

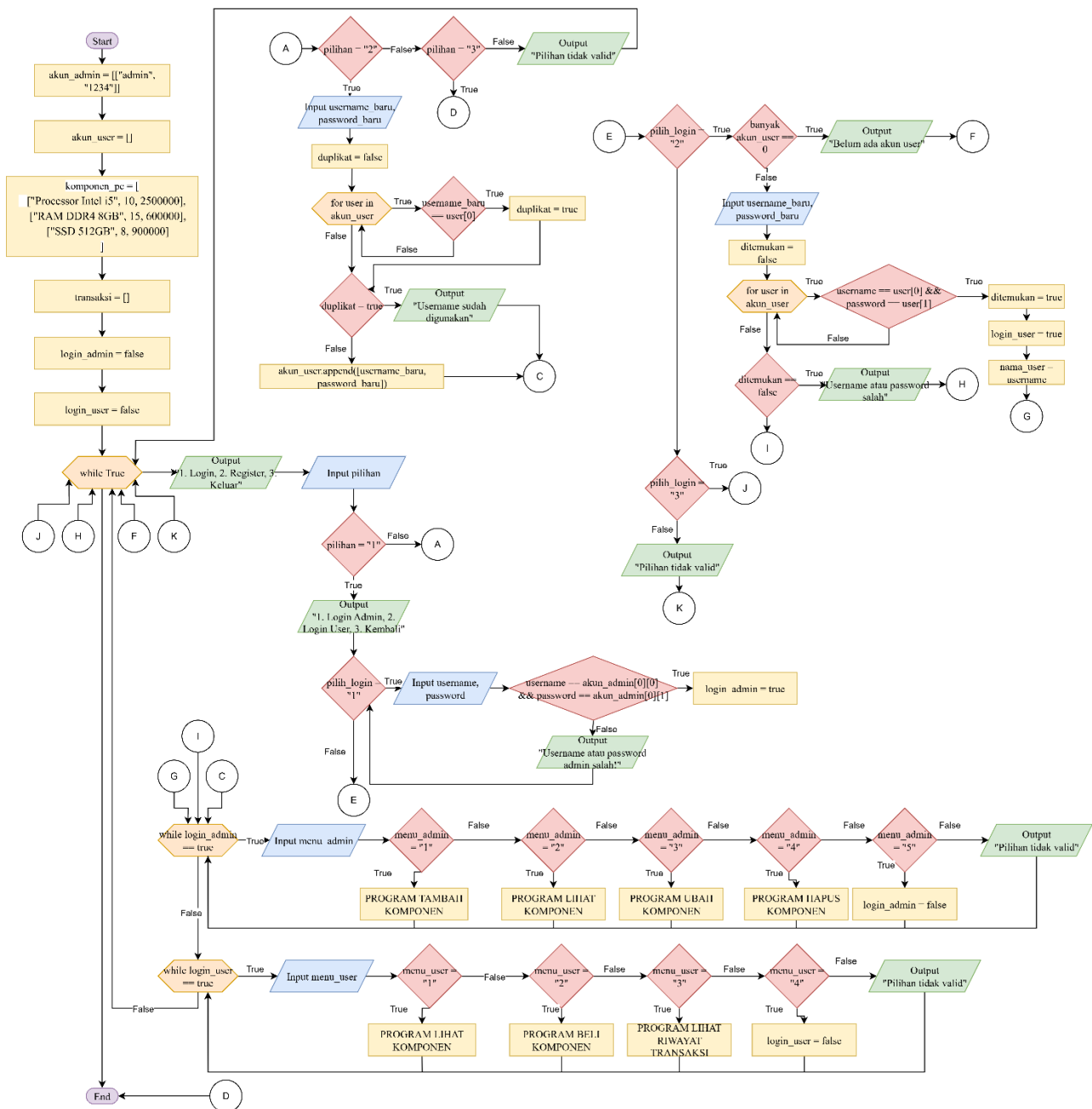
LAPORAN PRAKTIKUM
POSTTEST 5
ALGORITMA PEMROGRAMAN DASAR



Disusun oleh:
Hamam Syamil (2509106073)
Kelas (B2 '25)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

1. Flowchart



Program ini diawali dengan menampilkan menu utama yang memberikan pilihan kepada pengguna untuk melakukan login, register, atau keluar dari program. Jika pengguna memilih register, sistem akan meminta input berupa username dan password baru, lalu menyimpannya ke dalam daftar akun user. Jika memilih login, pengguna dapat masuk sebagai admin atau user dengan memasukkan username dan password yang sesuai. Setelah login berhasil, admin dapat melakukan operasi CRUD terhadap data komponen PC, seperti menambah, melihat, mengubah, dan menghapus data. Sementara itu, user dapat melihat daftar komponen dan melakukan pembelian, yang kemudian dicatat sebagai transaksi. Proses berlanjut sesuai pilihan pengguna hingga mereka memilih untuk keluar dari program. Output program berupa pesan-

pesan hasil validasi (seperti login berhasil atau gagal), tampilan daftar komponen, hasil pembelian, serta notifikasi dari setiap aksi yang dilakukan.

2. Deskripsi Singkat Program

Tujuan dan fungsi utama dari program ini adalah untuk membantu proses pengelolaan penjualan komponen PC secara efisien dan terstruktur. Program ini berfungsi sebagai sistem manajemen yang memungkinkan pengguna untuk menambahkan data komponen baru, memperbarui informasi stok dan harga, menampilkan daftar komponen serta riwayat transaksi, hingga menghapus data yang sudah tidak relevan. Dengan adanya pembagian peran antara admin dan user, program ini mempermudah admin dalam mengatur ketersediaan barang dan harga, sementara pengguna dapat melakukan pembelian dengan mudah dan transparan. Secara keseluruhan, program ini bermanfaat untuk menjaga keakuratan data penjualan, memantau stok barang, dan mendukung kelancaran transaksi dalam sistem penjualan komponen PC.

3. Source Code

```
import os

akun_admin = [["admin", "123"]]
akun_user = []
komponen_pc = [
    ["Processor Intel i5", 10, 2500000],
    ["RAM DDR4 8GB", 15, 600000],
    ["SSD 512GB", 8, 900000]]
transaksi = []

login_admin = False
login_user = False

while True:
    os.system('cls')
    print("=== SISTEM MANAJEMEN PENJUALAN KOMPONEN PC ===")
    print("1. Login")
    print("2. Register")
    print("3. Keluar")

    pilihan = input("Pilih menu: ")

    if pilihan == "1":
        os.system('cls')
```

```

print("=== LOGIN ===")
print("1. Login sebagai Admin")
print("2. Login sebagai User")
print("3. Kembali")

pilih_login = input("Pilih: ")

if pilih_login == "1":
    username = input("Masukkan username admin: ")
    password = input("Masukkan password admin: ")
    if username == akun_admin[0][0] and password == akun_admin[0][1]:
        print("Login Admin berhasil!")
        login_admin = True
    else:
        print("Username atau password admin salah!")
        input("Tekan Enter untuk kembali...")

elif pilih_login == "2":
    if len(akun_user) == 0:
        print("Belum ada akun user! Silakan register dulu.")
        input("Tekan Enter untuk kembali...")
        continue
    username = input("Masukkan username user: ")
    password = input("Masukkan password user: ")
    ditemukan = False
    for user in akun_user:
        if username == user[0] and password == user[1]:
            print("Login User berhasil!")
            login_user = True
            nama_user = username
            ditemukan = True
            break
    if not ditemukan:
        print("Username atau password salah!")
        input("Tekan Enter untuk kembali...")

elif pilih_login == "3":
    continue
else:
    print("Pilihan tidak valid!")
    input("Tekan Enter untuk kembali...")
    continue

elif pilihan == "2":
    os.system('cls')
    print("=== REGISTER USER BARU ===")
    username_baru = input("Masukkan username: ")
    password_baru = input("Masukkan password: ")

    duplikat = False
    for user in akun_user:

```

```

        if username_baru == user[0]:
            duplikat = True
            break

    if duplikat:
        print("Username sudah digunakan!")
    else:
        akun_user.append([username_baru, password_baru])
        print("Akun berhasil dibuat!")
        input("Tekan Enter untuk kembali...")

elif pilihan == "3":
    print("Keluar dari program...")
    break

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

while login_admin == True:
    os.system('cls')
    print("=== MENU ADMIN ===")
    print("1. Tambah Komponen")
    print("2. Lihat Komponen")
    print("3. Ubah Komponen")
    print("4. Hapus Komponen")
    print("5. Logout")
    menu_admin = input("Pilih: ")

    if menu_admin == "1":
        nama = input("Nama komponen: ")
        stok = input("Jumlah stok: ")
        harga = input("Harga: ")
        komponen_pc.append([nama, int(stok), int(harga)])
        print("Komponen berhasil ditambahkan!")
        input("Tekan Enter untuk lanjut...")

    elif menu_admin == "2":
        if len(komponen_pc) == 0:
            print("Belum ada data komponen.")
        else:
            print("\n=== DATA KOMPONEN ===")
            for i in range(len(komponen_pc)):
                data = komponen_pc[i]
                print(f"{i+1}. {data[0]} | Stok: {data[1]} | Harga:
Rp{data[2]}")
            input("Tekan Enter untuk lanjut...")

    elif menu_admin == "3":
        if len(komponen_pc) == 0:

```

```

        print("Belum ada data untuk diubah.")
        input("Tekan Enter...")
        continue
    print("\n=== UBAH DATA KOMPONEN ===")
    for i in range(len(komponen_pc)):
        print(f"{i+1}. {komponen_pc[i][0]}")
    pilih = int(input("Pilih nomor komponen: ")) - 1
    if 0 <= pilih < len(komponen_pc):
        komponen_pc[pilih][1] = int(input("Stok baru: "))
        komponen_pc[pilih][2] = int(input("Harga baru: "))
        print("Data berhasil diubah!")
    else:
        print("Nomor tidak valid.")
        input("Tekan Enter...")

elif menu_admin == "4":
    if len(komponen_pc) == 0:
        print("Tidak ada data untuk dihapus.")
        input("Tekan Enter...")
        continue
    print("\n=== HAPUS DATA KOMPONEN ===")
    for i in range(len(komponen_pc)):
        print(f"{i+1}. {komponen_pc[i][0]}")
    pilih = int(input("Pilih nomor komponen: ")) - 1
    if 0 <= pilih < len(komponen_pc):
        del komponen_pc[pilih]
        print("Data berhasil dihapus!")
    else:
        print("Nomor tidak valid.")
        input("Tekan Enter...")

elif menu_admin == "5":
    login_admin = False
    print("Logout berhasil!")
    input("Tekan Enter...")
else:
    print("Pilihan tidak valid!")
    input("Tekan Enter...")

while login_user == True:
    os.system('cls')
    print(f"=== MENU USER ({nama_user}) ===")
    print("1. Lihat Komponen")
    print("2. Beli Komponen")
    print("3. Lihat Riwayat Transaksi")
    print("4. Logout")

    menu_user = input("Pilih: ")

    if menu_user == "1":
        if len(komponen_pc) == 0:

```

```

        print("Belum ada komponen tersedia.")
    else:
        print("\n=== DAFTAR KOMPONEN ===")
        for i in range(len(komponen_pc)):
            data = komponen_pc[i]
            print(f"{i+1}. {data[0]} | Stok: {data[1]} | Harga:
Rp{data[2]}")
        input("Tekan Enter...")

    elif menu_user == "2":
        if len(komponen_pc) == 0:
            print("Tidak ada barang untuk dibeli.")
            input("Tekan Enter...")
            continue
        print("\n=== BELI KOMPONEN ===")
        for i in range(len(komponen_pc)):
            print(f"{i+1}. {komponen_pc[i][0]} | Stok: {komponen_pc[i][1]} |
Harga: Rp{komponen_pc[i][2]}")
        pilih = int(input("Pilih nomor komponen: ")) - 1
        if 0 <= pilih < len(komponen_pc):
            jumlah = int(input("Jumlah dibeli: "))
            if jumlah <= komponen_pc[pilih][1]:
                total = jumlah * komponen_pc[pilih][2]
                komponen_pc[pilih][1] -= jumlah
                transaksi.append([nama_user, komponen_pc[pilih][0], jumlah,
total])
            print(f"Pembelian berhasil! Total: Rp{total}")
        else:
            print("Stok tidak cukup.")
        else:
            print("Nomor tidak valid.")
            input("Tekan Enter...")

    elif menu_user == "3":
        print("\n=== RIWAYAT TRANSAKSI ===")
        kosong = True
        for t in transaksi:
            if t[0] == nama_user:
                print(f"- {t[1]} x{t[2]} = Rp{t[3]}")
                kosong = False
        if kosong:
            print("Belum ada transaksi.")
            input("Tekan Enter...")

    elif menu_user == "4":
        login_user = False
        print("Logout berhasil!")
        input("Tekan Enter...")
    else:
        print("Pilihan tidak valid!")
        input("Tekan Enter...")

```

4. Hasil Output

```
=== REGISTER USER BARU ===  
Masukkan username: Hammam Syamil  
Masukkan password: 2509106073  
Akun berhasil dibuat!  
Tekan Enter untuk kembali...
```

Gambar 4. 1 Pengguna memilih “Register” lalu berhasil membuat akun baru

```
=== REGISTER USER BARU ===  
Masukkan username: Hammam Syamil  
Masukkan password: 2509106073  
Username sudah digunakan!  
Tekan Enter untuk kembali...|
```

Gambar 4. 2 Pengguna memilih “Register” dengan username yang sudah digunakan

```
=== MENU ADMIN ===  
1. Tambah Komponen  
2. Lihat Komponen  
3. Ubah Komponen  
4. Hapus Komponen  
5. Logout  
Pilih: |
```

Gambar 4. 3 Pengguna memilih “Login” sebagai admin dan berhasil masuk

```
=== LOGIN ===  
1. Login sebagai Admin  
2. Login sebagai User  
3. Kembali  
Pilih: 1  
Masukkan username admin: admin  
Masukkan password admin: bukan123  
Username atau password admin salah!  
Tekan Enter untuk kembali...|
```

Gambar 4. 4 Pengguna memilih “Login” sebagai admin namun salah password/username

```
=== MENU USER (Hammam Syamil) ===  
1. Lihat Komponen  
2. Beli Komponen  
3. Lihat Riwayat Transaksi  
4. Logout  
Pilih: |
```

Gambar 4. 5 Pengguna memilih “Login” sebagai user dan berhasil masuk


```

=== LOGIN ===
1. Login sebagai Admin
2. Login sebagai User
3. Kembali
Pilih: 2
Belum ada akun user! Silakan register dulu.
Tekan Enter untuk kembali...|

```

Gambar 4. 6 Pengguna memilih “Login” sebagai user namun akun belum terdaftar

```

=== MENU ADMIN ===
1. Tambah Komponen
2. Lihat Komponen
3. Ubah Komponen
4. Hapus Komponen
5. Logout
Pilih: 1
Nama komponen: Processor Intel i-7
Jumlah stok: 5
Harga: 1500000
Komponen berhasil ditambahkan!
Tekan Enter untuk lanjut...|

```

Gambar 4. 7 Admin menambahkan komponen PC baru

```

=== MENU ADMIN ===
1. Tambah Komponen
2. Lihat Komponen
3. Ubah Komponen
4. Hapus Komponen
5. Logout
Pilih: 2

=== DATA KOMPONEN ===
1. Processor Intel i5 | Stok: 10 | Harga: Rp2500000
2. RAM DDR4 8GB | Stok: 15 | Harga: Rp600000
3. SSD 512GB | Stok: 8 | Harga: Rp900000
4. Processor Intel i-7 | Stok: 5 | Harga: Rp1500000
Tekan Enter untuk lanjut...|

```

Gambar 4. 8 Admin melihat daftar komponen dan transaksi

```
=== MENU ADMIN ===
1. Tambah Komponen
2. Lihat Komponen
3. Ubah Komponen
4. Hapus Komponen
5. Logout
Pilih: 3

=== UBAH DATA KOMPONEN ===
1. Processor Intel i5
2. RAM DDR4 8GB
3. SSD 512GB
4. Proccessor Intel i-7
Pilih nomor komponen: 4
Stok baru: 15
Harga baru: 1600000
Data berhasil diubah!
Tekan Enter...|
```

Gambar 4. 9 Admin memperbarui data harga atau stok

```
=== MENU ADMIN ===
1. Tambah Komponen
2. Lihat Komponen
3. Ubah Komponen
4. Hapus Komponen
5. Logout
Pilih: 4

=== HAPUS DATA KOMPONEN ===
1. Processor Intel i5
2. RAM DDR4 8GB
3. SSD 512GB
4. Proccessor Intel i-7
Pilih nomor komponen: 4
Data berhasil dihapus!
Tekan Enter...|
```

Gambar 4. 10 Admin menghapus data komponen atau transaksi

```

=== MENU USER (Hammam Syamil) ===
1. Lihat Komponen
2. Beli Komponen
3. Lihat Riwayat Transaksi
4. Logout
Pilih: 2

=== BELI KOMPONEN ===
1. Processor Intel i5 | Stok: 10 | Harga: Rp2500000
2. RAM DDR4 8GB | Stok: 15 | Harga: Rp600000
3. SSD 512GB | Stok: 8 | Harga: Rp900000
Pilih nomor komponen: 1
Jumlah dibeli: 2
Pembelian berhasil! Total: Rp5000000
Tekan Enter...|

```

Gambar 4. 11 User melakukan transaksi pembelian komponen

```

=== MENU USER (Hammam Syamil) ===
1. Lihat Komponen
2. Beli Komponen
3. Lihat Riwayat Transaksi
4. Logout
Pilih: 2

=== BELI KOMPONEN ===
1. Processor Intel i5 | Stok: 0 | Harga: Rp2500000
2. RAM DDR4 8GB | Stok: 15 | Harga: Rp600000
3. SSD 512GB | Stok: 8 | Harga: Rp900000
Pilih nomor komponen: 1
Jumlah dibeli: 1
Stok tidak cukup.
Tekan Enter...|

```

Gambar 4. 12 User mencoba membeli komponen yang stoknya habis

```

=== MENU USER (Hammam Syamil) ===
1. Lihat Komponen
2. Beli Komponen
3. Lihat Riwayat Transaksi
4. Logout
Pilih: 3

=== RIWAYAT TRANSAKSI ===
- Processor Intel i5 x2 = Rp5000000
- Processor Intel i5 x8 = Rp20000000
Tekan Enter...|

```

Gambar 4. 13 User melihat riwayat transaksi

```
=== MENU USER (Hammam Syamil) ===
1. Lihat Komponen
2. Beli Komponen
3. Lihat Riwayat Transaksi
4. Logout
Pilih: 1

=== DAFTAR KOMPONEN ===
1. Processor Intel i5 | Stok: 0 | Harga: Rp2500000
2. RAM DDR4 8GB | Stok: 15 | Harga: Rp600000
3. SSD 512GB | Stok: 8 | Harga: Rp900000
Tekan Enter...|
```

Gambar 4. 14 User melihat daftar komponen sebelum membeli

```
=== SISTEM MANAJEMEN PENJUALAN KOMPONEN PC ===
1. Login
2. Register
3. Keluar
Pilih menu: 4
Pilihan tidak valid!
Tekan Enter untuk kembali...|
```

Gambar 4. 15 Pengguna memilih opsi yang tidak valid di menu utama

```
=== SISTEM MANAJEMEN PENJUALAN KOMPONEN PC ===
1. Login
2. Register
3. Keluar
Pilih menu: 3
Keluar dari program...
```

Gambar 4. 16 Pengguna memilih opsi “Keluar”

5. Langkah-langkah GIT

5.1 GIT Add

Perintah “git add” digunakan untuk menambahkan file ke staging area sebelum dilakukan commit. Pada gambar yang digunakan “git add .” agar semua perubahan di folder saat ini langsung ditambahkan sekaligus, tanpa harus memilih file satu per satu.

```
PS C:\Users\arifi\Documents\praktikum-apd\post-test\post-test-apd-5> git add .
```

5.2 GIT Commit

Perintah “git commit” digunakan untuk menyimpan perubahan dari staging area ke riwayat repository lokal. Pada gambar digunakan opsi -m, dipakai untuk memberi pesan singkat.

```
PS C:\Users\arifi\Documents\praktikum-apd\post-test\post-test-apd-5> git commit -m "POSTTEST 5"
[main 1c24b87] POSTTEST 5
3 files changed, 1893 insertions(+)
create mode 100644 post-test/post-test-apd-5/.$2509106073_HammamSyamil_PT-5.drawio.bkp
create mode 100644 post-test/post-test-apd-5/2509106073-HammamSyamil-PT5.py
create mode 100644 post-test/post-test-apd-5/2509106073_HammamSyamil_PT-5.drawio
```

5.3 GIT Push

Perintah “git push” digunakan untuk mengirim commit dari repository lokal ke repository remote.

```
PS C:\Users\arifi\Documents\praktikum-apd\post-test\post-test-apd-5> git push
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 4 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (7/7), 13.75 KiB | 2.29 MiB/s, done.
Total 7 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 1 local object.
To https://github.com/Aeloxyll/praktikum-apd.git
74c0350..1c24b87 main -> main
```