



Kampus
Merdeka
INDONESIA JAYA

MODUL PRAKTIKUM PEMOGRAMAN BERGERAK



PEMBELAJARAN
DARING KOLABORATIF
(PDK)



TIM Penyusun :

KRISTINA ANNTASIA BR SITEPU
HERU SAPUTRA

PERTEMUAN 6

DIALOG

6.1 TUJUAN PEMBELAJARAN :

- a. Mahasiswa dapat memahami konsep dan penggunaan dialog dalam aplikasi Android.
- b. Mahasiswa dapat membuat aplikasi sederhana yang menggunakan Dialog.

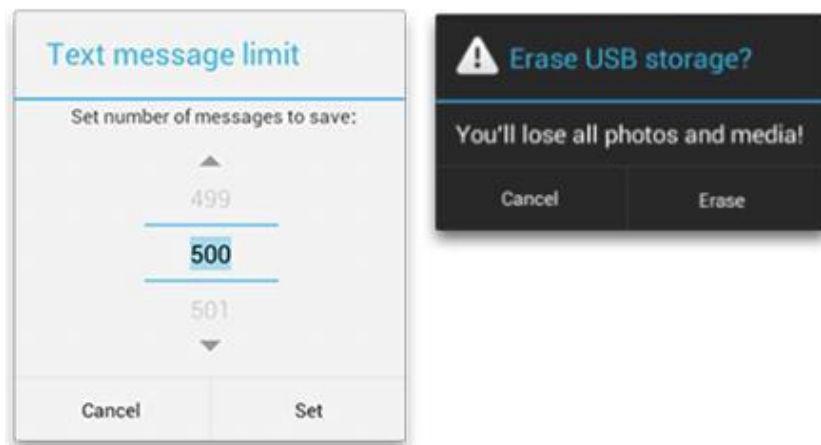
6.2 ALAT DAN BAHAN :

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

6.3 MATERI

1. Dialog

Dialog adalah sebuah jendela kecil yang muncul di atas *User Interface* (UI) aktif aplikasi. Dialog digunakan untuk menampilkan pesan, meminta input pengguna, atau memberikan pilihan. Dialog biasanya tidak memenuhi seluruh layar dan mengizinkan pengguna untuk berinteraksi dengan aplikasi di belakang dialog tersebut.



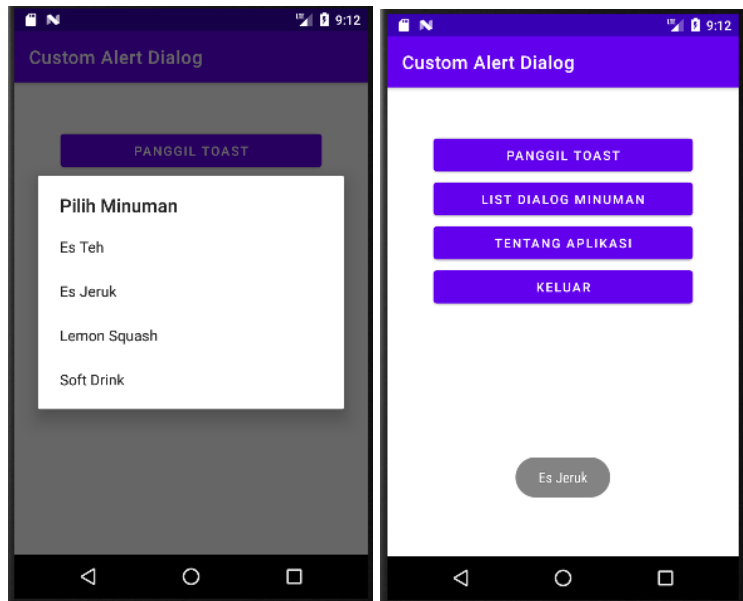
Jenis-jenis Dialog :

- a. AlertDialog : Digunakan untuk menampilkan pesan atau pilihan sederhana seperti "OK" atau "Cancel".
- b. Custom Dialog : Dialog yang dapat dikustomisasi tampilannya sesuai kebutuhan.

Contoh AlertDialog :



Contoh Custom Dialog :



2. ProgressDialog

ProgressDialog digunakan untuk menunjukkan bahwa proses sedang berlangsung dan membutuhkan waktu. Misalnya, proses unduhan, unggahan, atau pemrosesan data.

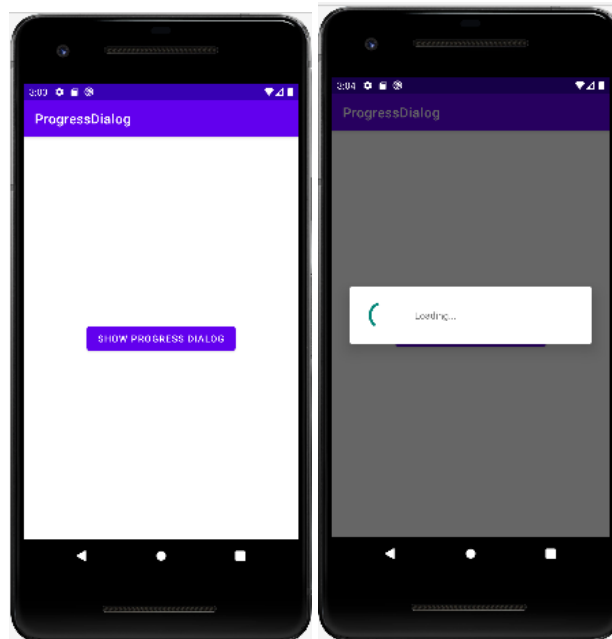
Jenis Progress Dialog:

- Determinate Progress Dialog: Menunjukkan progres yang jelas (misalnya, 50% selesai).
- Indeterminate Progress Dialog: Menunjukkan bahwa proses sedang berlangsung, tetapi tanpa detail progress.

Catatan: ProgressDialog sudah deprecated pada API level 26, dan penggunaan ProgressBar lebih disarankan.

Contoh Sederhana Penggunaan ProgressDialog :

```
ProgressDialog progressDialog = new ProgressDialog(this);  
progressDialog.setTitle("Loading");  
progressDialog.setMessage("Please wait...");  
progressDialog.setProgressStyle(ProgressDialog.STYLE_SPINNER);  
progressDialog.show();
```



Gambar contoh penggunaan ProgressDialog

3. Notifikasi

Notifikasi adalah pesan yang ditampilkan di status bar Android. Pengguna dapat mengetuk notifikasi untuk membuka aplikasi atau melakukan tindakan tertentu.

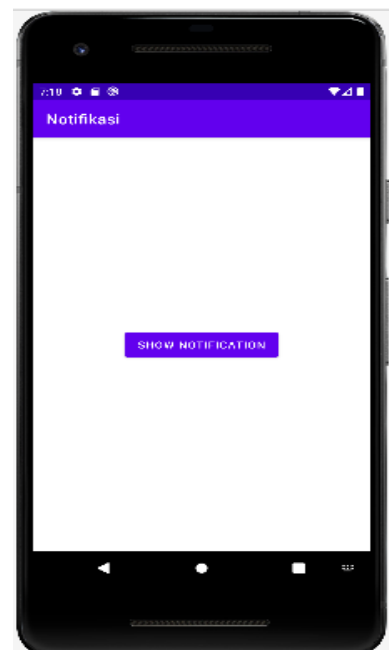
Langkah-langkah membuat notifikasi :

- Tentukan konten notifikasi (icon, judul, teks).
- Tentukan intent yang akan dieksekusi saat notifikasi ditekan.

Contoh Sederhana Penggunaan Notification :

```
NotificationCompat.Builder builder = new
NotificationCompat.Builder(this, "default")
    .setSmallIcon(R.drawable.ic_launcher_foreground)
    .setContentTitle("Notification Title")
    .setContentText("This is a notification message.")
    .setPriority(NotificationCompat.PRIORITY_DEFAULT);

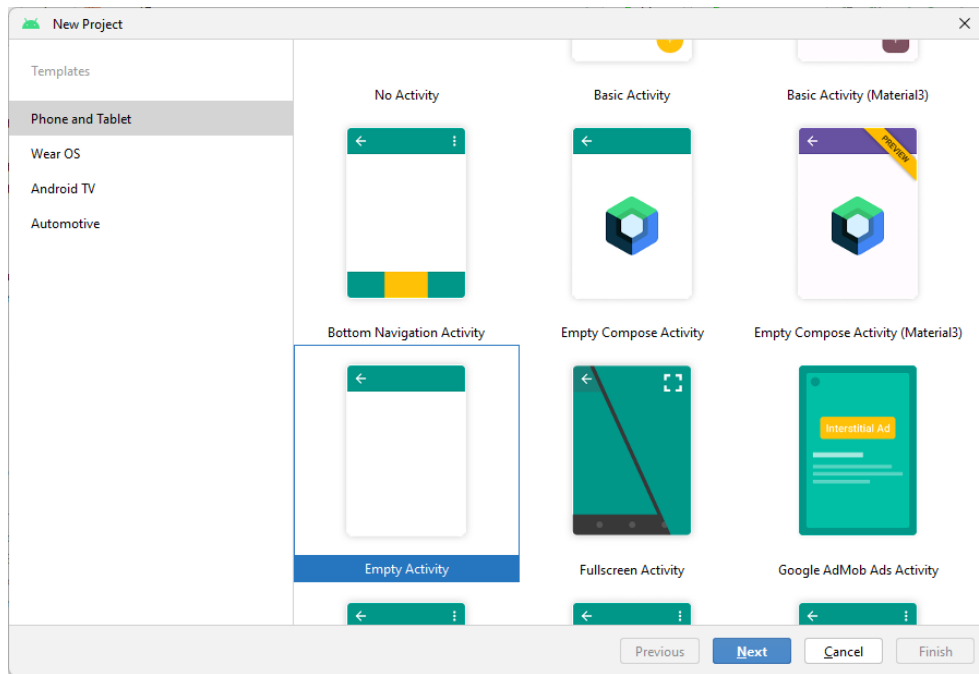
NotificationManagerCompat notificationManager =
NotificationManagerCompat.from(this);
notificationManager.notify(1, builder.build());
```



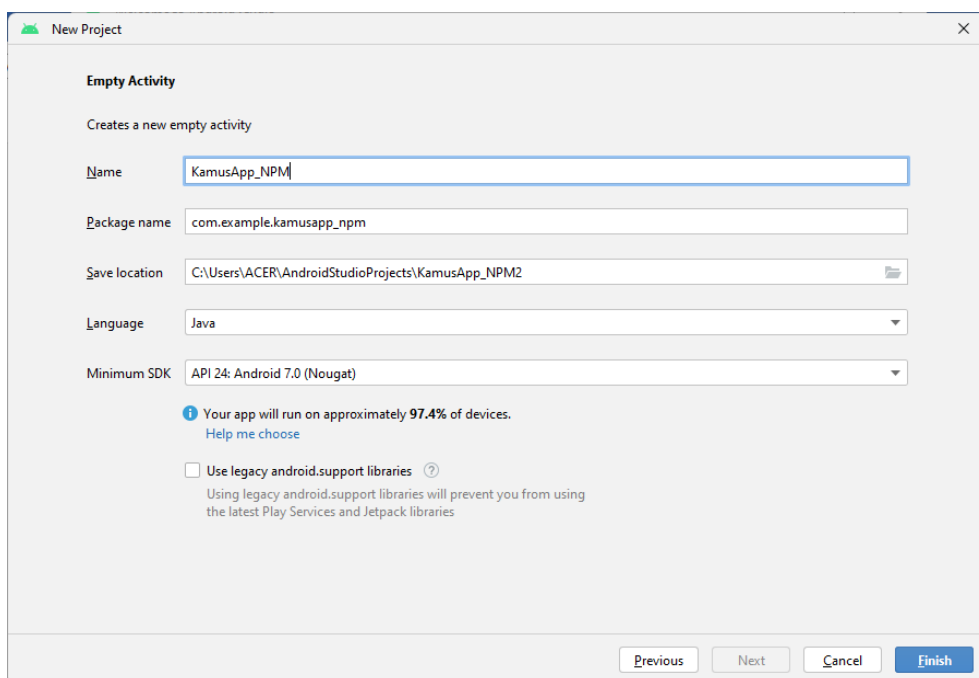
6.4 PRAKTIKUM

1) Buat Proyek Baru:

- Buka Android Studio, pilih "New Project".
- Pilih template "Empty Activity" dan klik "Next".



- Masukkan nama proyek : KamusApp_NPMMasingMasing
- Set language ke Java dan Minimum API Level sesuai kebutuhan
- Klik Finish



2) Ubah file activity_main.xml hingga seperti berikut :

```
<?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=".MainActivity">

    <!-- Spinner untuk memilih bahasa sumber -->
    <Spinner
        android:id="@+id/spinnerSourceLanguage"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        android:layout_margin="16dp"
        android:entries="@array/languages" />

    <!-- Spinner untuk memilih bahasa tujuan -->
    <Spinner
        android:id="@+id/spinnerTargetLanguage"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        app:layout_constraintTop_toBottomOf="@id/spinnerSourceLanguage"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        android:layout_margin="16dp" />

    <!-- EditText untuk memasukkan kata yang akan diterjemahkan -->
    <EditText
        android:id="@+id/editTextInput"
```

```
android:layout_width="0dp"
android:layout_height="wrap_content"
android:hint="Masukkan kata"
android:padding="10dp"
app:layout_constraintTop_toBottomOf="@id/spinnerTargetLanguage"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintEnd_toEndOf="parent"
android:layout_margin="16dp" />
```

```
<!-- Button untuk menerjemahkan -->
```

```
<Button
    android:id="@+id/btnTranslate"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:text="Terjemahkan"
    app:layout_constraintTop_toBottomOf="@id/editTextInput"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_margin="16dp" />
```

```
<!-- TextView untuk menampilkan hasil terjemahan -->
```

```
<TextView
    android:id="@+id/textViewOutput"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:textSize="18sp"
    android:padding="10dp"
    app:layout_constraintTop_toBottomOf="@id/btnTranslate"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_margin="16dp" />
```

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

Penjelasan:

- Spinner: Digunakan untuk memilih bahasa sumber dan bahasa tujuan.
- EditText: Tempat pengguna mengetikkan kata yang akan diterjemahkan.
- Button: Untuk memulai proses terjemahan.
- TextView: Untuk menampilkan hasil terjemahan.

3) Menambahkan Data Bahasa di Strings.xml

Buka file `res/values/strings.xml` dan tambahkan array bahasa yang akan digunakan dalam Spinner:

```
<resources>
  <string name="app_name">KamusTigaBahasa</string>

  <!-- Array bahasa untuk Spinner -->
  <string-array name="languages">
    <item>Inggris</item>
    <item>Indonesia</item>
    <item>Korea</item>
  </string-array>
</resources>
```

Penjelasan :

Array bahasa digunakan untuk mengisi item dalam Spinner agar pengguna bisa memilih bahasa sumber dan bahasa tujuan.

4) Membuat Logika Aplikasi di MainActivity.java

```
import android.app.ProgressDialog; // Import ProgressDialog
import android.os.Bundle;
import android.os.Handler;
import android.view.View;
import android.widget.AdapterView;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Spinner;
```



```

import android.widget.TextView;
import androidx.appcompat.app.AlertDialog; // Import AlertDialog
import androidx.appcompat.app.AppCompatActivity;
import java.util.HashMap;

public class MainActivity extends AppCompatActivity {

    private Spinner spinnerSourceLanguage, spinnerTargetLanguage;
    private EditText editTextInput;
    private Button btnTranslate;
    private TextView textViewOutput;
    private HashMap<String, String> dictionary;
    private ProgressDialog progressDialog; // Deklarasi ProgressDialog

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

        // Inisialisasi komponen UI
        spinnerSourceLanguage = findViewById(R.id.spinnerSourceLanguage);
        spinnerTargetLanguage = findViewById(R.id.spinnerTargetLanguage);
        editTextInput = findViewById(R.id.editTextInput);
        btnTranslate = findViewById(R.id.btnTranslate);
        textViewOutput = findViewById(R.id.textViewOutput);

        // Buat adapter untuk Spinner
        ArrayAdapter<CharSequence> adapter = ArrayAdapter.createFromResource(this,
R.array.languages, android.R.layout.simple_spinner_item);

        adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
        spinnerSourceLanguage.setAdapter(adapter);
        spinnerTargetLanguage.setAdapter(adapter);

```

```

// Inisialisasi kamus (dictionary)
initializeDictionary();

// Inisialisasi ProgressDialog
ProgressDialog = new ProgressDialog(this);
ProgressDialog.setMessage("Menerjemahkan...");

// Event klik tombol translate
btnTranslate.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String sourceLang = spinnerSourceLanguage.getSelectedItem().toString();
        String targetLang = spinnerTargetLanguage.getSelectedItem().toString();
        String inputWord = editTextInput.getText().toString().trim();

        // Tampilkan ProgressDialog
        ProgressDialog.show();

        // Simulasikan penundaan terjemahan (misalnya selama 2 detik)
        new Handler().postDelayed(new Runnable() {
            @Override
            public void run() {
                String translatedWord = translateWord(inputWord, sourceLang,
targetLang);

                // Tutup ProgressDialog setelah "proses" selesai
                ProgressDialog.dismiss();

                // Tampilkan hasil terjemahan di AlertDialog
                showTranslationDialog(inputWord, translatedWord);
            }
        }, 2000); // Simulasi delay 2 detik
    }
});
}

```

```

// Method untuk inialisasi kamus
private void initializeDictionary() {
    dictionary = new HashMap<>();

    // Inggris ke Indonesia
    dictionary.put("hello_inggris_indonesia", "halo");
    dictionary.put("cat_inggris_indonesia", "kucing");

    // Indonesia ke Inggris
    dictionary.put("halo_indonesia_inggris", "hello");
    dictionary.put("kucing_indonesia_inggris", "cat");

    // Indonesia ke Korea
    dictionary.put("halo_indonesia_korea", "안녕하세요 (annyeonghaseyo)");
    dictionary.put("kucing_indonesia_korea", "고양이 (goyang-i)");

    // Korea ke Indonesia
    dictionary.put("안녕하세요_korea_indonesia", "halo");
    dictionary.put("고양이_korea_indonesia", "kucing");

    // Inggris ke Korea
    dictionary.put("hello_inggris_korea", "안녕하세요 (annyeonghaseyo)");
    dictionary.put("cat_inggris_korea", "고양이 (goyang-i)");

    // Korea ke Inggris
    dictionary.put("안녕하세요_korea_inggris", "hello");
    dictionary.put("고양이_korea_inggris", "cat");
}

```

```

// Method untuk menerjemahkan kata
private String translateWord(String word, String sourceLang, String targetLang) {
    String key = word.toLowerCase() + "_" + sourceLang.toLowerCase() + "_" +
targetLang.toLowerCase();
    return dictionary.get(key);
}

// Method untuk menampilkan hasil di AlertDialog
private void showTranslationDialog(String originalWord, String translatedWord) {
    AlertDialog.Builder builder = new AlertDialog.Builder(this);
    builder.setTitle("Hasil Terjemahan");

    if (translatedWord != null) {
        builder.setMessage("Kata " + originalWord + " diterjemahkan menjadi:\n" +
translatedWord);
    } else {
        builder.setMessage("Kata " + originalWord + " tidak ditemukan dalam kamus.");
    }

    builder.setPositiveButton("OK", null);
    builder.show();
}
}

```

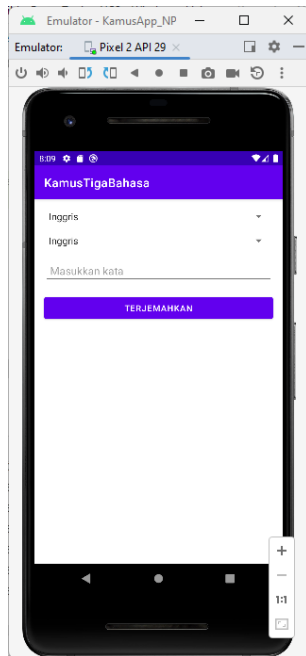
Penjelasan:

- ProgressDialog : Ditampilkan saat pengguna menekan tombol Terjemahkan dan disembunyikan setelah proses selesai (setelah 2 detik, sebagai simulasi proses).
- AlertDialog : Menampilkan hasil terjemahan setelah proses selesai. Jika kata ditemukan di kamus, hasil terjemahan akan ditampilkan. Jika tidak ditemukan, pesan akan menampilkan bahwa kata tidak ada di kamus.

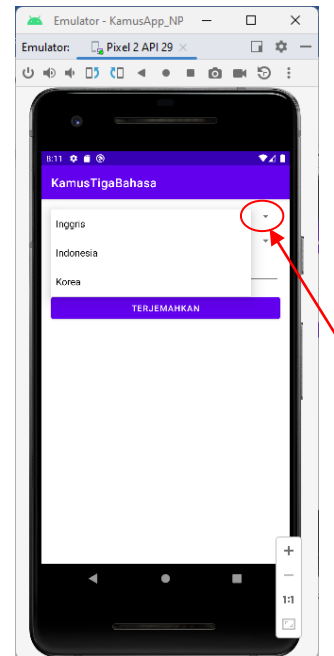
5) Menjalankan Aplikasi

- Hubungkan perangkat Android atau buka Android Emulator.

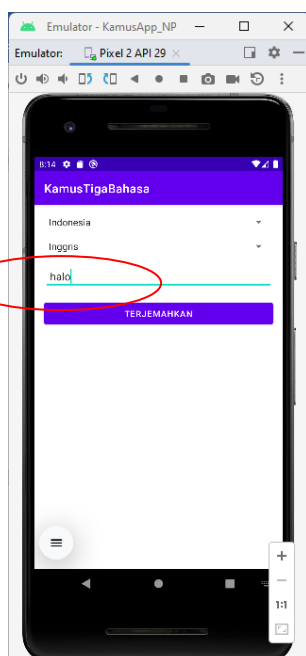
- Klik tombol Run di Android Studio.
- Pilih perangkat yang ingin digunakan untuk menjalankan aplikasi.
- Aplikasi *KamusApp* sederhana akan ditampilkan.



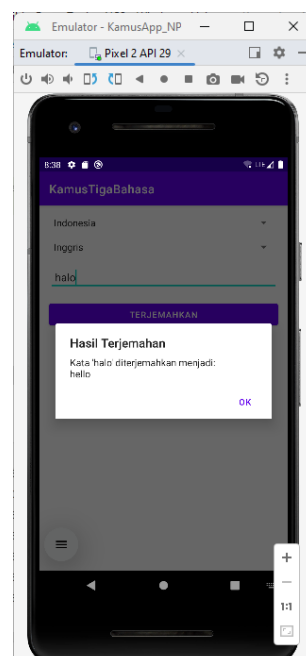
Tampilan saat aplikasi pertama kali di running



Pilih bahasa sumber dan bahasa tujuan menggunakan Spinner.



Masukkan kata yang ingin diterjemahkan, lalu klik Terjemahkan.



ProgressDialog akan muncul selama 2 detik untuk mensimulasikan proses terjemahan. Setelah selesai, hasil terjemahan akan ditampilkan dalam AlertDialog.

6.5 TUGAS

1. Demokan project tersebut hingga muncul di emulator!
2. 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 tersebut dituliskan :
 - NPM
 - Nama Lengkap
 - Program Studi
 - Kelas