

LAPORAN PRAKTIKUM
POSTTEST (7)
ALGORITMA PEMROGRAMAN DASAR

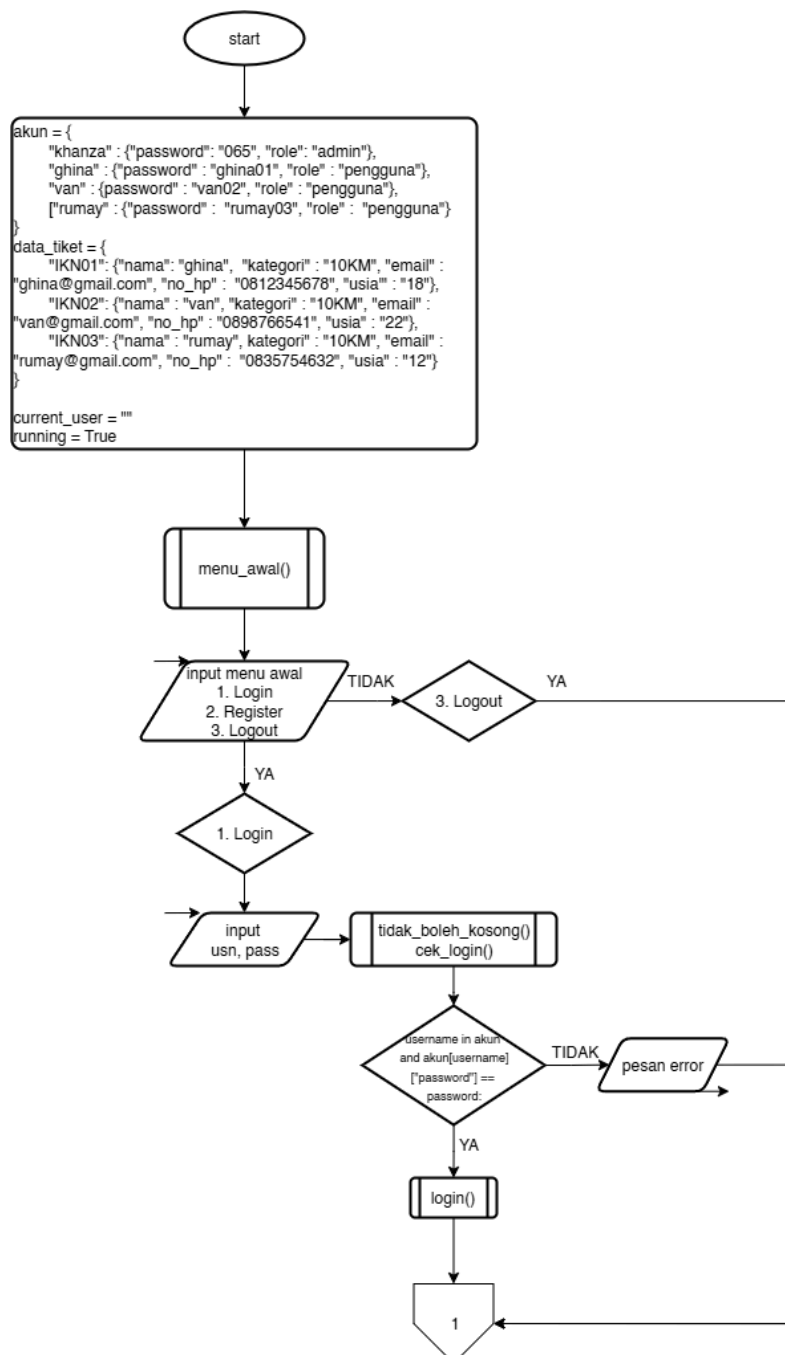


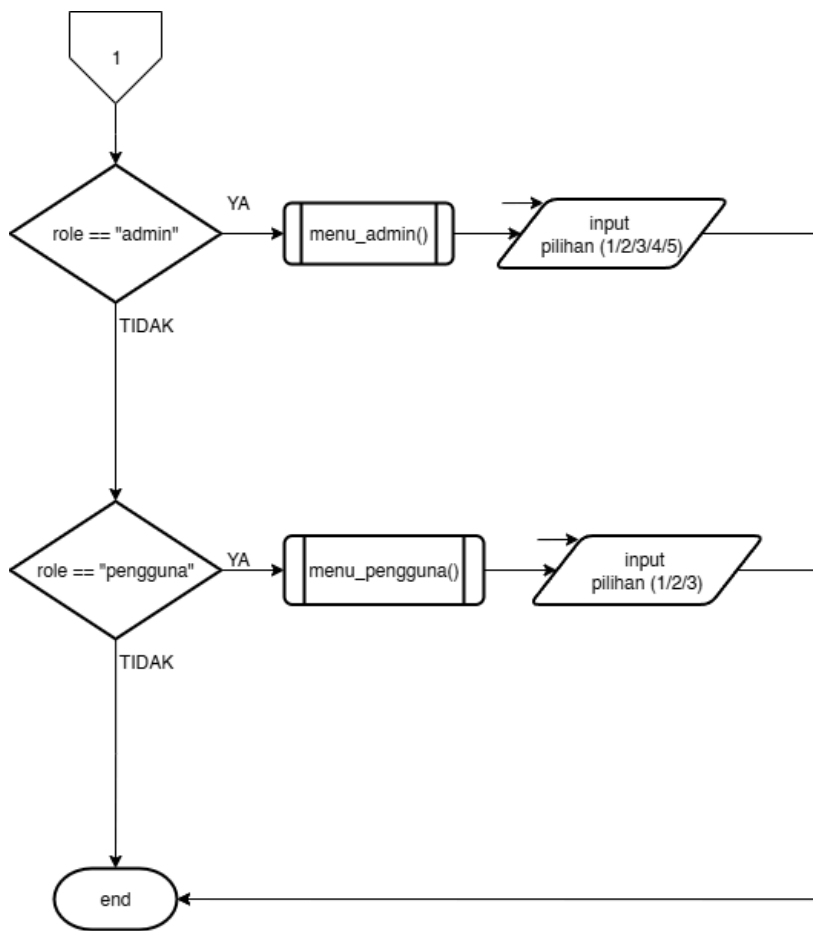
Disusun oleh:
Khanza Humaira (2509106065)
Kelas (B1 '25)

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

Program ini dibuat untuk mengelola data e-tiket IKN FUN RUN 10KM secara sederhana. Di dalamnya ada multiuser yaitu admin dan pengguna biasa. Admin bisa menambah, melihat, mengubah, dan menghapus data e-tiket, sedangkan pengguna biasa hanya bisa melihat dan mendaftarkan tiket atas nama mereka sendiri jika sudah melakukan register akun baru. Program ini juga punya fitur untuk register akun baru. Tujuan utamanya agar pengelolaan data e-tiket jadi lebih praktis dan terorganisir. Pada kode kali ini terdapat penambahan seperti fungsi (rekursif, parameter & tanpa parameter) serta error handling menggunakan try except yang membuat program lebih efisien.

3. Source Code

A. Kode Program Manajemen Data E-Tiket IKN FUN RUN 10KM

Source Code:

```
import os
# Variabel Global
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
```

```

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

def pause():
    input("Tekan ENTER untuk kembali.")

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

# Fungsi dgn Parameter
def cek_login(username, password):
    if username in akun and akun[username]["password"] == password:
        return akun[username]["role"]
    else:
        raise KeyError("Username atau Password salah!")

# Fungsi dgn parameter
def cek_no_hp(no_hp):
    return no_hp.isdigit()

# Fungsi tanpa parameter
def total_tiket():
    return len(data_tiket)

# Fungsi Rekursif (menu)
def login():
    global current_user
    clear()
    print("Silakan Login.")
    try:
        username = input("Masukkan Username : ")
        password = input("Masukkan Password : ")

        tidak_boleh_kosong(username)
        tidak_boleh_kosong(password)

        role = cek_login(username, password)
        current_user = username

```

```

        if role == "admin":
            menu_admin()
        else:
            menu_pengguna()

    except Exception as e:
        print(f"Error: {e}")
        login()

# Fungsi Tanpa Parameter
def menu_awal():
    global 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. Logout")
    pilih = input("Pilih menu (1/2/3): ")

    try:
        if pilih == "1":
            login()
        elif pilih == "2":
            register()
        elif pilih == "3":
            running = False
        else:
            raise ValueError("Pilihan tidak valid.")
    except Exception as e:
        print(f"Error: {e}")
        pause()
        menu_awal()

# Menu Register
def register():
    clear()
    print("REGISTER E-TIKET IKN FUN RUN 10KM")
    try:
        new_user = input("Masukkan username baru: ")
        tidak_boleh_kosong(new_user)

```

```

        if new_user in akun:
            raise KeyError("Username sudah terdaftar!")

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

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

# Menu Admin
def menu_admin():
    clear()
    print(f"MENU ADMIN MANAJEMEN DATA E-TIKET IKN FUN RUN
10KM({current_user})")
    print("1. Tambah E-Tiket Baru (Create)")
    print("2. Lihat Data E-Tiket (Read)")
    print("3. Ubah Data E-Tiket (Update)")
    print("4. Hapus Data E-Tiket (Delete)")
    print("5. Logout Menu Admin")

    pilih = input("Pilih menu (1/2/3/4/5): ")

    try:
        if pilih == "1":
            clear()
            print("TAMBAH DATA E-TIKET")
            id_tiket = input("Masukkan ID E-Tiket : ")
            tidak_boleh_kosong(id_tiket)

            if id_tiket in data_tiket:
                raise KeyError("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("Data E-Tiket sukses ditambahkan.")

elif pilih == "2":
    clear()
    print("CHECK DATA E-TIKET")
    if len(data_tiket) == 0:
        print("Belum ada data E-Tiket.")
    else:
        print(f"{'ID':<8} {'Nama':<15} {'Kategori':<8} {'Email':<20} {'No HP':<12} {'Usia':<5}")
        print("-" * 75)
        for id_t, t in data_tiket.items():
            print(f"{'id_t':<8} {t['nama']:<15} {t['kategori']:<8} {t['email']:<20} {t['no_hp']:<12} {t['usia']:<5}")

elif pilih == "3":
    clear()
    print("UPDATE DATA E=TIKET")
    id_up = input("Masukkan ID : ")
    if id_up not in data_tiket:
        raise KeyError("ID E-Tiket tidak ditemukan!")

    data_tiket[id_up]['nama'] = input("Nama baru : ")
    data_tiket[id_up]['email'] = input("Email baru : ")
    data_tiket[id_up]['no_hp'] = input("Nomor HP baru : ")
    data_tiket[id_up]['usia'] = input("Usia baru : ")
    print("Data E-Tiket sukses diperbarui.")

elif pilih == "4":
    clear()
    print("DELETE DATA E-TIKET")
    id_del = input("Masukkan ID : ")
    if id_del not in data_tiket:
        raise KeyError("ID E-Tiket tidak ditemukan!")

```



```

        del data_tiket[id_del]
        print("Data E-Tiket sukses dihapus.")

    elif pilih == "5":
        return
    else:
        raise ValueError("Pilihan menu tidak valid.")
except Exception as e:
    print(f"Error: {e}")
pause()
menu_admin()

# Menu Pengguna
def menu_pengguna():
    clear()
    print(f"MENU PENGGUNA E-TIKET IKN FUN RUN 10KM({current_user})")
    print("1. Lihat E-Tiket Saya")
    print("2. Daftar E-Tiket Baru")
    print("3. Logout Menu Pengguna")
    pilih = input("Pilih menu: ")

    try:
        if pilih == "1":
            clear()
            ada = False

            print(f"{'ID':<8} {'Nama':<16} {'Kategori':<8} {'Email':<16} {'Usia':<8} {'No HP':<16}")
            print("-" * 75)

            for id_t, v in data_tiket.items():
                if v['nama'] == current_user:
                    ada = True
                    print(f"{id_t:<8} {v['nama']:<16} {v['kategori']:<8} {v['email']:<16} {v['usia']:<8} {v['no_hp']:<16}")

            if not ada:
                print("Belum ada e-tiket atas nama Anda.")
                print("-" * 75)

        elif pilih == "2":

```

```

clear()
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("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(f"E-Tiket dengan id {new_id} sukses didaftarkan.")
print(f"Total Tiket Terdaftar: {total_tiket()}")

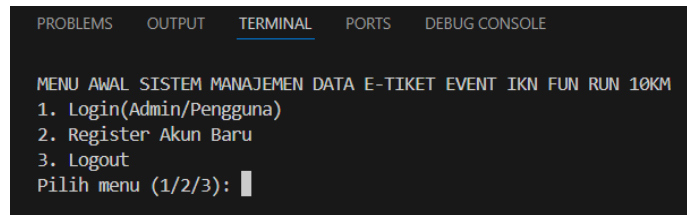
elif pilih == "3":
    return
else:
    raise ValueError("Pilihan menu tidak valid.")
except Exception as e:
    print(f"Error: {e}")
    pause()
    menu_pengguna()

while running:
    menu_awal()

clear()
print("Yeay, terimakasih sudah berkunjung:")

```

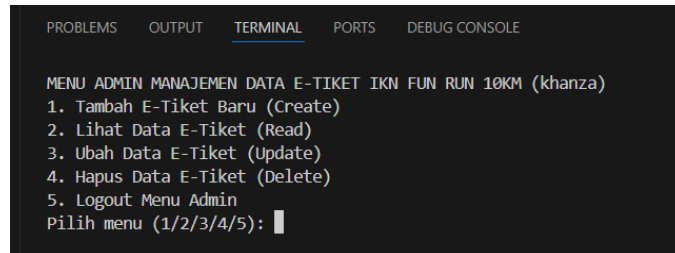
4. Hasil Output



```
PROBLEMS OUTPUT TERMINAL PORTS DEBUG CONSOLE

MENU AWAL SISTEM MANAJEMEN DATA E-TIKET EVENT IKN FUN RUN 10KM
1. Login(Admin/Pengguna)
2. Register Akun Baru
3. Logout
Pilih menu (1/2/3):
```

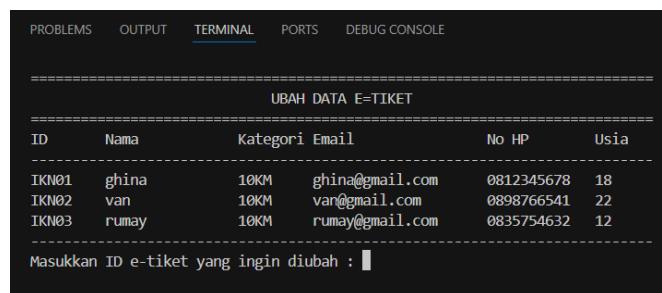
Gambar 4.1 Screenshot output menu awal program



```
PROBLEMS OUTPUT TERMINAL PORTS DEBUG CONSOLE

MENU ADMIN MANAJEMEN DATA E-TIKET IKN FUN RUN 10KM (khanza)
1. Tambah E-Tiket Baru (Create)
2. Lihat Data E-Tiket (Read)
3. Ubah Data E-Tiket (Update)
4. Hapus Data E-Tiket (Delete)
5. Logout Menu Admin
Pilih menu (1/2/3/4/5):
```

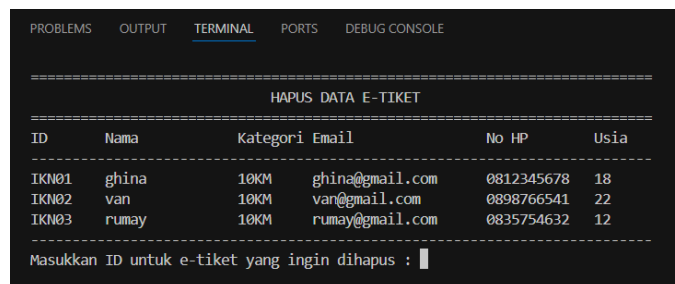
Gambar 4.2 Screenshot output menu admin + {current_user}



```
PROBLEMS OUTPUT TERMINAL PORTS DEBUG CONSOLE

=====
                        UBAH DATA E-TIKET
=====
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
-----
Masukkan ID e-tiket yang ingin diubah :
```

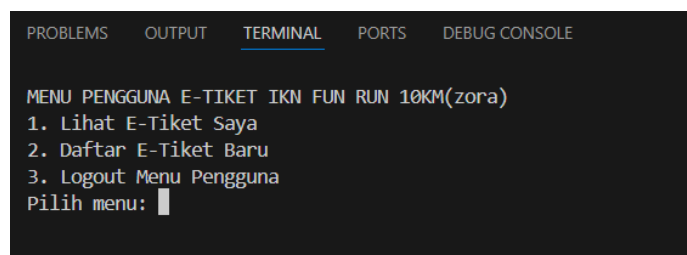
Gambar 4.3 Screenshot hasil output saat ingin mengupdate data e-tiket



```
PROBLEMS OUTPUT TERMINAL PORTS DEBUG CONSOLE

=====
                        HAPUS DATA E-TIKET
=====
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
-----
Masukkan ID untuk e-tiket yang ingin dihapus :
```

Gambar 4.4 Screenshot output saat ingin menghapus data e-tiket



```
PROBLEMS OUTPUT TERMINAL PORTS DEBUG CONSOLE

MENU PENGGUNA E-TIKET IKN FUN RUN 10KM(zora)
1. Lihat E-Tiket Saya
2. Daftar E-Tiket Baru
3. Logout Menu Pengguna
Pilih menu:
```

Gambar 4.5 Screenshot output saat login menu pengguna baru

PROBLEMS OUTPUT <u>TERMINAL</u> PORTS DEBUG CONSOLE					
ID	Nama	Kategori	Email	Usia	No HP
IKN04	zora	10KM	zora@gmail.com	10	083153165370
Tekan ENTER untuk kembali.					

Gambar 4.6 Screenshot hasil output setelah daftar e-tiket baru

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 Screenshoot 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 “pt7”.

```
PS C:\Praktikum APD> git commit -m 'pt7'
[main 7a9141e] pt7
1 file changed, 229 insertions(+), 40 deletions(-)
```

Gambar 5.2 Screenshoot 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: 14, done.
Counting objects: 100% (14/14), done.
Delta compression using up to 12 threads
Compressing objects: 100% (10/10), done.
Writing objects: 100% (10/10), 2.94 KiB | 1.47 MiB/s, done.
Total 10 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), completed with 1 local object.
To https://github.com/KhanzaHumaira29/praktikum_apd.git
02d1e72..7b8be36 main -> main
branch 'main' set up to track 'origin/main'.
```

Gambar 5.3 Screenshoot perintah git push