

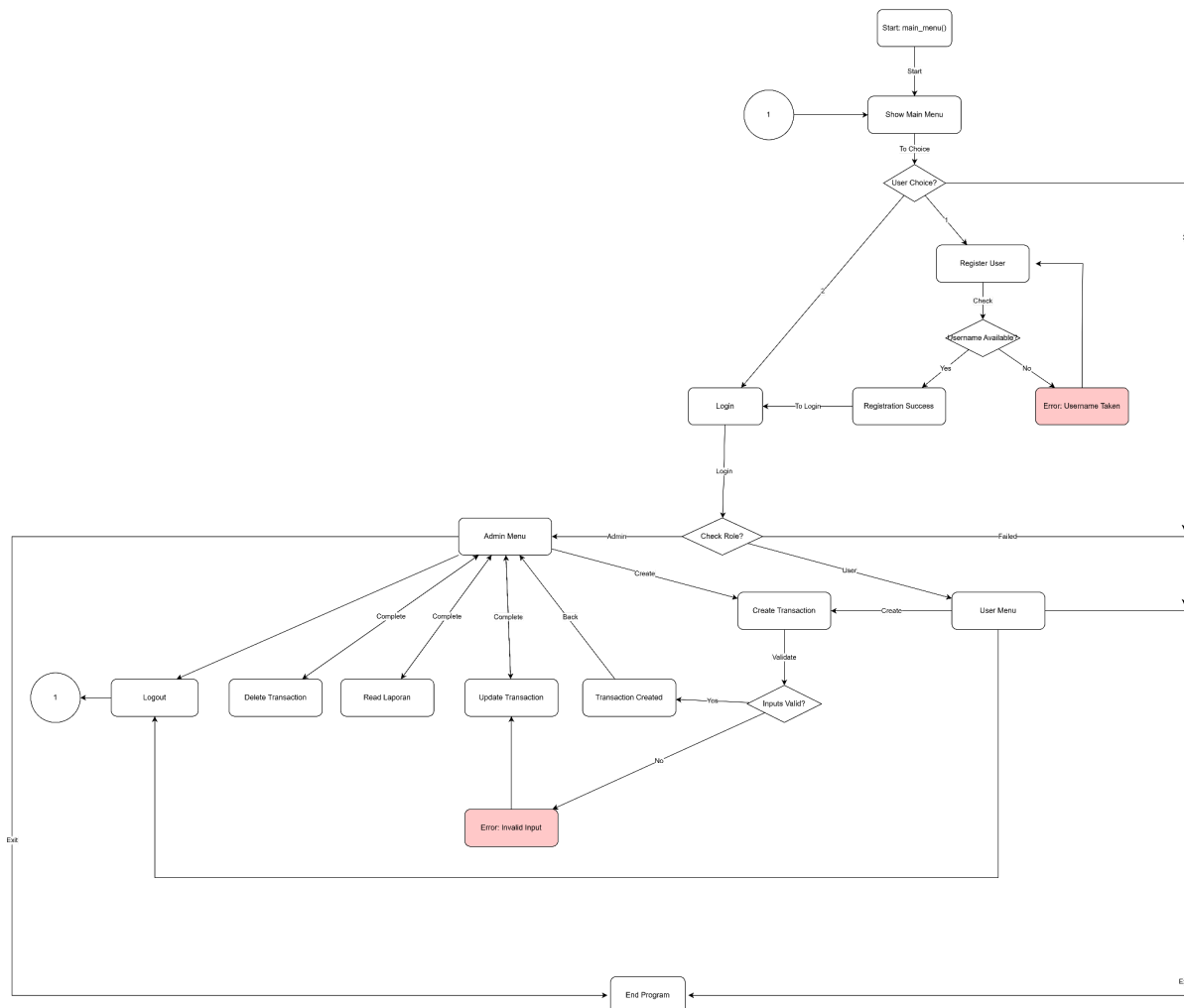
LAPORAN PRAKTIKUM
POSTTEST (6)
ALGORITMA PEMROGRAMAN DASAR



Disusun oleh:
Arya Fickri Al Farazi (2509106022)
Kelas (A1 '25)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

1. Flowchart



2. Deskripsi Singkat Program

program ini merupakan aplikasi manajemen warnet sederhana yang dibuat menggunakan bahasa Python. Sistem ini memungkinkan pengguna untuk melakukan registrasi dan login dengan dua jenis peran, yaitu **admin** dan **user**. Data pengguna dan transaksi disimpan dalam dictionary, di mana setiap pengguna memiliki username, password, dan peran masing-masing. Saat pertama kali dijalankan, program otomatis menambahkan akun admin bawaan agar dapat mengakses seluruh fitur utama.

Setelah login, pengguna dapat mengakses menu sesuai perannya. Pengguna biasa hanya dapat membuat transaksi penyewaan komputer dengan memasukkan waktu mulai, durasi, dan nomor PC. Sementara itu, admin memiliki hak lebih luas,

termasuk membuat transaksi, melihat laporan pendapatan harian, memperbarui data transaksi, serta membatalkan transaksi yang sudah dibuat. Program juga menampilkan total pendapatan pada tanggal tertentu berdasarkan transaksi aktif yang tercatat.

Selain itu, sistem ini dilengkapi dengan validasi input untuk memastikan data yang dimasukkan sesuai format dan bernilai positif, seperti pengecekan format tanggal dan angka. Setiap transaksi akan menghasilkan ID unik, status aktif atau dibatalkan, serta perhitungan otomatis total biaya berdasarkan durasi dan harga per jam. Dengan fitur-fitur tersebut, program ini berfungsi sebagai simulasi sederhana dari sistem administrasi penyewaan komputer di warnet.

3. Source Code

```
import os
from datetime import datetime

data = {
    'users': {},
    'transactions': {}
}

if 'admin' not in data['users']:
    data['users']['admin'] = {'password': '123', 'role': 'admin'}

HARGA_PER_JAM = 12000

def register_user():
    username = input("Masukkan username baru: ")
    if username in data['users']:
        print("Username sudah digunakan.")
        return None
    password = input("Masukkan password: ")
    data['users'][username] = {'password': password, 'role': 'user'}
    print(f"Pengguna {username} berhasil terdaftar sebagai user.")
    return username

def login():
    username = input("Masukkan username: ")
    password = input("Masukkan password: ")
```

```

    if username in data['users'] and data['users'][username]['password'] ==
password:
        print(f"Login berhasil sebagai {data['users'][username]['role']}.")
        return data['users'][username]['role']
    print("Login gagal. Username atau password salah.")
    return None

def get_next_id():
    return len(data['transactions']) + 1

def create_transaksi(waktu_mulai, durasi, nomor_pc):
    date_part = waktu_mulai.split(' ')[0]
    if len(date_part.split('-')) != 3:
        print("Error: Waktu mulai harus dalam format YYYY-MM-DD.")
        return

    if not isinstance(durasi, int) or durasi < 0:
        print("Error: Durasi harus angka positif.")
        return

    if not isinstance(nomor_pc, int) or nomor_pc < 0:
        print("Error: Nomor PC harus angka positif.")
        return

    total_biaya = HARGA_PER_JAM * durasi
    new_transaksi = {
        'id': get_next_id(),
        'waktu_mulai': waktu_mulai,
        'durasi': durasi,
        'total_biaya/jam': total_biaya,
        'nomor_pc': nomor_pc,
        'total_biaya': total_biaya,
        'status': 'aktif'
    }
    data['transactions'][new_transaksi['id']] = new_transaksi
    print(f"Transaksi berhasil dibuat ID: {new_transaksi['id']}, Dengan
Harga/Jam: Rp 12.00, Total Biaya Menjadi: Rp {total_biaya}")

def read_laporan(tanggal):
    if not isinstance(tanggal, str) or len(tanggal.split('-')) != 3:
        print("Error: Tanggal harus dalam format YYYY-MM-DD.")
        return

```

```

    harian = [t for t in data['transactions'].values() if
t['waktu_mulai'].split(' ')[0] == tanggal and t['status'] == 'aktif']
    total_pendapatan = sum(t['total_biaya'] for t in harian)
    print(f"Laporan Pendapatan untuk {tanggal}: Rp {total_pendapatan}")
    for t in harian:
        print(f"ID: {t['id']}, Waktu: {t['waktu_mulai']}, Total Biaya: Rp
{t['total_biaya']}")

def update_transaksi(id_transaksi, durasi_baru):
    if id_transaksi not in data['transactions']:
        print("Transaksi tidak ditemukan.")
        return
    if not isinstance(durasi_baru, int) or durasi_baru < 0:
        print("Error: Durasi harus angka positif.")
        return
    t = data['transactions'][id_transaksi]
    if t['status'] == 'aktif':
        t['durasi'] = durasi_baru
        t['total_biaya'] = HARGA_PER_JAM * durasi_baru
        print(f"Transaksi ID {id_transaksi} diupdate.")
    else:
        print("Transaksi tidak aktif.")

def delete_transaksi(id_transaksi):
    if id_transaksi not in data['transactions']:
        print("Transaksi tidak ditemukan.")
        return
    t = data['transactions'][id_transaksi]
    t['status'] = 'dibatalkan'
    print(f"Transaksi ID {id_transaksi} dibatalkan.")

def main_menu():
    while True:
        print("Selamat datang! Pilih: 1. Registrasi 2. Login 3. Keluar")
        pilihan_awal = input("Masukkan pilihan: ")
        if pilihan_awal == '1':
            register_user()
            role = login()
        elif pilihan_awal == '2':
            role = login()
        elif pilihan_awal == '3':
            print("Terima kasih Telah Datang.")

```

```

        return
    else:
        print("Pilihan tidak valid.")
        continue

    while role:
        os.system('cls' if os.name == 'nt' else 'clear')
        if role == 'admin':
            print("1. Buat Transaksi  2. Laporan  3. Update  4. Hapus  5. Logout")
        else:
            print("1. Buat Transaksi  2. Logout")

        pilihan = input("Pilih: ")
        if pilihan == '1':
            waktu_mulai = input("Waktu mulai (YYYY-MM-DD): ")
            durasi = int(input("Durasi: "))
            nomor_pc = int(input("Nomor PC: "))
            create_transaksi(waktu_mulai, durasi, nomor_pc)
        elif pilihan == '2' and role == 'admin':
            read_laporan(input("Tanggal: "))
        elif pilihan == '2' and role != 'admin':
            break
        elif pilihan == '3' and role == 'admin':
            update_transaksi(int(input("ID: ")), int(input("Durasi baru: ")))
        elif pilihan == '4' and role == 'admin':
            delete_transaksi(int(input("ID: ")))
        elif pilihan == '5' and role == 'admin':
            print("Anda telah logout.")
            role = None # Logout
        else:
            print("Pilihan tidak valid.")

        input("Tekan Enter")

if __name__ == "__main__":
    main_menu()

```

4. Hasil Output

Contoh hasil output jika termasuk admin

```
//Login
Selamat datang! Pilih: 1. Registrasi 2. Login 3. Keluar
Masukkan pilihan: 2
Masukkan username: admin
Masukkan password: 123

//Buat transaksi
1. Buat Transaksi 2. Laporan 3. Update 4. Hapus 5. Logout
Pilih: 1
Waktu mulai (YYYY-MM-DD): 2025-12-21
Durasi: 3
Nomor PC: 1
Transaksi berhasil dibuat ID: 1, Dengan Harga/Jam: Rp 12.00, Total Biaya
Menjadi: Rp 36000
Tekan Enter

//Laporan
1. Buat Transaksi 2. Laporan 3. Update 4. Hapus 5. Logout
Pilih: 2
Tanggal: 2025-12-21
Laporan Pendapatan untuk 2025-12-21: Rp 36000
ID: 1, Waktu: 2025-12-21, Total Biaya: Rp 36000
Tekan Enter

//Update
1. Buat Transaksi 2. Laporan 3. Update 4. Hapus 5. Keluar
Pilih: 3
ID: 1
Durasi baru: 6
Transaksi ID 1 diupdate.
Tekan Enter

//Delete
1. Buat Transaksi 2. Laporan 3. Update 4. Hapus 5. Keluar
Pilih: 4
ID: 1
Transaksi ID 1 dibatalkan.
Tekan Enter
```

Contoh hasil output jika termasuk user

```
//Register
Selamat datang! Pilih: 1. Registrasi 2. Login 3. Keluar
Masukkan pilihan: 1
Masukkan username baru: arya
Masukkan password: 123
Pengguna arya berhasil terdaftar sebagai user.

Masukkan username: arya
Masukkan password: 123

//Buat Transaksi
1. Buat Transaksi 2. Logout
Pilih: 1
Waktu mulai (YYYY-MM-DD): 2025-12-21
Durasi: 4
Nomor PC: 1
Transaksi berhasil dibuat ID: 1, Dengan Harga/Jam: Rp 12.00, Total Biaya
Menjadi: Rp 48000
Tekan Enter
```