

**LAPORAN PRAKTIKUM**  
**POSTTEST 5**  
**ALGORITMA PEMROGRAMAN DASAR**

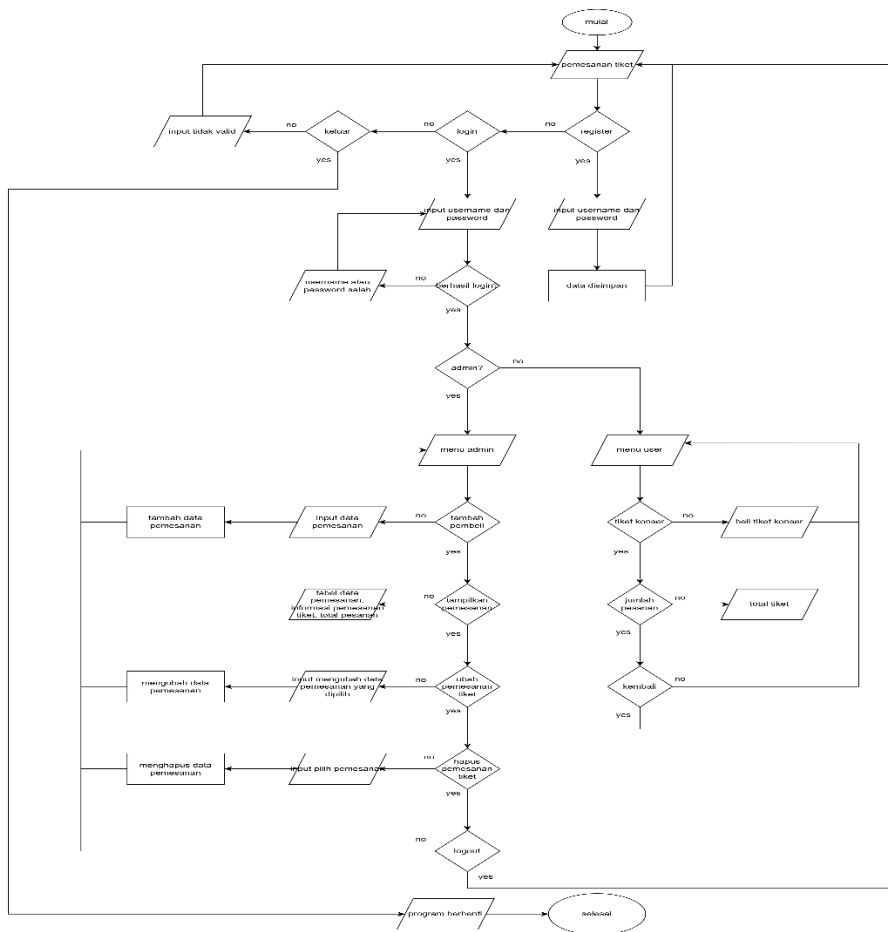


**Disusun oleh:HIBRIZI YUSRIANI (2509106131)**

**Kelas :(C2'25)**

**PROGRAM STUDI INFORMATIKA**  
**UNIVERSITAS MULAWARMAN**  
**SAMARINDA 2025**

**1. Flowchart**



Gambar 1.1 flowchart

Declare berfungsi sebagai digunakan untuk memberi tahu program bahwa sebuah variabel, fungsi, atau struktur data akan digunakan, dan menentukan jenisnya.

Assign berfungsi sebagai untuk memberi nilai atau tugas. Input berfungsi sebagai untuk memasukkan data atau informasi.

if berfungsi sebagai untuk mengecek Keputusan atau mengoreksi apakah benar atau salah. Fungsi

while dalam pemrograman digunakan untuk menjalankan sebuah blok kode berulang kali selama kondisi tertentu bernilai benar (True).

Fungsi for dalam pemrograman digunakan untuk melakukan perulangan dengan jumlah yang sudah diketahui atau bisa ditentukan. Ini sangat berguna saat kamu ingin mengulang sesuatu berdasarkan urutan, seperti daftar, range angka, atau karakter dalam string. output berfungsi untuk menyebut hasilnya .

## 2. Penjelasan singkat flowchart

pertama-tama kita akan disuruh masukkan jika sudah login kalian akan di tampilkan ke menu, disitu kalian akan ditampilkan;

1. Lihat semua pemesanan
2. Tambahkan pemesanan
3. Ubah pemesanan
4. Hapus pemesanan
5. Keluar

## 3. sourcecode

```
import os

from prettytable import PrettyTable

data_pemesanan = [

    ("Rijie", "081234567890", 2, "Transfer Bank"),

    ("Luna", "082145678921", 1, "QRIS"),

]

data_pengguna = [("rijie", "admin123", "admin")]

# ===== LOGIN SYSTEM =====

os.system("cls || clear")

print("=====")

print("      LOGIN SISTEM PEMESANAN      ")

print("=====")

username_input = input("Masukkan username: ").lower()

password_input = input("Masukkan password: ").strip()
```

```

# Verifikasi login

for u in data_pengguna:

    if u[0] == username_input and u[1] == password_input:

        level_akses = u[2]

while True:

    os.system("cls || clear")

    print("\n=====")

    print("    BOOKING TIKET KONSER THE SIGIT")

    print("=====")


    table = PrettyTable()

    table.field_names = ["No", "Menu"]

    table.align["No"] = "c"

    table.align["Menu"] = "l"


    table.add_row(["1", "Lihat semua pemesanan"])

    table.add_row(["2", "Tambah pemesanan"])

    table.add_row(["3", "Ubah pemesanan"])

    table.add_row(["4", "Hapus pemesanan"])

    table.add_row(["5", "Keluar"])


    print(table)


    # Input pilihan menu

    menu = input("Pilih menu (1-5): ").strip()

# READ

```

```

if menu == "1":

    os.system("cls")

    print("=== Daftar ===")

    if len(data_pemesanan) == 0:

        print("Belum ada Pemesanan")

    else:

        tabel = PrettyTable()

        tabel.field_names = ["No", "Nama", "No hp", "jumlah", "nama
pembayaran"]

        tabel.hrules = 1

        for i in range(len(data_pemesanan)):

            nama, noHp, jumlah, bayar = data_pemesanan[i]

            tabel.add_row([i+1, nama, noHp, jumlah, bayar ])

        print(tabel)

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

elif menu == "2":

    os.system("cls")

    print("=== Tambah Pemesanan ===")

    nama = input("nama :")

    nomor_hp = input("nomor hp : ")

    jumlah_tiket = input("jumlah tiket: ")

    metode_pembayaran = input("metode pembayaran: ")

    data_pemesanan.append([nama, nomor_hp, jumlah_tiket,
metode_pembayaran])

    print("\npemesanan berhasil ditambahkan")

# UPDATE

```

```

elif menu == "3":

    print("\n=== UBAH PEMESANAN ===")

    if not data_pemesanan:

        print("Belum ada data untuk diubah.")

    else:

        for i, (nama, hp, jumlah, bayar) in enumerate(data_pemesanan,
start=1):

            print(f"{i}. {nama} | {hp} | {jumlah} Tiket | {bayar}")

        pilih = input("Pilih nomor data yang ingin diubah: ").strip()

        if not pilih.isdigit():

            print("Input harus berupa angka!")

        else:

            pilih = int(pilih)

            if not (1 <= pilih <= len(data_pemesanan)):

                print(" Nomor tidak valid!")

            else:

                nama_baru = input("Nama baru: ").strip()

                hp_baru = input("No HP baru: ").strip()

                jumlah_baru_input = input("Jumlah tiket baru:
").strip()

                bayar_baru = input("Metode pembayaran baru: ").strip()

                if not nama_baru or not hp_baru or not
jumlah_baru_input or not bayar_baru:

                    print(" Semua data harus diisi.")

                elif not jumlah_baru_input.isdigit() or
int(jumlah_baru_input) <= 0:

                    print(" Jumlah tiket harus angka lebih dari 0.")

                else:

```

```

        jumlah_baru = int(jumlah_baru_input)

        data_pemesanan[pilih - 1] = (nama_baru, hp_baru,
jumlah_baru, bayar_baru)

        print(" Data berhasil diperbarui!")

# DELETE

elif menu == "4":

    print("\n=== HAPUS PEMESANAN ===")

    if not data_pemesanan:

        print("Belum ada data untuk dihapus.")

    else:

        for i, (nama, hp, jumlah, bayar) in enumerate(data_pemesanan,
start=1):

            print(f"{i}. {nama} | {hp} | {jumlah} Tiket | {bayar}")

        pilih = input("Pilih nomor data yang ingin dihapus: ").strip()

        if not pilih.isdigit():

            print("Input harus berupa angka!")

        else:

            pilih = int(pilih)

            if not (1 <= pilih <= len(data_pemesanan)):

                print(" Nomor tidak valid!")

            else:

                konfirmasi = input(f"Yakin hapus {data_pemesanan[pilih
- 1][0]}? (y/n): ").lower().strip()

                if konfirmasi == "y":

                    del data_pemesanan[pilih - 1]

                    print(" Data berhasil dihapus!")

                else:

                    print("Batal dihapus.")

```

```

# EXIT

elif menu == "5":

    print("Terima kasih! Program selesai.")

    break

else:

    print("Pilihan menu tidak valid, coba lagi!")

```

#### 4.Hasil Output

```

+---+-----+-----+-----+-----+
=====
      BOOKING TIKET KONSER THE SIGIT
=====
+---+-----+-----+-----+
| No | Menu                |
+---+-----+-----+-----+
| 1  | Lihat semua pemesanan |
| 2  | Tambah pemesanan      |
| 3  | Ubah pemesanan        |
| 4  | Hapus pemesanan       |
| 5  | Keluar                 |
+---+-----+-----+-----+
Pilih menu (1-5): █

```

Gambar 1 menampilkan menu pertama



```
=== Daftar ===
+---+-----+-----+-----+-----+
| No | Nama | No hp | jumlah | nama pembayaran |
+---+-----+-----+-----+
| 1 | Rjie | 081234567890 | 2 | Transfer Bank |
+---+-----+-----+-----+
| 2 | Luna | 082145678921 | 1 | QRIS |
+---+-----+-----+-----+
| 3 | rijie2 | 849830830803 | 4 | qris |
+---+-----+-----+-----+

Tekan Enter untuk kembali...
```

Gambar 2 menampilkan crud READ

```
=== Tambah Pemesanan ===
nama :kanda
nomor hp : 0826r275275
jumlah tiket: 1
metode pembayaran: pinjol
```

Gambar 3 menampilkan crud CREATE

```
=== UBAH PEMESANAN ===
1. Rjie | 081234567890 | 2 Tiket | Transfer Bank
2. Luna | 082145678921 | 1 Tiket | QRIS
3. rijie2 | 849830830803 | 4 Tiket | qris
4. kanda karca | 0873636646 | 1 Tiket | cash
5. kanda | 0826r275275 | 1 Tiket | pinjol
Pilih nomor data yang ingin diubah: 4
Nama baru: rando
No HP baru: 082537253
Jumlah tiket baru: 1
Metode pembayaran baru: utang
```

Gambar 4 menampilkan crud UPDATE

```
=== HAPUS PEMESANAN ===
1. Rijie | 081234567890 | 2 Tiket | Transfer Bank
2. Luna | 082145678921 | 1 Tiket | QRIS
3. rijie2 | 849830830803 | 4 Tiket | qris
4. rando | 082537253 | 1 Tiket | utang
5. kanda | 0826r275275 | 1 Tiket | pinjol
Pilih nomor data yang ingin dihapus: 5
Yakin hapus kanda? (y/n): y
```

Gambar 5 menampilkan crud delete

## 5. Langkah-langkah GIT

### 5.1 GIT Add

```
PS D:\praktikum-apd> git add .
```

Gambar 1 Git add . ini digunakan untuk menambahkan semua file yang ada di folder saat ini ke staging area.

### 5.2 GIT Commit

```
PS D:\praktikum-apd> git commit -m "upload pt 5"
[main 17ccee0] upload pt 5
3 files changed, 536 insertions(+)
create mode 100644 kelas/pertemuan-5/pertemuan5.py
create mode 100644 post-test/post-test-apd-5/2509106093-israndomirhjah-pt-5.py
create mode 100644 post-test/post-test-apd-5/2509106093_israndomirhjah.drawio
```

Gambar 2 Git commit -m “first commit” ini digunakan untuk menyimpan file yang ada di staging area ke dalam repository lokal, dan -m “first commit” untuk menambahkan komentar, pada tahap ini file masih berada di repository lokal komputer belum terupload ke github.

### 5.3 GIT Push

```
PS D:\praktikum-apd> git push
Enumerating objects: 12, done.
Counting objects: 100% (12/12), done.
Delta compression using up to 12 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (9/9), 4.83 KiB | 1.61 MiB/s, done.
Total 9 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/israndomirhjah/praktikum-apd.git
0a7ddc8..17ccee0 main -> main
```

Gambar 3 Git push ini dipakai untuk mengupload commit dari repository lokal ke repository github di branch main.