

## PERTEMUAN 2:

### DESAIN USER INTERFACE

#### A. TUJUAN PEMBELAJARAN

Pada bab ini akan dijelaskan mengenai kegunaan user interface dan membuat project baru dalam aplikasi Android. Melalui praktikum, Anda harus mampu:

- 2.1 Menjelaskan kegunaan User Interface.
- 2.2 Membuat Project Baru Dengan Komponen User Interface
- 2.3 Mengatur Proses Aplikasi
- 2.4 Mengeksekusi Aplikasi

#### B. URAIAN MATERI

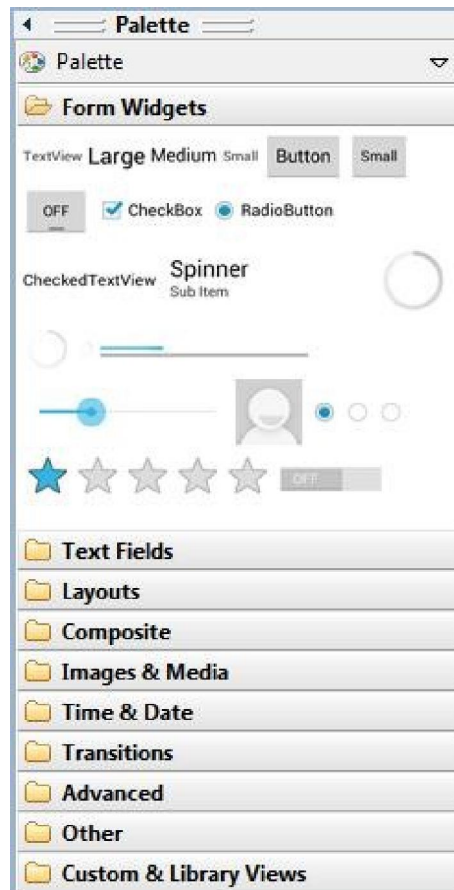
Tujuan Pembelajaran 2.1:
--------------------------

Menjelaskan Kegunaan User Interface
-------------------------------------

User interface (antarmuka pengguna) merupakan bentuk tampilan yang berfungsi untuk menghubungkan antar pengguna (user) dengan aplikasi atau sistem operasi, sehingga pengguna (user) dapat mengerti dan berkomunikasi. User interface dapat berupa teks maupun grafis. Untuk aplikasi yang menekankan pada kecepatan proses dan ditujukan untuk pengguna (user) dengan pengetahuan lebih baik, biasanya dibuat bentuk teks. Sedangkan untuk aplikasi yang ditujukan untuk pengguna akhir (end user) dan lebih menekankan user friendly, maka digunakan bentuk grafis.

Dalam pembuatan aplikasi android menggunakan eclipse, kita dapat membuat desain user interface melalui mode grafis maupun mode teks. Untuk mode grafis, dapat kita lakukan dengan drag and drop melalui jendela palette. Sedangkan desain melalui mode teks dapat kita lakukan menggunakan kode XML.

Tampilan jendela palette dapat dilihat pada gambar 2.1 berikut ini:

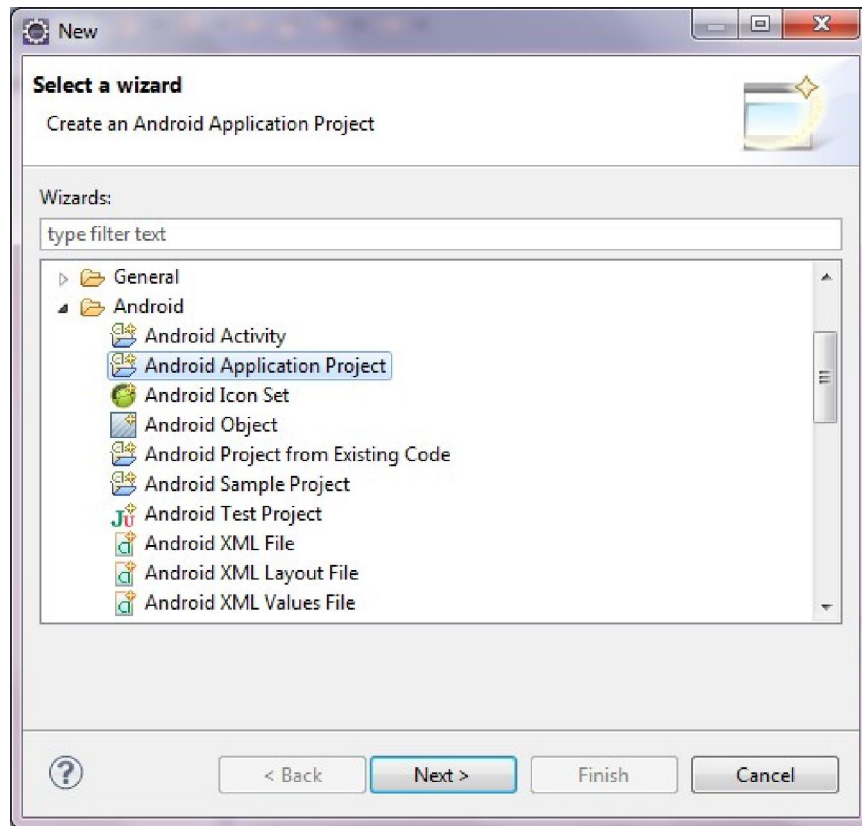


Gambar 2.1 Jendela palette

Tujuan Pembelajaran 2.2:

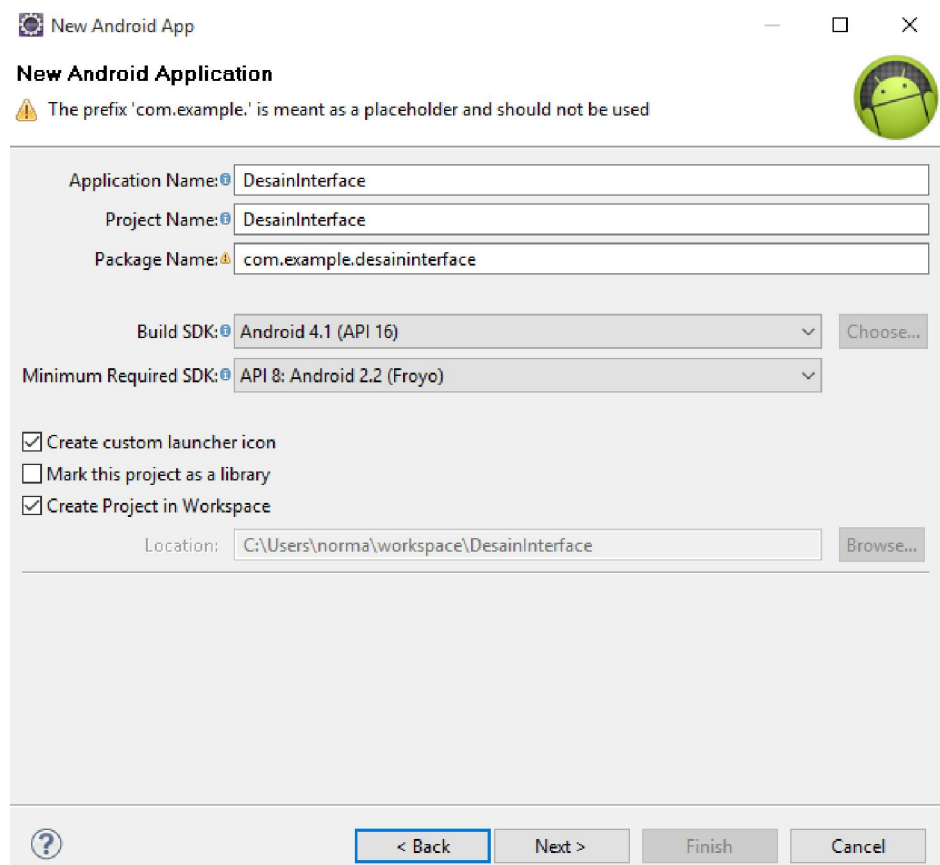
Membuat Project Baru Dengan Komponen User Interface

Untuk membuat project baru, klik menu File >> New >> Other... sehingga tampil kotak dialog seperti gambar 2.2 berikut ini:



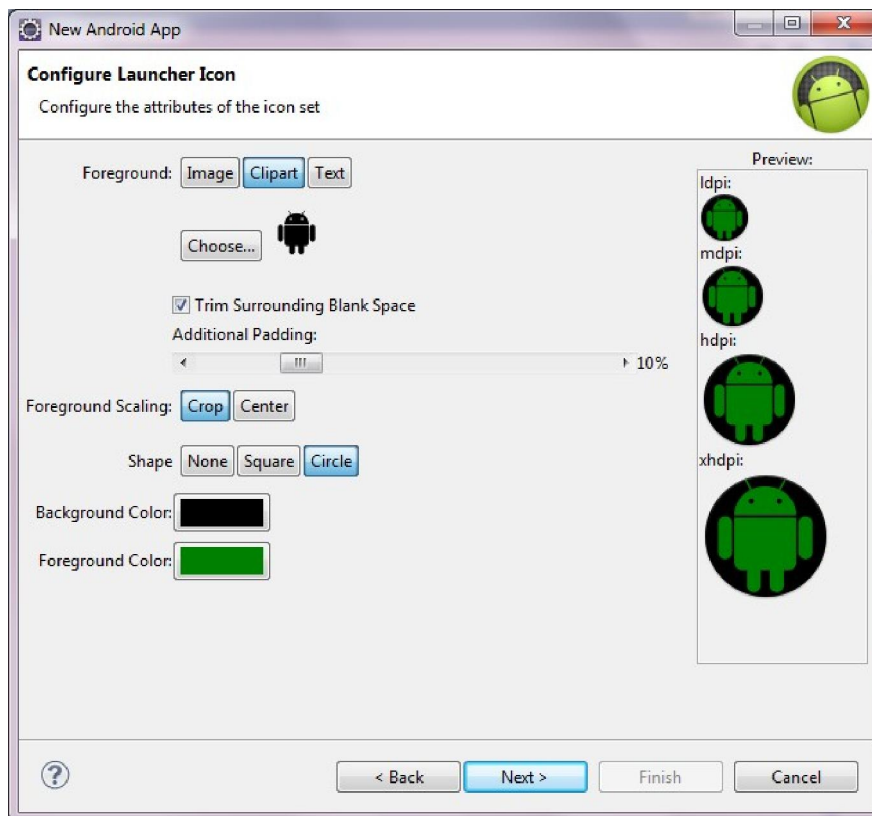
Gambar 2. 2 Kotak dialog membuat project baru

Pilih Android Application Project di dalam folder Android, kemudian klik tombol Next. Setelah tampil kotak dialog pengaturan seperti gambar 2.3, beri nama aplikasi “UserInterface”, dan pada nama package ubah “example” menjadi “unpam”. Tentukan versi android SDK untuk aplikasi yang akan dibuat pada menu dropdown Build SDK, dan minimum required SDK.

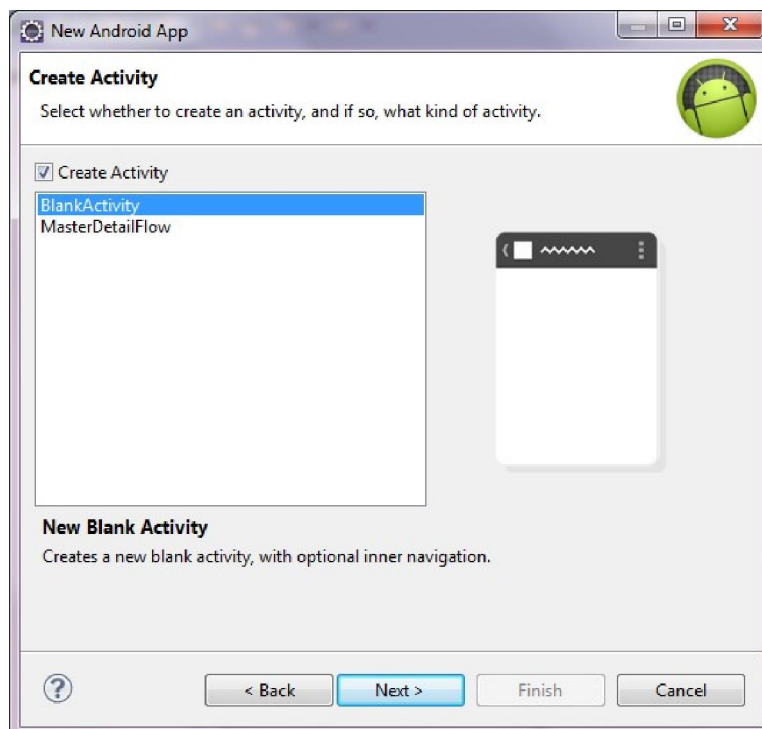


Gambar 2. 3 Kotak dialog pengaturan project

Setelah pengaturan nama aplikasi, nama project, nama package, dan SDK dari aplikasi, klik tombol next untuk melanjutkan sehingga tampil kotak dialog seperti gambar 2.4. Atur icon dari aplikasi yang akan dibuat, bisa menggunakan image, clipart, maupun text. Atur bentuk dan warna dari icon jika diperlukan, jika sudah selesai lanjutkan dengan mengklik tombol Next.

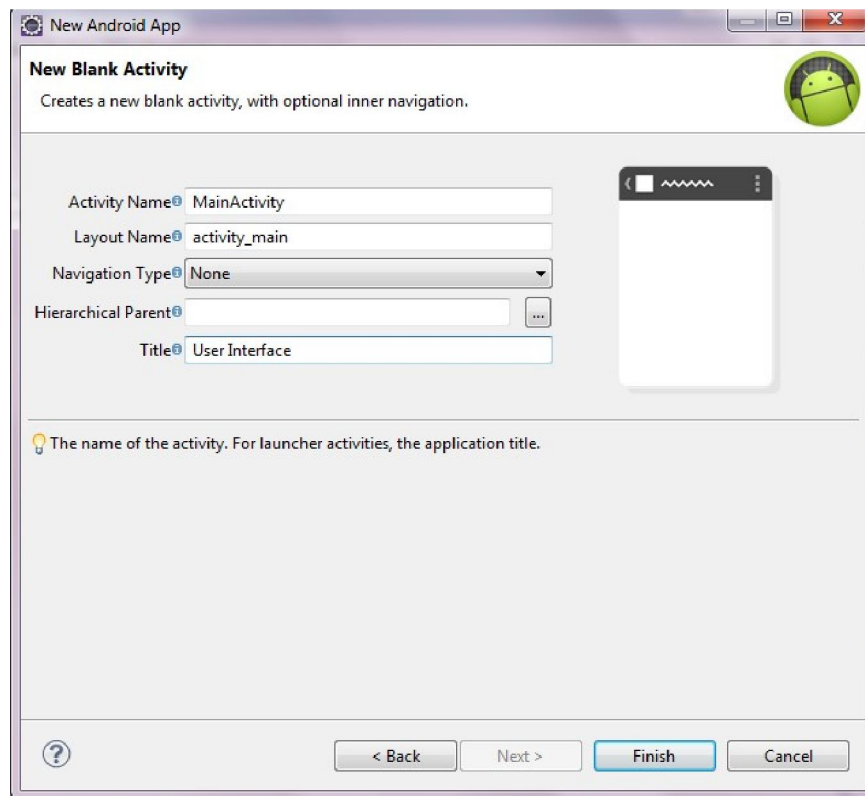


Gambar 2. 4 Pengaturan icon aplikasi



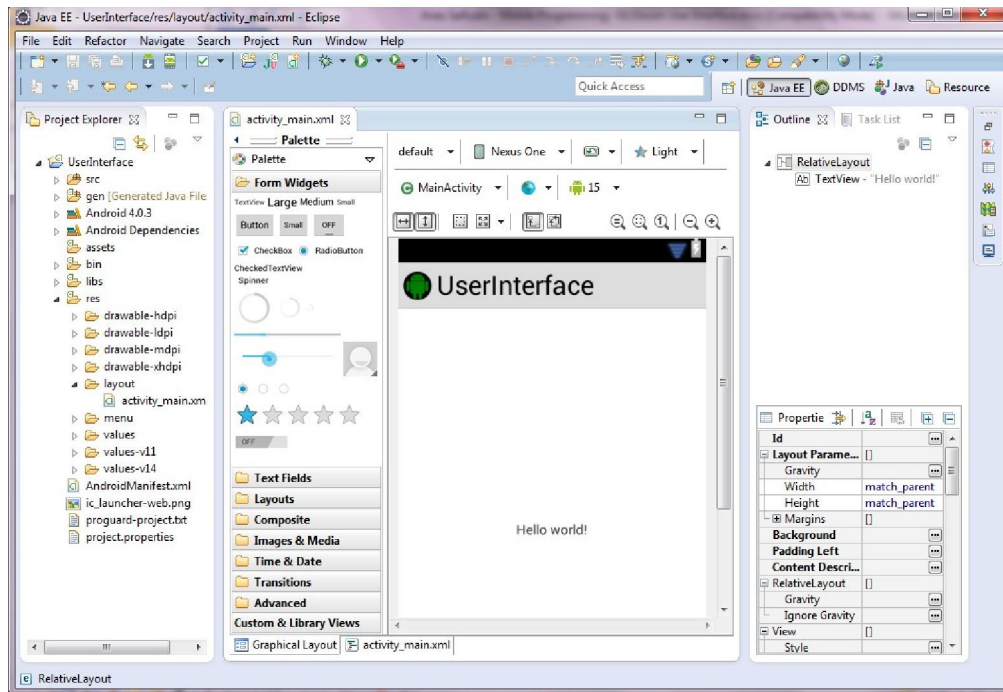
Gambar 2. 5 Memilih jenis Activity

Pilih Blank Activity untuk jenis Activity dari aplikasi yang akan dibuat. Lanjutkan dengan mengklik tombol Next. Ubah judul (Title) aplikasi menjadi “User Interface”.



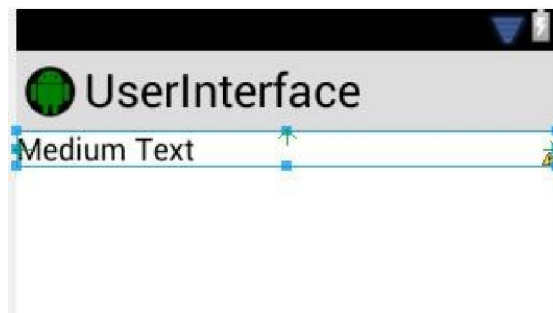
Gambar 2. 6 Mengatur nama dan judul activity

Klik tombol Finish untuk mengakhiri pengaturan pembuatan aplikasi. Tunggu beberapa saat sampai tampil desain aplikasi seperti gambar 2.7. Jika jendela yang tampil tidak sama, dapat diatur dari menu Window >> Show View dan pilih jendela yang ingin ditampilkan. Jika ingin mengubah tampilan ke bentuk standar, dapat dilakukan melalui menu Window >> Reset Perspective.



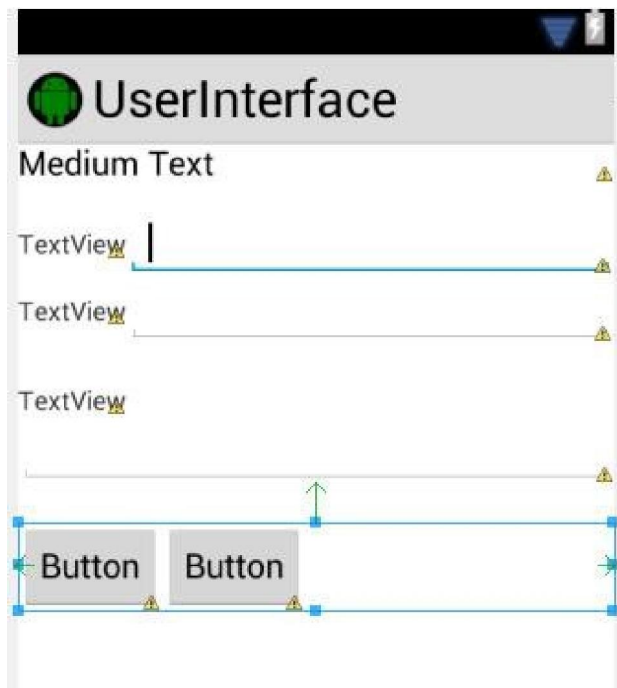
Gambar 2. 7 Tampilan IDE aplikasi android

HapusTextView (teks “Hello Word!”), tambahkanLinearLayout (Horizontal) di sudutkiriatas, kemudian di dalamnya tambahkanMedium, atursehinggatampilsepertiberikutini:

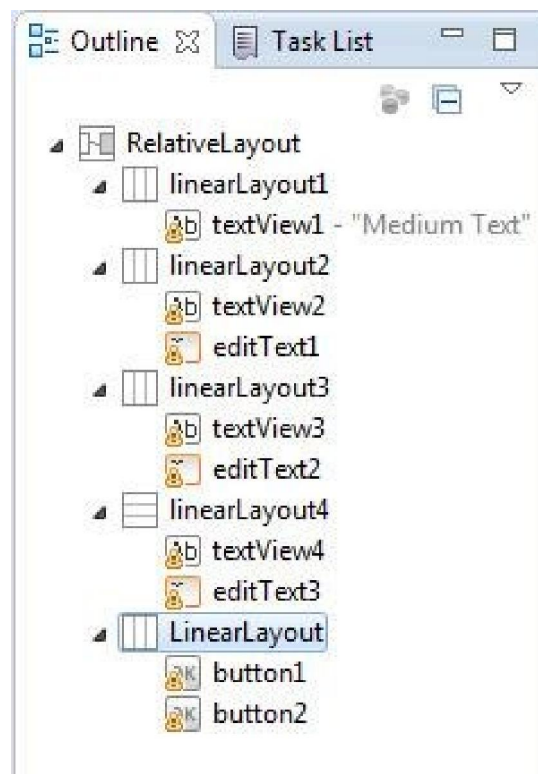


Gambar 2. 8 Menambahkankomponen layout dantextview

LanjutkandenganmenambahkankomponenTextView, EditText, danButton, janganlupamenggunakanLinearLayout (Horizontal), kecualiuntuknilaiakhirmenggunakanLinearLayout (Vertical). Aturdesaintampilanmenjadisepertigambar 2.9 dandesainoutlinemenjadisepertigambar 2.10.

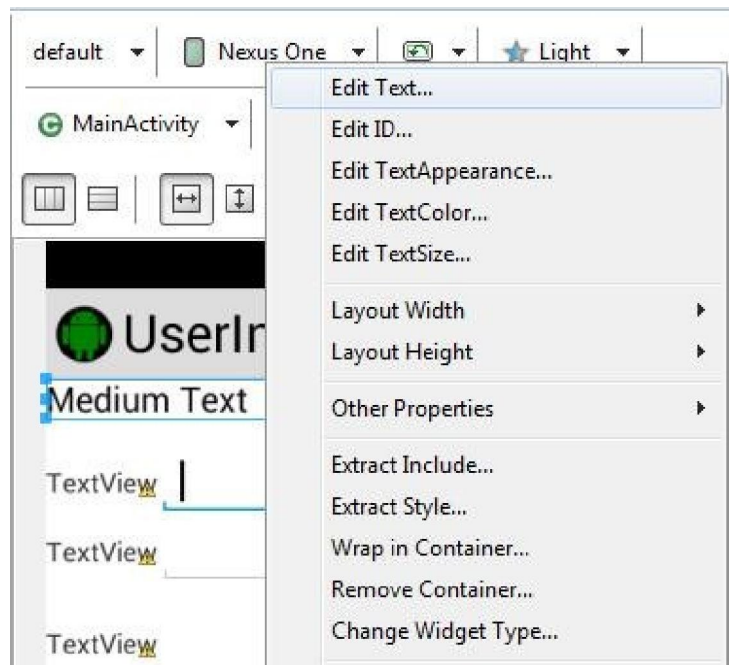


Gambar 2. 9 DesainTampilan

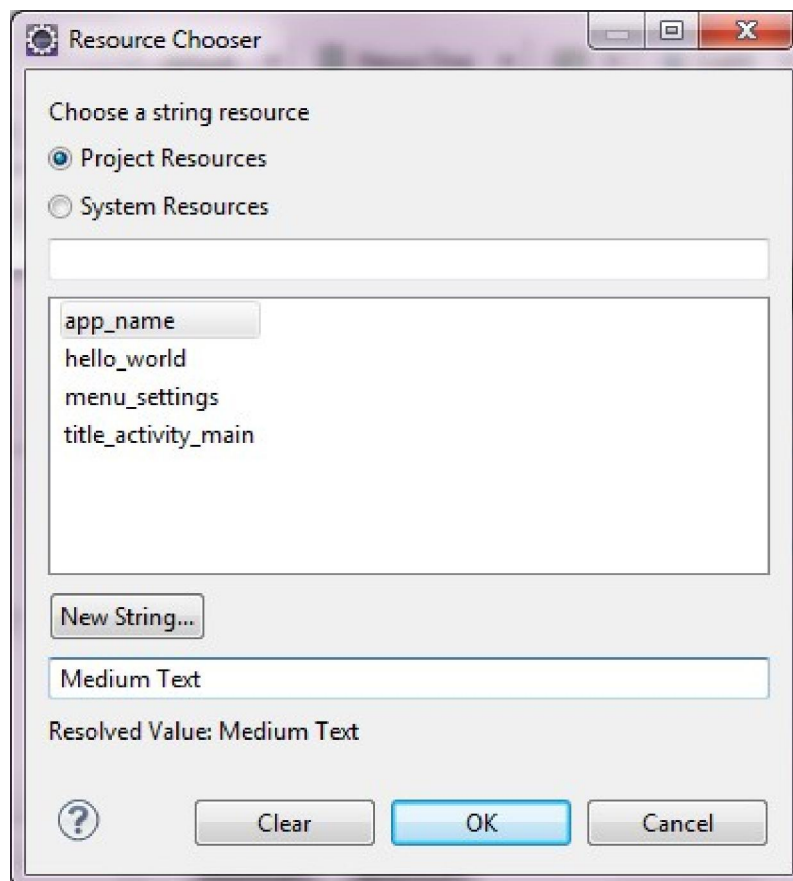


Gambar 2. 10 Desainoutline

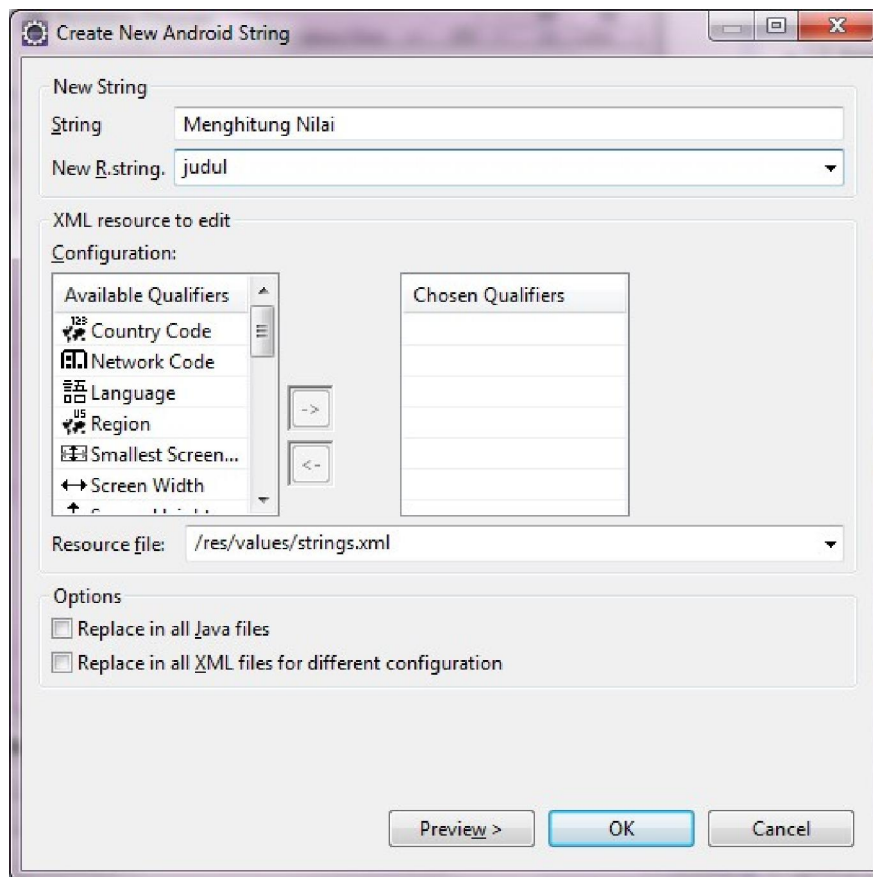




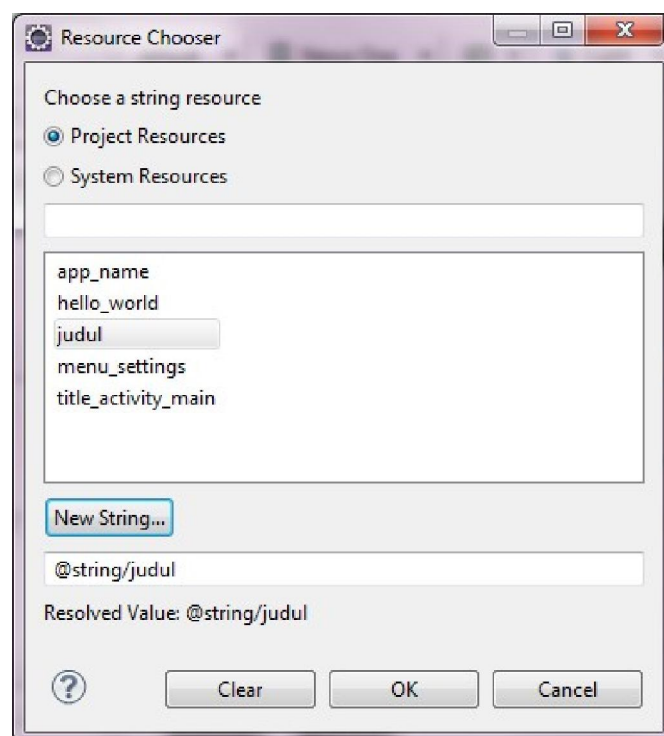
Gambar 2. 11 KlikkananpadaMedium Text



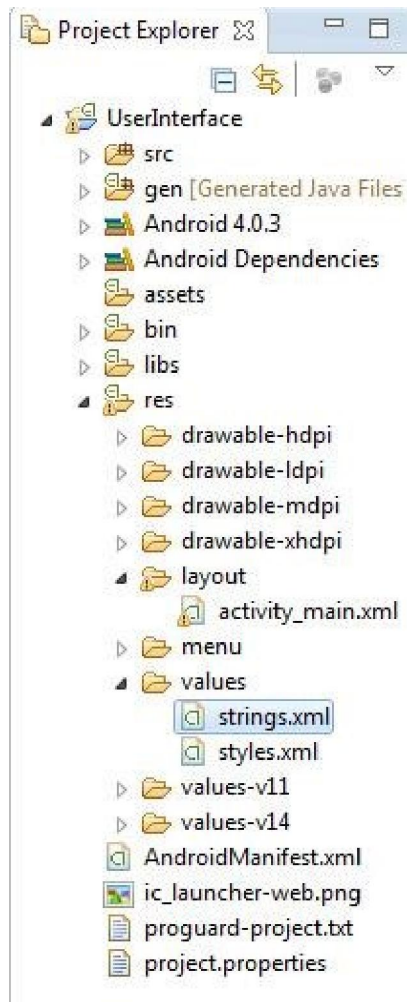
Gambar 2. 12 Jendelapengaturanteks



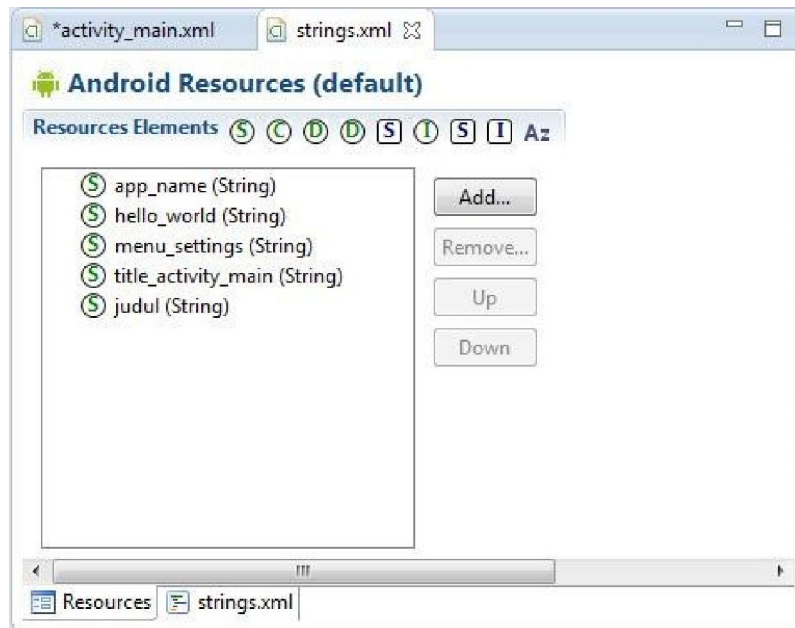
Gambar 2. 13 Membuatvariabel reference



Gambar 2. 14 Mengisinilaiteksdenganvariabel reference

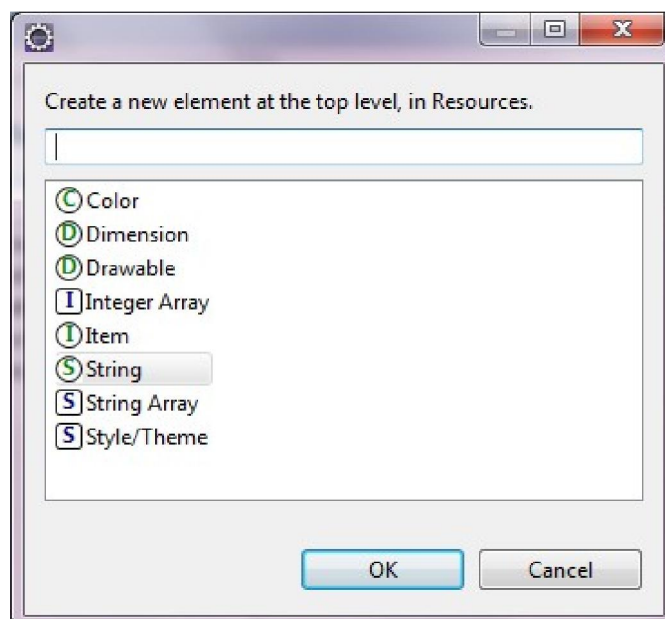


Gambar 2. 15 Lokasistrings.xml di dalamproject



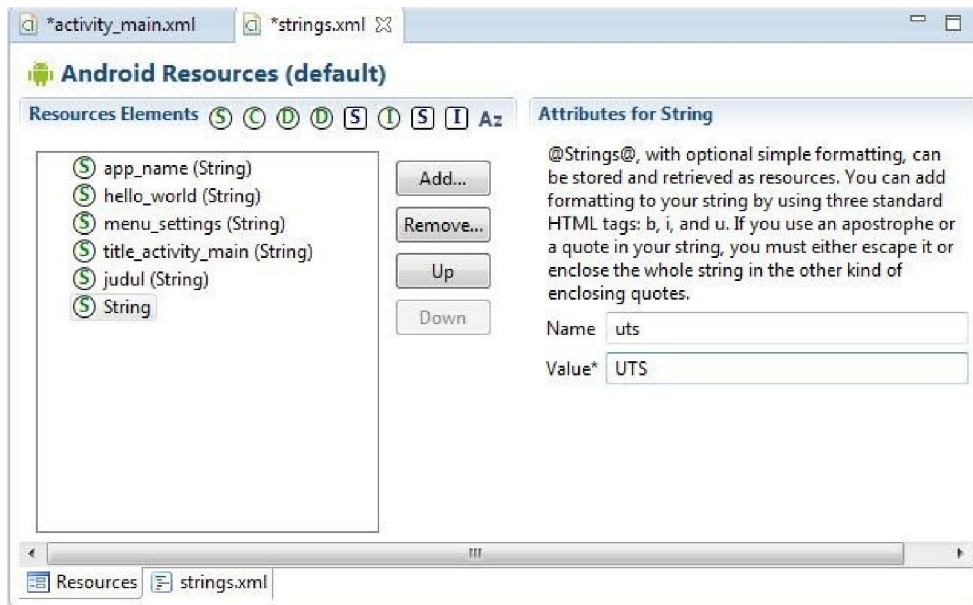
Gambar 2. 16 Halamanresources strings.xml

Klik tombol Add untuk menambahkan nilai baru, sehingga tampil kotak dialog seperti gambar 2.17 berikut ini:



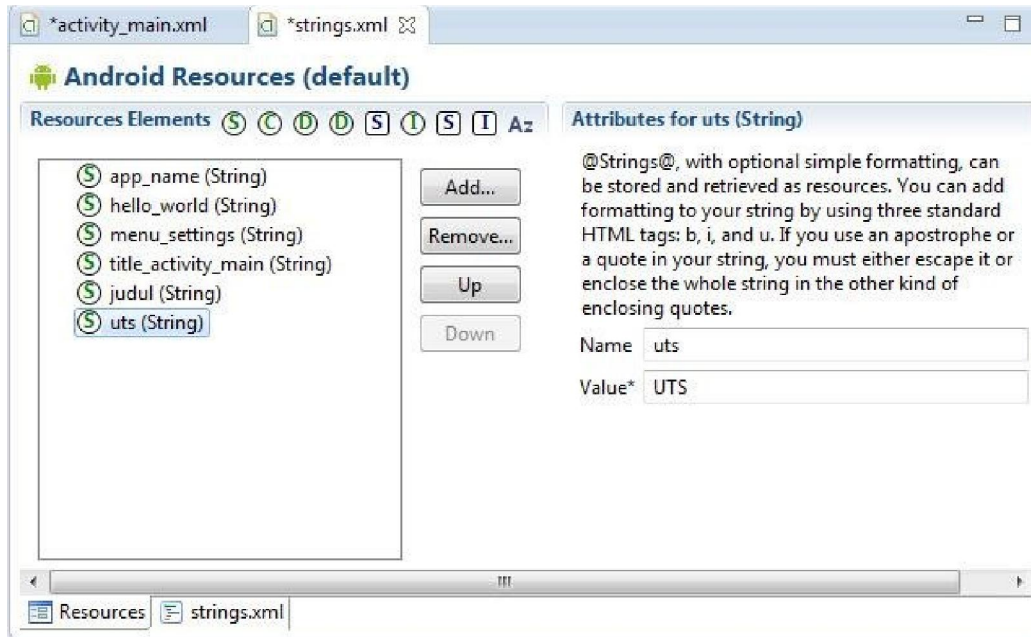
Gambar 2. 17 Menentukan jenis element

Pilih String dan klik tombol Ok untuk melanjutkan. Berikan nilai “uts” untuk name dan “UTS” untuk value seperti gambar 2.18.



Gambar 2. 18 Memberikan nilai element

Untuk meng-update nilai element, klik String di kotak sebelah kiri, sehingga tampilan menjadi seperti berikut ini:



Gambar 2. 19 Update nilai element

Untuk membuat/mengubah/menghapus element dapat dilakukan melalui source code XML dengan cara mengklik tab strings.xml di bagian bawah. Ubah source code XML menjadi seperti berikut ini:

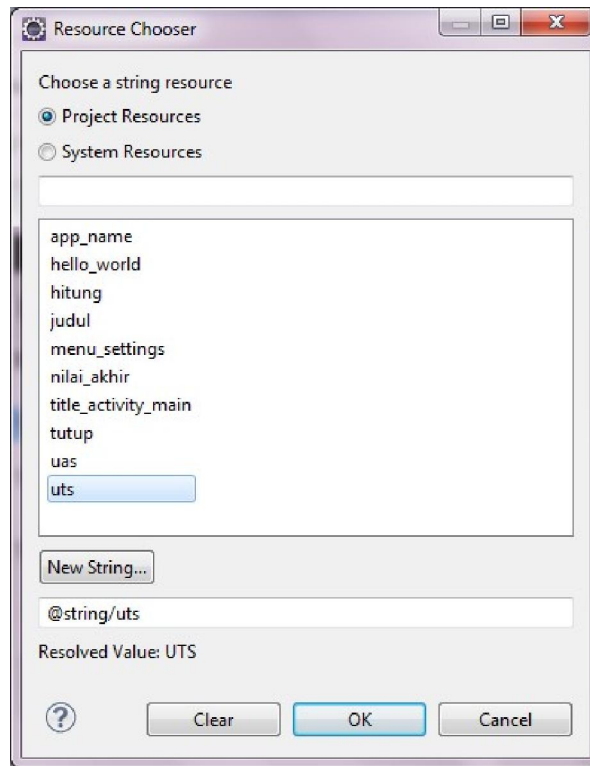
```
<resources>

<stringname="app_name">UserInterface</string>
<stringname="hello_world">Hello world!</string>
<stringname="menu_settings">Settings</string>
<stringname="title_activity_main">User Interface</string>
<stringname="judul">MenghitungNilai</string>
<stringname="uts">UTS</string>
<stringname="uas">UAS</string>
<stringname="nilai_akhir">NilaiAkhir</string>
<stringname="hitung">Hitung</string>
<stringname="tutup">Tutup</string>

<colorname="merah">#FF0000</color>
<colorname="hijau">#00FF00</color>
<colorname="biru">#0000FF</color>

</resources>
```

Kemudian simpan semua perubahan melalui menu File >> Save All, atau dari toolbar Save All. Bukakan kembali `activity_main.xml`, dan ubah teks dari `TextView` dan `Button` sehingga seperti gambar 2.21.

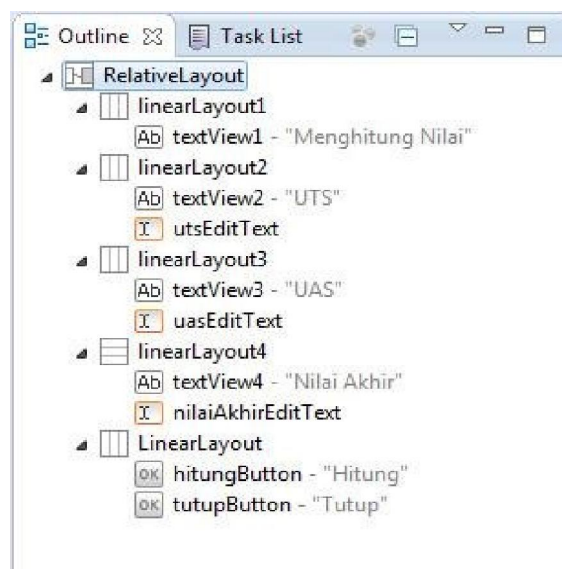


Gambar 2. 20 Memilih nama variabel reference



Gambar 2. 21 Desainaplikasi

Ubah ID dari EditText dan Button dengan cara klik kanan >> Edit ID, sehingga outline menjadi seperti berikut ini:



Gambar 2. 22 Outline aplikasi



Untuk mengatur alignment dari judul, sehingga tampil di tengah, klik kanan >> Other Properties >> All By Name >> Gravity >> Center. Ubah juga alignment dari nilai Akhir EditText agar rata kanan dengan cara klik kanan >> Other Properties >> All By Name >> Gravity >> Right.

Ubah properti layout\_width dan inputType dari EditText, clickable dan focusable dari nilai Akhir EditText, dan layout\_weight dari button melalui source code XML sehingga menjadi seperti berikut ini:

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

    <LinearLayout
        android:id="@+id/linearLayout1" android:layout_width="wrap_content" and
        roid:layout_height="wrap_content" android:layout_alignParentLeft="true"
        "android:layout_alignParentRight="true" android:layout_alignParentTop="
        "true">

        <TextView
            android:id="@+id/textView1" android:layout_width="match_pare
            nt" android:layout_height="wrap_content" android:gravity="cen
            ter"
            android:text="@string/judul"
            android:textAppearance="?android:attr/textAppearanceMedium" />

    </LinearLayout>

    <LinearLayout
        android:id="@+id/linearLayout2" android:layout_width="wrap_content" andro
        id:layout_height="wrap_content" android:layout_alignParentLeft="true" and
        roid:layout_alignParentRight="true" android:layout_below="@+id/linearLay
        out1" android:layout_marginTop="14dp">

        <TextView
            android:id="@+id/textView2" android:layout_width="wrap_content" and
            roid:layout_height="wrap_content" android:text="@string/uts" />

        <EditText
            android:id="@+id/utsEditText"

            android:layout_width="0dp"
```

```
        android:layout_height="wrap_content"android:layout_weight="1"
```

```
        android:inputType="numberDecimal">
```

```
</requestFocus/>
</EditText>
</LinearLayout>
```

```
<LinearLayout
    android:id="@+id/linearLayout3"android:layout_width="wrap_content"android:layo
    ut_height="wrap_content"android:layout_alignParentLeft="true"android:layout_al
    ignParentRight="true"android:layout_below="@+id/linearLayout2">
```

```
<TextView
    android:id="@+id/textView3"android:layout_width="wrap_content"android:layout_h
    eight="wrap_content"android:text="@string/uas"/>
```

```
<EditText
    android:id="@+id/uasEditText"
```

```
    android:layout_width="0dip"
```

```
        android:layout_height="wrap_content"android:layout_weight="1"
```

```
    android:inputType="numberDecimal"/>
```

```
</LinearLayout>
```

```
<LinearLayout
    android:id="@+id/linearLayout4"android:layout_width="wrap_content"androi
    d:layout_height="wrap_content"android:layout_alignParentLeft="true"andro
    id:layout_alignParentRight="true"android:layout_below="@+id/linearLayout
    3"android:layout_marginTop="22dp"
    android:orientation="vertical">
```

```
<TextView
    android:id="@+id/textView4"android:layout_width="wrap_content"android:layout_h
    eight="wrap_content"android:text="@string/nilai_akhir"/>
```

```
<EditText
    android:id="@+id/nilaiAkhirEditText"android:layout_width="match_par
    ent"android:layout_height="wrap_content"
    android:gravity="right"
```

```

        android:clickable="false" android:focusable="false"
        android:inputType="numberDecimal"/>

</LinearLayout>

<LinearLayout
    android:layout_width="wrap_content"android:layout_height="wrap_content"android
    :layout_alignParentLeft="true"android:layout_alignParentRight="true"android:la
    yout_below="@+id/linearLayout4"android:layout_marginTop="20dp">

    <Button
        android:id="@+id/hitungButton"android:layout_width="wrap_content "

        android:layout_height="wrap_content "
        android:layout_weight="3"
        android:text="@string/hitung"/>

    <Button
        android:id="@+id/tutupButton"android:layout_width="wrap_content"android:layout
        _height="wrap_content"

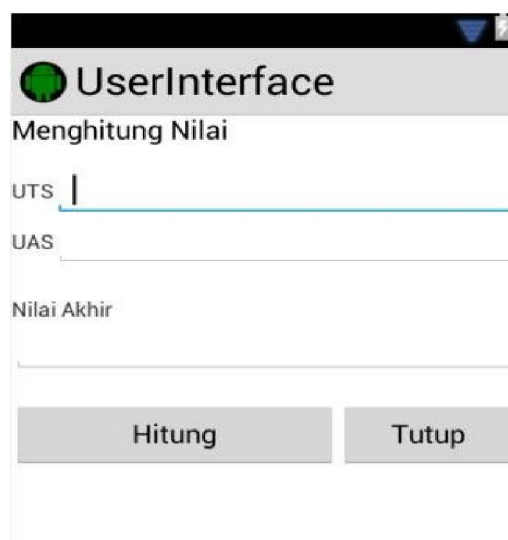
        android:layout_weight="1"
        android:text="@string/tutup"/>

</LinearLayout>

</RelativeLayout>

```

Sehinggadesainaplikasimenjadisepertiberikutini:

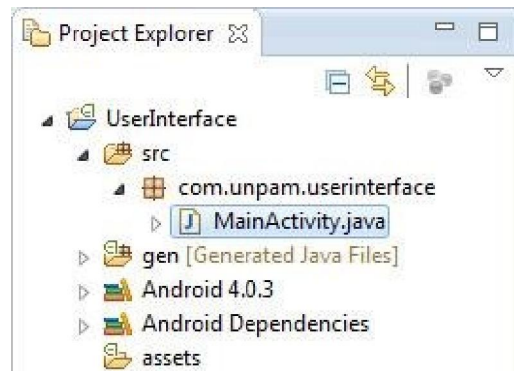


Gambar 2. 23 Desainaplikasi

Tujuan Pembelajaran 2.3:

Mengatur Proses Aplikasi

Selanjutnya double klik MainActivity.java untuk mengubah source code java yang digunakan untuk mengatur proses aplikasi. Lokasi MainActivity.java berada di folder src>>nama package aplikasi, seperti pada gambar di bawah ini:



Gambar 2. 24 Lokasi MainActivity dalam project

Ubah source code menjadi seperti berikut ini:

```

package com.unpam.userinterface;

import android.os.Bundle;
import android.app.Activity;
import android.view.Menu;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.EditText;

public class MainActivity extends Activity implements OnClickListener {
    private EditText utsET, uasET, nilaiAkhireT;

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

        utsET = (EditText) findViewById(R.id.utsEditText);
        uasET = (EditText) findViewById(R.id.uasEditText);
        nilaiAkhireT = (EditText) findViewById(R.id.nilaiAkhireEditText);

        findViewById(R.id.hitungButton).setOnClickListener(this);
        findViewById(R.id.tutupButton).setOnClickListener(this);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.activity_main, menu);
        return true;
    }

    public void onClick(View v) {
        // TODO Auto-generated method stub
        switch (v.getId()) {
            case R.id.hitungButton:
                double uts=0, uas=0, nilaiAkhire;

                try {
                    uts =
Double.parseDouble(utsET.getText().toString());
                } catch (Exception ex) {}

```

```

        try {
            uas =
Double.parseDouble(uasET.getText().toString());
        } catch (Exception ex) {}

        nilaiAkhir = (uts + uas)/2;

        if (nilaiAkhir >= 60){

nilaiAkhirET.setBackgroundResource(R.color.hijau);
        } else {

nilaiAkhirET.setBackgroundResource(R.color.merah);
        }

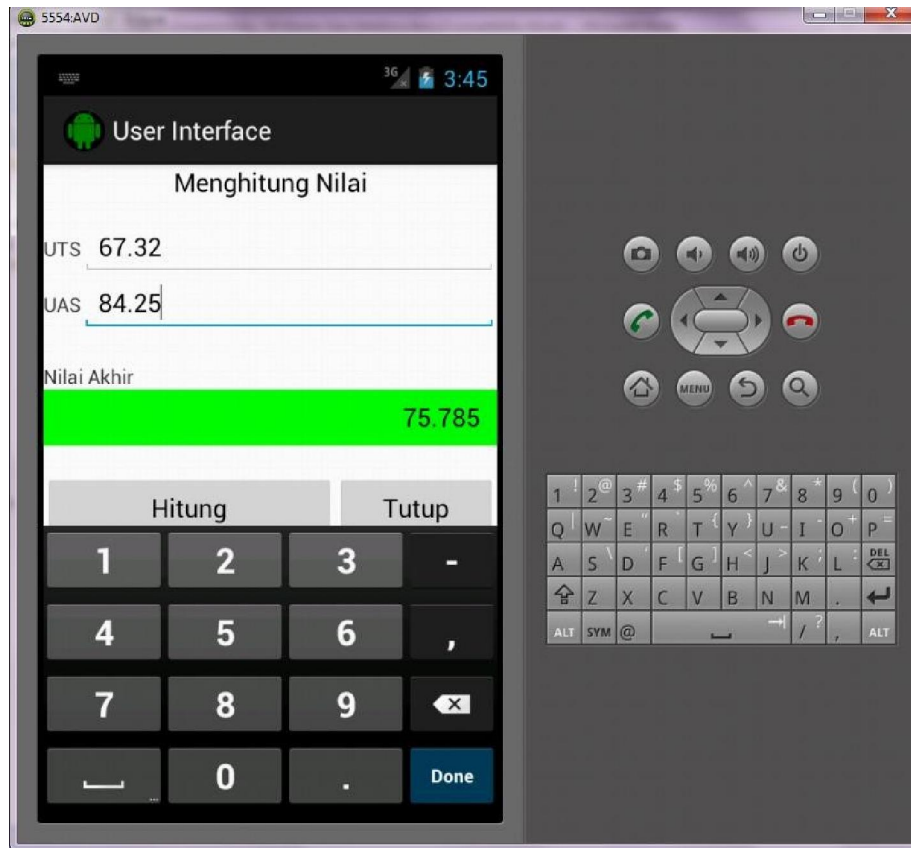
        nilaiAkhirET.setText(Double.toString(nilaiAkhir));
        break;
    case R.id.tutupButton:
        System.exit(0);
    }
}
}

```

Tujuan Pembelajaran 2.4:
--------------------------

Mengeksekusi Aplikasi
-----------------------

Untuk mengeksekusi, klik kanan pada nama project atau pilih menu Run, kemudian pilih Run As >> Android Application.



Gambar 2. 25 Tampilanaplikasidalam AVD (Android Virtual Device)

### C. SOAL LATIHAN/TUGAS

Buatlah project untuk menghitung luas segitiga!



#### D. DAFTAR PUSTAKA

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

Safaat, H. Nazruddin. 2015.ANDROID PemrogramanAplikasi Mobile Smartphone danTablet PC Berbasis Android. Bandung: Informatika