

PERTEMUAN 11:

THREAD DAN ASYNTASK

A. TUJUAN PEMBELAJARAN

Adapun tujuan pembelajaran yang akan dicapai sebagai berikut:

- 11.1 Mengetahui Thread
- 11.2 Mengetahui AsyncTask
- 11.3 Membuat Project Dengan AsyncTask

B. URAIAN MATERI

Tujuan Pembelajaran 11.1:

Mengetahui Thread

User tidak suka dengan aplikasi yang terasa lambat. Aplikasi yang dibangun secara default memiliki satu proses. Jika sedang melakukan satu proses, maka tidak dapat melakukan proses lain sebelum proses sebelumnya selesai. Hal ini menyebabkan aplikasi terasa lambat. Agar aplikasi yang dibuat dapat terasa lebih cepat, dapat dilakukan dengan menambahkan threads lain.

Aplikasi standar akan diproses oleh thread UI (main thread), jika prosesnya lama, maka akan menyebabkan not responding. Untuk menghindari hal tersebut maka dapat digunakan thread.

Contoh threads sederhana:

```
public void onClick(View v)
{
    new Thread(new
    Runnable() {
        public
        void run() {
            // put your long and time consuming codes here...
        }
    }).start();
}
```

Sering terjadi ketika thread telah melakukan melakukan sesuatu, maka perlu melakukan pembaharuan beberapa teks atau gambar pada Android UI toolkit yang

tidak aman pada thread UI utama. Hal ini dapat menyebabkan masalah ketika mencoba untuk memperbarui Android UI toolkit seperti TextView atau EditText dari thread non-UI yang dibuat. Sebuah perbaikan cepat seperti di bawah ini dapat memecahkan masalah:

```
public void onClick(View v) {
    new Thread(new Runnable() {
        public void run() {
            // put your long and time consuming codes
            here...
            myTextView.post(new
            Runnable() {
                public void run() {
                    myTextView.setText("Loading complete!");
                }
            });
        }
    }).start();
}
```

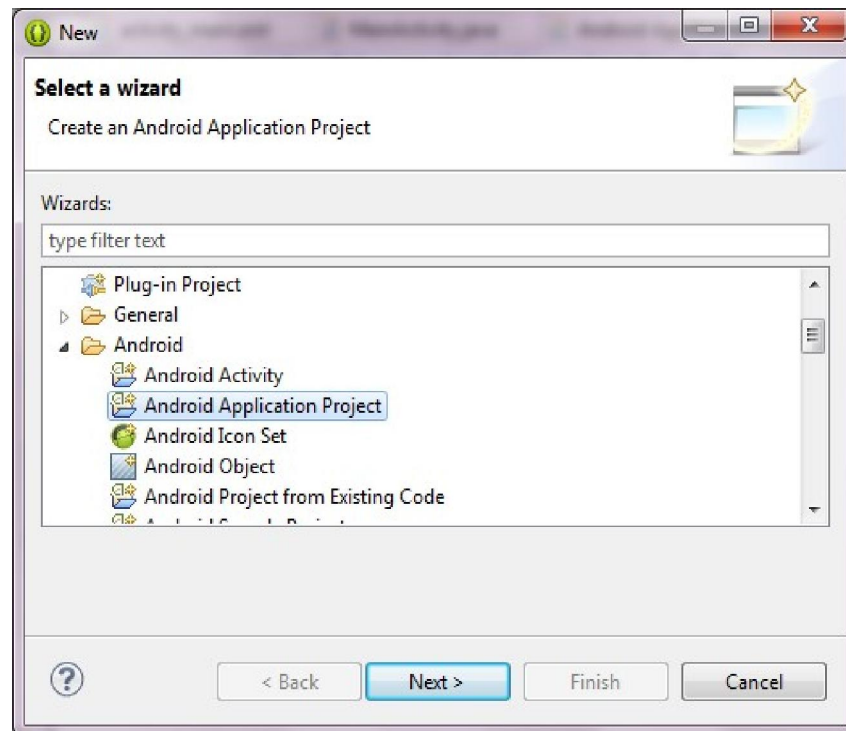
Tujuan Pembelajaran 11.2:

Mengenal AsyncTask

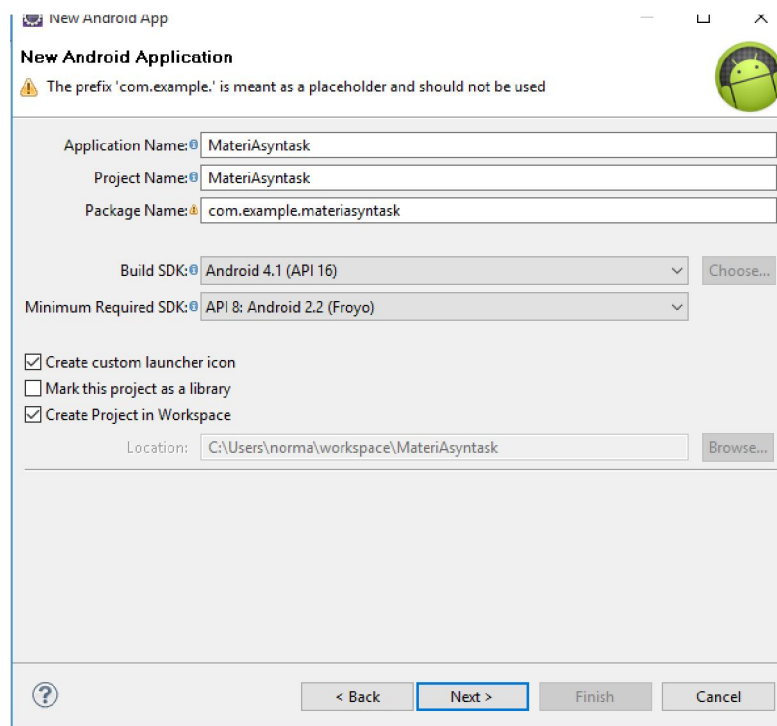
AsyncTask memungkinkan kita untuk melakukan pekerjaan asynchronous pada antarmuka pengguna Anda. Ia melakukan operasi memblokir di thread pekerja dan kemudian menerbitkan hasilnya pada thread UI, tanpa memerlukan kita untuk menangani thread dan/atau penanganan sendiri.

Tujuan Pembelajaran 11.3:

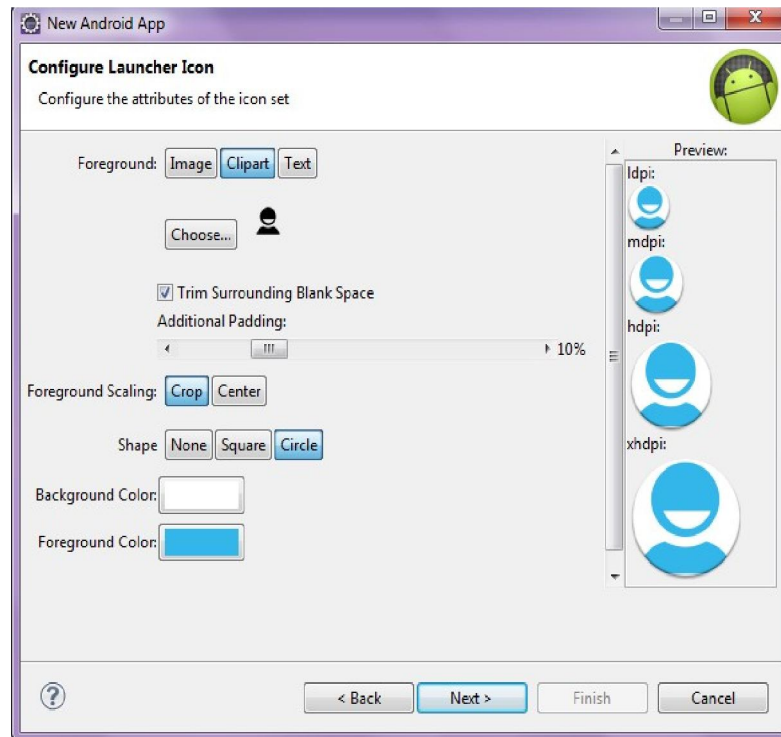
Membuat Project Dengan AsyncTask



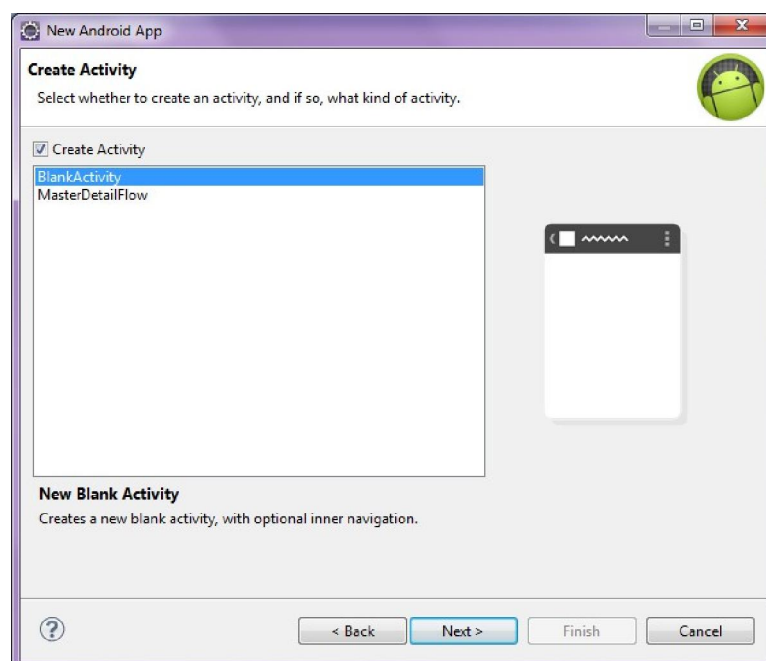
Gambar 11.1



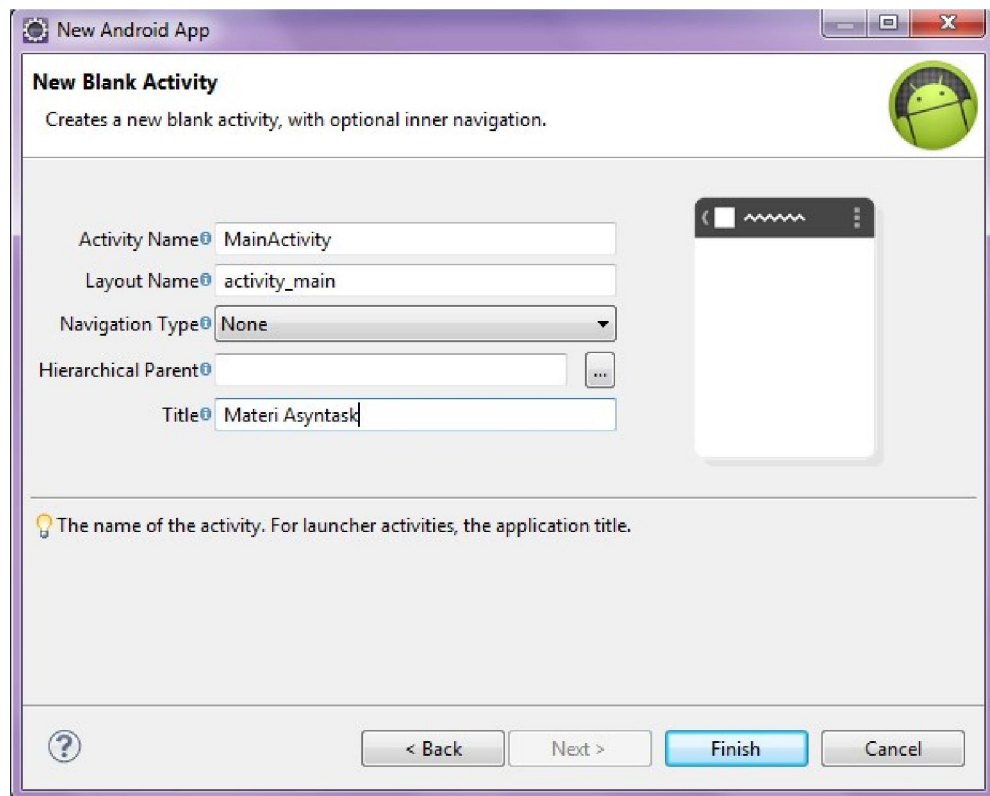
Gambar 11.2



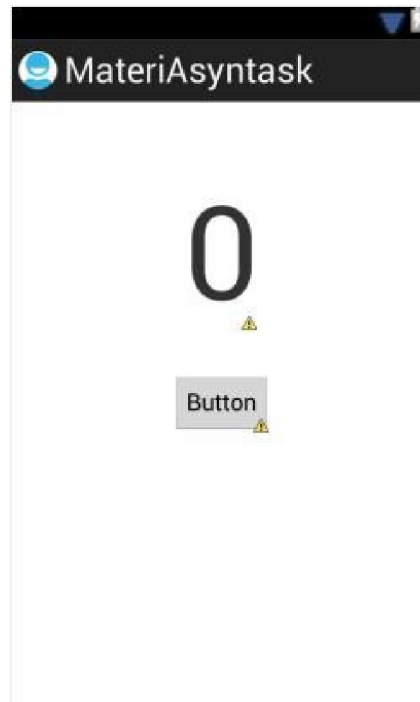
Gambar 11.3



Gambar 11.4



Gambar 11.5



Gambar 11.6



Gambar 11.7

Source code XML (activity_main.):

activity_main.xml

```
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"xmlns:t
ools="http://schemas.android.com/tools"android:layout_width="matc
h_parent"android:layout_height="match_parent">

<TextView
```

```

        android:id="@+id/countTextView"android:la
        yout_width="wrap_content"android:layout_h
        eight="wrap_content"android:layout_alignP
        arentTop="true"android:layout_centerHoriz
        ontal="true"android:layout_marginTop="58d
        p"android:text="0"
        android:textSize="100sp"/>

<Button
        android:id="@+id/startButton"android:layout_width="w
        rap_content"android:layout_height="wrap_content"andr
        oid:layout_centerHorizontal="true"android:layout_cen
        terVertical="true"android:text="Button"/>

</RelativeLayout>

```

Source Code MainActivity.java

```

package com.unpam.materiasyntask;

import android.os.AsyncTask;
import android.os.Bundle;
import android.app.Activity;
import android.view.Menu;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.TextView;
import android.widget.Toast;

publicclass MainActivity extends Activity implements
    OnClickListener{
    intnilai;
    TextView textView;

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

        textView = (TextView) findViewById(R.id.countTextView);
        findViewById(R.id.startButton).setOnClickListener(this);
    }

    @Override
    publicboolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.activity_main, menu);
        returntrue;
    }
}

```

```

public void onClick(View v) {
    // TODO Auto-generated method stub
    switch (v.getId()) {
        case R.id.startButton:
            MyAsyncTask asyncTask = new MyAsyncTask();
            asyncTask.execute(10);
        }
    }

    class MyAsyncTask extends AsyncTask<Integer, Integer, String> {
        @Override
        protected void onPreExecute() {
            Toast.makeText(MainActivity.this, "mulai",
                Toast.LENGTH_SHORT).show();
        }

        @Override
        protected void onProgressUpdate(Integer... values) {
            textView.setText(values[0] + "");
        }

        @Override
        protected String doInBackground(Integer... params) {
            nilai = params[0];
            while (nilai >= 0) {
                publishProgress(nilai);
                nilai--;
                try {
                    Thread.sleep(1000);
                } catch (InterruptedException e) {
                    e.printStackTrace();
                }
            }
            return "Selesai";
        }

        @Override
        protected void onPostExecute(String result) {
            Toast.makeText(MainActivity.this, result,
                Toast.LENGTH_SHORT).show();
        }
    }
}

```


C. SOAL LATIHAN/TUGAS

Buatlah aplikasi input data user (Fullname, username, email, password, gender) dengan JSON

D. DAFTAR PUSTAKA

Allen, Grant. 2012. Beginning Android 4. New York : Apress.

Safaat, H. Nazruddin. 2015. ANDROID Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android. Bandung: Informatika