

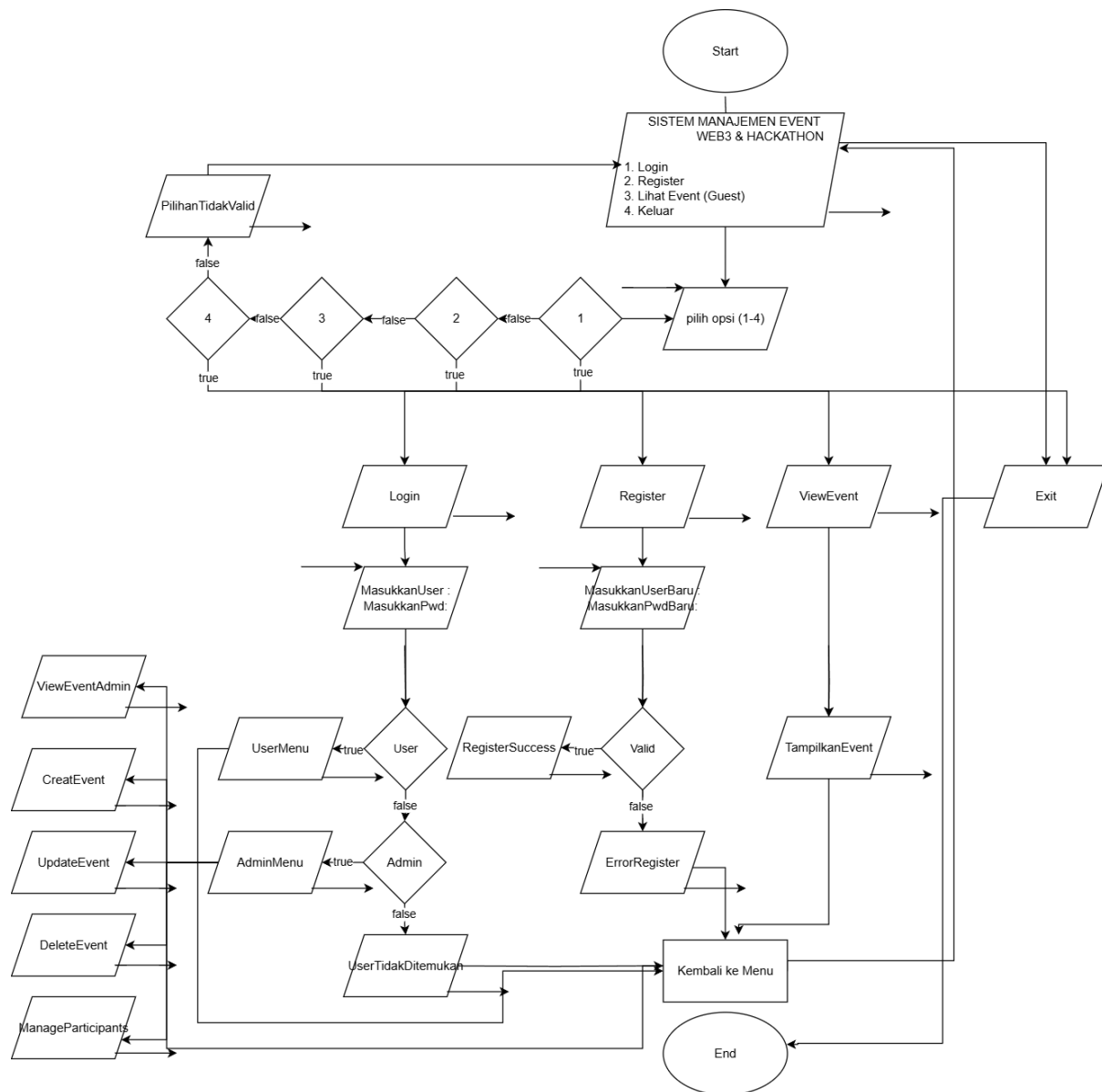
LAPORAN PRAKTIKUM
POSTTEST 5
ALGORITMA PEMROGRAMAN DASAR



Disusun oleh:
Nuril Akmal 2509106074
Kelas B2'25

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

1. Flowcart



Gambar 1.1 Flowcart

2. Deskripsi Singkat Program

Alur Program Program:

Dimulai dari titik "start", pengguna memilih tindakan melalui keputusan "pilihtindakvalid?".

- Pilihan yang tersedia: login, register, lihat event (guest), atau keluar.

Login flow (alur login):

- Setelah pengguna memilih login, sistem melakukan proses validasi kredensial.
- Jika login gagal, sistem menampilkan pesan error dan mengarahkan pengguna kembali ke menu utama.
- Jika login berhasil, sistem mengecek jenis pengguna:
- Jika pengguna adalah admin, maka diarahkan ke adminmenu yang berisi:
 1. Vieweventadmin
 2. Createevent
 3. Updateevent
 4. Deleteevent
 5. Manageparticipant
- Jika pengguna adalah user biasa, maka diarahkan ke usermenu dengan akses terbatas (kemungkinan hanya bisa melihat event atau mendaftar sebagai peserta).

Register:

- Proses pendaftaran dilakukan, lalu muncul konfirmasi sukses.

Lihat event (guest):

- Pengguna dapat melihat event tanpa login.

Keluar:

- Langsung menuju akhir proses.

Akhir proses:

- Semua alur berujung pada "kembali ke menu" atau "end".

3. Source Code

A. Inisialisasi Data

```
import os
import sys
admin = ["Nuril Akmal", "Akmal Ganteng", "admin"]
daftaruser = ["Bang Pernanda", "Mba Triya", "pengguna"]
pwd_user = ["123", "321", "user123"]
password = ["123", "321", "admin123"]
events = [
    [1, "ETH Hackathon Samarinda", "10.000 USDT + NFT", "ongoing", ["Team Alpha", "Team Beta"]],
    [2, "Solana DeFi Camp", "5.000 USDT", "finished", ["DeFi Ninjas"]],
]
data = {}
index = 0
```

B. Menu Utama

```
while True:
    os.system('cls' if os.name == 'nt' else 'clear')
    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")
    pilihan = input("Pilih opsi (1-4): ")
```

C. Fitur Login

```
elif pilihan == '1':
    os.system('cls' if os.name == 'nt' else 'clear')
    user = input("Masukkan username: ")
    pwd = input("Masukkan password: ")
    if user in daftaruser and pwd in pwd_user:
        if daftaruser.index(user) == pwd_user.index(pwd):
            print(f"Selamat datang, {user}!")
            input("Tekan Enter untuk melanjutkan...")
            continue
        else:
            print("Username dan password tidak sesuai.")
            input("Tekan Enter untuk mencoba lagi...")
            continue
```

D. Fitur Registrasi

```
elif pilihan == '2':
    os.system('cls' if os.name == 'nt' else 'clear')
    nama_baru = input("Masukkan nama baru: ")
    if nama_baru in admin or nama_baru in daftaruser:
        print("Nama sudah terdaftar. Silakan coba lagi.")
        input("Tekan Enter untuk melanjutkan...")
        continue
    pwd_baru = input("Masukkan password baru: ")
    daftaruser.append(nama_baru)
    pwd_user.append(pwd_baru)
    print("Registrasi berhasil! Silakan login.")
    input("Tekan Enter untuk melanjutkan...")
    continue
```

E. Lihat Event (Guest)

```
elif pilihan == '3':
    os.system('cls' if os.name == 'nt' else 'clear')
    print("\nDaftar Event:")
    for event in events:
        print(f"ID: {event[0]}, Nama: {event[1]}, Hadiah: {event[2]}, Status: {event[3]}, Peserta: {'', '.join(event[4])}")
    input("\nTekan Enter untuk kembali ke menu...")
    continue
```

F. Keluar dari Program

```
if pilihan == '4':
    os.system('cls' if os.name == 'nt' else 'clear')
    print("Terima kasih telah menggunakan sistem ini.")

    sys.exit()
```

G. Login sebagai Admin & Menu Admin

```
if user in admin and pwd in password:
    os.system('cls' if os.name == 'nt' else 'clear')
    if admin.index(user) == password.index(pwd):
        print(f"Selamat datang, {admin}!")
        while True:
            print("\nMenu:")
            print("1. Lihat Event")
            print("2. Create Event")
            print("3. Update Event")
            print("4. Delete Event")
            print("5. Kelola Peserta Event")
            print("6. Logout")
            choice = input("Pilih menu (1-6): ")
```

H. Sub Fitur Admin

1. Lihat Event

```
if choice == '1':
    os.system('cls' if os.name == 'nt' else 'clear')
    print("\nDaftar Event:")
    for event in events:
        print(f"ID: {event[0]}, Nama: {event[1]}, Hadiah: {event[2]}, Status: {event[3]}, Peserta: {'', '.join(event[4])}")
    input("\nTekan Enter untuk kembali ke menu...")
```

2. Creat Event

```
elif choice == '2':
    os.system('cls' if os.name == 'nt' else 'clear')
    nama_event = input("Masukkan nama event: ")
    hadiah_event = input("Masukkan hadiah event: ")
    status_event = input("Masukkan status event (ongoing/finished): ")
    events.append([len(events)+1, nama_event, hadiah_event, status_event, []])
    print("Event berhasil ditambahkan!")
    input("\nTekan Enter untuk kembali ke menu...")
```

3. Update Event

```
elif choice == '3':
    os.system('cls' if os.name == 'nt' else 'clear')
    event_id = int(input("Masukkan ID event yang ingin diupdate: "))
    for event in events:
        os.system('cls' if os.name == 'nt' else 'clear')
        if event[0] == event_id:
            nama_event = input("Masukkan nama event baru: ")
            hadiah_event = input("Masukkan hadiah event baru: ")
            status_event = input("Masukkan status event baru (ongoing/finished): ")
            event[1] = nama_event
            event[2] = hadiah_event
            event[3] = status_event
            print("Event berhasil diupdate!")
            break
    else:
```

```

        os.system('cls' if os.name == 'nt' else 'clear')
        print("Event dengan ID tersebut tidak ditemukan.")
        input("\nTekan Enter untuk kembali ke menu...")

```

4. Delete Event

```

elif choice == '4':
    os.system('cls' if os.name == 'nt' else 'clear')
    event_id = int(input("Masukkan ID event yang ingin
dihapus: "))

    for event in events:

        if event[0] == event_id:
            events.remove(event)
            print("Event berhasil dihapus!")
            break
        else:

            print("Event dengan ID tersebut tidak ditemukan.")
            input("\nTekan Enter untuk kembali ke menu...")

```

5. Kelola Peserta

```

elif choice == '5':
    os.system('cls' if os.name == 'nt' else 'clear')
    event_id = int(input("Masukkan ID event untuk kelola
peserta: "))

    for event in events:
        if event[0] == event_id:
            print(f"Peserta saat ini: {' '.join(event[4])}
if event[4] else 'Tidak ada peserta'")
            print("1. Tambah Peserta")
            print("2. Hapus Peserta")
            sub_choice = input("Pilih opsi (1-2): ")
            if sub_choice == '1':
                nama_peserta = input("Masukkan nama peserta:
")

                event[4].append(nama_peserta)
                print("Peserta berhasil ditambahkan!")
            elif sub_choice == '2':
                nama_peserta = input("Masukkan nama peserta
yang ingin dihapus: ")

                if nama_peserta in event[4]:
                    event[4].remove(nama_peserta)
                    print("Peserta berhasil dihapus!")
                else:

```

```

                print("Peserta tidak ditemukan.")
            else:
                print("Opsi tidak valid.")
                break
        else:
            os.system('cls' if os.name == 'nt' else 'clear')
            print("Event dengan ID tersebut tidak ditemukan.")
            input("\nTekan Enter untuk kembali ke menu...")

```

6. Logout

```

elif choice == '6':
    os.system('cls' if os.name == 'nt' else 'clear')
    print("Logout berhasil.")
    break
else:
    os.system('cls' if os.name == 'nt' else 'clear')
    print("Opsi tidak valid. Silakan coba lagi.")
    input("\nTekan Enter untuk kembali ke menu...")
else:
    os.system('cls' if os.name == 'nt' else 'clear')
    print("Username dan password tidak sesuai.")
    input("Tekan Enter untuk mencoba lagi...")
else:
    os.system('cls' if os.name == 'nt' else 'clear')
    print("Username atau password salah.")
    input("Tekan Enter untuk mencoba lagi...")

```


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)

```
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 4.3 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 4.4 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 4.5 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 4.6 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...
```

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

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

Gambar 4.7 Update/Delete Peserta

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

Gambar 4.8 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 4.9 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 4.10 Logout

5. Langkah-langkah GIT

1. GIT Init

```
=====
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> █
```

Gambar Git Init 5.1

Git Ini adalah Perintah ini digunakan untuk membuat repository Git baru di folder yang sedang kita buka. Dengan kata lain, folder biasa akan “diubah” menjadi folder yang bisa dilacak perubahannya oleh Git. Biasanya ini adalah langkah pertama ketika kita mulai proyek baru.

2. GIT Add

```
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 5.2

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.

3. 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 5.3

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.

4. GIT Remote

```
PS C:\Users\Acer\Documents\PRAKTIKUM APD\github\PRAKTIKUM_APD_B2_25\PRAKTIKUM_APD_B2-25> git remote origin
PS C:\Users\Acer\Documents\PRAKTIKUM APD\github\PRAKTIKUM_APD_B2_25\PRAKTIKUM_APD_B2-25> |
```

Gambar Git Remote 5.4

Dan adapun Git Remote kegunaannya adalah merepository Git yang kita buat tadi hanya ada di komputer lokal. Supaya bisa terhubung dengan repository yang ada di internet (misalnya di GitHub, GitLab, atau Bitbucket), kita perlu menghubungkannya dengan git remote. Jadi, perintah ini semacam “menyambungkan” repository lokal dengan repository online.

5. 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 5.5

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).