

LAPORAN TUGAS
APLIKASI MOBILE PRAKTIKUM 4
“Database SQLite (Edit dan Delete)”



Oleh:

Octavio Yusda Wildani

NIM. 2211533020

Dosen Pengampu Mata Kuliah

Nurfiah, S.ST, M.Kom

DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS
PADANG
2024

A. Pelanggan Activity

Digunakan untuk mengarahkan pengguna ke PelangganEditActivity ketika sebuah item dipilih di RecyclerView. Kemudian kode ini mengambil informasi pelanggan yang dipilih dan meneruskannya ke PelangganEditActivity untuk diedit atau dihapus.

```
private View.OnClickListener onClickListener = new View.OnClickListener() { 1 usage
    @Override
    public void onClick(View v) {
        v.startAnimation(btnAnimasi);
        RecyclerView.ViewHolder viewHolder = (RecyclerView.ViewHolder) v.getTag();
        int position = viewHolder.getAdapterPosition();
        ModelPelanggan mp = list.get(position);
        Intent intent= new Intent( packageContext, PelangganActivity.this, PelangganEditActivity.class);
        intent.putExtra(Constant.ID, mp.getId());
        intent.putExtra(Constant.NAMA, mp.getNama());
        intent.putExtra(Constant.EMAIL, mp.getEmail());
        intent.putExtra(Constant.HP, mp.getHp());
        startActivity(intent);
    }
};
```

B. SQLHelper

Fungsi deletePelanggan ditambahkan pada kelas SQLiteHelper untuk menghapus data pelanggan berdasarkan ID. Fungsi ini menerima parameter ID pelanggan, menghapus data berdasarkan ID tersebut, dan mengembalikan boolean sebagai indikator keberhasilan.

```
public boolean deletePelanggan(String id) { 1 usage
    SQLiteDatabase db = this.getWritableDatabase();
    int result = db.delete(TABLE_PELANGGAN, whereClause: KEY_PELANGGAN_ID + " = ?", new String[]{id});
    db.close();
    return result > 0;
}

}
```

C. Pelanggan Edit Activity

Kode ini merupakan bagian dari aplikasi Android yang bertujuan untuk mengelola data pelanggan, khususnya dalam mengedit dan menghapus informasi pelanggan yang tersimpan dalam database lokal SQLite. Pada aktivitas ini, data pelanggan (seperti nama, email, dan nomor hp) diterima dari PelangganActivity dan ditampilkan dalam kolom input untuk diedit. Pengguna dapat memperbarui data dengan mengetikkan informasi baru dan menekan tombol "Simpan", yang kemudian akan memvalidasi apakah semua data telah diisi. Jika valid, data pelanggan diperbarui di database, dan pengguna kembali ke halaman utama pelanggan. Ada pula tombol "Hapus" yang menampilkan dialog konfirmasi sebelum menghapus data pelanggan. Jika pengguna mengonfirmasi, data tersebut dihapus dari database, dan PelangganActivity diperbarui untuk merefleksikan perubahan ini. Terakhir, terdapat tombol "Batal" yang memungkinkan pengguna membatalkan pengeditan dan menutup halaman ini tanpa menyimpan perubahan.

Java:

```
package com.mylauvi.laundryapps.pelanggan;

import ...

public class PelangganEditActivity extends AppCompatActivity {

    private String id, name, email, hp; 3 usages
    private EditText edPelEditNama, edPelEditEmail, edPelEditHp; 3 usages
    private Button btnPelEditSimpan, btnPelEditHapus, btnPelEditBatal; 2 usages
    private SQLiteHelper db; 3 usages

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        EdgeToEdge.enable(this);
        setContentView(R.layout.activity_pelanggan_edit);

        id = getIntent().getStringExtra(Constant.ID);
        name = getIntent().getStringExtra(Constant.NAMA);
        email = getIntent().getStringExtra(Constant.EMAIL);
        hp = getIntent().getStringExtra(Constant.HP);

        db = new SQLiteHelper(context, this);
        edPelEditNama = findViewById(R.id.edPelEditNama);
        edPelEditEmail = findViewById(R.id.edPelEditEmail);
        edPelEditHp = findViewById(R.id.edPelEditHp);
        btnPelEditSimpan = findViewById(R.id.btnPelEditSimpan);
        btnPelEditHapus = findViewById(R.id.btnPelEditHapus);
        btnPelEditBatal = findViewById(R.id.btnPelEditBatal);

        // Set data pelanggan ke EditText
        edPelEditNama.setText(name);
        edPelEditEmail.setText(email);
        edPelEditHp.setText(hp);

        // Tombol Simpan untuk memperbarui data pelanggan
        btnPelEditSimpan.setOnClickListener(v -> {
            ModelPelanggan mp = new ModelPelanggan();
            mp.setId(id);
            mp.setName(edPelEditNama.getText().toString());
            mp.setEmail(edPelEditEmail.getText().toString());
            mp.setHp(edPelEditHp.getText().toString());

            if (db.updatePelanggan(mp)) {
                Toast.makeText(context, this, text: "Pelanggan berhasil diperbarui", Toast.LENGTH_SHORT).show();
                finish(); // Kembali ke PelangganActivity setelah berhasil
            } else {
                Toast.makeText(context, this, text: "Gagal memperbarui pelanggan", Toast.LENGTH_SHORT).show();
            }
        });

        // Set tombol hapus untuk menghapus data pelanggan
        btnPelEditHapus.setOnClickListener(v -> {
            if (db.deletePelanggan(id)) {
                Toast.makeText(context, this, text: "Pelanggan berhasil dihapus", Toast.LENGTH_SHORT).show();
                finish(); // Kembali ke PelangganActivity setelah dihapus
            } else {
                Toast.makeText(context, this, text: "Gagal menghapus pelanggan", Toast.LENGTH_SHORT).show();
            }
        });
    }
}
```

XML:

[illegible]

```

64         android:drawableLeft="@drawable/ic_email"
65         android:drawablePadding="8dp"
66         android:inputType="textEmailAddress"
67         android:padding="12dp"
68         android:hint="Masukkan email anda"/>
69     </LinearLayout>
70
71     <LinearLayout
72         android:layout_marginTop="12dp"
73         android:layout_width="match_parent"
74         android:layout_height="wrap_content"
75         android:orientation="vertical">
76         <TextView
77             android:layout_width="match_parent"
78             android:layout_height="wrap_content"
79             android:text="HP"
80             android:textColor="@android:color/black"/>
81         <EditText
82             android:id="@+id/edPelEditHp"
83             android:layout_width="match_parent"
84             android:layout_height="wrap_content"
85             android:layout_marginTop="8dp"
86             android:background="@drawable/border"
87             android:drawableLeft="@drawable/ic_hp"
88             android:drawablePadding="8dp"
89             android:inputType="number"
90             android:padding="12dp"

```

```

91         android:hint="Masukkan nomor anda" />
92     </LinearLayout>
93
94     <LinearLayout
95         android:layout_marginTop="20dp"
96         android:layout_width="match_parent"
97         android:layout_height="wrap_content"
98         android:orientation="horizontal"
99         android:weightSum="3">
100         <Button
101             android:id="@+id/btnPelEditSimpan"
102             android:layout_width="match_parent"
103             android:layout_height="wrap_content"
104             android:layout_marginRight="4dp"
105             android:layout_weight="1"
106             android:backgroundTint="@color/hijau1"
107             android:text="Simpan"
108             android:textColor="@color/white" />
109         <Button
110             android:id="@+id/btnPelEditHapus"
111             android:layout_width="match_parent"
112             android:layout_height="wrap_content"
113             android:layout_marginRight="4dp"
114             android:layout_weight="1"
115             android:backgroundTint="@color/hijau1"
116             android:text="Hapus"
117             android:textColor="@color/white" />
118         <Button
119             android:id="@+id/btnPelEditBatal"
120             android:layout_marginLeft="4dp"


```

```
121 ■ android:textColor="@color/red"
122 ■ android:backgroundTint="@color/hijau1"
123 android:layout_width="match_parent"
124 android:layout_height="wrap_content"
125 android:layout_weight="1"
126 android:text="Batal"/>
127 </LinearLayout>
128 </LinearLayout>
129 </LinearLayout>
```


Output:

PELANGGAN


Nama

 vio

Email

 vio0@gmail.com

HP

 089976543

Simpan

Batal

Tambah Pelanggan

ini

0192182

vio

089976543

D. Layanan Activity

Tahap ini merupakan implementasi fitur pengeditan data pada aplikasi Android yang menggunakan SQLite Database. Kode ini mendefinisikan sebuah *OnClickListener* untuk menangani klik pada item dalam *RecyclerView*. Saat item diklik, animasi pada tombol akan dijalankan menggunakan *startAnimation()*. Kemudian, posisi item yang diklik diambil menggunakan *getAdapterPosition()*, dan data dari item tersebut diakses melalui daftar data menggunakan model data yang sesuai. Setelah itu, sebuah *intent* dibuat untuk membuka aktivitas *LayananEditActivity*, yang berfungsi sebagai layar pengeditan data. Data layanan seperti ID, tipe, dan harga dikirimkan ke aktivitas edit menggunakan metode *putExtra()* dalam *intent*. Terakhir, aktivitas pengeditan dimulai dengan memanggil *startActivity(intent)*. Kode ini memungkinkan pengguna untuk mengedit data yang diambil dari SQLite Database melalui antarmuka aplikasi.

```

49     }
50     private View.OnClickListener onClickListener = new View.OnClickListener() { 1 usage
51         @Override
52         public void onClick(View view) {
53             view.startAnimation(btnAnimasi);
54             RecyclerView.ViewHolder viewHolder = (RecyclerView.ViewHolder)
55                 view.getTag();
56             int position = viewHolder.getAdapterPosition();
57             ModelLayanan mp = list.get(position);
58             Intent intent= new Intent( packageContext: LayananActivity.this, LayananEditActivity.class);
59             intent.putExtra(Constant.ID, mp.getId());
60             intent.putExtra(Constant.TIPE, mp.getTipe());
61             intent.putExtra(Constant.HARGA, mp.getHarga());
62             startActivity(intent);
63         }
64     };

```

E. SQLHelper2

Tahap ini merupakan implementasi fungsi penghapusan data dalam aplikasi Android yang menggunakan SQLite Database.

```

83     // Fungsi untuk menghapus layanan berdasarkan id
84     public boolean deleteLayanan(String id) { 1 usage
85         SQLiteDatabase db = this.getWritableDatabase();
86         // Menghapus data layanan dengan id tertentu
87         return db.delete(TABLE_LAYANAN, whereClause: KEY_LAYANAN_ID + "=?", new String[]{id}) > 0;
88     }

```

F. Layanan Edit Activity

Tahap ini berperan sebagai antarmuka utama bagi pengguna untuk memperbarui atau menghapus data layanan secara langsung, memastikan sinkronisasi perubahan dengan database SQLite.

```

1     package id.my.lauvi.laundryapps.layanan;
2
3     > import ...
4
15
16     public class LayananEditActivity extends AppCompatActivity {
17
18         private String id, tipe, harga; 3 usages
19         private EditText edLayEditLayanan, edLayEditHarga; 3 usages
20         private Button btnLayEditSimpan, btnLayEditHapus, btnLayEditBatal; 2 usages
21         private SQLiteHelper2 db; 3 usages
22
23         @SuppressWarnings("MissingInflatedId")
24         @Override
25         protected void onCreate(Bundle savedInstanceState) {
26             super.onCreate(savedInstanceState);
27             EdgeToEdge.enable( $this$SenableEdgeToEdge: this);
28             setContentView(R.layout.activity_layanan_edit);
29
30             // Get data from intent
31             id = getIntent().getStringExtra(Constant.ID);
32             tipe = getIntent().getStringExtra(Constant.TIPE);
33             harga = getIntent().getStringExtra(Constant.HARGA);
34
35             // Initialize database helper
36             db = new SQLiteHelper2( context: this);
37
38             // Bind UI elements
39             edLayEditLayanan = findViewById(R.id.edLayEditLayanan);
40             edLayEditHarga = findViewById(R.id.edLayEditHarga);
41             btnLayEditSimpan = findViewById(R.id.btnLayEditSimpan);
42             btnLayEditHapus = findViewById(R.id.btnLayEditHapus);
43             btnLayEditBatal = findViewById(R.id.btnLayEditBatal);
44

```

```

45 // Set initial values in EditText fields
46 edLayEditLayanan.setText(tipe);
47 edLayEditHarga.setText(harga);
48
49 // Handle "Hapus" button click to delete data
50 btnLayEditHapus.setOnClickListener(v -> {
51     if (db.deleteLayanan(id)) {
52         Toast.makeText( context: this, text: "Layanan berhasil dihapus", Toast.LENGTH_SHORT).show();
53         setResult(RESULT_OK);
54         finish();
55     } else {
56         Toast.makeText( context: this, text: "Gagal menghapus layanan", Toast.LENGTH_SHORT).show();
57     }
58 });
59
60 // Handle "Batal" button click to cancel and return to the previous activity
61 btnLayEditBatal.setOnClickListener(v -> finish());
62
63 // Handle "Simpan" button click to update the service
64 btnLayEditSimpan.setOnClickListener(v -> {
65     String newTipe = edLayEditLayanan.getText().toString().trim();
66     String newHarga = edLayEditHarga.getText().toString().trim();
67
68     // Validate input
69     if (newTipe.isEmpty() || newHarga.isEmpty()) {
70         Toast.makeText( context: this, text: "Tipe layanan dan harga tidak boleh kosong", Toast.LENGTH_SHORT).show();
71         return;
72     }
73
74     // Update the layanan in the database
75     boolean isUpdated = db.updateLayanan(id, newTipe, newHarga);
76
77     if (isUpdated) {
78         Toast.makeText( context: this, text: "Layanan berhasil diperbarui", Toast.LENGTH_SHORT).show();
79         setResult(RESULT_OK);
80         finish();
81     } else {
82         Toast.makeText( context: this, text: "Gagal memperbarui layanan", Toast.LENGTH_SHORT).show();
83     }
84 });
85 }
86 }

```

Output:

Tambah Layanan

cuci
120000

gosok
1213

gosok
1444444

LAYANAN

Layanan

cuci

Harga

\$ 120000

Simpan

Hapus

Batal

Tambah Layanan

gosok

1213

gosok

1444444