

LAPORAN PRAKTIKUM
POSTTEST (4)
ALGORITMA PEMROGRAMAN DASAR



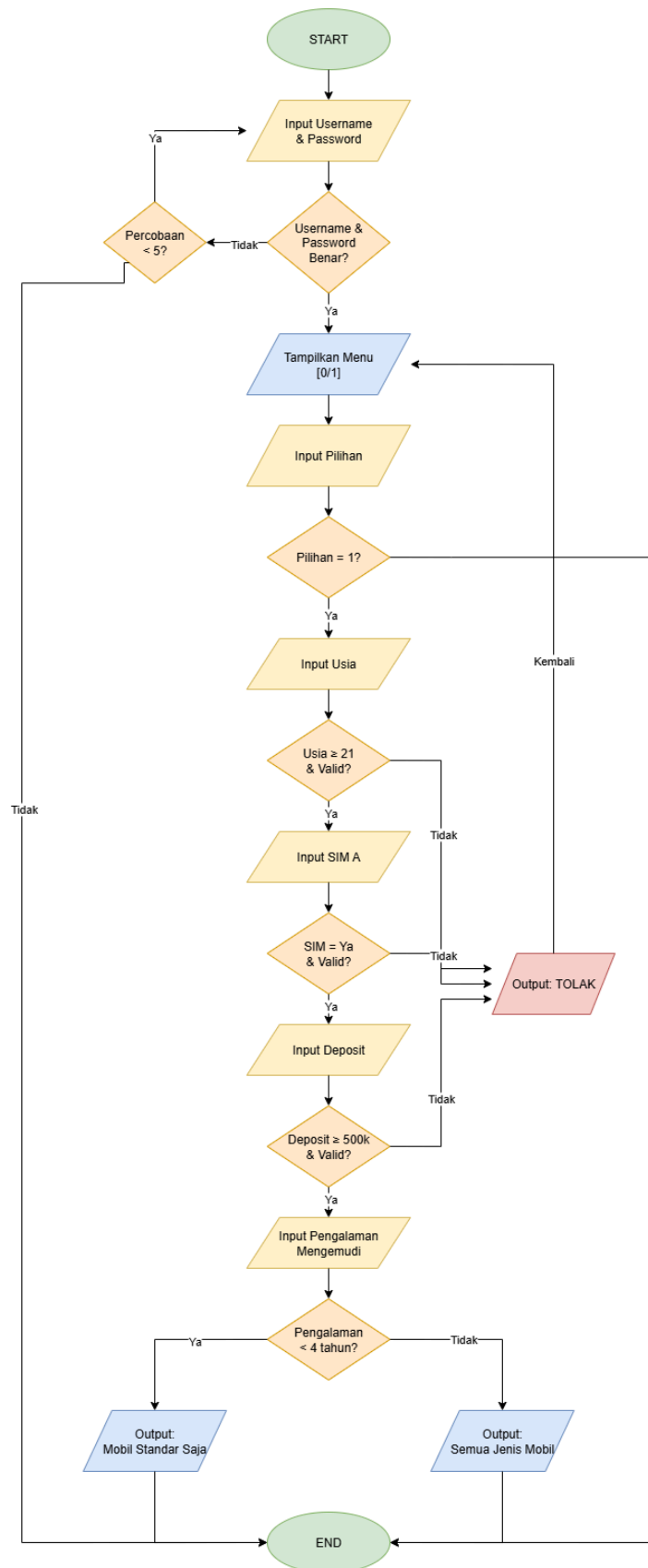
Disusun oleh:

Athasyahri Syawal Fahrezy (2509106045)

Kelas (A2'25)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

1. Flowchart



Gambar 1.1 Flowchart

2. Deskripsi Singkat Program

Program Sistem Rental Mobil ini membantu petugas rental mengecek kelayakan calon penyewa secara cepat dan adil. Sistem diawali dengan login yang memberikan maksimal 5 kali kesempatan memasukkan username dan password - jika gagal terus, program akan otomatis berhenti demi keamanan. Setelah berhasil login, petugas akan masuk ke menu utama yang terus berulang, sehingga bisa melayani banyak customer tanpa perlu restart. Pada menu ini, petugas bisa memilih verifikasi rental atau keluar. Saat verifikasi, sistem akan menanyakan data customer satu per satu (usia, SIM A, deposit, dan pengalaman mengemudi) dan langsung memvalidasi setiap input. Jika ada syarat yang tidak terpenuhi seperti usia kurang dari 21 tahun, tidak punya SIM A, atau deposit kurang dari Rp 500.000, customer langsung ditolak dan sistem kembali ke menu. Kalau semua syarat terpenuhi, sistem akan menentukan jenis mobil yang boleh disewa: hanya mobil standar untuk yang pengalaman mengemudi di bawah 4 tahun, atau semua jenis mobil termasuk premium untuk yang sudah berpengalaman 4 tahun ke atas.

3. Source Code

3.1. Login

Bagian login ini memvalidasi username dan password dengan memberikan maksimal 5 kali kesempatan, dimana username menggunakan fungsi `.lower()` agar tidak case-sensitive, kemudian jika berhasil login program akan lanjut ke menu utama menggunakan `break`, tetapi jika gagal 5 kali maka variabel percobaan akan mencapai angka 5 dan program otomatis berhenti tanpa melanjutkan ke bagian selanjutnya.

```
import os

username = "pares"
password = "045"
percobaan = 0

while percobaan < 5:
    print("Login dulu broo")
    masukusername = input("Username: ")
    masukpassword = input("Password: ")

    if (masukusername.lower() == username and masukpassword == password):
        # pakai func bawaan python lower() semua inputan jadi lower jadi
        # biarapun input PARES tetap akan bisa login (Menangani case-sensitive)
        print("Login berhasil!")
        break # keluar dari loop login next ke loop pt sebelumnya (pt-3)

    else:
        percobaan = percobaan + 1
        print(
            "Username/Password mu salah, coba lagi. Sisa percobaan:", 5 - percobaan
        ) # print berapa kali sudah dicoba

if percobaan == 5:
    print("Batas percobaan habis. Program selesai.") # percobaan dah abis
```

3.2. Code Posttest Sebelumnya (PT-3) ditambah dengan menu untuk pilih dan looping

Setelah login berhasil, program masuk ke loop posttest sebelumnya yang menampilkan menu utama secara terus-menerus, dimana user bisa memilih opsi 1 untuk verifikasi rental atau 0 untuk keluar, kemudian jika memilih verifikasi, sistem akan meminta input data customer satu per satu (usia, SIM, deposit, pengalaman) dan langsung memvalidasi setiap input menggunakan `.isdigit()` untuk memastikan input berupa angka, `.lower()` untuk case-insensitive pada input SIM, dan jika ada validasi yang gagal program akan menggunakan `continue` untuk kembali ke menu utama, sedangkan jika semua validasi berhasil maka sistem akan menentukan apakah customer disetujui untuk mobil standar saja ($\text{pengalaman} < 4$ tahun) atau semua jenis mobil termasuk premium ($\text{pengalaman} \geq 4$ tahun).

```
else: # lanjut ke posttest sebelumnya (pt-3)
    while True:
        print("\n" + "=" * 40)
        print("SISTEM RENTAL MOBIL")
        print("=" * 40)
        print("1. Proses verifikasi rental")
        print("0. Berhenti")
        pilihan = input("\nPilih menu [0/1]: ")

        if pilihan == "1":
            print("\n--- INPUT DATA CUSTOMER ---")

            # input usia dan validasinya
            usia_input = input("Masukkan usia Anda: ")
            if (
                usia_input.isdigit()
            ): # isdigit untuk cek apakah isi dari string nya
                adalah digit (angka)

                usia = int(usia_input)
                if usia < 21:
                    print("\n--- HASIL VERIFIKASI ---")
                    print("TOLAK: Usia tidak mencukupi")
                    print("    Usia minimal 21 tahun untuk rental
mobil")
```

```

        print("\n" + "=" * 40)
        print("    Terima kasih telah menggunakan")
        print("        sistem rental mobil kami!")
        print("=" * 40)
        continue # beri continue agar saat ditolak
ia kembali ke pilih menu
    else:
        print("INPUT TIDAK VALID: Masukkan angka untuk
usia")

        print("\n" + "=" * 40)
        print("    Terima kasih telah menggunakan")
        print("        sistem rental mobil kami!")
        print("=" * 40)
        continue # beri continue agar saat ditolak ia
kembali ke pilih menu

# Validasi input SIM
sim = input("Apakah Anda memiliki SIM A? (ya/tidak):
").lower()
if sim == "ya":
    pass # lanjut ke input berikutnya
elif sim == "tidak":
    print("\n--- HASIL VERIFIKASI ---")
    print("TOLAK: Tidak memiliki SIM A")
    print("    SIM A diperlukan untuk menyewa mobil")
    print("\n" + "=" * 40)
    print("    Terima kasih telah menggunakan")
    print("        sistem rental mobil kami!")
    print("=" * 40)
    continue # beri continue agar saat ditolak ia
kembali ke pilih menu
else:
    print("INPUT TIDAK VALID: Ketik 'ya' atau
'tidak' saja")

    print("\n" + "=" * 40)
    print("    Terima kasih telah menggunakan")
    print("        sistem rental mobil kami!")
    print("=" * 40)
    continue # beri continue agar saat ditolak ia
kembali ke pilih menu

```

```

        # Validasi input deposit
        deposit_input = input("Masukkan jumlah deposit (Rp):
")
        if (
            deposit_input.isdigit()
        ): # isdigit untuk cek apakah isi dari string nya
            adalah digit (angka)
            deposit = int(deposit_input)
            if deposit < 500000:
                print("\n--- HASIL VERIFIKASI ---")
                print("TOLAK: Deposit tidak cukup")
                print(
                    f"    Deposit minimal Rp 500.000, Anda
hanya menyeter Rp {deposit:,"}
                )
                print("\n" + "=" * 40)
                print("    Terima kasih telah menggunakan")
                print("        sistem rental mobil kami!")
                print("=" * 40)
                continue # beri continue agar saat ditolak
ia kembali ke pilih menu
            else:
                print("INPUT TIDAK VALID: Masukkan angka untuk
deposit")

                print("\n" + "=" * 40)
                print("    Terima kasih telah menggunakan")
                print("        sistem rental mobil kami!")
                print("=" * 40)
                continue # beri continue agar saat ditolak ia
kembali ke pilih menu

        # Validasi input pengalaman
        pengalaman_input = input("Berapa tahun pengalaman
mengemudi Anda: ")
        if (
            pengalaman_input.isdigit()
        ): # isdigit untuk cek apakah isi dari string nya
            adalah digit (angka)
            pengalaman = int(pengalaman_input)
        else:
            print("INPUT TIDAK VALID: Masukkan angka untuk
pengalaman mengemudi")

```

```

        print("\n" + "=" * 40)
        print("    Terima kasih telah menggunakan")
        print("        sistem rental mobil kami!")
        print("=" * 40)
        continue # beri continue agar saat ditolak ia
kembali ke pilih menu

        print("\n--- HASIL VERIFIKASI ---")
        if pengalaman < 4:
            print("SETUJUI UNTUK MOBIL STANDAR SAJA")
            print("Pengalaman mengemudi Anda kurang dari 4
tahun")
            print("Mobil premium memerlukan pengalaman
minimal 4 tahun")
        else:
            print("SETUJUI UNTUK SEMUA JENIS MOBIL")
            print("Selamat! Anda memenuhi semua
persyaratan")
            print("Anda bisa menyewa mobil standar maupun
premium")

```

3.3. Menu Pilihan 0 dan Pembersihan Terminal

Jika user memilih 0 maka terminal akan dibersihkan dengan `os.system()`, menampilkan pesan terima kasih, lalu `break` untuk keluar dari loop dan mengakhiri program, sedangkan jika input selain 0 atau 1 maka akan muncul pesan error dan `continue` mengembalikan ke menu utama.

```

elif pilihan == "0":
    # bersihkan terminal saat berhenti (poin tambah)
    os.system("cls" if os.name == "nt" else "clear")
    print("\n" + "=" * 40)
    print("    Terima kasih telah menggunakan")
    print("        sistem rental mobil kami!")
    print("=" * 40)
    print("Program selesai. Sampai jumpa!")
    break # keluar dari loop menu
else:
    print("\nPilihan tidak valid.")
    input("Enter untuk kembali ke menu...")
    continue # beri continue agar kembali ke pilih menu

```


4. Hasil Output

```
batas percobaan mobil. Program selesai.  
PS D:\praktikum-apd\post-test\post-test-apd-4> python 2509106045-AthasyahrSyawalFahrezy-PT-4.py  
Login dulu brooo  
Username: Pares  
Password: 045  
Login berhasil!  
  
=====
```

SISTEM RENTAL MOBIL

```
=====
```

1. Proses verifikasi rental
0. Berhenti

Pilih menu [0/1]: █

Gambar 4.1 Login dan Pilihan Menu

```
batas percobaan mobil. Program selesai.  
PS D:\praktikum-apd\post-test\post-test-apd-4> python 2509106045-AthasyahrSyawalFahrezy-PT-4.py  
Login dulu brooo  
Username: Pares  
Password: 045  
Login berhasil!  
  
=====
```

SISTEM RENTAL MOBIL

```
=====
```

1. Proses verifikasi rental
0. Berhenti

Pilih menu [0/1]: 1

```
--- INPUT DATA CUSTOMER ---  
Masukkan usia Anda: 21  
Apakah Anda memiliki SIM A? (ya/tidak): ya  
Masukkan jumlah deposit (Rp): 500000  
Berapa tahun pengalaman mengemudi Anda: 3  
  
--- HASIL VERIFIKASI ---  
SETUJUT UNTUK MOBIL STANDAR SAJA  
Pengalaman mengemudi Anda kurang dari 4 tahun  
Mobil premium memerlukan pengalaman minimal 4 tahun  
  
=====
```

SISTEM RENTAL MOBIL

```
=====
```

1. Proses verifikasi rental
0. Berhenti

Pilih menu [0/1]: █

Gambar 4.2 Input Data dan Looping Menu

```
=====
```

Terima kasih telah menggunakan
sistem rental mobil kami!

```
=====
```

Program selesai. Sampai jumpa!
PS D:\praktikum-apd\post-test\post-test-apd-4> █

Gambar 4.3 Jika Berhenti

5. Langkah-langkah GIT

5.1 GIT Init

```
PS D:\praktikum-apd> git init
Reinitialized existing Git repository in D:/praktikum-apd/.git/
```

Gambar 5.1 Git Init

Saya menjalankan git init, muncul seperti itu karena folder sudah pernah dijadikan repository Git sebelumnya, jadi Git hanya menginisialisasi ulang repo lama, bukan membuat yang baru.

5.2 GIT Add

```
Reinitialized existing Git repository in D:/praktikum-apd/.git/
PS D:\praktikum-apd> git add .
PS D:\praktikum-apd>
```

Gambar 5.2 Git Add

Saya menjalankan git add, maksudnya saya sedang memasukkan perubahan file ke staging area supaya siap untuk di-commit.

5.3 GIT Commit

```
PS D:\praktikum-apd> git commit -m "Tambah Posttest 4"
[main 41ef4cd] Tambah Posttest 4
10 files changed, 255 insertions(+), 21 deletions(-)
create mode 100644 A2_2025/pertemuan4/break.py
create mode 100644 A2_2025/pertemuan4/continue.py
create mode 100644 A2_2025/pertemuan4/for.py
create mode 100644 A2_2025/pertemuan4/kursi.py
create mode 100644 A2_2025/pertemuan4/listcompre.py
create mode 100644 A2_2025/pertemuan4/nestedloop.py
create mode 100644 A2_2025/pertemuan4/segitiga.py
create mode 100644 A2_2025/pertemuan4/while.py
create mode 100644 post-test/post-test-apd-4/2509100045-AthasyahriSyawalFahrezy-PT-4.py
PS D:\praktikum-apd>
```

Gambar 5.3 Git Commit

Git commit adalah perintah untuk menyimpan perubahan yang sudah masuk staging area ke dalam riwayat repository.

5.4 GIT Remote

```
delete mode 100644 post-test/post-test-apd-4/2509100045-AthasyahriSyawalFahrezy-PT-4-copy.py
PS D:\praktikum-apd> git remote add origin https://github.com/asyaress/praktikum-apd-25.git
error: remote origin already exists.
PS D:\praktikum-apd>
```

Gambar 5.4 Git Remote

Saya menjalankan git remote add origin, muncul error karena sebenarnya remote bernama origin sudah ada sebelumnya di repo ini. Jadi Git menolak menambah ulang dengan nama yang sama, karena setiap remote hanya boleh punya satu nama unik.

5.5 GIT Push

```
PS D:\praktikum-apd> git push -u origin main
Enumerating objects: 22, done.
Counting objects: 100% (22/22), done.
Delta compression using up to 12 threads
Compressing objects: 100% (16/16), done.
Writing objects: 100% (17/17), 3.41 KiB | 973.00 KiB/s, done.
Total 17 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To https://github.com/asyaress/praktikum-apd-25.git
 6d79385..41ef4cd main -> main
branch 'main' set up to track 'origin/main'.
PS D:\praktikum-apd>
```

Gambar 5.5 Git Push

Saya menjalankan git push -u origin main, hasilnya adalah mengirimkan (upload) commit yang ada di branch main lokal ke GitHub (remote origin).