

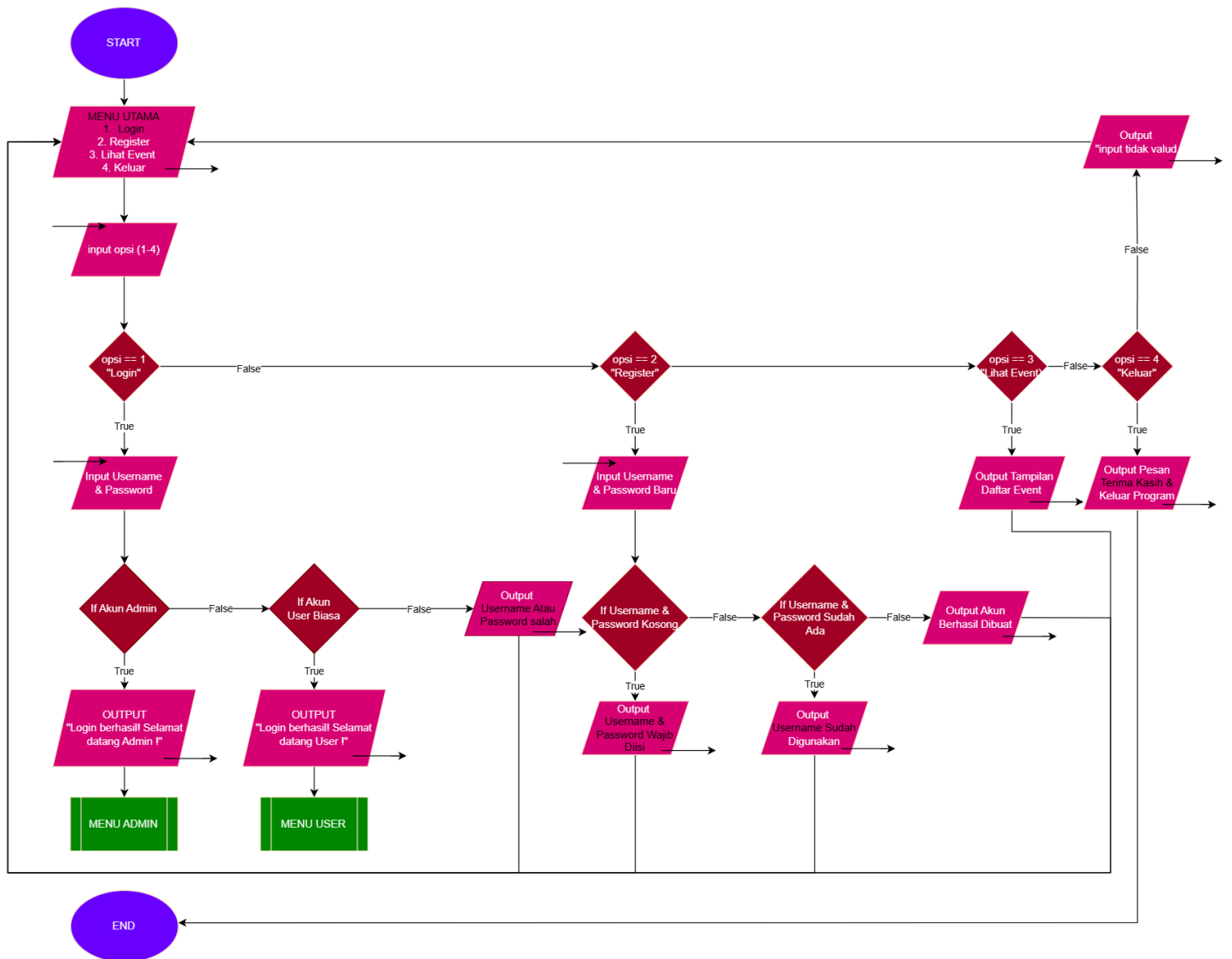
**LAPORAN PRAKTIKUM**  
**POSTTEST 7**  
**ALGORITMA PEMROGRAMAN DASAR**



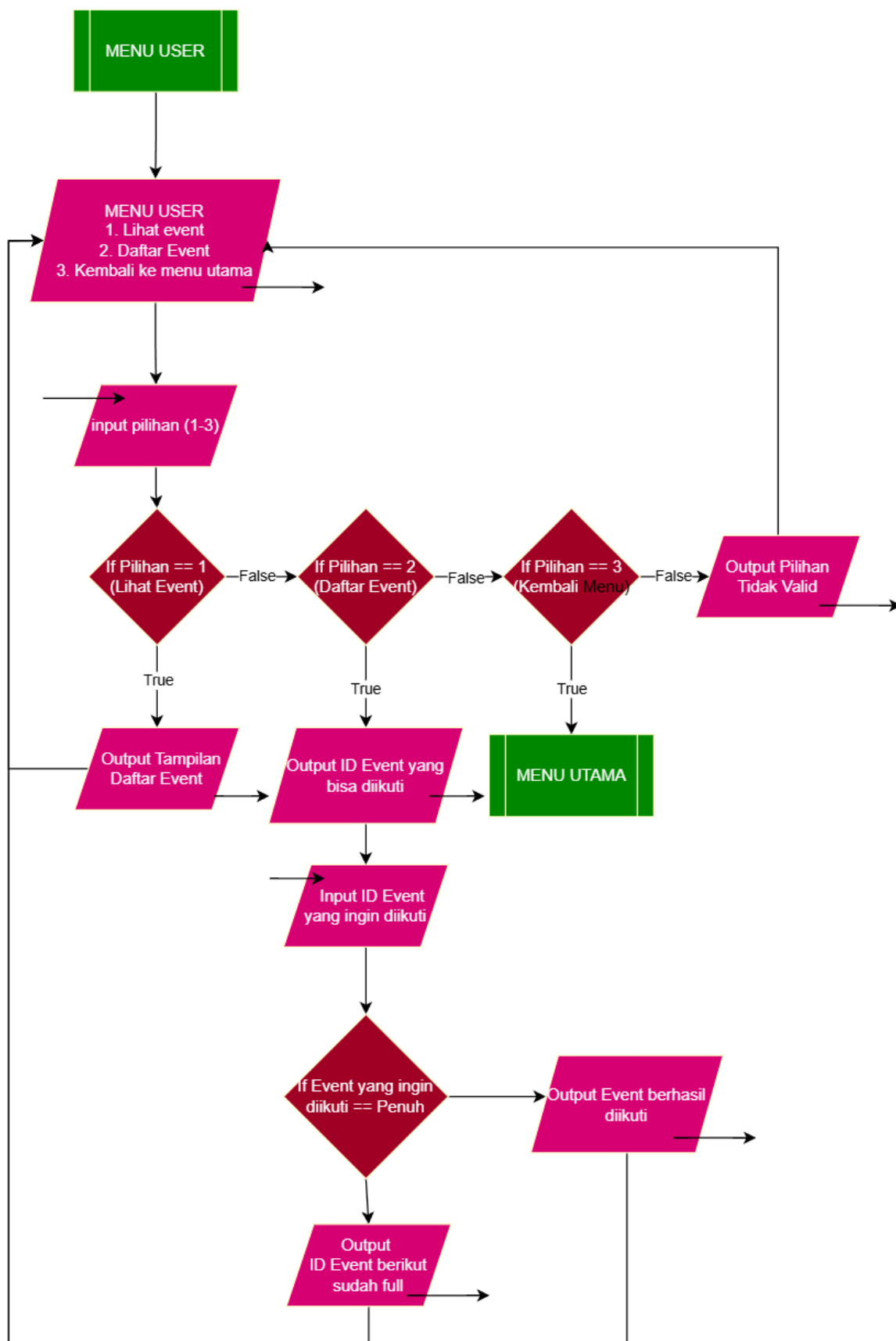
**Disusun oleh:**  
**Nuril Akmal 2509106074**  
**Kelas B2'25**

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

# 1. Flowcart



Gambar 1.1 Flowchart Menu Utama



Gambar 1.2 Flowchart Menu User



Gambar 1.3 Flowcart Menu Admin

## PENJELASAN FLOWCHART SISTEM MANAJEMEN EVENT WEB3 & HACKATHON

### 1. MENU UTAMA

- Start → Sistem menampilkan 4 pilihan:
  - Login
  - Register
  - Lihat Event (Guest)
  - Keluar
- Jika input tidak valid → tampil pesan error dan ulangi

### 2. LOGIN

- User input username & password
- Verifikasi:
  - Jika Admin → arahkan ke Menu Admin
  - Jika User Biasa → arahkan ke Menu User
  - Jika salah → tampil "Username atau password salah"

### 3. REGISTER

- Input nama baru & password
- Cek duplikasi username
- Jika tersedia → tambah ke database users
- Tampil "Registrasi berhasil"

### 4. MENU ADMIN

Opsi yang tersedia:

- Lihat Event → tampilkan semua event
- Create Event → input data event baru + validasi status
- Update Event → edit data event tertentu
- Delete Event → hapus event dari sistem
- Kelola Peserta → tambah/hapus peserta event
- Logout → kembali ke Menu Utama

### 5. MENU USER BIASA

Opsi yang tersedia:

- Lihat Event → tampilkan daftar event
- Daftar Event → mendaftar ke event ongoing
- Logout → kembali ke Menu Utama

### 6. GUEST MODE

- Lihat Event saja tanpa bisa daftar
- Kembali ke Menu Utama

### 7. VALIDASI & PENGULANGAN

- Setiap aksi selesai → kembali ke menu sebelumnya

- Input tidak valid → minta input ulang
- Event tidak ditemukan → tampil pesan error

## 8. END

- User pilih Keluar → tampil pesan penutup → Program selesai

## 2. Deskripsi Singkat Program

Program ini merupakan sistem manajemen event Web3 dan Hackathon berbasis terminal yang memungkinkan pengguna untuk melakukan login, registrasi, serta mengelola data event. Program ini membedakan antara akun admin dan user biasa: admin dapat menambah, melihat, memperbarui, menghapus event, serta mengelola peserta, sedangkan user biasa hanya dapat melihat daftar event dan mendaftar sebagai peserta pada event yang sedang berlangsung.

Data admin, pengguna, dan event disimpan dalam struktur data Python sederhana menggunakan dictionary dan list of dictionaries. Antarmuka program disajikan melalui menu berbasis teks yang interaktif dengan perintah `os.system('cls'/'clear')` untuk membersihkan layar. Program berjalan dalam loop utama hingga pengguna memilih untuk keluar dari sistem.

## 3. Source Code

### A. Inisialisasi Data

```
import os
import sys
admin = {
    "Nuril Akmal": "admin123",
    "Akmal Ganteng": "admin123",
    "admin": "admin123"
}
users = {
    "Bang Pernanda": "123",
    "Mba Triya": "321",
    "pengguna": "user123"
}
events = [
    {
        "id": 1,
        "nama": "ETH Hackathon Samarinda",
        "hadiah": "10.000 USDT + NFT",
        "status": "ongoing",
        "peserta": ["Team Alpha", "Team Beta"]
    },

```

## B. Menu Utama

```
while True:
    bersihkan_layar()
    print("=" * 60)
    print("SISTEM MANAJEMEN EVENT WEB3 & HACKATHON".center(60))
    print("=" * 60)
    print("1. Login")
    print("2. Register")
    print("3. Lihat Event (Guest)")
    print("4. Keluar")
    print("=" * 60)

    pilihan = input("Pilih opsi (1-4): ")
```

## C. Fitur Login

```
if pilihan == '1':
    bersihkan_layar()
    print("=== LOGIN ===\n")

    try:
        user = input("Username: ")
        pwd = input("Password: ")

        # Cek user biasa
        if user in users and users[user] == pwd:
            print(f"\nLogin berhasil! Selamat datang {user}!")
            input("Tekan Enter...")
            menu_user(user)

        # Cek admin
        elif user in admin and admin[user] == pwd:
            print(f"\nLogin berhasil! Selamat datang Admin {user}!")
            input("Tekan Enter...")
            menu_admin(user)

        # Login gagal
        else:
            print("\nUsername atau password salah!")
            input("Tekan Enter...")
    except:
        print("Terjadi error saat login")
        input("Tekan Enter...")
```

## D. Fitur Registrasi

```
# Menu 2: Register
```

```

elif pilihan == '2':
    bersihkan_layar()
    print("=== REGISTER USER BARU ===\n")

    try:
        nama_baru = input("Username baru: ")

        if nama_baru == "":
            print("Username tidak boleh kosong!")
            input("Tekan Enter...")
            continue

        if nama_baru in admin or nama_baru in users:
            print("Username sudah dipakai! Coba yang lain.")
            input("Tekan Enter...")
            continue

        pwd_baru = input("Password baru: ")

        if pwd_baru == "":
            print("Password tidak boleh kosong!")
            input("Tekan Enter...")
            continue

        if len(pwd_baru) < 3:
            print("Password minimal 3 karakter!")
            input("Tekan Enter...")
            continue

        users[nama_baru] = pwd_baru
        print(f"\nRegistrasi berhasil! Silakan login dengan username '{nama_baru}'")
        input("Tekan Enter...")
    except:
        print("Terjadi error saat register")
        input("Tekan Enter...")

```

#### E. Lihat Event (Guest)

```

# Menu 3: Lihat event sebagai guest
elif pilihan == '3':
    bersihkan_layar()
    tampilkan_semua_event()

    try:
        total_peserta = hitung_peserta()
        print(f"\nTotal peserta di semua event: {total_peserta}")
    except:

```



```
print("Error hitung peserta")

input("\nTekan Enter...")
```

#### F. Keluar dari Program

```
# Menu 4: Keluar
elif pilihan == '4':
    bersihkan_layar()
    print("Terima kasih sudah pakai sistem ini!")
    print("Sampai jumpa!")
    sys.exit()

else:
    print("\nPilihan salah! Coba lagi.")
    input("Tekan Enter...")
```

#### G. Login sebagai User

```
# ===== MENU USER BIASA =====
def menu_user(nama_user):
    """Menu untuk user biasa"""

    while True:
        bersihkan_layar()
        print(f"\nSelamat datang, {nama_user}!")
        print("=" * 60)
        print("1. Lihat Event")
        print("2. Daftar Event")
        print("3. Logout")
        print("=" * 60)

        pilih = input("Pilih menu (1-3): ")
```

#### H. Sub Fitur User

##### 1. Lihat Event

```
def tampilkan_semua_event():
    """Prosedur untuk tampilkan semua event"""
    try:
        print("\nDaftar Event:")
        print("=" * 80)

        if len(events) == 0:
            print("Tidak ada event")
            return
```

```

    for ev in events:
        print(f"ID: {ev['id']}")
        print(f>Nama: {ev['nama']}")
        print(f"Hadiah: {ev['hadiah']}")
        print(f>Status: {ev['status']}")

        if len(ev['peserta']) > 0:
            print(f"Peserta: {' '.join(ev['peserta'])}")
        else:
            print("Peserta: Belum ada")

    print("-" * 80)
except:
    print("Error tampilkan event")
if pilih == '1':
    bersihkan_layar()
    tampilkan_semua_event()
    input("\nTekan Enter untuk kembali...")

```

## 2. Daftar Event

```

# Menu 2: Daftar event
elif pilih == '2':
    bersihkan_layar()
    print("\nEvent yang bisa diikuti:")
    print("-" * 60)

    # Cari event yang ongoing
    ada_event = False
    for ev in events:
        if ev['status'] == 'ongoing':
            ada_event = True
            print(f"ID: {ev['id']}, Nama: {ev['nama']}, Hadiah: {ev['hadiah']}")

    if not ada_event:
        print("Tidak ada event yang bisa diikuti")
        input("\nTekan Enter untuk kembali...")
        continue

    print("-" * 60)

    try:
        id_pilih = cek_angka("Masukkan ID event (0 untuk batal): ",
0)

```

```

        if id_pilih == 0:
            continue

        event_dipilih = cari_event(id_pilih, "ongoing")

        if event_dipilih != None:
            if nama_user in event_dipilih['peserta']:
                print("Kamu sudah terdaftar di event ini!")
            else:
                event_dipilih['peserta'].append(nama_user)
                print(f"Berhasil daftar ke event {event_dipilih['nama']}!")
            else:
                print("Event tidak ditemukan atau sudah selesai")
        except:
            print("Terjadi error")

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

```

### 3. Logout

```

# Menu 3: Logout
elif pilih == '3':
    print("Logout berhasil")
    input("\nTekan Enter...")
    break

else:
    print("Pilihan salah!")
    input("\nTekan Enter...")

```

#### I. Login sebagai Admin & Menu Admin

```

# ===== MENU ADMIN =====
def menu_admin(nama_admin):
    """Menu untuk admin"""

    while True:
        bersihkan_layar()
        print(f"\nSelamat datang Admin {nama_admin}!")
        print("=" * 60)
        print("1. Lihat Event")
        print("2. Tambah Event")
        print("3. Edit Event")
        print("4. Hapus Event")

```

## J. Sub Fitur Admin

### 1. Lihat Event

```
def tampilkan_semua_event():
    """Prosedur untuk tampilkan semua event"""
    try:
        print("\nDaftar Event:")
        print("=" * 80)

        if len(events) == 0:
            print("Tidak ada event")
            return

        for ev in events:
            print(f"ID: {ev['id']}")
            print(f>Nama: {ev['nama']}")
            print(f>Hadiah: {ev['hadiah']}")
            print(f>Status: {ev['status']}")

            if len(ev['peserta']) > 0:
                print(f>Peserta: {' '.join(ev['peserta'])}")
            else:
                print("Peserta: Belum ada")

            print("-" * 80)
    except:
        print("Error tampilkan event")
    if pilih == '1':
        bersihkan_layar()
        tampilkan_semua_event()
        input("\nTekan Enter untuk kembali...")
```

### 2. Creat Event

```
elif pilih == '2':
    bersihkan_layar()
    print("\n=== TAMBAH EVENT BARU ===")

    try:
        nama = input("Nama event: ")
        if nama == "":
            print("Nama tidak boleh kosong!")
```

```

        input("\nTekan Enter...")
        continue

    hadiah = input("Hadiah event: ")
    if hadiah == "":
        print("Hadiah tidak boleh kosong!")
        input("\nTekan Enter...")
        continue

    while True:
        status = input("Status (ongoing/finished): ").lower()
        if status == "ongoing" or status == "finished":
            break
        else:
            print("Status harus ongoing atau finished!")

    id_baru = bikin_id_baru()

    events.append({
        "id": id_baru,
        "nama": nama,
        "hadiah": hadiah,
        "status": status,
        "peserta": []
    })

    print(f"\nEvent berhasil ditambahkan dengan ID {id_baru}!")
except:
    print("Terjadi error saat tambah event")

input("\nTekan Enter...")

```

### 3. Update Event

```

# Menu 3: Edit event
elif pilih == '3':
    bersihkan_layar()
    tampilkan_semua_event()

    try:
        id_edit = cek_angka("\nMasukkan ID event yang mau diedit (0
untuk batal): ", 0)

        if id_edit == 0:
            continue

        event_edit = cari_event(id_edit)

```

```

        if event_edit != None:
            print(f"\nEvent sekarang: {event_edit['nama']}")

            nama_baru = input("Nama baru: ")
            hadiah_baru = input("Hadiah baru: ")

            while True:
                status_baru = input("Status baru (ongoing/finished): ").lower()

                if status_baru == "ongoing" or status_baru == "finished":
                    break
                else:
                    print("Status harus ongoing atau finished!")

            if nama_baru != "":
                event_edit['nama'] = nama_baru
            if hadiah_baru != "":
                event_edit['hadiah'] = hadiah_baru
            event_edit['status'] = status_baru

            print("\nEvent berhasil diedit!")
        else:
            print("\nEvent tidak ditemukan!")
    except:
        print("Terjadi error saat edit event")

    input("\nTekan Enter...")

```

#### 4. Delete Event

```

# Menu 4: Hapus event
elif pilih == '4':
    bersihkan_layar()
    tampilkan_semua_event()

    try:
        id_hapus = cek_angka("\nMasukkan ID event yang mau dihapus (0 untuk batal): ", 0)

        if id_hapus == 0:
            continue

        event_hapus = cari_event(id_hapus)

        if event_hapus != None:
            konfirmasi = input(f"Yakin mau hapus

```

```

'{event_hapus['nama']}'? (y/n): ")
    if konfirmasi.lower() == 'y':
        events.remove(event_hapus)
        print("\nEvent berhasil dihapus!")
    else:
        print("\nBatal hapus event")
else:
    print("\nEvent tidak ditemukan!")
except:
    print("Terjadi error saat hapus event")

```

## 5. Kelola Peserta

```

# Menu 5: Kelola peserta
elif pilih == '5':
    bersihkan_layar()
    tampilkan_semua_event()

    try:
        id_peserta = cek_angka("\nMasukkan ID event (0 untuk batal):", 0)

        if id_peserta == 0:
            continue

        event_peserta = cari_event(id_peserta)

        if event_peserta != None:
            print(f"\nEvent: {event_peserta['nama']}")
            if len(event_peserta['peserta']) > 0:
                print(f"Peserta sekarang: {'',
'.join(event_peserta['peserta'])}")
            else:
                print("Peserta sekarang: Belum ada")

            print("\n1. Tambah Peserta")
            print("2. Hapus Peserta")
            sub = input("Pilih (1-2): ")

            if sub == '1':
                nama_peserta = input("Nama peserta: ")
                if nama_peserta != "":
                    if nama_peserta not in event_peserta['peserta']:
                        event_peserta['peserta'].append(nama_peserta)
                        print(f"Peserta {nama_peserta} berhasil
ditambah!")

```

```

        else:
            print("Peserta sudah terdaftar!")
        else:
            print("Nama tidak boleh kosong!")

    elif sub == '2':
        nama_peserta = input("Nama peserta yang mau dihapus:
")

        if nama_peserta in event_peserta['peserta']:
            event_peserta['peserta'].remove(nama_peserta)
            print(f"Peserta {nama_peserta} berhasil
dihapus!")
        else:
            print("Peserta tidak ditemukan!")
    else:
        print("Pilihan salah!")
    else:
        print("\nEvent tidak ditemukan!")
except:
    print("Terjadi error")

input("\nTekan Enter...")

```

#### 6. Logout

```

# Menu 6: Logout
elif pilih == '6':
    print("Logout berhasil")
    input("\nTekan Enter...")
    break
else:
    print("Pilihan salah!")
    input("\nTekan Enter...")

```

## 4. Hasil Output

```

=====
                SISTEM MANAJEMEN EVENT WEB3 & HACKATHON
=====
1. Login
2. Register
3. Lihat Event (Guest)
4. Keluar
Pilih opsi (1-4): █

```

Gambar 4.1 Menu



```
Masukkan username: Bang Pernanda
Masukkan password: 123
Selamat datang, Bang Pernanda!
Tekan Enter untuk melanjutkan...
```

Gambar 4.2 Login Sebagai User(Peserta)

```
Menu Pengguna - Bang Pernanda:
1. Lihat Event
2. Daftar Event
3. Logout
Pilih menu (1-3):
```

Gambar 4.3 Menu User

```
Daftar Event yang Tersedia:
ID: 1, Nama: ETH Hackathon Samarinda, Hadiah: 10.000 USDT + NFT
Masukkan ID event yang ingin diikuti: 1
Berhasil mendaftar ke event ETH Hackathon Samarinda!

Tekan Enter untuk kembali ke menu...
```

Gambar 4.4 Daftar Event

```
Selamat datang ADMIN!

Menu:
1. Lihat Event
2. Create Event
3. Update Event
4. Delete Event
5. Kelola Peserta Event
6. Logout
Pilih menu (1-6):
```

Gambar 4.5 Login Sebagai Admin

```
Daftar Event:
ID: 1, Nama: ETH Hackathon Samarinda, Hadiah: 10.000 USDT + NFT, Status: ongoing, Peserta: Team Alpha, Team Beta
ID: 2, Nama: Solana DeFi Camp, Hadiah: 5.000 USDT, Status: finished, Peserta: DeFi Ninjas

Tekan Enter untuk kembali ke menu...
```

Gambar 4.6 Lihat Event

```
Masukkan nama event: Solona Run
Masukkan hadiah event: 100 USD
Masukkan status event (ongoing/finished): ongoing
Event berhasil ditambahkan!

Tekan Enter untuk kembali ke menu...
```

Gambar 4.7 Creat Event

```
Masukkan nama event baru: Etharium Atlas Beach
Masukkan hadiah event baru: 200 USD
Masukkan status event baru (ongoing/finished): ongaing
Event berhasil diupdate!

Tekan Enter untuk kembali ke menu...
```

Gambar 4.8 Update Event

```
Masukkan ID event untuk kelola peserta: 2
Peserta saat ini: DeFi Ninjas
1. Tambah Peserta
2. Hapus Peserta
Pilih opsi (1-2): 1
Masukkan nama peserta: Nibung Guysss
Peserta berhasil ditambahkan!

Tekan Enter untuk kembali ke menu...
```

Gambar 4.9 Update Peserta

```
Masukkan ID event yang ingin dihapus: 1
Event berhasil dihapus!

Tekan Enter untuk kembali ke menu...
```

Gambar 4.10 Delete Peserta

```
Masukkan nama baru: Dapupu teman Jarvis
Masukkan password baru: 123
Registrasi berhasil! Silakan login.
Tekan Enter untuk melanjutkan...
```

Gambar 4.11 Register

```
Daftar Event:
ID: 2, Nama: Etharium Atlas Beach, Hadiah: 200 USD , Status: ongaing, Peserta: DeFi Ninjas, Nibung Guysss
ID: 3, Nama: Solona Run, Hadiah: 100 USD , Status: ongoing, Peserta:

Tekan Enter untuk kembali ke menu...S
```

Gambar 4.12 Lihat Event (Guest)

```
Terima kasih telah menggunakan sistem ini.
PS C:\Users\Acer\Documents\PRAKTIKUM APD\github\PRAKTIKUM_APD_B2_25\PRAKTIKUM_APD_B2-25>
```

Gambar 4.13 Logout

## 5. Langkah-langkah GIT

### 1. GIT Add

```
=====
PS C:\Users\Acer\Documents\PRAKTIKUM APD\github\PRAKTIKUM_APD_B2_25\PRAKTIKUM_APD_B2-25> git init
Reinitialized existing Git repository in C:/Users/Acer/Documents/PRAKTIKUM APD/github/PRAKTIKUM APD B2 25/PRAKTIKUM APD B2-25/.git/
PS C:\Users\Acer\Documents\PRAKTIKUM APD\github\PRAKTIKUM_APD_B2_25\PRAKTIKUM_APD_B2-25> git add .
PS C:\Users\Acer\Documents\PRAKTIKUM APD\github\PRAKTIKUM_APD_B2_25\PRAKTIKUM_APD_B2-25>
```

Gambar Git Add 5.1

Kegunaan Git add adalah setelah kita mengubah atau menambahkan file, kita perlu “menandai” file itu agar siap disimpan ke Git. Perintah git add berfungsi untuk memasukkan file atau perubahan ke dalam staging area, yaitu tempat sementara sebelum benar-benar disimpan permanen ke riwayat proyek.

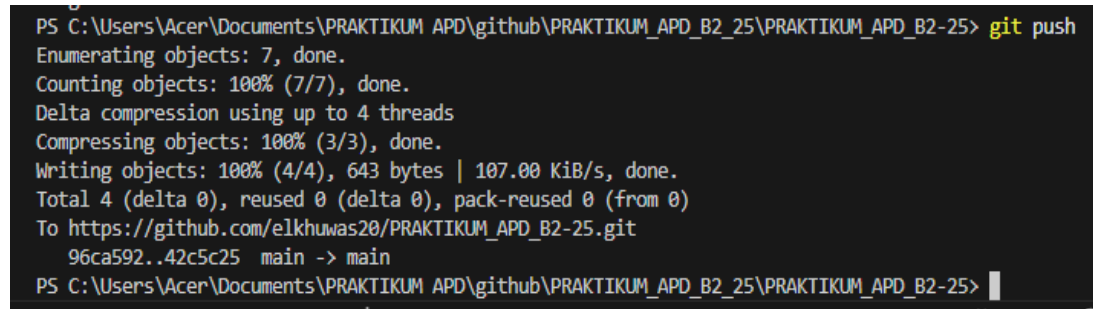
### 2. GIT Commit

```
PS C:\Users\Acer\Documents\PRAKTIKUM APD\github\PRAKTIKUM_APD_B2_25\PRAKTIKUM_APD_B2-25> git add .
PS C:\Users\Acer\Documents\PRAKTIKUM APD\github\PRAKTIKUM_APD_B2_25\PRAKTIKUM_APD_B2-25>
```

Gambar Git Commit 5.2

Adapun Git Commit kegunaannya adalah melakukan perubahan sudah ditandai dengan git add, maka git commit dipakai untuk benar-benar menyimpan perubahan itu ke riwayat proyek Git. Setiap commit biasanya disertai pesan (commit message) yang menjelaskan apa yang diubah, supaya lebih mudah dilacak di kemudian hari.

### 3. GIT Push



```
PS C:\Users\Acer\Documents\PRAKTIKUM APD\github\PRAKTIKUM_APD_B2_25\PRAKTIKUM_APD_B2-25> git push
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 643 bytes | 107.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/elkhuwas20/PRAKTIKUM_APD_B2-25.git
 96ca592..42c5c25  main -> main
PS C:\Users\Acer\Documents\PRAKTIKUM APD\github\PRAKTIKUM_APD_B2_25\PRAKTIKUM_APD_B2-25>
```

Gambar Git Push 5.3

Setelah kita punya repository lokal yang sudah terhubung dengan remote, kita bisa mengirim commit yang ada di komputer ke repository online dengan git push. Dengan begitu, semua perubahan akan tersimpan di server dan bisa dilihat atau digunakan oleh orang lain (atau oleh kita sendiri di komputer lain).