

UTS ESSAY DAN STUDI KASUS

PEMOGRAMAN MOBILE

Dosen Pengampu : Nova Agustina, ST., M.Kom.



Disusun oleh:

Mochamad Alphin

23552011067

TIF RP 223 CNS A

UNIVERSITAS TEKNOLOGI BANDUNG

TEKNIK INFORMATIKA

2025

Soal Essay :

1. Apa fungsi findViewById?
2. Apa syarat pemanggilan method findViewById? Buat contohnya dan screenshot source code nya!
3. Error apa yang terjadi jika file kotlin salah menginisialisasi findViewById atau objek pada xml belum diinisialisasi?
4. Buat sebuah contoh program untuk menampilkan pesan error Resources.NotFoundException! Screenshot logcat-nya!
5. Kumpulkan dalam bentuk pdf di Elearning (Soal essay digabung dengan soal studi kasus cek point 7 Studi Kasus)

Jawaban :

1. **findViewById** adalah method yang digunakan di Android (Java/Kotlin) untuk mengambil referensi ke komponen UI (seperti Button, TextView, EditText, dll) yang didefinisikan di file XML layout. Dengan menggunakan findViewById, kita bisa memanipulasi komponen tersebut di dalam kode Kotlin atau Java (misalnya mengatur teks, menangani klik, dll).
2.
 - **Layout harus sudah di-inflate** terlebih dahulu dengan setContentView().
 - ID yang dipanggil lewat R.id.nama_view **harus ada** di layout XML yang sedang digunakan.
 - Pemanggilan findViewById harus berada di dalam konteks yang memiliki akses ke view hierarchy (misalnya: Activity, Fragment, atau View).
 - **ID view harus sudah dibuat dan didefinisikan di layout XML** menggunakan atribut android:id.

Id :

android:id="@+id/myButton"

```
<Button  
    android:id="@+id/myButton"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="Click Me"  
    tools:ignore="HardcodedText,MissingConstraints" />
```

Pemanggilan berdasarkan id :

```
val button = findViewById<Button>(R.id.myButton)

></> class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        val button = findViewById<Button>(R.id.myButton)

        button.setOnClickListener {
            Toast.makeText(context, text: "Button clicked!", Toast.LENGTH_SHORT).show()
        }
    }
}
```

3. • **NullPointerException**: Terjadi ketika Anda memanggil metode atau properti dari objek yang belum diinisialisasi (misalnya findViewById mengembalikan null).
 - **ClassCastException**: Terjadi ketika Anda melakukan casting ke tipe yang salah, misalnya findViewById<TextView>(R.id.button1) padahal sebenarnya button1 adalah Button.
 - **Resources.NotFoundException**: Terjadi jika ID yang digunakan di findViewById tidak ditemukan di R.id.

4. untuk menampilkan pesan error Resources.NotFoundException!

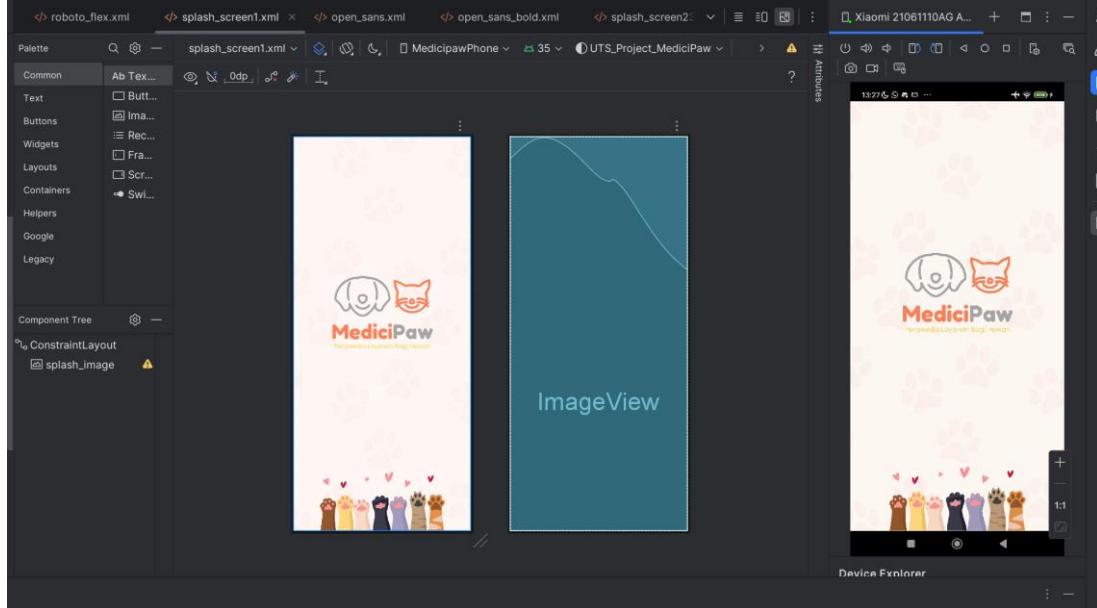
```
class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_main)

        val angka = resources.getString(id: 12)
    }
}
```

```
E FATAL EXCEPTION: main (Ask Gemini)
Process: com.example.apaaja, PID: 30372
java.lang.RuntimeException: Unable to start activity ComponentInfo{com.example.apaaja/com.example.apaaja.MainActivity}: android.content.res.Resources$NotFoundException: String resource ID #0x12
    at android.app.ActivityThread.performLaunchActivity(ActivityThread.java:3782)
    at android.app.ActivityThread.handleLaunchActivity(ActivityThread.java:3922)
    at android.app.servertransaction.LaunchActivityItem.execute(LaunchActivityItem.java:103)
    at android.app.servertransaction.TransactionExecutor.executeCallbacks(TransactionExecutor.java:139)
    at android.app.servertransaction.TransactionExecutor.execute(TransactionExecutor.java:96)
    at android.app.ActivityThread$H.handleMessage(ActivityThread.java:2443)
```

Studi kasus

A. Splash Screen



Splash screen akan menampilkan gambar beberapa detik lalu akan berpindah ke halaman berikutnya

Kode : SplashActivity1

```
package com.example.uts_project_medicipaw // Mendefinisikan package untuk file ini

// Mendefinisikan package untuk file ini, yang digunakan untuk mengelompokkan
kelas-kelas terkait.

import android.content.Intent
import android.os.Bundle
import android.view.animation.AnimationUtils
import android.widget.ImageView
import android.os.Handler
import androidx.appcompat.app.AppCompatActivity

class SplashActivity1 : AppCompatActivity() {
    private lateinit var splashImage: ImageView // Mendeklarasikan variabel
    splashImage dari tipe ImageView

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.splash_screen1) // Mengatur layout untuk activity ini

        splashImage = findViewById(R.id.splash_image) // Menginisialisasi splashImage
        dengan ImageView dari layout

        val fadeIn = AnimationUtils.loadAnimation(this, R.anim.fade_in) // Memuat
        animasi fade_in
```

```
splashImage.startAnimation(fadeIn) // Memulai animasi fadeIn pada
splashImage

    Handler().postDelayed({ // Membuat objek Handler untuk menjadwalkan
eksekusi kode
        startSecondSplashScreen() // Memanggil metode untuk memulai activity
berikutnya
        }, 3000) // Menjadwalkan eksekusi setelah 3000 milidetik (3 detik)
    }

    private fun startSecondSplashScreen() { // Mendeklarasikan fungsi untuk memulai
activity berikutnya
        val intent = Intent(this, SplashActivity234::class.java) // Membuat Intent untuk
berpindah ke SplashActivity234
        startActivity(intent) // Memulai activity baru
        finish() // Menutup SplashActivity1
    }
}
```

Kode : splash_screen1

```
<?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"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#ffd043">

    //Memanggil gambar yang dibuat menjadi image view untuk background
    <ImageView
        android:id="@+id/splash_image" //menjadikan id
        android:layout_width="0dp"
        android:layout_height="0dp"
        android:scaleType="centerCrop"
        android:src="@drawable/splash_screen1" //memanggil gambar yang sudah
dibuat di drawble

        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

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

Kode : Animasi :Fade in dan Fade out

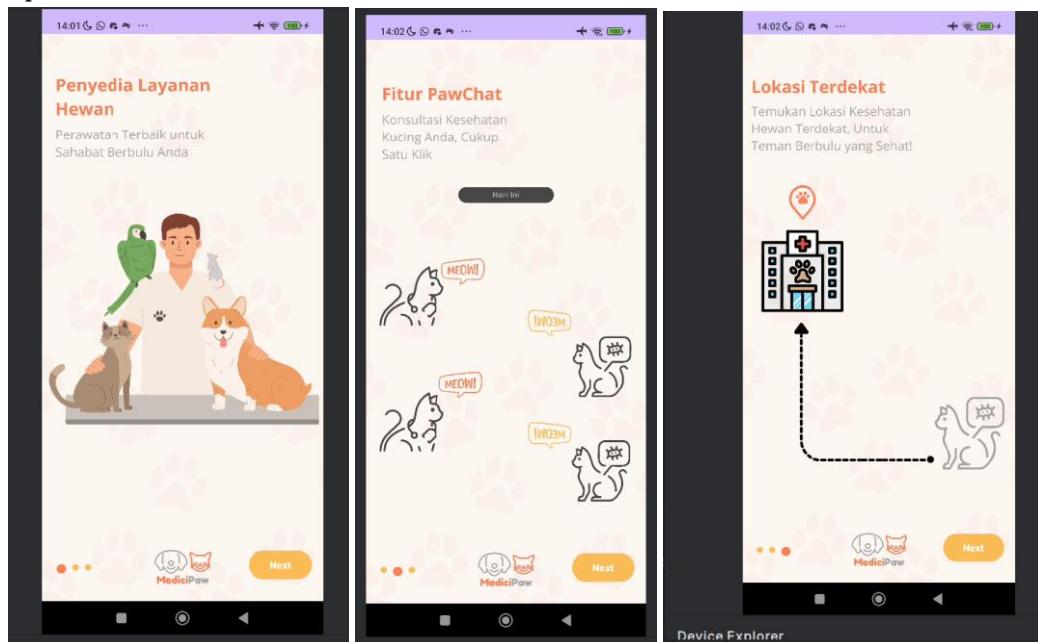
Fade in

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
    <alpha xmlns:android="http://schemas.android.com/apk/res/android"
        // Animasi alpha (transparansi)
        android:fromAlpha="0.0"
        android:toAlpha="1.0"
        android:duration="300"/>
</set>
```

Fade out

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
    <alpha xmlns:android="http://schemas.android.com/apk/res/android"
        //Animasi alpha (transparansi)
        android:fromAlpha="1.0"
        android:toAlpha="0.0"
        android:duration="300"/>
</set>
```

B. Splash Screen



Kode : SplashActivity234

```
package com.example.uts_project_medicipaw

import android.content.Intent
import android.os.Bundle
import android.view.View
```

```
import android.view.animation.Animation
import android.view.animation.AnimationUtils
import android.graphics.Color
import android.widget.Button
import android.widget.ImageView
import android.widget.LinearLayout
import androidx.appcompat.app.AppCompatActivity

class SplashActivity234 : AppCompatActivity() {

    private lateinit var splashImage: ImageView // Gambar splash
    private lateinit var nextButton: Button // Tombol untuk melanjutkan
    private lateinit var indicatorLayout: LinearLayout // Layout indikator

    private val images = intArrayOf(R.drawable.splash_screen2,
        R.drawable.splash_screen3, R.drawable.splash_screen4) // Gambar splash
    private var currentIndex = 0 // Indeks gambar saat ini

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.splash_screen234) // Set layout

        // Inisialisasi komponen UI
        splashImage = findViewById(R.id.splash_image)
        nextButton = findViewById(R.id.next_button)
        indicatorLayout = findViewById(R.id.indicator)

        createIndicators(images.size) // Buat indikator untuk gambar
        splashImage.setImageResource(images[currentIndex]) // Tampilkan gambar pertama
        updateIndicators(currentIndex) // Perbarui indikator

        nextButton.setOnClickListener {
            if (currentIndex < images.size - 1) {
                currentIndex++ // Naikkan indeks
                animateSplashScreen() // Animasi transisi gambar
                updateIndicators(currentIndex) // Perbarui indikator
            } else {
                startActivity(Intent(this, LoginActivity::class.java)) // Pindah ke LoginActivity
                finish() // Tutup SplashActivity
            }
        }
    }

    private fun animateSplashScreen() {
        val fadeOut = AnimationUtils.loadAnimation(this, R.anim.fade_out)
        val fadeIn = AnimationUtils.loadAnimation(this, R.anim.fade_in)
```

```
val backgroundView: View = findViewById(R.id.background_view)

splashImage.startAnimation(fadeOut) // Mulai animasi keluar

fadeOut.setAnimationListener(object : Animation.AnimationListener {
    override fun onAnimationStart(animation: Animation?) {
        backgroundView.setBackgroundColor(Color.parseColor("#fff7f4")) // Ubah
        warna latar
    }

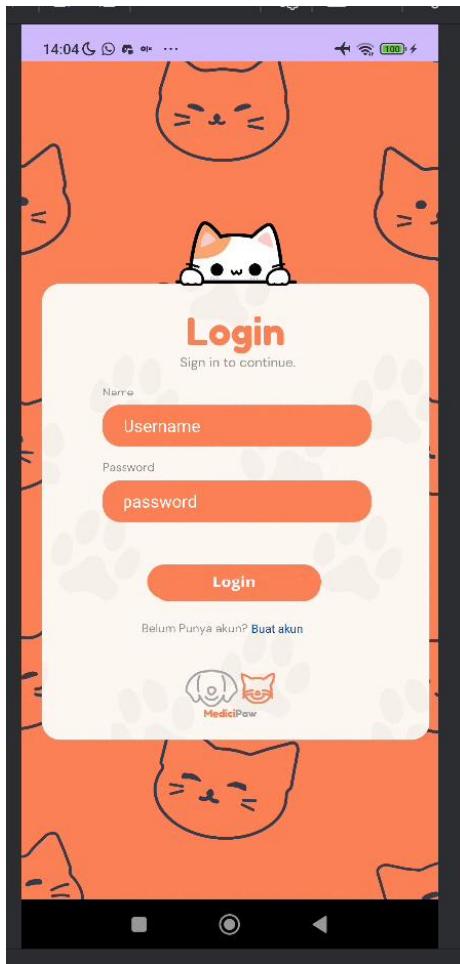
    override fun onAnimationEnd(animation: Animation?) {
        splashImage.setImageResource(images[currentIndex]) // Ubah gambar
        splashImage.startAnimation(fadeIn) // Mulai animasi masuk
    }

    override fun onAnimationRepeat(animation: Animation?) {}
})

private fun createIndicators(count: Int) {
    indicatorLayout.removeAllViews() // Hapus indikator lama
    for (i in 0 until count) {
        val indicator = ImageView(this)
        indicator.setImageResource(R.drawable.circle_inactive) // Indikator tidak aktif
        val params = LinearLayout.LayoutParams(24, 24) // Ukuran indikator
        params.setMargins(12, 0, 12, 0) // Margin antar indikator
        indicatorLayout.addView(indicator, params) // Tambah indikator ke layout
    }
}

private fun updateIndicators(position: Int) {
    for (i in 0 until indicatorLayout.childCount) {
        val indicator = indicatorLayout.getChildAt(i) as ImageView
        if (i == position) {
            indicator.layoutParams.width = 36 // Ukuran untuk indikator aktif
            indicator.layoutParams.height = 36
            indicator.setImageResource(R.drawable.circle_active) // Set indikator aktif
        } else {
            indicator.layoutParams.width = 24 // Ukuran untuk indikator tidak aktif
            indicator.layoutParams.height = 24
            indicator.setImageResource(R.drawable.circle_inactive) // Set indikator
            tidak aktif
        }
        indicator.requestLayout() // Meminta layout ulang
    }
}
```

C. Login Screen



Kode : LoginActivity

```
package com.example.uts_project_medicipaw

import android.content.Intent
import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import android.widget.TextView
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity
import com.google.firebase.auth.FirebaseAuth

class LoginActivity : AppCompatActivity() {
    private lateinit var auth: FirebaseAuth // Inisialisasi FirebaseAuth

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.login_screen) // Set layout untuk activity login
```

```
auth = FirebaseAuth.getInstance() // Mendapatkan instance FirebaseAuth

// Inisialisasi komponen UI
val emailEditText = findViewById<EditText>(R.id.et_text)
val passwordEditText = findViewById<EditText>(R.id.et_password)
val loginButton = findViewById<Button>(R.id.button)
val registerTextView = findViewById<TextView>(R.id.textView)

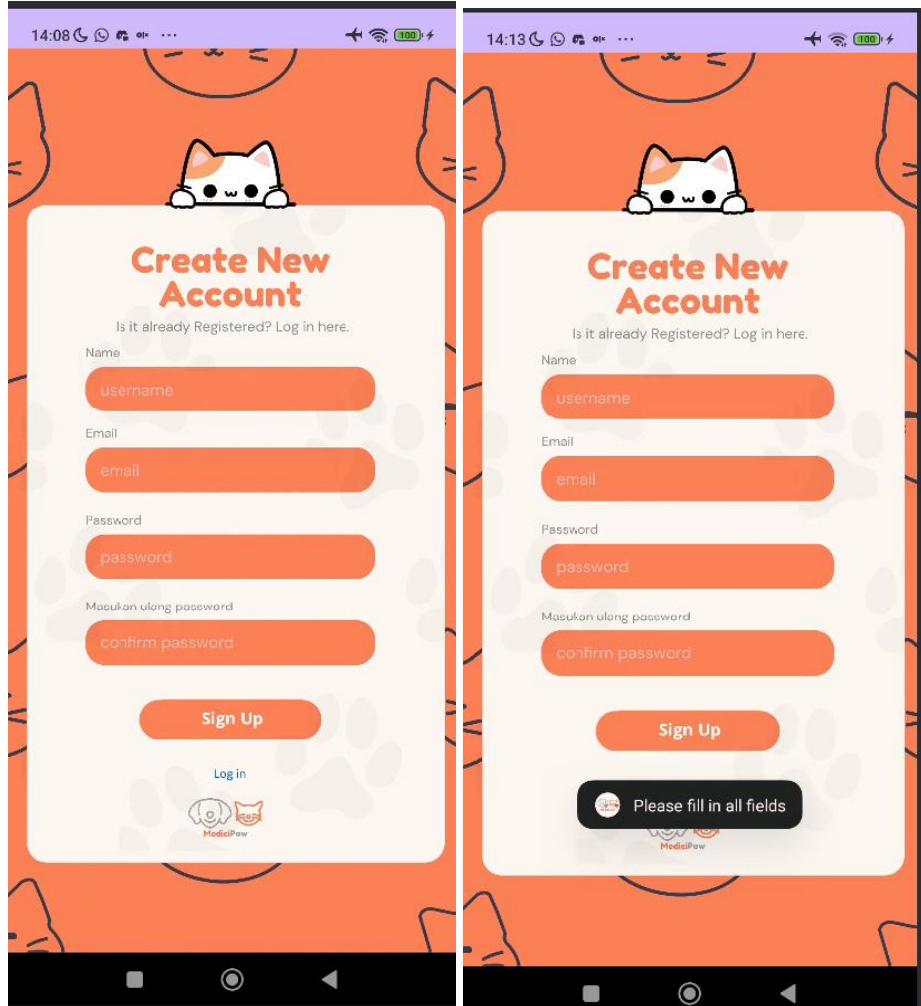
// Set listener untuk tombol login
loginButton.setOnClickListener {
    val email = emailEditText.text.toString().trim() // Mendapatkan email
    val password = passwordEditText.text.toString() // Mendapatkan password

    // Validasi input
    if (email.isEmpty() || password.isEmpty()) {
        Toast.makeText(this, "Please enter email and password",
        Toast.LENGTH_SHORT).show()
    } else {
        login(email, password) // Panggil fungsi login
    }
}

// Set listener untuk navigasi ke activity registrasi
registerTextView.setOnClickListener {
    startActivity(Intent(this, RegisterActivity::class.java)) // Pindah ke
RegisterActivity
}

private fun login(email: String, password: String) {
    // Proses login dengan Firebase
    auth.signInWithEmailAndPassword(email, password)
        .addOnCompleteListener(this) { task -
            if (task.isSuccessful) {
                // Login berhasil, navigasi ke MainActivity
                startActivity(Intent(this, MainActivity::class.java))
                finish() // Tutup activity login
            } else {
                // Menampilkan pesan kesalahan jika login gagal
                val errorMessage = task.exception?.message ?: "Login failed."
                Toast.makeText(baseContext, errorMessage,
                Toast.LENGTH_SHORT).show()
            }
        }
}
```

D. Register Screen



1. Pada Register Activity, minimal terdapat objek: TextView, EditText, Button, ImageView!
2. Tampilkan event Log, Toast dan Toast pada saat Button Register di klik.

Text view : digunakan pada tulisan login

EditText : pada username, email, password dan confirm password

Button : Pada Sign Up

Image View : digunakan pada background

Log dan toast : digunakan untuk notifikasi

Kode : RegisterActivity

```
package com.example.uts_project_medicipaw  
  
import android.content.Intent  
import android.os.Bundle  
import android.widget.Button
```

```
import android.widget.EditText
import android.widget.TextView
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity
import com.google.firebase.auth.FirebaseAuth
import com.google.firebaseio.database.FirebaseDatabase
import android.util.Log

class RegisterActivity : AppCompatActivity() {
    private lateinit var auth: FirebaseAuth // Inisialisasi FirebaseAuth

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.register_screen) // Set layout untuk
        activity registrasi

        auth = FirebaseAuth.getInstance() // Mendapatkan instance
        FirebaseAuth

        // Inisialisasi komponen UI
        val emailEditText = findViewById<EditText>(R.id.et_email)
        val usernameEditText = findViewById<EditText>(R.id.et_text2)
        val passwordEditText =
            findViewById<EditText>(R.id.et_password2)
        val confirmPasswordEditText =
            findViewById<EditText>(R.id.et_password3)
        val registerButton = findViewById<Button>(R.id.button2)
        val loginTextView = findViewById<TextView>(R.id.textView2)

        // Listener untuk tombol registrasi
        registerButton.setOnClickListener {
            val email = emailEditText.text.toString().trim() // Mendapatkan
            email
            val username = usernameEditText.text.toString().trim() // Mendapatkan username
            val password = passwordEditText.text.toString() // Mendapatkan password
            val confirmPassword = confirmPasswordEditText.text.toString() // Mendapatkan konfirmasi password

            // Validasi input
            if (email.isEmpty() || username.isEmpty() || password.isEmpty() ||
                confirmPassword.isEmpty()) {
                Toast.makeText(this, "Please fill in all fields",
                Toast.LENGTH_SHORT).show()
            } else if (password != confirmPassword) {
                Toast.makeText(this, "Passwords do not match",
                Toast.LENGTH_SHORT).show()
            } else {
```

```
        register(email, password, username) // Panggil fungsi
    registrasi
    }
}

// Listener untuk navigasi ke activity login
loginTextview.setOnClickListener {
    startActivity(Intent(this, LoginActivity::class.java)) // Pindah ke
LoginActivity
}
}

private fun register(email: String, password: String, username:
String) {
    Log.d("RegisterActivity", "Attempting to register user with email:
$email")

    // Proses pendaftaran dengan Firebase
    auth.createUserWithEmailAndPassword(email, password)
        .addOnCompleteListener(this) { task ->
            if (task.isSuccessful) {
                val userId = auth.currentUser?.uid // Mendapatkan ID
pengguna
                val database =
                    FirebaseDatabase.getInstance().getReference("Users/$userId")
                database.child("username").setValue(username) // Simpan
username ke database

                Toast.makeText(baseContext, "Registration successful.",
Toast.LENGTH_SHORT).show()
                finish() // Tutup activity registrasi
            } else {
                val errorMessage = task.exception?.message ?: "Unknown
error"
                Log.e("RegisterActivity", "Registration failed:
$errorMessage")
                Toast.makeText(baseContext, "Registration failed:
$errorMessage", Toast.LENGTH_SHORT).show()
            }
        }
    }
}
```

E. List Chatting



1. Pada List Chating terdapat data yang ditampilkan dalam listview
List tersebut menggunakan list view dan didalamnya terdapat data dari dokter

Kode : MainActivity

```
package com.example.uts_project_medicipaw
```

```
import android.annotation.SuppressLint
import android.os.Bundle
import androidx.appcompat.app.AppCompatActivity
import androidx.recyclerview.widget.LinearLayoutManager
import androidx.recyclerview.widget.RecyclerView

class MainActivity : AppCompatActivity() {

    private lateinit var recyclerView: RecyclerView
    private lateinit var chatAdapter: ChatAdapter
    private lateinit var chatList: List<ChatMessage>

    @Suppress("MissingInflatedId")
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.list_chatting)

        // Inisialisasi RecyclerView
        recyclerView = findViewById(R.id.recyclerView_chats)
        recyclerView.layoutManager = LinearLayoutManager(this)

        // Buat data dummy
        val chatList = listOf(
            ChatMessage(R.drawable.icon, "Dr. Dicki Firmansyah", "Jangan lupa berikan obatnya, ya! 😊", "5m ago", "Spesialis Kardiologi"),
            ChatMessage(R.drawable.icon, "Dr. Muhammad Akbar", "Jika ada gejala baru, konsultasikan lagi! 😊", "3m ago", "Dokter Hewan Umum"),
            ChatMessage(R.drawable.icon, "Dr. Muhamad Abdul", "PROMO SHAMPO KUCING GRATIS! 🎉", "10m ago", "Dokter Hewan Umum"),
        )
    }
}
```

ChatMessage(R.drawable.icon, "Dr. Muhhamad Fikri",
"Bagaimana kesehatan kucingmu? Semoga baik! 😊", "1h ago",
"Spesialis Kucing"),
ChatMessage(R.drawable.icon, "Dr. Salsabila Syazwani",
"Pastikan makanan dan minumannya cukup, ya! 🐱", "1h ago",
"Spesialis Nutrisi"),
ChatMessage(R.drawable.icon, "Dr. Lisa Ayu", "Cek kesehatan
hewanmu, jangan sampai mereka mengancam! 🐾", "2h ago", "Dokter
Hewan Umum"),
ChatMessage(R.drawable.icon, "Dr. Sheila Hamidah", "Jangan
biarkan hewanmu sakit, bawa ke dokter jika perlu! 🏥", "5h ago",
"Spesialis Bedah"),
ChatMessage(R.drawable.icon, "Dr. Anisa Wulandari", "Pastikan
hewanmu fit, agar tidak mengacaukan rencana! 🌟", "3h ago", "Dokter
Hewan Umum"),
ChatMessage(R.drawable.icon, "Dr. Arika Azhar", "Cek
kesehatan hewanmu, agar tidak ada sesi 'apa yang salah'? 🙄", "10h
ago", "Spesialis Penyakit Dalam"),
ChatMessage(R.drawable.icon, "Dr. Rafti Rahayu", "Pastikan
hewanmu bugar, biar tidak mengalihkan perhatian! 🤪", "2d ago",
"Spesialis Gigi"),
ChatMessage(R.drawable.icon, "Dr. Galih Andre", "Jangan lupa
jaga kesehatan hewanmu! 🐱", "2h ago", "Dokter Hewan Umum"),
ChatMessage(R.drawable.icon, "Dr. Its Taca", "Cek kesehatan
hewanmu, jangan sampai mereka menuntut! 🐱", "30m ago",
"Spesialis Dermatologi"),
ChatMessage(R.drawable.icon, "Dr. Koh Reven Handoko",
"Sayangi hewan peliharaanmu seperti pacarmu! 🐱", "15m ago",
"Spesialis Reproduksi")
)

```
// Inisialisasi adapter dan set ke RecyclerView
chatAdapter = ChatAdapter(chatList)
recyclerView.adapter = chatAdapter
}

}
```

Kode : ChatMessage

```
package com.example.uts_project_medicipaw
data class ChatMessage(
    val profileImage: Int,
    val userName: String,
    val lastMessage: String,
    val time: String,
    val category: String
)
// untuk mendeklarasikan chat message
```

Kode : ChatMessage

```
package com.example.uts_project_medicipaw

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

class ChatAdapter(private val chatList: List<ChatMessage>) :
    RecyclerView.Adapter<ChatAdapter.ViewHolder>() {

    // ViewHolder menyimpan referensi ke elemen UI
    class ViewHolder(itemView: View) :
        RecyclerView.ViewHolder(itemView) {
        val profileImage: ImageView =
            itemView.findViewById(R.id.imageView_profile) // Gambar profil
        val userName: TextView =
            itemView.findViewById(R.id.textView_name) // Nama pengguna
        val lastMessage: TextView =
            itemView.findViewById(R.id.textView_message) // Pesan terakhir
    }
}
```

```
    val time: TextView = itemView.findViewById(R.id.textView_time)
    // Waktu pesan
    val category: TextView =
    itemView.findViewById(R.id.textView_category) // Kategori pesan
}

// Menginflate layout untuk setiap item
override fun onCreateViewHolder(parent: ViewGroup, viewType:
Int): ViewHolder {
    val itemView =
    LayoutInflater.from(parent.context).inflate(R.layout.item_chat, parent,
false)
    return ViewHolder(itemView)
}

// Mengisi data ke dalam elemen UI
override fun onBindViewHolder(holder: ViewHolder, position: Int) {
    val chat = chatList[position]
    holder.profileImage.setImageResource(chat.profileImage) // Set
gambar profil
    holder.userName.text = chat.userName // Set nama pengguna
    holder.lastMessage.text = chat.lastMessage // Set pesan terakhir
    holder.time.text = chat.time // Set waktu pesan
    holder.category.text = chat.category // Set kategori pesan
}

// Mengembalikan jumlah item dalam daftar
override fun getItemCount() = chatList.size
}
```