

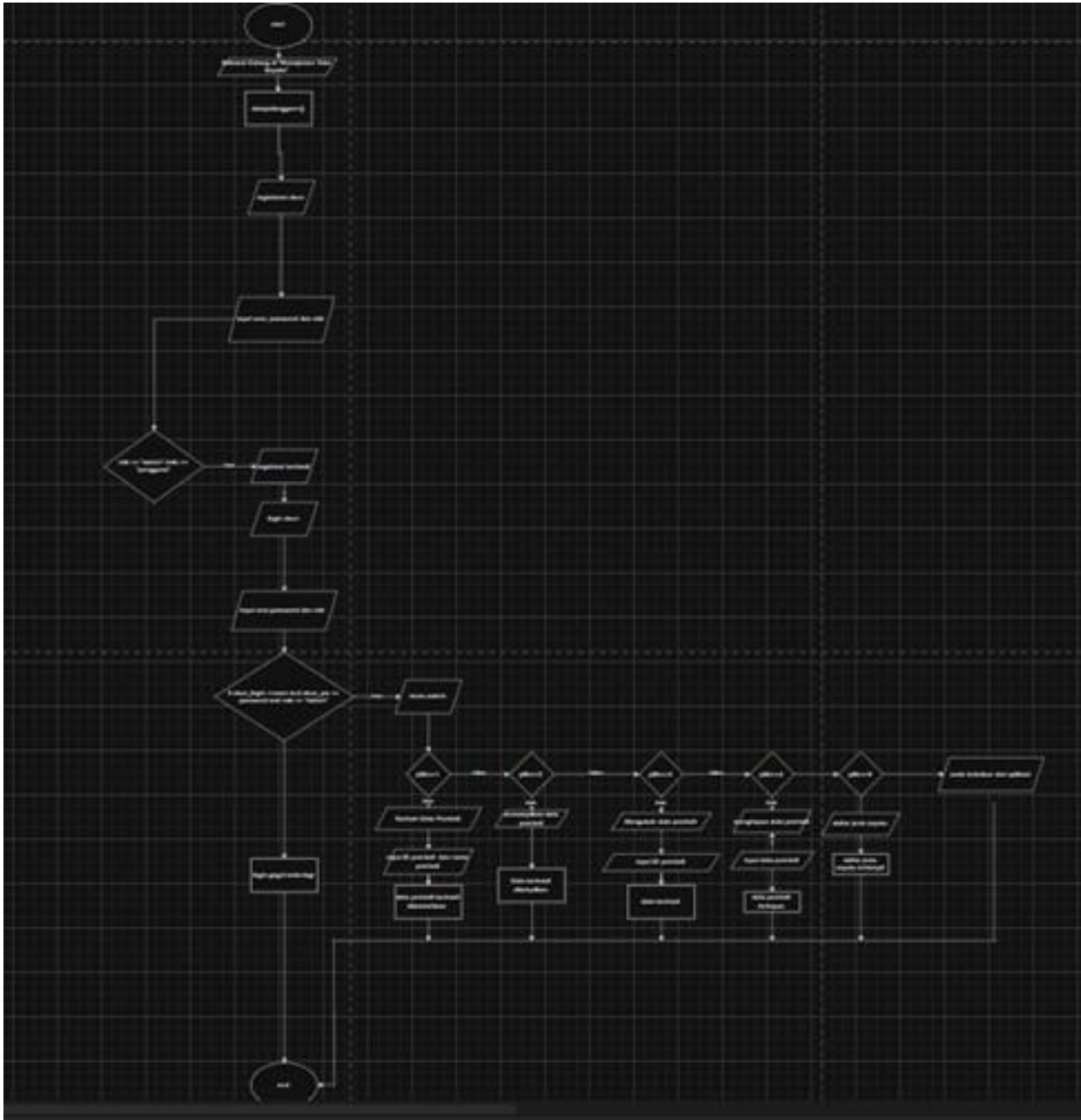
**LAPORAN PRAKTIKUM**  
**POSTTEST 6**  
**ALGORITMA PEMROGRAMAN DASAR**



**Disusun oleh:**  
**Andhika Bima Saputra(2509106116)**  
**Kelas (C2 '25)**

**PROGRAM STUDI INFORMATIKA**  
**UNIVERSITAS MULAWARMAN SAMARINDA**

## 1.Flowchart



Gambar 1.1 Flowchart Alur Program

## **2. Deskripsi Singkat program**

Program ini adalah aplikasi berbasis terminal untuk mengelola data pelanggan dan transaksi di sebuah toko sepatu

### 3. Source Code

```
import os

akun = {
    "andhikaaa": "123456", # admin
    "andhika": "12345"     # guest
}

akun_tambahan = {}

paket_sepatu = {
    "1": {"nama": "Sneakers Basic", "stok": 1, "harga": 250000},
    "2": {"nama": "Running Pro", "stok": 1, "harga": 400000},
    "3": {"nama": "Casual Street", "stok": 1, "harga": 350000},
    "4": {"nama": "Formal Leather", "stok": 1, "harga": 500000},
    "5": {"nama": "Anak Sekolah", "stok": 1, "harga": 200000}
}

data_pembeli = {}
```

Gambar 2.1 Source Code Data Dasar

```
while True:
    os.system('cls')
    print("=== SISTEM MANAJEMEN TOKO SEPATU ===")
    print("1. Login")
    print("2. Register Akun Baru")
    print("3. Keluar")

    pilih_awal = input("Pilih menu: ")
```

Gambar 2.2 Source Code Menu awal

```

if pilih_awal == "2":
    os.system('cls')
    print("=== REGISTER AKUN BARU ===")
    new_user = input("Masukkan username baru: ")
    new_pass = input("Masukkan password baru: ")

    if new_user in akun or new_user in akun_tambahan:
        print("Username sudah terdaftar! Silakan coba lagi.")
    else:
        akun_tambahan[new_user] = new_pass
        print("Akun berhasil dibuat! Silakan login menggunakan akun baru.")

    input("\nTekan Enter untuk kembali ke menu awal...")

```

**Gambar 2.3 Source Code Fitur Register akun baru**

```

elif pilih_awal == "1":
    os.system('cls')
    print("=== LOGIN SISTEM TOKO SEPATU ===")
    username = input("Masukkan username: ")
    password = input("Masukkan password: ")

    login = False
    is_admin = False

    if username in akun and akun[username] == password:
        login = True
        if username == "andhikaaa":
            is_admin = True
    elif username in akun_tambahan and akun_tambahan[username] == password:
        login = True

    if not login:
        print("Login gagal! Username atau password salah.")
        input("\nTekan Enter untuk kembali...")
        continue

    print("Login berhasil!\n")

```

**Gambar 2.4 Source Code Fitur Login**

```

while login:
    os.system('cls')
    print("=== MENU UTAMA ===")
    print("1. Tambah Data Pembeli (Create)")
    print("2. Lihat Data Pembeli (Read)")
    if is_admin:
        print("3. Ubah Data Pembeli (Update)")
        print("4. Hapus Data Pembeli (Delete)")
    print("5. Lihat Daftar Sepatu")
    print("6. Logout")

    menu = input("Pilih menu: ")

```

**Gambar 2.5 Source Code Menu utama setelah Login**

```

if menu == "1":
    os.system('cls')
    print("=== TAMBAH DATA PEMBELI ===")
    id_p = input("Masukkan ID pembeli: ")

    if id_p in data_pembeli:
        print("ID pembeli sudah ada!")
    else:
        nama = input("Masukkan nama pembeli: ")
        print("\nPilih jenis sepatu yang dibeli:")
        for k, v in paket_sepatu.items():
            print(f"{k}. {v['nama']} - Rp{v['harga']:,}")

        pilih_sepatu = input("Masukkan nomor sepatu: ")
        if pilih_sepatu in paket_sepatu:
            sepatu = paket_sepatu[pilih_sepatu]['nama']
            harga = paket_sepatu[pilih_sepatu]['harga']
        else:
            sepatu = "Custom"
            harga = 0

        data_pembeli[id_p] = {
            "nama": nama,
            "sepatu": sepatu,
            "status": "Belum Bayar",
            "harga": harga
        }
    print("\nData pembeli berhasil ditambahkan!")

```

**Gambar 2.5 Source Code Fitur Create**

```

elif menu == "2":
    os.system('cls')
    print("=== DATA PEMBELI TOKO SEPATU ===")
    if not data_pembeli:
        print("Belum ada data pembeli.")
    else:
        for id_p, info in data_pembeli.items():
            print(f"ID: {id_p}")
            print(f>Nama: {info['nama']}")
            print(f>Sepatu: {info['sepatu']}")
            print(f>Status Pembayaran: {info['status']}")
            print(f>Harga: Rp{info['harga']:,} ")
            print("-" * 35)
        print(f>Total pembeli terdaftar: {len(data_pembeli)}")
    input("\nTekan Enter untuk kembali...")

```

**Gambar 2.6 Source Code Fitur Read**

```

elif menu == "3" and is_admin:
    os.system('cls')
    print("=== UBAH DATA PEMBELI ===")
    id_cari = input("Masukkan ID pembeli: ")

    if id_cari in data_pembeli:
        print("1. Ubah Nama")
        print("2. Ubah Jenis Sepatu")
        print("3. Ubah Status Pembayaran")
        pilihan = input("Pilih data yang ingin diubah: ")

        if pilihan == "1":
            data_pembeli[id_cari]['nama'] = input("Masukkan nama baru: ")
        elif pilihan == "2":
            for k, v in paket_sepatu.items():
                print(f">{k}. {v['nama']} - Rp{v['harga']:,} ")
            pilih_sepatu = input("Masukkan nomor sepatu: ")
            if pilih_sepatu in paket_sepatu:
                data_pembeli[id_cari]['sepatu'] = paket_sepatu[pilih_sepatu]['nama']
                data_pembeli[id_cari]['harga'] = paket_sepatu[pilih_sepatu]['harga']
        elif pilihan == "3":
            data_pembeli[id_cari]['status'] = input("Masukkan status baru (Belum Bayar/Lunas): ")
        else:
            print("Pilihan tidak valid.")
            print("Data berhasil diubah!")
    else:
        print("Data pembeli tidak ditemukan.")

    input("\nTekan Enter untuk kembali...")

```

**Gambar 2.7 Source Code Fitur Update**

```

elif menu == "4" and is_admin:
    os.system('cls')
    print("=== HAPUS DATA PEMBELI ===")
    id_hapus = input("Masukkan ID pembeli: ")

    if id_hapus in data_pembeli:
        del data_pembeli[id_hapus]
        print("Data pembeli berhasil dihapus!")
    else:
        print("Data pembeli tidak ditemukan.")
    input("\nTekan Enter untuk kembali...")

```

**Gambar 2.7 Source Code Fitur Delete**

```

elif menu == "5":
    os.system('cls')
    print("=== DAFTAR JENIS SEPATU ===")
    for v in paket_sepatu.values():
        print(f"{v['nama']} | Harga: Rp{v['harga']:,}")
    input("\nTekan Enter untuk kembali...")

```

**Gambar 2.8 Source Code Fitur lihat Daftar Paket**

```

# FITUR LOGOUT

elif menu == "6":
    print("Anda telah logout. Terima kasih!")
    login = False
    input("Tekan Enter untuk kembali ke menu awal...")
    break

```

**Gambar 2.9 Source Code Fitur Logout**

```

else:
    print("Menu tidak valid!")
    input("Tekan Enter untuk kembali...")

```

**Gambar 2.10 Source Code Handle Menu Tidak valid**



```
elif pilih_awal == "3":  
    print("\nTerima kasih telah menggunakan sistem toko sepatu!")  
    break
```

**Gambar 2.11 Source Code Keluar dari program**

```
else:  
    print("Pilihan tidak valid!")  
    input("Tekan Enter untuk kembali...")
```

**Gambar 2.12 Source Code Handle menu tidak valid (Menu awal)**

#### 4. Hasil Output

```
=== SISTEM MANAJEMEN TOKO SEPATU ===
1. Login
2. Register Akun Baru
3. Keluar
Pilih menu: █
```

Gambar 4.1 Tampilan menu awal

```
=== LOGIN SISTEM TOKO SEPATU ===
Masukkan username: 123456
Masukkan password: 12345█
```

Gambar 4.2 Tampilan input username dan password

```
=== MENU UTAMA TOKO SEPATU ===
1. Tambah Data Pembeli (Create)
2. Lihat Data Pembeli (Read)
5. Lihat Daftar Sepatu
6. Logout
Pilih menu: █
```

Gambar 4.3 Tampilan login admin

```
=== DAFTAR JENIS SEPATU ===
Sneakers Basic | Harga: Rp250,000
Running Pro | Harga: Rp400,000
Casual Street | Harga: Rp350,000
Formal Leather | Harga: Rp500,000
Anak Sekolah | Harga: Rp200,000
```

Gambar 4.4 Tampilan ingin menambah pelanggan

```
=== UBAH DATA PEMBELI ===
Masukkan ID pembeli yang ingin diubah: 2

Data ditemukan!
1. Ubah nama
2. Ubah jenis sepatu
3. Ubah status pembayaran
Pilih data yang ingin diubah: 2

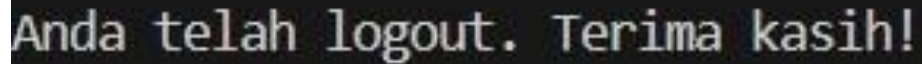
Pilih jenis sepatu baru:
1. Sneakers Basic - Rp250,000
2. Running Pro - Rp400,000
3. Casual Street - Rp350,000
4. Formal Leather - Rp500,000
5. Anak Sekolah - Rp200,000
Masukkan nomor sepatu: 4

Data berhasil diubah!
```

Gambar 4.6 Tampilan data yang mau di ubah

```
=== HAPUS DATA PEMBELI ===
Masukkan ID pembeli yang ingin dihapus: 2
Data pembeli berhasil dihapus!
```

Gambar 4.8 Tampilan data yang mau di hapus

A terminal window with a black background and white text. The text reads "Anda telah logout. Terima kasih!".

```
Anda telah logout. Terima kasih!
```

Gambar 4.11 Tampilan Keluar

## 5. Langkah-langkah GIT

### 5.1 GIT Add

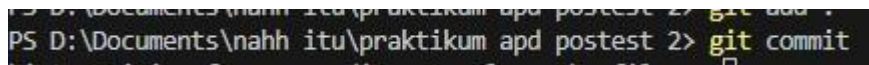
A terminal window with a black background and white text. The text shows a command prompt and a git push command.

```
PS C:\partikum-apd\pratikum-apd> git push -u origin main
```

Gambar 5.1 Langkah Git

Perintah git add . digunakan untuk menambahkan semua perubahan file yang ada di dalam folder proyek ke dalam staging area Git. Staging area adalah tempat sementara di mana perubahan file disiapkan sebelum benar-benar disimpan ke dalam riwayat repository melalui perintah git commit.

### 5.2 GIT Commit


A terminal window with a black background and white text. The text shows two git commands being entered.

```
PS D:\Documents\nahh itu\praktikum apd posttest 2> git add .  
PS D:\Documents\nahh itu\praktikum apd posttest 2> git commit
```

Gambar 5.2 Langkah Git

Commit dalam Git dapat diibaratkan seperti menyimpan catatan atau rekaman atas perubahan yang telah dilakukan pada proyek. Git commit berfungsi untuk menyimpan (merekam) snapshot atau perubahan pada kode atau file di repository Git. Saat kamu melakukan commit, Git akan menyimpan semua perubahan yang sudah kamu staging (dimasukkan ke area staging) dalam sebuah commit object

### 5.3 GIT Push

A terminal window with a black background and white text. The text shows a git push command being entered.

```
PS D:\Documents\nahh itu\praktikum apd posttest 2> git push
```

Gambar 5.3 Langkah Git

Git push adalah perintah yang digunakan untuk mengirimkan perubahan atau hasil kerja dari komputer Anda (repository lokal) ke penyimpanan Git yang ada di internet atau server (repository remote), seperti GitHub atau GitLab. Setelah Anda melakukan perubahan dan menyimpannya secara lokal, perintah ini berfungsi agar perubahan tersebut dapat tersimpan secara online dan dapat diakses oleh orang lain yang memiliki akses ke repository tersebut.