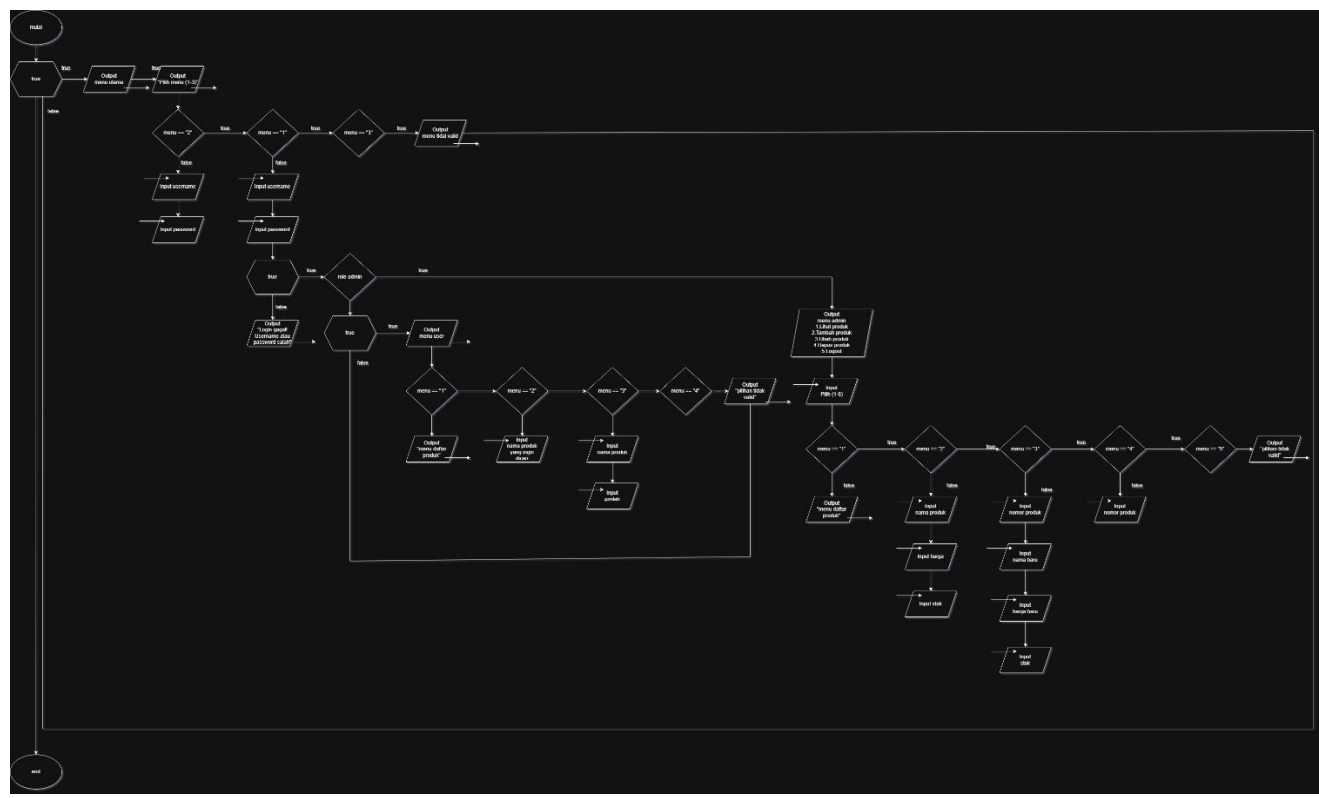


LAPORAN PRAKTIKUM
POSTTEST 7
ALGORITMA PEMROGRAMAN DASAR



Disusun oleh:
REGA WAHYU FIRENZA (2509106085)
Informatika (B2'25)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025



Gambar 1.1 flowchart

2.deskripsi Singkat Program

FUNGSI PROGRAM:

Program ini berfungsi sebagai sistem toko aksesoris rumah yang bisa digunakan oleh dua jenis pengguna:

1. Admin, untuk mengelola data produk (menambah, mengubah, menghapus, melihat daftar produk).
2. User, untuk melihat produk, mencari produk, dan menghitung total harga pembelian.

Program juga menyediakan fitur login dan registrasi, serta sudah menggunakan:

1. Fungsi dan prosedur untuk memisahkan setiap tugas (seperti login, daftar, tampilkan produk, dll).
2. Error handling untuk mencegah input salah.
3. Fungsi rekursif agar menu utama bisa muncul lagi jika terjadi kesalahan input.

3.Source code

A.BAGIAN IMPORT & VARIABEL GLOBAL

```
import os

produk = [
    {"nama": "Sofa", "harga": 75000, "stok": 10},
    {"nama": "Lampu Tidur", "harga": 125000, "stok": 5},
    {"nama": "Gantungan Kunci", "harga": 15000, "stok": 20}
]

pengguna = [
    {"username": "rega", "password": "123", "role": "admin"}
]

login_user = None
```

import os digunakan untuk membersihkan layar terminal dengan os.system().
produk list berisi data produk (nama, harga, stok).
pengguna list berisi data pengguna (admin & user).
login_user variabel global untuk menyimpan siapa yang sedang login.

B.FUNGSI CLEAR()

```
def clear():
    if os.name == "nt":
        os.system("cls")
    else:
        os.system("clear")
```

Membersihkan layar terminal agar tampilan lebih rapi.

C.FUNGSI HITUNG_TOTAL

```
def hitung_total(harga, jumlah):
    return harga * jumlah
```

Menghitung total harga pembelian (harga x jumlah).

D.FUNGSI CARI_PRODUK

```
def cari_produk(nama_produk):  
    for p in produk:  
        if p["nama"].lower() == nama_produk.lower():  
            return p  
    return None
```

Mencari produk berdasarkan nama di daftar produk.

Mengembalikan data produk kalau ditemukan.

F.FUNGSI REKURSIF MENU_UTAMA

```
def menu_utama():  
    clear()  
    print("""  
=====   
|          TOKO AKSESORIS RUMAH          |   
=====   
| 1. Login                               |   
| 2. Register                            |   
| 3. Keluar                              |   
=====   
""")  
    try:  
        menu = input("Pilih menu (1-3): ")  
        if menu == "1":  
            login()  
        elif menu == "2":  
            registrasi()  
        elif menu == "3":  
            print("Terima kasih telah menggunakan program ini.")  
            exit()  
        else:  
            raise ValueError("Menu tidak valid!")  
    except ValueError as e:  
        print("Error:", e)  
        input("Tekan Enter...")  
        menu_utama()  
    finally:  
        print("Kembali ke menu utama...")
```

Menampilkan menu utama (Login, Register, Keluar).

Kalau input salah fungsi ini memanggil dirinya sendiri

G.PROSEDUR REGISTRASI()

```
def registrasi():
    global pengguna
    clear()
    print("=== REGISTER PENGGUNA BARU ===")
    uname = input("Masukkan username baru: ").strip()
    pw = input("Masukkan password (minimal 3 angka): ").strip()
    try:
        if not uname or not pw:
            raise ValueError("Username dan password tidak boleh kosong!")
        if len(pw) < 3:
            raise ValueError("Password minimal 3 karakter!")
        if not pw.isdigit():
            raise ValueError("Password hanya boleh angka!")
    except ValueError as e:
        print(e)
        input("Tekan Enter...")
        return
    for p in pengguna:
        if p["username"] == uname:
            print("Username sudah terdaftar!")
            input("Tekan Enter...")
            return
    pengguna.append({"username": uname, "password": pw, "role": "user"})
    print("Registrasi berhasil!")
    input("Tekan Enter untuk kembali...")
    menu_utama()
```

Mendaftarkan akun baru (username & password angka minimal 3 digit).

Kalau data salah tampil pesan error.

H.PROSEDUR LOGIN()

```
def login():
    global login_user
    clear()
    print("=== LOGIN ===")
    uname = input("Username: ")
    pw = input("Password: ")
    try:
        user = next(p for p in pengguna if p["username"] == uname and
p["password"] == pw)
        login_user = user
        print(f"Selamat datang, {uname}!")
        input("Tekan Enter...")
        if user["role"] == "admin":
            menu_admin()
        else:
            menu_user()
```

```
except StopIteration:
    print("Login gagal! Username atau password salah.")
    input("Tekan Enter...")
    menu_utama()
```

Memeriksa username & password.

Jika benar masuk ke menu sesuai role (admin/user).

I.FUNGSI TAMPILAN_PRODUK()

```
def tampilkan_produk():
    clear()
    print("===== DAFTAR PRODUK =====")
    if not produk:
        print("Belum ada produk.")
    else:
        print("No | Nama Produk      | Harga      | Stok")
        print("-----")
        for i, p in enumerate(produk, start=1):
            print(f"{i}. {p['nama']:17} Rp{p['harga']:8}   {p['stok']}")
        input("\nTekan Enter...")
```

Menampilkan daftar semua produk beserta harga & stok.

J.FUNGSI MENU_ADMIN()

```
def menu_admin():
    while True:
        clear()
        print("""
=====
|      MENU ADMIN      |
=====
| 1. Lihat Produk      |
| 2. Tambah Produk     |
| 3. Ubah Produk       |
| 4. Hapus Produk      |
| 5. Logout            |
=====
""")
        pilih = input("Pilih menu (1-5): ")
        if pilih == "1":
            tampilkan_produk()
        elif pilih == "2":
            tambah_produk()
        elif pilih == "3":
            ubah_produk()
```

```

elif pilih == "4":
    hapus_produk()
elif pilih == "5":
    menu_utama()
    break
else:
    print("Pilihan tidak valid.")
    input("Tekan Enter...")

```

Menu untuk admin: bisa lihat, tambah, ubah, hapus produk, atau logout.

K.FUNGSI MENU_USER()

```

def menu_user():
    while True:
        clear()
        print("""
=====
|      MENU USER      |
=====
| 1. Lihat Produk     |
| 2. Cari Produk      |
| 3. Hitung Total Harga |
| 4. Logout           |
=====
""")

        pilih = input("Pilih menu (1-4): ")
        if pilih == "1":
            tampilkan_produk()
        elif pilih == "2":
            nama = input("Masukkan nama produk yang ingin dicari: ")
            hasil = cari_produk(nama)
            if hasil:
                print(f"Produk ditemukan: {hasil['nama']} - Harga Rp{hasil['harga']} - Stok {hasil['stok']}")
            else:
                print("Produk tidak ditemukan.")
                input("Tekan Enter...")
        elif pilih == "3":
            try:
                nama = input("Masukkan nama produk: ")
                jumlah = int(input("Masukkan jumlah: "))
                hasil = cari_produk(nama)
                if hasil:
                    total = hitung_total(hasil['harga'], jumlah)
                    print(f"Total harga: Rp{total}")
                else:
                    raise KeyError("Produk tidak ditemukan.")
            except:
                print("Produk tidak ditemukan.")

```

```

        except ValueError:
            print("Jumlah harus berupa angka.")
        except KeyError as e:
            print(e)
        finally:
            input("Tekan Enter...")
    elif pilih == "4":
        menu_utama()
        break
    else:
        print("Pilihan tidak valid.")
        input("Tekan Enter...")

```

Menu untuk user biasa: bisa lihat produk, cari produk, dan hitung total harga.

L.FUNGSI CRUD PRODUK

1.TAMBAH PRODUK

```

def tambah_produk():
    try:
        nama = input("Nama produk: ")
        harga = int(input("Harga: "))
        stok = int(input("Stok: "))
        produk.append({"nama": nama, "harga": harga, "stok": stok})
        print("Produk berhasil ditambahkan!")
    except ValueError:
        print("Input harus berupa angka untuk harga/stok.")
    finally:
        input("Tekan Enter...")

```

Menambah produk baru

2.UBAH PRODUK

```

def ubah_produk():
    tampilkan_produk()
    try:
        idx = int(input("Masukkan nomor produk yang ingin diubah: ")) - 1
        if idx < 0 or idx >= len(produk):
            raise IndexError("Nomor produk tidak valid.")
        nama = input("Nama baru: ")
        harga = input("Harga baru: ")
        stok = input("Stok baru: ")
        if nama:
            produk[idx]["nama"] = nama
        if harga.isdigit():

```



```

        produk[idx]["harga"] = int(harga)
    if stok.isdigit():
        produk[idx]["stok"] = int(stok)
    print("Produk berhasil diubah!")
except (ValueError, IndexError) as e:
    print("Error:", e)
finally:
    input("Tekan Enter...")

```

Mengedit produk berdasarkan nomor

3.HAPUS PRODUK

```

def hapus_produk():
    tampilkan_produk()
    try:
        idx = int(input("Masukkan nomor produk yang ingin dihapus: ")) - 1
        if idx < 0 or idx >= len(produk):
            raise IndexError("Nomor tidak valid.")
        del produk[idx]
        print("Produk berhasil dihapus!")
    except (ValueError, IndexError) as e:
        print(e)
    finally:
        input("Tekan Enter...")

```

Menghapus produk dari daftar

3.HAPUS PRODUK

```

menu_utama()

```

4.Hasil output

```
=====
|          TOKO AKSESORIS RUMAH          |
=====
| 1. Login                               |
| 2. Register                           |
| 3. Keluar                             |
=====

Pilih menu (1-3): █
```

Gambar 4.1 output menu utama

```
=== REGISTER PENGGUNA BARU ===
Masukkan username baru: bagas
Masukkan password (minimal 3 angka): 444
Registrasi berhasil!
Tekan Enter untuk kembali...█
```

Gambar 4.2 output registrasi berhasil

```
=== REGISTER PENGGUNA BARU ===
Masukkan username baru: bagas
Masukkan password (minimal 3 angka): 12
Password minimal 3 karakter!
Tekan Enter...█
```

Gambar 4.3 output registrasi tidak sesuai perintah

```
=== LOGIN ===
Username: rega
Password: 123
Selamat datang, rega!
Tekan Enter...█
```

Gambar 4.4 output login berhasil

```
=== LOGIN ===  
Username: rega  
Password: 222  
Login gagal! Username atau password salah.  
Tekan Enter...
```

Gambar 4.5 output login gagal

```
=====
```

MENU ADMIN
1. Lihat Produk
2. Tambah Produk
3. Ubah Produk
4. Hapus Produk
5. Logout

```
=====
```

Pilih menu (1-5):

Gambar 4.6 output menu admin

```
===== DAFTAR PRODUK =====
```

No	Nama Produk	Harga	Stok
1.	Sofa	Rp 75000	10
2.	Lampu Tidur	Rp 125000	5
3.	Gantungan Kunci	Rp 15000	20

```
Tekan Enter...
```

Gambar 4.7 output lihat produk

```
Pilih menu (1-5): 2
Nama produk: rak dinding
Harga: 80000
Stok: 10
Produk berhasil ditambahkan!
Tekan Enter...
```

Gambar 4.8 output tambah produk

```
===== DAFTAR PRODUK =====
No | Nama Produk      | Harga      | Stok
-----
1. Sofa            Rp 75000    10
2. Lampu Tidur     Rp 125000   5
3. Gantungan Kunci Rp 15000    20
4. rak dinding     Rp 80000    10
Tekan Enter...
```

Gambar 4.9 output setelah berhasil ditambahkan

```
-----
1. Sofa            Rp 75000    10
2. Lampu Tidur     Rp 125000   5
3. Gantungan Kunci Rp 15000    20
4. rak dinding     Rp 80000    10

Tekan Enter...
Masukkan nomor produk yang ingin diubah: 1
Nama baru: sofa panjang
Harga baru: 90000
Stok baru: 8
Produk berhasil diubah!
Tekan Enter...
```

Gambar 4.10 output ubah produk

```
===== DAFTAR PRODUK =====
```

No	Nama Produk	Harga	Stok
1.	sofa panjang	Rp 90000	8
2.	Lampu Tidur	Rp 125000	5
3.	Gantungan Kunci	Rp 15000	20
4.	rak dinding	Rp 80000	10

Tekan Enter...

Gambar 4.11 output setelah berhasil diubah

```
Tekan Enter...
Masukkan nomor produk yang ingin diubah: 5
Error: Nomor produk tidak valid.
Tekan Enter...
```

Gambar 4.12 output jika salah nomor

```
===== DAFTAR PRODUK =====
```

No	Nama Produk	Harga	Stok
1.	sofa panjang	Rp 90000	8
2.	Lampu Tidur	Rp 125000	5
3.	Gantungan Kunci	Rp 15000	20
4.	rak dinding	Rp 80000	10

```
Tekan Enter...
Masukkan nomor produk yang ingin dihapus: 2
Produk berhasil dihapus!
Tekan Enter...
```

Gambar 4.13 output hapus produk

```
Tekan Enter...  
Masukkan nomor produk yang ingin dihapus: 5  
Nomor tidak valid.  
Tekan Enter...█
```

Gambar 4.14 output jika salah input

```
=====
|          TOKO AKSESORIS RUMAH          |
=====
| 1. Login                               |
| 2. Register                           |
| 3. Keluar                             |
=====
Pilih menu (1-3): █
```

Gambar 4.15 output logout program kembali ke menu utama

```
=====
|          MENU USER                    |
=====
| 1. Lihat Produk                       |
| 2. Cari Produk                       |
| 3. Hitung Total Harga                 |
| 4. Logout                           |
=====
Pilih menu (1-4): █
```

Gambar 4.16 output menu user

```
===== DAFTAR PRODUK =====
```

No	Nama Produk	Harga	Stok
1.	sofa panjang	Rp 90000	8
2.	Gantungan Kunci	Rp 15000	20
3.	rak dinding	Rp 80000	10

Tekan Enter...

Gambar 4.16 output lihat produk

```
Pilih menu (1-4): 2
Masukkan nama produk yang ingin dicari: rak dinding
Produk ditemukan: rak dinding - Harga Rp80000 - Stok 10
Tekan Enter...
```

Gambar 4.17 output cari produk

```
Pilih menu (1-4): 2
Masukkan nama produk yang ingin dicari: 5
Produk tidak ditemukan.
Tekan Enter...
```

Gambar 4.18 output salah input

```
Pilih menu (1-4): 3
Masukkan nama produk: rak dinding
Masukkan jumlah: dua
Jumlah harus berupa angka.
Tekan Enter...
```

Gambar 4.19 output salah input jumlah

```
Pilih menu (1-4): 3
Masukkan nama produk: vas
Masukkan jumlah: 2
'Produk tidak ditemukan.'
Tekan Enter...
```

Gambar 4.20 output produk tidak ada

```
=====
|          TOKO AKSESORIS RUMAH          |
=====
| 1. Login                               |
| 2. Register                           |
| 3. Keluar                             |
=====

Pilih menu (1-3):
```

Gambar 4.21 output logout kembali ke menu utama

```
Pilih menu (1-3): 3
Terima kasih telah menggunakan program ini.
```

Gambar 4.22 output pilih keluar di menu utama

5.Langkah-langkah GIT

5.2 GIT Add

```
PS C:\Users\ASUS\OneDrive\praktikum-apd> git add .
```

Gambar 5.1 Git add

Menyimpan dan mengunggah file program ke GitHub

5.3 GIT Commit

```
PS C:\Users\ASUS\OneDrive\praktikum-apd> git commit -m "posttest 7"  
[main acc5f8a] posttest 7  
1 file changed, 241 insertions(+)  
create mode 100644 post-test/post-test-apd-7/2509106085-RegaWahyuFirenza-PT-7.py  
PS C:\Users\ASUS\OneDrive\praktikum-apd>
```

Gambar 5.2 Git commit

Menyimpan versi terbaru dari kode di repo lokal dengan pesan penjelasan perubahan

5.4 GIT Push

```
PS C:\Users\ASUS\OneDrive\praktikum-apd> git push  
Everything up-to-date
```

Gambar 5.3 Git push

Mengirim commit (yang sudah dibuat di repo lokal) ke repo di server GitHub