

# **MODUL PRAKTIKUM PEMOGRAMAN BERGERAK**



**TIM Penyusun :** 

STMIK KAPUTAMA

KRISTINA ANNATASIA BR SITEPU HERU SAPUTRA

## PERTEMUAN 7 ACTIVITY DAN INTENT

#### 7.1 TUJUAN PEMBELAJARAN :

- a. Mahasiswa memahami konsep Intent dalam pemrograman Android.
- b. Mahasiswa mampu memindahkan data antar Activity menggunakan Intent.
- c. Mahasiswa mampu membedakan antara Explicit Intent dan Implicit Intent.
- d. Mahasiswa mampu membuat aplikasi multi-activity dan menggunakan Intent untuk memanggil aplikasi lain.

#### 7.2 ALAT DAN BAHAN :

- a. Laptop/PC
- b. Android Studio
- c. Sistem Operasi Windows

#### 7.3 MATERI

#### **A. Pengertian Intent**

Intent merupakan objek yang menyediakan waktu proses yang mengikat komponenkomponen terpisah. Intent biasanya digunakan untuk berpindah ke activity lain pada aplikasi android. Sederhananya, intent adalah cara untuk memberikan Action pada suatu komponen dalam satu aplikasi yang sama maupu aplikasi lain. Jika di-web kita mengenalnya dengan istilah Hyperlink.

#### **B. Fungsi Intent**

Berikut ini beberapa fungsi dari intent pada android, antara lain :

1) Untuk berpindah halaman dari satu Activity ke Activity lain.

Contoh-nya : Kita membuat aplikasi dengan 3 Activity. Activity utama berisi tombol informasi dan login. Ketika tombol informasi di klik akan tampil activity informasi begitupun dengan tombol login ketika di klik akan menampilkan activity login.

// MainActivity.java
Intent intent = new Intent(MainActivity.this, SecondActivity.class);
startActivity(intent);
finish(); // Menutup MainActivity setelah berpindah

2) Untuk transfer data dari satu Activity ke Activity lain.

Contoh-nya : Pada saat kita mengisi data pada Activity login maka nanti akan ditampilkan informasinya pada Activity infomasi user.

Di MainActivity, buat Intent dan tambahkan data menggunakan putExtra().

// MainActivity.java
Intent intent = new Intent(MainActivity.this, SecondActivity.class);
intent.putExtra("username", "JohnDoe");
startActivity(intent);

Di SecondActivity, ambil data menggunakan getIntent() dan tampilkan di TextView.

// SecondActivity.java
Intent intent = getIntent();
String username = intent.getStringExtra("username");
textView.setText("Hello, " + username);

3) Untuk memanggil activity pada internal android.

Contoh-nya : Melakukan Dial Number, Open Email ataupun lainnya.

// MainActivity.java

Intent intent = new Intent(Intent.ACTION\_PICK,

android.provider.MediaStore.Images.Media.EXTERNAL\_CONTENT\_URI);

startActivityForResult(intent, 1);

### C. Tipe Intent



Ada 2 (dua) tipe intent, yaitu :

1) Intent Explicit (Eksplisit)

Intent Explicit digunakan untuk memanggil Activity lain yang masih dalam satu project ataupun berbeda.

// MainActivity.java
Intent intent = new Intent(MainActivity.this, SecondActivity.class);
startActivity(intent);

#### Contoh :

🖀 N 🦉 🛙 12:15	🖆 N 🖳 🖞 12:17
Belajar Intent Android	Belajar Intent Android
TELEPHONE	Belajar tentang Aplikasi Intent Explicit dan Intent Implicit
INTERNET BROWSER	
ABOUT APPLICATION	
< 0 □	

Keterangan :

Saat di klik Button "About Application", maka akan berpindah kehalaman yang menampilkan mengenai keterangan aplikasi.

2) Intent Implicit (Implisit)

Intent Implicit adalah intent yang digunakan untuk memanggil fungsi Activity yang sudah ada di fungsi internal android. Seperti membuka browser, melakukan panggilan dan lainnya.

// MainActivity.java
Intent intent = new Intent(Intent.ACTION_VIEW,
Uri.parse("https://www.google.com"));
startActivity(intent);

Contoh :



Keterangan :

- Saat di klik button "Telephone" maka akan terhubung ke aplikasi kontak yang ada di perangkat handphone kita
- Saat di klik button "Internet Browser" maka akan terhubung ke browser yang ada di perangkat handphone kita

#### 7.4 PRAKTIKUM

1) Gunakan aplikasi Kamus yang telah dibuat sebelumnya pada pertemuan 6 (enam) yaitu KamusApp.



2) Modifikasi file XML untuk activity\_main.xml hingga menjadi seperti berikut :

xml version="1.0" encoding="utf-8"?
<linearlayout <="" td="" xmlns:android="http://schemas.android.com/apk/res/android"></linearlayout>
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
android:padding="16dp">
<autocompletetextview< td=""></autocompletetextview<>
android:id="@+id/autoCompleteTextView"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:hint="Type a word"
android:padding="8dp" />
android:loue.utidth "month"
android:layout_widtn="match_parent"
android:layout_neight="wrap_content"
android:iayout_marginiop="16dp"/>
<spinner< td=""></spinner<>
android:id="@+id/spinnerToLanguage"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_marginTop="16dp"/>
<button< td=""></button<>
android:id="@+id/buttonTranslate"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Translate"
android:layout_marginTop="16dp"/>

3) Modifikasi MainActivity.java untuk memasukkan pilihan bahasa ke dalam Intent.

import android.content.Intent; import android.os.Bundle; import android.view.View; import android.widget.ArrayAdapter; import android.widget.AutoCompleteTextView; import android.widget.Button; import android.widget.Spinner; import android.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
 private AutoCompleteTextView autoCompleteTextView;
 private Spinner spinnerFromLanguage;
 private Spinner spinnerToLanguage;
 private Button buttonTranslate;

private String[] words = {"Hello", "Goodbye", "Cat", "Dog", "Book"};
private String[] languages = {"English", "Indonesian", "Korean"};

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
```

autoCompleteTextView = findViewById(R.id.autoCompleteTextView); spinnerFromLanguage = findViewById(R.id.spinnerFromLanguage); spinnerToLanguage = findViewById(R.id.spinnerToLanguage); buttonTranslate = findViewById(R.id.buttonTranslate);

// Adapter untuk Spinner (bahasa asal dan tujuan)

```
ArrayAdapter<String> languageAdapter = new ArrayAdapter<>(this,
          android.R.layout.simple_spinner_item, languages);
languageAdapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown
_item);
     spinnerFromLanguage.setAdapter(languageAdapter);
     spinnerToLanguage.setAdapter(languageAdapter);
     buttonTranslate.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
          String word = autoCompleteTextView.getText().toString();
          String fromLanguage = spinnerFromLanguage.getSelectedItem().toString();
          String toLanguage = spinnerToLanguage.getSelectedItem().toString();
          // Mengirim data ke TranslationActivity
          Intent intent = new Intent(MainActivity.this, TranslationActivity.class);
          intent.putExtra("WORD", word);
          intent.putExtra("FROM_LANGUAGE", fromLanguage);
          intent.putExtra("TO_LANGUAGE", toLanguage);
          startActivity(intent);
        }
     });
  }
}
```

4) Buat file XML bernama activity\_translation.xml, dengan cara klik kanan pada layout →
 New → Layout Resource File. Kemudian akan muncul kotak dialog New Resource File (1).
 Pada bagia File name isikan nama file (2).

Android *	0 ± ÷ \$ = =	<ul> <li>activity_main.x</li> </ul>	ml 🛛 🤄 MainActivity.java 🖄 🍯 Definiti				
manifests AndroidMa java	nifest.xml	1 xml<br 2 <linea< td=""><td>version="1.8" encoding="utf-8" arLayout xmlns:android="http://</td><td>📥 New Resou</td><td>rce File</td><td></td><td></td></linea<>	version="1.8" encoding="utf-8" arLayout xmlns:android="http://	📥 New Resou	rce File		
<ul> <li>Cli com.exan</li> <li>Befinit</li> </ul>	New	>	👫 Kotlin Class/File	Eile name:	activity_definition.xml		
S MainA	Add C++ to Module		Layout Resource File Sample Data Directory	Root element:	androidx.constraintlayout.widget.ConstraintL	ayout	
> Di com.exan > Di com.exan > Di java (generati	Cut <u>Copy</u> Copy Path/Reference	Ctrl+X Ctrl+C		Source set:	main src/main/res		
res	Paste	Ctrl+V	<ul> <li>Directory</li> <li>Image Asset</li> </ul>	Directory name:	layout		
> Di layout	Find Usages	Alt+F7	🛎 Vector Asset	A <u>v</u> ailable qualifie	rs:		Chosen qualifiers:
Dil mipmap     Dil values     Dil xml	Find in Files Replace in Files Analyze	Ctrl+Shift+F Ctrl+Shift+R ≯	Kotlin Script Kotlin Worksheet	🐔 Country Co 🗂 Network Co	de ode		
res (generate)	Refactor	>	Activity	Locale	ction		
ar build.gradle ( ar build.gradle (	Bookmarks A Show in Resource Manager	) Ctrl+Shift+T	K Fragment	Smallest Sc Screen Wid	reen Width th	>>	Nothing to show
i proguard-ruk gradle.proper gradle-wrapp local.properti	<u>R</u> eformat Code Optimige Imports <u>D</u> elete	Ctrl+Alt+L Ctrl+Alt+O Delete	Service     Service     UiComponent     Automotive     XML	<ul> <li>Screen Heig</li> <li>Size</li> <li>Ratio</li> </ul>	jht		
er settings.grad	Open In	>	🛎 Wear >	UI Mode			
	Local History Repair IDE © Reload from Disk	>	ADL     ADL     Google	Ninht Mode     ?			OK Cance

5) Kemudian ketikkan koding berikut pada file activity\_translation.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="match_parent"
  android:orientation="vertical"
  android:padding="16dp">
  <TextView
     android:id="@+id/textViewWord"
     android:layout_width="wrap_content"
     android:layout_height="wrap_content"
     android:textSize="24sp"
     android:textStyle="bold" />
  <TextView
     android:id="@+id/textViewTranslation"
     android:layout_width="wrap_content"
     android:layout_height="wrap_content"
     android:textSize="18sp"
     android:layout_marginTop="8dp"/>
</LinearLayout>
```

6) Kemudian buat file java dengan nama TranslationActivity.java. Di file ini, kita akan mendapatkan kata dan bahasa yang dipilih dan menampilkan definisinya. Adapun cara menambahkan file tersebut yaitu :

a. Klik kanan pada folder java  $\rightarrow$  New  $\rightarrow$  Java Class

ĸ	amusApp_NPM2	!	pp $ angle$ src $ angle$ main	) java	) c	$om$ $\rangle$	exar	mple	🖿 kamu	sapp	_npm
E.	🛎 Android	Ŧ		0	\$	*	Ф	—	activit	ty_m	ain.xml 👋 🧿 MainActivity.java 👋
ource Manag	✓ Ing app ✓ Ing mani ✓ Ing mani Age A	ifests .ndro	idManifest.xml						1	)</th <th>xml version="1.0" encoding</th>	xml version="1.0" encoding
* Res	Y 📄 java								7	C	Java Class
			New						>	í	Kotlin Class/File
oject			Add C++ to Mo	dule						Ö	Android Resource File
ž	> 🖬 c	ж	Cut						Ctrl+X		Sample Data Directory
	> 📑 java	ē	<u>C</u> opy						Ctrl+C	É	File
	✓ Ingres > Ingres	Ö	Copy Path/Refe <u>P</u> aste	ence					Ctrl+V	Ē	Scratch File Ctrl+Alt+Shift+Inser Package
	✓ E■ k		Find <u>U</u> sages						Alt+F7		Image Asset Vector Asset
			Replace in Files					Ct	rl+Shift+R	1	Kotlin Script

b. Ketikkan nama file kemudian tekan Enter

	New Java Class
C	TranslationActivity.java
_	
9)	Class
I	Interface
E	Enum
9	Annotation

c. Ketikkan kode program berikut ke file tersebut

import android.content.Intent; import android.os.Bundle; import android.widget.TextView; import androidx.appcompat.app.AppCompatActivity; public class TranslationActivity extends AppCompatActivity { private TextView textViewWord; private TextView textViewTranslation; @Override protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity\_translation);

```
textViewWord = findViewById(R.id.textViewWord);
     textViewTranslation = findViewById(R.id.textViewTranslation);
     // Mendapatkan Intent yang dipanggil dan mengambil data
     Intent intent = getIntent();
     String word = intent.getStringExtra("WORD");
     String fromLanguage = intent.getStringExtra("FROM_LANGUAGE");
     String toLanguage = intent.getStringExtra("TO_LANGUAGE");
     String translation = getTranslation(word, fromLanguage, toLanguage); //
Mengambil terjemahan
     // Menampilkan kata dan terjemahan
     textViewWord.setText(word);
     textViewTranslation.setText(translation);
  }
  private String getTranslation(String word, String fromLanguage, String toLanguage) {
     // Menyimpan beberapa terjemahan untuk contoh
     if (fromLanguage.equals("English")) {
        switch (toLanguage) {
          case "Indonesian":
             return getIndonesianTranslation(word);
          case "Korean":
             return getKoreanTranslation(word);
          default:
             return "Translation not found.";
        }
     } else if (fromLanguage.equals("Indonesian")) {
        switch (toLanguage) {
          case "English":
             return getEnglishTranslation(word);
          case "Korean":
             return getKoreanFromIndonesian(word);
          default:
             return "Translation not found.";
```

```
}
  } else if (fromLanguage.equals("Korean")) {
     switch (toLanguage) {
        case "Indonesian":
           return getIndonesianFromKorean(word);
        case "English":
           return getEnglishFromKorean(word);
        default:
           return "Translation not found.";
     }
  }
  return "Translation not found.";
}
private String getIndonesianTranslation(String word) {
  switch (word.toLowerCase()) {
     case "hello":
        return "Halo";
     case "goodbye":
        return "Selamat tinggal";
     case "cat":
        return "Kucing";
     case "dog":
        return "Anjing";
     case "book":
        return "Buku";
     default:
        return "Terjemahan tidak ditemukan.";
  }
}
private String getKoreanTranslation(String word) {
  switch (word.toLowerCase()) {
     case "hello":
```

```
return "안녕하세요 (Annyeonghaseyo)";
     case "goodbye":
       return "안녕히 가세요 (Annyeonghi gaseyo)";
     case "cat":
       return "고양이 (Goyangi)";
     case "dog":
       return "개 (Gae)";
     case "book":
       return "책 (Chaek)";
     default:
       return "번역을 찾을 수 없습니다.";
  }
}
private String getEnglishTranslation(String word) {
  switch (word.toLowerCase()) {
     case "halo":
       return "Hello";
     case "selamat tinggal":
       return "Goodbye";
     case "kucing":
       return "Cat";
     case "anjing":
       return "Dog";
     case "buku":
       return "Book";
     default:
       return "Translation not found.";
  }
}
private String getKoreanFromIndonesian(String word) {
```

```
switch (word.toLowerCase()) {
     case "halo":
       return "안녕하세요 (Annyeonghaseyo)";
     case "selamat tinggal":
       return "안녕히 가세요 (Annyeonghi gaseyo)";
     case "kucing":
       return "고양이 (Goyangi)";
     case "anjing":
       return "개 (Gae)";
     case "buku":
       return "책 (Chaek)";
     default:
       return "번역을 찾을 수 없습니다.";
  }
}
private String getIndonesianFromKorean(String word) {
  switch (word.toLowerCase()) {
     case "안녕하세요":
       return "Halo";
    case "안녕히 가세요":
       return "Selamat tinggal";
     case "고양이":
       return "Kucing";
     case "개":
       return "Anjing";
     case "책":
       return "Buku";
     default:
       return "Terjemahan tidak ditemukan.";
```

```
}
  }
  private String getEnglishFromKorean(String word) {
     switch (word.toLowerCase()) {
        case "안녕하세요":
          return "Hello";
       case "안녕히 가세요":
          return "Goodbye";
       case "고양이":
          return "Cat";
        case "개":
          return "Dog";
        case "책":
          return "Book";
        default:
          return "Translation not found.";
     }
  }
}
```

7) Penjelasan Kode

- MainActivity: Mengumpulkan kata yang ingin diterjemahkan dan memilih bahasa asal serta tujuan dari dua Spinner. Setelah itu, saat tombol diterjemahkan ditekan, akan memanggil TranslationActivity dan mengirimkan data menggunakan Intent.
- TranslationActivity: Menerima data dari MainActivity dan menampilkan terjemahan berdasarkan kata yang dimasukkan. Fungsi getTranslation digunakan untuk memanggil metode yang sesuai untuk mendapatkan terjemahan berdasarkan bahasa asal dan tujuan.

8) Menjalankan Aplikasi

- Hubungkan perangkat Android atau buka Android Emulator.
- Klik tombol Run di Android Studio.
- Pilih perangkat yang ingin digunakan untuk menjalankan aplikasi.
- Aplikasi KamusApp sederhana akan ditampilkan.

10:34 🏟 🖬 🕲		₹4∎
KamusTigaBahas	а	
Type a word		
English		
English		•
TRANSLATE		
	•	

Tampilan saat aplikasi pertama kali dirunning



Masukkan kata yang ingin di terjemahkan, Pilih bahasa sumber dan bahasa tujuan menggunakan Spinner kemudian klik Terjemahkan



Setelah selesai, hasil terjemahan akan ditampilkan pada activity lain.

#### 7.5 TUGAS

- 1. Demokan project tersebut hingga muncul di emulator!
- Tambahkan minimal 10 kosakata baru ke dalam kamus yang sudah ada. Pastikan kosakata tersebut mencakup terjemahan untuk setiap kombinasi bahasa (Inggris-Indonesia, Indonesia-Korea, dll.).
- 3. Tugas dikumpul dalam format pdf dengan nama file : NPM\_Tugaske?
- 4. Didalam file tersebu dituliskan :
  - NPM
  - Nama Lengkap
  - Program Studi
  - Kelas