasuhan bayi, anak-anak, dan lansia.
- 4) Meningkatkan kinerja aplikasi dengan optimalisasi algoritma pengolahan data yang lebih akurat.

1.5 Ruang Lingkup

Berikut merupakan batasan-batasan ruang lingkup dalam pembuatan aplikasi FamCare:

- 1) Aplikasi ini berfokus dari perancangan sistem dari sisi pengguna yang ingin mencari jasa pengasuh, tidak termasuk pembuatan aplikasi dari sisi admin maupun dari sisi pengasuh.
- 2) Aplikasi ini tidak memfasilitasi proses transaksi pembayaran antara pengguna dengan pengasuh dalam proses reservasi yang dilakukan.
- 3) Data pengasuh yang tersedia dalam aplikasi terbatas pada wilayah Indonesia. Informasi mengenai fasilitas Daycare dan Elderly Care yang ada juga terbatas pada

wilayah tersebut, memastikan bahwa cakupan data tidak meluas ke area di luar Indonesia.

- 4) Seluruh antarmuka aplikasi ini disajikan dalam bahasa Inggris.
- 5) Proyek ini mencakup pembuatan dokumentasi teknis, namun tidak mencakup pemeliharaan dan pembaruan rutin aplikasi yang berkelanjutan setelahnya.

1.6 Peran

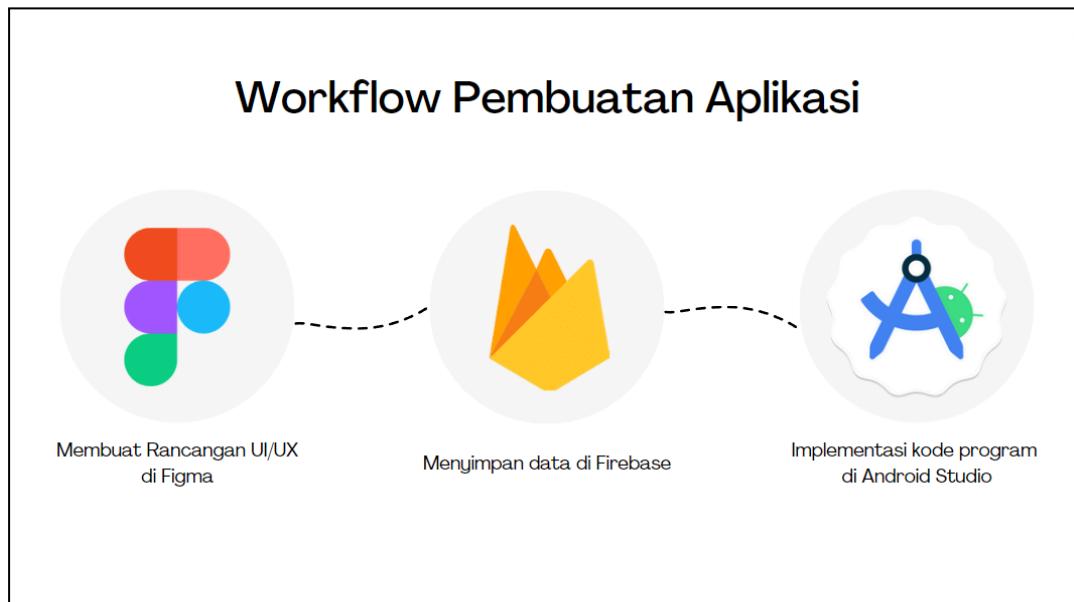
Nama Anggota	Peran
Aji Wijayatama	Pembuatan fitur Register, Login, dan Lupa Password
Stefanny	Pembuatan fitur Pencarian (Search), Filter, dan Logout
Ni Wayan Giri Puspawati	Pembuatan fitur Profile, Edit Profile, Reset Password
Novi Nur Rahmawati	Pembuatan fitur Book Nanny, Cancel Book, dan Bookmark
Siti Aisyah Nur Alim	Pembuatan fitur Chat Nanny, History, dan Pencarian Fasilitas berbasis Geolocation

Tabel 1.6.1 Peran Anggota.

BAB II

PEMBAHASAN

2.1 Workflow Aplikasi



Gambar 2.1. Workflow Pembuatan Aplikasi

Alur penggeraan aplikasi FamCare dimulai dari beberapa tahapan utama yang mencakup desain antarmuka, konfigurasi backend, dan pengembangan aplikasi. Berikut adalah tahapan-tahapan lengkapnya beserta tools yang digunakan:

a. Figma

Figma digunakan untuk desain dan prototyping antarmuka aplikasi FamCare. Dalam Figma, desainer dapat menentukan warna-warna yang sesuai dan menyimpannya di color palette. Figma memungkinkan desainer untuk membuat wireframe, mockup, dan prototipe interaktif yang membantu dalam visualisasi dan pengujian desain sebelum diimplementasikan dalam kode. Warna, font, dan elemen UI lainnya didefinisikan secara konsisten di seluruh aplikasi, memastikan tampilan yang profesional dan mudah digunakan.

b. Firebase

Firebase adalah platform milik Google yang digunakan oleh para developer aplikasi karena menyediakan berbagai layanan backend. Alasan pemilihan Firebase untuk proyek ini adalah karena integrasinya yang mudah dengan aplikasi Android, skalabilitasnya, dan berbagai fitur yang mendukung pengembangan aplikasi secara cepat dan efisien. Berikut adalah layanan Firebase yang digunakan dalam aplikasi FamCare:

- 1) Authentication: Firebase Authentication digunakan untuk autentifikasi pengguna menggunakan metode email dan password. Metode ini digunakan untuk memvalidasi bahwa hanya pengguna terdaftar yang dapat mengakses aplikasi.

- 2) Realtime Database: Realtime Database digunakan untuk menyimpan data chat antara pengguna dengan pengasuh secara real-time. Penggunaan Realtime Database ini memungkinkan pesan dikirim dan diterima tanpa penundaan, sehingga nantinya pengalaman pengguna saat menggunakan fitur chat akan menjadi lebih baik.
- 3) Firestore Database: Pada aplikasi ini, Firestore Database digunakan untuk menyimpan berbagai data yang digunakan dalam aplikasi. Firestore terdiri dari beberapa koleksi, yaitu BookingDaily, BookingMonthly, Bookmarks, Daycare, ElderlyCare, Nanny, dan User. Setiap koleksi menyimpan informasi yang spesifik dan terstruktur yang terkait dengan nama koleksi masing-masing.
- 4) Firebase Storage : Firebase Storage digunakan untuk menyimpan file gambar, termasuk foto nanny dan foto profil pengguna dalam format jpg maupun png.

c. Android Studio

Project FamCare ini menggunakan Android Studio sebagai Integrated Development Environment (IDE) untuk mengembangkan aplikasi. Android Studio adalah IDE resmi untuk pengembangan aplikasi Android yang menyediakan alat-alat canggih untuk pengkodean, debugging, dan pengujian. Bahasa pemrograman yang digunakan adalah Kotlin. Alasan penggunaan Kotlin adalah karena Kotlin menawarkan sintaks yang lebih bersih, aman, dan mendukung fitur modern dibandingkan dengan Java. Kotlin juga memiliki interoperabilitas penuh dengan Java, memungkinkan integrasi kode lama dengan mudah. Dalam pengembangan aplikasi FamCare, digunakan beberapa library seperti:

- 1) AppCompat: Menyediakan backward compatibility untuk fitur-fitur terbaru di Android.
- 2) Material: Menyediakan komponen Material Design untuk membangun UI yang modern dan konsisten.
- 3) ConstraintLayout: Layout manager yang memungkinkan penataan tampilan UI yang fleksibel dan responsif.
- 4) Play Services Maps: Digunakan untuk integrasi Google Maps dalam aplikasi.
- 5) Firebase Auth: Menyediakan layanan autentikasi pengguna menggunakan Firebase Authentication.
- 6) Firebase Firestore KTX: Menyediakan akses ke Firebase Firestore Database dengan ekstensi Kotlin.
- 7) Firebase Storage KTX: Menyediakan akses ke Firebase Storage untuk menyimpan dan mengambil file.
- 8) JUnit: Framework untuk unit testing pada aplikasi Java.
- 9) AndroidX Test Ext JUnit: Menyediakan ekstensi JUnit untuk pengujian Android.
- 10) Espresso: Digunakan untuk pengujian UI otomatis pada aplikasi Android.

- 11) CardView: Menyediakan widget CardView yang memudahkan implementasi kartu dalam UI.
- 12) Glide: Library untuk memuat URL gambar ke dalam aplikasi.
- 13) Play Services Location: Digunakan untuk mengakses layanan lokasi Google.
- 14) Firebase Analytics: Menyediakan layanan analitik untuk aplikasi dengan Firebase Analytics.
- 15) Firebase Realtime Database: Menyediakan akses ke Firebase Realtime Database.
- 16) JustifiedTextview: Menyediakan TextView dengan teks yang terjustifikasi.
- 17) Kotlin Coroutines Android: Menyediakan dukungan coroutine untuk operasi asynchronous di Android.
- 18) Preference KTX: Menyediakan UI dan API untuk mengelola preferensi aplikasi dengan ekstensi Kotlin.

Setiap dependency ini berkontribusi pada berbagai aspek fungsionalitas dan pengembangan aplikasi FamCare, baik itu dari aspek desain UI hingga backend dan pengujian aplikasi.

2.2 Implementasi Kode

2.2.1 Splash Screen

[File activity SplashActivity.kt]

```
package com.app.famcare.view.splash

import android.content.Intent
import android.os.Bundle
import android.os.Handler
import androidx.appcompat.app.AppCompatActivity
import com.app.famcare.R
import com.app.famcare.view.login.LoginActivity
import com.app.famcare.view.main.MainActivity
import com.app.famcare.view.onboarding.OnboardingActivity
import com.google.firebase.auth.FirebaseAuth

class SplashActivity : AppCompatActivity() {

    private val splashTimeOut: Long = 5000
    private lateinit var firebaseAuth: FirebaseAuth

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

        firebaseAuth = FirebaseAuth.getInstance()

        Handler().postDelayed({
            val currentUser = firebaseAuth.currentUser

            if (currentUser != null) {
                val intent = Intent(this, MainActivity::class.java)
                startActivity(intent)
            } else {

```

```

        val intent = if (isFirstTimeUser()) {
            Intent(this, OnboardingActivity::class.java)
        } else {
            Intent(this, LoginActivity::class.java)
        }
        startActivity(intent)
    }
    finish()
}, splashTimeOut)
}

private fun isFirstTimeUser(): Boolean {
    val currentUser = firebaseAuth.currentUser
    return currentUser?.metadata?.creationTimestamp ==
currentUser?.metadata?.lastSignInTimestamp
}
}

```

[File XML activity_splash.xml]

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/white">

    <ImageView
        android:id="@+id/imgLogo"
        android:layout_width="200dp"
        android:layout_height="200dp"
        android:layout_centerInParent="true"
        android:src="@drawable/logo_famcare" />
</RelativeLayout>

```

Penjelasan:

SplashActivity menampilkan layar pembuka yang diatur dalam activity_splash.xml selama 5 detik. Dalam fungsi onCreate, SplashActivity menggunakan FirebaseAuth untuk memeriksa status pengguna. Jika pengguna sudah masuk, aplikasi melanjutkan ke MainActivity. Jika belum, aplikasi menggunakan fungsi isFirstTimeUser untuk memeriksa apakah ini pertama kali pengguna menggunakan aplikasi, dan mengarahkan mereka ke OnboardingActivity atau LoginActivity, memastikan alur yang tepat bagi pengguna.

2.2.2 Onboarding

[File activity OnboardingActivity.kt]

```

package com.app.famcare.view.onboarding

import android.content.Intent
import android.os.Bundle
import android.widget.Button
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity
import com.app.famcare.R
import com.app.famcare.view.login.LoginActivity

```

```

class OnboardingActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_onboarding)
        val loginButton = findViewById<Button>(R.id.loginButton)
        val registerButton = findViewById<TextView>(R.id.registerButton)
        loginButton.setOnClickListener {
            val intent = Intent(this@OnboardingActivity,
SecondOnboardingActivity::class.java)
            startActivity(intent)
        }
        registerButton.setOnClickListener {
            val intent = Intent(this@OnboardingActivity,
LoginActivity::class.java)
            startActivity(intent)
        }
    }
}

```

[File activity SecondOnboardingActivity.kt]

```

package com.app.famcare.view.onboarding

import android.content.Intent
import android.os.Bundle
import android.widget.Button
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity
import com.app.famcare.R
import com.app.famcare.view.login.LoginActivity

class SecondOnboardingActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_second_onboarding)
        val loginButton = findViewById<Button>(R.id.loginButton)
        val registerButton = findViewById<TextView>(R.id.registerButton)
        loginButton.setOnClickListener {
            val intent = Intent(this@SecondOnboardingActivity,
ThirdOnboardingActivity::class.java)
            startActivity(intent)
        }
        registerButton.setOnClickListener {
            val intent = Intent(this@SecondOnboardingActivity,
LoginActivity::class.java)
            startActivity(intent)
        }
    }
}

```

[File activity ThirdOnboardingActivity.kt]

```

package com.app.famcare.view.onboarding

import android.content.Intent
import android.os.Bundle
import android.widget.Button
import androidx.appcompat.app.AppCompatActivity
import com.app.famcare.R
import com.app.famcare.view.login.LoginActivity

class ThirdOnboardingActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_third_onboarding)
        val loginButton = findViewById<Button>(R.id.loginButton)
        loginButton.setOnClickListener {

```

```

        val intent = Intent(this@ThirdOnboardingActivity,
LoginActivity::class.java)
        startActivity(intent)
    }
}
}

```

[File XML activity_onboarding.xml]

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".view.onboarding.OnboardingActivity">

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="match_parent"
        android:layout_height="500dp"
        android:layout_gravity="center_horizontal"
        android:layout_marginTop="-50dp"
        app:srcCompat="@drawable/onboarding1" />

    <TextView
        android:id="@+id/titleWelcome"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:layout_marginTop="28dp"
        android:fontFamily="@font/montserratsemibold"
        android:gravity="center"
        android:text="Find the Best Caregivers"
        android:textColor="@color/black"
        android:textSize="24sp"
        tools:alpha="100" />

    <TextView
        android:id="@+id/titleTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:layout_marginLeft="16dp"
        android:layout_marginTop="16dp"
        android:layout_marginRight="16dp"
        android:fontFamily="@font/montserrat"
        android:gravity="center"
        android:text="Easily select caregivers that match your preferences"
        android:textColor="@color/black"
        android:textSize="18sp"
        tools:alpha="100" />

    <Button
        android:id="@+id/loginButton"
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:layout_gravity="center_horizontal"
        android:layout_marginStart="32dp"
        android:layout_marginLeft="32dp"
        android:layout_marginTop="48dp"
        android:layout_marginEnd="32dp"
        android:layout_marginRight="32dp"
        android:backgroundTint="@color/icon_active"
        android:fontFamily="@font/montserratsemibold"
        android:text="Next"

```

```

        android:textSize="20sp"
        tools:alpha="100" />

    <TextView
        android:id="@+id/registerButton"
        android:layout_width="wrap_content"
        android:layout_height="60dp"
        android:layout_gravity="center"
        android:layout_marginLeft="16dp"
        android:layout_marginTop="16dp"
        android:layout_marginRight="16dp"
        android:backgroundTint="@color/icon_active"
        android:fontFamily="@font/montserratsemibold"
        android:text="Skip Tour"
        android:textColor="@color/dark_grey"
        android:textSize="16sp"
        tools:alpha="100" />

</LinearLayout>

```

[File XML activity_second_onboarding.xml]

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".view.onboarding.SecondOnboardingActivity">

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="match_parent"
        android:layout_height="500dp"
        android:layout_gravity="center_horizontal"
        android:layout_marginTop="-50dp"
        app:srcCompat="@drawable/onboarding2" />

    <TextView
        android:id="@+id/titleWelcome"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:layout_marginTop="28dp"
        android:fontFamily="@font/montserratsemibold"
        android:gravity="center"
        android:text="Book and Connect"
        android:textColor="@color/black"
        android:textSize="24sp"
        tools:alpha="100" />

    <TextView
        android:id="@+id/titleTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:layout_marginLeft="16dp"
        android:layout_marginTop="16dp"
        android:layout_marginRight="16dp"
        android:fontFamily="@font/montserrat"
        android:gravity="center"
        android:text="Schedule care sessions and connect directly with your chosen
caregivers"
        android:textColor="@color/black"
        android:textSize="18sp"
        tools:alpha="100" />

```

```

<Button
    android:id="@+id/loginButton"
    android:layout_width="match_parent"
    android:layout_height="60dp"
    android:layout_gravity="center_horizontal"
    android:layout_marginStart="32dp"
    android:layout_marginLeft="32dp"
    android:layout_marginTop="48dp"
    android:layout_marginEnd="32dp"
    android:layout_marginRight="32dp"
    android:backgroundTint="@color/icon_active"
    android:fontFamily="@font/montserratsemibold"
    android:text="Next"
    android:textSize="20sp"
    tools:alpha="100" />

<TextView
    android:id="@+id/registerButton"
    android:layout_width="wrap_content"
    android:layout_height="60dp"
    android:layout_gravity="center"
    android:layout_marginLeft="16dp"
    android:layout_marginTop="16dp"
    android:layout_marginRight="16dp"
    android:backgroundTint="@color/icon_active"
    android:fontFamily="@font/montserratsemibold"
    android:text="Skip Tour"
    android:textColor="@color/dark_grey"
    android:textSize="16sp"
    tools:alpha="100" />

</LinearLayout>

```

[File XML activity_third_onboarding.xml]

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".view.onboarding.ThirdOnboardingActivity">

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="match_parent"
        android:layout_height="500dp"
        android:layout_gravity="center_horizontal"
        android:layout_marginTop="-50dp"
        app:srcCompat="@drawable/family" />

    <TextView
        android:id="@+id/titleWelcome"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:layout_marginTop="28dp"
        android:fontFamily="@font/montserratsemibold"
        android:gravity="center"
        android:text="Welcome to FamCare"
        android:textColor="@color/black"
        android:textSize="24sp"
        tools:alpha="100" />

    <TextView

```

```

        android:id="@+id/titleTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:layout_marginLeft="16dp"
        android:layout_marginTop="16dp"
        android:layout_marginRight="16dp"
        android:fontFamily="@font/montserrat"
        android:gravity="center"
        android:text="Protect your Loved Ones with Us"
        android:textColor="@color/black"
        android:textSize="18sp"
        tools:alpha="100" />

    <Button
        android:id="@+id/loginButton"
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:layout_gravity="center_horizontal"
        android:layout_marginStart="32dp"
        android:layout_marginLeft="32dp"
        android:layout_marginTop="48dp"
        android:layout_marginEnd="32dp"
        android:layout_marginRight="32dp"
        android:backgroundTint="@color/icon_active"
        android:fontFamily="@font/montserratsemibold"
        android:text="Getting Started"
        android:textSize="20sp"
        tools:alpha="100" />

</LinearLayout>

```

Penjelasan:

Ketiga aktivitas OnboardingActivity.kt, SecondOnboardingActivity.kt, dan ThirdOnboardingActivity.kt menangani tiga tahap proses onboarding dalam aplikasi FamCare. Pada setiap tahap, pengguna diberi informasi mengenai fitur dan manfaat aplikasi. Setiap tahap memiliki tombol "Next" untuk melanjutkan ke tahap berikutnya, dan tombol "Skip Tour" atau "Getting Started" untuk melewati onboarding dan langsung ke LoginActivity. Pendefinisian layout ketiga tampilan pada tiap tahap onboarding diatur dalam activity_onboarding.xml, activity_second_onboarding.xml, dan activity_third_onboarding.xml. Masing-masing layout memiliki elemen ImageView untuk menampilkan gambar ilustrasi, TextView untuk judul dan deskripsi, serta Button untuk navigasi.

Proses onboarding dimulai dari OnboardingActivity yang menampilkan informasi awal dengan opsi untuk melanjutkan ke SecondOnboardingActivity atau langsung ke LoginActivity. Selanjutnya, SecondOnboardingActivity memberikan informasi tambahan dan menawarkan navigasi ke ThirdOnboardingActivity atau LoginActivity. Akhirnya, ThirdOnboardingActivity menyelesaikan proses onboarding dengan opsi untuk langsung ke LoginActivity.

2.2.3 Register

[File CustomView emailEditText.kt]

```

package com.app.famcare.view.customview

import android.content.Context

```

```

import android.text.Editable
import android.text.TextWatcher
import android.util.AttributeSet
import androidx.appcompat.widget.AppCompatEditText

class EmailEditText : AppCompatEditText {

    constructor(context: Context) : super(context) {
        init()
    }

    constructor(context: Context, attrs: AttributeSet) : super(context, attrs) {
        init()
    }

    constructor(context: Context, attrs: AttributeSet, defStyleAttr: Int) : super(
        context, attrs, defStyleAttr
    ) {
        init()
    }

    private fun init() {
        addTextChangedListener(object : TextWatcher {
            override fun beforeTextChanged(s: CharSequence?, start: Int, count: Int, after: Int) {

            }

            override fun onTextChanged(s: CharSequence?, start: Int, before: Int, count: Int) {
                if (s != null && !isValidEmail(s.toString())) {
                    error = "Format email tidak valid"
                } else {
                    error = null
                }
            }

            override fun afterTextChanged(s: Editable?) {
            }
        })
    }

    private fun isValidEmail(email: String): Boolean {
        val emailPattern = "[a-zA-Z0-9._-]+@[a-z]+\\".+[a-z]+"
        return email.matches(emailPattern.toRegex())
    }
}

```

[File CustomView passwordEditText.kt]

```

package com.app.famcare.view.customview

import android.content.Context
import android.text.Editable
import android.text.TextWatcher
import android.util.AttributeSet
import androidx.appcompat.widget.AppCompatEditText

class PasswordEditText : AppCompatEditText {

    constructor(context: Context) : super(context) {
        init()
    }

    constructor(context: Context, attrs: AttributeSet) : super(context, attrs) {
        init()
    }

    constructor(context: Context, attrs: AttributeSet, defStyleAttr: Int) : super(
        context, attrs, defStyleAttr
    ) {
        init()
    }

    private fun init() {
        setTransformationMethod(PasswordTransformationMethod())
        addTextChangedListener(object : TextWatcher {
            override fun beforeTextChanged(s: CharSequence?, start: Int, count: Int, after: Int) {

            }

            override fun onTextChanged(s: CharSequence?, start: Int, before: Int, count: Int) {
                if (s != null && !isValidEmail(s.toString())) {
                    error = "Format email tidak valid"
                } else {
                    error = null
                }
            }

            override fun afterTextChanged(s: Editable?) {
            }
        })
    }

    private fun isValidEmail(email: String): Boolean {
        val emailPattern = "[a-zA-Z0-9._-]+@[a-z]+\\".+[a-z]+"
        return email.matches(emailPattern.toRegex())
    }
}

```

```

constructor(context: Context, attrs: AttributeSet, defStyleAttr: Int) : super(
    context, attrs, defStyleAttr
) {
    init()
}

private fun init() {
    addTextChangedListener(object : TextWatcher {
        override fun beforeTextChanged(s: CharSequence, start: Int, count: Int, after: Int) {

        }

        override fun onTextChanged(s: CharSequence, start: Int, before: Int, count: Int) {
            if (s.toString().length < 8) {
                setError("Password must be at least 8 characters long", null)
            } else {
                error = null
            }
        }

        override fun afterTextChanged(s: Editable) {
        }
    })
}
}

```

[File activity Register.kt]

```

package com.app.famcare.view.register

import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.util.Log
import android.widget.Toast
import com.app.famcare.databinding.ActivityRegisterBinding
import com.app.famcare.view.login.LoginActivity
import com.google.firebase.auth.FirebaseAuth
import com.google.firebaseio.firebaseio.FirebaseFirestore

class RegisterActivity : AppCompatActivity() {
    private lateinit var binding: ActivityRegisterBinding
    private lateinit var firebaseAuth: FirebaseAuth
    private lateinit var firestore: FirebaseFirestore

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityRegisterBinding.inflate(layoutInflater)
        setContentView(binding.root)

        firebaseAuth = FirebaseAuth.getInstance()
        firestore = FirebaseFirestore.getInstance()

        binding.registerButton.setOnClickListener {
            val email = binding.emailEditText.text.toString()
            val password = binding.passwordEditText.text.toString()
            val confirmPassword = binding.confirmPasswordEditText.text.toString()
            val fullName = binding.fullnameEditText.text.toString()
            val phone = binding.phoneEditText.text.toString()

            if (email.isNotEmpty() && password.isNotEmpty() &&
            confirmPassword.isNotEmpty() && fullName.isNotEmpty() && phone.isNotEmpty()) {
                if (password == confirmPassword) {
                    firebaseAuth.createUserWithEmailAndPassword(email, password)
                        .addOnCompleteListener { createUserTask ->
                            if (createUserTask.isSuccessful) {

```

```

        val user = firebaseAuth.currentUser
        val uid = user?.uid

        uid?.let { uid ->
            saveUserDataToFirestore(uid, fullName, email,
            phone)
        }
        sendEmailVerification()
        firebaseAuth.signOut()
        val intent = Intent(this,
LoginActivity::class.java)
        intent.putExtra("isNewUser", true)
        startActivity(intent)
        finish()
    } else {
        Toast.makeText(
            this, createUserTask.exception.toString(),
            Toast.LENGTH_SHORT
        ).show()
    }
}
} else {
    Toast.makeText(
        this, "Passwords do not match. Please try again.",
        Toast.LENGTH_SHORT
    ).show()
}
} else {
    Toast.makeText(this, "All fields must be filled",
    Toast.LENGTH_SHORT).show()
}
}

binding.bottom2TextViews.setOnClickListener {
    val intent = Intent(this, LoginActivity::class.java)
    startActivity(intent)
}
}

private fun saveUserDataToFirestore(
    uid: String, fullName: String, email: String, phone: String
) {
    val userData = hashMapOf(
        "uid" to uid,
        "fullName" to fullName,
        "email" to email,
        "phone" to phone,
        "address" to "",
        "gender" to "",
        "birth" to ""
    )
}

firestore.collection("User").document(uid).set(userData).addOnSuccessListener {}
    .addOnFailureListener { exception ->
        Log.e("RegisterActivity", "Error writing document", exception)
    }
}

private fun sendEmailVerification() {
    val user = firebaseAuth.currentUser
    user?.sendEmailVerification()?.addOnCompleteListener { task ->
        if (task.isSuccessful) {
            Toast.makeText(
                this, "Verification email sent to ${user.email}",
                Toast.LENGTH_SHORT
            ).show()
        }
    }
}

```

```

        } else {
            Toast.makeText(this, "Failed to send verification email.",
Toast.LENGTH_SHORT)
            .show()
        }
    }
}

```

[File XML activity_register.xml]

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:paddingStart="32dp"
    android:paddingTop="2dp"
    android:paddingEnd="32dp"
    android:paddingBottom="8dp"
    tools:context=".view.register.RegisterActivity">

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="289dp"
        android:layout_height="120dp"
        android:layout_gravity="center"
        android:layout_marginStart="32dp"
        android:layout_marginEnd="32dp"
        app:srcCompat="@drawable/logo_famcare_2" />

    <TextView
        android:id="@+id/titleTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_marginTop="4dp"
        android:fontFamily="@font/montserratbold"
        android:text="Sign Up"
        android:textColor="@color/black"
        android:textSize="24sp" />

    <TextView
        android:id="@+id/fullnameTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="18dp"
        android:fontFamily="@font/montserrat"
        android:text="Full Name"
        android:textColor="@color/black"
        android:textSize="16sp" />

    <com.google.android.material.textfield.TextInputLayout
        android:id="@+id/fullnameEditTextLayout"
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="4dp"
        app:boxCornerRadiusBottomEnd="20dp"
        app:boxCornerRadiusBottomStart="20dp"
        app:boxCornerRadiusTopEnd="20dp"
        app:boxCornerRadiusTopStart="20dp"
        app:startIconDrawable="@drawable/baseline_person_24">

        <EditText

```

```

        android:id="@+id/fullnameEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="text" />
    </com.google.android.material.textfield.TextInputLayout>

    <TextView
        android:id="@+id/emailTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="12dp"
        android:fontFamily="@font/montserrat"
        android:text="Email"
        android:textColor="@color/black"
        android:textSize="16sp" />

    <com.google.android.material.textfield.TextInputLayout
        android:id="@+id/emailEditTextLayout"
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="4dp"
        app:boxCornerRadiusBottomEnd="20dp"
        app:boxCornerRadiusBottomStart="20dp"
        app:boxCornerRadiusTopEnd="20dp"
        app:boxCornerRadiusTopStart="20dp"
        app:startIconDrawable="@drawable/baseline_email_24">

        <com.app.famcare.view.customview.EmailEditText
            android:id="@+id/emailEditText"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:ems="10"
            android:inputType="textEmailAddress" />
    </com.google.android.material.textfield.TextInputLayout>

    <TextView
        android:id="@+id/phoneTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="12dp"
        android:fontFamily="@font/montserrat"
        android:text="Phone Number"
        android:textColor="@color/black"
        android:textSize="16sp" />

    <com.google.android.material.textfield.TextInputLayout
        android:id="@+id/phoneEditTextLayout"
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="4dp"
        app:boxCornerRadiusBottomEnd="20dp"
        app:boxCornerRadiusBottomStart="20dp"
        app:boxCornerRadiusTopEnd="20dp"
        app:boxCornerRadiusTopStart="20dp"
        app:startIconDrawable="@drawable/baseline_phone_24">

        <EditText
            android:id="@+id/phoneEditText"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:ems="10"
            android:inputType="phone" />
    </com.google.android.material.textfield.TextInputLayout>

```

```

<TextView
    android:id="@+id/passwordTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="12dp"
    android:fontFamily="@font/montserrat"
    android:text="Password"
    android:textColor="@color/black"
    android:textSize="16sp" />

<com.google.android.material.textfield.TextInputLayout
    android:id="@+id/passwordEditTextLayout"
    style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="4dp"
    app:boxCornerRadiusBottomEnd="20dp"
    app:boxCornerRadiusBottomStart="20dp"
    app:boxCornerRadiusTopEnd="20dp"
    app:boxCornerRadiusTopStart="20dp"
    app:endIconMode="password_toggle"
    app:startIconDrawable="@drawable/baseline_lock_24">

    <com.app.famcare.view.customview.PasswordEditText
        android:id="@+id/passwordEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPassword" />
</com.google.android.material.textfield.TextInputLayout>

<TextView
    android:id="@+id/confirmPasswordTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="12dp"
    android:fontFamily="@font/montserrat"
    android:text="Confirm Password"
    android:textColor="@color/black"
    android:textSize="16sp" />

<com.google.android.material.textfield.TextInputLayout
    android:id="@+id/confirmPasswordEditTextLayout"
    style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="4dp"
    app:boxCornerRadiusBottomEnd="20dp"
    app:boxCornerRadiusBottomStart="20dp"
    app:boxCornerRadiusTopEnd="20dp"
    app:boxCornerRadiusTopStart="20dp"
    app:endIconMode="password_toggle"
    app:startIconDrawable="@drawable/baseline_lock_24">

    <com.app.famcare.view.customview.PasswordEditText
        android:id="@+id/confirmPasswordEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPassword" />
</com.google.android.material.textfield.TextInputLayout>

<Button
    android:id="@+id/registerButton"
    android:layout_width="match_parent"
    android:layout_height="60dp"
    android:layout_gravity="center" />

```

```

        android:layout_marginStart="32dp"
        android:layout_marginTop="24dp"
        android:layout_marginEnd="32dp"
        android:backgroundTint="@color/icon_active"
        android:fontFamily="@font/montserratsemibold"
        android:text="Register"
        android:textSize="20sp" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:gravity="center"
        android:orientation="horizontal">

        <TextView
            android:id="@+id/bottom1TextView"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:fontFamily="@font/montserrat"
            android:text="Already have an account?"
            android:textColor="@color/black"
            android:textSize="16sp" />

        <TextView
            android:id="@+id/bottom2TextViews"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="4dp"
            android:fontFamily="@font/montserratsemibold"
            android:text="Log in"
            android:textColor="@color/button"
            android:textSize="16sp" />
    </LinearLayout>
</LinearLayout>

```

Penjelasan:

RegisterActivity.kt bertanggung jawab untuk menangani proses pendaftaran pengguna baru dalam aplikasi. Aktivitas ini mengatur tampilan registrasi, mengambil input dari pengguna (seperti email, kata sandi, nama lengkap, dan nomor telepon), dan memvalidasi input tersebut. Jika validasi berhasil dan kata sandi cocok, aktivitas ini akan membuat akun pengguna baru menggunakan Firebase Authentication dan menyimpan data pengguna lainnya ke dalam koleksi Firestore bernama “User”. Selain itu, aktivitas ini mengirimkan email verifikasi ke pengguna, mengarahkan pengguna ke LoginActivity, dan menampilkan pesan kesalahan jika ada masalah selama proses pendaftaran. Dalam activity_register.xml, kolom-kolom seperti email, kata sandi, dan konfirmasi kata sandi ditampilkan menggunakan custom view bernama editText dan passwordEditText untuk menampilkan pesan kesalahan apabila penulisan email atau kata sandi tidak benar.

2.2.4 Login

[File LoginActivity.kt]

```

package com.app.famcare.view.login

import android.content.Intent
import android.os.Bundle

```

```

import android.widget.Button
import android.widget.TextView
import android.widget.Toast
import androidx.appcompat.app.AlertDialog
import androidx.appcompat.app.AppCompatActivity
import com.app.famcare.R
import com.app.famcare.databinding.ActivityLoginBinding
import com.app.famcare.view.customview.EmailEditText
import com.app.famcare.view.main.MainActivity
import com.app.famcare.view.register.RegisterActivity
import com.google.firebase.auth.FirebaseAuth
import com.google.firebase.auth.FirebaseAuthInvalidCredentialsException
import com.google.firebase.auth.FirebaseAuthInvalidUserException
import com.google.firebaseio.FirebaseFirestore

class LoginActivity : AppCompatActivity() {
    private lateinit var binding: ActivityLoginBinding
    private lateinit var firebaseAuth: FirebaseAuth
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityLoginBinding.inflate(layoutInflater)
        setContentView(binding.root)
        firebaseAuth = FirebaseAuth.getInstance()

        binding.loginButton.setOnClickListener {
            val email = binding.emailEditText.text.toString()
            val password = binding.passwordEditText.text.toString()

            if (email.isNotEmpty() && password.isNotEmpty()) {
                firebaseAuth.signInWithEmailAndPassword(email, password)
                    .addOnCompleteListener { task ->
                        if (task.isSuccessful) {
                            val currentUser = firebaseAuth.currentUser
                            if (currentUser != null && currentUser.isEmailVerified) {
                                val intent = Intent(this,
MainActivity::class.java)
                                startActivity(intent)
                                finish()
                            } else {
                                Toast.makeText(
                                    this, "Please verify your email first",
Toast.LENGTH_SHORT
                                ).show()
                                firebaseAuth.signOut()
                            }
                        } else {
                            handleLoginError(task.exception)
                        }
                    }
            } else {
                Toast.makeText(this, "Email and Password cannot be empty",
Toast.LENGTH_SHORT)
                    .show()
            }
        }

        binding.bottom2TextViews.setOnClickListener {
            val intent = Intent(this, RegisterActivity::class.java)
            startActivity(intent)
        }

        binding.tvForgotPassword.setOnClickListener {
            showForgotPasswordPopup()
        }
    }
}

```

```

private fun handleLoginError(exception: Exception?) {
    when (exception) {
        is FirebaseAuthInvalidUserException -> {
            Toast.makeText(this, "No account found with this email",
            Toast.LENGTH_SHORT).show()
        }

        is FirebaseAuthInvalidCredentialsException -> {
            Toast.makeText(this, "Invalid password",
            Toast.LENGTH_SHORT).show()
        }

        else -> {
            Toast.makeText(this, "Login failed: ${exception?.message}",
            Toast.LENGTH_SHORT)
                .show()
        }
    }
}

private fun showForgotPasswordPopup() {
    val dialogView = LayoutInflater.inflate(R.layout.popup_reset_password,
null)
    val builder = AlertDialog.Builder(this).setView(dialogView)

    val dialog = builder.create()
    dialog.show()

    val emailEditText =
dialogView.findViewById<EditText>(R.id.emailEditText)
    val resetPasswordButton =
dialogView.findViewById<Button>(R.id.resetPasswordButton)
    val bottom2TextViews =
dialogView.findViewById<TextView>(R.id.bottom2TextViews)

    resetPasswordButton.setOnClickListener {
        val email = emailEditText.text.toString().trim()

        val firestore = FirebaseFirestore.getInstance()
        val usersCollection = firestore.collection("User")
        usersCollection.whereEqualTo("email", email).get()
            .addOnSuccessListener { querySnapshot ->
                if (!querySnapshot.isEmpty) {

firebaseAuth.sendPasswordResetEmail(email).addOnCompleteListener { task ->
                    if (task.isSuccessful) {
                        val alertDialog =
AlertDialog.Builder(this).setTitle("Email Sent")
                            .setMessage("Password reset email sent to
$email")
                            .setPositiveButton("OK") { _, _ ->
                                dialog.dismiss()
                            }.setCancelable(false).create()
                        alertDialog.show()
                    } else {
                        Toast.makeText(
                            this,
                            "Failed to send reset email:
${task.exception?.message}",
                            Toast.LENGTH_SHORT
                        ).show()
                    }
                }
            } else {
                val alertDialog =
AlertDialog.Builder(this).setTitle("Email Not Registered")

```

```

        .setMessage("The entered email is not registered.
Please check your email or register a new account.")
        .setPositiveButton("OK") { _, _ ->
            emailEditText.text?.clear()
            dialog.dismiss()
        }.setCancelable(false).create()
        alertDialog.show()
    }
}.addOnFailureListener { e ->
    Toast.makeText(
        this, "Failed to check email validity: ${e.message}",
        Toast.LENGTH_SHORT
    ).show()
}

bottom2TextViews.setOnClickListener {
    dialog.dismiss()
}
}

}

```

[File XML activity_login.xml]

```

<?xml version="1.0" encoding="utf-8"?>
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <androidx.fragment.app.FragmentContainerView
        android:id="@+id/fragmentContainer"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <ImageView
            android:id="@+id/imageView"
            android:layout_width="match_parent"
            android:layout_height="150dp"
            android:layout_gravity="center_horizontal"
            android:layout_marginStart="32dp"
            android:layout_marginTop="64dp"
            android:layout_marginEnd="32dp"
            app:srcCompat="@drawable/logo_famcare_2" />

        <TextView
            android:id="@+id/titleTextView"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:layout_marginStart="32dp"
            android:layout_marginTop="12dp"
            android:layout_marginEnd="32dp"
            android:fontFamily="@font/montserratbold"
            android:gravity="center_horizontal"
            android:text="Sign In"
            android:textColor="@color/black"
            android:textSize="26sp" />

        <TextView

```

```

        android:id="@+id/emailTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="32dp"
        android:layout_marginTop="32dp"
        android:fontFamily="@font/montserrat"
        android:text="Email"
        android:textColor="@color/black"
        android:textSize="16sp" />

    <com.google.android.material.textfield.TextInputLayout
        android:id="@+id/emailEditTextLayout"
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginStart="32dp"
        android:layout_marginEnd="32dp"
        app:boxCornerRadiusBottomEnd="20dp"
        app:boxCornerRadiusBottomStart="20dp"
        app:boxCornerRadiusTopEnd="20dp"
        app:boxCornerRadiusTopStart="20dp"
        app:startIconDrawable="@drawable/baseline_email_24">
        <com.app.famcare.view.customview.EmailEditText
            android:id="@+id/emailEditText"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:ems="10"
            android:inputType="textEmailAddress"
            tools:ignore="SpeakableTextPresentCheck" />
    </com.google.android.material.textfield.TextInputLayout>

    <TextView
        android:id="@+id/passwordTextView"
        android:layout_width="97dp"
        android:layout_height="21dp"
        android:layout_marginStart="32dp"
        android:layout_marginTop="16dp"
        android:fontFamily="@font/montserrat"
        android:text="Password"
        android:textColor="@color/black"
        android:textSize="16sp" />

    <com.google.android.material.textfield.TextInputLayout
        android:id="@+id/passwordEditTextLayout"
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginStart="32dp"
        android:layout_marginEnd="32dp"
        app:boxCornerRadiusBottomEnd="20dp"
        app:boxCornerRadiusBottomStart="20dp"
        app:boxCornerRadiusTopEnd="20dp"
        app:boxCornerRadiusTopStart="20dp"
        app:endIconMode="password_toggle"
        app:startIconDrawable="@drawable/baseline_lock_24">
        <com.app.famcare.view.customview.PasswordEditText
            android:id="@+id/passwordEditText"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:ems="10"
            android:inputType="textPassword"
            tools:ignore="SpeakableTextPresentCheck" />
    </com.google.android.material.textfield.TextInputLayout>

    <TextView
        android:id="@+id/tvForgotPassword"

```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="end"
        android:layout_marginStart="32dp"
        android:layout_marginTop="12dp"
        android:layout_marginEnd="32dp"
        android:fontFamily="@font/montserrat"
        android:text="Forgot Password?"
        android:textColor="@color/button"
        android:textSize="16sp" />

    <Button
        android:id="@+id/loginButton"
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:layout_marginStart="32dp"
        android:layout_marginTop="32dp"
        android:layout_marginEnd="32dp"
        android:layout_marginBottom="8dp"
        android:backgroundTint="@color/icon_active"
        android:fontFamily="@font/montserratsemibold"
        android:text="Log in"
        android:textSize="20sp" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:gravity="center"
        android:orientation="horizontal">

        <TextView
            android:id="@+id/bottom1TextViews"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:fontFamily="@font/montserrat"
            android:text="Don't have an account?"
            android:textColor="@color/black"
            android:textSize="16sp" />

        <TextView
            android:id="@+id/bottom2TextViews"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="4dp"
            android:fontFamily="@font/montserratsemibold"
            android:text="Register"
            android:textColor="@color/button"
            android:textSize="16sp" />
    </LinearLayout>

</LinearLayout>

</FrameLayout>

```

[File XML popup_reset_password.xml]

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="16dp">

```

```

<ImageView
    android:id="@+id/ivWarning"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:layout_gravity="center_horizontal"
    app:srcCompat="@drawable/ic_warning" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:layout_marginTop="8dp"
    android:fontFamily="@font/montserratsemibold"
    android:paddingBottom="16dp"
    android:text="@string/forgot_password"
    android:textSize="20sp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:fontFamily="@font/montserrat"
    android:paddingBottom="16dp"
    android:text="@string/enter_email"
    android:textSize="18sp" />

<com.google.android.material.textfield.TextInputLayout
    android:id="@+id/emailEditTextLayout"
    style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="4dp"
    android:layout_marginBottom="16dp"
    app:boxCornerRadiusBottomEnd="20dp"
    app:boxCornerRadiusBottomStart="20dp"
    app:boxCornerRadiusTopEnd="20dp"
    app:boxCornerRadiusTopStart="20dp"
    app:startIconDrawable="@drawable/baseline_email_24">

    <com.app.famcare.view.customview.EmailEditText
        android:id="@+id/emailEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textEmailAddress"
        tools:ignore="SpeakableTextPresentCheck" />
</com.google.android.material.textfield.TextInputLayout>

<Button
    android:id="@+id/resetPasswordButton"
    android:layout_width="match_parent"
    android:layout_height="60dp"
    android:layout_gravity="center"
    android:layout_marginStart="32dp"
    android:layout_marginEnd="32dp"
    android:fontFamily="@font/montserratsemibold"
    android:text="Submit"
    android:textSize="20sp" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="72dp"
    android:layout_marginTop="10dp"
    android:layout_marginRight="72dp"
    android:gravity="center"
    android:orientation="horizontal">

```

```

<ImageView
    android:id="@+id/imageView"
    android:layout_width="25dp"
    android:layout_height="25dp"
    android:layout_gravity="center_horizontal"
    app:srcCompat="@drawable/ic_arrowback" />

<TextView
    android:id="@+id/bottom2TextViews"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="8dp"
    android:fontFamily="@font/montserrat"
    android:text="@string/back_to_login"
    android:textColor="@color/black"
    android:textSize="16sp" />
</LinearLayout>
</LinearLayout>
```

Penjelasan:

LoginActivity.kt bertanggung jawab untuk menangani proses masuk pengguna ke dalam aplikasi dengan memanfaatkan Firebase Authentication. Ketika pengguna mengunjungi aktivitas ini, pengguna diminta untuk memasukkan email dan kata sandi mereka. Setelah input divalidasi, LoginActivity mengecek ke Firebase Authentication menggunakan firebaseAuth.signInWithEmailAndPassword(email, password). Apabila email sudah terdaftar dan kata sandi cocok, pengguna akan diarahkan ke MainActivity. Namun, jika email belum diverifikasi atau terdapat kesalahan lain dalam proses masuk, pesan kesalahan yang sesuai akan ditampilkan untuk memberitahu pengguna tentang masalah yang terjadi. LoginActivity juga memberikan opsi untuk pengguna berpindah ke RegisterActivity jika mereka belum memiliki akun. Selain itu, halaman LoginActivity juga menyediakan pilihan untuk memulihkan kata sandi apabila pengguna lupa kata sandi milik mereka. Pengguna dapat menekan pilihan Forgot Password, kemudian sebuah popup reset password (popup_reset_password.xml) akan muncul di layar dan pengguna akan diminta untuk mengisikan email yang sekarang. Setelahnya, activity akan menjalankan fungsi firebaseAuth.sendPasswordResetEmail(email) untuk mengirimkan email reset password kepada email milik pengguna.

2.2.5 Discover

[File data class Nanny.kt]

```

package com.app.famcare.model

import android.os.Parcel
import android.os.Parcelable

data class Nanny(
    var id: String = "",
    val age: String = "",
    val experience: String = "",
    val gender: String = "",
    val location: String = "",
    val name: String = "",
```

```

    val rate: String = "",
    val pricing: Int = 0,
    val skills: List<String> = listOf(),
    val type: String = "",
    val pict: String = "",
    val contact: String = "",
    val imageUrl: String = "",
    var isBookmarked: Boolean = false
    : Parcelable {
    constructor(parcel: Parcel) : this(
        id = parcel.readString() ?: "",
        age = parcel.readString() ?: "",
        experience = parcel.readString() ?: "",
        gender = parcel.readString() ?: "",
        location = parcel.readString() ?: "",
        name = parcel.readString() ?: "",
        rate = parcel.readString() ?: "",
        pricing = parcel.readInt(),
        skills = parcel.createStringArrayList() ?: listOf(),
        type = parcel.readString() ?: "",
        pict = parcel.readString() ?: "",
        contact = parcel.readString() ?: "",
        imageUrl = parcel.readString() ?: "",
        isBookmarked = parcel.readByte() != 0.toByte()
    )
    override fun writeToParcel(parcel: Parcel, flags: Int) {
        parcel.writeString(id)
        parcel.writeString(age)
        parcel.writeString(experience)
        parcel.writeString(gender)
        parcel.writeString(location)
        parcel.writeString(name)
        parcel.writeString(rate)
        parcel.writeInt(pricing)
        parcel.writeStringList.skills)
        parcel.writeString(type)
        parcel.writeString(pict)
        parcel.writeString(contact)
        parcel.writeString(imageUrl)
        parcel.writeByte(if (isBookmarked) 1 else 0)
    }
    override fun describeContents(): Int {
        return 0
    }
    companion object CREATOR : Parcelable.Creator<Nanny> {
        override fun createFromParcel(parcel: Parcel): Nanny {
            return Nanny(parcel)
        }
        override fun newArray(size: Int): Array<Nanny?> {
            return arrayOfNulls(size)
        }
    }
}

```

[File repository NannyRepository.kt]

```

package com.app.famcare.repository

import com.app.famcare.model.Nanny
import com.google.firebaseio.firebaseio.FirebaseFirestore
import com.google.firebaseio.firebaseio.Query

class NannyRepository {

```

```

private val db = Firebase Firestore.getInstance()

fun getNannies(
    onSuccess: (List<Nanny>) -> Unit, onFailure: (Exception) -> Unit
) {
    db.collection("Nanny").get().addOnSuccessListener { querySnapshot ->
        val nannyList = mutableListOf<Nanny>()
        for (document in querySnapshot) {
            val nanny = document.toObject(Nanny::class.java).apply {
                id = document.id
            }
            nannyList.add(nanny)
        }
        onSuccess(nannyList)
    }.addOnFailureListener { exception ->
        onFailure(exception)
    }
}

fun getNannies(
    filterCriteria: Map<String, Any>,
    onSuccess: (List<Nanny>) -> Unit,
    onFailure: (Exception) -> Unit
) {
    var query: Query = db.collection("Nanny")

    for ((key, value) in filterCriteria) {
        when (key) {
            "gender" -> query = query.whereEqualTo("gender", value)
            "type" -> query = query.whereEqualTo("type", value)
            "location" -> query = querywhereIn("location", value as
List<String>)
            "skills" -> query = query.whereArrayContains("skills", value)
            "experience" -> query = query.whereEqualTo("experience", value)
        }
    }

    query.get().addOnSuccessListener { querySnapshot ->
        val nannyList = mutableListOf<Nanny>()
        for (document in querySnapshot) {
            val nanny = document.toObject(Nanny::class.java).apply {
                id = document.id
            }
            nannyList.add(nanny)
        }
        onSuccess(nannyList)
    }.addOnFailureListener { exception ->
        onFailure(exception)
    }
}

fun searchNannies(
    queryText: String, onSuccess: (List<Nanny>) -> Unit, onFailure: (Exception) -> Unit
) {
    val lowercaseQueryText = queryText.lowercase()
    db.collection("Nanny").get().addOnSuccessListener { querySnapshot ->
        val result = mutableListOf<Nanny>()
        for (document in querySnapshot) {
            val nanny = document.toObject(Nanny::class.java).apply {
                id = document.id
            }

            if (nanny.age.lowercase().contains(lowercaseQueryText) ||
                nanny.experience.lowercase().contains(lowercaseQueryText) ||
                nanny.gender.lowercase().contains(lowercaseQueryText) ||

```

```

        nanny.location.lowercase().contains(lowercaseQueryText) ||
        nanny.name.lowercase().contains(lowercaseQueryText) ||
        nanny.type.lowercase().contains(lowercaseQueryText) ||
        nanny.skills.any { it.lowercase().contains(lowercaseQueryText) }
    )
    ) {
        result.add(nanny)
    }
    else {
        val pricingQuery = queryText.toIntOrNull()
        if (pricingQuery != null && nanny.pricing == pricingQuery) {
            result.add(nanny)
        }
    }
}
onSuccess(result)
}.addOnFailureListener { exception ->
    onFailure(exception)
}
}
}
}

```

[File adapter NannyAdapter.kt]

```

package com.app.famcare.adapter

import android.content.Context
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.ImageView
import android.widget.TextView
import androidx.recyclerview.widget.RecyclerView
import com.app.famcare.R
import com.app.famcare.model.Nanny
import com.app.famcare.repository.NannyRepository
import com.bumptech.glide.Glide

class NannyAdapter(
    private val context: Context, var layoutResource: Int, private val
    onClick: (String) -> Unit
) : RecyclerView.Adapter<NannyAdapter.NannyViewHolder>() {

    private var nannyList: MutableList<Nanny> = mutableListOf()
    private val nannyRepository = NannyRepository()

    init {
        fetchData()
    }

    class NannyViewHolder(itemView: View) : RecyclerView.ViewHolder(itemView) {
        val imageViewNanny: ImageView = itemView.findViewById(R.id.imageViewNanny)
        val textViewName: TextView = itemView.findViewById(R.id.textViewName)
        val textViewCategory: TextView =
itemView.findViewById(R.id.textViewCategory)
        val textViewRating: TextView = itemView.findViewById(R.id.textViewRating)
    }

    override fun onCreateViewHolder(parent: ViewGroup, viewType: Int):
NannyViewHolder {
        val view = LayoutInflater.from(parent.context).inflate(layoutResource,
parent, false)
        return NannyViewHolder(view)
    }

    override fun onBindViewHolder(holder: NannyViewHolder, position: Int) {
        val nanny = nannyList[position]

```

```

        Glide.with(context).load(nanny.pict).into(holder.imageViewNanny)
        holder.textViewName.text = nanny.name
        holder.textViewCategory.text = nanny.type
        holder.textViewRating.text = nanny.rate

        holder.itemView.setOnClickListener {
            onClick(nanny.id)
        }
    }

    override fun getItemCount(): Int {
        return nannyList.size
    }

    fun fetchData() {
        nannyRepository.getNannies(onSuccess = { nannies ->
            nannyList.clear()
            nannyList.addAll(nannies)
            notifyDataSetChanged()
        }, onFailure = { exception ->
        })
    }

    fun applyFilter(filterCriteria: Map<String, Any>, onFilterApplied: () -> Unit) {
        nannyRepository.getNannies(filterCriteria, onSuccess = { filteredNannies ->
            nannyList.clear()
            nannyList.addAll(filteredNannies)
            notifyDataSetChanged()
            onFilterApplied()
        }, onFailure = { exception ->
        })
    }

    fun updateNannyList(newNannyList: List<Nanny>) {
        nannyList.clear()
        nannyList.addAll(newNannyList)
        notifyDataSetChanged()
    }

    fun updateLayoutResource(newLayoutResource: Int) {
        layoutResource = newLayoutResource
        notifyDataSetChanged()
    }

    fun isListLayout(): Boolean {
        return layoutResource == R.layout.item_list_nanny
    } // Di dalam NannyAdapter.kt

    fun isEmpty(): Boolean {
        return nannyList.isEmpty()
    }

}

```

[File adapter BookmarkAdapter.kt]

```

package com.app.famcare.adapter

import android.content.Context
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.ImageView
import android.widget.TextView
import androidx.recyclerview.widget.RecyclerView

```

```

import com.app.famcare.R
import com.app.famcare.model.Nanny
import com.bumptech.glide.Glide

class BookmarkAdapter(
    private val context: Context,
    private val nannyList: List<Nanny>,
    private val clickListener: OnBookmarkItemClickListener
) : RecyclerView.Adapter<BookmarkAdapter.BookmarkViewHolder>() {

    interface OnBookmarkItemClickListener {
        fun onBookmarkItemClick(nanny: Nanny)
        fun onBookmarkIconClick(nanny: Nanny)
    }

    class BookmarkViewHolder(itemView: View) : RecyclerView.ViewHolder(itemView) {
        val imageViewNanny: ImageView = itemView.findViewById(R.id.imageViewNanny)
        val textViewName: TextView = itemView.findViewById(R.id.textViewName)
        val textViewCategory: TextView =
            itemView.findViewById(R.id.textViewCategory)
        val textViewRating: TextView = itemView.findViewById(R.id.textViewRating)
        val imageViewBookmark: ImageView = itemView.findViewById(R.id.iv_bookmark)
    }

    override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): BookmarkViewHolder {
        val view =
            LayoutInflater.from(parent.context).inflate(R.layout.item_user_bookmark, parent, false)
        return BookmarkViewHolder(view)
    }

    override fun onBindViewHolder(holder: BookmarkViewHolder, position: Int) {
        val nanny = nannyList[position]

        Glide.with(context).load(nanny.pict).placeholder(R.drawable.placeholder_image)
            .into(holder.imageViewNanny)

        holder.textViewName.text = nanny.name
        holder.textViewCategory.text = nanny.type
        holder.textViewRating.text = nanny.rate

        if (nanny.isBookmarked) {
            holder.imageViewBookmark.setImageResource(R.drawable.baseline_bookmark_24)
        } else {
            holder.imageViewBookmark.setImageResource(R.drawable.outline_bookmark_24)
        }

        holder.itemView.setOnClickListener {
            clickListener.onBookmarkItemClick(nanny)
        }

        holder.imageViewBookmark.setOnClickListener {
            clickListener.onBookmarkIconClick(nanny)
        }
    }

    override fun getItemCount(): Int {
        return nannyList.size
    }
}

```

[File activity MainActivity.kt]

```
package com.app.famcare.view.main

import android.content.Intent
import android.os.Bundle
import android.text.Editable
import android.text.TextWatcher
import android.view.View
import android.widget.EditText
import android.widget.ImageView
import android.widget.ProgressBar
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity
import androidx.recyclerview.widget.GridLayoutManager
import androidx.recyclerview.widget.LinearLayoutManager
import androidx.recyclerview.widget.RecyclerView
import com.app.famcare.R
import com.app.famcare.adapter.NannyAdapter
import com.app.famcare.repository.NannyRepository
import com.app.famcare.view.detailpost.DetailPostActivity
import com.app.famcare.view.facilities.FacilitiesActivity
import com.app.famcare.view.history.HistoryActivity
import com.app.famcare.view.profile.ProfileActivity
import com.google.android.material.bottomnavigation.BottomNavigationView
import com.google.firebase.auth.FirebaseAuth
import com.google.firebaseio.firebaseio.FirebaseFirestore

class MainActivity : AppCompatActivity(), FilterFragment.FilterListener {
    private lateinit var editTextSearch: EditText
    private lateinit var imageViewSearch: ImageView
    private lateinit var imageViewFilter: ImageView
    private lateinit var recyclerViewNanny: RecyclerView
    private lateinit var progressBar: ProgressBar
    private lateinit var tvEmptyData: TextView
    private lateinit var ivEmptyData: ImageView
    private lateinit var bottomNavigationView: BottomNavigationView
    private lateinit var tvName: TextView
    private lateinit var listOption: ImageView
    private lateinit var adapter: NannyAdapter
    private val nannyRepository = NannyRepository()
    private var currentQueryText: String = ""
    private var isGrid: Boolean = true

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

        setupViews()

        val auth = FirebaseAuth.getInstance()
        val currentUser = auth.currentUser
        tvName = findViewById(R.id.tv_name)
        val firestore = FirebaseFirestore.getInstance()

        if (currentUser != null) {
            loadUserName(currentUser.uid)
        } else {
            tvName.text = "Guest"
        }

        bottomNavigationView = findViewById(R.id.bottom_navigation)
        bottomNavigationView.selectedItemId = R.id.page_1
        bottomNavigationView.setOnNavigationItemSelectedListener { menuItem ->
            when (menuItem.itemId) {
                R.id.page_1 -> true
                R.id.page_2 -> {

```

```

        startActivity(Intent(this, HistoryActivity::class.java))
        true
    }

    R.id.page_3 -> {
        startActivity(Intent(this, FacilitiesActivity::class.java))
        true
    }

    R.id.page_4 -> {
        startActivity(Intent(this, ProfileActivity::class.java))
        true
    }

    else -> false
}
}

setupRecyclerView(isGrid)
fetchData()

imageViewFilter.setOnClickListener {
    val filterFragment = FilterFragment()
    filterFragment.setFilterListener(this)
    filterFragment.show(supportFragmentManager, "FilterFragment")
}

imageViewSearch.setOnClickListener {
    performSearch()
}

listOption.setOnClickListener {
    toggleRecyclerViewLayout()
}

editTextSearch.addTextChangedListener(object : TextWatcher {
    override fun beforeTextChanged(s: CharSequence?, start: Int, count: Int, after: Int) {}

    override fun onTextChanged(s: CharSequence?, start: Int, before: Int, count: Int) {}

    override fun afterTextChanged(s: Editable?) {
        val queryText = s.toString().trim()
        if (queryText.isEmpty()) {
            currentQueryText = ""
            fetchData()
        } else {
            currentQueryText = queryText
            performSearch()
        }
    }
})
}

private fun loadUserName(userId: String) {
    val userRef =
    FirebaseFirestore.getInstance().collection("User").document(userId)
    userRef.get().addOnSuccessListener { document ->
        if (document != null && document.exists()) {
            val fullName = document.getString("fullName")
            tvName.text = "${fullName ?: "No Name"}!"
        } else {
            tvName.text = "No Document!"
        }
    }.addOnFailureListener { exception ->
        tvName.text = "Error: ${exception.message}!"
    }
}

```

```

        }

    }

    private fun setupViews() {
        editTextSearch = findViewById(R.id.editTextSearch)
        imageViewSearch = findViewById(R.id.imageViewSearch)
        imageViewFilter = findViewById(R.id.imageViewFilter)
        recyclerViewNanny = findViewById(R.id.recyclerViewNanny)
        progressBar = findViewById(R.id.loadingIndicator)
        tvEmptyData = findViewById(R.id.tvEmptyData)
        ivEmptyData = findViewById(R.id.ivEmptyData)
        listOption = findViewById(R.id.listOption)
    }

    private fun setupRecyclerView(isGrid: Boolean) {
        val layoutManager = if (isGrid) {
            GridLayoutManager(this, 2)
        } else {
            LinearLayoutManager(this)
        }
        recyclerViewNanny.layoutManager = layoutManager

        listOption.setImageResource(if (isGrid) R.drawable.ic_list else
R.drawable.ic_grid)

        val layoutResource = if (isGrid) {
            R.layout.item_row_nanny
        } else {
            R.layout.item_list_nanny
        }

        if (!::adapter.isInitialized || adapter.layoutResource != layoutResource)
{
            adapter = NannyAdapter(this, layoutResource) { nannyId ->
                val intent = Intent(this, DetailPostActivity::class.java)
                intent.putExtra("nannyId", nannyId)
                startActivity(intent)
            }
            recyclerViewNanny.adapter = adapter
        } else {
            adapter.updateLayoutResource(layoutResource)
        }
    }

    private fun fetchData() {
        progressBar.visibility = View.VISIBLE
        tvEmptyData.visibility = View.GONE
        ivEmptyData.visibility = View.GONE

        nannyRepository.getNannies(onSuccess = { nannies ->
            progressBar.visibility = View.GONE
            adapter.updateNannyList(nannies)
            checkAdapterEmpty()
        }, onFailure = { exception ->
            progressBar.visibility = View.GONE
        })
    }

    private fun performSearch() {
        val query = editTextSearch.text.toString().trim()
        if (query.isEmpty()) {
            currentQueryText = ""
            fetchData()
        } else {
            currentQueryText = query
            progressBar.visibility = View.VISIBLE
            tvEmptyData.visibility = View.GONE
        }
    }
}

```

```

        nannyRepository.searchNannies(query, onSuccess = { nannies ->
            progressBar.visibility = View.GONE
            if (nannies.isEmpty()) {
                tvEmptyData.visibility = View.VISIBLE
                ivEmptyData.visibility = View.VISIBLE
                recyclerViewNanny.visibility = View.GONE
            } else {
                tvEmptyData.visibility = View.GONE
                ivEmptyData.visibility = View.GONE
                recyclerViewNanny.visibility = View.VISIBLE
                adapter.updateNannyList(nannies)
            }
        }, onFailure = { exception ->
            progressBar.visibility = View.GONE
        })
    }
}

private fun toggleRecyclerViewLayout() {
    isGrid = !isGrid
    setupRecyclerView(isGrid)
    if (currentQueryText.isNotEmpty()) {
        performSearch()
    } else {
        fetchData()
    }
}

override fun onFilterApplied(filterCriteria: Map<String, Any>) {
    if (filterCriteria.isEmpty()) {
        fetchData()
    } else {
        adapter.applyFilter(filterCriteria) {
            checkAdapterEmpty()
        }
    }
}

private fun checkAdapterEmpty() {
    if (adapter.isEmpty()) {
        tvEmptyData.visibility = View.VISIBLE
        ivEmptyData.visibility = View.VISIBLE
        recyclerViewNanny.visibility = View.GONE
    } else {
        tvEmptyData.visibility = View.GONE
        ivEmptyData.visibility = View.GONE
        recyclerViewNanny.visibility = View.VISIBLE
    }
}
}

```

[File fragment FilterFragment.kt]

```

package com.app.famcare.view.main

import android.os.Bundle
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.Button
import android.widget.ToggleButton
import androidx.fragment.app.DialogFragment
import com.app.famcare.R

class FilterFragment : DialogFragment() {
    interface FilterListener {

```

```

        fun onFilterApplied(filterCriteria: Map<String, Any>)
    }

private var filterListener: FilterListener? = null
private var currentFilters: Map<String, Any>? = null

fun setFilterListener(listener: FilterListener) {
    filterListener = listener
}

fun setCurrentFilters(filters: Map<String, Any>?) {
    currentFilters = filters
}

override fun onCreateView(
    inflater: LayoutInflater, container: ViewGroup?, savedInstanceState: Bundle?
): View? {
    return inflater.inflate(R.layout.fragment_filter, container, false)
}

override fun onStart() {
    super.onStart()
    val displayMetrics = resources.displayMetrics
    val width = (displayMetrics.widthPixels * 0.90).toInt()
    val height = ViewGroup.LayoutParams.MATCH_PARENT

    val leftMargin = (displayMetrics.widthPixels * 0.10).toInt()

    dialog?.window?.setLayout(width, height)

    val layoutParams = dialog?.window?.attributes
    layoutParams?.x = leftMargin
    dialog?.window?.attributes = layoutParams

    dialog?.window?.setBackgroundDrawableResource(android.R.color.transparent)
}

override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setStyle(STYLE_NO_FRAME, R.style.DialogTheme)
}

override fun onViewCreated(view: View, savedInstanceState: Bundle?) {
    super.onViewCreated(view, savedInstanceState)
    val buttonApply = view.findViewById<Button>(R.id.buttonApply)
    val buttonReset = view.findViewById<Button>(R.id.buttonReset)

    val buttonNanny = view.findViewById<ToggleButton>(R.id.buttonNanny)
    val buttonManny = view.findViewById<ToggleButton>(R.id.buttonManny)
    val buttonDaily = view.findViewById<ToggleButton>(R.id.buttonDaily)
    val buttonMonthly = view.findViewById<ToggleButton>(R.id.buttonMonthly)
    val buttonJabodetabek =
        view.findViewById<ToggleButton>(R.id.buttonJabodetabek)
    val buttonElderlyCare =
        view.findViewById<ToggleButton>(R.id.buttonElderlyCare)
    val buttonChildCare =
        view.findViewById<ToggleButton>(R.id.buttonChildCare)
    val buttonDisabilityCare =
        view.findViewById<ToggleButton>(R.id.buttonDisabilityCare)
    val buttonExperience =
        view.findViewById<ToggleButton>(R.id.buttonExperience)

    currentFilters?.let { filters ->
        buttonNanny.isChecked = filters["gender"] == "female"
        buttonManny.isChecked = filters["gender"] == "male"
        buttonDaily.isChecked = filters["type"] == "daily"
    }
}

```

```

        buttonMonthly.isChecked = filters["type"] == "monthly"
        buttonJabodetabek.isChecked =
            filters["location"] == listOf("Jakarta", "Bogor", "Depok",
"Tangerang", "Bekasi")
        buttonElderlyCare.isChecked = filters["skills"] == "Elderly Care"
        buttonChildCare.isChecked = filters["skills"] == "Child Care"
        buttonDisabilityCare.isChecked = filters["skills"] == "Disability
Care"
        buttonExperience.isChecked = filters["experience"] == "> 5 years"
    }

    buttonApply.setOnClickListener {
        val filterCriteria = mutableMapOf<String, Any>()

        if (buttonNanny.isChecked) filterCriteria["gender"] = "female"
        if (buttonManny.isChecked) filterCriteria["gender"] = "male"
        if (buttonDaily.isChecked) filterCriteria["type"] = "daily"
        if (buttonMonthly.isChecked) filterCriteria["type"] = "monthly"
        if (buttonJabodetabek.isChecked) filterCriteria["location"] =
            listOf("Jakarta", "Bogor", "Depok", "Tangerang", "Bekasi")
        if (buttonElderlyCare.isChecked) filterCriteria["skills"] = "Elderly
Care"
            if (buttonChildCare.isChecked) filterCriteria["skills"] = "Child Care"
            if (buttonDisabilityCare.isChecked) filterCriteria["skills"] =
"Disability Care"
            if (buttonExperience.isChecked) filterCriteria["experience"] = "> 5
years"

        filterListener?.onFilterApplied(filterCriteria)
        setCurrentFilters(filterCriteria)

        dismiss()
    }

    buttonReset.setOnClickListener {
        resetFilters(view)
        filterListener?.onFilterApplied(emptyMap())
        setCurrentFilters(emptyMap())
        dismiss()
    }

    restoreFilterState(view)
}

private fun resetFilters(view: View) {
    val buttonNanny = view.findViewById<ToggleButton>(R.id.buttonNanny)
    val buttonManny = view.findViewById<ToggleButton>(R.id.buttonManny)
    val buttonDaily = view.findViewById<ToggleButton>(R.id.buttonDaily)
    val buttonMonthly = view.findViewById<ToggleButton>(R.id.buttonMonthly)
    val buttonJabodetabek =
view.findViewById<ToggleButton>(R.id.buttonJabodetabek)
        val buttonElderlyCare =
view.findViewById<ToggleButton>(R.id.buttonElderlyCare)
        val buttonChildCare =
view.findViewById<ToggleButton>(R.id.buttonChildCare)
        val buttonDisabilityCare =
view.findViewById<ToggleButton>(R.id.buttonDisabilityCare)
        val buttonExperience =
view.findViewById<ToggleButton>(R.id.buttonExperience)

        buttonNanny.isChecked = false
        buttonManny.isChecked = false
        buttonDaily.isChecked = false
        buttonMonthly.isChecked = false
        buttonJabodetabek.isChecked = false
        buttonElderlyCare.isChecked = false
        buttonChildCare.isChecked = false
}

```

```

        buttonDisabilityCare.isChecked = false
        buttonExperience.isChecked = false
    }

    private fun restoreFilterState(view: View) {
        val buttonNanny = view.findViewById<ToggleButton>(R.id.buttonNanny)
        val buttonManny = view.findViewById<ToggleButton>(R.id.buttonManny)
        val buttonDaily = view.findViewById<ToggleButton>(R.id.buttonDaily)
        val buttonMonthly = view.findViewById<ToggleButton>(R.id.buttonMonthly)
        val buttonJabodetabek =
            view.findViewById<ToggleButton>(R.id.buttonJabodetabek)
        val buttonElderlyCare =
            view.findViewById<ToggleButton>(R.id.buttonElderlyCare)
        val buttonChildCare =
            view.findViewById<ToggleButton>(R.id.buttonChildCare)
        val buttonDisabilityCare =
            view.findViewById<ToggleButton>(R.id.buttonDisabilityCare)
        val buttonExperience =
            view.findViewById<ToggleButton>(R.id.buttonExperience)

        currentFilters?.let { filters ->
            buttonNanny.isChecked = filters["gender"] == "female"
            buttonManny.isChecked = filters["gender"] == "male"
            buttonDaily.isChecked = filters["type"] == "daily"
            buttonMonthly.isChecked = filters["type"] == "monthly"
            buttonJabodetabek.isChecked = (filters["location"] as?
                List<String>)?.containsAll(
                    listOf("Jakarta", "Bogor", "Depok", "Tangerang", "Bekasi")
                ) ?: false
            buttonElderlyCare.isChecked = filters["skills"] == "Elderly Care"
            buttonChildCare.isChecked = filters["skills"] == "Child Care"
            buttonDisabilityCare.isChecked = filters["skills"] == "Disability
Care"
            buttonExperience.isChecked = filters["experience"] == "> 5 years"
        }
    }
}

```

[File activity DetailPostActivity.kt]

```

package com.app.famcare.view.detailpost

import android.content.Intent
import android.os.Bundle
import android.util.Log
import android.widget.TextView
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity
import androidx.core.content.ContextCompat
import androidx.core.content.ResourcesCompat
import com.app.famcare.R
import com.app.famcare.databinding.ActivityDetailPostBinding
import com.app.famcare.model.Nanny
import com.app.famcare.view.booking.BookDailyActivity
import com.app.famcare.view.booking.BookMonthlyActivity
import com.bumptech.glide.Glide
import com.google.firebase.auth.FirebaseAuth
import com.google.firebase.firestore.FirebaseFirestore
import java.text.NumberFormat
import java.util.*

class DetailPostActivity : AppCompatActivity() {

    private lateinit var binding: ActivityDetailPostBinding
    private lateinit var nannyId: String
    private lateinit var userId: String

```

```

private var nanny: Nanny? = null
private var isBookmarked: Boolean = false

override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    binding = ActivityDetailPostBinding.inflate(layoutInflater)
    setContentView(binding.root)

    nannyId = intent.getStringExtra("nannyId")
        ?: throw IllegalArgumentException("Nanny ID must be provided")

    userId = FirebaseAuth.getInstance().currentUser?.uid
        ?: throw IllegalStateException("User must be logged in to view this
page")

    setSupportActionBar(binding.toolbar)
    supportActionBar?.apply {
        title = ""
        setDisplayHomeAsUpEnabled(true)
        setDisplayShowHomeEnabled(true)
    }

    loadNannyDataFromFirestore()

    binding.buttonBookNanny.setOnClickListener {
        navigateToBookingActivity()
    }

    binding.ivBookmark.setOnClickListener {
        toggleBookmark()
    }

    binding.toolbar.setNavigationOnClickListener {
        onBackPressed()
    }
}

private fun loadNannyDataFromFirestore() {
    val db = FirebaseFirestore.getInstance()
    val docRef = db.collection("Nanny").document(nannyId)

    docRef.get().addOnSuccessListener { document ->
        if (document != null && document.exists()) {
            nanny = document.toObject(Nanny::class.java)

            nanny?.let { nannyData ->
                isBookmarked = document.get("bookmarkedBy.$userId") != null
                val buttonBookNanny = binding.buttonBookNanny
                buttonBookNanny.text = if (nannyData.gender == "male") {
                    "Book Manny"
                } else {
                    "Book Nanny"
                }

                with(binding) {
                    Glide.with(this@DetailPostActivity).load(nannyData.pict)
                        .placeholder(R.drawable.placeholder_image).into(imageViewNannyDP)

                    imageViewGenderDP.setImageResource(
                        if (nannyData.gender == "male") R.drawable.ic_male
                        else R.drawable.ic_female
                    )

                    textViewNameDP.text = nannyData.name
                    textViewAgeDP.text = "${nannyData.age} years old"
                    textViewRateDP.text = nannyData.rate
                }
            }
        }
    }
}

```

```

        textViewTypeDP.text = nannyData.type
        textViewPerHour.text = if (nannyData.type == "daily") "per
hour" else "per month"
        textViewExperienceDP.text = "${nannyData.experience}
experiences"
        textViewLocationDP.text = nannyData.location
        textViewSalaryDP.text =
nannyData.pricing.formatToCurrency()
        skillListLayout.removeAllViews()
        nannyData.skills.forEach { skill ->
            val textView = TextView(this@DetailPostActivity).apply
{
                text = "* $skill"
                setTextColor(
                    ContextCompat.getColor(
                        this@DetailPostActivity,
                        android.R.color.black
                    )
                )
                textSize = 17f
                typeface = ResourcesCompat.getFont(
                    this@DetailPostActivity, R.font.montserrat
                )
            }
            skillListLayout.addView(textView)
        }

        updateBookmarkIcon()
    }
} ?: run {
    Log.e("DetailPostActivity", "Nanny object is null")
    Toast.makeText(this, "Failed to load nanny details",
Toast.LENGTH_SHORT).show()
}
} else {
    Log.e("DetailPostActivity", "Document does not exist or is null")
    Toast.makeText(this, "Nanny data not found",
Toast.LENGTH_SHORT).show()
}
}.addOnFailureListener { exception ->
    Log.e("DetailPostActivity", "Error getting document", exception)
    Toast.makeText(this, "Failed to load nanny data",
Toast.LENGTH_SHORT).show()
}
}

private fun Int.formatToCurrency(): String {
    val format = NumberFormat.getCurrencyInstance(Locale("id", "ID"))
    return format.format(this)
}

private fun toggleBookmark() {
    val db = FirebaseFirestore.getInstance()
    val bookmarkRef = db.collection("Bookmarks").document(nannyId)

    if (isBookmarked) {
        bookmarkRef.delete().addOnSuccessListener {
            isBookmarked = false
            updateBookmarkIcon()
            Toast.makeText(this, "Bookmark removed",
Toast.LENGTH_SHORT).show()
        }.addOnFailureListener { e ->
            Log.e("DetailPostActivity", "Error removing bookmark", e)
            Toast.makeText(this, "Failed to remove bookmark",
Toast.LENGTH_SHORT).show()
        }
    } else {
}
}

```

```

        val bookmarkData = hashMapOf(
            "userId" to userId,
            "nannyId" to nannyId,
            "name" to nanny?.name,
            "type" to nanny?.type,
            "rate" to nanny?.rate,
        )

        bookmarkRef.set(bookmarkData).addOnSuccessListener {
            isBookmarked = true
            updateBookmarkIcon()
            Toast.makeText(this, "Bookmark added", Toast.LENGTH_SHORT).show()
        }.addOnFailureListener { e ->
            Log.e("DetailPostActivity", "Error adding bookmark", e)
            Toast.makeText(this, "Failed to add bookmark",
            Toast.LENGTH_SHORT).show()
        }
    }
}

private fun updateBookmarkIcon() {
    val bookmarkIcon = binding.ivBookmark
    if (isBookmarked) {
        bookmarkIcon.setImageResource(R.drawable.baseline_bookmark_24)
    } else {
        bookmarkIcon.setImageResource(R.drawable.outline_bookmark_24)
    }
}

private fun navigateToBookingActivity() {
    val intent = when (nanny?.type) {
        "daily" -> Intent(this, BookDailyActivity::class.java)
        "monthly" -> Intent(this, BookMonthlyActivity::class.java)
        else -> {
            Toast.makeText(this, "Invalid nanny type",
            Toast.LENGTH_SHORT).show()
            return
        }
    }

    intent.putExtra("nannyId", nannyId)
    intent.putExtra("activityName", this::class.java.simpleName)
    startActivity(intent)
}

override fun onSupportNavigateUp(): Boolean {
    onBackPressed()
    return true
}
}

```

[File activity BookmarkActivity.kt]

```

package com.app.famcare.view.bookmark

import android.content.Intent
import android.os.Bundle
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity
import androidx.recyclerview.widget.GridLayoutManager
import com.app.famcare.adapter.BookmarkAdapter
import com.app.famcare.databinding.ActivityBookmarkBinding
import com.app.famcare.model.Nanny
import com.app.famcare.view.detailpost.DetailPostActivity
import com.google.firebase.auth.FirebaseAuth
import com.google.firebaseio.firestore.FirebaseFirestore

```

```

class BookmarkActivity : AppCompatActivity(),
BookmarkAdapter.OnBookmarkItemClickListener {

    private lateinit var binding: ActivityBookmarkBinding
    private lateinit var adapter: BookmarkAdapter
    private val bookmarkedNannies = mutableListOf<Nanny>()
    private lateinit var userId: String

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityBookmarkBinding.inflate(layoutInflater)
        setContentView(binding.root)

        setSupportActionBar(binding.toolbar)
        supportActionBar?.apply {
            title = ""
            setDisplayHomeAsUpEnabled(true)
            setDisplayShowHomeEnabled(true)
        }

        binding.toolbar.setNavigationOnClickListener {
            onBackPressed()
        }

        userId = FirebaseAuth.getInstance().currentUser?.uid
            ?: throw IllegalStateException("User must be logged in to view this
page")

        adapter = BookmarkAdapter(
            this, bookmarkedNannies, this
        )
        binding.recyclerView.layoutManager = GridLayoutManager(this, 1)
        binding.recyclerView.adapter = adapter

        loadBookmarkedNannies()
    }

    private fun loadBookmarkedNannies() {
        val db = FirebaseFirestore.getInstance()
        val bookmarkRef = db.collection("Bookmarks").whereEqualTo("userId",
userId)

        bookmarkRef.get().addOnSuccessListener { documents ->
            bookmarkedNannies.clear()

            for (document in documents) {
                val nannyId = document.getString("nannyId") ?: continue

                db.collection("Nanny").document(nannyId).get().addOnSuccessListener { nannyDoc ->
                    if (nannyDoc != null && nannyDoc.exists()) {
                        val nanny = nannyDoc.toObject(Nanny::class.java)
                        nanny?.let {
                            it.id = nannyId
                            it.isBookmarked = true
                            bookmarkedNannies.add(it)
                            adapter.notifyDataSetChanged()
                        }
                    }
                }.addOnFailureListener { exception ->
                    exception.printStackTrace()
                    Toast.makeText(
                        this@BookmarkActivity, "Failed to load nanny details",
Toast.LENGTH_SHORT
                    ).show()
                }
            }
        }
    }
}

```

```

        }.addOnFailureListener { exception ->
            exception.printStackTrace()
            Toast.makeText(
                this@BookmarkActivity, "Failed to load bookmarked nannies",
                Toast.LENGTH_SHORT
            ).show()
        }
    }

    override fun onBookmarkItemClick(nanny: Nanny) {
        val intent = Intent(this, DetailPostActivity::class.java)
        intent.putExtra("nannyId", nanny.id)
        startActivity(intent)
    }

    override fun onBookmarkIconClick(nanny: Nanny) {
        val db = FirebaseFirestore.getInstance()
        val bookmarkRef =
db.collection("Bookmarks").document("${userId}_${nanny.id}")

        if (nanny.isBookmarked) {
            bookmarkRef.delete().addOnSuccessListener {
                removeNannyFromList(nanny)
                adapter.notifyDataSetChanged()
                Toast.makeText(this@BookmarkActivity, "Bookmark removed",
Toast.LENGTH_SHORT).show()
            }.addOnFailureListener { e ->
                e.printStackTrace()
                Toast.makeText(
                    this@BookmarkActivity, "Failed to remove bookmark",
                    Toast.LENGTH_SHORT
                ).show()
            }
        } else {
            val bookmarkData = hashMapOf(
                "userId" to userId,
                "nannyId" to nanny.id,
                "name" to nanny.name,
                "type" to nanny.type,
                "rate" to nanny.rate,
            )

            bookmarkRef.set(bookmarkData).addOnSuccessListener {
                nanny.isBookmarked = true
                bookmarkedNannies.add(nanny)
                adapter.notifyDataSetChanged()
                Toast.makeText(this@BookmarkActivity, "Bookmark added",
Toast.LENGTH_SHORT).show()
            }.addOnFailureListener { e ->
                e.printStackTrace()
                Toast.makeText(
                    this@BookmarkActivity, "Failed to add bookmark",
                    Toast.LENGTH_SHORT
                ).show()
            }
        }
    }

    private fun removeNannyFromList(nanny: Nanny) {
        val iterator = bookmarkedNannies.iterator()
        while (iterator.hasNext()) {
            val item = iterator.next()
            if (item.id == nanny.id) {
                iterator.remove()
                break
            }
        }
    }
}

```

```
}
```

[File XML bottom_navigation_menu.xml]

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:id="@+id/page_1"
        android:enabled="true"
        android:icon="@drawable/round_business_center_24"
        android:title="Discover" />
    <item
        android:id="@+id/page_2"
        android:enabled="true"
        android:icon="@drawable/baseline_history_24"
        android:title="History" />
    <item
        android:id="@+id/page_3"
        android:enabled="true"
        android:icon="@drawable/baseline_fmd_good_24"
        android:title="Facilities" />
    <item
        android:id="@+id/page_4"
        android:enabled="true"
        android:icon="@drawable/baseline_person_24"
        android:title="Profile" />
</menu>
```

[File XML item_row_nanny.xml]

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.cardview.widget.CardView
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:clickable="true"
    android:focusable="true"
    app:cardCornerRadius="18dp"
    app:cardElevation="4dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:orientation="vertical"
        android:padding="16dp">

        <de.hdodenhof.circleimageview.CircleImageView
            android:id="@+id/imageViewNanny"
            android:layout_width="100dp"
            android:layout_height="100dp"
            android:scaleType="centerCrop"
            android:src="@drawable/nanny_placeholder" />

        <TextView
            android:id="@+id/textViewName"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="8dp"
            android:fontFamily="@font/montserratsemibold"
            android:gravity="center"
            android:text="Nanny's Name"
            android:textColor="@android:color/black"
```

```

        android:textSize="18sp"
        android:textStyle="bold" />

    <LinearLayout
        android:id="@+id/rating"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_marginTop="8dp"
        android:gravity="center"
        android:orientation="horizontal">

        <ImageView
            android:id="@+id/imageViewRating"
            android:layout_width="25dp"
            android:layout_height="25dp"
            android:layout_marginEnd="4dp"
            android:src="@drawable/ic_rate" />

        <TextView
            android:id="@+id/textViewRating"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:fontFamily="@font/montserrat"
            android:gravity="center_vertical"
            android:text="4.5"
            android:textColor="@android:color/black"
            android:textSize="16sp" />

    </LinearLayout>

    <LinearLayout
        android:id="@+id/category"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_marginTop="4dp"
        android:gravity="center"
        android:orientation="horizontal">

        <ImageView
            android:id="@+id/imageViewCategory"
            android:layout_width="25dp"
            android:layout_height="25dp"
            android:layout_marginEnd="4dp"
            android:src="@drawable/ic_calendar" />

        <TextView
            android:id="@+id/textViewCategory"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:fontFamily="@font/montserrat"
            android:gravity="center_vertical"
            android:text="Category"
            android:textColor="@android:color/black"
            android:textSize="16sp" />

    </LinearLayout>

</LinearLayout>

</androidx.cardview.widget.CardView>

```

[File XML item_list_nanny.xml]

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.cardview.widget.CardView
    xmlns:android="http://schemas.android.com/apk/res/android"

```

```
xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:clickable="true"
    android:focusable="true"
    app:cardCornerRadius="18dp"
    app:cardElevation="8dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:orientation="horizontal"
        android:padding="16dp">

        <de.hdodenhof.circleimageview.CircleImageView
            android:id="@+id/imageViewNanny"
            android:layout_width="100dp"
            android:layout_height="100dp"
            android:scaleType="centerCrop"
            android:src="@drawable/nanny_placeholder" />

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:padding="16dp">

            <TextView
                android:id="@+id/textViewName"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:fontFamily="@font/montserratsemibold"
                android:text="Nanny's Name"
                android:textColor="@android:color/black"
                android:textSize="18sp"
                android:textStyle="bold" />

            <LinearLayout
                android:id="@+id/rating"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"

                android:layout_marginTop="8dp"

                android:orientation="horizontal">

                <ImageView
                    android:id="@+id/imageViewRating"
                    android:layout_width="25dp"
                    android:layout_height="25dp"
                    android:layout_marginEnd="4dp"
                    android:src="@drawable/ic_rate" />

                <TextView
                    android:id="@+id/textViewRating"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:fontFamily="@font/montserrat"
                    android:gravity="center_vertical"
                    android:text="4.5"
                    android:textColor="@android:color/black"
                    android:textSize="16sp" />
            
        
    

```

```

<LinearLayout
    android:id="@+id/category"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="4dp"
    android:orientation="horizontal">

    <ImageView
        android:id="@+id/imageViewCategory"
        android:layout_width="25dp"
        android:layout_height="25dp"
        android:layout_marginEnd="4dp"
        android:src="@drawable/ic_calendar" />

    <TextView
        android:id="@+id/textViewCategory"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:fontFamily="@font/montserrat"
        android:gravity="center_vertical"
        android:text="Category"
        android:textColor="@android:color/black"
        android:textSize="16sp" />
</LinearLayout>
</LinearLayout>
</androidx.cardview.widget.CardView>

```

[File XML activity_main.xml]

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.app.famcare.view.main.MainActivity">

    <RelativeLayout
        android:id="@+id/contentLayout"
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <androidx.cardview.widget.CardView
            android:id="@+id/welcomeLayout"
            android:layout_width="match_parent"
            android:layout_height="100dp"
            android:layout_marginStart="16dp"
            android:layout_marginTop="16dp"
            android:layout_marginEnd="16dp"
            android:layout_marginBottom="10dp"
            android:clipToPadding="true"
            app:cardCornerRadius="12dp">

            <FrameLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content">

                <ImageView
                    android:id="@+id/backgroundImage"
                    android:layout_width="match_parent"
                    android:layout_height="100dp"
                    android:contentDescription="welcome_background"
                    android:scaleType="centerCrop" />

```

```
        android:src="@drawable/welcoming" />

    <View
        android:layout_width="match_parent"
        android:layout_height="200dp"
        android:background="#80000000" />

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="16dp">

        <TextView
            android:id="@+id/welcomeTextView"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:fontFamily="@font/montserratsemibold"
            android:text="Welcome,"
            android:textColor="@android:color/white"
            android:textSize="18sp"
            android:textStyle="bold" />

        <TextView
            android:id="@+id/tv_name"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_below="@+id/welcomeTextView"
            android:layout_marginTop="8dp"
            android:fontFamily="@font/montserratsemibold"
            android:text=" "
            android:textColor="@android:color/white"
            android:textSize="24sp" />
    
```

</RelativeLayout>

```
</FrameLayout>
</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView
    android:id="@+id/SearchBarLayout"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/welcomeLayout"
    android:layout_marginStart="16dp"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="16dp"
    android:clipToPadding="true"
    app:cardCornerRadius="12dp">

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="8dp">

        <EditText
            android:id="@+id/editTextSearch"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_alignParentStart="true"
            android:layout_alignParentLeft="true"
            android:layout_toStartOf="@+id/imageViewSearch"
            android:background="@android:color/transparent"
            android:hint="@string/search_hint"
            android:imeOptions="actionSearch"
            android:inputType="text"
            android:paddingStart="16dp"
            android:paddingEnd="16dp"
            android:singleLine="true"
            android:textColorHint="@android:color/darker_gray" />

```

```
<ImageView  
    android:id="@+id/imageViewSearch"  
    android:layout_width="28dp"  
    android:layout_height="28dp"  
    android:layout_alignParentEnd="true"  
    android:layout_alignParentRight="true"  
    android:layout_centerVertical="true"  
    android:layout_marginStart="8dp"  
    android:layout_marginEnd="8dp"  
    android:adjustViewBounds="true"  
    android:contentDescription="Search"  
    android:src="@drawable/ic_search" />  
  
</RelativeLayout>  
  
</androidx.cardview.widget.CardView>  
  
<RelativeLayout  
    android:id="@+id/bookmarkActivity"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_below="@+id/SearchBarLayout"  
    android:layout_marginStart="16dp"  
    android:layout_marginTop="8dp"  
    android:layout_marginEnd="8dp"  
    android:gravity="center_horizontal"  
    android:paddingTop="10dp"  
    android:paddingBottom="10dp">  
  
<LinearLayout  
    android:id="@+id/leftAlignedBookmark"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_alignParentStart="true"  
    android:gravity="start"  
    android:orientation="horizontal">  
  
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_gravity="center_vertical"  
    android:fontFamily="@font/montserratsemibold"  
    android:text="Find Nanny and Manny"  
    android:textColor="@color/icon_active"  
    android:textSize="18sp" />  
  
</LinearLayout>  
  
<LinearLayout  
    android:id="@+id/rightAlignedArrow"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_alignParentEnd="true"  
    android:layout_marginEnd="12dp"  
    android:gravity="end"  
    android:orientation="horizontal">  
  
<ImageView  
    android:id="@+id/listOption"  
    android:layout_width="30dp"  
    android:layout_height="30dp"  
    android:layout_marginLeft="12dp"  
    android:contentDescription="arrow"  
    android:src="@drawable/ic_list" />
```

```

        <ImageView
            android:id="@+id/imageViewFilter"
            android:layout_width="27dp"
            android:layout_height="27dp"
            android:layout_alignParentEnd="true"
            android:layout_alignParentRight="true"
            android:layout_centerVertical="true"
            android:layout_marginStart="8dp"
            android:layout_marginEnd="8dp"
            android:adjustViewBounds="true"
            android:contentDescription="Filter"
            android:src="@drawable/ic_filterr" />

    </LinearLayout>

</RelativeLayout>

<FrameLayout
    android:id="@+id/recyclerFrameLayout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_below="@+id/bookmarkActivity"
    android:layout_marginTop="0dp"
    android:layout_marginBottom="56dp"
    android:padding="8dp">

    <ProgressBar
        android:id="@+id/loadingIndicator"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:visibility="gone" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center_vertical"
        android:gravity="center"
        android:orientation="vertical">

        <ImageView
            android:id="@+id/ivEmptyData"
            android:layout_width="50dp"
            android:layout_height="50dp"
            android:layout_gravity="center"
            android:src="@drawable/ic_leaf"
            android:visibility="gone" />

        <TextView
            android:id="@+id/tvEmptyData"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:layout_marginTop="16dp"
            android:fontFamily="@font/montserrat"
            android:text="Sorry, no nannies/mannies found"
            android:textColor="@color/dark_grey"
            android:textSize="16sp"
            android:visibility="gone" />
    </LinearLayout>

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recyclerViewNanny"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_below="@+id/bookmarkActivity"
        android:layout_marginTop="0dp"

```

```

        android:padding="8dp"
        app:layoutManager="androidx.recyclerview.widget.GridLayoutManager"
        app:spanCount="2"
        tools:listItem="@layout/item_row_nanny" />
    </FrameLayout>

    <FrameLayout
        android:id="@+id/fragmentLayout"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />

</RelativeLayout>

<com.google.android.material.bottomnavigation.BottomNavigationView
    android:id="@+id/bottom_navigation"
    style="@style/Widget.MaterialComponents.BottomNavigationView.Colored"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="bottom"
    app:menu="@menu/bottom_navigation_menu" />

</androidx.coordinatorlayout.widget.CoordinatorLayout>
```

[File XML fragment_filter.xml]

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@android:color/white"
    tools:context=".view.main.FilterFragment">

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="16dp">

        <LinearLayout
            android:id="@+id/contentFilter"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical">

            <TextView
                android:id="@+id/textViewFilter"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginStart="4dp"
                android:layout_marginTop="2dp"
                android:layout_marginBottom="8dp"
                android:fontFamily="@font/montserratsemibold"
                android:text="Filter"
                android:textColor="@android:color/black"
                android:textSize="22sp"
                android:textStyle="bold" />

            <androidx.cardview.widget.CardView
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginStart="4dp"
                android:layout_marginTop="12dp"
                android:layout_marginEnd="4dp"
                android:layout_marginBottom="8dp"
                app:cardCornerRadius="5dp">
```

```

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginStart="4dp"
        android:layout_marginTop="2dp"
        android:fontFamily="@font/montserrat"
        android:text="Gender"
        android:textColor="@color/black"
        android:textSize="18sp" />

    <LinearLayout
        android:id="@+id/optionGender"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_marginTop="12dp"
        android:layout_marginBottom="16dp"
        android:gravity="center"
        android:orientation="vertical">

        <ToggleButton
            android:id="@+id/buttonNanny"
            android:layout_width="280dp"
            android:layout_height="35dp"
            android:layout_marginBottom="16dp"
            android:background="@drawable/toggle_selector"
            android:fontFamily="@font/montserrat"
            android:text="Female"
            android:textAllCaps="false"
            android:textColor="@color/toggle_text_selector"
            android:textOff="Female"
            android:textOn="Female"
            android:textSize="16sp" />

        <ToggleButton
            android:id="@+id/buttonManny"
            android:layout_width="280dp"
            android:layout_height="35dp"
            android:background="@drawable/toggle_selector"
            android:fontFamily="@font/montserrat"
            android:text="Male"
            android:textAllCaps="false"
            android:textColor="@color/toggle_text_selector"
            android:textOff="Male"
            android:textOn="Male"
            android:textSize="16sp" />
    

```

```
        android:layout_height="wrap_content"
        android:orientation="vertical">

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginStart="4dp"
            android:layout_marginTop="2dp"
            android:fontFamily="@font/montserrat"
            android:text="Type"
            android:textColor="@color/black"
            android:textSize="18sp" />

        <LinearLayout
            android:id="@+id/optionType"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:layout_marginTop="12dp"
            android:layout_marginBottom="16dp"
            android:gravity="center"
            android:orientation="vertical">

            <ToggleButton
                android:id="@+id/buttonDaily"
                android:layout_width="280dp"
                android:layout_height="35dp"
                android:layout_marginBottom="16dp"
                android:background="@drawable/toggle_selector"
                android:fontFamily="@font/montserrat"
                android:text="Daily"
                android:textAllCaps="false"
                android:textColor="@color/toggle_text_selector"
                android:textOff="Daily"
                android:textOn="Daily"
                android:textSize="16sp" />

            <ToggleButton
                android:id="@+id/buttonMonthly"
                android:layout_width="280dp"
                android:layout_height="35dp"
                android:background="@drawable/toggle_selector"
                android:fontFamily="@font/montserrat"
                android:text="Monthly"
                android:textAllCaps="false"
                android:textColor="@color/toggle_text_selector"
                android:textOff="Monthly"
                android:textOn="Monthly"
                android:textSize="16sp" />
        
```

</LinearLayout>

</androidx.cardview.widget.CardView>

```
<androidx.cardview.widget.CardView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="4dp"
    android:layout_marginTop="12dp"
    android:layout_marginEnd="4dp"
    android:layout_marginBottom="8dp"
    app:cardCornerRadius="5dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">
```

```

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="4dp"
    android:layout_marginTop="2dp"
    android:fontFamily="@font/montserrat"
    android:text="Most Popular"
    android:textColor="@color/black"
    android:textSize="18sp" />

<LinearLayout
    android:id="@+id/optionMost1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:layout_marginTop="12dp"
    android:layout_marginBottom="16dp"
    android:gravity="center"
    android:orientation="horizontal">

    <ToggleButton
        android:id="@+id/buttonJabodetabek"
        android:layout_width="130dp"
        android:layout_height="35dp"
        android:background="@drawable/toggle_selector"
        android:fontFamily="@font/montserrat"
        android:text="Jabodetabek"
        android:textAllCaps="false"
        android:textColor="@color/toggle_text_selector"
        android:textOff="Jabodetabek"
        android:textOn="Jabodetabek"
        android:textSize="16sp" />

    <ToggleButton
        android:id="@+id/buttonElderlyCare"
        android:layout_width="130dp"
        android:layout_height="35dp"
        android:layout_marginStart="16dp"
        android:background="@drawable/toggle_selector"
        android:fontFamily="@font/montserrat"
        android:text="Elderly Care"
        android:textAllCaps="false"
        android:textColor="@color/toggle_text_selector"
        android:textOff="Elderly Care"
        android:textOn="Elderly Care"
        android:textSize="16sp" />

```

</LinearLayout>

```

<LinearLayout
    android:id="@+id/optionMost2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:layout_marginTop="8dp"
    android:layout_marginBottom="16dp"
    android:gravity="center"
    android:orientation="horizontal">

    <ToggleButton
        android:id="@+id/buttonChildCare"
        android:layout_width="130dp"
        android:layout_height="35dp"
        android:background="@drawable/toggle_selector"
        android:fontFamily="@font/montserrat"
        android:text="Child Care"
        android:textAllCaps="false"

```

```

        android:textColor="@color/toggle_text_selector"
        android:textOff="Child Care"
        android:textOn="Child Care"
        android:textSize="16sp" />

    <ToggleButton
        android:id="@+id/buttonDisabilityCare"
        android:layout_width="130dp"
        android:layout_height="35dp"
        android:layout_marginStart="16dp"
        android:background="@drawable/toggle_selector"
        android:fontFamily="@font/montserrat"
        android:text="Disability Care"
        android:textAllCaps="false"
        android:textColor="@color/toggle_text_selector"
        android:textOff="Disability Care"
        android:textOn="Disability Care"
        android:textSize="16sp" />
    </LinearLayout>

    <LinearLayout
        android:id="@+id/optionMost3"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_marginTop="8dp"
        android:layout_marginBottom="16dp"
        android:gravity="center"
        android:orientation="horizontal">

        <ToggleButton
            android:id="@+id/buttonExperience"
            android:layout_width="280dp"
            android:layout_height="35dp"
            android:background="@drawable/toggle_selector"
            android:fontFamily="@font/montserrat"
            android:gravity="center"
            android:orientation="horizontal"
            android:text="> 5 Years Experience"
            android:textAllCaps="false"
            android:textColor="@color/toggle_text_selector"
            android:textOff="> 5 Years Experience"
            android:textOn="> 5 Years Experience"
            android:textSize="16sp" />
        </LinearLayout>
    </LinearLayout>
</androidx.cardview.widget.CardView>
</LinearLayout>

<LinearLayout
    android:id="@+id/buttonFilter"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/contentFilter"
    android:layout_gravity="center"
    android:layout_marginTop="100dp"
    android:layout_marginBottom="12dp"
    android:gravity="center"
    android:orientation="horizontal">

    <Button
        android:id="@+id/buttonReset"
        android:layout_width="150dp"
        android:layout_height="50dp"
        android:layout_marginStart="16dp"
        android:backgroundTint="@color/grey"
        android:fontFamily="@font/montserratsemibold"

```

```

        android:gravity="center"
        android:radius="8dp"
        android:text="Reset"
        android:textAllCaps="false"
        android:textColor="@color/white"
        android:textSize="22sp" />

    <Button
        android:id="@+id/buttonApply"
        android:layout_width="150dp"
        android:layout_height="50dp"
        android:layout_marginStart="16dp"
        android:backgroundTint="@color/icon_active"
        android:fontFamily="@font/montserratsemibold"
        android:gravity="center"
        android:radius="8dp"
        android:text="Apply"
        android:textAllCaps="false"
        android:textColor="@color/white"
        android:textSize="22sp" />

    </LinearLayout>
</RelativeLayout>
</androidx.coordinatorlayout.widget.CoordinatorLayout>
```

[File XML activity_detail_post.xml]

```

<androidx.coordinatorlayout.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".view.bookmark.BookmarkActivity">

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/appToolbar"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar">

        <com.google.android.material.appbar.CollapsingToolbarLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            app:layout_scrollFlags="scroll|exitUntilCollapsed">

            <androidx.appcompat.widget.Toolbar
                android:id="@+id/toolbar"
                android:layout_width="match_parent"
                android:layout_height="?attr/actionBarSize"
                android:background="?attr/colorPrimary"
                android:elevation="4dp"
                app:layout_scrollFlags="scroll|enterAlways">

                <TextView
                    android:id="@+id/toolbar_title"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:fontFamily="@font/montserratsemibold"
                    android:text="Nanny's Detail"
                    android:textColor="@color/white"
                    android:textSize="18sp" />

            </androidx.appcompat.widget.Toolbar>
        </com.google.android.material.appbar.CollapsingToolbarLayout>
    </com.google.android.material.appbar.AppBarLayout>
```

```
<androidx.core.widget.NestedScrollView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="fill_vertical"
    android:layout_marginBottom="72dp"
    android:fillViewport="true"
    app:layout_behavior="@string/appbar_scrolling_view_behavior">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <RelativeLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:padding="16dp">

            <RelativeLayout
                android:id="@+id/ratingLayout"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginTop="6dp"
                android:gravity="center_horizontal">

                <LinearLayout
                    android:id="@+id/leftAlignedRating"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:layout_alignParentStart="true"
                    android:gravity="start"
                    android:orientation="horizontal">

                    <ImageView
                        android:id="@+id/imageViewRate"
                        android:layout_width="25dp"
                        android:layout_height="25dp"
                        android:layout_marginStart="8dp"
                        android:contentDescription="rate"
                        android:src="@drawable/ic_rate" />

                    <TextView
                        android:id="@+id/textViewRateDP"
                        android:layout_width="wrap_content"
                        android:layout_height="wrap_content"
                        android:layout_marginStart="8dp"
                        android:fontFamily="@font/montserratsemibold"
                        android:textColor="@android:color/black"
                        android:textSize="18sp"
                        tools:text="4.0" />
                
```

```

        android:src="@drawable/ic_calendar" />

    <TextView
        android:id="@+id/textViewTypeDP"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="8dp"
        android:fontFamily="@font/montserratsemibold"
        android:textColor="@android:color/black"
        android:textSize="18sp"
        tools:text="daily" />
    </LinearLayout>

</RelativeLayout>

<LinearLayout
    android:id="@+id/imageViewNannyBookmark"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/ratingLayout"
    android:layout_gravity="center"
    android:layout_marginTop="16dp"
    android:orientation="vertical">

    <de.hdodenhof.circleimageview.CircleImageView
        android:id="@+id/imageViewNannyDP"
        android:layout_width="160dp"
        android:layout_height="160dp"
        android:layout_gravity="center"
        android:contentDescription="nanny image"
        android:src="@drawable/nanny_placeholder" />

    <ImageView
        android:id="@+id/iv_bookmark"
        android:layout_width="53dp"
        android:layout_height="53dp"
        android:layout_gravity="center"
        android:layout_marginStart="60dp"
        android:layout_marginTop="-35dp"
        android:contentDescription="bookmark"
        android:src="@drawable/baseline_bookmark_24" />
    </LinearLayout>

    <ImageView
        android:id="@+id/imageViewGenderDP"
        android:layout_width="40dp"
        android:layout_height="40dp"
        android:layout_below="@+id/imageViewNannyBookmark"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="8dp"
        android:contentDescription="gender"
        android:src="@drawable/ic_male" />

    <TextView
        android:id="@+id/textViewNameDP"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/imageViewGenderDP"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="8dp"
        android:fontFamily="@font/montserratsemibold"
        android:textColor="@android:color/black"
        android:textSize="20sp"
        android:textStyle="bold"
        tools:text="Nanny's Name" />

    <TextView

```

```
        android:id="@+id/textViewAgeDP"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/textViewNameDP"
        android:layout_centerHorizontal="true"
        android:layout_marginStart="8dp"
        android:fontFamily="@font/montserratsemibold"
        android:textColor="@android:color/black"
        android:textSize="16sp"
        tools:text="24 years old" />

    <LinearLayout
        android:id="@+id/experienceLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/textViewAgeDP"
        android:layout_marginTop="25dp"
        android:orientation="horizontal">

        <ImageView
            android:id="@+id/imageViewExperience"
            android:layout_width="22dp"
            android:layout_height="22dp"
            android:contentDescription="experience"
            android:src="@drawable/ic_timespan" />

        <TextView
            android:id="@+id/textViewExperienceDP"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="8dp"
            android:fontFamily="@font/montserratsemibold"
            android:textColor="@android:color/black"
            android:textSize="17sp"
            tools:text="@string/years_experiences" />
    </LinearLayout>

    <LinearLayout
        android:id="@+id/locationLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/experienceLayout"
        android:layout_marginTop="15dp"
        android:orientation="horizontal">

        <ImageView
            android:id="@+id/imageViewLocation"
            android:layout_width="22dp"
            android:layout_height="22dp"
            android:contentDescription="location"
            android:src="@drawable/ic_location" />

        <TextView
            android:id="@+id/textViewLocationDP"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="8dp"
            android:fontFamily="@font/montserratsemibold"
            android:textColor="@android:color/black"
            android:textSize="17sp"
            tools:text="Jakarta" />
    </LinearLayout>

    <LinearLayout
        android:id="@+id/salaryLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="25dp"
        android:orientation="horizontal">
```

```

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/locationLayout"
        android:layout_marginTop="8dp"
        android:orientation="horizontal">

        <ImageView
            android:id="@+id/imageViewSalary"
            android:layout_width="22dp"
            android:layout_height="22dp"
            android:contentDescription="salary"
            android:src="@drawable/ic_salary" />

        <TextView
            android:id="@+id/textViewSalaryDP"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="8dp"
            android:fontFamily="@font/montserratsemibold"
            android:textColor="@android:color/black"
            android:textSize="17sp"
            tools:text="@string/minimum_salary" />
        <TextView
            android:id="@+id/textViewPerHour"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="7dp"
            android:fontFamily="@font/montserratsemibold"
            android:textColor="@android:color/black"
            android:textSize="17sp"
            tools:text="per hour" />

    </LinearLayout>

    <LinearLayout
        android:id="@+id/skillLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/locationLayout"
        android:layout_marginTop="40dp"
        android:orientation="vertical">

        <LinearLayout
            android:id="@+id/skillLayouting"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_below="@+id/locationLayout"
            android:layout_marginTop="8dp"
            android:orientation="horizontal">

            <ImageView
                android:id="@+id/imageViewSkill"
                android:layout_width="22dp"
                android:layout_height="22dp"
                android:contentDescription="skill"
                android:src="@drawable/ic_skills" />

            <TextView
                android:id="@+id/textSkill"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_marginStart="8dp"
                android:fontFamily="@font/montserratsemibold"
                android:text="Skills Offered"
                android:textColor="@android:color/black"
                android:textSize="17sp" />

        </LinearLayout>
    </LinearLayout>

```

```

        </LinearLayout>

        <LinearLayout
            android:id="@+id/skillListLayout"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginStart="30dp"
            android:layout_marginTop="5dp"
            android:orientation="vertical" />
    </LinearLayout>

    <Button
        android:id="@+id/buttonBookNanny"
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:layout_below="@+id/skillLayout"
        android:layout_centerHorizontal="true"
        android:layout_gravity="center"
        android:layout_marginStart="32dp"
        android:layout_marginTop="35dp"
        android:layout_marginEnd="32dp"
        android:layout_marginBottom="15dp"
        android:fontFamily="@font/montserratsemibold"
        android:text="Book Nanny"
        android:textSize="20sp"
        android:textStyle="bold" />

</RelativeLayout>

</LinearLayout>
</androidx.core.widget.NestedScrollView>
</androidx.coordinatorlayout.widget.CoordinatorLayout>

```

[File XML item_user_bookmark.xml]

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.cardview.widget.CardView
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/card_view"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="center_horizontal"
    android:layout_margin="4dp"
    android:elevation="8dp"
    app:cardBackgroundColor="@color/white"
    app:cardCornerRadius="10dp">

    <androidx.constraintlayout.widget.ConstraintLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="8dp">

        <de.hdodenhof.circleimageview.CircleImageView
            android:id="@+id/imageViewNanny"
            android:layout_width="88dp"
            android:layout_height="88dp"
            android:layout_marginStart="16dp"
            android:layout_marginTop="16dp"
            android:layout_marginEnd="16dp"
            android:layout_marginBottom="16dp"
            android:scaleType="centerCrop"
            android:src="@drawable/nanny_placeholder"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintEnd_toStartOf="@+id/guideline"
            app:layout_constraintStart_toStartOf="parent" />
    
```

```

        app:layout_constraintTop_toTopOf="parent" />

    <ImageView
        android:id="@+id/ratingImageView"
        android:layout_width="20dp"
        android:layout_height="20dp"
        android:layout_marginStart="2dp"
        android:layout_marginTop="4dp"
        app:layout_constraintStart_toStartOf="@+id/guideline"
        app:layout_constraintTop_toBottomOf="@+id/scheduleImageView"
        app:srcCompat="@drawable/ic_calendar" />

    <ImageView
        android:id="@+id/scheduleImageView"
        android:layout_width="20dp"
        android:layout_height="20dp"
        android:layout_marginStart="2dp"
        android:layout_marginTop="10dp"
        app:layout_constraintStart_toStartOf="@+id/guideline"
        app:layout_constraintTop_toBottomOf="@+id/textViewName"
        app:srcCompat="@drawable/ic_rate" />

    <TextView
        android:id="@+id/textViewName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="2dp"
        android:layout_marginTop="16dp"
        android:fontFamily="@font/montserratsemibold"
        android:text="Nanny Name"
        android:textSize="18sp"
        app:layout_constraintStart_toStartOf="@+id/guideline"
        app:layout_constraintTop_toTopOf="parent" />

    <TextView
        android:id="@+id/textViewRating"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="3dp"
        android:layout_marginTop="10dp"
        android:fontFamily="@font/montserrat"
        android:text="4.5"
        android:textSize="16sp"
        app:layout_constraintEnd_toStartOf="@+id/guideline2"
        app:layout_constraintHorizontal_bias="0.098"
        app:layout_constraintStart_toEndOf="@+id/scheduleImageView"
        app:layout_constraintTop_toBottomOf="@+id/textViewName" />

    <TextView
        android:id="@+id/textViewCategory"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="3dp"
        android:layout_marginTop="0dp"
        android:text="monthly"
        android:textSize="16sp"
        app:layout_constraintEnd_toStartOf="@+id/guideline2"
        app:layout_constraintHorizontal_bias="0.128"
        app:layout_constraintStart_toEndOf="@+id/ratingImageView"
        app:layout_constraintTop_toBottomOf="@+id/textViewRating" />

    <ImageView
        android:id="@+id/iv_bookmark"
        android:layout_width="50dp"
        android:layout_height="50dp"
        android:layout_marginStart="16dp"

```

```

        android:layout_marginTop="16dp"
        android:layout_marginEnd="16dp"
        android:layout_marginBottom="16dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.68"
        app:layout_constraintStart_toStartOf="@+id/guideline2"
        app:layout_constraintTop_toTopOf="parent"
        app:srcCompat="@drawable/baseline_bookmark_24" />

    <androidx.constraintlayout.widget.Guideline
        android:id="@+id/guideline"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        app:layout_constraintGuide_begin="110dp" />

    <androidx.constraintlayout.widget.Guideline
        android:id="@+id/guideline2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        app:layout_constraintGuide_begin="315dp" />

</androidx.constraintlayout.widget.ConstraintLayout>
</androidx.cardview.widget.CardView>
```

[File XML activity_bookmark.xml]

```

<androidx.coordinatorlayout.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".view.bookmark.BookmarkActivity">

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <com.google.android.material.appbar.AppBarLayout
            android:id="@+id/toolbar_layout"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar">

            <androidx.appcompat.widget.Toolbar
                android:id="@+id/toolbar"
                android:layout_width="match_parent"
                android:layout_height="?attr/actionBarSize"
                android:background="?attr/colorPrimary"
                android:elevation="4dp"
                app:layout_scrollFlags="scroll|enterAlways">

                <TextView
                    android:id="@+id/toolbar_title"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:fontFamily="@font/montserratsemibold"
                    android:text="Bookmark"
                    android:textColor="@color/white"
                    android:textSize="18sp" />

            </androidx.appcompat.widget.Toolbar>

```

```
</com.google.android.material.appbar.AppBarLayout>

<androidx.recyclerview.widget.RecyclerView
    android:id="@+id/recyclerView"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/toolbar_layout"
    android:layout_alignParentBottom="true"
    android:padding="8dp"
    tools:listitem="@layout/item_user_bookmark" />
</RelativeLayout>
</CoordinatorLayout>
```

Penjelasan:

Pada dasarnya fungsi utama menu Discover adalah untuk membantu pengguna menemukan dan mengelola pengasuh anak dengan memanfaatkan Firebase Firestore sebagai sumber data dan RecyclerView untuk menampilkan daftar pengasuh. Secara keseluruhan, menu Discover ini terdiri dari beberapa komponen utama: model data, repository, adaptor, fragment, dan activity.

Model data (Nanny.kt) mendefinisikan struktur objek pengasuh dengan berbagai atribut seperti nama, umur, pengalaman, lokasi, dan lainnya. Objek ini juga mengimplementasikan Parcelable untuk memudahkan pengiriman data antar komponen Android. Data pengasuh dapat dibaca dari dan ditulis ke sebuah Parcel.

Repositori (NannyRepository.kt) bertanggung jawab untuk mengambil dan menyaring data dari Firestore, berinteraksi langsung dengan Firestore untuk mengembalikan data sesuai permintaan pencarian dan filter. Metode dalam repositori ini mengambil data pengasuh dari Firestore, baik secara keseluruhan, berdasarkan kriteria filter, maupun berdasarkan teks pencarian. NannyRepository menggunakan metode get() untuk mendapatkan data dari koleksi "Nanny" di Firestore dan juga menggunakan addOnSuccessListener serta addOnFailureListener untuk menangani hasil sukses atau gagal dari permintaan data. NannyRepository juga memiliki metode getFilteredNanny yang menerima parameter untuk filter, seperti kategori "Nanny", "Daily", atau "Disability Care" yang ada pada pilihan filter aplikasi FamCare.

Adaptor (NannyAdapter.kt dan BookmarkAdapter.kt) digunakan untuk menampilkan data pengasuh dalam tampilan list nanny dan manny serta mengatur menampilkan data dalam bentuk list atau grid dalam RecyclerView. Adaptor ini mengatur bagaimana setiap item pengasuh ditampilkan dan berinteraksi. Metode onBindViewHolder mengatur tampilan setiap item, termasuk menampilkan gambar dan informasi pengasuh, serta menangani klik pada item.

Aktivitas utama (MainActivity.kt) adalah pusat dari aplikasi ini. Aktivitas ini mengatur tampilan utama aplikasi, menangani pencarian dan filter pengasuh, serta navigasi antar halaman. Saat aplikasi dijalankan, MainActivity akan memuat data pengasuh dari repositori dan menampilkannya menggunakan adaptor. Pengguna dapat mencari pengasuh dengan mengetikkan teks di kotak pencarian atau menggunakan filter untuk menyaring hasil. Aktivitas ini juga memungkinkan pengguna untuk berganti tampilan antara daftar dan grid, serta navigasi ke halaman

lain seperti history, facilities dan profile. Pada saat onCreate, MainActivity menginisialisasi tampilan dan mengatur RecyclerView dengan NannyAdapter. Data pengasuh diambil dari NannyRepository dan diperbarui ke dalam adaptor. Saat pengguna memasukkan teks ke dalam kotak pencarian, metode performSearch dipanggil. performSearch mengirimkan teks pencarian ke NannyRepository melalui metode searchNannyByText. NannyRepository kemudian melakukan query ke Firestore untuk mencari pengasuh yang sesuai dengan teks pencarian. Hasil pencarian dikembalikan ke MainActivity dan diperbarui dalam RecyclerView melalui NannyAdapter.

Di dalam MainActivity ini, terdapat tampilan fragment filter. Fragment filter ini pada dasarnya menyediakan antarmuka bagi pengguna untuk memilih filter populer pada aplikasi FamCare yang dapat diterapkan pada pencarian pengasuh. Fragment ini mungkin mengandung pengaturan logic tiap filter yang ditampilkan toggle button atau yang memungkinkan pengguna memilih filter. Setelah pengguna memilih filter dan menekan tombol "Apply", FragmentFilter mengumpulkan semua pilihan filter dan mengirimkannya langsung ke repositori untuk melakukan pencarian/filter. Di sini, NannyRepository memanggil applyFilters() yang akan memanggil getFilteredNanny dengan parameter filter yang sesuai. Data yang difilter kemudian dikirim kembali ke MainActivity dan diperbarui dalam RecyclerView melalui NannyAdapter.

Dengan demikian, integrasi dari model data, repositori, adaptor, fragment, dan aktivitas utama membentuk menu Discover pada aplikasi FamCare memungkinkan pengguna untuk dengan mudah mencari, melihat, dan mengelola pengasuh anak sesuai kebutuhan mereka. Pengguna juga bisa membuka halaman detail pengasuh atau halaman lain seperti riwayat dan profil melalui BottomNavigationView.

2.2.6 Booking, History, dan Chat

[File data class Booking.kt]

```
package com.app.famcare.model

data class BookingDaily(
    val bookID: String,
    val nannyName: String,
    val bookDate: String,
    val bookHours: String,
    var endHours: String,
    val type: BookingType
)

data class BookingMonthly(
    val bookID: String,
    val nannyName: String,
    val startDate: String,
    val endDate: String,
    val type: BookingType
)

data class BookingDailyHistory(
    val bookID: String = "",
    val nannyName: String = "",
    val bookDate: String = "",
```

```

    val bookHours: String = "",
    var bookDuration: String = "",
    var startTime: String = "",
    var endHours: String = "",
    var salary: String = "",
    val type: BookingType = BookingType.DAILY,
    val nannyID: String = "",
    val totalCost: String = "",
    val totalPricing: Long = 0,
)

data class BookingMonthlyHistory(
    val bookID: String = "",
    val nannyName: String = "",
    val startDate: String = "",
    val endDate: String = "",
    val type: BookingType = BookingType.MONTHLY,
    val nannyID: String = "",
    val totalCost: String = "",
    val totalPricing: Long = 0,
)

enum class BookingType {
    DAILY, MONTHLY
}

```

[File data class ChatMessage.kt]

```

package com.app.famcare.view.chat

data class ChatMessage(
    val senderId: String = "",
    var senderName: String = "",
    var senderPhotoUrl: String = "",
    val message: String = "",
    val timestamp: Long = 0L,
    val formattedTimestamp: String = "",
    var receiverId: String = ""
)

```

[File adapter ChatAdapter.kt]

```

package com.app.famcare.view.chat

import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import androidx.recyclerview.widget.RecyclerView
import com.app.famcare.R
import com.app.famcare.databinding.ItemMessageBinding
import com.bumptech.glide.Glide
import com.google.firebase.firestore.FirebaseFirestore
import kotlinx.coroutines.Dispatchers
import kotlinx.coroutines.GlobalScope
import kotlinx.coroutines.launch
import kotlinx.coroutines.tasks.await
import java.text.SimpleDateFormat
import java.util.Date
import java.util.Locale

class ChatAdapter : RecyclerView.Adapter<ChatAdapter.MessageViewHolder>() {

    private val messages: MutableList<ChatMessage> = mutableListOf()
    private val firestore = FirebaseFirestore.getInstance()

    inner class MessageViewHolder(itemView: View) :
        RecyclerView.ViewHolder(itemView) {

```

```

private val binding = ItemMessageBinding.bind(itemView)

fun bind(message: ChatMessage) {
    GlobalScope.launch(Dispatchers.Main) {
        val latestUserInfo = getLatestUserInfo(message.senderId)
        val displayName = latestUserInfo?.first ?: message.senderName
        val profileImageUrl = latestUserInfo?.second

        binding.tvMessenger.text = displayName
        if (profileImageUrl != null) {

Glide.with(binding.ivMessenger.context).load(profileImageUrl).circleCrop()
            .into(binding.ivMessenger)
        } else {

Glide.with(binding.ivMessenger.context).load(R.drawable.user).circleCrop()
            .into(binding.ivMessenger)
        }
        binding.tvMessage.text = message.message
        binding.tvTimestamp.text = formatTime(message.timestamp)
    }

    private fun formatTime(timestamp: Long): String {
        val dateFormat = SimpleDateFormat("HH:mm", Locale.getDefault())
        return dateFormat.format(Date(timestamp))
    }

    private suspend fun getLatestUserInfo(senderId: String): Pair<String?, String?>? {
        return try {
            val document =
firestore.collection("User").document(senderId).get().await()
            val fullName = document.getString("fullName")
            val profileImageUrl = document.getString("profileImageUrl")
            Pair(fullName, profileImageUrl)
        } catch (e: Exception) {
            null
        }
    }
}

override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): MessageViewHolder {
    val view =
LayoutInflater.from(parent.context).inflate(R.layout.item_message, parent, false)
    return MessageViewHolder(view)
}

override fun onBindViewHolder(holder: MessageViewHolder, position: Int) {
    val message = messages[position]
    holder.bind(message)
}

override fun getItemCount(): Int {
    return messages.size
}

fun addMessage(message: ChatMessage) {
    messages.add(message)
    notifyItemInserted(messages.size - 1)
}
}

```

[File adapter BookingAdapter.kt]

```
package com.app.famcare.adapter
```

```

import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.ImageView
import android.widget.TextView
import androidx.recyclerview.widget.RecyclerView
import com.app.famcare.R
import com.app.famcare.model.BookingDaily
import com.app.famcare.model.BookingMonthly
import com.app.famcare.model.BookingMonthlyHistory

class BookingAdapter(
    private var bookingList: List<Any>, private val listener: OnItemClickListener
) : RecyclerView.Adapter<RecyclerView.ViewHolder>() {

    interface OnItemClickListener {
        fun onItemClick(bookingID: String)
        fun onChatClick(bookingID: String)
    }

    override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): RecyclerView.ViewHolder {
        return when (viewType) {
            DAILY_VIEW_TYPE -> {
                val view = LayoutInflater.from(parent.context)
                    .inflate(R.layout.item_row_history_b_d, parent, false)
                DailyViewHolder(view)
            }
            MONTHLY_VIEW_TYPE -> {
                val view = LayoutInflater.from(parent.context)
                    .inflate(R.layout.item_row_history_b_m, parent, false)
                MonthlyViewHolder(view)
            }
            MONTHLY_HISTORY_VIEW_TYPE -> {
                val view = LayoutInflater.from(parent.context).inflate(
                    R.layout.item_row_history_b_m, parent, false
                )
                MonthlyHistoryViewHolder(view)
            }
            else -> throw IllegalArgumentException("Invalid view type")
        }
    }

    override fun onBindViewHolder(holder: RecyclerView.ViewHolder, position: Int) {
        val booking = bookingList[position]
        when (holder) {
            is DailyViewHolder -> {
                val bookingDaily = booking as BookingDaily
                holder.bind(bookingDaily)
            }
            is MonthlyViewHolder -> {
                val bookingMonthly = booking as BookingMonthly
                holder.bind(bookingMonthly)
            }
            is MonthlyHistoryViewHolder -> {
                val bookingMonthlyHistory = booking as BookingMonthlyHistory
                holder.bind(bookingMonthlyHistory)
            }
        }
    }
}

```

```

        override fun getItemCount(): Int {
            return bookingList.size
        }

        override fun getItemViewType(position: Int): Int {
            return when (bookingList[position]) {
                is BookingDaily -> DAILY_VIEW_TYPE
                is BookingMonthly -> MONTHLY_VIEW_TYPE
                is BookingMonthlyHistory -> MONTHLY_HISTORY_VIEW_TYPE
                else -> throw IllegalArgumentException("Invalid booking type")
            }
        }

        inner class DailyViewHolder(itemView: View) :
    RecyclerView.ViewHolder(itemView) {
            private val textViewID: TextView = itemView.findViewById(R.id.textViewID)
            private val textViewName: TextView =
    itemView.findViewById(R.id.textViewName)
            private val textViewDate: TextView =
    itemView.findViewById(R.id.textViewDate)
            private val textViewBookHours: TextView =
    itemView.findViewById(R.id.textViewBookHours)
            private val textViewEndHours: TextView =
    itemView.findViewById(R.id.textViewEndHours)
            private val chatIcon: ImageView = itemView.findViewById(R.id.chatIcon)

            fun bind(booking: BookingDaily) {
                textViewID.text = booking.bookID
                textViewName.text = booking.nannyName
                textViewDate.text = booking.bookDate
                textViewBookHours.text = booking.bookHours
                textViewEndHours.text = booking.endHours

                itemView.setOnClickListener {
                    val position = adapterPosition
                    if (position != RecyclerView.NO_POSITION) {
                        listener.onItemClick(booking.bookID)
                    }
                }
            }

            chatIcon.setOnClickListener {
                val position = adapterPosition
                if (position != RecyclerView.NO_POSITION) {
                    listener.onChatClick(booking.bookID)
                }
            }
        }

        inner class MonthlyViewHolder(itemView: View) :
    RecyclerView.ViewHolder(itemView) {
            private val textViewID: TextView = itemView.findViewById(R.id.textViewID)
            private val textViewName: TextView =
    itemView.findViewById(R.id.textViewName)
            private val textViewStartDate: TextView =
    itemView.findViewById(R.id.textViewStartDate)
            private val textViewEndDate: TextView =
    itemView.findViewById(R.id.textViewEndDate)
            private val chatIcon: ImageView = itemView.findViewById(R.id.chatIcon)

            fun bind(booking: BookingMonthly) {
                textViewID.text = booking.bookID
                textViewName.text = booking.nannyName
                textViewStartDate.text = booking.startDate
                textViewEndDate.text = booking.endDate
            }
        }
    }
}

```

```

        itemView.setOnClickListener {
            val position = adapterPosition
            if (position != RecyclerView.NO_POSITION) {
                listener.onItemClick(booking.bookID)
            }
        }

        chatIcon.setOnClickListener {
            val position = adapterPosition
            if (position != RecyclerView.NO_POSITION) {
                listener.onChatClick(booking.bookID)
            }
        }
    }
}

inner class MonthlyHistoryViewHolder(itemView: View) : RecyclerView.ViewHolder(itemView) {
    private val textViewID: TextView = itemView.findViewById(R.id.textViewID)
    private val textViewName: TextView =
itemView.findViewById(R.id.textViewName)
    private val textViewStartDate: TextView =
itemView.findViewById(R.id.textViewStartDate)
    private val textViewEndDate: TextView =
itemView.findViewById(R.id.textViewEndDate)
    private val chatIcon: ImageView = itemView.findViewById(R.id.chatIcon)

    fun bind(booking: BookingMonthlyHistory) {
        textViewID.text = booking.bookID
        textViewName.text = booking.nannyName
        textViewStartDate.text = booking.startDate
        textViewEndDate.text = booking.endDate

        itemView.setOnClickListener {
            val position = adapterPosition
            if (position != RecyclerView.NO_POSITION) {
                listener.onItemClick(booking.bookID)
            }
        }

        chatIcon.setOnClickListener {
            val position = adapterPosition
            if (position != RecyclerView.NO_POSITION) {
                listener.onChatClick(booking.bookID)
            }
        }
    }
}

fun setData(newList: List<Any>) {
    bookingList = newList
    notifyDataSetChanged()
}

companion object {
    private const val DAILY_VIEW_TYPE = 1
    private const val MONTHLY_VIEW_TYPE = 2
    private const val MONTHLY_HISTORY_VIEW_TYPE = 3
}
}

```

[File adapter HistoryPagerAdapter.kt]

```

package com.app.famcare.adapter

import androidx.fragment.app.Fragment
import androidx.fragment.app.FragmentActivity

```

```

import androidx.viewpager2.adapter.FragmentStateAdapter
import com.app.famcare.view.historyimport.HistoryBDFragment
import com.app.famcare.view.historyimport.HistoryBMFragment

class HistoryPagerAdapter(fragmentActivity: FragmentActivity) :
    FragmentStateAdapter(fragmentActivity) {

    override fun getItemCount(): Int {
        return NUM_PAGES
    }

    override fun createFragment(position: Int): Fragment {
        return when (position) {
            0 -> HistoryBDFragment()
            1 -> HistoryBMFragment()
            else -> throw IllegalArgumentException("Invalid position $position")
        }
    }

    companion object {
        private const val NUM_PAGES = 2
    }
}

```

[File activity BookDailyActivity.kt]

```

package com.app.famcare.view.booking

import android.app.AlertDialog
import android.app.DatePickerDialog
import android.app.ProgressDialog
import android.app.TimePickerDialog
import android.content.DialogInterface
import android.content.Intent
import android.os.Bundle
import android.util.Log
import android.view.View
import android.widget.*
import androidx.appcompat.app.AppCompatActivity
import androidx.appcompat.widget.Toolbar
import com.app.famcare.R
import com.app.famcare.model.Nanny
import com.app.famcare.view.history.HistoryActivity
import com.bumptech.glide.Glide
import com.google.firebase.auth.FirebaseAuth
import com.google.firebase.firestore.FieldValue
import com.google.firebase.firestore.FirebaseFirestore
import java.text.NumberFormat
import java.text.ParseException
import java.text.SimpleDateFormat
import java.util.*

class BookDailyActivity : AppCompatActivity() {
    private lateinit var bookingEditText: EditText
    private lateinit var bookingDurationSpinner: Spinner
    private lateinit var startHourEditText: EditText
    private lateinit var outputBookingDate: TextView
    private lateinit var outputBookingDuration: TextView
    private lateinit var totalPricingText: TextView
    private lateinit var outputStartHour: TextView
    private lateinit var outputEndHour: TextView
    private lateinit var buttonBookNanny: Button
    private lateinit var auth: FirebaseAuth
    private lateinit var nannyId: String
    private lateinit var db: FirebaseFirestore
    private var selectedDate: String = ""
    private var selectedDuration: String = ""

```

```

private var selectedStartHour: String = ""
private var selectedEndHour: String = ""

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

    val toolbar = findViewById<Toolbar>(R.id.toolbar)
    setSupportActionBar(toolbar)
    supportActionBar?.setDisplayHomeAsUpEnabled(true)
    supportActionBar?.title = ""

    auth = FirebaseAuth.getInstance()
    db = FirebaseFirestore.getInstance()

    nannyId = intent.getStringExtra("nannyId") ?: ""

    loadNannyDataFromFirestore(nannyId)

    bookingDateEditText = findViewById(R.id.bookingDateEditText)
    bookingDurationSpinner = findViewById(R.id.bookingDurationSpinner)
    startHourEditText = findViewById(R.id.startHourEditText)
    outputBookingDate = findViewById(R.id.outputBookingDate)
    outputBookingDuration = findViewById(R.id.outputBookingDuration)
    outputStartHour = findViewById(R.id.outputStartHour)
    outputEndHour = findViewById(R.id.outputEndHour)
    buttonBookNanny = findViewById(R.id.buttonBookNanny)
    totalPricingText = findViewById(R.id.textCost)
    totalPricingText.text = "Total Pricing :"

    bookingDateEditText.setOnClickListener {
        showDatePickerDialog()
    }

    val durationAdapter = ArrayAdapter.createFromResource(
        this, R.array.working_hours_duration,
        android.R.layout.simple_spinner_item
    )

    durationAdapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item)
    bookingDurationSpinner.adapter = durationAdapter

    bookingDurationSpinner.onItemSelectedListener =
        object : AdapterView.OnItemSelectedListener {
            override fun onItemSelected(
                parent: AdapterView<*>?, view: View?, position: Int, id: Long
            ) {
                selectedDuration =
                    parent?.getItemAtPosition(position).toString()
                outputBookingDuration.text = "Booking Duration : $selectedDuration hours"
                if (selectedStartHour.isNotEmpty()) {
                    calculateEndHour(selectedDuration.substringBefore(":").toInt())
                }
            }

            override fun onNothingSelected(parent: AdapterView<*>?) {
            }
        }

    startHourEditText.setOnClickListener {
        if (selectedDate.isEmpty()) {
            Toast.makeText(this, "Please select date first",
            Toast.LENGTH_SHORT).show()
        } else {
    
```

```

        showStartTimePickerDialog()
    }
}

buttonBookNanny.setOnClickListener {
    if (validateInputs()) {
        db.collection("BookingDaily").whereEqualTo("nannyID", nannyId)
            .whereEqualTo("bookDate", selectedDate).get()
            .addOnSuccessListener { documents ->
                if (!documents.isEmpty) {
                    showBookingExistsDialog()
                } else {
                    saveBookingToFirestore()
                }
            }.addOnFailureListener { e ->
                Log.w(TAG, "Error getting documents", e)
                Toast.makeText(
                    this, "Failed to check existing bookings",
                    Toast.LENGTH_SHORT
                ).show()
            }
    }
}

private fun showDatePickerDialog() {
    val calendar = Calendar.getInstance()
    val datePickerDialog = DatePickerDialog(
        this,
        { _, year, month, dayOfMonth ->
            val selectedCalendar = Calendar.getInstance()
            selectedCalendar.set(year, month, dayOfMonth)
            val sdf = SimpleDateFormat("dd/MM/yyyy", Locale.getDefault())
            selectedDate = sdf.format(selectedCalendar.time)
            outputBookingDate.text = "Booking Date           : $selectedDate"
            bookingDateEditText.setText(selectedDate)
            resetOutputTimes()

            db.collection("BookingDaily")
                .whereEqualTo("nannyID", nannyId)
                .whereEqualTo("bookDate", selectedDate)
                .get()
                .addOnSuccessListener { documents ->
                    if (!documents.isEmpty) {
                        showBookingExistsDialog()
                    }
                }.addOnFailureListener { e ->
                    Log.w(TAG, "Error getting documents", e)
                    Toast.makeText(this, "Failed to check existing bookings",
                    Toast.LENGTH_SHORT).show()
                }
        },
        calendar.get(Calendar.YEAR),
        calendar.get(Calendar.MONTH),
        calendar.get(Calendar.DAY_OF_MONTH)
    )
    calendar.add(Calendar.DAY_OF_MONTH, 1)
    datePickerDialog.datePicker.minDate = calendar.timeInMillis
    datePickerDialog.show()
}

private fun showStartTimePickerDialog() {
    val calendar = Calendar.getInstance()
    val hour = calendar.get(Calendar.HOUR_OF_DAY)
    val minute = calendar.get(Calendar.MINUTE)
}

```

```

        val timePickerDialog = TimePickerDialog(
            this, { _, selectedHour, selectedMinute ->
                val selectedTime =
                    String.format(Locale.getDefault(), "%02d:%02d", selectedHour,
selectedMinute)
                    val endTimeCalendar = Calendar.getInstance()
                    endTimeCalendar.set(Calendar.HOUR_OF_DAY, selectedHour)
                    endTimeCalendar.set(Calendar.MINUTE, selectedMinute)
                    endTimeCalendar.add(
                        Calendar.HOUR_OF_DAY, selectedDuration.substringBefore(
                    "")).toInt()
                )
                val endTime = endTimeCalendar.get(Calendar.HOUR_OF_DAY)

                if (selectedHour < 5 || selectedHour > 18 || (selectedHour == 18
&& selectedMinute > 0)) {
                    val builder = AlertDialog.Builder(this)
                    builder.setTitle("Invalid Time")
                    builder.setMessage("You cannot book at this time as the end
time will exceed the working hours limit of 8:00 PM. Please select an earlier
start time.")
                    builder.setPositiveButton("OK") { dialog, _ ->
                        dialog.dismiss()
                        showStartTimePickerDialog()
                    }
                    builder.show()
                } else {
                    selectedStartHour = selectedTime
                    startHourEditText.setText(selectedTime)
                    outputStartHour.text = "Start Time
":
$selectedStartHour"
                    calculateEndHour(selectedDuration.substringBefore(
                    "")).toInt())
                }
            }, hour, minute, true
        )

        timePickerDialog.setTitle("Select Start Time")
        timePickerDialog.show()
    }

    private fun calculateEndHour(durationHours: Int) {
        val format = SimpleDateFormat("HH:mm", Locale.getDefault())
        val calendar = Calendar.getInstance()

        try {
            calendar.time = format.parse(selectedStartHour) ?: Date()
            calendar.add(Calendar.HOUR_OF_DAY, durationHours)
            val endTime = calendar.get(Calendar.HOUR_OF_DAY)
            val endMinute = calendar.get(Calendar.MINUTE)

            if (endTime >= 20 && endMinute > 0) {
                val builder = AlertDialog.Builder(this)
                builder.setTitle("Invalid Time")
                builder.setMessage("You cannot book at this time as the end time
will exceed the working hours limit of 20:00. Please select an earlier start
time.")
                builder.setPositiveButton("OK") { dialog, _ ->
                    dialog.dismiss()
                    selectedStartHour = ""
                    selectedEndHour = ""
                    startHourEditText.setText("")
                    outputStartHour.text = "Start Time
":
outputEndHour.text = "End Time
"
                }
                builder.show()
            }
        }
    }
}

```

```

        } else {
            selectedEndHour = format.format(calendar.time)
            outputEndHour.text = "End Time"
        }
    }

    val nannyRef = db.collection("Nanny").document(nannyId)
    nannyRef.get().addOnSuccessListener { document ->
        if (document != null && document.exists()) {
            val nanny = document.toObject(Nanny::class.java)
            nanny?.let { nannyData ->
                val totalPricing = nannyData.pricing * durationHours
                val currencyFormat =
                    NumberFormat.getCurrencyInstance(Locale("in", "ID"))
                val formattedTotalPricing =
                    currencyFormat.format(totalPricing)

                totalPricingText.text = "Total Pricing"
                $formattedTotalPricing
            }
        }
    }.addOnFailureListener { e ->
        Log.d(TAG, "Error calculating total pricing", e)
        Toast.makeText(this, "Error calculating total pricing",
        Toast.LENGTH_SHORT).show()
    }
}

catch (e: ParseException) {
    e.printStackTrace()
    Toast.makeText(this, "Error calculating end hour",
    Toast.LENGTH_SHORT).show()
}
}

private fun validateInputs(): Boolean {
    if (selectedDate.isEmpty()) {
        Toast.makeText(this, "Please select date", Toast.LENGTH_SHORT).show()
        return false
    }
    if (selectedDuration.isEmpty()) {
        Toast.makeText(this, "Please select booking duration",
        Toast.LENGTH_SHORT).show()
        return false
    }
    if (selectedStartHour.isEmpty()) {
        Toast.makeText(this, "Please select start hour",
        Toast.LENGTH_SHORT).show()
        return false
    }
    return true
}

private fun saveBookingToFirestore() {
    val user = auth.currentUser
    val userId = user?.uid ?: ""

    val nannyRef = db.collection("Nanny").document(nannyId)
    nannyRef.get().addOnSuccessListener { document ->
        if (document != null && document.exists()) {
            val nanny = document.toObject(Nanny::class.java)
            nanny?.let { nannyData ->
                val durationHours = selectedDuration.substringBefore(
                "").toInt()
                val totalPricing = nannyData.pricing * durationHours
                val booking = hashMapOf(
                    "userID" to userId,
                    "nannyID" to nannyId,
                )
            }
        }
    }
}

```

```

        "bookDate" to selectedDate,
        "bookHours" to selectedStartHour,
        "endHours" to selectedEndHour,
        "bookDuration" to selectedDuration,
        "totalPricing" to totalPricing
    )

    db.collection("BookingDaily")
        .whereEqualTo("nannyID", nannyId)
        .whereEqualTo("bookDate", selectedDate)
        .get()
        .addOnSuccessListener { documents ->
            if (!documents.isEmpty) {
                showBookingExistsDialog()
            } else {
                db.collection("BookingDaily").add(booking)
                    .addOnSuccessListener { documentReference ->
                        Log.d(TAG, "DocumentSnapshot added with ID: ${documentReference.id}")
                        nannyRef.update("bookID",
                            documentReference.id)
                            .addOnSuccessListener {
                                Log.d(TAG, "bookID updated successfully")
                            }
                            .addOnFailureListener { e ->
                                Log.w(TAG, "Error updating bookID", e)
                            }
                        }
                    }
            }

            val userRef =
            db.collection("User").document(userId)
                userRef.update(
                    "bookIDs",
                    FieldValue.arrayUnion(documentReference.id)
                ).addOnSuccessListener {
                    Log.d(TAG, "bookID added to user document successfully")
                    showSuccessDialog()
                }.addOnFailureListener { e ->
                    Log.w(TAG, "Error updating user document with bookID", e)
                }
            }

            .addOnFailureListener { e ->
                Log.w(TAG, "Error adding document", e)
                Toast.makeText(this, "Failed to book nanny", Toast.LENGTH_SHORT).show()
            }
        }
    }

    .addOnFailureListener { e ->
        Log.w(TAG, "Error getting documents", e)
        Toast.makeText(this, "Failed to check existing bookings", Toast.LENGTH_SHORT).show()
    }
}

} else {
    Log.d(TAG, "No such document")
}

}.addOnFailureListener { exception ->
    Log.d(TAG, "get failed with ", exception)
}
}

private fun showBookingExistsDialog() {
    val builder = AlertDialog.Builder(this)

```

```

        builder.setTitle("Nanny Not Available")
        builder.setMessage("This nanny is already booked on the selected date.
Please choose another date.")
        builder.setPositiveButton("OK") { dialogInterface: DialogInterface, i: Int
->
        dialogInterface.dismiss()
    }
    builder.show()
}

private fun resetOutputTimes() {
    outputStartHour.text = "Start Hour" :"
    outputEndHour.text = "End Time" :"
    selectedStartHour = ""
    selectedEndHour = ""
}

private fun showSuccessDialog() {
    val builder = AlertDialog.Builder(this)
    builder.setTitle("Booking Success")
    builder.setMessage("Your booking has been successfully made.")
    builder.setPositiveButton("OK") { dialogInterface: DialogInterface, i: Int
->
        val intent = Intent(this, HistoryActivity::class.java)
        startActivity(intent)
        dialogInterface.dismiss()
    }
    builder.show()
}

private fun loadNannyDataFromFirestore(nannyId: String) {
    val docRef = db.collection("Nanny").document(nannyId)

    docRef.get().addOnSuccessListener { document ->
        if (document != null && document.exists()) {
            val nanny = document.toObject(Nanny::class.java)
            nanny?.let { nannyData ->
                displayNannyInformation(nannyData)
            }
        } else {
            Log.d(TAG, "No such document")
        }
    }.addOnFailureListener { exception ->
        Log.d(TAG, "get failed with ", exception)
    }
}

private fun displayNannyInformation(nanny: Nanny) {
    val nameTextView = findViewById<TextView>(R.id.text_name)
    val typeTextView = findViewById<TextView>(R.id.text_email)
    val profileImageView = findViewById<ImageView>(R.id.image_profile)

    nameTextView.text = nanny.name
    typeTextView.text = nanny.type

    Glide.with(this).load(nanny.pict).placeholder(R.drawable.nanny_placeholder)
        .error(R.drawable.placeholder_image).into(profileImageView)
}

companion object {
    private const val TAG = "BookDailyActivity"
}
}

```

[File activity BookMonthlyActivity.kt]

```
package com.app.famcare.view.booking
```

```

import android.app.AlertDialog
import android.app.DatePickerDialog
import android.app.ProgressDialog
import android.content.DialogInterface
import android.content.Intent
import android.os.Bundle
import android.util.Log
import android.view.MenuItem
import android.widget.*
import androidx.appcompat.app.AppCompatActivity
import androidx.appcompat.widget.Toolbar
import com.app.famcare.R
import com.app.famcare.model.Nanny
import com.app.famcare.view.history.HistoryActivity
import com.bumptech.glide.Glide
import com.google.firebase.auth.FirebaseAuth
import com.google.firebase.firestore.FieldValue
import com.google.firebase.firestore.FirebaseFirestore
import java.text.NumberFormat
import java.text.SimpleDateFormat
import java.util.*

class BookMonthlyActivity : AppCompatActivity() {
    private lateinit var startDateEditText: EditText
    private lateinit var outputBookingDuration: TextView
    private lateinit var outputStartDate: TextView
    private lateinit var outputEndDate: TextView
    private lateinit var textCost: TextView
    private lateinit var nannyId: String
    private lateinit var auth: FirebaseAuth
    private var selectedBookingDuration: Int = 0
    private var endDate: String = ""
    private lateinit var db: FirebaseFirestore

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

        val toolbar = findViewById<Toolbar>(R.id.toolbar)
        setSupportActionBar(toolbar)
        supportActionBar?.setDisplayHomeAsUpEnabled(true)
        supportActionBar?.title = ""

        auth = FirebaseAuth.getInstance()
        db = FirebaseFirestore.getInstance()

        nannyId = intent.getStringExtra("nannyId") ?: ""

        loadNannyDataFromFirestore(nannyId)

        val bookingDurationSpinner =
            findViewById<Spinner>(R.id.bookingDurationSpinner)
        val adapter = ArrayAdapter.createFromResource(
            this, R.array.booking_duration, android.R.layout.simple_spinner_item
        )

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

        startDateEditText = findViewById(R.id.startDateEditText)
        outputBookingDuration = findViewById(R.id.outputBookingDuration)
        outputStartDate = findViewById(R.id.outputStartDate)
        outputEndDate = findViewById(R.id.outputEndDate)

        startDateEditText.setOnClickListener {
            showDatePickerDialog()
        }
    }

    private fun loadNannyDataFromFirestore(nannyId: String) {
        db.collection("nannies")
            .document(nannyId)
            .get()
            .addOnSuccessListener { documentSnapshot ->
                if (documentSnapshot.exists()) {
                    val nannyData = documentSnapshot.toObject(Nanny::class.java)
                    nannyId = nannyData?.id ?: ""
                    textCost.text = nannyData?.cost?.toString() ?: ""
                } else {
                    Log.e("BookMonthlyActivity", "Nanny document does not exist")
                }
            }
            .addOnFailureListener { exception ->
                Log.e("BookMonthlyActivity", "Error getting document", exception)
            }
    }

    private fun showDatePickerDialog() {
        val datePickerDialog = DatePickerDialog(this, R.style.AppTheme_Dark, { _, year, month, dayOfMonth ->
            val date = "$dayOfMonth-$month-$year"
            outputStartDate.text = date
            outputEndDate.text = date
        }, 0, 0, 0)
        datePickerDialog.show()
    }
}

```

```

        }

        textCost = findViewById(R.id.textCost)
        textCost.text = "Total Cost : " + formatCurrency(0)
        outputBookingDuration = findViewById(R.id.outputBookingDuration)
        outputBookingDuration.text = ""

        bookingDurationSpinner.onItemSelectedListener =
            object : AdapterView.OnItemSelectedListener {
                override fun onItemSelected(
                    parent: AdapterView<*>?, view: android.view.View?, position: Int, id: Long
                ) {
                    selectedBookingDuration =
                        parent?.getItemAtPosition(position).toString().toInt()
                    outputBookingDuration.text = "Booking Duration : $selectedBookingDuration"

                    if (outputStartDate.text.isNotEmpty()) {
                        calculateEndDate()
                        updateTotalPricing()
                    }
                }

                override fun onNothingSelected(parent: AdapterView<*>?) {
                    outputBookingDuration.text = ""
                }
            }
    }

    val buttonBookNanny = findViewById<Button>(R.id.buttonBookNanny)
    buttonBookNanny.setOnClickListener {
        bookNanny()
    }
}

override fun onOptionsItemSelected(item: MenuItem): Boolean {
    return when (item.itemId) {
        android.R.id.home -> {
            onBackPressed()
            true
        }

        else -> super.onOptionsItemSelected(item)
    }
}

private fun showDatePickerDialog() {
    val calendar = Calendar.getInstance()

    fetchBookedDates { bookedDatesRanges ->
        val datePickerDialog = DatePickerDialog(
            this,
            { _, year, month, dayOfMonth ->
                val selectedCalendar = Calendar.getInstance()
                selectedCalendar.set(year, month, dayOfMonth)

                val sdf = SimpleDateFormat("dd/MM/yyyy", Locale.getDefault())
                val selectedDate = sdf.format(selectedCalendar.time)

                if (isDateBooked(selectedCalendar.time, bookedDatesRanges)) {
                    showDateAlreadyBookedDialog()
                } else {
                    outputStartDate.text = "Start Date : $selectedDate"
                    startDateEditText.setText(selectedDate)
                    calculateEndDate()
                }
            }
        )
    }
}

```

```

        updateTotalPricing()
    }
},
calendar.get(Calendar.YEAR),
calendar.get(Calendar.MONTH),
calendar.get(Calendar.DAY_OF_MONTH)
)

calendar.add(Calendar.DAY_OF_MONTH, 1)
datePickerDialog.datePicker.minDate = calendar.timeInMillis

datePickerDialog.show()
}
}

private fun showDateAlreadyBookedDialog() {
    val builder = AlertDialog.Builder(this)
    builder.setTitle("Date Already Booked")
    .setMessage("This date is already booked. Please select another date.")
    .setPositiveButton("OK") { dialog, which ->
    }.show()
}

private fun fetchBookedDates(callback: (List<Pair<Date, Date>>) -> Unit) {
    val db = FirebaseFirestore.getInstance()
    val bookingsRef = db.collection("BookingMonthly")

    bookingsRef.whereEqualTo("nannyID", nannyId).get().addOnSuccessListener {
documents ->
        val sdf = SimpleDateFormat("dd/MM/yyyy", Locale.getDefault())
        val bookedDatesRanges = mutableListOf<Pair<Date, Date>>()
        for (document in documents) {
            val existingStartDate = document.getString("startDate") ?: ""
            val existingEndDate = document.getString("endDate") ?: ""
            val existingStartDateDate = sdf.parse(existingStartDate)
            val existingEndDateDate = sdf.parse(existingEndDate)
            bookedDatesRanges.add(existingStartDateDate to
existingEndDateDate)
        }
        callback(bookedDatesRanges)
    }.addOnFailureListener { exception ->
        Log.w(TAG, "Error getting documents: ", exception)
        callback(emptyList())
    }
}

private fun isDateBooked(date: Date, bookedDatesRanges: List<Pair<Date, Date>>): Boolean {
    for ((startDate, endDate) in bookedDatesRanges) {
        if (!date.before(startDate) && !date.after(endDate)) {
            return true
        }
    }
    return false
}

private fun checkForExistingBookings(
    startDate: String, endDate: String, callback: (Boolean) -> Unit
) {
    val db = FirebaseFirestore.getInstance()
    val sdf = SimpleDateFormat("dd/MM/yyyy", Locale.getDefault())

    try {
        val newStartDate = sdf.parse(startDate)
        val newEndDate = sdf.parse(endDate)

```

```

        val bookingsRef = db.collection("BookingMonthly")
        bookingsRef.whereEqualTo("nannyID",
nannyId).get().addOnSuccessListener { documents ->
            var isAvailable = true
            for (document in documents) {
                val existingStartDateStr = document.getString("startDate") ?:
"""
                val existingEndDateStr = document.getString("endDate") ?: ""
                val existingStartDate = sdf.parse(existingStartDateStr)
                val existingEndDate = sdf.parse(existingEndDateStr)

                if (newEndDate != existingStartDate &&
!newStartDate.equals(existingEndDate) && newStartDate.before(
                    existingEndDate
                ) && newEndDate.after(existingStartDate)
                ) {
                    isAvailable = false
                    break
                }
            }
            callback(isAvailable)
        }.addOnFailureListener { exception ->
            Log.w(TAG, "Error getting documents: ", exception)
            callback(false)
        }
    } catch (e: Exception) {
        Log.e(TAG, "Error parsing dates", e)
        callback(false)
    }
}

private fun bookNanny() {
    val selectedStartDate = startDateEditText.text.toString()
    val selectedEndDate = endDate

    checkForExistingBookings(selectedStartDate, selectedEndDate) { isAvailable
->
        if (isAvailable) {
            val progressDialog = ProgressDialog(this)
            progressDialog.setMessage("Booking nanny...")
            progressDialog.show()

            val user = auth.currentUser
            val userId = user?.uid ?: ""

            val nannyRef = db.collection("Nanny").document(nannyId)
            nannyRef.get().addOnSuccessListener { document ->
                if (document != null && document.exists()) {
                    val nanny = document.toObject(Nanny::class.java)

                    val nannyPricing = nanny?.pricing ?: 0
                    val totalPricing = nannyPricing * selectedBookingDuration

                    val booking = hashMapOf(
                        "userID" to userId,
                        "nannyID" to nannyId,
                        "bookingDuration" to
selectedBookingDuration.toString(),
                        "startDate" to selectedStartDate,
                        "endDate" to selectedEndDate,
                        "totalPricing" to totalPricing
                    )

                    db.collection("BookingMonthly").add(booking)
                        .addOnSuccessListener { documentReference ->

```

```

        Log.d(TAG, "DocumentSnapshot added with ID:  

        ${documentReference.id}")

        nannyRef.update("bookID", documentReference.id)
            .addOnSuccessListener {
                Log.d(TAG, "bookID updated successfully")
            }.addOnFailureListener { e ->
                Log.w(TAG, "Error updating bookID", e)
            }

        val userRef =
db.collection("User").document(userId)
            userRef.update(
                "bookIDs",
                FieldValue.arrayUnion(documentReference.id)
            ).addOnSuccessListener {
                Log.d(TAG, "bookID added to user document  

successfully")
            }.addOnFailureListener { e ->
                Log.w(TAG, "Error updating user document with  

bookID", e)
            }

        progressDialog.dismiss()
        showSuccessDialog()
    }.addOnFailureListener { e ->
        Log.w(TAG, "Error adding document", e)
        progressDialog.dismiss()
        Toast.makeText(this, "Failed to book nanny",  

Toast.LENGTH_SHORT).show()
    }
} else {
    Log.d(TAG, "No such document")
    progressDialog.dismiss()
    Toast.makeText(this, "Failed to get nanny data",  

Toast.LENGTH_SHORT).show()
}
}.addOnFailureListener { exception ->
Log.d(TAG, "get failed with ", exception)
progressDialog.dismiss()
Toast.makeText(this, "Failed to get nanny data",  

Toast.LENGTH_SHORT).show()
}
} else {
    Toast.makeText(
        this,
        "The selected date range is already booked for this nanny.",  

        Toast.LENGTH_SHORT
    ).show()
}
}

private fun calculateEndDate() {
    val sdf = SimpleDateFormat("dd/MM/yyyy", Locale.US)
    val startDateString = startDateEditText.text.toString()
    val calendar = Calendar.getInstance()

    try {
        val startDate = sdf.parse(startDateString)
        calendar.time = startDate
        calendar.add(Calendar.DAY_OF_MONTH, 30 * selectedBookingDuration)
        endDate = sdf.format(calendar.time)
        outputEndDate.text = "End Date" : $endDate"
    } catch (e: Exception) {
        e.printStackTrace()
    }
}

```

```

    }

    private fun showSuccessDialog() {
        val builder = AlertDialog.Builder(this)
        builder.setTitle("Booking Success")
        builder.setMessage("Your booking has been successfully made.")
        builder.setPositiveButton("OK") { dialogInterface: DialogInterface, i: Int
            ->
            val intent = Intent(this, HistoryActivity::class.java)
            intent.putExtra(
                "selectedTab", 1
            )
            startActivity(intent)
            dialogInterface.dismiss()
        }
        builder.show()
    }

    private fun loadNannyDataFromFirestore(nannyId: String) {
        val db = Firebase Firestore.getInstance()
        val docRef = db.collection("Nanny").document(nannyId)

        docRef.get().addOnSuccessListener { document ->
            if (document != null && document.exists()) {
                val nanny = document.toObject(Nanny::class.java)
                nanny?.let { nannyData ->
                    displayNannyInformation(nannyData)
                }
            } else {
                Log.d(TAG, "No such document")
            }
        }.addOnFailureListener { exception ->
            Log.d(TAG, "get failed with ", exception)
        }
    }

    private fun updateTotalPricing() {
        val nannyRef = db.collection("Nanny").document(nannyId)
        nannyRef.get().addOnSuccessListener { document ->
            if (document != null && document.exists()) {
                val nanny = document.toObject(Nanny::class.java)
                val nannyPricing = nanny?.pricing ?: 0
                val totalPricing = nannyPricing * selectedBookingDuration
                textCost.text = "Total Cost : " +
formatCurrency(totalPricing)
            } else {
                Log.d(TAG, "No such document")
            }
        }.addOnFailureListener { exception ->
            Log.d(TAG, "get failed with ", exception)
        }
    }

    private fun displayNannyInformation(nanny: Nanny) {
        val nameTextView = findViewById<TextView>(R.id.text_name)
        val typeTextView = findViewById<TextView>(R.id.text_email)
        val profileImageView = findViewById<ImageView>(R.id.image_profile)

        nameTextView.text = nanny.name
        typeTextView.text = nanny.type

        Glide.with(this).load(nanny.pict).placeholder(R.drawable.nanny_placeholder)
            .error(R.drawable.placeholder_image).into(profileImageView)
    }

    companion object {
        private const val TAG = "BookMonthlyActivity"
    }
}

```

```

    }

    private fun formatCurrency(value: Int): String {
        val format = NumberFormat.getCurrencyInstance(Locale("in", "ID"))
        return format.format(value)
    }
}

```

[File activity HistoryActivity.kt]

```

package com.app.famcare.view.history

import android.content.Intent
import android.os.Bundle
import androidx.appcompat.app.AppCompatActivity
import androidx.appcompat.widget.Toolbar
import androidx.viewpager2.widget.ViewPager2
import com.app.famcare.R
import com.app.famcare.adapter.HistoryPagerAdapter
import com.app.famcare.view.facilities.FacilitiesActivity
import com.app.famcare.view.historyimport.HistoryBDFragment
import com.app.famcare.view.historyimport.HistoryBMFragment
import com.app.famcare.view.main.MainActivity
import com.app.famcare.view.profile.ProfileActivity
import com.google.android.material.bottomnavigation.BottomNavigationView
import com.google.android.material.tabs.TabLayout
import com.google.android.material.tabs.TabLayoutMediator

class HistoryActivity : AppCompatActivity() {

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

        val toolbar = findViewById<Toolbar>(R.id.toolbar)
        setSupportActionBar(toolbar)
        supportActionBar?.title = ""

        val tabLayout = findViewById<TabLayout>(R.id.tab_layout)
        val viewPager = findViewById<ViewPager2>(R.id.view_pager)

        val adapter = HistoryPagerAdapter(this)
        viewPager.adapter = adapter

        TabLayoutMediator(tabLayout, viewPager) { tab, position ->
            tab.text = when (position) {
                0 -> "Daily"
                1 -> "Monthly"
                else -> "Undefined"
            }
        }.attach()

        viewPager.registerOnPageChangeCallback(object :
Viewpager2.OnPageChangeCallback() {
            override fun onPageSelected(position: Int) {
                super.onPageSelected(position)
                when (position) {
                    0 -> {
                        val fragment = HistoryBDFragment()
                        supportFragmentManager.beginTransaction()
                            .replace(R.id.fragment_container, fragment).commit()
                    }
                    1 -> {
                        val fragment = HistoryBMFragment()
                        supportFragmentManager.beginTransaction()
                            .replace(R.id.fragment_container, fragment).commit()
                    }
                }
            }
        })
    }
}

```

```

        }
    }
}

val selectedTab = intent.getIntExtra("selectedTab", 0)

if (selectedTab == 1) {
    viewPager.currentItem = 1
} else {
    viewPager.currentItem = 0
}

val bottomNavigation: BottomNavigationView =
findViewById(R.id.bottom_navigation)
bottomNavigation.selectedItemId = R.id.page_2

bottomNavigation.setOnNavigationItemSelected { item ->
    when (item.itemId) {
        R.id.page_1 -> {
            val intent = Intent(this, MainActivity::class.java)
            startActivity(intent)
            true
        }

        R.id.page_2 -> true
        R.id.page_3 -> {
            val intent = Intent(this, FacilitiesActivity::class.java)
            startActivity(intent)
            true
        }

        R.id.page_4 -> {
            val intent = Intent(this, ProfileActivity::class.java)
            startActivity(intent)
            true
        }

        else -> false
    }
}
}
}

```

[File fragment HistoryBDFragment.kt]

```

package com.app.famcare.view.historyimport

import android.app.AlertDialog
import android.content.Intent
import android.os.Bundle
import android.util.Log
import android.view.LayoutInflater
import android.view.MenuItem
import android.view.View
import android.view.ViewGroup
import android.widget.Toast
import androidx.fragment.app.Fragment
import androidx.recyclerview.widget.LinearLayoutManager
import androidx.recyclerview.widget.RecyclerView
import com.app.famcare.R
import com.app.famcare.adapter.BookingAdapter
import com.app.famcare.model.BookingDaily
import com.app.famcare.model.BookingType
import com.app.famcare.view.chat.ChatActivity

```

```

import com.app.famcare.view.detailhistory.DetailHistoryBDActivity
import com.google.firebase.auth.FirebaseAuth
import com.google.firebaseio.FirebaseFirestore

class HistoryBDFragment : Fragment(), BookingAdapter.OnItemClickListener {

    private lateinit var recyclerView: RecyclerView
    private lateinit var adapter: BookingAdapter
    private var selectedBookingID: String = ""

    override fun onCreateView(
        inflater: LayoutInflater, container: ViewGroup?, savedInstanceState: Bundle?
    ): View? {
        val rootView = inflater.inflate(R.layout.fragment_history_b_d, container, false)

        recyclerView = rootView.findViewById(R.id.recyclerViewHistory)
        recyclerView.layoutManager = LinearLayoutManager(requireContext())

        adapter = BookingAdapter(emptyList(), this)
        recyclerView.adapter = adapter

        fetchDataFromFirestore()

        return rootView
    }

    override fun onOptionsItemSelected(item: MenuItem): Boolean {
        return when (item.itemId) {
            android.R.id.home -> {
                requireActivity().onBackPressed()
                true
            }
            else -> super.onOptionsItemSelected(item)
        }
    }

    private fun fetchDataFromFirestore() {
        val progressDialog = ProgressDialog(requireContext())
        progressDialog.setMessage("Loading data...")
        progressDialog.show()

        val currentUserID = FirebaseAuth.getInstance().currentUser?.uid ?: ""
        val db = FirebaseFirestore.getInstance()

        db.collection("User").document(currentUserID).get().addOnSuccessListener {
            document ->
            if (document != null && document.exists()) {
                val bookIDs = document.get("bookIDs") as? List<String> ?: emptyList()
                if (bookIDs.isNotEmpty()) {
                    progressDialog.dismiss()
                    fetchBookings(bookIDs)
                } else {
                    Log.w(TAG, "No bookings found for user")
                    progressDialog.dismiss()
                }
            } else {
                Log.w(TAG, "User document does not exist")
                progressDialog.dismiss()
            }
        }.addOnFailureListener { exception ->
            Log.w(TAG, "Error getting user document: ", exception)
            progressDialog.dismiss()
        }
    }
}

```

```

    }

    private fun fetchBookings(bookIDs: List<String>) {
        val db = Firebase Firestore.getInstance()
        val bookingList = mutableListOf<BookingDaily>()

        bookIDs.forEach { bookID ->

            db.collection("BookingDaily").document(bookID).get().addOnSuccessListener {
                document ->
                if (document != null && document.exists()) {
                    val nannyID = document.getString("nannyID") ?: ""
                    val bookDate = document.getString("bookDate") ?: ""
                    val bookHours = document.getString("bookHours") ?: ""
                    val endHours = document.getString("endHours") ?: ""
                    db.collection("Nanny").document(nannyID).get()
                        .addOnSuccessListener { nannyDoc ->
                            val nannyName = nannyDoc.getString("name") ?: ""
                            val bookingDaily = BookingDaily(
                                bookID, nannyName, bookDate, bookHours, endHours,
                                BookingType.DAILY
                            )
                            bookingList.add(bookingDaily)

                            adapter.setData(bookingList.reversed())
                        }.addOnFailureListener { e ->
                            Log.w(TAG, "Error getting nanny document", e)
                        }
                }
            }.addOnFailureListener { e ->
                Log.w(TAG, "Error getting booking document: ", e)
            }
        }
    }

    override fun onItemClick(bookingID: String) {
        selectedBookingID = bookingID
        val intent = Intent(requireContext(), DetailHistoryBDActivity::class.java)
        intent.putExtra("bookingID", selectedBookingID)
        startActivity(intent)
    }

    override fun onChatClick(bookingID: String) {
        val db = Firebase Firestore.getInstance()

        db.collection("BookingDaily").document(bookingID).get().addOnSuccessListener {
            document ->
            if (document != null && document.exists()) {
                val nannyID = document.getString("nannyID") ?: ""
                if (nannyID.isNotEmpty()) {
                    val intent = Intent(requireContext(),
                        ChatActivity::class.java)
                    intent.putExtra("nannyID", nannyID)
                    startActivity(intent)
                } else {
                    Toast.makeText(requireContext(), "Nanny ID not found",
                        Toast.LENGTH_SHORT)
                        .show()
                }
            }
        }.addOnFailureListener { exception ->
            Toast.makeText(requireContext(), "Failed to load booking",
                Toast.LENGTH_SHORT).show()
        }
    }

    companion object {

```

```

        private const val TAG = "HistoryBDFragment"
    }
}

```

[File fragment HistoryBMFragment.kt]

```

package com.app.famcare.view.historyimport

import android.app.AlertDialog
import android.content.Intent
import android.os.Bundle
import android.util.Log
import android.view.LayoutInflater
import android.view.MenuItem
import android.view.View
import android.view.ViewGroup
import android.widget.TextView
import android.widget.Toast
import androidx.fragment.app.Fragment
import androidx.recyclerview.widget.LinearLayoutManager
import androidx.recyclerview.widget.RecyclerView
import com.app.famcare.R
import com.app.famcare.adapter.BookingAdapter
import com.app.famcare.model.BookingMonthlyHistory
import com.app.famcare.model.BookingType
import com.app.famcare.view.chat.ChatActivity
import com.app.famcare.view.detailhistory.DetailHistoryBMActivity
import com.google.firebase.auth.FirebaseAuth
import com.google.firebaseio.firebaseio.Firestore

class HistoryBMFragment : Fragment(), BookingAdapter.OnItemClickListener {
    private lateinit var recyclerView: RecyclerView
    private lateinit var adapter: BookingAdapter
    private lateinit var emptyTextView: TextView
    private var selectedBookingID: String = ""

    override fun onCreateView(
        inflater: LayoutInflater, container: ViewGroup?, savedInstanceState: Bundle?
    ): View? {
        val rootView = inflater.inflate(R.layout.fragment_history_bm, container, false)

        recyclerView = rootView.findViewById(R.id.recyclerViewHistoryBM)
        recyclerView.layoutManager = LinearLayoutManager(requireContext())

        adapter = BookingAdapter(emptyList(), this)
        recyclerView.adapter = adapter

        emptyTextView = rootView.findViewById(R.id.emptyTextView)

        return rootView
    }

    override fun onViewCreated(view: View, savedInstanceState: Bundle?) {
        super.onViewCreated(view, savedInstanceState)
        fetchDataFromFirestore()
    }

    override fun onOptionsItemSelected(item: MenuItem): Boolean {
        return when (item.itemId) {
            android.R.id.home -> {
                requireActivity().onBackPressed()
                true
            }
            else -> super.onOptionsItemSelected(item)
        }
    }
}

```

```

        }

    }

    private fun fetchDataFromFirestore() {
        val progressDialog = ProgressDialog(requireContext())
        progressDialog.setMessage("Loading data...")
        progressDialog.show()

        val currentUserID = FirebaseAuth.getInstance().currentUser?.uid ?: ""
        val db = FirebaseFirestore.getInstance()

        db.collection("User").document(currentUserID).get().addOnSuccessListener {
            document ->
            if (document != null && document.exists()) {
                val bookIDs = document.get("bookIDs") as? List<String> ?: emptyList()
                if (bookIDs.isNotEmpty()) {
                    progressDialog.dismiss()
                    fetchMonthlyBookings(bookIDs)
                } else {
                    emptyTextView.visibility = View.VISIBLE
                    recyclerView.visibility = View.GONE
                    Log.w(TAG, "No bookings found for user")
                    progressDialog.dismiss()
                }
            } else {
                Log.w(TAG, "User document does not exist")
                progressDialog.dismiss()
            }
        }.addOnFailureListener { exception ->
            Log.w(TAG, "Error getting user document: ", exception)
            progressDialog.dismiss()
        }
    }

    private fun fetchMonthlyBookings(bookIDs: List<String>) {
        val db = FirebaseFirestore.getInstance()
        val bookingList = mutableListOf<BookingMonthlyHistory>()
        var fetchedBookings = 0

        bookIDs.forEach { bookID ->
            db.collection("BookingMonthly").document(bookID).get()
                .addOnSuccessListener { document ->
                    fetchedBookings++
                    if (document != null && document.exists()) {
                        val nannyID = document.getString("nannyID") ?: ""
                        val bookStartDate = document.getString("startDate") ?: ""
                        val bookEndDate = document.getString("endDate") ?: ""
                        val totalCost = document.getString("totalCost") ?: ""
                        db.collection("Nanny").document(nannyID).get()
                            .addOnSuccessListener { nannyDoc ->
                                val nannyName = nannyDoc.getString("name") ?: ""
                                val bookingMonthlyHistory = BookingMonthlyHistory(
                                    bookID,
                                    nannyName,
                                    bookStartDate,
                                    bookEndDate,
                                    BookingType.MONTHLY,
                                    nannyID,
                                    totalCost
                                )
                                bookingList.add(bookingMonthlyHistory)
                            }
                    }
                }
        }

        if (fetchedBookings == bookIDs.size) {
            adapter.setData(bookingList.reversed())
            emptyTextView.visibility = View.GONE
        }
    }
}

```

```

                recyclerView.visibility = View.VISIBLE
            }
        }.addOnFailureListener { e ->
            Log.w(TAG, "Error getting nanny document", e)
        }
    } else {
        if (fetchedBookings == bookIDs.size) {
            if (bookingList.isEmpty()) {
                emptyTextView.visibility = View.VISIBLE
                recyclerView.visibility = View.GONE
            } else {
                adapter.setData(bookingList.reversed())
                emptyTextView.visibility = View.GONE
                recyclerView.visibility = View.VISIBLE
            }
        }
    }
}.addOnFailureListener { e ->
    Log.w(TAG, "Error getting booking document: ", e)
    fetchedBookings++
    if (fetchedBookings == bookIDs.size) {
        if (bookingList.isEmpty()) {
            emptyTextView.visibility = View.VISIBLE
            recyclerView.visibility = View.GONE
        } else {
            adapter.setData(bookingList.reversed())
            emptyTextView.visibility = View.GONE
            recyclerView.visibility = View.VISIBLE
        }
    }
}

if (bookIDs.isEmpty()) {
    emptyTextView.visibility = View.VISIBLE
    recyclerView.visibility = View.GONE
}
}

override fun onItemClick(bookingID: String) {
    selectedBookingID = bookingID
    val intent = Intent(requireContext(), DetailHistoryBMAActivity::class.java)
    intent.putExtra("bookingID", selectedBookingID)
    startActivity(intent)
}

override fun onChatClick(bookingID: String) {
    val db = Firebase Firestore.getInstance()

    db.collection("BookingMonthly").document(bookingID).get().addOnSuccessListener {
        document ->
        if (document != null && document.exists()) {
            val nannyID = document.getString("nannyID") ?: ""
            if (nannyID.isNotEmpty()) {
                val intent = Intent(requireContext(),
                    ChatActivity::class.java)
                intent.putExtra("nannyID", nannyID)
                startActivity(intent)
            } else {
                Toast.makeText(requireContext(), "Nanny ID not found",
                    Toast.LENGTH_SHORT)
                    .show()
            }
        }
    }.addOnFailureListener { exception ->
        Toast.makeText(requireContext(), "Failed to load booking",
            Toast.LENGTH_SHORT).show()
    }
}

```

```

        }
    }

    companion object {
        private const val TAG = "HistoryBMFragment"
    }
}

```

[File activity ChatListActivity.kt]

```

package com.app.famcare.view.chat

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import com.app.famcare.R

class ChatListActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_chat_list)
    }
}

```

[File activity ChatActivity.kt]

```

package com.app.famcare.view.chat

import android.os.Bundle
import android.util.Log
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity
import androidx.recyclerview.widget.LinearLayoutManager
import com.app.famcare.databinding.ActivityChatBinding
import com.bumptech.glide.Glide
import com.google.firebase.auth.FirebaseAuth
import com.google.firebaseio.database.*
import com.google.firebaseio.firestore.FirebaseFirestore
import kotlinx.coroutines.Dispatchers
import kotlinx.coroutines.GlobalScope
import kotlinx.coroutines.launch
import kotlinx.coroutines.tasks.await
import java.text.SimpleDateFormat
import java.util.Date
import java.util.Locale
import java.util.TimeZone

class ChatActivity : AppCompatActivity() {

    private lateinit var binding: ActivityChatBinding
    private lateinit var auth: FirebaseAuth
    private lateinit var firestore: FirebaseFirestore
    private lateinit var adapter: ChatAdapter
    private lateinit var messageReference: DatabaseReference
    private var currentUserName: String? = null
    private var currentUserPhotoUrl: String? = null
    private var currentUserId: String? = null
    private var nannyId: String? = null

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityChatBinding.inflate(layoutInflater)
        setContentView(binding.root)

        val database =
            FirebaseDatabase.getInstance("https://famcare-mobprog-default-rtdb.firebaseio.com")
    }
}

```

```

        auth = FirebaseAuth.getInstance()
        firestore = FirebaseFirestore.getInstance()

        messageReference = database.getReference("messages")

        adapter = ChatAdapter()
        binding.messageRecyclerView.adapter = adapter
        binding.messageRecyclerView.layoutManager = LinearLayoutManager(this)

        binding.sendButton.setOnClickListener {
            sendMessage()
        }

        loadCurrentUserNameAndPhotoUrl()

        loadMessages()

        nannyId = intent.getStringExtra("nannyID")
        if (nannyId != null) {
            loadNannyDetails(nannyId!!)
        } else {
            Toast.makeText(this, "Nanny ID is missing", Toast.LENGTH_SHORT).show()
            finish()
        }

        binding.ivBack.setOnClickListener {
            onBackPressed()
        }
    }

    private fun loadNannyDetails(nannyID: String) {
        GlobalScope.launch(Dispatchers.Main) {
            try {
                val document =
                    firestore.collection("Nanny").document(nannyID).get().await()
                val nannyName = document.getString("name")
                val nannyPhotoUrl = document.getString("pict")

                if (nannyName != null) {
                    binding.tvNannyName.text = nannyName
                } else {
                    binding.tvNannyName.text = "Nanny"
                }

                if (nannyPhotoUrl != null) {
                    Glide.with(this@ChatActivity).load(nannyPhotoUrl).circleCrop()
                        .into(binding.ivNannyPhoto)
                }
            } catch (e: Exception) {
                Toast.makeText(
                    this@ChatActivity, "Failed to load nanny details",
                    Toast.LENGTH_SHORT
                ).show()
                Log.e(TAG, "Failed to load nanny details: ${e.message}")
            }
        }
    }

    private fun loadCurrentUserNameAndPhotoUrl() {
        val currentUser = auth.currentUser
        currentUser?.uid

        if (currentUser != null) {
            GlobalScope.launch(Dispatchers.Main) {
                try {
                    val document =

```

```

        firestore.collection("User").document(currentUserId!!).get().await()
        currentUserName = document.getString("fullName")
        currentUserPhotoUrl = document.getString("profileImageUrl")
        Log.d(
            TAG, "Current user name: $currentUserName, photo URL: $currentUserPhotoUrl"
        )
    } catch (exception: Exception) {
        Log.e(TAG, "Failed to fetch user data: ${exception.message}")
    }
}
}

private fun loadMessages() {
    messageReference.addChildEventListener(object : ChildEventListener {
        override fun onChildAdded(snapshot: DataSnapshot, previousChildName: String?) {
            val message = snapshot.getValue(ChatMessage::class.java)
            if (message != null && ((message.senderId == currentUserId &&
                message.receiverId == nannyId) || (message.senderId == nannyId &&
                message.receiverId == currentUserId))) {
                GlobalScope.launch(Dispatchers.Main) {
                    val latestUserInfo = getLatestUserInfo(message.senderId)
                    if (latestUserInfo != null) {
                        message.senderName = latestUserInfo.first ?: message.senderName
                        message.senderPhotoUrl = latestUserInfo.second ?: message.senderPhotoUrl
                    }
                    adapter.addMessage(message)
                }
                binding.messageRecyclerView.scrollToPosition(adapter.itemCount - 1)
                Log.d(TAG, "Message added: $message")
            }
        } else {
            Log.e(TAG, "Failed to parse message or message is not part of
the current chat")
        }
    }

    override fun onChildChanged(snapshot: DataSnapshot, previousChildName: String?) {}
    override fun onChildRemoved(snapshot: DataSnapshot) {}
    override fun onChildMoved(snapshot: DataSnapshot, previousChildName: String?) {}
    override fun onCancelled(error: DatabaseError) {
        Toast.makeText(this@ChatActivity, "Failed to load messages",
        Toast.LENGTH_SHORT)
            .show()
        Log.e(TAG, "Failed to load messages: ${error.message}")
    }
})
}

private suspend fun getLatestUserInfo(senderId: String): Pair<String?, String?>? {
    return try {
        val document =
            firestore.collection("User").document(senderId).get().await()
        val fullName = document.getString("fullName")
        val profileImageUrl = document.getString("profileImageUrl")
        Pair(fullName, profileImageUrl)
    } catch (e: Exception) {
        null
    }
}

```

```

    }

    private fun sendMessage() {
        val messageText = binding.messageEditText.text.toString().trim()
        if (messageText.isNotEmpty()) {
            val currentUser = auth.currentUser
            val currentUserId = currentUser?.uid ?: ""
            val currentUserName = this.currentUserName ?: "Anonymous"
            val currentUserPhotoUrl = this.currentUserPhotoUrl ?: ""
            val timestamp = System.currentTimeMillis()
            val formattedTimestamp = formatDate(timestamp)
            val message = ChatMessage(
                senderId = currentUserId,
                senderName = currentUserName,
                senderPhotoUrl = currentUserPhotoUrl,
                message = messageText,
                timestamp = timestamp,
                formattedTimestamp = formattedTimestamp,
                receiverId = nannyId!!
            )
            Log.d(TAG, "Trying to send message: $message")
            messageReference.push().setValue(message).addOnSuccessListener {
                binding.messageEditText.text.clear()
                Log.d(TAG, "Message sent successfully: $messageText")
            }.addOnFailureListener { exception ->
                Toast.makeText(this, "Failed to send message",
                Toast.LENGTH_SHORT).show()
                Log.e(TAG, "Failed to send message: ${exception.message}")
            }
        } else {
            Toast.makeText(this, "Please enter a message",
            Toast.LENGTH_SHORT).show()
        }
    }

    private fun formatDate(timestamp: Long): String {
        val dateFormat = SimpleDateFormat("dd/MM/yyyy HH:mm:ss",
        Locale.getDefault())
        dateFormat.timeZone = TimeZone.getDefault()
        return dateFormat.format(Date(timestamp))
    }

    companion object {
        private const val TAG = "ChatActivity"
    }

    override fun onBackPressed() {
        super.onBackPressed()
        finish()
    }
}

```

[File activity DetailHistoryBDActivity.kt]

```

package com.app.famcare.view.detailhistory

import android.content.DialogInterface
import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.util.Log
import android.view.MenuItem
import android.widget.Button
import android.widget.ImageView
import android.widget.TextView

```

```

import android.widget.Toast
import androidx.appcompat.app.AlertDialog
import androidx.appcompat.widget.Toolbar
import com.app.famcare.R
import com.app.famcare.model.BookingDailyHistory
import com.app.famcare.view.history.HistoryActivity
import com.bumptech.glide.Glide
import com.google.firebase.auth.FirebaseAuth
import com.google.firebase.firestore.FieldValue
import com.google.firebase.firestore.FirebaseFirestore
import java.text.NumberFormat
import java.util.Locale

class DetailHistoryBDActivity : AppCompatActivity() {

    private lateinit var bookingID: String
    private lateinit var nannyID: String
    private lateinit var imageProfile: ImageView
    private lateinit var textName: TextView
    private lateinit var textEmail: TextView
    private lateinit var viewBookDate: TextView
    private lateinit var viewBookingDuration: TextView
    private lateinit var viewStartTime: TextView
    private lateinit var viewEndTime: TextView
    private lateinit var textSalary: TextView
    private lateinit var textViewID: TextView

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

        val toolbar = findViewById<Toolbar>(R.id.toolbar)
        setSupportActionBar(toolbar)
        supportActionBar?.setDisplayHomeAsUpEnabled(true)
        supportActionBar?.title = ""

        imageProfile = findViewById(R.id.image_profile)
        textName = findViewById(R.id.text_name)
        textEmail = findViewById(R.id.text_email)
        viewBookDate = findViewById(R.id.viewBookDate)
        viewBookingDuration = findViewById(R.id.viewBookingDuration)
        viewStartTime = findViewById(R.id.viewStartTime)
        viewEndTime = findViewById(R.id.viewEndTime)
        textSalary = findViewById(R.id.text_salary)
        textViewID = findViewById(R.id.textViewID)

        bookingID = intent.getStringExtra("bookingID") ?: ""
        if (bookingID.isNotEmpty()) {
            loadBookingData(bookingID)
        } else {
            Toast.makeText(this, "Booking ID not found",
            Toast.LENGTH_SHORT).show()
        }

        val buttonCancelBook = findViewById<Button>(R.id.buttonCancelBook)
        buttonCancelBook.setOnClickListener {
            showConfirmationDialog()
        }
    }

    override fun onOptionsItemSelected(item: MenuItem): Boolean {
        return when (item.itemId) {
            android.R.id.home -> {
                onBackPressed()
                true
            }
        }
    }
}

```

```

        else -> super.onOptionsItemSelected(item)
    }
}

private fun loadBookingData(bookingID: String) {
    val db = Firebase Firestore.getInstance()

    db.collection("BookingDaily").document(bookingID).get().addOnSuccessListener {
        document ->
        if (document != null && document.exists()) {
            val booking = document.toObject(BookingDailyHistory::class.java)
            booking?.let {
                textViewID.text = bookingID
                textName.text = it.nannyName
                textEmail.text = it.type.name.toLowerCase(Locale.ROOT)
                viewBookDate.text = it.bookDate
                viewBookingDuration.text = "${it.bookDuration} Hours"
                viewStartTime.text = it.bookHours
                viewEndTime.text = it.endHours

                val localeID = Locale("id", "ID")
                val formattedCurrency =
                    NumberFormat.getCurrencyInstance(localeID).format(it.totalPricing.toLong())
                textSalary.text = formattedCurrency

                nannyID = it.nannyID

                loadNannyData(it.nannyID)
            }
        } else {
            Toast.makeText(this, "No booking data found",
                Toast.LENGTH_SHORT).show()
        }
    }.addOnFailureListener { exception ->
        Log.w(TAG, "Error getting booking document", exception)
        Toast.makeText(this, "Failed to load booking data",
            Toast.LENGTH_SHORT).show()
    }
}

private fun loadNannyData(nannyID: String) {
    val db = Firebase Firestore.getInstance()
    db.collection("Nanny").document(nannyID).get().addOnSuccessListener {
        document ->
        if (document != null && document.exists()) {
            val nannyName = document.getString("name") ?: ""
            val nannyPict = document.getString("pict") ?: ""
            textName.text = nannyName

            Glide.with(this).load(nannyPict).placeholder(R.drawable.nanny_placeholder)
                .error(R.drawable.placeholder_image).into(imageProfile)
        } else {
            Toast.makeText(this, "Nanny data not found",
                Toast.LENGTH_SHORT).show()
        }
    }.addOnFailureListener { exception ->
        Log.w(TAG, "Error getting nanny document", exception)
        Toast.makeText(this, "Failed to load nanny data",
            Toast.LENGTH_SHORT).show()
    }
}

private fun showConfirmationDialog() {
    val builder = AlertDialog.Builder(this)
    builder.setTitle("Cancel Booking")
    builder.setMessage("Are you sure you want to cancel this booking?")
}

```

```

        builder.setPositiveButton("Yes") { dialogInterface: DialogInterface, i: Int ->
            cancelBooking()
        }
        builder.setNegativeButton("Cancel") { dialogInterface: DialogInterface, i: Int ->
            dialogInterface.dismiss()
        }
        builder.show()
    }

    private fun cancelBooking() {
        val db = FirebaseFirestore.getInstance()
        val currentUserID = FirebaseAuth.getInstance().currentUser?.uid ?: return

        db.collection("User").document(currentUserID)
            .update("bookIDs",
                FieldValue.arrayRemove(bookingID)).addOnSuccessListener {
            db.collection("BookingDaily").document(bookingID).delete().addOnSuccessListener {
                showSuccessDialog()
            }.addOnFailureListener { e ->
                Log.w(TAG, "Error deleting booking document", e)
                Toast.makeText(this, "Failed to cancel booking",
                    Toast.LENGTH_SHORT).show()
            }
        }.addOnFailureListener { e ->
            Log.w(TAG, "Error updating user document", e)
            Toast.makeText(this, "Failed to update user data",
                Toast.LENGTH_SHORT).show()
        }
    }

    private fun showSuccessDialog() {
        val builder = AlertDialog.Builder(this)
        builder.setTitle("Booking Cancelled")
        builder.setMessage("Your booking has been cancelled successfully.")
        builder.setPositiveButton("OK") { dialogInterface: DialogInterface, i: Int ->
            dialogInterface.dismiss()
            val intent = Intent(this, HistoryActivity::class.java)
            intent.putExtra("selectedTab", 0)
            startActivity(intent)
            finish()
        }
        builder.show()
    }

    companion object {
        private const val TAG = "DetailHistoryBDAcivity"
    }
}

```

[File activity DetailHistoryBMActivity.kt]

```

package com.app.famcare.view.detailhistory

import android.content.DialogInterface
import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.util.Log
import android.view.MenuItem
import android.widget.Button
import android.widget.ImageView
import android.widget.TextView
import android.widget.Toast

```

```

import androidx.appcompat.app.AlertDialog
import androidx.appcompat.widget.Toolbar
import com.app.famcare.R
import com.app.famcare.model.BookingMonthlyHistory
import com.app.famcare.view.history.HistoryActivity
import com.google.firebase.auth.FirebaseAuth
import com.bumptech.glide.Glide
import com.google.firebaseio.firestore.FieldValue
import com.google.firebaseio.firebaseio.FirebaseFirestore
import java.text.NumberFormat
import java.text.SimpleDateFormat
import java.util.*
import java.util.concurrent.TimeUnit

class DetailHistoryBMActivity : AppCompatActivity() {
    private lateinit var bookingID: String
    private lateinit var nannyID: String
    private lateinit var imageProfile: ImageView
    private lateinit var textName: TextView
    private lateinit var textEmail: TextView
    private lateinit var viewBookingDuration: TextView
    private lateinit var viewStartDate: TextView
    private lateinit var viewEndDate: TextView
    private lateinit var textSalary: TextView
    private lateinit var textViewID: TextView

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

        val toolbar = findViewById<Toolbar>(R.id.toolbar)
        setSupportActionBar(toolbar)
        supportActionBar?.setDisplayHomeAsUpEnabled(true)
        supportActionBar?.title = ""

        imageProfile = findViewById(R.id.image_profile)
        textName = findViewById(R.id.text_name)
        textEmail = findViewById(R.id.text_email)
        viewBookingDuration = findViewById(R.id.viewBookingDuration)
        viewStartDate = findViewById(R.id.viewStartDate)
        viewEndDate = findViewById(R.id.viewEndDate)
        textSalary = findViewById(R.id.text_salary)
        textViewID = findViewById(R.id.textViewID)

        bookingID = intent.getStringExtra("bookingID") ?: ""
        if (bookingID.isNotEmpty()) {
            loadBookingData(bookingID)
        } else {
            Toast.makeText(this, "Booking ID not found",
            Toast.LENGTH_SHORT).show()
        }

        val buttonCancelBook = findViewById<Button>(R.id.buttonCancelBook)
        buttonCancelBook.setOnClickListener {
            showConfirmationDialog()
        }
    }

    override fun onOptionsItemSelected(item: MenuItem): Boolean {
        return when (item.itemId) {
            android.R.id.home -> {
                onBackPressed()
                true
            }
            else -> super.onOptionsItemSelected(item)
        }
    }
}

```

```

    }

    private fun loadBookingData(bookingID: String) {
        val db = FirebaseFirestore.getInstance()

        db.collection("BookingMonthly").document(bookingID).get().addOnSuccessListener {
            document ->
                if (document != null && document.exists()) {
                    val booking = document.toObject(BookingMonthlyHistory::class.java)
                    booking?.let {
                        textViewID.text = bookingID
                        textName.text = it.nannyName
                        textEmail.text = it.type.name.toLowerCase(Locale.ROOT)
                        viewStartDate.text = it.startDate
                        viewEndDate.text = it.endDate
                        val localeID = Locale("id", "ID")
                        val formattedCurrency =
                            NumberFormat.getCurrencyInstance(localeID).format(it.totalPricing.toLong())
                        textSalary.text = formattedCurrency

                        nannyID = it.nannyID

                        viewBookingDuration.text =
                            calculateBookingDuration(it.startDate, it.endDate)

                        loadNannyData(it.nannyID)
                    }
                } else {
                    Toast.makeText(this, "No booking data found",
                        Toast.LENGTH_SHORT).show()
                }
            }.addOnFailureListener { exception ->
                Log.w(TAG, "Error getting booking document", exception)
                Toast.makeText(this, "Failed to load booking data",
                    Toast.LENGTH_SHORT).show()
            }
        }
    }

    private fun calculateBookingDuration(startDateStr: String, endDateStr: String): String {
        val dateFormat = SimpleDateFormat("dd/MM/yyyy", Locale.US)
        return try {
            val startDate = dateFormat.parse(startDateStr)
            val endDate = dateFormat.parse(endDateStr)
            val diffInMillis = endDate.time - startDate.time
            val diffInDays = TimeUnit.MILLISECONDS.toDays(diffInMillis)
            val months = (diffInDays / 30).toInt()
            "$months months"
        } catch (e: Exception) {
            e.printStackTrace()
            "Unknown duration"
        }
    }

    private fun loadNannyData(nannyID: String) {
        val db = FirebaseFirestore.getInstance()
        db.collection("Nanny").document(nannyID).get().addOnSuccessListener {
            document ->
                if (document != null && document.exists()) {
                    val nannyName = document.getString("name") ?: ""
                    val nannyPict = document.getString("pict") ?: ""

                    textName.text = nannyName

                    Glide.with(this).load(nannyPict).placeholder(R.drawable.nanny_placeholder)
                        .error(R.drawable.placeholder_image).into(imageProfile)
                }
            }
        }
    }
}

```

```

        } else {
            Toast.makeText(this, "Nanny data not found",
Toast.LENGTH_SHORT).show()
        }
    }.addOnFailureListener { exception ->
    Log.w(TAG, "Error getting nanny document", exception)
    Toast.makeText(this, "Failed to load nanny data",
Toast.LENGTH_SHORT).show()
}
}

private fun showConfirmationDialog() {
    val builder = AlertDialog.Builder(this)
    builder.setTitle("Cancel Booking")
    builder.setMessage("Are you sure you want to cancel this booking?")
    builder.setPositiveButton("Yes") { dialogInterface: DialogInterface, i: Int ->
        cancelBooking()
    }
    builder.setNegativeButton("Cancel") { dialogInterface: DialogInterface, i: Int ->
        dialogInterface.dismiss()
    }
    builder.show()
}

private fun cancelBooking() {
    val db = FirebaseFirestore.getInstance()
    val currentUserID = FirebaseAuth.getInstance().currentUser?.uid ?: return

    db.collection("User").document(currentUserID)
        .update("bookIDs",
        FieldValue.arrayRemove(bookingID)).addOnSuccessListener {
        db.collection("BookingMonthly").document(bookingID).delete().addOnSuccessListener {
            showSuccessDialog()
        }.addOnFailureListener { e ->
            Log.w(TAG, "Error deleting booking document", e)
            Toast.makeText(this, "Failed to cancel booking",
Toast.LENGTH_SHORT).show()
        }
    }.addOnFailureListener { e ->
        Log.w(TAG, "Error updating user document", e)
        Toast.makeText(this, "Failed to update user data",
Toast.LENGTH_SHORT).show()
    }
}

private fun showSuccessDialog() {
    val builder = AlertDialog.Builder(this)
    builder.setTitle("Booking Cancelled")
    builder.setMessage("Your booking has been cancelled successfully.")
    builder.setPositiveButton("OK") { dialogInterface: DialogInterface, i: Int ->
        dialogInterface.dismiss()
        val intent = Intent(this, HistoryActivity::class.java)
        intent.putExtra("selectedTab", 1)
        startActivity(intent)
        finish()
    }
    builder.show()
}

companion object {
    private const val TAG = "DetailHistoryBMAActivity"
}

```

```
}
```

[File XML activity_book_daily.kt]

```
<androidx.coordinatorlayout.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".view.booking.BookDailyActivity">

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/appToolbar"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar">

        <com.google.android.material.appbar.CollapsingToolbarLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            app:layout_scrollFlags="scroll|exitUntilCollapsed">

            <androidx.appcompat.widget.Toolbar
                android:id="@+id/toolbar"
                android:layout_width="match_parent"
                android:layout_height="?attr/actionBarSize"
                android:background="?attr/colorPrimary"
                android:elevation="4dp"
                app:layout_scrollFlags="scroll|enterAlways">

                <TextView
                    android:id="@+id/toolbar_title"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:fontFamily="@font/montserratsemibold"
                    android:text="Booking Confirmation"
                    android:textColor="@color/white"
                    android:textSize="18sp" />

            </androidx.appcompat.widget.Toolbar>
        </com.google.android.material.appbar.CollapsingToolbarLayout>
    </com.google.android.material.appbar.AppBarLayout>

    <androidx.core.widget.NestedScrollView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="fill_vertical"
        android:fillViewport="true"
        app:layout_behavior="@string/appbar_scrolling_view_behavior">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:orientation="vertical">

            <androidx.cardview.widget.CardView
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginStart="16dp"
                android:layout_marginTop="16dp"
                android:layout_marginEnd="16dp"
                android:layout_marginBottom="16dp"
                app:cardCornerRadius="20dp"
                app:layout_constraintEnd_toEndOf="parent"
                app:layout_constraintStart_toStartOf="parent"
                app:layout_constraintTop_toBottomOf="@+id/imageViewNanny">
```

```
<RelativeLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:padding="16dp">

    <androidx.cardview.widget.CardView
        android:id="@+id/profileNanny"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="16dp"
        app:cardBackgroundColor="@color/main"
        app:cardCornerRadius="30dp"
        app:cardElevation="10dp">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center_vertical"
            android:orientation="horizontal"
            android:paddingTop="16dp"
            android:paddingBottom="16dp">

            <de.hdodenhof.circleimageview.CircleImageView
                android:id="@+id/image_profile"
                android:layout_width="100dp"
                android:layout_height="100dp"
                android:layout_marginStart="10dp"
                android:scaleType="centerCrop"
                android:src="@drawable/nanny_placeholder" />

            <LinearLayout
                android:layout_width="0dp"
                android:layout_height="match_parent"
                android:layout_weight="1"
                android:gravity="center_vertical|start"
                android:orientation="vertical"
                android:paddingStart="16dp"
                android:paddingEnd="16dp">

                <TextView
                    android:id="@+id/text_name"
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content"
                    android:fontFamily="@font/montserratsemibold"
                    android:text="Putri Utami"
                    android:textColor="@color/black"
                    android:textSize="18sp"
                    android:textStyle="bold" />

                <LinearLayout
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content"
                    android:layout_marginTop="8dp"
                    android:gravity="center_vertical">

                    <ImageView
                        android:layout_width="20dp"
                        android:layout_height="20dp"
                        android:src="@drawable/ic_calendar" />

                    <TextView
                        android:id="@+id/text_email"
                        android:layout_width="wrap_content"
                        android:layout_height="wrap_content"
                        android:layout_marginStart="4dp"
```

```
        android:fontFamily="@font/montserratsemibold"
                android:text="daily"
                android:textColor="@color/black"
                android:textSize="16sp" />
            </LinearLayout>
        </LinearLayout>
    </LinearLayout>
</androidx.cardview.widget.CardView>

<LinearLayout
    android:id="@+id/formBooking"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/profileNanny"
    android:orientation="vertical">

    <TextView
        android:id="@+id/bookingDateTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="20dp"
        android:layout_marginTop="15dp"
        android:fontFamily="@font/montserratsemibold"
        android:text="Booking Date"
        android:textColor="@color/black"
        android:textSize="16sp"
        android:textStyle="bold" />

    <EditText
        android:id="@+id/bookingDateEditText"
        android:layout_width="match_parent"
        android:layout_height="30dp"
        android:layout_marginStart="20dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="20dp"
        android:background="@drawable/bg_spinner"
        android:focusable="false"
        android:fontFamily="@font/montserrat"
        android:hint=" choose booking Date"
        android:inputType="none"
        android:textSize="16sp" />

    <TextView
        android:id="@+id/bookingDurationTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="20dp"
        android:layout_marginTop="15dp"
        android:fontFamily="@font/montserratsemibold"
        android:text="Booking Duration (Hours)"
        android:textColor="@color/black"
        android:textSize="16sp"
        android:textStyle="bold" />

    <Spinner
        android:id="@+id/bookingDurationSpinner"
        android:layout_width="match_parent"
        android:layout_height="30dp"
        android:layout_marginStart="20dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="20dp"
        android:background="@drawable/bg_spinner"
        android:fontFamily="@font/montserrat"
        android:spinnerMode="dropdown"
        android:textSize="16sp" />

```

```
<TextView
    android:id="@+id/startHourTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="20dp"
    android:layout_marginTop="15dp"
    android:fontFamily="@font/montserratsemibold"
    android:text="Select Start Hour"
    android:textColor="@color/black"
    android:textSize="16sp"
    android:textStyle="bold" />

<EditText
    android:id="@+id/startHourEditText"
    android:layout_width="match_parent"
    android:layout_height="30dp"
    android:layout_marginStart="20dp"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="20dp"
    android:background="@drawable/bg_spinner"
    android:focusable="false"
    android:fontFamily="@font/montserrat"
    android:hint=" choose start booking time"
    android:inputType="none"
    android:textSize="16sp" />

<TextView
    android:id="@+id/outputBookingDate"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="20dp"
    android:layout_marginTop="15dp"
    android:layout_marginEnd="20dp"
    android:fontFamily="@font/montserratsemibold"
    android:text="Booking Date : "
    android:textColor="@color/black"
    android:textSize="16sp"
    android:textStyle="bold" />

<TextView
    android:id="@+id/outputBookingDuration"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="20dp"
    android:layout_marginTop="15dp"
    android:fontFamily="@font/montserratsemibold"
    android:text="Booking Duration : "
    android:textColor="@color/black"
    android:textSize="16sp"
    android:textStyle="bold" />

<TextView
    android:id="@+id/outputStartHour"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="20dp"
    android:layout_marginTop="15dp"
    android:layout_marginEnd="20dp"
    android:fontFamily="@font/montserratsemibold"
    android:text="Start Hour : "
    android:textColor="@color/black"
    android:textSize="16sp"
    android:textStyle="bold" />

<TextView
    android:id="@+id/outputEndHour"
    android:layout_width="wrap_content"
```

```

        android:layout_height="wrap_content"
        android:layout_marginStart="20dp"
        android:layout_marginTop="15dp"
        android:layout_marginEnd="20dp"
        android:fontFamily="@font/montserratsemibold"
        android:text="End Time : "
        android:textColor="@color/black"
        android:textSize="16sp"
        android:textStyle="bold" />

    <TextView
        android:id="@+id/textCost"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="20dp"
        android:layout_marginTop="10dp"
        android:layout_marginEnd="5dp"
        android:fontFamily="@font/montserratsemibold"
        android:gravity="top|start"
        android:text="Total Cost : "
        android:textColor="@color/black"
        android:textSize="16sp"
        android:textStyle="bold" />

</LinearLayout>

<Button
    android:id="@+id/buttonBookNanny"
    android:layout_width="match_parent"
    android:layout_height="60dp"
    android:layout_below="@+id/formBooking"
    android:layout_centerHorizontal="true"
    android:layout_gravity="center"
    android:layout_marginStart="32dp"
    android:layout_marginTop="60dp"
    android:layout_marginEnd="32dp"
    android:layout_marginBottom="15dp"
    android:fontFamily="@font/montserratsemibold"
    android:text="Book Now"
    android:textSize="20sp"
    android:textStyle="bold" />

</RelativeLayout>

</androidx.cardview.widget.CardView>
</LinearLayout>
</androidx.core.widget.NestedScrollView>
</androidx.coordinatorlayout.widget.CoordinatorLayout>

```

[File XML activity_book_monthly.kt]

```

<androidx.coordinatorlayout.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".view.booking.BookDailyActivity">

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/appToolbar"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar">

        <com.google.android.material.appbar.CollapsingToolbarLayout
            android:layout_width="match_parent"

```

```

        android:layout_height="wrap_content"
        app:layout_scrollFlags="scroll|exitUntilCollapsed">

        <androidx.appcompat.widget.Toolbar
            android:id="@+id/toolbar"
            android:layout_width="match_parent"
            android:layout_height="?attr/actionBarSize"
            android:background="?attr/colorPrimary"
            android:elevation="4dp"
            app:layout_scrollFlags="scroll|enterAlways">

            <TextView
                android:id="@+id/toolbar_title"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:fontFamily="@font/montserratsemibold"
                android:text="Booking Confirmation"
                android:textColor="@color/white"
                android:textSize="18sp" />

        </androidx.appcompat.widget.Toolbar>
    </com.google.android.material.appbar.CollapsingToolbarLayout>
</com.google.android.material.AppBarLayout>

<androidx.core.widget.NestedScrollView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="fill_vertical"
    android:fillViewport="true"
    app:layout_behavior="@string/appbar_scrolling_view_behavior">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <androidx.cardview.widget.CardView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginStart="16dp"
            android:layout_marginTop="16dp"
            android:layout_marginEnd="16dp"
            android:layout_marginBottom="16dp"
            app:cardCornerRadius="20dp"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@+id/imageViewNanny">

            <RelativeLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:padding="16dp">

                <androidx.cardview.widget.CardView
                    android:id="@+id/profileNanny"
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content"
                    android:layout_margin="16dp"
                    app:cardBackgroundColor="@color/main"
                    app:cardCornerRadius="30dp"
                    app:cardElevation="10dp">

                    <LinearLayout
                        android:layout_width="match_parent"
                        android:layout_height="wrap_content"
                        android:gravity="center_vertical"
                        android:orientation="horizontal">

```

```

        android:paddingTop="16dp"
        android:paddingBottom="16dp">

        <de.hdodenhof.circleimageview.CircleImageView
            android:id="@+id/image_profile"
            android:layout_width="100dp"
            android:layout_height="100dp"
            android:layout_marginStart="10dp"
            android:scaleType="centerCrop"
            android:src="@drawable/nanny_placeholder" />

        <LinearLayout
            android:layout_width="0dp"
            android:layout_height="match_parent"
            android:layout_weight="1"
            android:gravity="center_vertical|start"
            android:orientation="vertical"
            android:paddingStart="16dp"
            android:paddingEnd="16dp">

            <TextView
                android:id="@+id/text_name"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:fontFamily="@font/montserratsemibold"
                android:text="Putri Utami"
                android:textColor="@color/black"
                android:textSize="18sp"
                android:textStyle="bold" />

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginTop="8dp"
                android:gravity="center_vertical">

                <ImageView
                    android:layout_width="20dp"
                    android:layout_height="20dp"
                    android:src="@drawable/ic_calendar" />

                <TextView
                    android:id="@+id/text_email"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:layout_marginStart="4dp"
                    android:fontFamily="@font/montserratsemibold"
                    android:text="monthly"
                    android:textColor="@color/black"
                    android:textSize="16sp" />
            
```

</LinearLayout>

</LinearLayout>

</androidx.cardview.widget.CardView>

<LinearLayout

android:id="@+id/formBooking"

android:layout_width="match_parent"

android:layout_height="wrap_content"

android:layout_below="@+id/profileNanny"

android:orientation="vertical">

<TextView

android:id="@+id/startDateTextView"
 android:layout_width="wrap_content"

```
        android:layout_height="wrap_content"
        android:layout_marginStart="20dp"
        android:layout_marginTop="15dp"
        android:fontFamily="@font/montserratsemibold"
        android:text="Booking Duration (Months)"
        android:textColor="@color/black"
        android:textSize="16sp"
        android:textStyle="bold" />

    <Spinner
        android:id="@+id/bookingDurationSpinner"
        android:layout_width="match_parent"
        android:layout_height="30dp"
        android:layout_marginStart="20dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="20dp"
        android:background="@drawable/bg_spinner"
        android:fontFamily="@font/montserrat"
        android:spinnerMode="dropdown"
        android:textSize="16sp" />

    <TextView
        android:id="@+id/selectTimeTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="20dp"
        android:layout_marginTop="15dp"
        android:fontFamily="@font/montserratsemibold"
        android:text="Select Start Date"
        android:textColor="@color/black"
        android:textSize="16sp"
        android:textStyle="bold" />

    <EditText
        android:id="@+id/startDateEditText"
        android:layout_width="match_parent"
        android:layout_height="30dp"
        android:layout_marginStart="20dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="20dp"
        android:background="@drawable/bg_spinner"
        android:focusable="false"
        android:fontFamily="@font/montserrat"
        android:hint=" choose booking Date"
        android:inputType="none"
        android:textSize="16sp" />

    <TextView
        android:id="@+id/outputBookingDuration"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="20dp"
        android:layout_marginTop="15dp"
        android:fontFamily="@font/montserratsemibold"
        android:text="Booking Duration : "
        android:textColor="@color/black"
        android:textSize="16sp"
        android:textStyle="bold" />

    <TextView
        android:id="@+id/outputStartDate"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="20dp"
        android:layout_marginTop="15dp"
        android:layout_marginEnd="20dp"
        android:fontFamily="@font/montserratsemibold"
```

```

        android:text="Start Date : "
        android:textColor="@color/black"
        android:textSize="16sp"
        android:textStyle="bold" />

    <TextView
        android:id="@+id/outputEndDate"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="20dp"
        android:layout_marginTop="15dp"
        android:layout_marginEnd="20dp"
        android:fontFamily="@font/montserratsemibold"
        android:text="End Date : "
        android:textColor="@color/black"
        android:textSize="16sp"
        android:textStyle="bold" />

    <TextView
        android:id="@+id/textCost"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="20dp"
        android:layout_marginEnd="5dp"
        android:layout_marginTop="10dp"
        android:fontFamily="@font/montserratsemibold"
        android:gravity="top|start"
        android:text="Total Cost :"
        android:textColor="@color/black"
        android:textSize="16sp"
        android:textStyle="bold" />

    </LinearLayout>

    <Button
        android:id="@+id/buttonBookNanny"
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:layout_below="@+id/formBooking"
        android:layout_centerHorizontal="true"
        android:layout_gravity="center"
        android:layout_marginStart="32dp"
        android:layout_marginTop="60dp"
        android:layout_marginEnd="32dp"
        android:layout_marginBottom="15dp"
        android:fontFamily="@font/montserratsemibold"
        android:text="Book Now"
        android:textSize="20sp"
        android:textStyle="bold" />

    </RelativeLayout>

    </androidx.cardview.widget.CardView>
</LinearLayout>
</androidx.core.widget.NestedScrollView>
</androidx.coordinatorlayout.widget.CoordinatorLayout>
```

[File XML fragment_history_b_d.xml]

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
```

```

        android:layout_height="match_parent"
        tools:context=".view.historyimport.HistoryBDFragment">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <androidx.core.widget.NestedScrollView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_gravity="fill_vertical"
            android:fillViewport="true"
            app:layout_behavior="@string/appbar_scrolling_view_behavior">

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="match_parent"
                android:orientation="vertical">

                <androidx.recyclerview.widget.RecyclerView
                    android:id="@+id/recyclerViewHistory"
                    android:layout_width="match_parent"
                    android:layout_height="match_parent"
                    android:layout_marginTop="0dp"
                    android:layout_marginBottom="56dp"
                    android:padding="8dp"
                    tools:listitem="@layout/item_row_history_b_d" />

            </LinearLayout>
        </androidx.core.widget.NestedScrollView>
    </LinearLayout>

</androidx.coordinatorlayout.widget.CoordinatorLayout>

```

[File XML fragment_history_b_m.xml]

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".view.historyimport.HistoryBMFragment">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <androidx.core.widget.NestedScrollView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_gravity="fill_vertical"
            android:fillViewport="true"
            app:layout_behavior="@string/appbar_scrolling_view_behavior">

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="match_parent"
                android:orientation="vertical">

                <androidx.recyclerview.widget.RecyclerView
                    android:id="@+id/recyclerViewHistoryBM"
                    android:layout_width="match_parent"
                    android:layout_height="match_parent"

```

```

        android:layout_marginTop="0dp"
        android:layout_marginBottom="56dp"
        android:padding="8dp"
        tools:listitem="@layout/item_row_history_b_m" />

        <TextView
            android:id="@+id/emptyTextView"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="300dp"
            android:gravity="center"
            android:text=" "
            android:textSize="18sp"
            android:visibility="gone" />

    </LinearLayout>
</androidx.core.widget.NestedScrollView>
</LinearLayout>

</androidx.coordinatorlayout.widget.CoordinatorLayout>

```

[File XML item_message.xml]

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="8dp">

    <de.hdodenhoef.circleimageview.CircleImageView
        android:id="@+id/ivMessenger"
        android:layout_width="38dp"
        android:layout_height="35dp"
        android:layout_marginTop="8dp"
        android:src="@drawable/baseline_account_circle_24"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <TextView
        android:id="@+id/tvMessenger"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:layout_marginEnd="4dp"
        android:fontFamily="@font/montserratsemibold"
        android:textAppearance="?android:attr/textAppearanceSmall"
        android:textSize="16sp"
        app:layout_constraintEnd_toStartOf="@+id/ivMessenger"
        app:layout_constraintTop_toTopOf="parent"
        tools:text="FirstName LastName" />

    <TextView
        android:id="@+id/tvMessage"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="4dp"
        android:background="@drawable/rounded_message_tosca"
        android:fontFamily="@font/montserrat"
        android:maxLength="250dp"
        android:padding="10dp"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:textColor="@android:color/white"
        android:textSize="16sp"
        app:layout_constraintEnd_toEndOf="@+id/tvMessenger" />

```

```

        app:layout_constraintTop_toBottomOf="@+id/tvMessenger"
        tools:text="This is a message" />

    <TextView
        android:id="@+id/tvTimestamp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom"
        android:layout_marginEnd="4dp"
        android:fontFamily="@font/montserrat"
        android:maxLines="2"
        android:textSize="12sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toStartOf="@+id/tvMessage"
        app:layout_constraintHorizontal_bias="1.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="1.0"
        tools:text="Just Now" />

</androidx.constraintlayout.widget.ConstraintLayout>

```

[File activity_chat_list.xml]

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".view.chat.ChatListActivity">

</androidx.constraintlayout.widget.ConstraintLayout>

```

[File activity_chat.xml]

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".view.chat.ChatActivity">

    <androidx.constraintlayout.widget.ConstraintLayout
        android:id="@+id/header"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@color/icon_active">

        <ImageView
            android:id="@+id/ivBack"
            android:layout_width="24dp"
            android:layout_height="24dp"
            android:layout_marginStart="10dp"
            android:layout_marginTop="28dp"
            android:src="@drawable/ic_arrow_back"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintHorizontal_bias="0.005"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent" />

        <de.hdodenhof.circleimageview.CircleImageView
            android:id="@+id/ivNannyPhoto"
            android:layout_width="56dp"
            android:layout_height="56dp"

```

```

        android:layout_marginStart="12dp"
        android:layout_marginTop="12dp"
        android:layout_marginBottom="12dp"
        android:src="@drawable/nanny_placeholder"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintStart_toEndOf="@+id/ivBack"
        app:layout_constraintTop_toTopOf="parent" />

    <TextView
        android:id="@+id/tvNannyName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="8dp"
        android:fontFamily="@font/montserratsemibold"
        android:text="Putri Utami"
        android:textColor="@color/white"
        android:textSize="18sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.003"
        app:layout_constraintStart_toEndOf="@+id/ivNannyPhoto"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.508" />

    </androidx.constraintlayout.widget.ConstraintLayout>

    <LinearLayout
        android:id="@+id/linearLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:background="@color/main"
        android:orientation="horizontal"
        android:paddingHorizontal="10dp">

        <EditText
            android:id="@+id/messageEditText"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:layout_margin="10dp"
            android:layout_marginStart="10dp"
            android:layout_marginTop="10dp"
            android:layout_marginEnd="10dp"
            android:layout_marginBottom="10dp"
            android:layout_weight="1"
            android:background="@drawable/edit_text_shadow"
            android:hint="Say something..."
            android:inputType="text"
            android:minHeight="48dp"
            android:padding="10dp" />

        <ImageView
            android:id="@+id/sendButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:enabled="false"
            android:src="@drawable/ic_send" />

    </LinearLayout>

    <ProgressBar
        android:id="@+id/progressBar"
        style="?android:attr/progressBarStyleLarge"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"

```

```

        android:layout_centerHorizontal="true"
        android:layout_centerVertical="true" />

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/messageRecyclerView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_above="@+id/linearLayout"
        android:layout_below="@+id/header"
        android:layout_marginBottom="8dp"
        android:background="@android:color/white" />

</RelativeLayout>

```

[File XML activity_detail_history_b_d.xml]

```

<androidx.coordinatorlayout.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".view.detailhistory.DetailHistoryBDActivity">

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/appToolbar"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar">

        <com.google.android.material.appbar.CollapsingToolbarLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            app:layout_scrollFlags="scroll|exitUntilCollapsed">

            <androidx.appcompat.widget.Toolbar
                android:id="@+id/toolbar"
                android:layout_width="match_parent"
                android:layout_height="?attr/actionBarSize"
                android:background="?attr/colorPrimary"
                android:elevation="4dp"
                app:layout_scrollFlags="scroll|enterAlways">

                <TextView
                    android:id="@+id/toolbar_title"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:fontFamily="@font/montserratsemibold"
                    android:text="Booking's Details"
                    android:textColor="@color/white"
                    android:textSize="18sp" />

            </androidx.appcompat.widget.Toolbar>
        </com.google.android.material.appbar.CollapsingToolbarLayout>
    </com.google.android.material.appbar.AppBarLayout>

    <androidx.core.widget.NestedScrollView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="fill_vertical"
        android:fillViewport="true"
        app:layout_behavior="@string/appbar_scrolling_view_behavior">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:orientation="vertical">

```

```

<androidx.cardview.widget.CardView
    android:id="@+id/cardViewHistoryBD"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="16dp"
    android:layout_marginTop="16dp"
    android:layout_marginEnd="16dp"
    android:layout_marginBottom="16dp"
    app:cardCornerRadius="20dp"
    app:cardElevation="8dp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/imageViewNanny">

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="16dp">

        <LinearLayout
            android:id="@+id/idText"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="20dp"
            android:layout_marginTop="8dp"
            android:gravity="start"
            android:orientation="horizontal">

            <TextView
                android:id="@+id/textID"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_marginBottom="8dp"
                android:fontFamily="@font/montserratsemibold"
                android:text="ID Booking:"
                android:textColor="@color/black"
                android:textSize="15sp"
                android:textStyle="bold" />

            <TextView
                android:id="@+id/textViewID"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_marginStart="5dp"
                android:layout_marginBottom="8dp"
                android:fontFamily="@font/montserratsemibold"
                android:text="A4352hydsas89"
                android:textColor="@color/black"
                android:textSize="15sp"
                android:textStyle="bold" />
        

        </RelativeLayout>

        <androidx.cardview.widget.CardView
            android:id="@+id/profileNanny"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_below="@+id/idText"
            android:layout_margin="16dp"
            app:cardBackgroundColor="@color/main"
            app:cardCornerRadius="30dp"
            app:cardElevation="10dp">

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:gravity="center_vertical">

```

```

        android:orientation="horizontal"
        android:paddingTop="16dp"
        android:paddingBottom="16dp">

    <de.hdodenhof.circleimageview.CircleImageView
        android:id="@+id/image_profile"
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:layout_marginStart="10dp"
        android:scaleType="centerCrop"
        android:src="@drawable/nanny_placeholder" />

    <LinearLayout
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:gravity="center_vertical|start"
        android:orientation="vertical"
        android:paddingStart="16dp"
        android:paddingEnd="16dp">

        <TextView
            android:id="@+id/text_name"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:fontFamily="@font/montserratsemibold"
            android:text="Putri Utami"
            android:textColor="@color/black"
            android:textSize="18sp"
            android:textStyle="bold" />

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="8dp"
            android:gravity="center_vertical">

            <ImageView
                android:layout_width="20dp"
                android:layout_height="20dp"
                android:src="@drawable/ic_calendar" />

            <TextView
                android:id="@+id/text_email"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_marginStart="4dp"

                android:fontFamily="@font/montserratsemibold"
                android:text="daily"
                android:textColor="@color/black"
                android:textSize="16sp" />
        
```

</LinearLayout>

</LinearLayout>

</androidx.cardview.widget.CardView>

<LinearLayout

```

        android:id="@+id/formBooking"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/profileNanny"
        android:orientation="vertical">

        <LinearLayout
            android:layout_width="wrap_content"

```

```
        android:layout_height="wrap_content"
        android:layout_marginTop="8dp"
        android:gravity="start"
        android:orientation="horizontal">

        <TextView
            android:id="@+id/outputBookingDate"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="20dp"
            android:layout_marginTop="15dp"
            android:fontFamily="@font/montserrat"
            android:text="Booking Date : "
            android:textColor="@color/black"
            android:textSize="16sp" />

        <TextView
            android:id="@+id/viewBookDate"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="15dp"
            android:fontFamily="@font/montserrat"
            android:text="6/12/2024"
            android:textColor="@color/black"
            android:textSize="16sp" />
    </LinearLayout>

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="8dp"
        android:gravity="start"
        android:orientation="horizontal">
        <TextView
            android:id="@+id/outputBookingDuration"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="20dp"
            android:layout_marginTop="15dp"
            android:fontFamily="@font/montserrat"
            android:text="Booking Duration : "
            android:textColor="@color/black"
            android:textSize="16sp" />

        <TextView
            android:id="@+id/viewBookingDuration"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="15dp"
            android:fontFamily="@font/montserrat"
            android:text="2 Hours"
            android:textColor="@color/black"
            android:textSize="16sp" />
    </LinearLayout>

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="8dp"
        android:gravity="start"
        android:orientation="horizontal">
        <TextView
            android:id="@+id/outputStartTime"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="20dp"
```

```
        android:layout_marginTop="15dp"
        android:fontFamily="@font/montserrat"
        android:text="Start Time : "
        android:textColor="@color/black"
        android:textSize="16sp" />

    <TextView
        android:id="@+id/viewStartTime"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="15dp"
        android:fontFamily="@font/montserrat"
        android:text="17.00"
        android:textColor="@color/black"
        android:textSize="16sp" />
    </LinearLayout>

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="8dp"
        android:gravity="start"
        android:orientation="horizontal">

        <TextView
            android:id="@+id/outputEndTime"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="20dp"
            android:layout_marginTop="15dp"
            android:fontFamily="@font/montserrat"
            android:text="End Time : "
            android:textColor="@color/black"
            android:textSize="16sp" />

        <TextView
            android:id="@+id/viewEndTime"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="15dp"
            android:fontFamily="@font/montserrat"
            android:text="2 Months"
            android:textColor="@color/black"
            android:textSize="16sp" />
    </LinearLayout>

    <LinearLayout
        android:id="@+id/skillLayouting"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/locationLayout"
        android:layout_marginTop="15dp"
        android:layout_marginBottom="20dp"
        android:baselineAligned="true"
        android:orientation="horizontal">

        <TextView
            android:id="@+id/textCost"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="20dp"
            android:layout_marginEnd="5dp"
            android:fontFamily="@font/montserrat"
            android:gravity="top"
            android:text="Total Cost : "
            android:textColor="@color/black"
            android:textSize="16sp" />
```

```

        <TextView
            android:id="@+id/text_salary"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:fontFamily="@font/montserrat"
            android:text="Rp400.000,00/8 hours"
            android:textColor="@color/black"
            android:textSize="16sp" />
    </LinearLayout>

</LinearLayout>

</RelativeLayout>

</androidx.cardview.widget.CardView>

<Button
    android:id="@+id/buttonCancelBook"
    android:layout_width="match_parent"
    android:layout_height="60dp"
    android:layout_below="@+id/cardViewHistoryBD"
    android:layout_gravity="center"
    android:layout_marginStart="32dp"
    android:layout_marginTop="15dp"
    android:layout_marginEnd="32dp"
    android:layout_marginBottom="15dp"
    android:backgroundTint="#B22222"
    android:fontFamily="@font/montserratsemibold"
    android:text="Cancel Book"
    android:textSize="20sp"
    android:textStyle="bold" />
</LinearLayout>
</androidx.core.widget.NestedScrollView>
</androidx.coordinatorlayout.widget.CoordinatorLayout>

```

[File XML activity_detail_history_b_m.xml]

```

<androidx.coordinatorlayout.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".view.detailhistory.DetailHistoryBMActivity">

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/appToolbar"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar">

        <com.google.android.material.appbar.CollapsingToolbarLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            app:layout_scrollFlags="scroll|exitUntilCollapsed">

            <androidx.appcompat.widget.Toolbar
                android:id="@+id/toolbar"
                android:layout_width="match_parent"
                android:layout_height="?attr/actionBarSize"
                android:background="?attr/colorPrimary"
                android:elevation="4dp"
                app:layout_scrollFlags="scroll|enterAlways">

                <TextView
                    android:id="@+id/toolbar_title"

```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:fontFamily="@font/montserratsemibold"
        android:text="Booking's Details"
        android:textColor="@color/white"
        android:textSize="18sp" />

    </androidx.appcompat.widget.Toolbar>
</com.google.android.material.appbar.CollapsingToolbarLayout>
</com.google.android.material.appbar.AppBarLayout>

<androidx.core.widget.NestedScrollView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="fill_vertical"
    android:fillViewport="true"
    app:layout_behavior="@string/appbar_scrolling_view_behavior">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <androidx.cardview.widget.CardView
            android:id="@+id/cardViewHistoryBD"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginStart="16dp"
            android:layout_marginTop="16dp"
            android:layout_marginEnd="16dp"
            android:layout_marginBottom="16dp"
            app:cardCornerRadius="20dp"
            app:cardElevation="8dp"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@+id/imageViewNanny">

            <RelativeLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:padding="16dp">

                <LinearLayout
                    android:id="@+id/idText"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:layout_marginStart="20dp"
                    android:layout_marginTop="8dp"
                    android:gravity="start"
                    android:orientation="horizontal">

                    <TextView
                        android:id="@+id/textID"
                        android:layout_width="wrap_content"
                        android:layout_height="wrap_content"
                        android:layout_marginBottom="8dp"
                        android:fontFamily="@font/montserratsemibold"
                        android:text="ID Booking:"
                        android:textColor="@color/black"
                        android:textSize="15sp"
                        android:textStyle="bold" />

                    <TextView
                        android:id="@+id/textViewID"
                        android:layout_width="wrap_content"
                        android:layout_height="wrap_content"
                        android:layout_marginStart="5dp"

```

```
        android:layout_marginBottom="8dp"
        android:fontFamily="@font/montserratsemibold"
        android:text="A4352hydsas89"
        android:textColor="@color/black"
        android:textSize="15sp"
        android:textStyle="bold" />
    
```

```
</LinearLayout>

<androidx.cardview.widget.CardView
    android:id="@+id/profileNanny"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/idText"
    android:layout_margin="16dp"
    app:cardBackgroundColor="@color/main"
    app:cardCornerRadius="30dp"
    app:cardElevation="10dp">
```

```
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center_vertical"
        android:orientation="horizontal"
        android:paddingTop="16dp"
        android:paddingBottom="16dp">
```

```
        <de.hdodenhof.circleimageview.CircleImageView
            android:id="@+id/image_profile"
            android:layout_width="100dp"
            android:layout_height="100dp"
            android:layout_marginStart="10dp"
            android:scaleType="centerCrop"
            android:src="@drawable/nanny_placeholder" />
```

```
        <LinearLayout
            android:layout_width="0dp"
            android:layout_height="match_parent"
            android:layout_weight="1"
            android:gravity="center_vertical|start"
            android:orientation="vertical"
            android:paddingStart="16dp"
            android:paddingEnd="16dp">
```

```
            <TextView
                android:id="@+id/text_name"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:fontFamily="@font/montserratsemibold"
                android:text="Putri Utami"
                android:textColor="@color/black"
                android:textSize="18sp"
                android:textStyle="bold" />
```

```
            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginTop="8dp"
                android:gravity="center_vertical">
```

```
                <ImageView
                    android:layout_width="20dp"
                    android:layout_height="20dp"
                    android:src="@drawable/ic_calendar" />
```

```
                <TextView
                    android:id="@+id/text_email"
                    android:layout_width="wrap_content"
```

```
        android:layout_height="wrap_content"
        android:layout_marginStart="4dp"

    android:fontFamily="@font/montserratsemibold"
            android:text="monthly"
            android:textColor="@color/black"
            android:textSize="16sp" />
    </LinearLayout>

        </LinearLayout>
    </LinearLayout>
</androidx.cardview.widget.CardView>

<LinearLayout
    android:id="@+id/formBooking"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/profileNanny"
    android:orientation="vertical">

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="8dp"
        android:gravity="start"
        android:orientation="horizontal">
        <TextView
            android:id="@+id/outputBookingDuration"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="20dp"
            android:layout_marginTop="15dp"
            android:fontFamily="@font/montserrat"
            android:text="Booking Duration : "
            android:textColor="@color/black"
            android:textSize="16sp" />

        <TextView
            android:id="@+id/viewBookingDuration"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="15dp"
            android:fontFamily="@font/montserrat"
            android:text="2 Months"
            android:textColor="@color/black"
            android:textSize="16sp" />
    </LinearLayout>
    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="8dp"
        android:gravity="start"
        android:orientation="horizontal">

        <TextView
            android:id="@+id/outputStartDate"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="20dp"
            android:layout_marginTop="15dp"
            android:fontFamily="@font/montserrat"
            android:text="Start Date : "
            android:textColor="@color/black"
            android:textSize="16sp" />

        <TextView
            android:id="@+id/viewStartDate"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="20dp"
            android:layout_marginTop="15dp"
            android:fontFamily="@font/montserrat"
            android:text="2023-08-15"
            android:textColor="@color/black"
            android:textSize="16sp" />
    </LinearLayout>

```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="15dp"
        android:fontFamily="@font/montserrat"
        android:text="2 Months"
        android:textColor="@color/black"
        android:textSize="16sp" />
    </LinearLayout>

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="8dp"
        android:gravity="start"
        android:orientation="horizontal">

        <TextView
            android:id="@+id/outputEndDate"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="20dp"
            android:layout_marginTop="15dp"
            android:fontFamily="@font/montserrat"
            android:text="End Date : "
            android:textColor="@color/black"
            android:textSize="16sp" />

        <TextView
            android:id="@+id/viewEndDate"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="15dp"
            android:fontFamily="@font/montserrat"
            android:text="2 Months"
            android:textColor="@color/black"
            android:textSize="16sp" />
    </LinearLayout>

    <LinearLayout
        android:id="@+id/skillLayouting"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/locationLayout"
        android:layout_marginTop="15dp"
        android:layout_marginBottom="20dp"
        android:baselineAligned="true"
        android:orientation="horizontal">

        <TextView
            android:id="@+id/textCost"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="20dp"
            android:layout_marginEnd="5dp"
            android:fontFamily="@font/montserrat"
            android:gravity="top"
            android:text="Total Cost : "
            android:textColor="@color/black"
            android:textSize="16sp" />

        <TextView
            android:id="@+id/text_salary"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:fontFamily="@font/montserrat"
            android:text="Rp400.000,00/8 hours"
            android:textColor="@color/black"
```

```

        android:textSize="16sp" />
    </LinearLayout>

    </RelativeLayout>

</androidx.cardview.widget.CardView>

<Button
    android:id="@+id/buttonCancelBook"
    android:layout_width="match_parent"
    android:layout_height="60dp"
    android:layout_below="@+id/cardViewHistoryBM"
    android:layout_gravity="center"
    android:layout_marginStart="32dp"
    android:layout_marginTop="15dp"
    android:layout_marginEnd="32dp"
    android:layout_marginBottom="15dp"
    android:backgroundTint="#B22222"
    android:fontFamily="@font/montserratsemibold"
    android:text="Cancel Book"
    android:textSize="20sp"
    android:textStyle="bold" />
</LinearLayout>
</androidx.core.widget.NestedScrollView>
</androidx.coordinatorlayout.widget.CoordinatorLayout>
```

Penjelasan:

Secara keseluruhan, fitur Booking, History, dan Chat dalam aplikasi FamCare saling terintegrasi untuk memungkinkan pengguna melakukan pemesanan pengasuh, melihat riwayat pemesanan, dan berkomunikasi dengan pengasuh.

Fungsi utama dari fitur Booking adalah memungkinkan pengguna untuk melakukan pemesanan pengasuh anak berdasarkan waktu dan durasi yang mereka tentukan. Pada dasarnya, fitur ini terdiri dari tiga alur penting: pengisian dan pengambilan data pada formulir pemesanan (BookDailyActivity dan BookMonthlyActivyty), validasi waktu pemesanan, serta penyimpanan data di Firestore. Formulir pemesanan memungkinkan pengguna untuk mengisi detail seperti tanggal pemesanan, durasi pemesanan, waktu mulai dan berakhir, serta total harga. Pengguna juga dapat memilih dari daftar pengasuh yang tersedia. Setelah detail diisi, sistem akan melakukan validasi untuk memastikan bahwa waktu yang dipilih tidak bertabrakan dengan pemesanan yang sudah ada. Fungsi checkAvailability digunakan untuk memeriksa ketersediaan pengasuh dengan melakukan query ke Firestore untuk mencari pemesanan yang sudah ada dalam rentang waktu yang dipilih. Jika validasi berhasil, data pemesanan disimpan di Firestore dalam koleksi "Bookings" melalui fungsi saveBooking. Fungsi ini menambahkan data pemesanan ke Firestore dan menampilkan pesan sukses atau gagal. Terdapat juga logika untuk menghitung harga total berdasarkan durasi dan tarif pengasuh yang dipilih oleh pengguna. Fungsi calculateTotalPrice mengambil input durasi dan tarif per jam untuk menghitung total harga yang harus dibayar.

Fungsi History menyediakan pengguna dengan riwayat pemesanan sebelumnya, terbagi menjadi dua bagian: harian dan bulanan. HistoryActivity

menampilkan antarmuka dengan dua tab menggunakan ViewPager2 dan TabLayout, memungkinkan pengguna beralih antara riwayat harian dan bulanan. HistoryBDFragment menangani riwayat harian, mengambil data dari Firestore dan menampilkannya dalam RecyclerView menggunakan BookingAdapter. Data pemesanan harian termasuk nama pengasuh, tanggal, dan jam pemesanan. HistoryBMFragment menangani riwayat bulanan dengan cara yang serupa, menampilkan data pemesanan bulanan.

Fungsi Chat memungkinkan pengguna berkomunikasi langsung dengan pengasuh secara real-time melalui Firebase Realtime Database. ChatActivity menampilkan antarmuka percakapan di mana pengguna dapat mengirim dan menerima pesan. Pesan diambil dan ditampilkan dalam RecyclerView menggunakan ChatAdapter, sementara ChatActivity mengelola pengiriman pesan ke Firebase. Selain itu, detail pengasuh dimuat dari Firestore untuk ditampilkan dalam antarmuka percakapan. Fungsi ini memastikan komunikasi langsung dan cepat antara pengguna dan pengasuh.

2.2.7 Facilities

[File Activity FacilitiesAcitivity.kt]

```
package com.app.famcare.view.facilities

import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import com.app.famcare.R
import com.app.famcare.databinding.ActivityFacilitiesBinding
import com.app.famcare.view.history.HistoryActivity
import com.app.famcare.view.main.MainActivity
import com.app.famcare.view.maps.DaycareMapsActivity
import com.app.famcare.view.maps.ElderlyMapsActivity
import com.app.famcare.view.profile.ProfileActivity
import com.google.android.material.bottomnavigation.BottomNavigationView

class FacilitiesActivity : AppCompatActivity() {
    private lateinit var binding: ActivityFacilitiesBinding
    private lateinit var bottomNavigationView: BottomNavigationView

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityFacilitiesBinding.inflate(layoutInflater)
        setContentView(binding.root)

        setSupportActionBar(binding.toolbar)
        supportActionBar?.title = ""

        val bottomNavigation: BottomNavigationView =
            findViewById(R.id.bottom_navigation)
        bottomNavigation.selectedItemId = R.id.page_3

        binding.cardViewDaycare.setOnClickListener {
            val intent = Intent(this, DaycareMapsActivity::class.java)
            startActivity(intent)
        }

        binding.cardViewElderlycare.setOnClickListener {
            val intent = Intent(this, ElderlyMapsActivity::class.java)
            startActivity(intent)
        }
    }
}
```

```

        }

        binding.bottomNavigation.setOnNavigationItemSelected { item ->
            when (item.itemId) {
                R.id.page_1 -> {
                    val intent = Intent(this, MainActivity::class.java)
                    startActivity(intent)
                    true
                }
                R.id.page_2 -> {
                    val intent = Intent(this, HistoryActivity::class.java)
                    startActivity(intent)
                    true
                }
                R.id.page_3 -> true
                R.id.page_4 -> {
                    val intent = Intent(this, ProfileActivity::class.java)
                    startActivity(intent)
                    true
                }
                else -> false
            }
        }
    }
}

```

[File Activity DaycareMapsActivity.kt]

```

package com.app.famcare.view.maps

import android.Manifest
import android.content.Intent
import android.content.pm.PackageManager
import android.location.Address
import android.location.Geocoder
import android.location.Location
import android.os.Bundle
import android.util.Log
import android.view.View
import android.widget.ProgressBar
import android.widget.TextView
import androidx.appcompat.app.AlertDialog
import androidx.appcompat.app.AppCompatActivity
import androidx.cardview.widget.CardView
import androidx.core.app.ActivityCompat
import androidx.core.content.ContextCompat
import androidx.recyclerview.widget.DividerItemDecoration
import androidx.recyclerview.widget.LinearLayoutManager
import androidx.recyclerview.widget.RecyclerView
import com.app.famcare.R
import com.app.famcare.databinding.ActivityMapsBinding
import com.google.android.gms.location.*
import com.google.firebase.firestore.FirebaseFirestore
import java.io.IOException
import java.math.RoundingMode
import java.text.DecimalFormat
import java.util.*

class DaycareMapsActivity : AppCompatActivity() {

    private lateinit var binding: ActivityMapsBinding
    private lateinit var fusedLocationClient: FusedLocationProviderClient

```

```

private lateinit var recyclerView: RecyclerView
private lateinit var adapter: DaycareAdapter
private lateinit var tvUserLocation: TextView
private lateinit var progressBar: ProgressBar
private lateinit var locationRequest: LocationRequest
private lateinit var locationCallback: LocationCallback

companion object {
    private const val LOCATION_PERMISSION_REQUEST_CODE = 1
    private const val TAG = "DaycareMapsActivity"
}

override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    binding = ActivityMapsBinding.inflate(layoutInflater)
    setContentView(binding.root)

    setSupportActionBar(binding.toolbar)
    supportActionBar?.apply {
        title = ""
        setDisplayHomeAsUpEnabled(true)
        setDisplayShowHomeEnabled(true)
    }

    binding.toolbar.setNavigationOnClickListener {
        onBackPressed()
    }
    tvUserLocation = findViewById(R.id.tv_UserLocation)
    progressBar = findViewById(R.id.progressBar)
    recyclerView = findViewById(R.id.recyclerView)
    recyclerView.layoutManager = LinearLayoutManager(this)

    val dividerItemDecoration = DividerItemDecoration(this,
    LinearLayoutManager.VERTICAL)
    recyclerView.addItemDecoration(dividerItemDecoration)

    adapter = DaycareAdapter(emptyList()) { websiteURL ->
        val intent = Intent(this, WebViewActivity::class.java)
        intent.putExtra("WEBSITE_URL", websiteURL)
        startActivity(intent)
    }
    recyclerView.adapter = adapter

    fusedLocationClient =
    LocationServices.getFusedLocationProviderClient(this)

    locationRequest = LocationRequest.create().apply {
        interval = 10000
        fastestInterval = 5000
        priority = LocationRequest.PRIORITY_HIGH_ACCURACY
    }

    locationCallback = object : LocationCallback() {
        override fun onLocationResult(locationResult: LocationResult?) {
            locationResult ?: return
            for (location in locationResult.locations) {
                Log.d(TAG, "User location: Lat=${location.latitude}, "
                Lng=${location.longitude}")
                getUserAddress(location)
                loadDaycareData(location)
            }
        }
    }
}

if (ContextCompat.checkSelfPermission(
    this, Manifest.permission.ACCESS_FINE_LOCATION
) == PackageManager.PERMISSION_GRANTED

```

```

    )
    enableMyLocation()
} else {
    ActivityCompat.requestPermissions(
        this,
        arrayOf(Manifest.permission.ACCESS_FINE_LOCATION),
        LOCATION_PERMISSION_REQUEST_CODE
    )
}

val locationLayout = findViewById<CardView>(R.id.locationLayout)
locationLayout.setOnClickListener {
    if (ContextCompat.checkSelfPermission(
            this, Manifest.permission.ACCESS_FINE_LOCATION
        ) == PackageManager.PERMISSION_GRANTED
    ) {
        fusedLocationClient.lastLocation.addOnSuccessListener { location: Location? ->
            if (location != null) {
                Log.d(
                    TAG,
                    "User location: Lat=${location.latitude}, "
                )
                showLocationPopup(location)
            } else {
                showLocationPopup(null)
            }
        }.addOnFailureListener { exception ->
            Log.e(TAG, "Error obtaining location", exception)
            showLocationPopup(null)
        }
    } else {
        ActivityCompat.requestPermissions(
            this,
            arrayOf(Manifest.permission.ACCESS_FINE_LOCATION),
            LOCATION_PERMISSION_REQUEST_CODE
        )
    }
}
}

private fun showLocationPopup(location: Location?) {
    val message = if (location != null) {
        val geocoder = Geocoder(this, Locale.getDefault())
        try {
            val addresses: List<Address>? =
                geocoder.getFromLocation(location.latitude,
                location.longitude, 1)
            addresses?.get(0)?.getAddressLine(0) ?: "Unknown location"
        } catch (e: IOException) {
            Log.e(TAG, "Geocoder failed", e)
            "Unknown location"
        }
    } else {
        "Location not available"
    }

    AlertDialog.Builder(this).setTitle("Your Location").setMessage(message)
        .setPositiveButton("OK", null).show()
}

private fun enableMyLocation() {
    if (ActivityCompat.checkSelfPermission(
            this, Manifest.permission.ACCESS_FINE_LOCATION
        ) != PackageManager.PERMISSION_GRANTED
    ) {

```

```

        ActivityCompat.requestPermissions(
            this,
            arrayOf(Manifest.permission.ACCESS_FINE_LOCATION),
            LOCATION_PERMISSION_REQUEST_CODE
        )
        return
    }

    fusedLocationClient.lastLocation.addOnSuccessListener { location: Location? ->
        if (location != null) {
            Log.d(TAG, "User location: Lat=${location.latitude}, Lng=${location.longitude}")
            getUserAddress(location)
            loadDaycareData(location)
        } else {
            fusedLocationClient.requestLocationUpdates(locationRequest,
                locationCallback, null)
        }
    }.addOnFailureListener { exception ->
        Log.e(TAG, "Error obtaining location", exception)
        tvUserLocation.text = "Location not available"
    }
}

private fun getUserAddress(location: Location) {
    val geocoder = Geocoder(this, Locale.getDefault())
    try {
        val addresses: List<Address>? =
            geocoder.getFromLocation(location.latitude, location.longitude, 1)
        val address = addresses?.get(0)?.getAddressLine(0) ?: "Unknown
location"
        if (address == "Unnamed Road") {
            Log.d(TAG, "Geocoder returned 'Unnamed Road'")
        }
        tvUserLocation.text = address
    } catch (e: IOException) {
        Log.e(TAG, "Geocoder failed", e)
        tvUserLocation.text = "Unknown location"
    }
}

private fun loadDaycareData(userLocation: Location) {
    progressBar.visibility = View.VISIBLE

    val firestore = FirebaseFirestore.getInstance()
    val daycareCollection = firestore.collection("Daycare")

    daycareCollection.get().addOnSuccessListener { result ->
        val daycares = mutableListOf<Daycare>()
        for (document in result) {
            val geoPoint = document.getGeoPoint("Coordinate")
            val locationName = document.getString("Location")
            val daycareName = document.getString("Name")
            val photoURL = document.getString("PhotoURL")
            val websiteURL = document.getString("WebsiteURL")

            if (geoPoint != null && locationName != null && daycareName != null && photoURL != null && websiteURL != null) {
                val daycareLocation = Location("").apply {
                    latitude = geoPoint.latitude
                    longitude = geoPoint.longitude
                }
                val distanceInMeters =
userLocation.distanceTo(daycareLocation)
                val distanceInKilometers = distanceInMeters / 1000
            }
        }
    }
}

```

```

        val df = DecimalFormat("#.#").apply { roundingMode =
RoundingMode.HALF_UP }
        val roundedDistanceInKilometers =
df.format(distanceInKilometers).toDouble()

        daycares.add(
            Daycare(
                daycareName,
                locationName,
                photoURL,
                geoPoint,
                roundedDistanceInKilometers,
                websiteURL
            )
        )
        Log.d(
            TAG,
            "Daycare: $daycareName, Lat=${geoPoint.latitude}, Lng=${geoPoint.longitude}, Distance: $roundedDistanceInKilometers km"
        )
    } else {
        Log.d(TAG, "Document is missing required fields")
    }
}

progressBar.visibility = View.GONE
daycares.sortBy { it.distanceFromUser }
adapter.updateDaycares(daycares)
}.addOnFailureListener { exception ->
Log.e(TAG, "get failed with ", exception)
progressBar.visibility = View.GONE
}
}

override fun onRequestPermissionsResult(
    requestCode: Int, permissions: Array<String>, grantResults: IntArray
) {
    super.onRequestPermissionsResult(requestCode, permissions, grantResults)
    if (requestCode == LOCATION_PERMISSION_REQUEST_CODE) {
        if ((grantResults.isNotEmpty() && grantResults[0] ==
PackageManager.PERMISSION_GRANTED)) {
            enableMyLocation()
        } else {
            tvUserLocation.text = "Location not enabled"
            loadDaycareDataWithoutUserLocation()
        }
    }
}

private fun loadDaycareDataWithoutUserLocation() {
    progressBar.visibility = View.VISIBLE

    val firestore = FirebaseFirestore.getInstance()
    val daycareCollection = firestore.collection("Daycare")

    daycareCollection.get().addOnSuccessListener { result ->
        val daycares = mutableListOf<Daycare>()
        for (document in result) {
            val geoPoint = document.getGeoPoint("Coordinate")
            val locationName = document.getString("Location")
            val daycareName = document.getString("Name")
            val photoURL = document.getString("PhotoURL")
            val websiteURL = document.getString("WebsiteURL")

            if (geoPoint != null && locationName != null && daycareName != null && photoURL != null && websiteURL != null) {
                daycares.add(

```

```

        Daycare(
            daycareName, locationName, photoURL, geoPoint, 0.0,
            websiteURL
        )
    }
} else {
    Log.d(TAG, "Document is missing required fields")
}
}
adapter.updateDaycares(daycares)
progressBar.visibility = View.GONE
}.addOnFailureListener { exception ->
Log.d(TAG, "get failed with ", exception)
progressBar.visibility = View.GONE
}
}

override fun onSupportNavigateUp(): Boolean {
    onBackPressed()
    return true
}
}

```

[File Activity ElderlyMapsActivity.kt]

```

package com.app.famcare.view.maps

import android.Manifest
import android.content.Intent
import android.content.pm.PackageManager
import android.location.Address
import android.location.Geocoder
import android.location.Location
import android.os.Bundle
import android.util.Log
import android.view.View
import android.widget.ProgressBar
import android.widget.TextView
import androidx.appcompat.app.AlertDialog
import androidx.appcompat.app.AppCompatActivity
import androidx.cardview.widget.CardView
import androidx.core.app.ActivityCompat
import androidx.core.content.ContextCompat
import androidx.recyclerview.widget.DividerItemDecoration
import androidx.recyclerview.widget.LinearLayoutManager
import androidx.recyclerview.widget.RecyclerView
import com.app.famcare.R
import com.app.famcare.databinding.ActivityElderlyMapsBinding
import com.google.android.gms.location.*
import com.google.firebase.firestore.FirebaseFirestore
import java.io.IOException
import java.math.RoundingMode
import java.text.DecimalFormat
import java.util.*

class ElderlyMapsActivity : AppCompatActivity() {

    private lateinit var binding: ActivityElderlyMapsBinding
    private lateinit var fusedLocationClient: FusedLocationProviderClient
    private lateinit var recyclerView: RecyclerView
    private lateinit var adapter: ElderlyCareAdapter
    private lateinit var tvUserLocation: TextView
    private lateinit var progressBar: ProgressBar
    private lateinit var locationRequest: LocationRequest
    private lateinit var locationCallback: LocationCallback

    companion object {

```

```

        private const val LOCATION_PERMISSION_REQUEST_CODE = 1
        private const val TAG = "ElderlyMapsActivity"
    }

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityElderlyMapsBinding.inflate(layoutInflater)
        setContentView(binding.root)

        setSupportActionBar(binding.toolbar)
        supportActionBar?.apply {
            title = ""
            setDisplayHomeAsUpEnabled(true)
            setDisplayShowHomeEnabled(true)
        }

        binding.toolbar.setNavigationOnClickListener {
            onBackPressed()
        }

        tvUserLocation = findViewById(R.id.tv_UserLocation)
        progressBar = findViewById(R.id.progressBar)
        recyclerView = findViewById(R.id.recyclerView)
        recyclerView.layoutManager = LinearLayoutManager(this)

        val dividerItemDecoration = DividerItemDecoration(this,
        LinearLayoutManager.VERTICAL)
        recyclerView.addItemDecoration(dividerItemDecoration)

        adapter = ElderlyCareAdapter(emptyList()) { websiteURL ->
            val intent = Intent(this, WebViewActivity::class.java)
            intent.putExtra("WEBSITE_URL", websiteURL)
            startActivity(intent)
        }
        recyclerView.adapter = adapter

        fusedLocationClient =
        LocationServices.getFusedLocationProviderClient(this)

        locationRequest = LocationRequest.create().apply {
            interval = 10000
            fastestInterval = 5000
            priority = LocationRequest.PRIORITY_HIGH_ACCURACY
        }

        locationCallback = object : LocationCallback() {
            override fun onLocationResult(locationResult: LocationResult?) {
                locationResult ?: return
                for (location in locationResult.locations) {
                    Log.d(TAG, "User location: Lat=${location.latitude}, "
                    "Long=${location.longitude}")
                    getUserAddress(location)
                    loadDaycareData(location)
                }
            }
        }

        if (ContextCompat.checkSelfPermission(this,
        Manifest.permission.ACCESS_FINE_LOCATION)
            == PackageManager.PERMISSION_GRANTED
        ) {
            enableMyLocation()
        } else {
            ActivityCompat.requestPermissions(
                this,
                arrayOf(Manifest.permission.ACCESS_FINE_LOCATION),
                LOCATION_PERMISSION_REQUEST_CODE
            )
        }
    }
}

```

```

        )

    }

    val locationLayout = findViewById<CardView>(R.id.locationLayout)
    locationLayout.setOnClickListener {
        if (ContextCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION)
            == PackageManager.PERMISSION_GRANTED
        ) {
            fusedLocationClient.lastLocation.addOnSuccessListener { location:
Location? ->
                if (location != null) {
                    Log.d(
                        TAG,
                        "User location: Lat=${location.latitude},
Lng=${location.longitude}"
                    )
                    showLocationPopup(location)
                } else {
                    showLocationPopup(null)
                }
            }.addOnFailureListener { exception ->
                Log.e(TAG, "Error obtaining location", exception)
                showLocationPopup(null)
            }
        } else {
            ActivityCompat.requestPermissions(
                this,
                arrayOf(Manifest.permission.ACCESS_FINE_LOCATION),
                LOCATION_PERMISSION_REQUEST_CODE
            )
        }
    }
}

private fun showLocationPopup(location: Location?) {
    val message = if (location != null) {
        val geocoder = Geocoder(this, Locale.getDefault())
        try {
            val addresses: List<Address>? =
                geocoder.getFromLocation(location.latitude,
location.longitude, 1)
            addresses?.get(0)?.getAddressLine(0) ?: "Unknown location"
        } catch (e: IOException) {
            Log.e(TAG, "Geocoder failed", e)
            "Unknown location"
        }
    } else {
        "Location not available"
    }

    AlertDialog.Builder(this)
        .setTitle("Your Location")
        .setMessage(message)
        .setPositiveButton("OK", null)
        .show()
}

private fun enableMyLocation() {
    if (ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION)
        != PackageManager.PERMISSION_GRANTED
    ) {
        ActivityCompat.requestPermissions(
            this,
            arrayOf(Manifest.permission.ACCESS_FINE_LOCATION),
            LOCATION_PERMISSION_REQUEST_CODE
        )
    }
}

```

```

        )
        return
    }

    fusedLocationClient.lastLocation.addOnSuccessListener { location: Location? ->
        if (location != null) {
            Log.d(TAG, "User location: Lat=${location.latitude}, Lng=${location.longitude}")
            getUserAddress(location)
            loadDaycareData(location)
        } else {
            fusedLocationClient.requestLocationUpdates(locationRequest, locationCallback, null)
        }
    }.addOnFailureListener { exception ->
        Log.e(TAG, "Error obtaining location", exception)
        tvUserLocation.text = "Location not available"
    }
}

private fun getUserAddress(location: Location) {
    val geocoder = Geocoder(this, Locale.getDefault())
    try {
        val addresses: List<Address>? =
            geocoder.getFromLocation(location.latitude, location.longitude, 1)
        val address = addresses?.get(0)?.getAddressLine(0) ?: "Unknown $location"
        if (address == "Unnamed Road") {
            Log.d(TAG, "Geocoder returned 'Unnamed Road'")
        }
        tvUserLocation.text = address
    } catch (e: IOException) {
        Log.e(TAG, "Geocoder failed", e)
        tvUserLocation.text = "Unknown location"
    }
}

private fun loadDaycareData(userLocation: Location) {
    progressBar.visibility = View.VISIBLE

    val firestore = FirebaseFirestore.getInstance()
    val daycareCollection = firestore.collection("ElderlyCare")

    daycareCollection.get()
        .addOnSuccessListener { result ->
            val daycares = mutableListOf<ElderlyCare>()
            for (document in result) {
                val geoPoint = document.getGeoPoint("Coordinate")
                val locationName = document.getString("Location")
                val daycareName = document.getString("Name")
                val photoURL = document.getString("PhotoURL")
                val websiteURL = document.getString("WebsiteURL")

                if (geoPoint != null && locationName != null && daycareName != null && photoURL != null && websiteURL != null) {
                    val daycareLocation = Location("").apply {
                        latitude = geoPoint.latitude
                        longitude = geoPoint.longitude
                    }
                    val distanceInMeters =
                        userLocation.distanceTo(daycareLocation)
                    val distanceInKilometers = distanceInMeters / 1000
                    val df = DecimalFormat("#.#").apply { roundingMode =
RoundingMode.HALF_UP }
                    val roundedDistanceInKilometers =
                        df.format(distanceInKilometers).toDouble()
                }
            }
        }
}

```

```

        daycares.add(
            ElderlyCare(
                daycareName,
                locationName,
                photoURL,
                geoPoint,
                roundedDistanceInKilometers,
                websiteURL
            )
        )
        Log.d(
            TAG,
            "Daycare: $daycareName, Lat=${geoPoint.latitude}, Lng=${geoPoint.longitude}, Distance: $roundedDistanceInKilometers km"
        )
    } else {
        Log.d(TAG, "Document is missing required fields")
    }
}

progressBar.visibility = View.GONE
daycares.sortBy { it.distanceFromUser }
adapter.updateDaycares(daycares)
}
.addOnFailureListener { exception ->
    Log.e(TAG, "get failed with ", exception)
    progressBar.visibility = View.GONE
}
}
}

override fun onRequestPermissionsResult(
    requestCode: Int,
    permissions: Array<String>,
    grantResults: IntArray
) {
    super.onRequestPermissionsResult(requestCode, permissions, grantResults)
    if (requestCode == LOCATION_PERMISSION_REQUEST_CODE) {
        if ((grantResults.isNotEmpty() && grantResults[0] ==
PackageManager.PERMISSION_GRANTED)) {
            enableMyLocation()
        } else {
            tvUserLocation.text = "Location not enabled"
            loadDaycareDataWithoutUserLocation()
        }
    }
}

private fun loadDaycareDataWithoutUserLocation() {
    progressBar.visibility = View.VISIBLE

    val firestore = FirebaseFirestore.getInstance()
    val daycareCollection = firestore.collection("ElderlyCare")

    daycareCollection.get()
        .addOnSuccessListener { result ->
            val daycares = mutableListOf<ElderlyCare>()
            for (document in result) {
                val geoPoint = document.getGeoPoint("Coordinate")
                val locationName = document.getString("Location")
                val daycareName = document.getString("Name")
                val photoURL = document.getString("PhotoURL")
                val websiteURL = document.getString("WebsiteURL")

                if (geoPoint != null && locationName != null && daycareName != null && photoURL != null && websiteURL != null) {
                    daycares.add(

```

```

        ElderlyCare(
            daycareName,
            locationName,
            photoURL,
            geoPoint,
            0.0,
            websiteURL
        )
    )
} else {
    Log.d(TAG, "Document is missing required fields")
}
}
adapter.updateDaycares(daycares)
progressBar.visibility = View.GONE
}
.addOnFailureListener { exception ->
    Log.d(TAG, "get failed with ", exception)
    progressBar.visibility = View.GONE
}
}
}

override fun onSupportNavigateUp(): Boolean {
    onBackPressed()
    return true
}
)

```

[File adapter DaycareAdapter.kt]

```

package com.app.famcare.view.maps

import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.ImageView
import android.widget.TextView
import androidx.recyclerview.widget.RecyclerView
import com.app.famcare.R
import com.bumptech.glide.Glide
import com.google.firebase.firestore.GeoPoint

data class Daycare(
    val name: String,
    val location: String,
    val photoURL: String,
    val coordinat: GeoPoint,
    val distanceFromUser: Double,
    val websiteURL: String
)

class DaycareAdapter(
    private var daycares: List<Daycare>, private val itemClickListener: (String) -> Unit
) : RecyclerView.Adapter<DaycareAdapter.DaycareViewHolder>() {

    override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): DaycareViewHolder {
        val itemView =
            LayoutInflater.from(parent.context).inflate(R.layout.item_daycare, parent, false)
        return DaycareViewHolder(itemView)
    }

    override fun onBindViewHolder(holder: DaycareViewHolder, position: Int) {
        val daycare = daycares[position]
        holder.nameTextView.text = daycare.name
    }
}

```

```

        holder.locationTextView.text = daycare.location
        holder.distanceTextView.text = if (daycare.distanceFromUser == 0.0) {
            "-"
        } else {
            "$${daycare.distanceFromUser} km"
        }

Glide.with(holder.itemView.context).load(daycare.photoURL).into(holder.photoImage
view)

        holder.itemView.setOnClickListener {
            itemClickListener(daycare.websiteURL)
        }
    }

    override fun getItemCount() = daycares.size

    fun updateDaycares(newDaycares: List<Daycare>) {
        daycares = newDaycares
        notifyDataSetChanged()
    }

    class DaycareViewHolder(itemView: View) : RecyclerView.ViewHolder(itemView) {
        val nameTextView: TextView = itemView.findViewById(R.id.daycare_name)
        val locationTextView: TextView =
itemView.findViewById(R.id.daycare_location)
        val distanceTextView: TextView =
itemView.findViewById(R.id.tvDistancetoUser)
        val photoImageView: ImageView = itemView.findViewById(R.id.daycare_image)
    }
}

```

[File adapter ElderlyCareAdapter.kt]

```

package com.app.famcare.view.maps

import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.ImageView
import android.widget.TextView
import androidx.recyclerview.widget.RecyclerView
import com.app.famcare.R
import com.bumptech.glide.Glide
import com.google.firebase.firestore.GeoPoint

data class ElderlyCare(
    val name: String,
    val location: String,
    val photoURL: String,
    val coordinat: GeoPoint,
    val distanceFromUser: Double,
    val websiteURL: String
)

class ElderlyCareAdapter(
    private var daycares: List<ElderlyCare>, private val itemClickListener:
(String) -> Unit
) : RecyclerView.Adapter<ElderlyCareAdapter.DaycareViewHolder>() {

    override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): DaycareViewHolder {
        val itemView =
            LayoutInflater.from(parent.context).inflate(R.layout.item_daycare,
parent, false)
        return DaycareViewHolder(itemView)
    }
}

```

```

        }

    override fun onBindViewHolder(holder: DaycareViewHolder, position: Int) {
        val daycare = daycares[position]
        holder.nameTextView.text = daycare.name
        holder.locationTextView.text = daycare.location
        holder.distanceTextView.text = if (daycare.distanceFromUser == 0.0) {
            "_"
        } else {
            "${daycare.distanceFromUser} km"
        }
    }

    Glide.with(holder.itemView.context).load(daycare.photoURL).into(holder.photoImageView)

    holder.itemView.setOnClickListener {
        itemClickListener(daycare.websiteURL)
    }
}

override fun getItemCount() = daycares.size

fun updateDaycares(newDaycares: List<ElderlyCare>) {
    daycares = newDaycares
    notifyDataSetChanged()
}

class DaycareViewHolder(itemView: View) : RecyclerView.ViewHolder(itemView) {
    val nameTextView: TextView = itemView.findViewById(R.id.daycare_name)
    val locationTextView: TextView =
    itemView.findViewById(R.id.daycare_location)
    val distanceTextView: TextView =
    itemView.findViewById(R.id.tvDistanceToUser)
    val photoImageView: ImageView = itemView.findViewById(R.id.daycare_image)
}
)

```

[File activity WebViewActivity.kt]

```

package com.app.famcare.view.maps

import android.os.Bundle
import android.webkit.WebView
import android.webkit.WebViewClient
import androidx.appcompat.app.AppCompatActivity
import com.app.famcare.databinding.ActivityWebViewBinding

class WebViewActivity : AppCompatActivity() {
    private lateinit var binding: ActivityWebViewBinding

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityWebViewBinding.inflate(layoutInflater)
        setContentView(binding.root)

        setSupportActionBar(binding.toolbar)
        supportActionBar?.apply {
            setDisplayHomeAsUpEnabled(true)
            title = ""
        }
    }

    val webView: WebView = binding.webView
    webView.webViewClient = WebViewClient()
    webView.settings.javaScriptEnabled = true

    val websiteURL = intent.getStringExtra("WEBSITE_URL")

```

```

        if (websiteURL != null) {
            webView.loadUrl(websiteURL)
        }
    }

    override fun onSupportNavigateUp(): Boolean {
        onBackPressed()
        return true
    }
}

```

[File XML activity_facilities.xml]

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".view.facilities.FacilitiesActivity">

    <com.google.android.material.appbar.AppBarLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar">

        <androidx.appcompat.widget.Toolbar
            android:id="@+id/toolbar"
            android:layout_width="match_parent"
            android:layout_height="?attr/actionBarSize"
            android:background="?attr/colorPrimary"
            android:elevation="4dp"
            app:layout_scrollFlags="scroll|enterAlways">

            <TextView
                android:id="@+id/toolbar_title"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:fontFamily="@font/montserratsemibold"
                android:text="Explore Care Facilites"
                android:textColor="@color/white"
                android:textSize="18sp" />

        </androidx.appcompat.widget.Toolbar>

    </com.google.android.material.appbar.AppBarLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="8dp"
        android:orientation="vertical"
        android:padding="16dp"
        app:layout_behavior="@string/appbar_scrolling_view_behavior">

        <androidx.cardview.widget.CardView
            android:id="@+id/cardViewDaycare"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_margin="8dp"
            android:clickable="true"
            android:focusable="true"
            app:cardCornerRadius="18dp">

            <androidx.constraintlayout.widget.ConstraintLayout
                android:layout_width="match_parent"

```

```

        android:layout_height="wrap_content"
        android:padding="16dp">

        <ImageView
            android:id="@+id/imageViewDaycare"
            android:layout_width="80dp"
            android:layout_height="80dp"
            android:layout_marginEnd="8dp"
            android:src="@drawable/ic_daycare"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent" />

        <TextView
            android:id="@+id/textViewDaycare"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="20dp"
            android:fontFamily="@font/montserratsemibold"
            android:text="Day Care Facilities"
            android:textColor="@color/black"
            android:textSize="17sp"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintStart_toEndOf="@+id/imageViewDaycare"
            app:layout_constraintTop_toTopOf="parent"
            app:layout_constraintVertical_bias="0.506" />
    </androidx.constraintlayout.widget.ConstraintLayout>
</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView
    android:id="@+id/cardViewElderlycare"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:clickable="true"
    android:focusable="true"
    app:cardCornerRadius="18dp">

    <androidx.constraintlayout.widget.ConstraintLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="16dp">

        <ImageView
            android:id="@+id/imageViewElderlycare"
            android:layout_width="80dp"
            android:layout_height="80dp"
            android:layout_marginEnd="8dp"
            android:src="@drawable/ic_elderlycare"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent" />

        <TextView
            android:id="@+id/textViewElderlyCare"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="20dp"
            android:fontFamily="@font/montserratsemibold"
            android:text="Elderly Care Facilities"
            android:textColor="@color/black"
            android:textSize="17sp"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintStart_toEndOf="@+id/imageViewElderlycare"
            app:layout_constraintTop_toTopOf="parent"
            app:layout_constraintVertical_bias="0.506" />
    </androidx.constraintlayout.widget.ConstraintLayout>

```

```

        </androidx.cardview.widget.CardView>

    </LinearLayout>

    <com.google.android.material.bottomnavigation.BottomNavigationView
        android:id="@+id/bottom_navigation"
        style="@style/Widget.MaterialComponents.BottomNavigationView.Colored"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom"
        app:menu="@menu/bottom_navigation_menu" />

</androidx.coordinatorlayout.widget.CoordinatorLayout>

```

[File XML activity_daycare_maps.xml]

```

<?xml version="1.0" encoding="utf-8"?>
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <ProgressBar
        android:id="@+id/progressBar"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <com.google.android.material.appbar.AppBarLayout
            android:id="@+id/appToolbar"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar">

            <com.google.android.material.appbar.CollapsingToolbarLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                app:layout_scrollFlags="scroll|exitUntilCollapsed">

                <androidx.appcompat.widget.Toolbar
                    android:id="@+id/toolbar"
                    android:layout_width="match_parent"
                    android:layout_height="?attr/actionBarSize"
                    android:background="?attr/colorPrimary"
                    android:elevation="4dp"
                    app:layout_scrollFlags="scroll|enterAlways">

                    <TextView
                        android:id="@+id/toolbar_title"
                        android:layout_width="wrap_content"
                        android:layout_height="wrap_content"
                        android:fontFamily="@font/montserratsemibold"
                        android:text="Daycare Facilities"
                        android:textColor="@color/white"
                        android:textSize="18sp" />

                </androidx.appcompat.widget.Toolbar>
            </com.google.android.material.appbar.CollapsingToolbarLayout>
        </com.google.android.material.appbar.AppBarLayout>

        <LinearLayout

```

```
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:orientation="vertical">

    <androidx.cardview.widget.CardView
        android:id="@+id/locationLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="16dp"
        android:layout_marginTop="16dp"
        android:layout_marginRight="16dp"
        android:layout_marginBottom="4dp"
        app:cardBackgroundColor="@color/white"
        app:cardCornerRadius="30dp"
        app:cardElevation="10dp">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center_vertical"
            android:orientation="horizontal"
            android:paddingTop="12dp"
            android:paddingBottom="12dp">

            <ImageView
                android:id="@+id/imageprofile"
                android:layout_width="30dp"
                android:layout_height="30dp"
                android:layout_marginStart="10dp"
                android:scaleType="centerCrop"
                android:src="@drawable/ic_location_primary" />

            <TextView
                android:id="@+id/tv_Location"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_marginLeft="4dp"
                android:fontFamily="@font/montserrat"
                android:text="Location :"
                android:textColor="@color/black"
                android:textSize="16sp" />

            <TextView
                android:id="@+id/tv_UserLocation"
                android:layout_width="0dp"
                android:layout_height="wrap_content"
                android:layout_marginLeft="8dp"
                android:layout_marginRight="10dp"
                android:layout_weight="1"
                android:ellipsize="end"
                android:fontFamily="@font/montserrat"
                android:maxLines="1"
                android:text="Jakarta"
                android:textColor="@color/black"
                android:textSize="16sp"
                android:textStyle="bold" />

        </LinearLayout>

    </androidx.cardview.widget.CardView>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="16dp"
        android:layout_marginRight="16dp"
```

```

        android:gravity="center_vertical"
        android:orientation="horizontal"
        android:paddingTop="12dp"
        android:paddingBottom="12dp"

        <TextView
            android:id="@+id/tv_NearestDaycare"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginLeft="4dp"
            android:fontFamily="@font/montserrat"
            android:text="Nearest Daycare from You"
            android:textColor="@color/black"
            android:textSize="16sp"
            android:textStyle="bold" />
    </LinearLayout>

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recyclerView"
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_marginLeft="16dp"
        android:layout_marginRight="16dp"
        android:layout_weight="1"
        android:paddingTop="12dp"
        android:paddingBottom="12dp" />

</LinearLayout>
</LinearLayout>
</FrameLayout>

```

[File XML activity_elderly_maps.xml]

```

<?xml version="1.0" encoding="utf-8"?>
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".view.maps.ElderlyMapsActivity">

    <ProgressBar
        android:id="@+id/progressBar"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <com.google.android.material.appbar.AppBarLayout
            android:id="@+id/appToolbar"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar">

            <com.google.android.material.appbar.CollapsingToolbarLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                app:layout_scrollFlags="scroll|exitUntilCollapsed">

                <androidx.appcompat.widget.Toolbar
                    android:id="@+id/toolbar"
                    android:layout_width="match_parent"
                    android:layout_height="?attr/actionBarSize" />

```

```

        android:background="?attr/colorPrimary"
        android:elevation="4dp"
        app:layout_scrollFlags="scroll|enterAlways">

        <TextView
            android:id="@+id/toolbar_title"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:fontFamily="@font/montserratsemibold"
            android:text="Elderly Care Facilities"
            android:textColor="@color/white"
            android:textSize="18sp" />

    </androidx.appcompat.widget.Toolbar>
</com.google.android.material.appbar.CollapsingToolbarLayout>
</com.google.android.material.appbar.AppBarLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_weight="1"
    android:orientation="vertical">

    <androidx.cardview.widget.CardView
        android:id="@+id/locationLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="16dp"
        android:layout_marginTop="16dp"
        android:layout_marginRight="16dp"
        android:layout_marginBottom="4dp"
        app:cardBackgroundColor="@color/white"
        app:cardCornerRadius="30dp"
        app:cardElevation="10dp">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center_vertical"
            android:orientation="horizontal"
            android:paddingTop="12dp"
            android:paddingBottom="12dp">

            <ImageView
                android:id="@+id/imageprofile"
                android:layout_width="30dp"
                android:layout_height="30dp"
                android:layout_marginStart="10dp"
                android:scaleType="centerCrop"
                android:src="@drawable/ic_location_primary" />

            <TextView
                android:id="@+id/tv_Location"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_marginLeft="4dp"
                android:fontFamily="@font/montserrat"
                android:text="Location :"
                android:textColor="@color/black"
                android:textSize="16sp" />

            <TextView
                android:id="@+id/tv_UserLocation"
                android:layout_width="0dp"
                android:layout_height="wrap_content"
                android:layout_marginLeft="8dp"

```

```

        android:layout_marginRight="10dp"
        android:layout_weight="1"
        android:ellipsize="end"
        android:fontFamily="@font/montserrat"
        android:maxLines="1"
        android:text="Jakarta"
        android:textColor="@color/black"
        android:textSize="16sp"
        android:textStyle="bold" />

    </LinearLayout>

</androidx.cardview.widget.CardView>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="16dp"
    android:layout_marginRight="16dp"
    android:gravity="center_vertical"
    android:orientation="horizontal"
    android:paddingTop="12dp"
    android:paddingBottom="12dp">

    <TextView
        android:id="@+id/tv_NearestDaycare"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="4dp"
        android:fontFamily="@font/montserrat"
        android:text="Nearest Elderly Care from You"
        android:textColor="@color/black"
        android:textSize="16sp"
        android:textStyle="bold" />

    </LinearLayout>

<androidx.recyclerview.widget.RecyclerView
    android:id="@+id/recyclerView"
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_marginLeft="16dp"
    android:layout_marginRight="16dp"
    android:layout_weight="1"
    android:paddingTop="12dp"
    android:paddingBottom="12dp" />

    </LinearLayout>
</LinearLayout>
</FrameLayout>
```

[File XML item_daycare.xml]

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:clickable="true"
    android:focusable="true"
    android:orientation="horizontal"
    android:padding="8dp">

    <ImageView
        android:id="@+id/daycare_image"
        android:layout_width="105dp"
        android:layout_height="105dp"
        android:layout_marginEnd="8dp"
        android:scaleType="centerCrop" />
```

```

<LinearLayout
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:orientation="vertical">

    <TextView
        android:id="@+id/daycare_name"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:fontFamily="@font/montserrat"
        android:text="Daycare Name"
        android:textColor="@color/black"
        android:textSize="16sp"
        android:textStyle="bold" />

    <TextView
        android:id="@+id/daycare_location"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="4dp"
        android:fontFamily="@font/montserrat"
        android:text="Location"
        android:textColor="@color/black"
        android:textSize="16sp" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="4dp"
        android:layout_weight="1"
        android:orientation="horizontal">

        <ImageView
            android:id="@+id/distanceLogo"
            android:layout_width="24dp"
            android:layout_height="24dp"
            android:layout_marginEnd="8dp"
            android:scaleType="centerCrop"
            android:src="@drawable/ic_distance" />

        <TextView
            android:id="@+id/tvDistancetoUser"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:fontFamily="@font/montserrat"
            android:text="8.3 km"
            android:textColor="@color/black"
            android:textSize="16sp"
            android:textStyle="bold" />
    </LinearLayout>

</LinearLayout>
</LinearLayout>

```

[File XML activity_web_view.xml]

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".view.maps.WebViewActivity">

```

```

<com.google.android.material.appbar.AppBarLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar">

    <com.google.android.material.appbar.CollapsingToolbarLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        app:layout_scrollFlags="scroll|exitUntilCollapsed">

        <androidx.appcompat.widget.Toolbar
            android:id="@+id/toolbar"
            android:layout_width="match_parent"
            android:layout_height="?attr/actionBarSize"
            android:background="?attr/colorPrimary"
            android:elevation="4dp"
            app:layout_scrollFlags="scroll|enterAlways">

            <TextView
                android:id="@+id/toolbar_title"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_gravity="start|center"
                android:layout_marginStart="16dp"
                android:fontFamily="@font/montserratsemibold"
                android:text="Facilities Page"
                android:textColor="@color/white"
                android:textSize="18sp" />
        </androidx.appcompat.widget.Toolbar>
    </com.google.android.material.appbar.CollapsingToolbarLayout>
</com.google.android.material.appbar.AppBarLayout>

<WebView
    android:id="@+id/webView"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_below="@+id/toolbar" />

</androidx.coordinatorlayout.widget.CoordinatorLayout>

```

Penjelasan:

DaycareMapsActivity.kt dan ElderlyCareMapsActivity.kt bertanggung jawab untuk menampilkan lokasi daycare dan elderly care terdekat kepada pengguna. Kedua aktivitas ini mengatur tampilan peta, mengambil lokasi pengguna, dan menampilkan daftar daycare dan elderly care yang diurutkan berdasarkan jarak dari lokasi pengguna menggunakan RecyclerView. Jika pengguna memberikan izin akses lokasi, aplikasi menggunakan FusedLocationProviderClient untuk mendapatkan lokasi terkini pengguna, menampilkan alamat pengguna, dan memuat data daycare dan elderly care dari Firestore. Jika izin tidak diberikan, aktivitas ini tetap memuat daftar fasilitas tetapi tanpa informasi jarak dari pengguna. Aktivitas ini juga menampilkan dialog popup yang menampilkan alamat lokasi pengguna saat ini. Kemudian, apabila pengguna memilih salah satu item dari list daycare maupun elderly care, maka activity akan pengguna ke WebViewActivity.

WebViewActivity.kt tersebut akan menampilkan halaman situs resmi dari fasilitas yang dipilih sebelumnya.

2.2.8 Profile

[File activity ProfileActivity.kt]

```
package com.app.famcare.view.profile

import android.content.DialogInterface
import android.content.Intent
import android.os.Bundle
import android.widget.RelativeLayout
import android.widget.TextView
import androidx.appcompat.app.AlertDialog
import androidx.appcompat.app.AppCompatActivity
import androidx.appcompat.widget.Toolbar
import androidx.cardview.widget.CardView
import com.app.famcare.R
import com.app.famcare.databinding.ActivityProfileBinding
import com.app.famcare.view.bookmark.BookmarkActivity
import com.app.famcare.view.facilities.FacilitiesActivity
import com.app.famcare.view.history.HistoryActivity
import com.app.famcare.view.login.LoginActivity
import com.app.famcare.view.main.MainActivity
import com.bumptech.glide.Glide
import
com.google.android.material.bottomnavigation.BottomNavigationView
import com.google.firebase.auth.FirebaseAuth
import com.google.firebaseio.firebaseio.FirebaseFirestore

class ProfileActivity : AppCompatActivity() {

    private lateinit var firebaseAuth: FirebaseAuth
    private lateinit var firestore: FirebaseFirestore
    private lateinit var textViewName: TextView
    private lateinit var textViewEmail: TextView
    private lateinit var textViewPhone: TextView
    private lateinit var binding: ActivityProfileBinding

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityProfileBinding.inflate(layoutInflater)
        setContentView(binding.root)

        firebaseAuth = FirebaseAuth.getInstance()
        firestore = FirebaseFirestore.getInstance()

        val toolbar: Toolbar = binding.toolbar
        setSupportActionBar(toolbar)
        supportActionBar?.title = ""

        val bottomNavigation: BottomNavigationView =
            binding.bottomNavigation
```

```

        bottomNavigation.selectedItemId = R.id.page_4

        bottomNavigation.setOnNavigationItemSelectedListener { item
->
            when (item.itemId) {
                R.id.page_1 -> {
                    val intent = Intent(this,
MainActivity::class.java)
                    startActivity(intent)
                    true
                }

                R.id.page_2 -> {
                    val intent = Intent(this,
HistoryActivity::class.java)
                    startActivity(intent)
                    true
                }

                R.id.page_3 -> {
                    val intent = Intent(this,
FacilitiesActivity::class.java)
                    startActivity(intent)
                    true
                }

                R.id.page_4 -> true
            }
        }

        val editProfileCardView: CardView = binding.editProfile
        editProfileCardView.setOnClickListener {
            val intent = Intent(this,
EditProfileActivity::class.java)
            startActivity(intent)
        }

        val settingsLayout: RelativeLayout = binding.accountSettings
        settingsLayout.setOnClickListener {
            val intent = Intent(this, SettingsActivity::class.java)
            startActivity(intent)
        }

        val bookmarkLayout: RelativeLayout = binding.bookmarkActivity
        bookmarkLayout.setOnClickListener {
            val intent = Intent(this, BookmarkActivity::class.java)
            startActivity(intent)
        }

        val termsoserviceLayout: RelativeLayout =
binding.termsOfServices
        termsoserviceLayout.setOnClickListener {
            val intent = Intent(this,
TermsOfServicesActivity::class.java)

```

```

        startActivity(intent)
    }
    val privacypolicyLayout: RelativeLayout =
binding.privacyPolicy
    privacypolicyLayout.setOnClickListener {
        val intent = Intent(this,
PrivacyPolicyActivity::class.java)
        startActivity(intent)
    }
    val aboutLayout: RelativeLayout = binding.aboutPage
aboutLayout.setOnClickListener {
        val intent = Intent(this, AboutActivity::class.java)
        startActivity(intent)
    }
    val logoutCardView: RelativeLayout = binding.logoutButton
logoutCardView.setOnClickListener {
        showLogoutConfirmationDialog()
    }
}

private fun showLogoutConfirmationDialog() {
    val builder = AlertDialog.Builder(this)
    builder.setMessage("Are you sure you want to log out?")
        .setPositiveButton("Yes", DialogInterface.OnClickListener { dialog, id ->
            firebaseAuth.signOut()
            val intent = Intent(this, LoginActivity::class.java)
            startActivity(intent)
            finish()
        }).setNegativeButton("No", DialogInterface.OnClickListener { dialog, id ->
            dialog.dismiss()
        })
    builder.create().show()
}

override fun onResume() {
    super.onResume()
    loadUserDataFromFirestore()
}

private fun loadUserDataFromFirestore() {
    val currentUserUid = firebaseAuth.currentUser?.uid

    if (currentUserUid != null) {

firestore.collection("User").document(currentUserUid).get()
        .addOnSuccessListener { document ->
            if (document != null) {
                val name = document.getString("fullName")
                val email = document.getString("email")
                val phone = document.getString("phone")

```

```
        binding.textName.text = name
        binding.textEmail.text = email
        binding.textPhone.text = phone

        val imageUrl = document.getString("profileImageUrl")
        if (imageUrl != null && imageUrl.isNotEmpty()) {
            Glide.with(this).load(imageUrl).into(binding.imageprofile)
        }
    }.addOnFailureListener { exception ->
}
}
}
```

[File activity EditProfileActivity.kt]

```
package com.app.famcare.view.profile

import android.Manifest
import android.app.Activity
import android.app.DatePickerDialog
import android.content.Intent
import android.content.pm.PackageManager
import android.graphics.Bitmap
import android.graphics.drawable.BitmapDrawable
import android.net.Uri
import android.os.Bundle
import android.text.Editable
import android.text.TextWatcher
import android.view.MenuItem
import android.view.View
import android.widget.DatePicker
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity
import androidx.core.app.ActivityCompat
import androidx.core.content.ContextCompat
import com.app.famcare.R
import com.app.famcare.databinding.ActivityEditProfileBinding
import com.bumptech.glide.Glide
import com.google.firebase.auth.FirebaseAuth
import com.google.firebase.firestore.FirebaseFirestore
import com.google.firebase.ktx.Firebase
import com.google.firebase.storage.ktx.storage
import java.io.ByteArrayOutputStream
import java.text.SimpleDateFormat
import java.util.*

class EditProfileActivity : AppCompatActivity() {
    private lateinit var firebaseAuth: FirebaseAuth
```

```

private lateinit var firestore: FirebaseFirestore
private lateinit var binding: ActivityEditProfileBinding
private val storageRef = Firebase.storage.reference
private val PICK_IMAGE_REQUEST = 1
private var imageUri: Uri? = null

override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    binding = ActivityEditProfileBinding.inflate(layoutInflater)
    setContentView(binding.root)

    firebaseAuth = FirebaseAuth.getInstance()
    firestore = FirebaseFirestore.getInstance()

    val toolbar = binding.toolbar
    setSupportActionBar(toolbar)
    supportActionBar?.setDisplayHomeAsUpEnabled(true)
    supportActionBar?.title = ""

    binding.profileImageView.setImageResource(R.drawable.user)

    binding.editProfileImage.setOnClickListener {
        if (ContextCompat.checkSelfPermission(
            this,
            Manifest.permission.READ_EXTERNAL_STORAGE
        ) == PackageManager.PERMISSION_GRANTED
        ) {
            openGallery()
        } else {
            ActivityCompat.requestPermissions(
                this,
                arrayOf(Manifest.permission.READ_EXTERNAL_STORAGE,
                    PERMISSION_REQUEST_CODE
                )
            )
        }
    }

    binding.saveChanges.setOnClickListener {
        if (validateInput()) {
            binding.saveChangesProgressBar.visibility =
View.VISIBLE
            uploadImageToFirebaseStorage()
        } else {
            Toast.makeText(this, "Please fill all fields and
select gender", Toast.LENGTH_SHORT)
                .show()
        }
    }

    binding.icDate.setOnClickListener {
        showDatePickerDialog()
    }
}

```

```

        loadUserDataFromFirestore()
        addTextWatchers()
    }

private fun showDatePickerDialog() {
    val calendar = Calendar.getInstance()
    val year = calendar.get(Calendar.YEAR)
    val month = calendar.get(Calendar.MONTH)
    val day = calendar.get(Calendar.DAY_OF_MONTH)

    val datePickerDialog = DatePickerDialog(
        this, { _: DatePicker, selectedYear: Int, selectedMonth: Int, selectedDay: Int ->
            val selectedDate = Calendar.getInstance()
            selectedDate.set(selectedYear, selectedMonth, selectedDay)
            val dateFormat = SimpleDateFormat("dd/MM/yyyy", Locale.US)
            binding.selectedDateTextView.text =
            dateFormat.format(selectedDate.time)
        }, year, month, day
    )

    datePickerDialog.datePicker.maxDate = calendar.timeInMillis -
    3L * 365 * 24 * 60 * 60 * 1000
    datePickerDialog.show()
}

override fun onRequestPermissionsResult(
    requestCode: Int, permissions: Array<out String>, grantResults: IntArray
) {
    super.onRequestPermissionsResult(requestCode, permissions, grantResults)
    if (requestCode == PERMISSION_REQUEST_CODE && grantResults.isNotEmpty() && grantResults[0] ==
    PackageManager.PERMISSION_GRANTED) {
        openGallery()
    } else {
        Toast.makeText(this, "Permission denied", Toast.LENGTH_SHORT).show()
    }
}

private fun openGallery() {
    val intent = Intent(Intent.ACTION_PICK)
    intent.type = "image/*"
    startActivityForResult(intent, PICK_IMAGE_REQUEST)
}

override fun onActivityResult(requestCode: Int, resultCode: Int, data: Intent?) {

```

```

        super.onActivityResult(requestCode, resultCode, data)

        if (requestCode == PICK_IMAGE_REQUEST && resultCode == Activity.RESULT_OK && data != null && data.data != null) {
            imageUri = data.data!!
            binding.profileImageView.setImageURI(imageUri)
        }
    }

private fun loadUserDataFromFirestore() {
    val currentUserUid = firebaseAuth.currentUser?.uid

    if (currentUserUid != null) {

        firestore.collection("User").document(currentUserUid).get()
            .addOnSuccessListener { document ->
                if (document != null) {
                    val name = document.getString("fullName")
                    val phone = document.getString("phone")
                    val address = document.getString("address")
                    ?: ""
                    val gender = document.getString("gender") ?:
                    ""
                    val birthDate = document.getString("birthDate") ?:
                    ""

                    binding.fullNameEditText.setText(name)
                    binding.phoneEditText.setText(phone)
                    binding.addressEditText.setText(address)
                    if (birthDate.isNullOrEmpty()) {
                        binding.selectedDateTextView.text =
                            "dd/mm/yyyy"
                    } else {
                        binding.selectedDateTextView.text =
                            birthDate
                    }

                    when (gender) {
                        "Male" ->
                            binding.genderRadioGroup.check(R.id.radioMale)
                        "Female" ->
                            binding.genderRadioGroup.check(R.id.radioFemale)
                        else ->
                            binding.genderRadioGroup.clearCheck()
                    }
                }
            }

            val imageUrl = document.getString("profileImageUrl")
            if (imageUrl != null && imageUrl.isNotEmpty()) {
                Glide.with(this).load(imageUrl).into(binding.profileImageView)
            }
        }
    }
}

```

```

        }

        }.addOnFailureListener { exception ->
            Toast.makeText(this, "Failed to load user data",
Toast.LENGTH_SHORT).show()
        }
    }
}

private fun uploadImageToFirebaseStorage() {
    val currentUserUid = firebaseAuth.currentUser?.uid
    if (currentUserUid != null && imageUri != null) {
        val imageName = "profile_${currentUserUid}.jpg"
        val imageRef = storageRef.child("User/$imageName")

        val bitmapDrawable = binding.profileImageView.drawable as
BitmapDrawable
        val bitmap = bitmapDrawable.bitmap
        val baos = ByteArrayOutputStream()
        bitmap.compress(Bitmap.CompressFormat.JPEG, 100, baos)
        val imageData = baos.toByteArray()

        val uploadTask = imageRef.putBytes(imageData)
        uploadTask.addOnSuccessListener { taskSnapshot ->
            imageRef.downloadUrl.addOnSuccessListener { uri ->
                val imageUrl = uri.toString()
                saveUserDataToFirestore(imageUrl)
            }
        }.addOnFailureListener { exception ->
            Toast.makeText(this, "Error uploading image",
Toast.LENGTH_SHORT).show()
        }
    } else {
        saveUserDataToFirestore("")
    }
}

private fun saveUserDataToFirestore(imageUrl: String) {
    val currentUserUid = firebaseAuth.currentUser?.uid
    if (currentUserUid != null) {
        val name = binding.fullNameEditText.text.toString()
        val phone = binding.phoneEditText.text.toString()
        val address = binding.addressEditText.text.toString()
        val birthDate =
binding.selectedDateTextView.text.toString()
        val gender = when
(binding.genderRadioGroup.checkedRadioButtonId) {
            R.id.radioMale -> "Male"
            R.id.radioFemale -> "Female"
            else -> ""
        }
    }
}

```

```

        val userData = mutableMapOf<String, Any>(
            "fullName" to name,
            "phone" to phone,
            "address" to address,
            "gender" to gender,
            "birthDate" to birthDate
        )

        if (imageUrl.isNotEmpty()) {
            userData["profileImageUrl"] = imageUrl
        }

        firestore.collection("User").document(currentUserId).update(userData)
            .addOnSuccessListener {
                binding.saveChangesProgressBar.visibility = View.GONE
                Toast.makeText(this, "Profile updated successfully", Toast.LENGTH_SHORT).show()
                finish()
            }
            .addOnFailureListener { exception ->
                Toast.makeText(this, "Error updating profile", Toast.LENGTH_SHORT).show()
                binding.saveChangesProgressBar.visibility = View.GONE
            }
    }

    private fun validateInput(): Boolean {
        val name = binding.fullNameEditText.text.toString().trim()
        val phone = binding.phoneEditText.text.toString().trim()
        val address = binding.addressEditText.text.toString().trim()
        val gender = binding.genderRadioGroup.checkedRadioButtonId
        val birthDate = binding.selectedDateTextView.text.toString().trim()

        return name.isNotEmpty() && phone.isNotEmpty() && address.isNotEmpty() && gender != -1 && birthDate.isNotEmpty()
    }

    private fun addTextWatchers() {
        val watcher = object : TextWatcher {
            override fun afterTextChanged(s: Editable?) {
                binding.saveChanges.isEnabled = validateInput()
            }

            override fun beforeTextChanged(s: CharSequence?, start: Int, count: Int, after: Int) {}
            override fun onTextChanged(s: CharSequence?, start: Int, before: Int, count: Int) {}
        }
    }
}

```

```

        binding.fullNameEditText.addTextChangedListener(watcher)
        binding.phoneEditText.addTextChangedListener(watcher)
        binding.addressEditText.addTextChangedListener(watcher)
        binding.genderRadioGroup.setOnCheckedChangeListener { _, _ ->
            binding.saveChanges.isEnabled = validateInput()
        }

        binding.saveChanges.isEnabled = validateInput()
    }

    override fun onOptionsItemSelected(item: MenuItem): Boolean {
        return when (item.itemId) {
            android.R.id.home -> {
                onBackPressed()
                true
            }
            else -> super.onOptionsItemSelected(item)
        }
    }

    companion object {
        private const val PERMISSION_REQUEST_CODE = 123
    }
}

```

[File activity SettingsActivity.kt]

```

package com.app.famcare.view.profile

import android.content.Intent
import android.os.Bundle
import android.view.View
import android.widget.Button
import android.widget.EditText
import android.widget.LinearLayout
import android.widget.RelativeLayout
import android.widget.TextView
import androidx.appcompat.app.AlertDialog
import androidx.appcompat.app.AppCompatActivity
import com.app.famcare.R
import com.app.famcare.databinding.ActivitySettingsBinding
import com.app.famcare.view.login.LoginActivity
import com.google.firebase.auth.EmailAuthProvider
import com.google.firebase.auth.FirebaseAuth

class SettingsActivity : AppCompatActivity() {
    private lateinit var binding: ActivitySettingsBinding
    private lateinit var firebaseAuth: FirebaseAuth
    private lateinit var resetPasswordLayout: RelativeLayout
    private lateinit var passwordTextView: TextView
    private lateinit var currentPasswordTextLayout: LinearLayout
    private lateinit var currentPasswordEditText: EditText

```

```

private lateinit var resetButton: Button

override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    binding = ActivitySettingsBinding.inflate(layoutInflater)
    setContentView(binding.root)

    firebaseAuth = FirebaseAuth.getInstance()

    newPasswordLayout = findViewById(R.id.resetPassword)
    passwordTextView = findViewById(R.id.passwordTextView)
    currentPasswordTextLayout = =
    findViewById(R.id.currentPasswordTextLayout)
    currentPasswordEditText = =
    findViewById(R.id.currentPasswordEditText)
    resetButton = findViewById(R.id.resetButton)

    setSupportActionBar(binding.toolbar)
    supportActionBar?.apply {
        title = ""
        setDisplayHomeAsUpEnabled(true)
        setDisplayShowHomeEnabled(true)
    }

    binding.toolbar.setNavigationOnClickListener {
        onBackPressed()
    }

    newPasswordLayout.setOnClickListener {
        passwordTextView.visibility = View.VISIBLE
        currentPasswordTextLayout.visibility = View.VISIBLE
        resetButton.visibility = View.VISIBLE
    }

    resetButton.setOnClickListener {
        val currentPassword = =
        currentPasswordEditText.text.toString().trim()
        if (currentPassword.isNotEmpty()) {
            verifyCurrentPassword(currentPassword)
        } else {
            showAlertDialog("Error", "Please enter your current
password.")
        }
    }
}

private fun verifyCurrentPassword(currentPassword: String) {
    val user = firebaseAuth.currentUser
    if (user != null && user.email != null) {
        val credential = =
        EmailAuthProvider.getCredential(user.email!!, currentPassword)
    }
}

```

```

        user.reauthenticate(credential).addOnCompleteListener {
task ->
    if (task.isSuccessful) {
        sendPasswordResetEmail(user.email!!)
    } else {
        showAlertDialog("Error", "Current password is
incorrect.")
    }
}

private fun sendPasswordResetEmail(userEmail: String) {
    firebaseAuth.sendPasswordResetEmail(userEmail).addOnCompleteListener
{ task ->
    if (task.isSuccessful) {
        showAlertDialog("Success", "Password reset email
sent.")
        logoutUser()
    } else {
        showAlertDialog("Error", "Failed to send reset
email.")
    }
}

private fun logoutUser() {
    firebaseAuth.signOut()
    navigateToLoginScreen()
}

private fun navigateToLoginScreen() {
    val intent = Intent(this, LoginActivity::class.java)
    intent.flags = Intent.FLAG_ACTIVITY_NEW_TASK or
Intent.FLAG_ACTIVITY_CLEAR_TASK
    startActivity(intent)
    finish()
}

private fun showAlertDialog(title: String, message: String,
onDismiss: () -> Unit)? = null {
    AlertDialog.Builder(this).setTitle(title).setMessage(message)
        .setPositiveButton(android.R.string.ok) { dialog, _ ->
            dialog.dismiss()
            onDismiss?.invoke()
        }.show()
}

override fun onSupportNavigateUp(): Boolean {
    onBackPressed()
    return true
}

```

```
}
```

[File activity PrivacyPolicyActivity.kt]

```
package com.app.famcare.view.profile

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.view.MenuItem
import androidx.appcompat.widget.Toolbar
import com.app.famcare.R

class PrivacyPolicyActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_privacy_policy)

        val toolbar = findViewById<Toolbar>(R.id.toolbar)
        setSupportActionBar(toolbar)
        supportActionBar?.setDisplayHomeAsUpEnabled(true)
        supportActionBar?.title = ""

    }

    override fun onOptionsItemSelected(item: MenuItem): Boolean {
        return when (item.itemId) {
            android.R.id.home -> {
                onBackPressed()
                true
            }
            else -> super.onOptionsItemSelected(item)
        }
    }
}
```

[File activity TermsOfServiceActivity.kt]

```
package com.app.famcare.view.profile

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.view.MenuItem
import androidx.appcompat.widget.Toolbar
import com.app.famcare.R

class TermsOfServicesActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_terms_of_services)

        val toolbar = findViewById<Toolbar>(R.id.toolbar)
```

```

        setSupportActionBar(toolbar)
        supportActionBar?.setDisplayHomeAsUpEnabled(true)
        supportActionBar?.title = ""

    }

    override fun onOptionsItemSelected(item: MenuItem): Boolean {
        return when (item.itemId) {
            android.R.id.home -> {
                onBackPressed()
                true
            }

            else -> super.onOptionsItemSelected(item)
        }
    }
}

```

[File activity AboutActivity.kt]

```

package com.app.famcare.view.profile

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import com.app.famcare.R
import android.view.MenuItem
import androidx.appcompat.widget.Toolbar

class AboutActivity : AppCompatActivity() {

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

        val toolbar = findViewById<Toolbar>(R.id.toolbar)
        setSupportActionBar(toolbar)
        supportActionBar?.setDisplayHomeAsUpEnabled(true)
        supportActionBar?.title = ""

    }

    override fun onOptionsItemSelected(item: MenuItem): Boolean {
        return when (item.itemId) {
            android.R.id.home -> {
                onBackPressed()
                true
            }

            else -> super.onOptionsItemSelected(item)
        }
    }
}

```

[File XML activity_profile.xml]

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".view.profile.ProfileActivity">

    <com.google.android.material.appbar.AppBarLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar">

        <com.google.android.material.appbar.CollapsingToolbarLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            app:layout_scrollFlags="scroll|exitUntilCollapsed">

            <androidx.appcompat.widget.Toolbar
                android:id="@+id/toolbar"
                android:layout_width="match_parent"
                android:layout_height="?attr/actionBarSize"
                android:background="?attr/colorPrimary"
                android:elevation="4dp"
                app:layout_scrollFlags="scroll|enterAlways">

                <TextView
                    android:id="@+id/toolbar_title"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:fontFamily="@font/montserratsemibold"
                    android:text="Profile"
                    android:textColor="@color/white"
                    android:textSize="18sp" />

            </androidx.appcompat.widget.Toolbar>
        </com.google.android.material.appbar.CollapsingToolbarLayout>

    </com.google.android.material.appbar.AppBarLayout>

    <androidx.core.widget.NestedScrollView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="fill_vertical"
        android:layout_marginBottom="122dp"
```

```
        android:fillViewport="true"
        app:layout_behavior="@string/appbar_scrolling_view_behavior">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <androidx.cardview.widget.CardView
            android:id="@+id/editProfile"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_margin="16dp"
            app:cardBackgroundColor="@color/white"
            app:cardCornerRadius="30dp"
            app:cardElevation="10dp">

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:gravity="center_vertical"
                android:orientation="horizontal"
                android:paddingTop="16dp"
                android:paddingBottom="16dp">

                <de.hdodenhof.circleimageview.CircleImageView
                    android:id="@+id/imageprofile"
                    android:layout_width="80dp"
                    android:layout_height="80dp"
                    android:layout_marginStart="10dp"
                    android:scaleType="centerCrop"
                    android:src="@drawable/user" />

                <LinearLayout
                    android:layout_width="0dp"
                    android:layout_height="match_parent"
                    android:layout_weight="1"
                    android:gravity="center_vertical|start"
                    android:orientation="vertical"
                    android:paddingStart="8dp"
                    android:paddingEnd="12dp">

                    <TextView
                        android:id="@+id/text_name"
                        android:layout_width="match_parent"
                        android:layout_height="wrap_content"
```

```
        android:fontFamily="@font/montserratsemibold"
                android:text="User"
                android:textColor="@color/black"
                android:textSize="18sp"
                android:textStyle="bold" />

        <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginTop="8dp"
                android:gravity="center_vertical">

                <ImageView
                        android:layout_width="18dp"
                        android:layout_height="18dp"
                        android:src="@drawable/ic_emailku" />

                <TextView
                        android:id="@+id/text_email"
                        android:layout_width="wrap_content"
                        android:layout_height="wrap_content"
                        android:layout_marginStart="4dp"
                        android:fontFamily="@font/montserrat"
                        android:text="user@mail.com"
                        android:textColor="@color/black"
                        android:textSize="16sp" />
        </LinearLayout>

        <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginTop="4dp"
                android:gravity="center_vertical">

                <ImageView
                        android:layout_width="18dp"
                        android:layout_height="18dp"
                        android:src="@drawable/ic_phone2" />

                <TextView
                        android:id="@+id/text_phone"
                        android:layout_width="wrap_content"
                        android:layout_height="wrap_content"
                        android:layout_marginStart="4dp"
                        android:fontFamily="@font/montserrat"
                        android:text="08xxxxxxxxxx"
```

```
        android:textColor="@color/black"
        android:textSize="16sp" />
    </LinearLayout>
</LinearLayout>

<ImageView
    android:id="@+id/btn_edit_profile"
    android:layout_width="35dp"
    android:layout_height="35dp"
    android:layout_marginLeft="4dp"
    android:layout_marginRight="10dp"
    android:contentDescription="Edit Profile"
    android:src="@drawable/ic_pencil" />
</LinearLayout>
</androidx.cardview.widget.CardView>

<RelativeLayout
    android:id="@+id/bookmarkActivity"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="16dp"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="16dp"
    android:gravity="center_horizontal"
    android:paddingTop="16dp"
    android:paddingBottom="16dp">

    <LinearLayout
        android:id="@+id/leftAlignedBookmark"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentStart="true"
        android:layout_marginStart="12dp"
        android:gravity="start"
        android:orientation="horizontal">

        <ImageView
            android:layout_width="30dp"
            android:layout_height="30dp"
            android:layout_gravity="center_vertical"
            android:src="@drawable/ic_bookmark" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_vertical"
            android:layout_marginStart="16dp"
            android:fontFamily="@font/montserrat"
            android:text="Bookmark"
            android:textColor="@color/black"
            android:textSize="18sp" />
    
```

```
</LinearLayout>

<LinearLayout
    android:id="@+id/rightAlignedArrow"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentEnd="true"
    android:layout_marginEnd="12dp"
    android:gravity="end"
    android:orientation="horizontal">

    <ImageView
        android:layout_width="25dp"
        android:layout_height="20dp"
        android:contentDescription="arrow"
        android:src="@drawable/ic_arrow" />

</LinearLayout>

</RelativeLayout>

<RelativeLayout
    android:id="@+id/accountSettings"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="16dp"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="16dp"
    android:gravity="center_horizontal"
    android:paddingTop="16dp"
    android:paddingBottom="16dp">

    <LinearLayout
        android:id="@+id/leftAlignedRating"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentStart="true"
        android:layout_marginStart="12dp"
        android:gravity="start"
        android:orientation="horizontal">

        <ImageView
            android:layout_width="30dp"
            android:layout_height="30dp"
            android:layout_gravity="center_vertical"
            android:src="@drawable/ic_accountsetting2" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_vertical"
```

```
        android:layout_marginStart="16dp"
        android:fontFamily="@font/montserrat"
        android:text="Account Settings"
        android:textColor="@color/black"
        android:textSize="18sp" />

    </LinearLayout>

    <LinearLayout
        android:id="@+id/rightAlignedType"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_marginEnd="12dp"
        android:gravity="end"
        android:orientation="horizontal">

        <ImageView
            android:id="@+id/imageViewType"
            android:layout_width="25dp"
            android:layout_height="20dp"
            android:contentDescription="arrow"
            android:src="@drawable/ic_arrow" />

    </LinearLayout>

</RelativeLayout>

<RelativeLayout
    android:id="@+id/termsOfServices"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="16dp"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="16dp"
    android:gravity="center_horizontal"
    android:paddingTop="16dp"
    android:paddingBottom="16dp">

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentStart="true"
        android:layout_marginStart="12dp"
        android:gravity="start"
        android:orientation="horizontal">

        <ImageView
            android:layout_width="30dp"
            android:layout_height="30dp"
            android:layout_gravity="center_vertical"
```

```
        android:src="@drawable/ic_terms_of_service"
    />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_vertical"
        android:layout_marginStart="16dp"
        android:fontFamily="@font/montserrat"
        android:text="Terms of Services"
        android:textColor="@color/black"
        android:textSize="18sp" />

    </LinearLayout>

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_marginEnd="12dp"
        android:gravity="end"
        android:orientation="horizontal">

        <ImageView
            android:layout_width="25dp"
            android:layout_height="20dp"
            android:contentDescription="arrow"
            android:src="@drawable/ic_arrow" />

    </LinearLayout>

</RelativeLayout>

<RelativeLayout
    android:id="@+id/privacyPolicy"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="16dp"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="16dp"
    android:gravity="center_horizontal"
    android:paddingTop="16dp"
    android:paddingBottom="16dp">

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentStart="true"
        android:layout_marginStart="12dp"
        android:gravity="start"
```

```
        android:orientation="horizontal">

        <ImageView
            android:layout_width="30dp"
            android:layout_height="30dp"
            android:layout_gravity="center_vertical"
            android:src="@drawable/ic_privacy_policy" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_vertical"
            android:layout_marginStart="16dp"
            android:fontFamily="@font/montserrat"
            android:text="Privacy Policy"
            android:textColor="@color/black"
            android:textSize="18sp" />

    </LinearLayout>

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_marginEnd="12dp"
        android:gravity="end"
        android:orientation="horizontal">

        <ImageView
            android:layout_width="25dp"
            android:layout_height="20dp"
            android:contentDescription="arrow"
            android:src="@drawable/ic_arrow" />

    </LinearLayout>

</RelativeLayout>

<RelativeLayout
    android:id="@+id/aboutPage"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="16dp"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="16dp"
    android:gravity="center_horizontal"
    android:paddingTop="16dp"
    android:paddingBottom="16dp">

    <LinearLayout
        android:layout_width="wrap_content"
```

```
        android:layout_height="wrap_content"
        android:layout_alignParentStart="true"
        android:layout_marginStart="12dp"
        android:gravity="start"
        android:orientation="horizontal">

    <ImageView
        android:layout_width="30dp"
        android:layout_height="30dp"
        android:layout_gravity="center_vertical"
        android:src="@drawable/ic_about" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_vertical"
        android:layout_marginStart="16dp"
        android:fontFamily="@font/montserrat"
        android:text="About"
        android:textColor="@color/black"
        android:textSize="18sp" />

</LinearLayout>

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentEnd="true"
    android:layout_marginEnd="12dp"
    android:gravity="end"
    android:orientation="horizontal">

    <ImageView
        android:layout_width="25dp"
        android:layout_height="20dp"
        android:contentDescription="arrow"
        android:src="@drawable/ic_arrow" />

</LinearLayout>

</RelativeLayout>

<RelativeLayout
    android:id="@+id/logoutButton"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="16dp"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="16dp"
    android:gravity="center_horizontal"
```

```
        android:paddingTop="16dp"
        android:paddingBottom="16dp">

        <LinearLayout
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_alignParentStart="true"
            android:layout_marginStart="12dp"
            android:gravity="start"
            android:orientation="horizontal">

            <ImageView
                android:layout_width="30dp"
                android:layout_height="30dp"
                android:layout_gravity="center_vertical"
                android:src="@drawable/ic_logout2" />

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_gravity="center_vertical"
                android:layout_marginStart="16dp"
                android:fontFamily="@font/montserrat"
                android:text="Log Out"
                android:textColor="@color/black"
                android:textSize="18sp" />

        </LinearLayout>

        <LinearLayout
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_alignParentEnd="true"
            android:layout_marginEnd="12dp"
            android:gravity="end"
            android:orientation="horizontal">

            <ImageView
                android:layout_width="25dp"
                android:layout_height="20dp"
                android:contentDescription="arrow"
                android:src="@drawable/ic_arrow" />

        </LinearLayout>

    </RelativeLayout>
</LinearLayout>

</androidx.core.widget.NestedScrollView>
```

```
<com.google.android.material.bottomnavigation.BottomNavigationView
    android:id="@+id/bottom_navigation"
    style="@style/Widget.MaterialComponents.BottomNavigationView.Colored"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="bottom"
    app:layout_constraintBottom_toBottomOf="parent"
    app:menu="@menu/bottom_navigation_menu" />

</androidx.coordinatorlayout.widget.CoordinatorLayout>
```

[File XML activity_edit_profile.xml]

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <ProgressBar
        android:id="@+id/saveChangesProgressBar"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:visibility="gone" />

    <androidx.coordinatorlayout.widget.CoordinatorLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        tools:context=".view.profile.EditProfileActivity">

        <com.google.android.material.appbar.AppBarLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"

            android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar">

            <com.google.android.material.appbar.CollapsingToolbarLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                app:layout_scrollFlags="scroll|exitUntilCollapsed">

                <androidx.appcompat.widget.Toolbar
                    android:id="@+id/toolbar"
                    android:layout_width="match_parent"
                    android:layout_height="?attr/actionBarSize" />
```

```
        android:background="?attr/colorPrimary"
        android:elevation="4dp">

        <TextView
            android:id="@+id/toolbar_title"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:fontFamily="@font/montserratsemibold"
            android:text="Edit Profile"
            android:textColor="@color/white"
            android:textSize="18sp" />

    </androidx.appcompat.widget.Toolbar>

</com.google.android.material.appbar.CollapsingToolbarLayout>

</com.google.android.material.appbar.AppBarLayout>

<androidx.core.widget.NestedScrollView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="fill_vertical"
    android:fillViewport="true"

app:layout_behavior="@string/appbar_scrolling_view_behavior">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">

        <de.hdodenhof.circleimageview.CircleImageView
            android:id="@+id/profileImageView"
            android:layout_width="120dp"
            android:layout_height="120dp"
            android:layout_gravity="center_horizontal"
            android:layout_marginTop="18dp"
            android:src="@drawable/nanny_placeholder"
            tools:src="@drawable/user" />

        <ImageView
            android:id="@+id/editProfileImage"
            android:layout_width="35dp"
            android:layout_height="35dp"
            android:layout_gravity="center_horizontal"
            android:layout_marginStart="45dp"
            android:layout_marginTop="-25dp"
            android:src="@drawable/ic_pencil" />

        <TextView
            android:id="@+id/fullNameTextView"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
```

```
        android:layout_marginStart="32dp"
        android:layout_marginTop="24dp"
        android:fontFamily="@font/montserrat"
        android:text="Full Name"
        android:textColor="@color/black"
        android:textSize="16sp" />

<com.google.android.material.textfield.TextInputLayout
    android:id="@+id/fullNameEditTextLayout"

    style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginStart="32dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="32dp"
        app:boxCornerRadiusBottomEnd="20dp"
        app:boxCornerRadiusBottomStart="20dp"
        app:boxCornerRadiusTopEnd="20dp"
        app:boxCornerRadiusTopStart="20dp"

    app:startIconDrawable="@drawable/baseline_person_24">

    <EditText
        android:id="@+id/fullNameEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPersonName"
        tools:ignore="SpeakableTextPresentCheck" />

</com.google.android.material.textfield.TextInputLayout>

    <TextView
        android:id="@+id/genderTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="32dp"
        android:layout_marginTop="16dp"
        android:fontFamily="@font/montserrat"
        android:text="Gender"
        android:textColor="@color/black"
        android:textSize="16sp" />

    <RadioGroup
        android:id="@+id/genderRadioGroup"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginStart="32dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="32dp"
        android:orientation="horizontal">
```

```
<RadioButton
    android:id="@+id/radioFemale"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:fontFamily="@font/montserrat"
    android:text="Female"
    android:textSize="15sp" />

<RadioButton
    android:id="@+id/radioMale"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:fontFamily="@font/montserrat"
    android:text="Male"
    android:textSize="15sp" />
</RadioGroup>
<TextView
    android:id="@+id/birthTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="32dp"
    android:layout_marginTop="12dp"
    android:fontFamily="@font/montserrat"
    android:text="Birth"
    android:textColor="@color/black"
    android:textSize="16sp" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="32dp"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="32dp"
    android:orientation="horizontal">

<ImageView
    android:id="@+id/ic_date"
    android:layout_width="25dp"
    android:layout_height="25dp"
    android:layout_gravity="center_horizontal"
    android:src="@drawable/ic_calendar2" />

<TextView
    android:id="@+id/selectedDateTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="16dp"
    android:fontFamily="@font/montserrat"
    android:text="dd/mm/yyyy"
    android:textColor="@color/black"
```

```
        android:textSize="16sp" />
    </LinearLayout>

    <TextView
        android:id="@+id/addressTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="32dp"
        android:layout_marginTop="16dp"
        android:fontFamily="@font/montserrat"
        android:text="Address"
        android:textColor="@color/black"
        android:textSize="16sp" />

<com.google.android.material.textfield.TextInputLayout
    android:id="@+id/addressEditTextLayout"

    style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginStart="32dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="32dp"
        app:boxCornerRadiusBottomEnd="20dp"
        app:boxCornerRadiusBottomStart="20dp"
        app:boxCornerRadiusTopEnd="20dp"
        app:boxCornerRadiusTopStart="20dp"

    app:startIconDrawable="@drawable/baseline_location_24">

    <EditText
        android:id="@+id/addressEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPersonName"
        tools:ignore="SpeakableTextPresentCheck" />

</com.google.android.material.textfield.TextInputLayout>

    <TextView
        android:id="@+id/phoneTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="32dp"
        android:layout_marginTop="16dp"
        android:fontFamily="@font/montserrat"
        android:text="Phone Number"
        android:textColor="@color/black"
        android:textSize="16sp" />
```

```
<com.google.android.material.textfield.TextInputLayout
    android:id="@+id/phoneEditTextLayout"

    style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginStart="32dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="32dp"
        app:boxCornerRadiusBottomEnd="20dp"
        app:boxCornerRadiusBottomStart="20dp"
        app:boxCornerRadiusTopEnd="20dp"
        app:boxCornerRadiusTopStart="20dp"

    app:startIconDrawable="@drawable/baseline_phone_24">

    <EditText
        android:id="@+id/phoneEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="phone"
        tools:ignore="SpeakableTextPresentCheck" />

</com.google.android.material.textfield.TextInputLayout>

<Button
    android:id="@+id/saveChanges"
    android:layout_width="match_parent"
    android:layout_height="60dp"
    android:layout_below="@+id/phoneEditTextLayout"
    android:layout_centerHorizontal="true"
    android:layout_gravity="center"
    android:layout_marginStart="32dp"
    android:layout_marginLeft="32dp"
    android:layout_marginTop="24dp"
    android:layout_marginEnd="32dp"
    android:layout_marginRight="32dp"
    android:layout_marginBottom="100dp"
    android:fontFamily="@font/montserratsemibold"
    android:text="Save Changes"
    android:textSize="18sp"

    app:layout_constraintBottom_toTopOf="@+id/bottom_navigation"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintTop_toBottomOf="parent" />

</LinearLayout>

</androidx.core.widget.NestedScrollView>
</androidx.coordinatorlayout.widget.CoordinatorLayout>
```

```
</FrameLayout>

[File XML activity_settings.xml]
<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".view.profile.SettingsActivity">

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/appToolbar"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar">

        <com.google.android.material.appbar.CollapsingToolbarLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            app:layout_scrollFlags="scroll|exitUntilCollapsed">

            <androidx.appcompat.widget.Toolbar
                android:id="@+id/toolbar"
                android:layout_width="match_parent"
                android:layout_height="?attr/actionBarSize"
                android:background="?attr/colorPrimary"
                android:elevation="4dp"
                app:layout_scrollFlags="scroll|enterAlways">

                <TextView
                    android:id="@+id/toolbar_title"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:fontFamily="@font/montserratsemibold"
                    android:text="Account Settings"
                    android:textColor="@color/white"
                    android:textSize="18sp" />

            </androidx.appcompat.widget.Toolbar>
        </com.google.android.material.appbar.CollapsingToolbarLayout>
    </com.google.android.material.appbar.AppBarLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="fill_vertical"
        android:layout_marginBottom="122dp"
        android:orientation="vertical"
        app:layout_behavior="@string/appbar_scrolling_view_behavior">

        <RelativeLayout
```

```
        android:id="@+id/resetPassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="16dp"
        android:gravity="center_horizontal"
        android:paddingTop="16dp"
        android:paddingBottom="16dp">

    <LinearLayout
        android:id="@+id/leftAlignedBookmark"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentStart="true"
        android:layout_marginStart="12dp"
        android:gravity="start"
        android:orientation="horizontal">

        <ImageView
            android:layout_width="30dp"
            android:layout_height="30dp"
            android:layout_gravity="center_vertical"
            android:src="@drawable/ic_resetpassword" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_vertical"
            android:layout_marginStart="16dp"
            android:fontFamily="@font/montserrat"
            android:text="Reset Password"
            android:textColor="@color/black"
            android:textSize="18sp" />

    
```

</LinearLayout>

```
    <LinearLayout
        android:id="@+id/rightAlignedArrow"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_marginEnd="12dp"
        android:gravity="end"
        android:orientation="horizontal">

        <ImageView
            android:layout_width="25dp"
            android:layout_height="20dp"
            android:contentDescription="arrow"
            android:src="@drawable/ic_arrow" />
```

```
</LinearLayout>

</RelativeLayout>

<TextView
    android:id="@+id/passwordTextView"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="24dp"
    android:layout_marginTop="24dp"
    android:layout_marginEnd="24dp"
    android:fontFamily="@font/montserrat"
    android:text="Enter Current Password"
    android:textColor="@color/black"
    android:textSize="16sp"
    android:visibility="gone" />

<com.google.android.material.textfield.TextInputLayout
    android:id="@+id/currentPasswordTextLayout"

    style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="24dp"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="24dp"
    android:visibility="gone"
    app:boxCornerRadiusBottomEnd="20dp"
    app:boxCornerRadiusBottomStart="20dp"
    app:boxCornerRadiusTopEnd="20dp"
    app:boxCornerRadiusTopStart="20dp"
    app:endIconMode="password_toggle"
    app:startIconDrawable="@drawable/baseline_lock_24">

    <com.app.famcare.view.customview.PasswordEditText
        android:id="@+id/currentPasswordEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPassword"
        tools:ignore="SpeakableTextPresentCheck" />
</com.google.android.material.textfield.TextInputLayout>

<Button
    android:id="@+id/resetButton"
    android:layout_width="match_parent"
    android:layout_height="60dp"
    android:layout_below="@+id/phoneEditTextLayout"
    android:layout_centerHorizontal="true"
    android:layout_gravity="center"
    android:layout_marginStart="32dp"
    android:layout_marginTop="30dp"
```

```

        android:layout_marginEnd="32dp"
        android:layout_marginBottom="100dp"
        android:fontFamily="@font/montserratsemibold"
        android:text="Reset Password"
        android:textSize="16sp"
        android:visibility="gone"

    app:layout_constraintBottom_toTopOf="@+id/bottom_navigation"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintTop_toBottomOf="parent" />

</LinearLayout>

</androidx.coordinatorlayout.widget.CoordinatorLayout>

```

[File XML activity_privacy_policy.xml]

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".view.profile.PrivacyPolicyActivity">

    <com.google.android.material.appbar.AppBarLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar">

        <com.google.android.material.appbar.CollapsingToolbarLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            app:layout_scrollFlags="scroll|exitUntilCollapsed">

            <androidx.appcompat.widget.Toolbar
                android:id="@+id/toolbar"
                android:layout_width="match_parent"
                android:layout_height="?attr/actionBarSize"
                android:background="?attr/colorPrimary"
                android:elevation="4dp"
                app:layout_scrollFlags="scroll|enterAlways">

                <TextView
                    android:id="@+id/toolbar_title"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:fontFamily="@font/montserratsemibold"
                    android:text="Privacy Policy"
                    android:textColor="@color/white"
                    android:textSize="18sp" />

            </androidx.appcompat.widget.Toolbar>

```

```
</com.google.android.material.appbar.CollapsingToolbarLayout>

</com.google.android.material.appbar.AppBarLayout>

<androidx.core.widget.NestedScrollView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="fill_vertical"
    android:layout_marginBottom="72dp"
    android:fillViewport="true"
    app:layout_behavior="@string/appbar_scrolling_view_behavior">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center_vertical"
            android:orientation="vertical"
            android:paddingTop="16dp"
            android:paddingBottom="16dp">

            <ImageView
                android:layout_width="220dp"
                android:layout_height="100dp"
                android:layout_gravity="center"
                android:contentDescription="logo"
                android:src="@drawable/logo_toolbar_fit" />

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_marginStart="20dp"
                android:fontFamily="@font/montserratsemibold"
                android:text="Privacy Policy"
                android:textColor="@android:color/black"
                android:textSize="20sp"
                android:textStyle="bold" />

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_marginStart="20dp"
                android:fontFamily="@font/montserrat"
                android:paddingTop="4dp"
                android:text="© FamCare"
                android:textColor="@android:color/black"
                android:textSize="16sp" />

```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:orientation="vertical"
    android:paddingStart="20dp">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginEnd="20dp"
        android:layout_marginBottom="16dp"
        android:fontFamily="@font/montserrat"
        android:text="1. Information We
Collect\nFamCare collects personal information when you use our app,
such as your name, email address, and phone number. We may also
collect information about your device, location, and usage of the
app."
        android:textSize="18sp" />

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginEnd="20dp"
        android:layout_marginBottom="16dp"
        android:fontFamily="@font/montserrat"
        android:text="2. How We Use Your
Information\nWe use your information to provide and improve the
FamCare app, including connecting you with caregivers, customizing
your experience, and sending you important updates and
notifications."
        android:textSize="18sp" />

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginEnd="20dp"
        android:layout_marginBottom="16dp"
        android:fontFamily="@font/montserrat"
        android:text="3. Information Sharing and
Disclosure\nWe may share your information with caregivers to
facilitate the services you request. We may also share aggregated
and anonymized data for analytics purposes."
        android:textSize="18sp" />

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginEnd="20dp"
        android:layout_marginBottom="16dp"
        android:fontFamily="@font/montserrat"
```

```
        android:text="4. Data Security\nFamCare takes  
measures to protect your information, but no method of transmission  
over the internet or electronic storage is 100% secure. We cannot  
guarantee absolute security of your data."  
        android:textSize="18sp" />  
  
<TextView  
        android:layout_width="match_parent"  
        android:layout_height="wrap_content"  
        android:layout_marginEnd="20dp"  
        android:layout_marginBottom="16dp"  
        android:fontFamily="@font/montserrat"  
        android:text="5. Your Choices\nYou can manage  
your information within the app, including updating your profile and  
preferences. You can also choose to delete your account, which will  
remove all associated data."  
        android:textSize="18sp" />  
  
<TextView  
        android:layout_width="match_parent"  
        android:layout_height="wrap_content"  
        android:layout_marginEnd="20dp"  
        android:layout_marginBottom="16dp"  
        android:fontFamily="@font/montserrat"  
        android:text="6. Changes to this Policy\nWe  
may update this Privacy Policy from time to time. We will notify you  
of any significant changes by email or within the app. By continuing  
to use FamCare, you agree to the updated policy."  
        android:textSize="18sp" />  
  
<TextView  
        android:layout_width="match_parent"  
        android:layout_height="wrap_content"  
        android:layout_marginEnd="20dp"  
        android:layout_marginBottom="16dp"  
        android:fontFamily="@font/montserrat"  
        android:text="7. Contact Us\nIf you have any  
questions or concerns about this Privacy Policy, please contact us  
at support@famcare.com."  
        android:textSize="18sp" />  
  
</LinearLayout>  
  
<TextView  
        android:layout_width="wrap_content"  
        android:layout_height="wrap_content"  
        android:layout_gravity="center"  
        android:layout_marginTop="20dp"  
        android:fontFamily="@font/montserratsemibold"  
        android:gravity="center_vertical"  
        android:paddingStart="20dp"  
        android:paddingTop="8dp"  
        android:paddingEnd="20dp"
```

```

        android:text="By using FamCare, you are
indicating your"
        android:textColor="@android:color/darker_gray"
        android:textSize="14sp" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_marginTop="5dp"
        android:fontFamily="@font/montserratsemibold"
        android:gravity="center_vertical"
        android:paddingStart="20dp"
        android:paddingTop="0dp"
        android:paddingEnd="20dp"
        android:text="acceptance of this Privacy Policy."
        android:textColor="@android:color/darker_gray"
        android:textSize="14sp" />

    </LinearLayout>

</LinearLayout>

</androidx.core.widget.NestedScrollView>
</androidx.coordinatorlayout.widget.CoordinatorLayout>
```

[File XML activity_terms_of_service.xml]

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".view.profile.TermsOfServicesActivity">

    <com.google.android.material.appbar.AppBarLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar">

        <com.google.android.material.appbar.CollapsingToolbarLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            app:layout_scrollFlags="scroll|exitUntilCollapsed">

            <androidx.appcompat.widget.Toolbar
                android:id="@+id/toolbar"
                android:layout_width="match_parent"
                android:layout_height="?attr/actionBarSize"
                android:background="?attr/colorPrimary"
                android:elevation="4dp"
```

```
    app:layout_scrollFlags="scroll|enterAlways">

    <TextView
        android:id="@+id/toolbar_title"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:fontFamily="@font/montserratsemibold"
        android:text="Terms of Services"
        android:textColor="@color/white"
        android:textSize="18sp" />

</androidx.appcompat.widget.Toolbar>

</com.google.android.material.appbar.CollapsingToolbarLayout>

</com.google.android.material.appbar.AppBarLayout>

<androidx.core.widget.NestedScrollView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="fill_vertical"
    android:layout_marginBottom="72dp"
    android:fillViewport="true"
    app:layout_behavior="@string/appbar_scrolling_view_behavior">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"

        android:orientation="vertical">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center_vertical"
            android:orientation="vertical"
            android:paddingTop="16dp"
            android:paddingBottom="16dp">

            <ImageView
                android:layout_width="220dp"
                android:layout_height="100dp"
                android:layout_gravity="center"
                android:contentDescription="logo"
                android:src="@drawable/logo_toolbar_fit" />

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_marginStart="20dp"
                android:fontFamily="@font/montserratsemibold"
                android:text="Terms of Services"
```

```
        android:textColor="@android:color/black"
        android:textSize="20sp"
        android:textStyle="bold" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="20dp"
        android:fontFamily="@font/montserrat"
        android:paddingTop="4dp"
        android:text="© FamCare"
        android:textColor="@android:color/black"
        android:textSize="16sp" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:orientation="vertical"
        android:paddingStart="20dp">

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginEnd="20dp"
            android:layout_marginBottom="16dp"
            android:fontFamily="@font/montserrat"
            android:text="FamCare is your go-to solution
for finding trustworthy caregivers for your little ones and elderly
family members."
            android:textSize="18sp" />

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginEnd="20dp"
            android:layout_marginBottom="16dp"
            android:fontFamily="@font/montserrat"
            android:text="Our app simplifies the process
of connecting with qualified nannies who are ready to provide
top-notch care. Whether you're a parent in search of a reliable
nanny for your toddler or an individual seeking compassionate
assistance for an elderly loved one, FamCare has you covered."
            android:textSize="18sp" />

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginEnd="20dp"
            android:layout_marginBottom="16dp"
            android:fontFamily="@font/montserrat"
```

```
        android:text="With a user-friendly interface,  
users can easily browse through profiles of skilled and vetted  
caregivers, complete with reviews and ratings from other users."  
        android:textSize="18sp" />  
  
<TextView  
        android:layout_width="match_parent"  
        android:layout_height="wrap_content"  
        android:layout_marginEnd="20dp"  
        android:layout_marginBottom="16dp"  
        android:fontFamily="@font/montserrat"  
        android:text="What sets FamCare apart is its  
innovative use of geolocation through integrated maps. Users can  
pinpoint nearby daycares and senior care facilities, making it  
convenient to explore options in their immediate vicinity."  
        android:textSize="18sp" />  
  
<TextView  
        android:layout_width="match_parent"  
        android:layout_height="wrap_content"  
        android:layout_marginEnd="20dp"  
        android:layout_marginBottom="16dp"  
        android:fontFamily="@font/montserrat"  
        android:text="1. Acceptance of Terms\nBy  
using FamCare, you agree to be bound by these Terms of Service, our  
Privacy Policy, and any other terms and conditions referenced  
herein."  
        android:textSize="18sp" />  
  
<TextView  
        android:layout_width="match_parent"  
        android:layout_height="wrap_content"  
        android:layout_marginEnd="20dp"  
        android:layout_marginBottom="16dp"  
        android:fontFamily="@font/montserrat"  
        android:text="2. Use of the Service\nFamCare  
provides a platform for connecting caregivers with families in need  
of care services. The service is intended for personal,  
non-commercial use."  
        android:textSize="18sp" />  
  
<TextView  
        android:layout_width="match_parent"  
        android:layout_height="wrap_content"  
        android:layout_marginEnd="20dp"  
        android:layout_marginBottom="16dp"  
        android:fontFamily="@font/montserrat"  
        android:text="3. User Accounts\nUsers may  
create accounts to access additional features of the service. You  
are responsible for maintaining the confidentiality of your account  
credentials."  
        android:textSize="18sp" />
```

```

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginEnd="20dp"
    android:layout_marginBottom="16dp"
    android:fontFamily="@font/montserrat"
        android:text="4. Content\nUsers may submit
content such as reviews and ratings. By submitting content, you
grant FamCare the right to use, modify, and display the content in
connection with the service."
        android:textSize="18sp" />

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginEnd="20dp"
    android:layout_marginBottom="16dp"
    android:fontFamily="@font/montserrat"
        android:text="5. Limitation of
Liability\nFamCare is not liable for any damages or losses arising
from the use of the service. Users use the service at their own
risk."
        android:textSize="18sp" />

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginEnd="20dp"
    android:layout_marginBottom="16dp"
    android:fontFamily="@font/montserrat"
        android:text="6. Contact Us\nIf you have any
questions or concerns about these Terms of Service, please contact
us at support@famcare.com."
        android:textSize="18sp" />

</LinearLayout>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:layout_marginTop="20dp"
    android:fontFamily="@font/montserratsemibold"
    android:gravity="center_vertical"
    android:paddingStart="20dp"
    android:paddingTop="8dp"
    android:paddingEnd="20dp"
        android:text="By using FamCare, you acknowledge
and"
        android:textColor="@android:color/darker_gray"
        android:textSize="14sp" />

<TextView

```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_marginTop="5dp"
        android:fontFamily="@font/montserratsemibold"
        android:gravity="center_vertical"
        android:paddingStart="20dp"
        android:paddingTop="0dp"
        android:paddingEnd="20dp"
        android:text="agree to these Terms of Services."
        android:textColor="@android:color/darker_gray"
        android:textSize="14sp" />

    </LinearLayout>

</LinearLayout>

</androidx.core.widget.NestedScrollView>
</androidx.coordinatorlayout.widget.CoordinatorLayout>
```

[File XML activity_about.xml]

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".view.profile.AboutActivity">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <com.google.android.material.appbar.AppBarLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"

            android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar">

            <androidx.appcompat.widget.Toolbar
                android:id="@+id/toolbar"
                android:layout_width="match_parent"
                android:layout_height="?attr/actionBarSize"
                android:background="?attr/colorPrimary"
                android:elevation="4dp"
                app:layout_scrollFlags="scroll|enterAlways">

                <TextView
                    android:id="@+id/toolbar_title"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
```

```
        android:fontFamily="@font/montserratsemibold"
        android:text="About"
        android:textColor="@color/white"
        android:textSize="18sp" />

    </androidx.appcompat.widget.Toolbar>

</com.google.android.material.appbar.AppBarLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center_vertical"
    android:orientation="vertical"
    android:paddingTop="16dp"
    android:paddingBottom="16dp">

    <ImageView
        android:layout_width="220dp"
        android:layout_height="100dp"
        android:layout_gravity="center"
        android:contentDescription="logo"
        android:src="@drawable/logo_toolbar_fit" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="20dp"
        android:fontFamily="@font/montserratsemibold"
        android:text="About"
        android:textColor="@android:color/black"
        android:textSize="20sp"
        android:textStyle="bold" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="20dp"
        android:fontFamily="@font/montserrat"
        android:paddingTop="4dp"
        android:text="Version 1.0"
        android:textColor="@android:color/black"
        android:textSize="16sp" />

    <com.codesgood.views.JustifiedTextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="15dp"
        android:fontFamily="@font/montserrat"
        android:paddingStart="20dp"
        android:paddingTop="4dp"
        android:paddingEnd="16dp"
```

```
        android:text="@string/desc_app"
        android:textColor="@android:color/black"
        android:textSize="18sp" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_marginTop="12dp"
        android:fontFamily="@font/montserratsemibold"
        android:paddingStart="20dp"
        android:paddingTop="6dp"
        android:text="© 2024 MobProg Team 5. All rights
reserved."
        android:textColor="@android:color/darker_gray"
        android:textSize="16sp" />

    </LinearLayout>

</LinearLayout>

</androidx.coordinatorlayout.widget.CoordinatorLayout>
```

Penjelasan:

ProfileActivity bertanggung jawab menampilkan dan mengelola profil pengguna dalam aplikasi. Aktivitas ini mengambil data pengguna dari koleksi Firestore dan menampilkannya, termasuk nama, email, nomor telepon, dan foto profil. Di sebelah data pengguna, terdapat button bergambar pensil yang akan mengarahkan ke halaman edit profile apabila ditekan. Pada halaman edit profil, pengguna dapat melakukan perubahan data pengguna seperti nama lengkap, jenis kelamin, tanggal lahir, alamat, nomor telepon, dan juga foto profil. Apabila pengguna ingin mengganti foto profil, maka aplikasi akan meminta izin untuk mengakses galeri handphone milik pengguna. Setelah pengguna menekan tombol Save Changes di halaman edit profile, maka aktivitas akan menjalankan fungsi saveUserDataToFirestore untuk mengupdate data pengguna yang ada di dalam koleksi Firestore. Pada halaman profil ini juga terdapat menu settings, di mana pada menu ini pengguna dapat mengubah password milik pengguna dengan cara mengisikan password yang sekarang. Apabila password yang diisikan sudah benar, dan pengguna menekan button reset password, maka sebuah email reset password akan dikirimkan ke email pengguna. Melalui email tersebut, pengguna dapat melakukan pengubahan password akun FamCare. Selain itu, melalui halaman profil ini, pengguna juga dapat mengakses halaman lain seperti Terms of Service, Privacy Policy, dan About dengan cara menekan CardView yang tersedia. Terms of Service adalah halaman yang berisi syarat dan ketentuan penggunaan aplikasi, Halaman Privacy Policy menjelaskan bagaimana data pengguna dikumpulkan, digunakan, dan dilindungi, serta Halaman About memberikan informasi tentang aplikasi dan pengembangnya. Halaman ProfileActivity ini juga menyediakan opsi

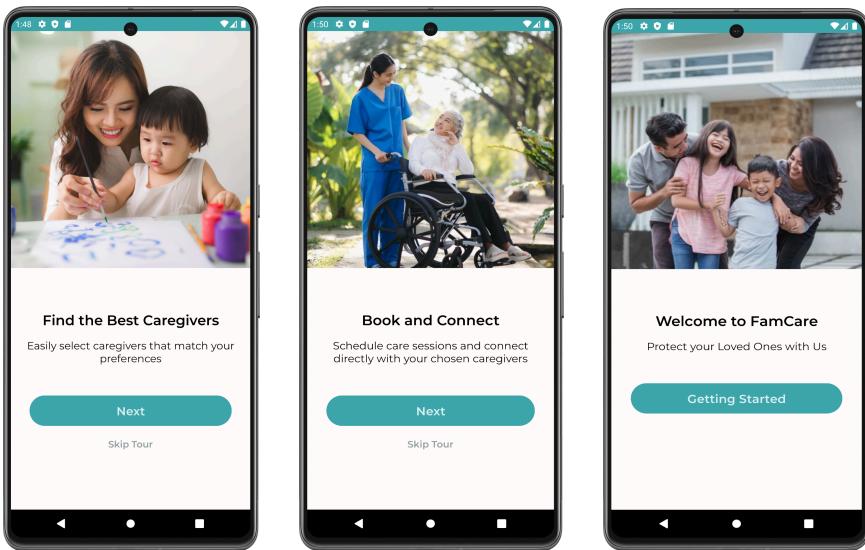
untuk logout dengan menampilkan dialog konfirmasi yang mengarahkan pengguna ke LoginActivity apabila pengguna memutuskan untuk log out dari aplikasi.

2. 3 Tampilan

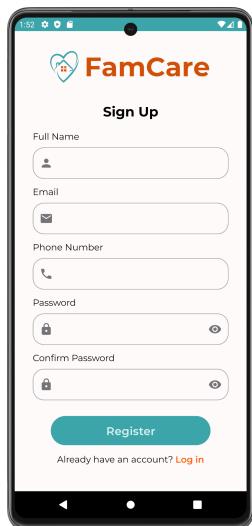
2.3.1 Splash Screen



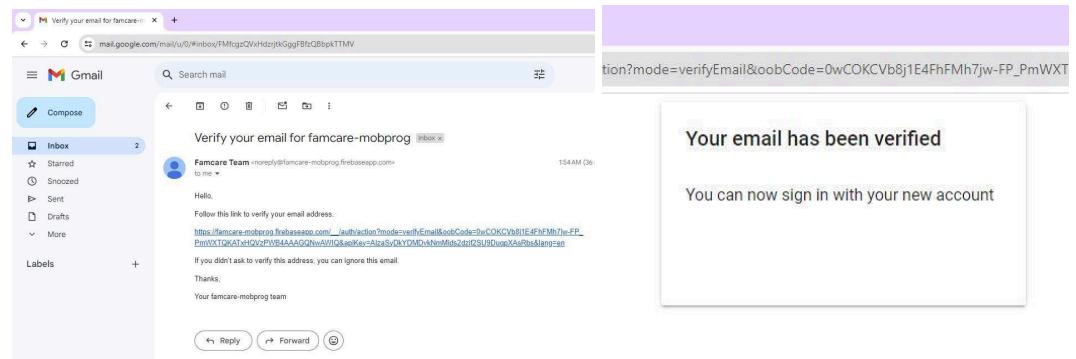
2.3.2 OnBoarding



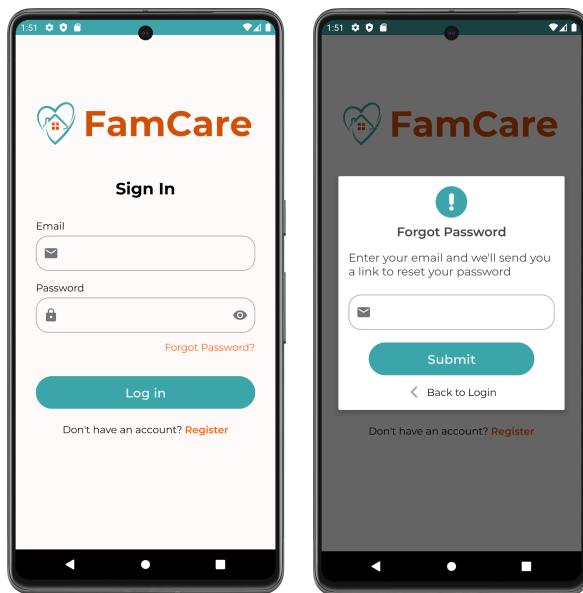
2.3.3 Register



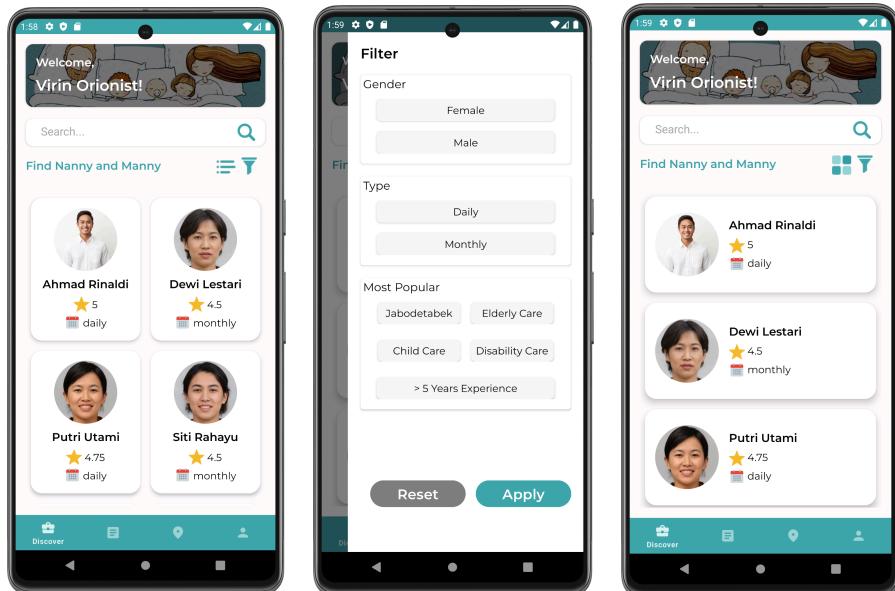
Tampilan konfirmasi email registrasi:



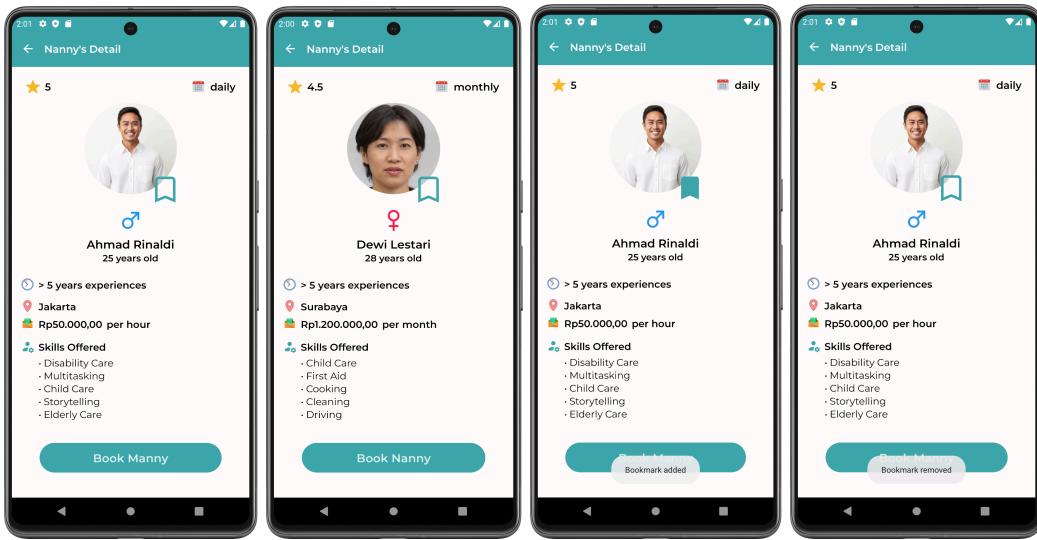
2.3.4 Log In



2.3.5 Discover

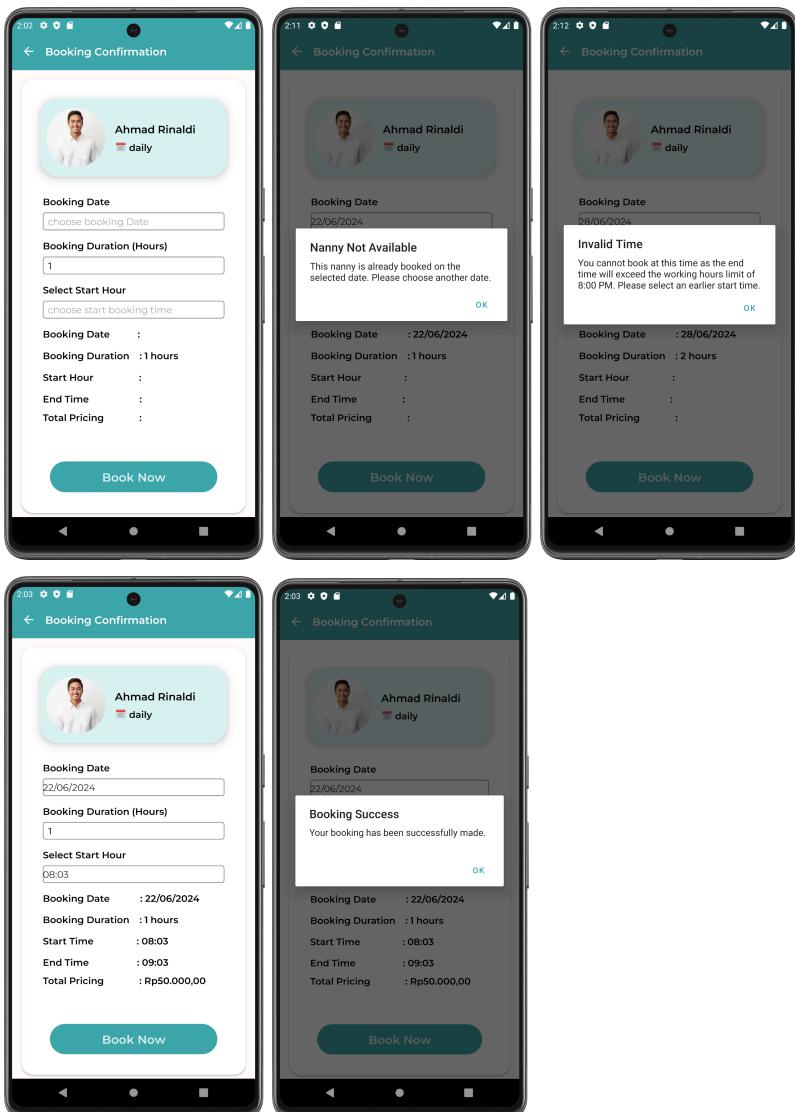


2.3.6 Detail Post

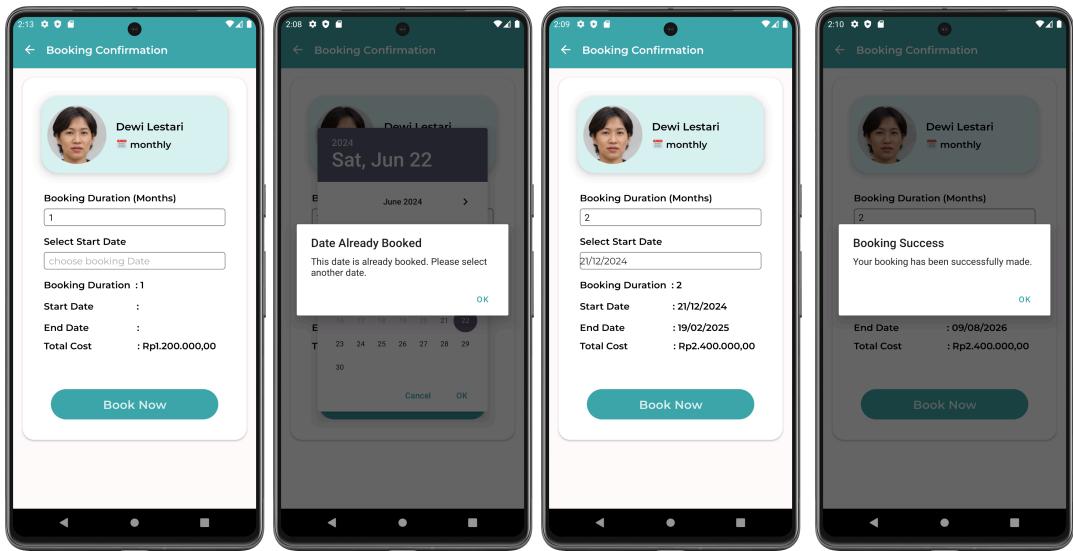


2.3.7 Booking

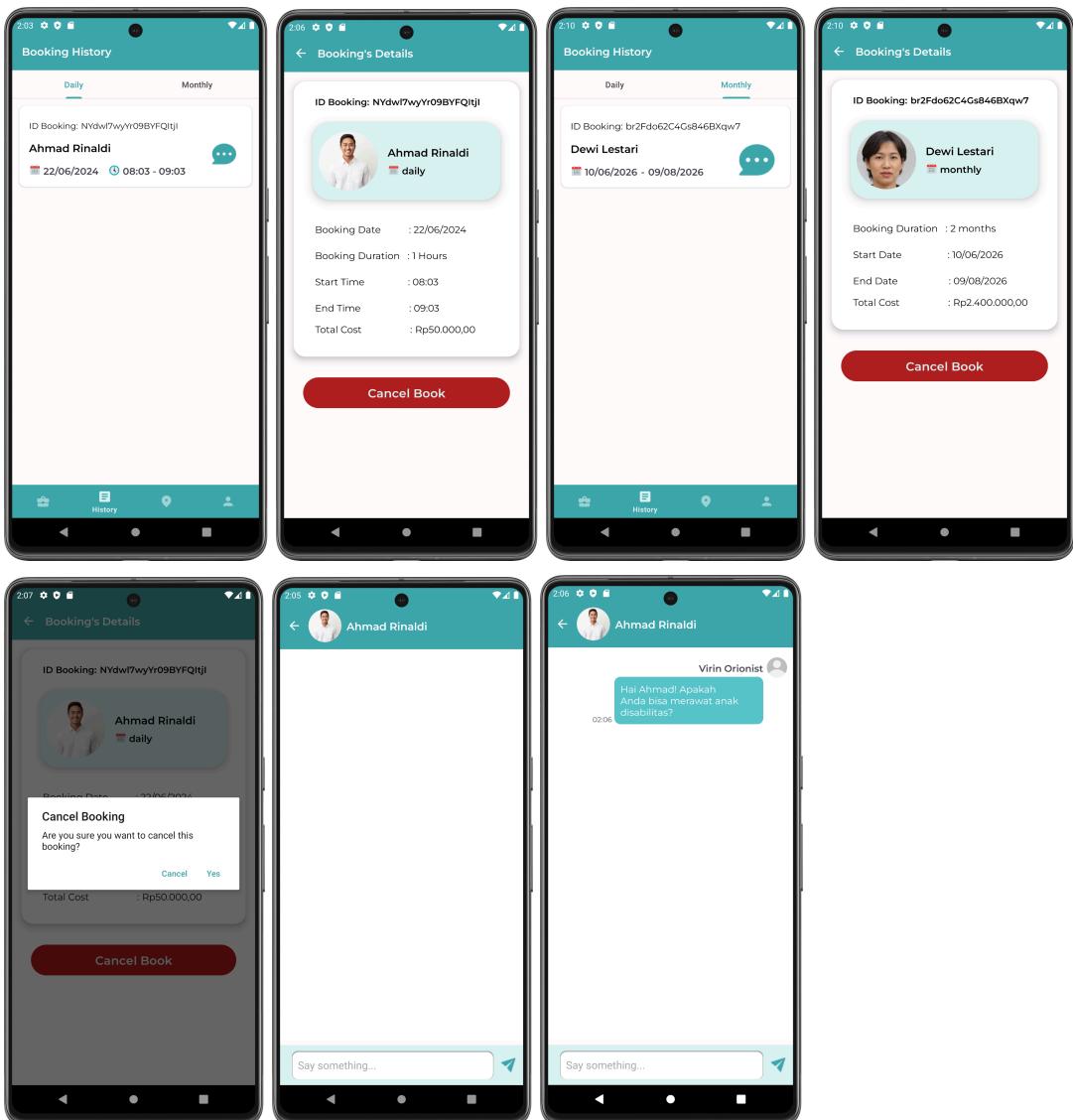
a) Booking Daily



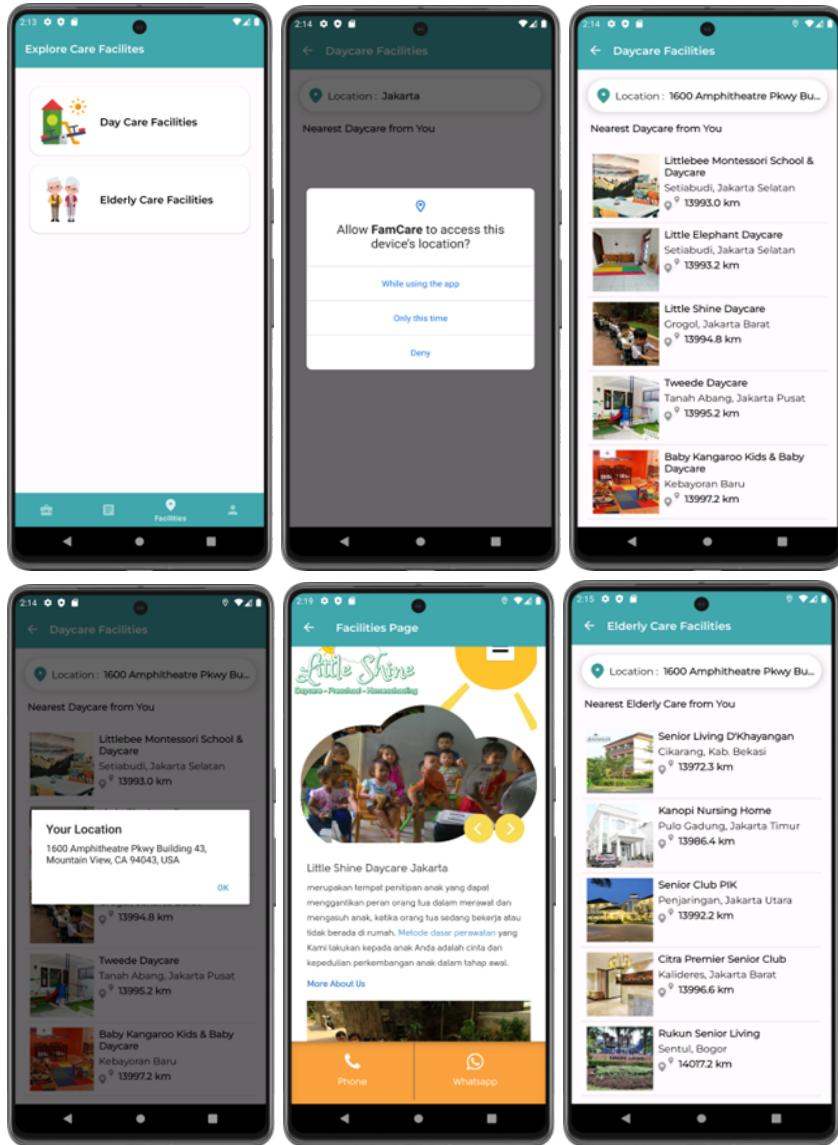
b) Booking Monthly



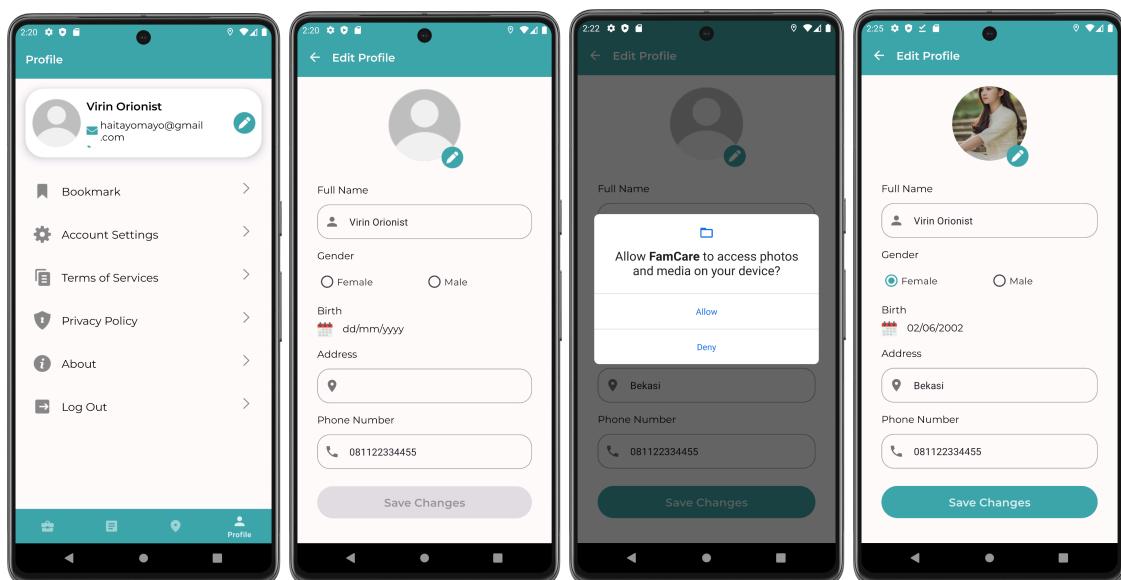
2.3.8 History dan Chat

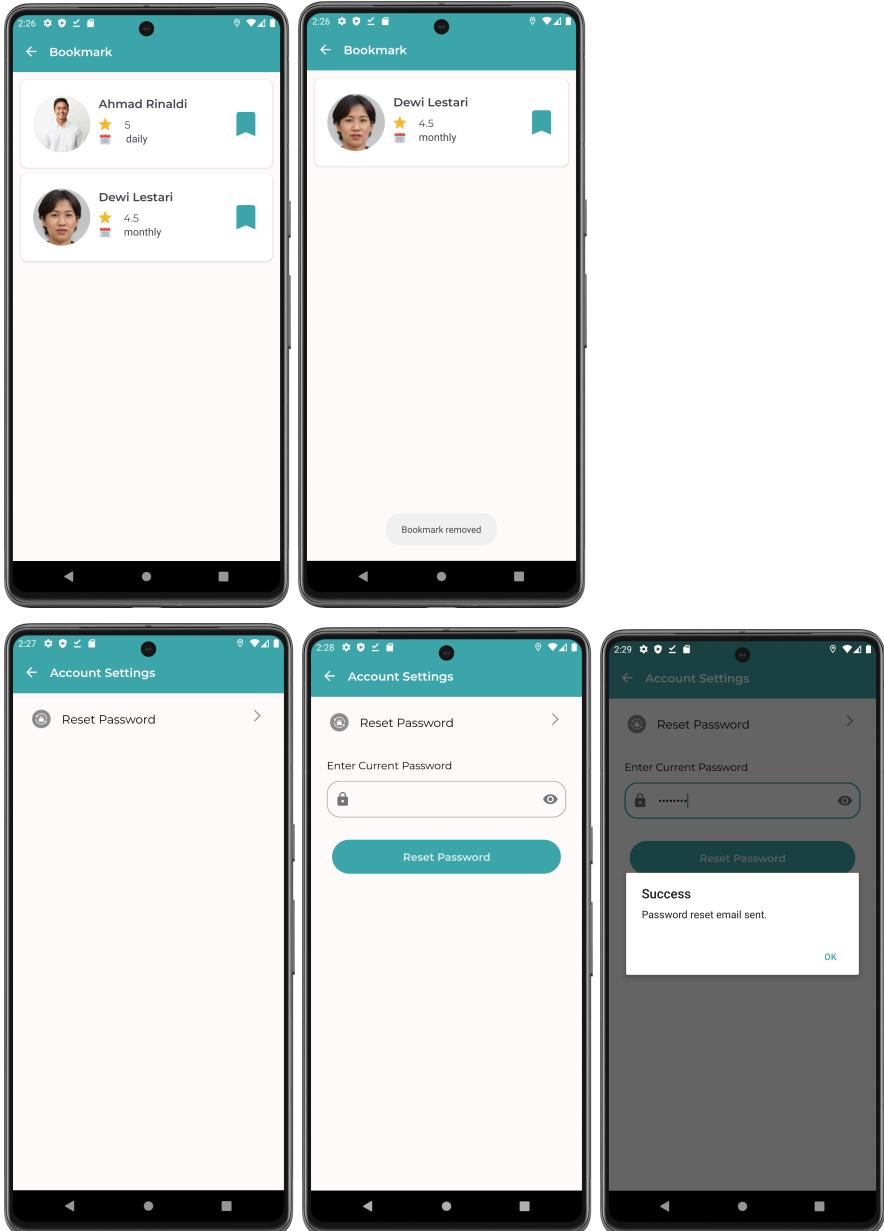


2.3.9 Facilities

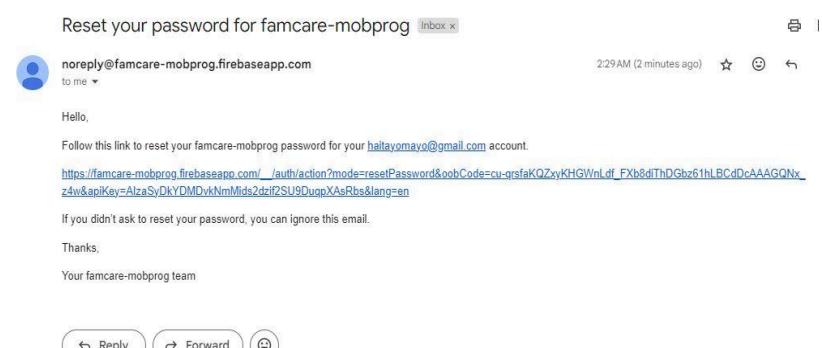


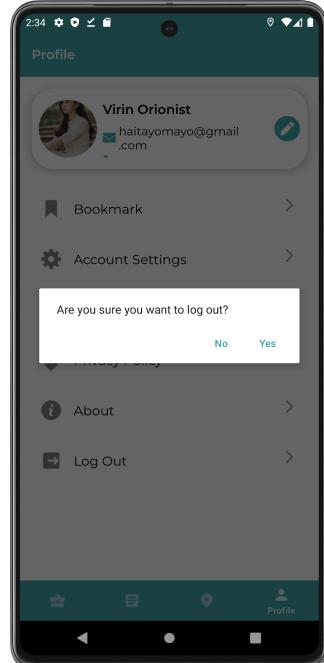
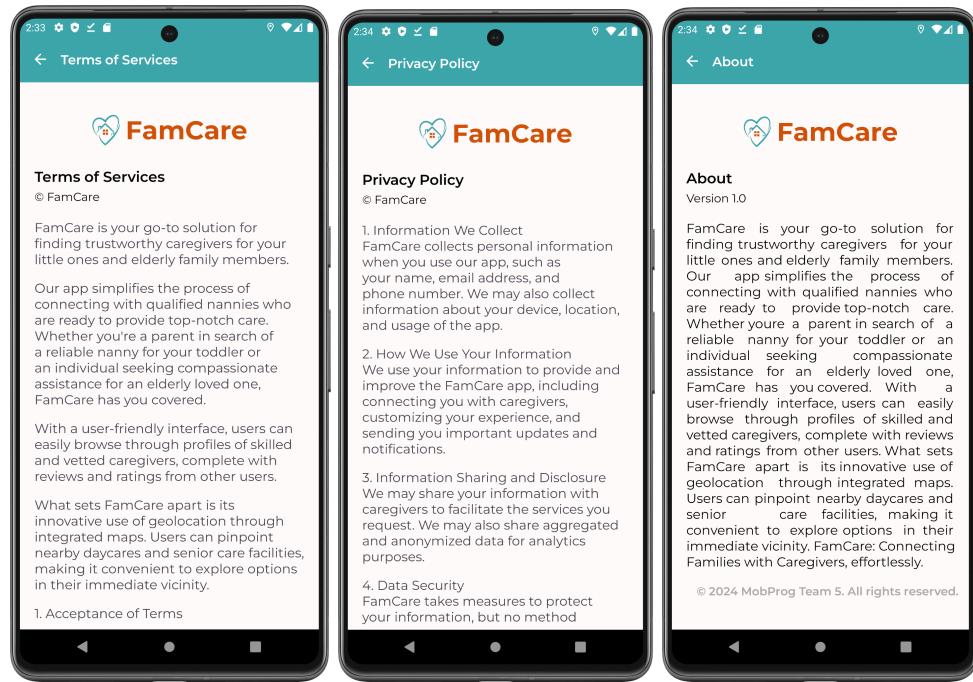
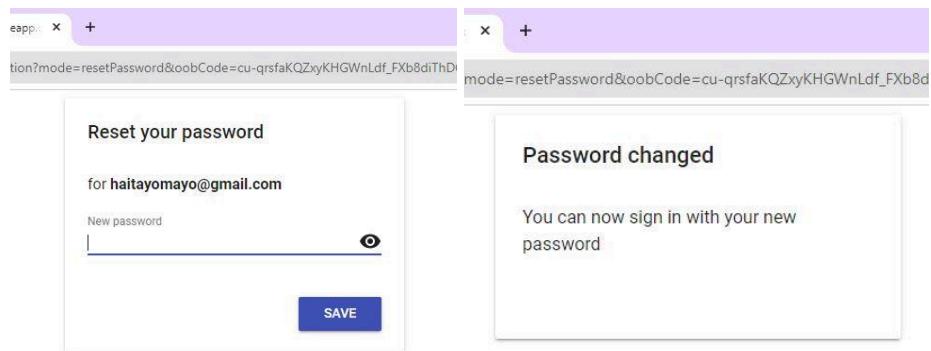
2.3.10 Profile





Tampilan email konfirmasi reset password:





BAB III

PENUTUP

3.1 Kesimpulan

Dalam era modern dengan gaya hidup yang semakin sibuk dan mobilitas yang tinggi, tantangan dalam merawat anggota keluarga yang membutuhkan perhatian khusus, seperti bayi, anak-anak, dan lansia, menjadi semakin kompleks. FamCare hadir sebagai solusi berbasis teknologi untuk mengatasi permasalahan ini melalui pengembangan aplikasi berbasis Android. Aplikasi ini dikembangkan menggunakan bahasa pemrograman Kotlin, yang dikenal sebagai bahasa pemrograman native yang didukung oleh Google. Kelebihan dari penggunaan bahasa Kotlin dalam pengembangan app meliputi kompatibilitas dengan Java, bahasa pemrograman native yang didukung secara resmi oleh Google, integrasi yang baik dengan perangkat API Android terbaru.

Secara teknis, FamCare dirancang dengan arsitektur yang memastikan stabilitas dan responsivitas aplikasi. FamCare memungkinkan pengguna untuk menemukan pengasuh sesuai preferensi, membuat reservasi jadwal dengan mudah, dan terhubung langsung dengan pengasuh melalui fitur chat. Selain itu, pengguna juga dapat menemukan lokasi fasilitas perawatan anak maupun orang tua melalui integrasi fitur geolokasi. Keamanan dan privasi data pengguna merupakan prioritas utama, dengan implementasi protokol enkripsi dan autentikasi yang kuat untuk melindungi informasi pribadi pengguna. Sistem pencarian dan filter pengasuh dikembangkan menggunakan algoritma yang efisien untuk memastikan hasil yang relevan dan cepat.

Secara keseluruhan, FamCare tidak hanya menawarkan solusi praktis untuk mencari pengasuh, tetapi juga memastikan bahwa setiap aspek teknis dari aplikasi ini dirancang untuk memberikan pengalaman terbaik bagi pengguna. Dengan demikian, FamCare berkontribusi signifikan dalam mengurangi beban pikiran orang tua dan meningkatkan kualitas hidup keluarga melalui penggunaan teknologi yang canggih dan aman.

3.2 Saran

Dalam mengembangkan aplikasi FamCare ke depan, kami menyarankan beberapa hal berikut:

- 1) Memperbarui dan meningkatkan kualitas database, memastikan bahwa database terus diperbarui dan ditingkatkan kualitasnya.
- 2) Melakukan strategi pemasaran dan promosi yang efektif dan lebih luas untuk meningkatkan kesadaran masyarakat tentang keberadaan dan manfaat FamCare untuk menjangkau target pengguna.
- 3) Memastikan bahwa keamanan dan privasi pengguna FamCare tetap terjaga dengan baik, termasuk perlindungan data pribadi pengguna aplikasi.
- 4) Melakukan pengujian pengguna untuk memastikan aplikasi mudah digunakan dan memenuhi kebutuhan pengguna.

- 5) Memantau dan menganalisis umpan balik dari pengguna untuk terus memahami kebutuhan pengguna dan melakukan perbaikan sesuai dengan kebutuhan dan harapan pengguna.

DAFTAR PUSTAKA

- Ilmi, H. Z., & Qhoiriah, A. (2016). Perancangan sistem informasi pencarian asisten rumah tangga berbasis web. *Jurnal Manajemen Informatika*, 1(6), 99–107.
- Indah, R., et al. (2021). Kesepian dan konflik dalam keluarga. *Jurnal Sosial dan Budaya*, 20(3), 201-215.
- Kementerian Pemberdayaan Perempuan dan Perlindungan Anak Republik Indonesia. (2020). Dampak kesibukan orang tua terhadap perawatan anak. KPPPA RI.
- Nurhidayati, R., & Suparman, U. (2020). Pengaruh stres kerja pada produktivitas orang tua. *Jurnal Psikologi Indonesia*, 9(1), 54-66.
- Putri, R., & Santoso, A. (2022). Ketersediaan informasi layanan pengasuhan di Indonesia. *Jurnal Sosiologi Indonesia*, 23(1), 89-104.
- Safitri, A. E., & Gilang, A. (2020). Pengaruh stres kerja terhadap produktivitas kerja karyawan pada PT. Telkom Witel Bekasi. *Jurnal Ilmiah MEA (Manajemen, Ekonomi, dan Akuntansi)*.
- Setiawan, E., & Nugroho, A. (2023). Teknologi dalam pencarian jasa pengasuh. *Jurnal Teknologi Informasi*, 15(2), 178-192.

LAMPIRAN

Link Google Drive Project Zip: [Project ZIP Famcare](#)

Link Google Drive APK: [Famcare APK](#)

Link Video Demo Penggunaan Aplikasi: [Video Demo FamCare](#)

Link Dokumentasi GitHub: [GitHub FamCare](#)