TUGAS AKHIR

ARIF FRIMA ARI SUWADJI – 221011700443

Flowchart

```
        ┌─────────────┐
        │    start     │
        └─────────────┘
              │
              ▼
        ╱─────────────╲
        │input pencarian│
        ╲─────────────╱
              │
              ▼
        ┌─────────────┐
        │    buka      │
        │   koneksi    │
        │  Database    │
        └─────────────┘
              │
              ▼
        ┌─────────────┐
        │search, get data│
        └─────────────┘
              │
              ▼
        ╱─────────────╲
        │tampilkan data │
        ╲─────────────╱
              │
              ▼
        ┌─────────────┐
        │    tutup     │
        │   koneksi    │
        │   databse    │
        └─────────────┘
              │
              ▼
        ┌─────────────┐
        │     end      │
        └─────────────┘
```

```
                        start

                 login / registrasi
                    pengguna

                      konek
                    database

        ┌──────────────────────────────────────────────────────────────┐
        │                                                                │
        │          Menu                                                  │
        │       - Jual Mobil                                             │
        │       - Cari Mobil                                             │
        │      - Lihat Transaksi                                         │
        │         - Keluar                                               │

              Jual Mobil   No    Cari Mobil   no   lihat          keluar
                  ?                  ?              transkasi        ?
                                                      ?
               Yes              yes             yes             yes

           input informasi   input kriteria   cari tranksaksi   logout
           mobil yang dijual     mobil         terkait           pengguna
                                               pengguna
                                                                 tutup
           Validasi Data     cari mobil        tampil data       koneksi
              Mobil          berdasarkan       transaksi         database
                            kriteria
                                                                   end
            Tampilkan        Tampilkan hasil   pilih transaksi yang
           Notifikasi        pencarian         diinginkan
     No
                             pilih mobil yang   tampilkan detail
                             diiginkan          transaksi

                             tampilkan
                             informasi
                             detail mobil

                                Beli Mobil
                                    ?

                                  yes

                             input data pembeli
                             dan detail pembayaran

                             Validasi Data Pembeli
                             dan pembayaran

                             simpan data
                             transaksi ke
                             database

                             tampilkan
                             notifikasi
                             berhasil
```

Code app.py:

```python
# app.py > ...

1   from flask import Flask, render_template, request
2   from mysql_connector import create_connection, close_connection, get_mobil
3
4   app = Flask(__name__)
5
6   # Route untuk halaman utama
7   @app.route('/')
8   def index():
9       return render_template('index.html')
10
11  # Route untuk hasil pencarian
12  @app.route('/search', methods=['GET'])
13  def search():
14      brand = request.args.get('brand')
15      min_price = request.args.get('min_price')
16      max_price = request.args.get('max_price')
17
18      # Membuat koneksi ke database
19      connection = create_connection()
20
21      # Mendapatkan data mobil dari database
22      mobil = get_mobil(connection)
23
24      #dump data mobil
25      print(mobil)
26
27      # Menutup koneksi
28      close_connection(connection)
29
30      # Render template HTML dan kirimkan data mobil ke template
31      return render_template('result.html', mobils=mobil)
32
```

```python
33  # Route untuk halaman input mobil
34  @app.route('/input', methods=['GET', 'POST'])
35  def input_mobil():
36      if request.method == 'POST':
37          # Ambil data dari form input
38          nama = request.form['nama']
39          tahun = int(request.form['tahun'])
40          harga = int(request.form['harga'])
41          merk = request.form['merk']
42          jenis = request.form['jenis']
43          nopol = request.form['nopol']
44
45          # Membuat koneksi ke database
46          connection = create_connection()
47
48          # Memasukkan data mobil ke database
49          insert_mobil(connection, nama, tahun, harga, merk, jenis, nopol)
50
51          # Menutup koneksi
52          connection.close()
53
54          # Redirect ke halaman utama setelah berhasil memasukkan data
55          return redirect('/')
56      else:
57          return render_template('input.html')
58
59  @app.route('/konsumen')
60  def konsumen():
61      connection = create_connection()
62      cursor = connection.cursor()
63      query = "SELECT * FROM konsumen"
64      cursor.execute(query)
65      result = cursor.fetchall()
66      data_konsumen = result
67      close_connection(connection)
68      return render_template('konsumen.html', data_konsumen=data_konsumen)
69
```

```python
app.py > ...
 69
 70    @app.route('/pegawai')
 71    def pegawai():
 72        # Mengambil data pegawai dari database
 73        connection = create_connection()
 74        cursor = connection.cursor()
 75        query = "SELECT * FROM pegawai"
 76        cursor.execute(query)
 77        result = cursor.fetchall()
 78        data_pegawai = result
 79        close_connection(connection)
 80        return render_template('pegawai.html', data_pegawai=data_pegawai)
 81
 82    @app.route('/divisi')
 83    def divisi():
 84        # Mengambil data divisi dari database
 85        connection = create_connection()
 86        cursor = connection.cursor()
 87        query = "SELECT * FROM divisi"
 88        cursor.execute(query)
 89        result = cursor.fetchall()
 90        data_divisi = result
 91        close_connection(connection)
 92        return render_template('divisi.html', data_divisi=data_divisi)
 93
 94    @app.route('/jabatan')
 95    def jabatan():
 96        # Mengambil data jabatan dari database
 97        connection = create_connection()
 98        cursor = connection.cursor()
 99        query = "SELECT * FROM jabatan"
100        cursor.execute(query)
101        result = cursor.fetchall()
102        data_jabatan = result
103        close_connection(connection)
104        return render_template('jabatan.html', data_jabatan=data_jabatan)
```
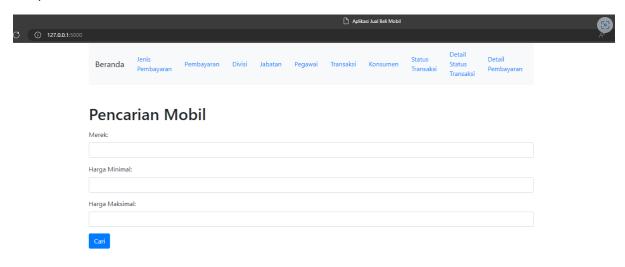
```python
106    @app.route('/transaksi')
107    def transaksi():
108        # Mengambil data transaksi dari database
109        connection = create_connection()
110        cursor = connection.cursor()
111        query = "SELECT * FROM transaksi"
112        cursor.execute(query)
113        result = cursor.fetchall()
114        data_transaksi = result
115        close_connection(connection)
116        return render_template('transaksi.html', data_transaksi=data_transaksi)
117
118    @app.route('/status_transaksi')
119    def status_transaksi():
120        # Mengambil data status transaksi dari database
121        connection = create_connection()
122        cursor = connection.cursor()
123        query = "SELECT * FROM status_transaksi"
124        cursor.execute(query)
125        result = cursor.fetchall()
126        data_status_transaksi = result
127        close_connection(connection)
128        return render_template('status_transaksi.html', data_status_transaksi=data_status_transaksi)
129
130    @app.route('/detail_status_transaksi')
131    def detail_status_transaksi():
132        # Mengambil data detail status transaksi dari database
133        connection = create_connection()
134        cursor = connection.cursor()
135        query = "SELECT * FROM detail_status_transaksi"
136        cursor.execute(query)
137        result = cursor.fetchall()
138        data_detail_status_transaksi = result
139        close_connection(connection)
140        return render_template('detail_status_transaksi.html', data_detail_status_transaksi=data_detail_status_transaksi)
141
```
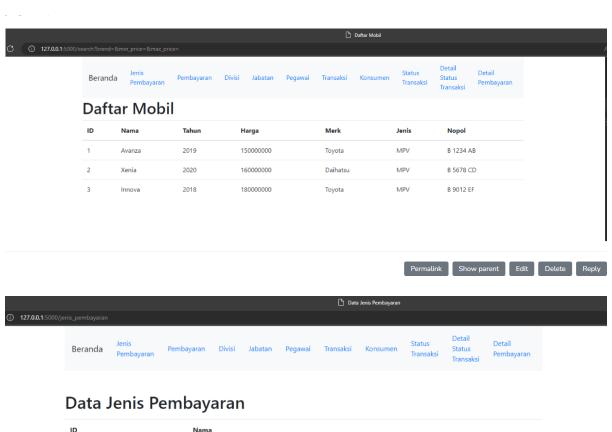
```python
142    @app.route('/detail_pembayaran')
143    def detail_pembayaran():
144        # Mengambil data detail pembayaran dari database
145        connection = create_connection()
146        cursor = connection.cursor()
147        query = "SELECT * FROM detail_pembayaran"
148        cursor.execute(query)
149        result = cursor.fetchall()
150        data_detail_pembayaran = result
151        close_connection(connection)
152        return render_template('detail_pembayaran.html', data_detail_pembayaran=data_detail_pembayaran)
153
154    @app.route('/jenis_pembayaran')
155    def jenis_pembayaran():
156        # Mengambil data jenis pembayaran dari database
157        connection = create_connection()
158        cursor = connection.cursor()
159        query = "SELECT * FROM jenis_pembayaran"
160        cursor.execute(query)
161        result = cursor.fetchall()
162        data_jenis_pembayaran = result
163        close_connection(connection)
164        return render_template('jenis_pembayaran.html', data_jenis_pembayaran=data_jenis_pembayaran)
165
166    @app.route('/pembayaran')
167    def pembayaran():
168        # Mengambil data pembayaran dari database
169        connection = create_connection()
170        cursor = connection.cursor()
171        query = "SELECT * FROM pembayaran"
172        cursor.execute(query)
173        result = cursor.fetchall()
174        data_pembayaran = result
175        close_connection(connection)
176        return render_template('pembayaran.html', data_pembayaran=data_pembayaran)
177
178    if __name__ == '__main__':
179        app.run(debug=True)
180
```

Code connection db:

```python
 mysql_connector.py > ...

1    import mysql.connector
2    from mysql.connector import Error
3
4    # Fungsi untuk membuat koneksi ke database MySQL
5    def create_connection():
6        connection = None
7        try:
8            connection = mysql.connector.connect(
9                host='localhost',
10               database='sim_app',
11               user='root',
12               password=''
13           )
14           if connection.is_connected():
15               print("Berhasil terhubung ke database")
16               return connection
17
18       except Error as e:
19           print(f"Error: {e}")
20
21       return connection
22
23   # Fungsi untuk menutup koneksi ke database
24   def close_connection(connection):
25       if connection:
26           connection.close()
27           print("Koneksi database ditutup")
28
```

```python
 mysql_connector.py > ...
40
41   # Fungsi untuk membaca data mobil dari tabel
42   def get_mobil(connection):
43       try:
44           cursor = connection.cursor()
45           query = "SELECT * FROM mobil"
46           cursor.execute(query)
47           result = cursor.fetchall()
48           print("Data mobil berhasil didapatkan")
49           print(result)
50           return result
51
52       except Error as e:
53           print(f"Error: {e}")
54           return []
55
56   # Fungsi untuk mengupdate data mobil di tabel berdasarkan ID
```

Tampilan web

127.0.0.1:5000

Beranda | Jenis Pembayaran | Pembayaran | Divisi | Jabatan | Pegawai | Transaksi | Konsumen | Status Transaksi | Detail Status Transaksi | Detail Pembayaran

Pencarian Mobil

Merek:

Harga Minimal:

Harga Maksimal:

Cari

127.0.0.1:5000/search?brand=&min_price=&max_price=

Beranda | Jenis Pembayaran | Pembayaran | Divisi | Jabatan | Pegawai | Transaksi | Konsumen | Status Transaksi | Detail Status Transaksi | Detail Pembayaran

Daftar Mobil

ID	Nama	Tahun	Harga	Merk	Jenis	Nopol
1	Avanza	2019	150000000	Toyota	MPV	B 1234 AB
2	Xenia	2020	160000000	Daihatsu	MPV	B 5678 CD
3	Innova	2018	180000000	Toyota	MPV	B 9012 EF

Permalink | Show parent | Edit | Delete | Reply

127.0.0.1:5000/jenis_pembayaran

Beranda | Jenis Pembayaran | Pembayaran | Divisi | Jabatan | Pegawai | Transaksi | Konsumen | Status Transaksi | Detail Status Transaksi | Detail Pembayaran

Data Jenis Pembayaran

ID	Nama
1	Transfer Bank
2	Tunai
3	E-wallet

127.0.0.1:5000/pembayaran

Beranda | Jenis Pembayaran | Pembayaran | Divisi | Jabatan | Pegawai | Transaksi | Konsumen | Status Transaksi | Detail Status Transaksi | Detail Pembayaran

Data Pembayaran

ID	Tanggal	Nomor Faktur	Jenis Pembayaran	Pegawai	Keterangan
1	2023-07-01	INV001	1	1	Pembayaran Transfer Bank
2	2023-07-02	INV002	2	2	Pembayaran Tunai
3	2023-07-03	INV003	3	3	Pembayaran melalui E-wallet

detail pembayaran

127.0.0.1:5000/detail_pembayaran

Beranda | Jenis Pembayaran | Pembayaran | Divisi | Jabatan | Pegawai | Transaksi | Konsumen | Status Transaksi | Detail Status Transaksi | Detail Pembayaran

Data Detail Pembayaran

Pembayaran	Transaksi	Qty	Jumlah	Total
1	1	2	5000000	10000000
1	2	1	3000000	3000000
2	3	3	1500000	4500000

127.0.0.1:5000/transaksi

Beranda | Jenis Pembayaran | Pembayaran | Divisi | Jabatan | Pegawai | Transaksi | Konsumen | Status Transaksi | Detail Status Transaksi | Detail Pembayaran

Data Transaksi

ID	Tanggal	Nomor Faktur	Konsumen	Mobil	Pegawai	Harga
1	2023-07-01	ABC123	1	1	2	250000000
2	2023-07-02	DEF456	2	3	1	150000000
3	2023-07-03	GHI789	3	2	3	180000000

🔄 ⓘ 127.0.0.1:5000/status_transaksi

Beranda | Jenis Pembayaran | Pembayaran | Divisi | Jabatan | Pegawai | Transaksi | Konsumen | Status Transaksi | Detail Status Transaksi | Detail Pembayaran

Data Status Transaksi

ID	Transaksi	Pembayaran
1	1	1
2	2	2
3	3	3

🔄 ⓘ 127.0.0.1:5000/detail_status_transaksi

Beranda | Jenis Pembayaran | Pembayaran | Divisi | Jabatan | Pegawai | Transaksi | Konsumen | Status Transaksi | Detail Status Transaksi | Detail Pembayaran

Data Detail Status Transaksi

Status Transaksi	Tanggal	Status Akhir	Tanggal Update
1	2023-07-01	Proses Dokumen	2023-07-01 16:02:42
1	2023-07-02	Proses Unit	2023-07-01 16:02:42
1	2023-07-03	Proses SPK	2023-07-01 16:02:42
2	2023-07-02	Proses Dokumen	2023-07-01 16:03:01
2	2023-07-03	Proses Unit	2023-07-01 16:03:01
2	2023-07-04	Proses SPK	2023-07-01 16:03:01
3	2023-07-02	Proses Dokumen	2023-07-01 16:04:16
3	2023-07-03	Proses Unit	2023-07-01 16:04:16

🔄 ⓘ 127.0.0.1:5000/pegawai

Beranda | Jenis Pembayaran | Pembayaran | Divisi | Jabatan | Pegawai | Transaksi | Konsumen | Status Transaksi | Detail Status Transaksi | Detail Pembayaran

Data Pegawai

ID	NIP	NIK/KTP	Nama	No. HP	Email	Divisi	Jabatan
1	1234567890	12345678901234567890	Agus Santoso	081234567890	agus.santoso@example.com	1	1
2	0987654321	09876543210987654321	Budi Setiawan	087654321098	budi.setiawan@example.com	2	2
3	111222333	11122233311122233311	Citra Sari	081111222333	citra.sari@example.com	2	1

127.0.0.1:5000/jabatan

Beranda | Jenis Pembayaran | Pembayaran | Divisi | Jabatan | Pegawai | Transaksi | Konsumen | Status Transaksi | Detail Status Transaksi | Detail Pembayaran

Data Jabatan

ID	Nama
1	Manager
2	Supervisor
3	Staff
4	Pramuniaga

127.0.0.1:5000/divisi

Beranda | Jenis Pembayaran | Pembayaran | Divisi | Jabatan | Pegawai | Transaksi | Konsumen | Status Transaksi | Detail Status Transaksi | Detail Pembayaran

Data Divisi

ID	Nama
1	Keuangan
2	Pemasaran
3	SDM
4	Produksi

konsumen

127.0.0.1:5000/konsumen

Beranda | Jenis Pembayaran | Pembayaran | Divisi | Jabatan | Pegawai | Transaksi | Konsumen | Status Transaksi | Detail Status Transaksi | Detail Pembayaran

Data Konsumen

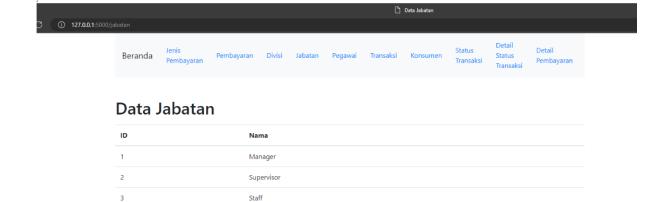| ID	NIK/KTP	Nama	Jenis Kelamin	No. HP	Email	Alamat
1	1234567890123456	Budi Santoso	Laki-Laki	081234567890	budisantoso@example.com	Jl. Contoh No. 123
2	9876543210987654	Anita Dewi	Perempuan	081234567891	anitadewi@example.com	Jl. Contoh No. 456
3	4567890123456789	Rudi Setiawan	Laki-Laki	081234567892	rudisetiawan@example.com	Jl. Contoh No. 789

Struktur data

detail_status_transaksi
- status_transaksi — mediumint(8) unsigned
- tanggal — date
- status_akhir — enum('Proses Dokumen','Proses Unit','Proses SPK','Pembayaran','Pengiriman')
- tanggal_update — timestamp NOT NULL

status_transaksi
- id — mediumint(8) unsigned NOT NULL
- transaksi — mediumint(8) unsigned NOT NULL
- pembayaran — mediumint(8) unsigned

detail_pembayaran
- pembayaran — mediumint(8) unsigned NOT NULL
- transaksi — mediumint(8) unsigned NOT NULL
- qty — smallint(5) unsigned NOT NULL
- jumlah — int(10) unsigned NOT NULL
- total — bigint(20) unsigned NOT NULL

dokumen_transaksi
- transaksi — mediumint(8) unsigned
- spk — blob

pembayaran
- id — mediumint(8) unsigned NOT NULL
- tanggal — date
- nomor_faktur — varchar(30)
- jenis_pembayaran — smallint(5) unsigned
- pegawai — smallint(5) unsigned
- keterangan — varchar(255)

transaksi
- id — mediumint(8) unsigned NOT NULL
- tanggal — date
- nomor_faktur — varchar(30)
- konsumen — smallint(5) unsigned
- mobil — smallint(5) unsigned
- pegawai — smallint(5) unsigned
- harga — int(10) unsigned

jenis_pembayaran
- id — smallint(5) unsigned NOT NULL
- nama — varchar(30)

pegawai
- id — smallint(5) unsigned NOT NULL
- nip — varchar(15)
- nik_ktp — varchar(20)
- nama — varchar(50)
- no_hp — varchar(13)
- email — varchar(25)
- divisi — smallint(5) unsigned
- jabatan — smallint(5) unsigned

mobil
- id — smallint(5) unsigned NOT NULL
- nama — varchar(20)
- tahun — smallint(5) unsigned
- harga — int(10) unsigned
- merk — varchar(10)
- jenis — enum('MPV','SUV','Sedan')
- nopol — varchar(10)

divisi
- id — smallint(5) unsigned NOT NULL
- nama — varchar(30)

jabatan
- id — smallint(5) unsigned NOT NULL
- nama — varchar(30)

dokumen_konsumen
- konsumen — smallint(5) unsigned
- foto_ktp — blob
- foto_kk — blob
- foto_selfie — blob
- foto_bpjs — blob
- foto_keterangan_kerja — blob
- foto_slip_gaji — blob

konsumen
- id — smallint(5) unsigned NOT NULL
- nik_ktp — varchar(20)
- nama — varchar(50)
- jenis_kelamin — enum('Laki-Laki','Perempuan')
- no_hp — varchar(13)
- email — varchar(50)
- alamat — varchar(100)