

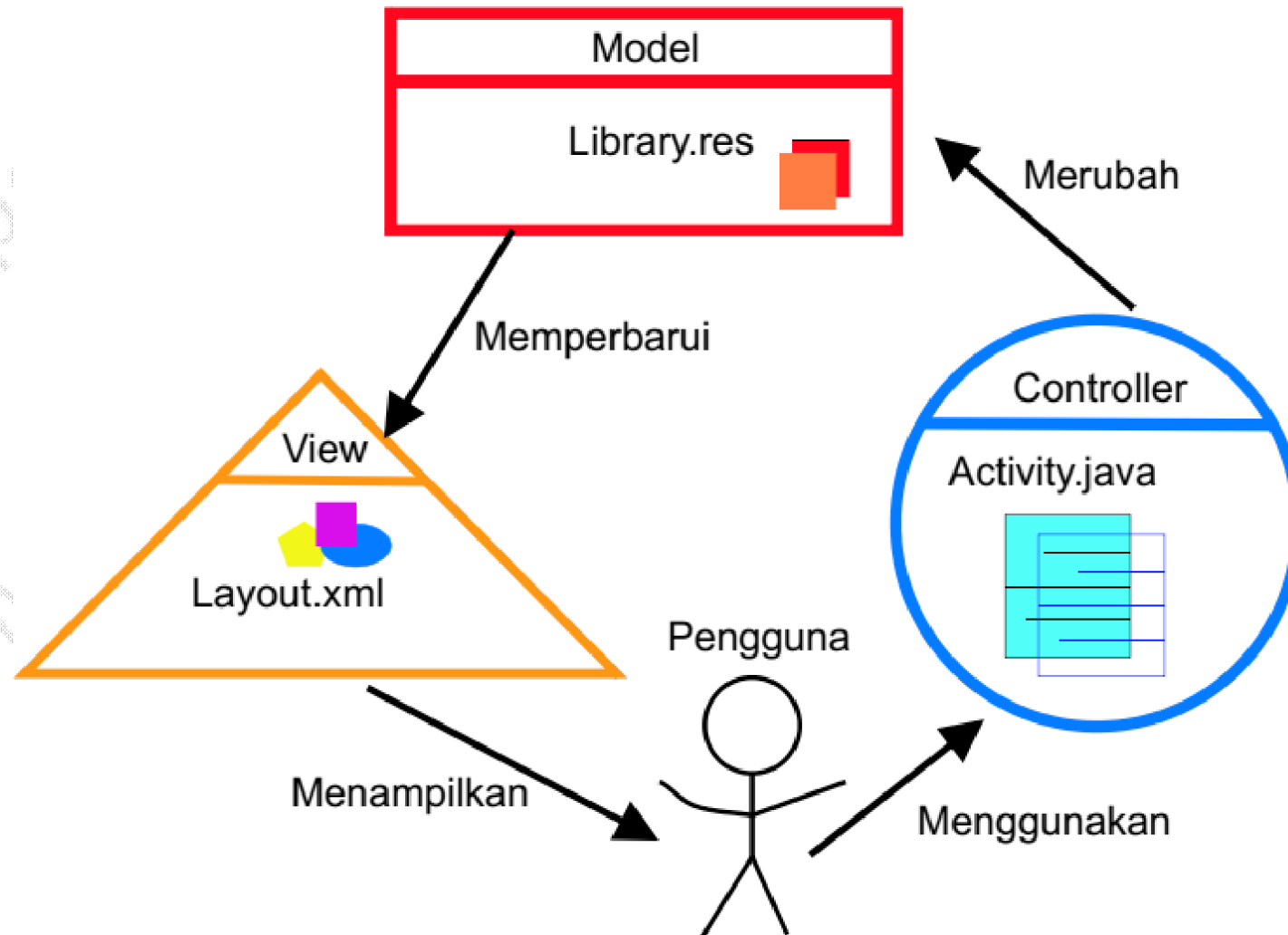
Pemrograman Android

Erwin Sutanto, S.T., M.Sc.

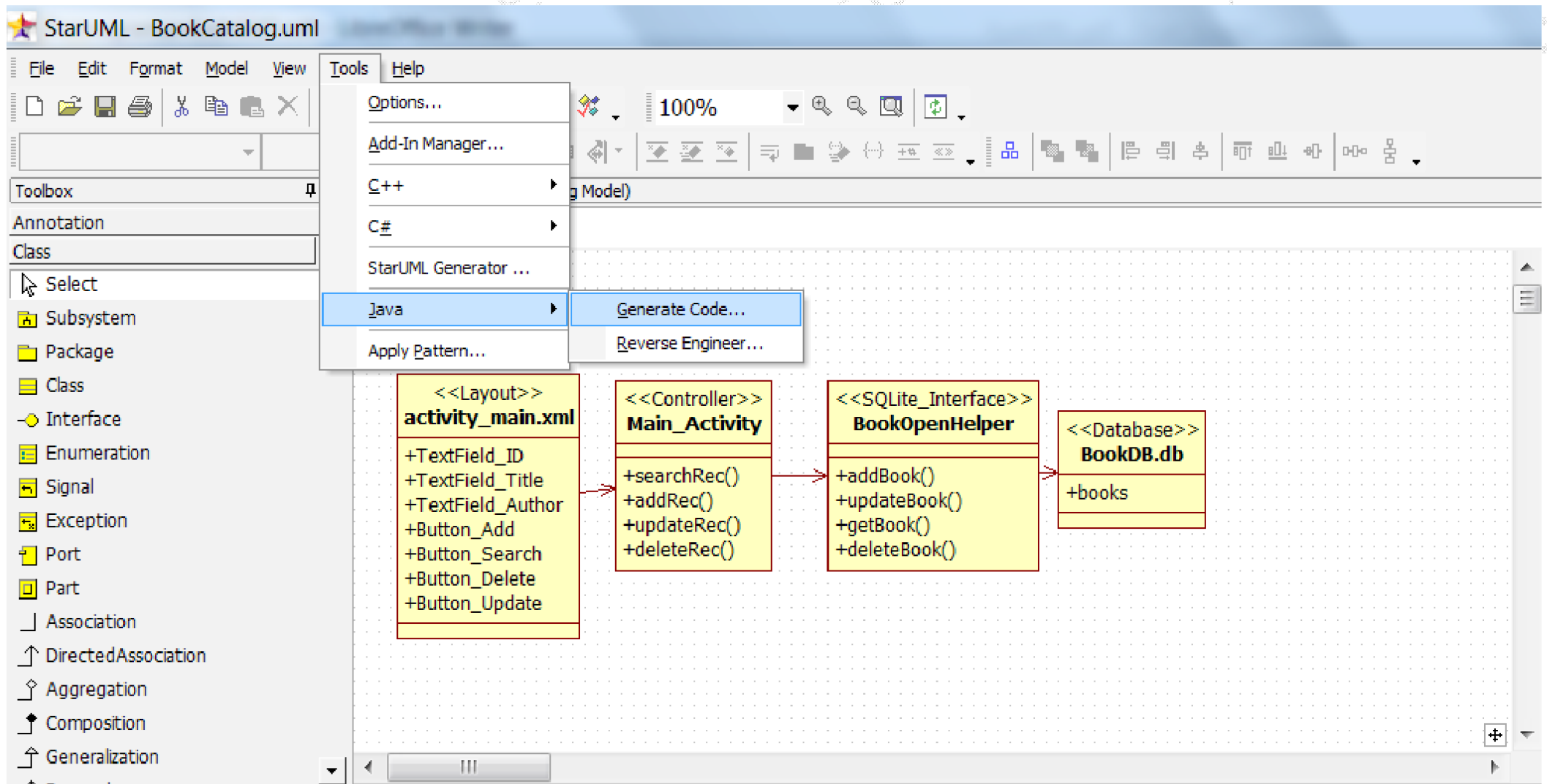
Daftar Isi

1. Model-View-Controller (MVC)
2. Realisasi MVC dengan UML
3. Implementasi dengan Java

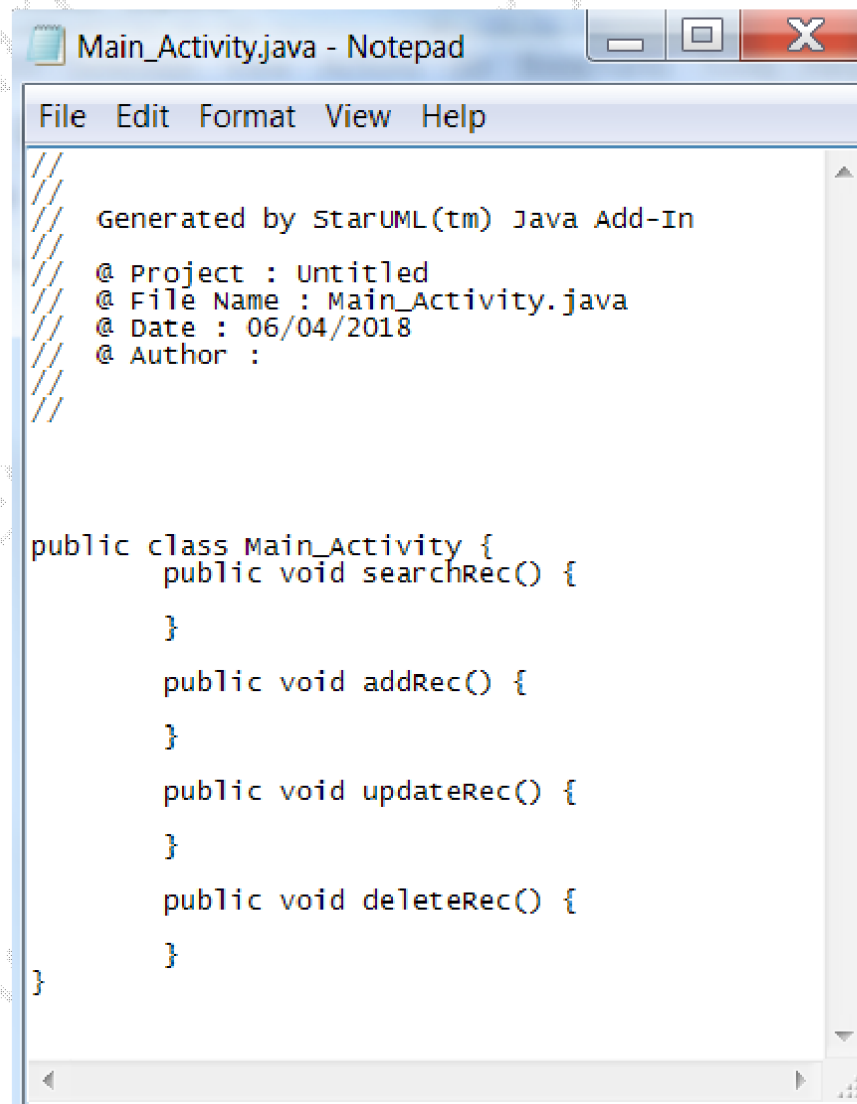
1. Model-View-Controller (MVC)



2. Realisasi MVC dengan UML



Program hasil Generate Code



```
File Edit Format View Help
///
/// Generated by StarUML(tm) Java Add-In
///
/// @ Project : Untitled
/// @ File Name : Main_Activity.java
/// @ Date : 06/04/2018
/// @ Author :
///

public class Main_Activity {
    public void searchRec() {
    }

    public void addRec() {
    }

    public void updateRec() {
    }

    public void deleteRec() {
    }
}
```

3. Implementasi dengan Java

The screenshot displays the Eclipse IDE interface for an Android project named 'HelloWorld'. The main workspace is divided into several panels:

- Package Explorer:** Shows the project structure with 'src' containing 'com.example.f' and 'MainActivity'.
- Palette:** A 'View' palette showing various Android widgets like TextView, Button, and EditText.
- Form Editor:** Displays the graphical layout of 'activity_main.xml' with a 'Hello world' text view and a 'Button' below it.
- Outline:** Shows the XML structure of the layout, including 'RelativeLayout', 'textView1', 'editText1', and 'button1'. The 'button1' is labeled as 'Model'.
- Object Properties:** Shows the properties for the selected 'Button' widget, including 'Id', 'Layout...', 'Style', 'Text', 'Hint', and 'Conte...'. The 'Text' property is set to 'Button'.

Annotations in red text and boxes identify key components:

- Controller:** Points to the 'MainActivity.java' file in the Package Explorer.
- Form Editor:** Points to the 'Graphical Layout' tab in the Form Editor.
- File .xml:** Points to the 'activity_main.xml' file in the Package Explorer.
- Object Properties:** Points to the 'Object Properties' panel.

Property	Value
Id	@+id/butto...
Layout...	[]
Style	android:but...
Text	Button
Hint	
Conte...	
TextView	[]
Text	Button

3.1 Design Database

- Di mana di dalamnya telah dibuat Use-Case Diagram, Activity Diagram, dan Class Diagramnya. Pada bagian ini, dicoba untuk melengkapi design yang kurang. Ada dua design yang perlu dibuat lebih detail. Yakni, dengan Struktur Tabel yang akan digunakan dalam aplikasi, dan detail tampilan dari aplikasi. Struktur tabel dibuat sesederhana mungkin sehingga mudah untuk diimplementasikan.

Database Name: BookDB

Table Name: books

Fields (Columns)	Data Type	Key
id	INTEGER	Primary Key
title	TEXT	
author	TEXT	

3.2 Design Layout

The screenshot displays the Android Studio IDE with the following components:

- Palette:** A sidebar on the left containing various widget categories such as Form Widgets, Text Fields, Layouts, Composite, Images & Media, Time & Date, Transitions, Advanced, and Other. The 'Form Widgets' section is currently expanded.
- Design View:** The central workspace shows a mobile device mockup (Nexus One) displaying the 'KatalogBuku' app. The app's layout includes a search bar, input fields for 'ID:', 'Title:', and 'Author:', and three buttons labeled 'Add', 'Update', and 'Delete'.
- Outline:** A window on the right showing the XML hierarchy of the layout. It is structured as follows:
 - RelativeLayout
 - LinearLayout
 - LinearLayout
 - textView1 - "ID :"
 - editText1
 - button4 - "Search"
 - LinearLayout
 - TextView2 - "Title :"
 - EditText2
 - LinearLayout
 - TextView3 - "Author :"
 - EditText3
 - LinearLayout
 - button1 - "Add"
 - button2 - "Update"
 - button3 - "Delete"
- Properties:** A window at the bottom right showing the properties for the selected 'Delete' button. The visible properties are:

Text	Delete
Hint	
Text Color	@android:color/primary_te...

3.3 SQLite DB Interface-BookOpenHelper

```
package com.example.sqliteapp;

import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.util.Log;

public class BookOpenHelper extends SQLiteOpenHelper {

    public int book id;
```

```
public BookOpenHelper(Context context) {
    super(context, "BookDB.db", null, 1);
}

@Override
public void onCreate(SQLiteDatabase db) {
    // create books table
    db.execSQL("CREATE TABLE books " +
        "(id INTEGER PRIMARY KEY, title TEXT, author TEXT)");
    Log.d("BookDB", "BookDB:CREATE books SUCCESS");
}

@Override
public void onUpgrade(SQLiteDatabase db, int arg1, int arg2) {
    // Drop older books table if existed
    db.execSQL("DROP TABLE IF EXISTS books");
    Log.d("BookDB", "BookDB:DROP books SUCCESS");
    // create fresh books table
    this.onCreate(db);
}

public void getBook(String id1) {
    SQLiteDatabase db = this.getWritableDatabase();
    String sql = "SELECT * FROM books where id=?";
    String[] args = new String[] {id1};
    Cursor cursor = db.rawQuery(sql, args);
    Log.d("BookDB", "BookDB-sql:"+sql+id1);
    if (cursor.moveToFirst()) {
        do {
            book_id = Integer.parseInt(cursor.getString(0));
            book_title = cursor.getString(1);
            book_author = cursor.getString(2);
        } while (cursor.moveToNext());
    }
}

public void addBook() {
    SQLiteDatabase db = this.getWritableDatabase();
    String sql = "INSERT INTO books (id,title,author) VALUES (" +
        "book_id+','+book_title+', '"+book_author+'";";
    Log.d("BookDB", "BookDB-sql:"+sql);
    db.execSQL(sql);
}

public void updateBook(String id1) {
    SQLiteDatabase db = this.getWritableDatabase();
    String sql = "UPDATE books SET "+ "title='"+book_title+"',
        "author='"+book_author+"' where id=?";
    String[] args = new String[] {id1};
    Log.d("BookDB", "BookDB-sql:"+sql+id1);
    db.execSQL(sql, args);
}

public void deleteBook(String id1) {
    SQLiteDatabase db = this.getWritableDatabase();
    String sql = "DELETE FROM books where id=?";
    String[] args = new String[] {id1};
}
```

3.4 Controller – Main Activity

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

public class MainActivity extends Activity {

    BookOpenHelper db;
    EditText et_id;
    EditText et_title;
    EditText et_author;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        db = new BookOpenHelper(this);
        et_id = (EditText) findViewById(R.id.editText1);
        et_title = (EditText) findViewById(R.id.editText2);
        et_author = (EditText) findViewById(R.id.editText3);
    }

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

    public void searchRec(View view) {
        String id1=et_id.getText().toString();
        db.getBook(id1);
        et_title.setText(db.book_title, TextView.BufferType.EDITABLE);
        et_author.setText(db.book_author, TextView.BufferType.EDITABLE);
    }

    public void addRec(View view) {
        db.book_id=Integer.parseInt(et_id.getText().toString());
        db.book_title=et_title.getText().toString();
        db.book_author=et_author.getText().toString();
        db.addBook();
    }

    public void updateRec(View view) {
        db.book_title=et_title.getText().toString();
        db.book_author=et_author.getText().toString();
        db.updateBook(et_id.getText().toString());
    }

    public void deleteRec(View view) {
        db.deleteBook(et_id.getText().toString());
    }
}
```

3.4.1 Add Notifikasi

```
private void alertView(String message) {  
    new AlertDialog.Builder(MainActivity.this)  
        .setTitle("Info")  
        .setMessage(message)  
        .setPositiveButton("OK", new DialogInterface.OnClickListener() {  
            public void onClick(DialogInterface dialog, int id) {  
                dialog.cancel();  
            }  
        }).show();  
}
```

3.5 Koneksikan Tombol-tombol dengan fungsi di MainActivity

Masukan ke Properties – View untuk setiap Button dengan Nama Fungsi yang ada di

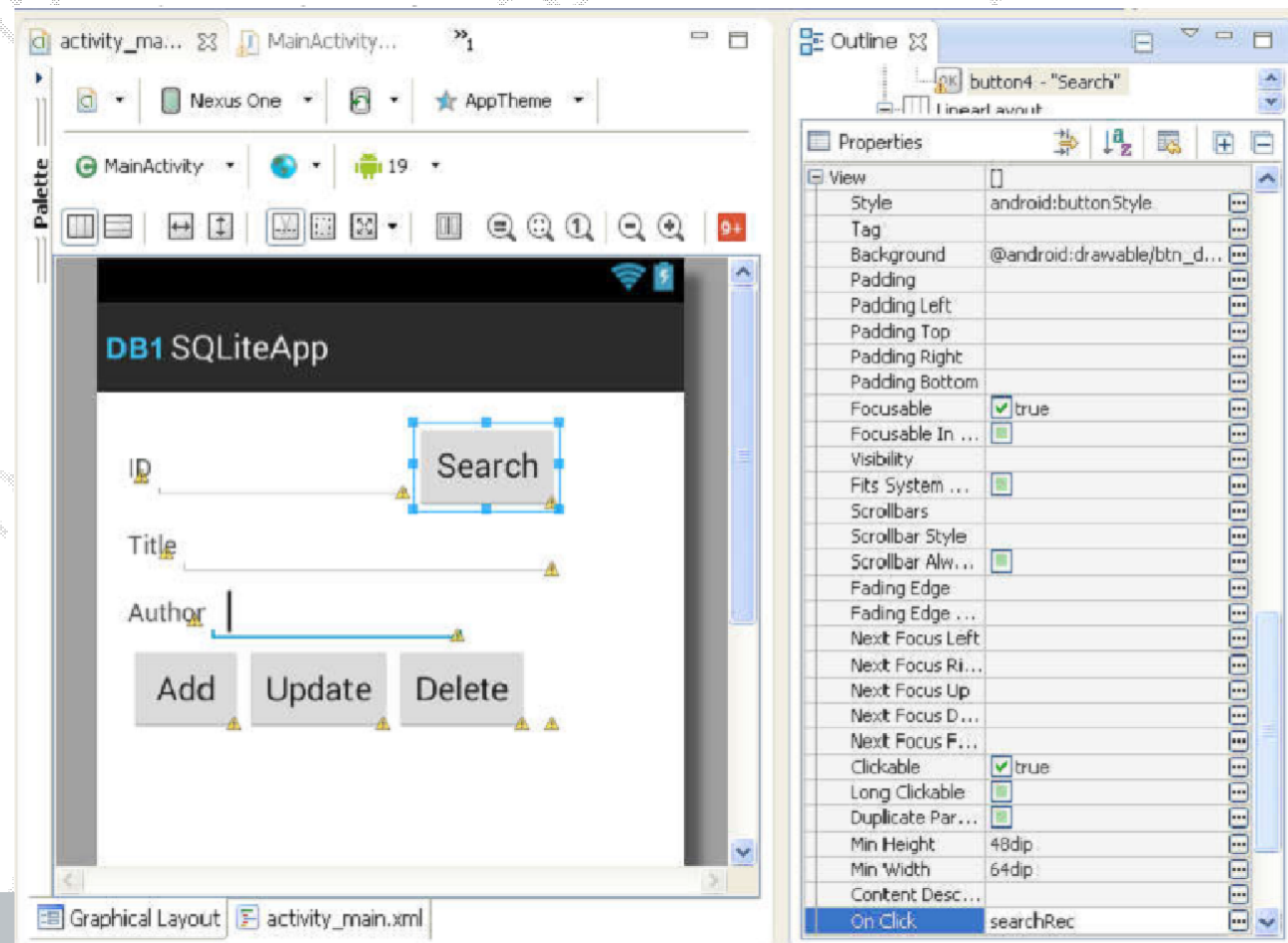
MainActivity.java

a. Search - SearchRec

b. Add - addRec

c. Update - updateRec

d. Delete - deleteRec



- Thank You -

Referensi:

Erwin S., Pemrograman android dengan menggunakan eclipse & staruml, 2018 – bab 11