

**LAPORAN PRAKTIKUM  
POSTTEST (8)  
ALGORITMA PEMROGRAMAN DASAR**

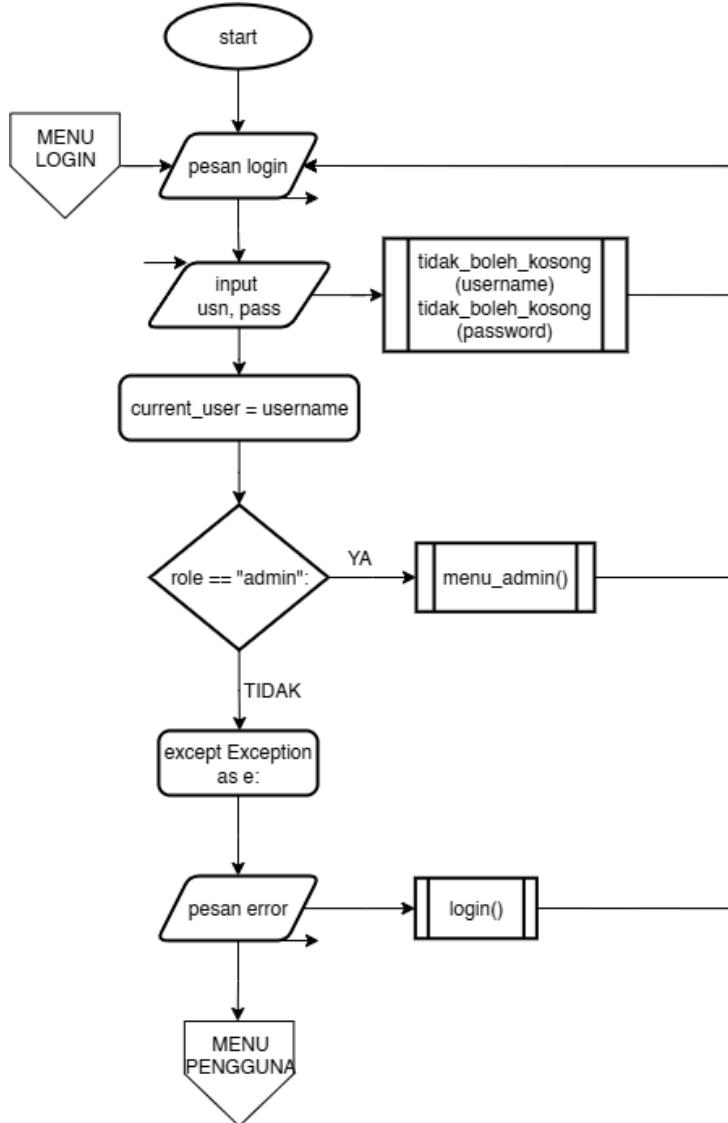


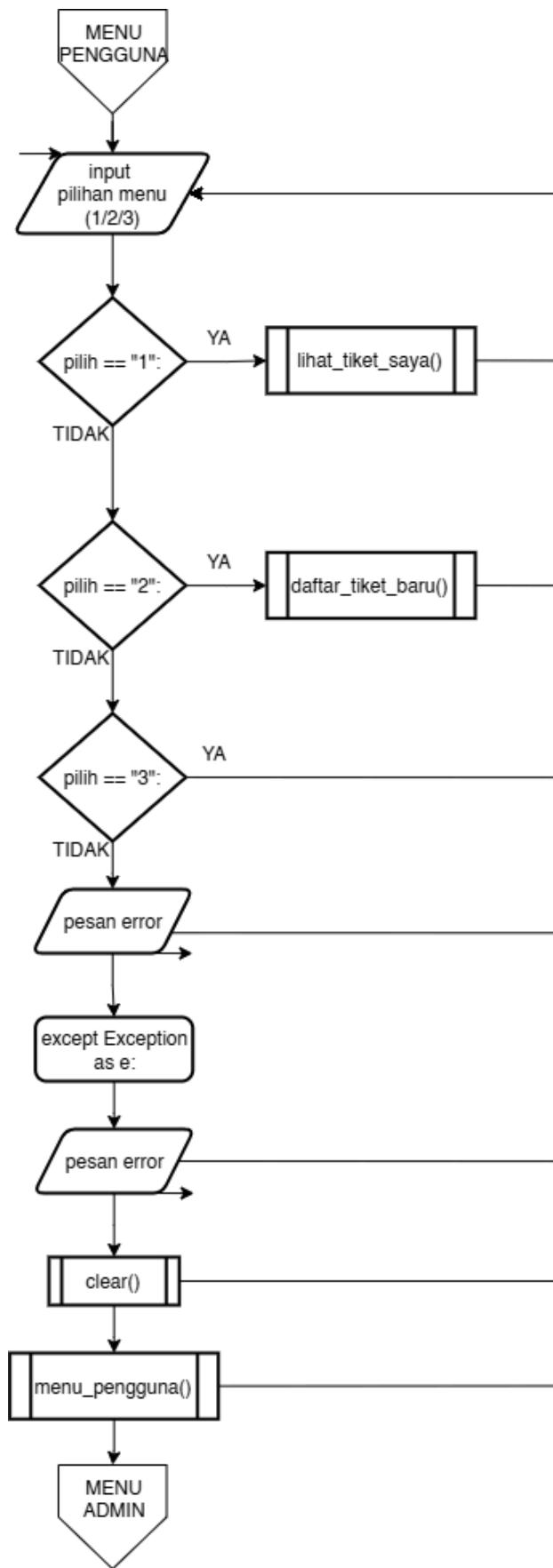
**Disusun oleh:  
Khanza Humaira (2509106065)  
Kelas (B1 '25)**

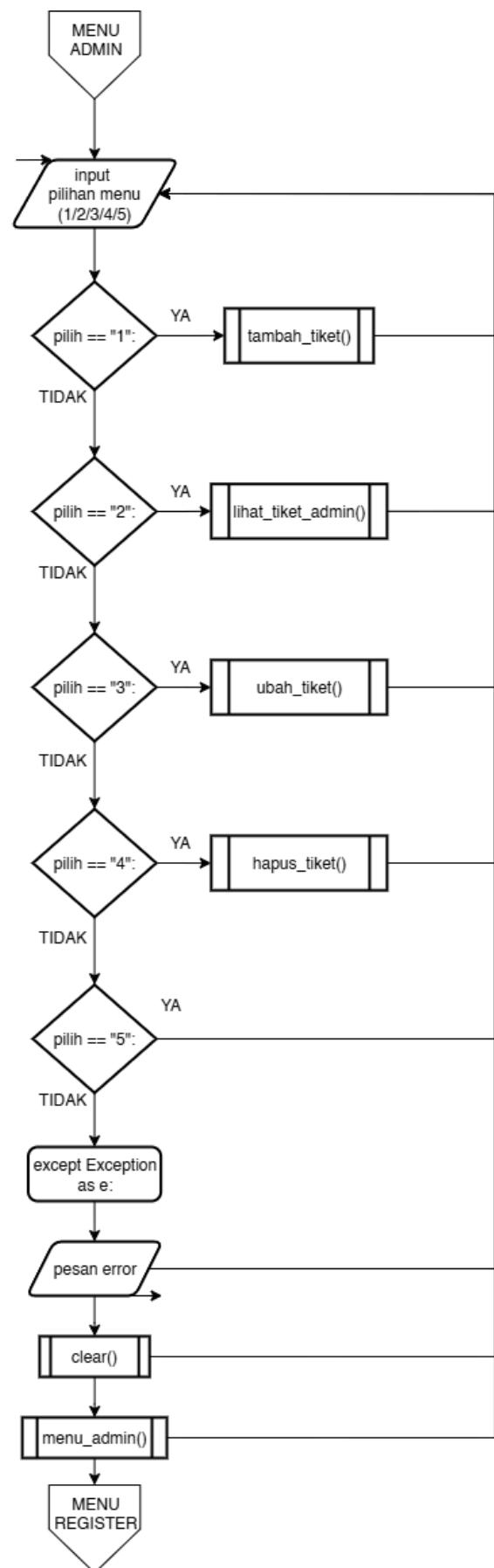
**PROGRAM STUDI INFORMATIKA  
UNIVERSITAS MULAWARMAN  
SAMARINDA  
2025**

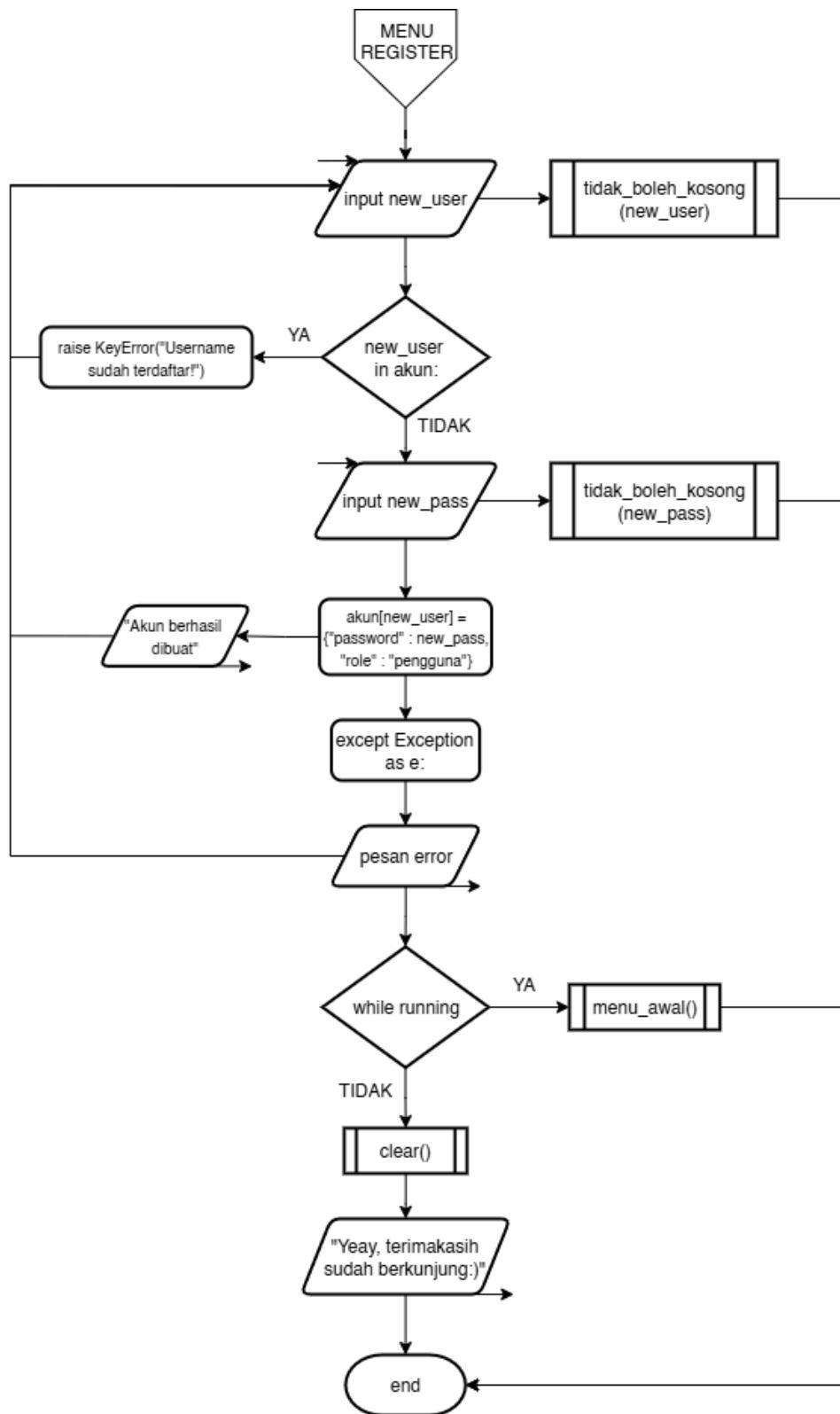
## 1. Flowchart

Secara singkatnya, flowchart yang saya buat ini lebih ringkas dan mudah dipahami alurnya. Diawali dengan start, lalu data variabel global serta proses user yang sedang login. Kemudian pemanggilan fungsi tanpa parameter yaitu menu\_awal(). Dilanjutkan dengan input pilihan menu awal (Login, Register, Logout). Jika memilih logout maka otomatis terhubung ke end. Jika memilih login maka user diminta menginput usn dan pass lalu dijalankan fungsi serta error handling (jika input kosong/salah). Jika benar, maka lanjut ke role (admin/pengguna biasa) yang dilakukan pemanggilan fungsi sesuai role tersebut lalu menginput menu pilihan role tersebut. Semua hasil mengarah ke berakhirnya program (end).









## 2. Deskripsi Singkat Program

Pada post test terakhir, program lebih terorganisir dan mudah dikelola karena menggunakan struktur modular, kode dipisah menjadi beberapa modul/file py. Dalam program yang saya buat terdapat beberapa jenis modul yaitu [main.py](#), [data.py](#), [login.py](#), [admin.py](#), [user.py](#), [fungsi.py](#). Terdapat juga penggunaan library yaitu PrettyTable dan Colorama. Dengan PrettyTable, saat melihat data tiket (admin/user) output lebih rapi dalam bentuk tabel. Sedangkan Colorama digunakan untuk memberi warna teks pada output tertentu, misal; jika pesan berhasil maka output teksnya menjadi hijau, kuning untuk pesan konfirmasi atau peringatan, biru/cyan untuk tampilan header/judul, merah jika pesan gagal/error,

## 3. Source Code

### A. Kode Program MANAJEMEN DATA E TIKET IKN FUN RUN 10KM

Source Code:

```
# MAIN
from fungsi import clear, pause
from data import running
from login import login, register
from colorama import Fore, Style, init
init(autoreset=True)

def menu_awal():
    global running
    while running:
        clear()
        print("MENU AWAL SISTEM MANAJEMEN DATA E-TIKET EVENT IKN FUN RUN
10KM")
        print("1. Login (Admin/Pengguna)")
        print("2. Register Akun Baru")
        print("3. Keluar")
        pilih = input("Pilih menu (1/2/3): ")

        if pilih == "1":
            login()
        elif pilih == "2":
```

```

        register()
    elif pilih == "3":
        print(Fore.CYAN + "Yeay, terimakasih sudah berkunjung :)")
        break
    else:
        print(Fore.RED + "Pilihan tidak valid.")
        pause()

if __name__ == "__main__":
    menu_awal()

# DATA
akun = {
    "khanza": {"password": "065", "role": "admin"},
    "ghina": {"password": "ghina01", "role": "pengguna"},
    "van": {"password": "van02", "role": "pengguna"},
    "rumay": {"password": "rumay03", "role": "pengguna"}
}

data_tiket = {
    "IKN01": {"nama": "ghina", "kategori": "10KM", "email": "ghina@gmail.com", "no_hp": "0812345678", "usia": "18"}, 
    "IKN02": {"nama": "van", "kategori": "10KM", "email": "van@gmail.com", "no_hp": "0898766541", "usia": "22"}, 
    "IKN03": {"nama": "rumay", "kategori": "10KM", "email": "rumay@gmail.com", "no_hp": "0835754632", "usia": "12"}
}

current_user = ""
running = True

```

```

# LOGIN

from data import akun, current_user
from fungsi import clear, pause, tidak_boleh_kosong
from admin import menu_admin
from user import menu_pengguna
from colorama import Fore, Style, init
init(autoreset=True)

def cek_login(username, password):
    if username in akun and akun[username]["password"] == password:
        return akun[username]["role"]
    else:
        raise KeyError(Fore.RED + "Username atau Password salah!")

def login():
    global current_user
    clear()
    print("Silakan Login sebagai (Admin/Pengguna)")
    try:
        username = input("Masukkan Username : ")
        password = input("Masukkan Password : ")
        tidak_boleh_kosong(username)
        tidak_boleh_kosong(password)

        role = cek_login(username, password)
        print(Fore.GREEN + f"Login berhasil sebagai {role}")
        from data import current_user as cu
        cu = username
        if role == "admin":
            menu_admin(username)
        else:
            menu_pengguna(username)
    except Exception as e:
        print(Fore.RED + f"Error: {e}")
        login()

def register():
    from data import akun
    clear()
    print("REGISTER NEW AKUN IKN FUN RUN 10KM")
    try:

```

```
new_user = input("Masukkan username baru: ")
tidak_boleh_kosong(new_user)

if new_user in akun:
    raise KeyError(Fore.RED + "Username sudah terdaftar!")

new_pass = input("Masukkan password: ")
tidak_boleh_kosong(new_pass)

akun[new_user] = {"password": new_pass, "role": "pengguna"}
print(Fore.GREEN + "Akun berhasil dibuat.")
except Exception as e:
    print(Fore.RED + f"Error: {e}")
pause()

# FUNGSI
import os
from colorama import Fore, Style

def clear():
    os.system('cls' if os.name == 'nt' else 'clear')

def pause():
    input(Fore.YELLOW + "\nTekan ENTER untuk kembali..." + Style.RESET_ALL)

def tidak_boleh_kosong(teks):
    if teks.strip() == "":
        raise ValueError("Input tidak boleh kosong!")

def cek_no_hp(no_hp):
    return no_hp.isdigit()
```

```

# ADMIN

from prettytable import PrettyTable
from fungsi import clear, pause, tidak_boleh_kosong
from data import data_tiket
from colorama import Fore, Style
from colorama import Fore, Style, init
init(autoreset=True)

def total_tiket():
    return len(data_tiket)

def lihat_tiket_admin():
    clear()
    print(Fore.CYAN + "=" * 65)
    print("DATA E-TIKET IKN FUN RUN 10KM".center(65))
    print("=" * 65 + Style.RESET_ALL)
    if len(data_tiket) == 0:
        print(Fore.RED + "Belum ada data E-Tiket.")
    else:
        tabel = PrettyTable(["ID", "Nama", "Kategori", "Email", "No HP",
"Usia"])
        for id_t, t in data_tiket.items():
            tabel.add_row([id_t, t['nama'], t['kategori'], t['email'],
t['no_hp'], t['usia']])
        print(tabel)
        print(Fore.GREEN + f"Total E-Tiket: {total_tiket()}")
    pause()

def tambah_tiket():
    clear()
    print("TAMBAH DATA E-TIKET IKN FUN RUN 10KM")
    try:
        id_tiket = input("Masukkan ID E-Tiket : ")
        tidak_boleh_kosong(id_tiket)

        if id_tiket in data_tiket:
            raise KeyError(Fore.RED + "ID sudah tersedia!")

        nama = input("Masukkan Nama Peserta : ")
        email = input("Masukkan Email : ")
        no_hp = input("Masukkan Nomor HP : ")
    
```

```

usia = input("Masukkan Usia : ")

data_tiket[id_tiket] = {
    "nama": nama,
    "kategori": "10KM",
    "email": email,
    "no_hp": no_hp,
    "usia": usia
}
print(Fore.GREEN + "Data E-Tiket sukses ditambahkan.")

except Exception as e:
    print(Fore.RED + f"Error: {e}")
pause()

def ubah_tiket():
    clear()
    print("=" * 65)
    print("UBAH DATA E-TIKET IKN FUN RUN 10KM".center(65))
    print("=" * 65)

    if len(data_tiket) == 0:
        print(Fore.RED + "Belum ada data E-Tiket.")
        pause()
        return
    else:
        tabel = PrettyTable()
        tabel.field_names = ["ID", "Nama", "Kategori", "Email", "No HP",
"Usia"]
        for id_t, t in data_tiket.items():
            tabel.add_row([id_t, t['nama'], t['kategori'], t['email'],
t['no_hp'], t['usia']])
        print(tabel)

try:
    id_up = input("Masukkan ID untuk e-tiket yang ingin diubah : ")
    if id_up not in data_tiket:
        raise KeyError(Fore.RED + "ID E-Tiket tidak ditemukan!")

    print("Masukkan data baru (biarkan kosong jika tidak ingin diubah)")
    nama_baru = input(f"Nama baru ({data_tiket[id_up]['nama']} : ") or
data_tiket[id_up]['nama']

```

```

        email_baru = input(f"Email baru ({data_tiket[id_up]['email']} : ")
or data_tiket[id_up]['email']
        nohp_baru = input(f"No HP baru ({data_tiket[id_up]['no_hp']} : ")
or data_tiket[id_up]['no_hp']
        usia_baru = input(f"Usia baru ({data_tiket[id_up]['usia']} : ") or
data_tiket[id_up]['usia']

        data_tiket[id_up].update({
            "nama": nama_baru,
            "email": email_baru,
            "no_hp": nohp_baru,
            "usia": usia_baru
        })

        print(Fore.GREEN + "Data E-Tiket sukses diubah")
    except Exception as e:
        print(Fore.RED + f"Error: {e}")
    pause()

def hapus_tiket():
    clear()
    print("=" * 65)
    print("HAPUS DATA E-TIKET IKN FUN RUN 10KM".center(65))
    print("=" * 65)

    if len(data_tiket) == 0:
        print(Fore.RED + "Belum ada data E-Tiket.")
        pause()
        return
    else:
        tabel = PrettyTable()
        tabel.field_names = ["ID", "Nama", "Kategori", "Email", "No HP",
"Usia"]
        for id_t, t in data_tiket.items():
            tabel.add_row([id_t, t['nama'], t['kategori'], t['email'],
t['no_hp'], t['usia']])
        print(tabel)

    try:
        id_del = input("Masukkan ID untuk e-tiket yang ingin dihapus : ")
        if id_del not in data_tiket:

```

```

        raise KeyError(Fore.RED + "ID E-Tiket tidak ditemukan!")

    konfirmasi = input(f"Apakah Anda yakin ingin menghapus e-tiket {id_del}? (y/t): ").lower()
    if konfirmasi == "y":
        del data_tiket[id_del]
        print(Fore.GREEN + "Data E-Tiket sukses dihapus")
    else:
        print(Fore.RED + "Tiket batal dihapus.")
except Exception as e:
    print(Fore.RED + f"Error: {e}")
pause()

def menu_admin(current_user):
    while True:
        clear()
        print(f"MENU ADMIN ({current_user})")
        print("1. Tambah E-Tiket")
        print("2. Lihat Data E-Tiket")
        print("3. Ubah Data E-Tiket")
        print("4. Hapus Data E-Tiket")
        print("5. Logout")
        pilih = input(Fore.YELLOW + "Pilih menu: ")

        if pilih == "1": tambah_tiket()
        elif pilih == "2": lihat_tiket_admin()
        elif pilih == "3": ubah_tiket()
        elif pilih == "4": hapus_tiket()
        elif pilih == "5": break
        else:
            print(Fore.RED + "Pilihan tidak valid!")
            pause()

```

```

# USER

from prettytable import PrettyTable
from fungsi import clear, pause
from data import data_tiket
from fungsi import cek_no_hp
from colorama import Fore, Style, init
init(autoreset=True)

def lihat_tiket_saya(current_user):
    clear()
    print(f"DATA E-TIKET ATAS NAMA: {current_user}")
    tabel = PrettyTable(["ID", "Nama", "Kategori", "Email", "No HP",
    "Usia"])
    ada = False
    for id_t, v in data_tiket.items():
        if v['nama'] == current_user:
            ada = True
            tabel.add_row([id_t, v['nama'], v['kategori'], v['email'],
v['no_hp'], v['usia']])
    if ada:
        print(tabel)
    else:
        print(Fore.RED + "Belum ada e-tiket atas nama Anda.")
    pause()

def daftar_tiket_baru(current_user):
    clear()
    print("DAFTAR E-TIKET BARU")
    try:
        new_id = "IKN" + str(len(data_tiket) + 1).zfill(2)
        email = input("Email : ")
        no_hp = input("No HP : ")
        if not cek_no_hp(no_hp):
            raise ValueError(Fore.RED + "Nomor HP harus angka!")
        usia = input("Usia : ")

        data_tiket[new_id] = {
            "nama": current_user,
            "kategori": "10KM",
            "email": email,
    
```

```
        "no_hp": no_hp,
        "usia": usia
    }
    print(Fore.GREEN + f"E-Tiket dengan id {new_id} sukses
didaftarkan.")

except Exception as e:
    print(Fore.RED + f"Error: {e}")
pause()

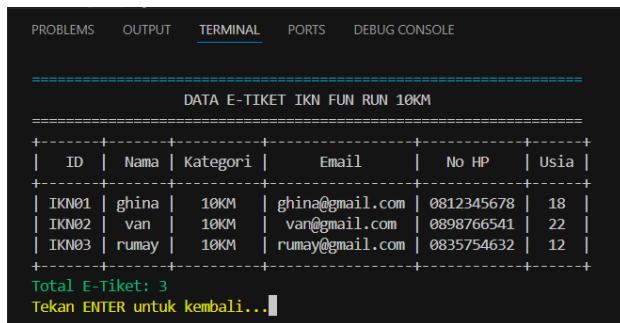
def menu_pengguna(current_user):
    while True:
        clear()
        print(f"MENU PENGGUNA ({current_user})")
        print("1. Lihat E-Tiket Saya")
        print("2. Daftar E-Tiket Baru")
        print("3. Logout")
        pilih = input("Pilih menu: ")

        if pilih == "1": lihat_tiket_saya(current_user)
        elif pilih == "2": daftar_tiket_baru(current_user)
        elif pilih == "3": break
        else:
            print(Fore.RED + "Pilihan tidak valid!")
            pause()
```

## 4. Hasil Output

```
MENU ADMIN (khanza)
1. Tambah E-Tiket
2. Lihat Data E-Tiket
3. Ubah Data E-Tiket
4. Hapus Data E-Tiket
5. Logout
Pilih menu: |
```

Gambar 4.1 Screenshot output teks konfirmasi dengan library colorama style yellow



The screenshot shows a terminal window with tabs: PROBLEMS, OUTPUT, TERMINAL (underlined), PORTS, DEBUG CONSOLE. The terminal displays a table of e-ticket data:

ID	Nama	Kategori	Email	No HP	Usia
IKN01	ghina	10KM	ghina@gmail.com	0812345678	18
IKN02	van	10KM	van@gmail.com	0898766541	22
IKN03	rumay	10KM	rumay@gmail.com	0835754632	12

Total E-Tiket: 3  
Tekan ENTER untuk kembali...|

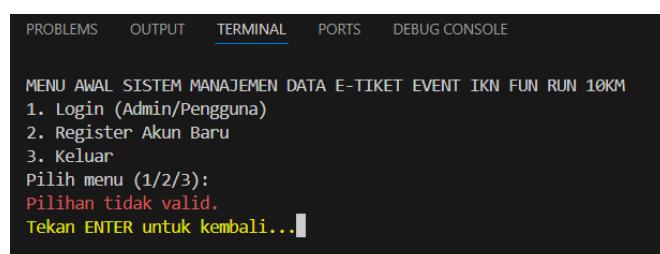
Gambar 4.2 Screenshot output dengan library PrettyTable  
dan colorama style biru untuk header



The screenshot shows a terminal window with tabs: PROBLEMS, OUTPUT, TERMINAL (underlined), PORTS, DEBUG CONSOLE. The terminal displays the following message:

DAFTAR E-TIKET BARU  
Email : azel@gmail.com  
No HP : 083153165370  
Usia : 19  
E-Tiket dengan id IKN04 sukses didaftarkan.  
Tekan ENTER untuk kembali...|

Gambar 4.3 Screenshot output berhasil daftar  
tiket baru dengan library colorama style green



The screenshot shows a terminal window with tabs: PROBLEMS, OUTPUT, TERMINAL (underlined), PORTS, DEBUG CONSOLE. The terminal displays the following message:

MENU AWAL SISTEM MANAJEMEN DATA E-TIKET EVENT IKN FUN RUN 10KM  
1. Login (Admin/Pengguna)  
2. Register Akun Baru  
3. Keluar  
Pilih menu (1/2/3):  
Pilihan tidak valid.  
Tekan ENTER untuk kembali...|

Gambar 4.4 Screenshot output teks error dengan library colorama style red



The screenshot shows a terminal window with tabs: PROBLEMS, OUTPUT, TERMINAL (underlined), PORTS, DEBUG CONSOLE. The terminal displays the following message:

MENU AWAL SISTEM MANAJEMEN DATA E-TIKET EVENT IKN FUN RUN 10KM  
1. Login (Admin/Pengguna)  
2. Register Akun Baru  
3. Keluar  
Pilih menu (1/2/3): 3  
Yeay, terimakasih sudah ber kunjung :)  
PS C:\Praktikum APD\post-test\2509106065\_KHANZA\_HUMAIRA\_PT-8> |

Gambar 4.5 Screenshot output teks tampilan dengan library colorama style cyan

## 5. Langkah-langkah GIT

### 5.1 GIT Add

GIT Add adalah tahap kedua setelah membuat atau mengubah file. Dengan GIT Add maka git akan otomatis tahu file mana yang harus di simpan versi terbarunya.

```
PS C:\Praktikum APD> git add .
PS C:\Praktikum APD>
```

Gambar 5.1 Screenshot perintah git add

### 5.2 GIT Commit

Setelah GIT Add, maka perlu GIT Commit untuk memberi catatan pada filenya atau juga termasuk diberi pesan seperti “pt8”.

```
PS C:\Praktikum APD> git commit -m 'pt8'
[main ed4ad7a] pt8
 11 files changed, 319 insertions(+)
```

Gambar 5.2 Screenshot perintah git commit

### 5.3 GIT Push

Tahap terakhir yaitu mendorong file repo ke github dengan git push agar dapat diakses secara online kapanpun dan dimanapun.

```
PS C:\Praktikum APD> git push -u origin main
Enumerating objects: 18, done.
Counting objects: 100% (18/18), done.
Delta compression using up to 12 threads
Compressing objects: 100% (16/16), done.
Writing objects: 100% (16/16), 12.09 KiB | 3.02 MiB/s, done.
Total 16 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/KhanzaHumaira29/praktikum_apd.git
  fe1e1a9..ed4ad7a  main -> main
branch 'main' set up to track 'origin/main'.
```

Gambar 5.3 Screenshot perintah git push