

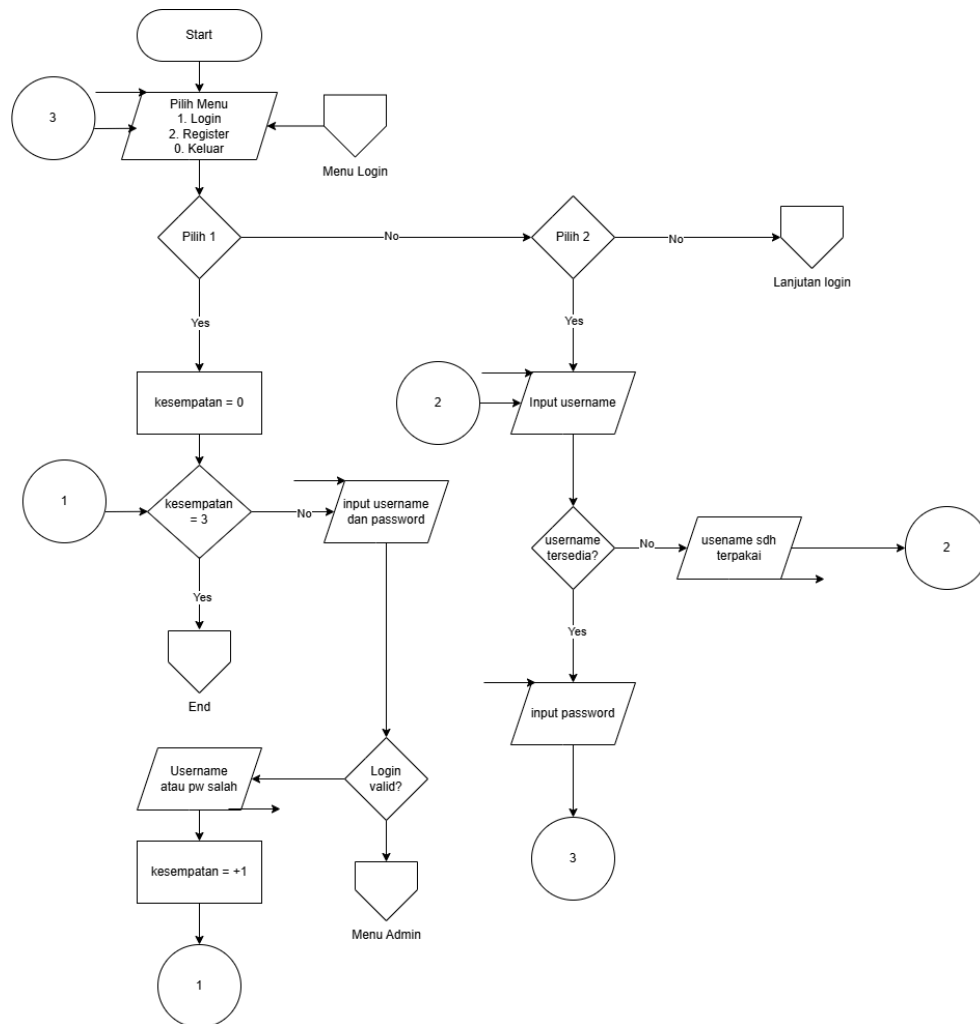
LAPORAN PRAKTIKUM
POSTTEST 6
ALGORITMA PEMROGRAMAN DASAR



Disusun oleh:
Nur Azizah Islamiyah (2509106018)
Kelas (A1 '25)

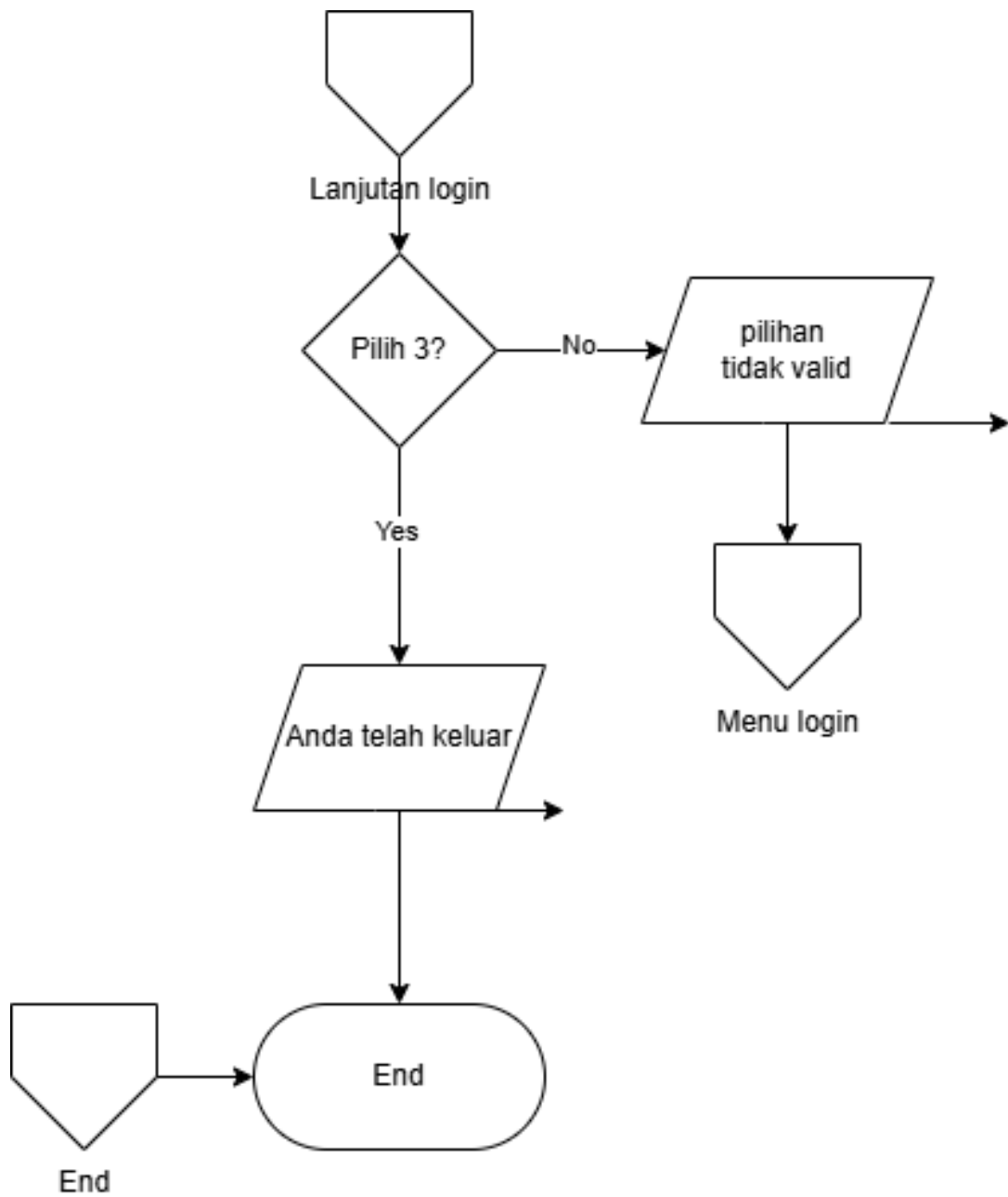
PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

1. Flowchart



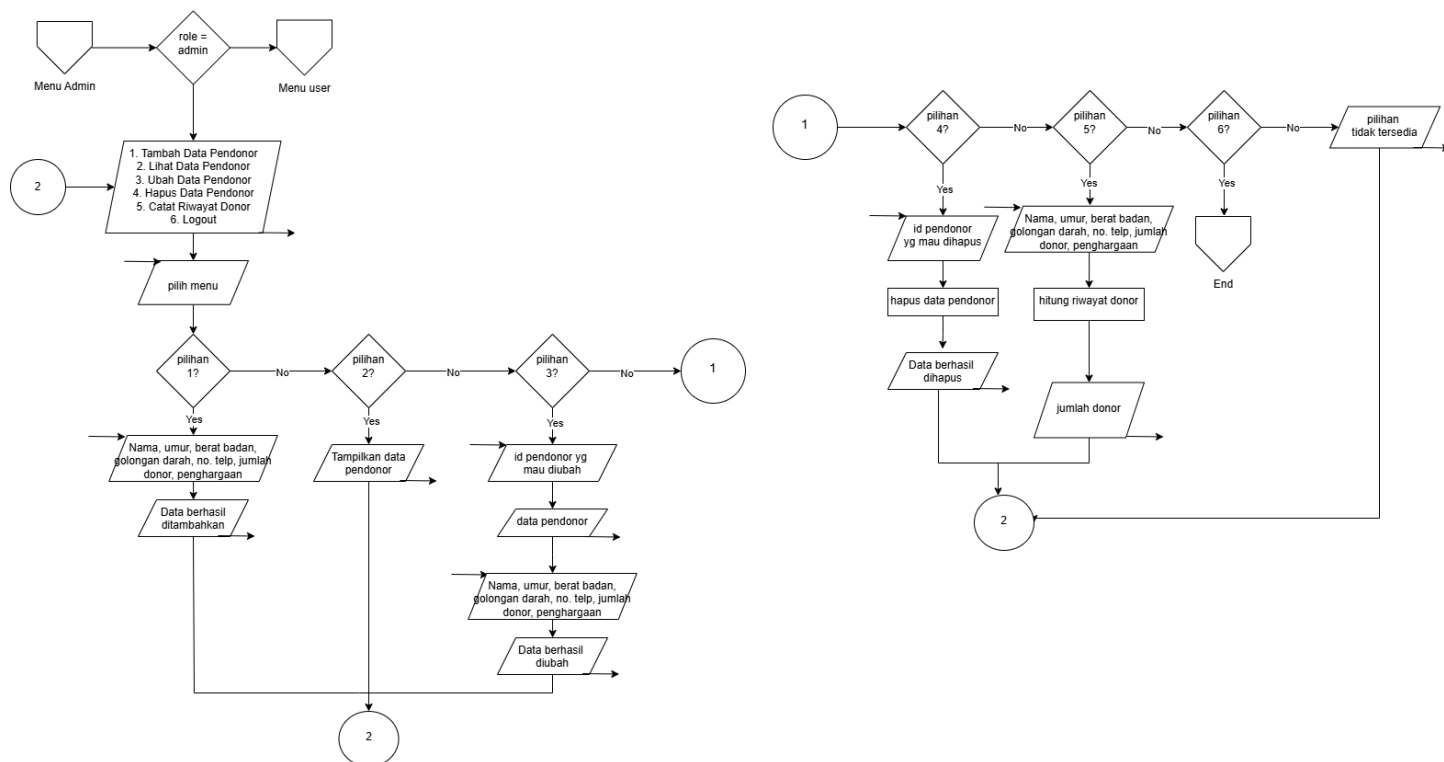
Gambar 1.1 Flowchart Login dan Menu

Pertama, program akan menampilkan menu login, pengguna diminta untuk memilih menu. Jika pengguna memilih menu 1, maka pengguna login dan menginput sesuai dengan yang data yang ada di sitem. Jika inputan username dan password tidak sesuai, login dinyatakan gagal dan pengguna diberi kesempatan 3 kali untuk mencoba login. Jika login berhasil program akan kembali ke menu utama. Namun jika pengguna memilih menu 2, maka pengguna diminta melakukan input username dan password, untuk username program akan memeriksa apakah username tersedia untuk menghindari duplikasi.



Gambar 1.2 Flowchart Menu 3

Jika pengguna memilih menu 3, maka pengguna akan keluar dari program.

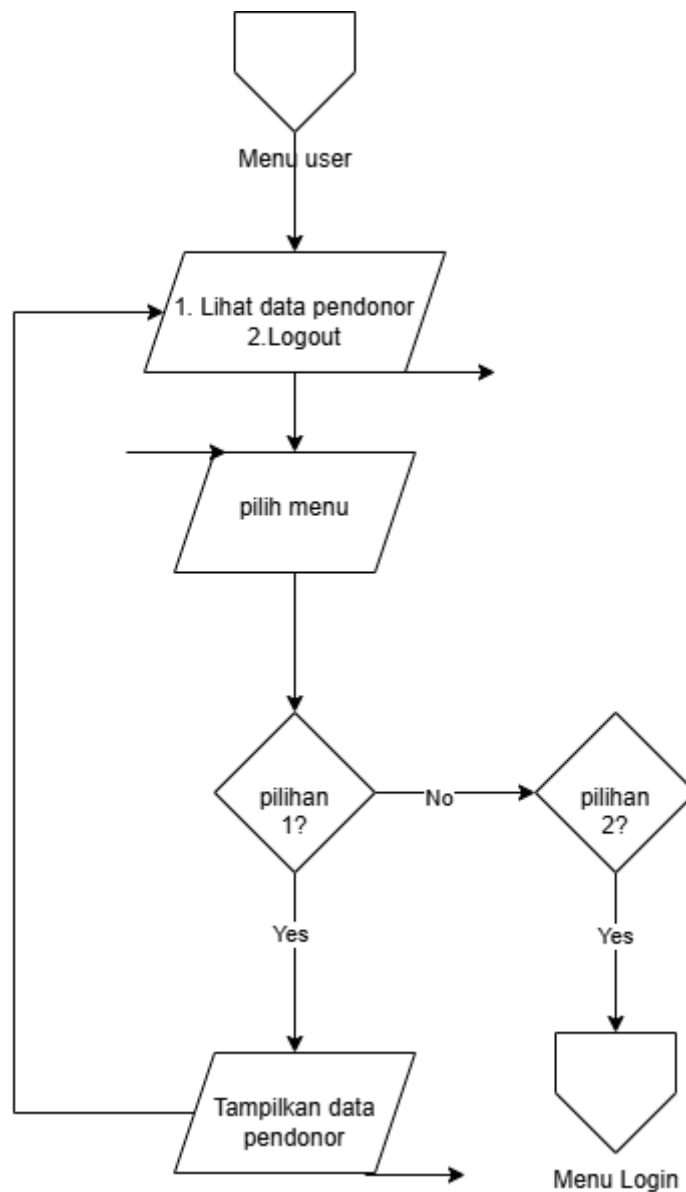


Gambar 1.3 Flowchart Role Admin

Setelah login, program akan mengecek apakah pengguna login sebagai admin atau pengguna biasa, jika sebagai admin maka menu yang ditampilkan oleh program yaitu menu admin. Pada menu 1, admin bisa menambahkan data pendonor berupa penginputan Nama, Umur; Berat Badan, Golongan Darah, No. Telepon. Setelah itu program akan menampilkan output “Data berhasil ditambahkan” dan data pendonor memulai jumlah donor dari 0 kali.

Jika admin memilih menu 2, maka program akan menampilkan data pendonor dalam bentuk tabel. Namun, jika admin memilih menu 3 program akan meminta admin untuk menginput ID pendonor untuk diubah datanya. Admin diminta untuk menginput seperti menu pertama, jika pengguna tidak melakukan input maka data yang dipakai adalah data sebelumnya.

Jika admin memilih menu 4, maka program akan meminta admin untuk menginput ID pendonor untuk dihapus datanya. Jika admin memilih menu 5, program akan mencatat riwayat donor darah setiap pendonor selesai melakukan donor dengan menginput ID pendonor. Jika admin memilih menu 6, program akan logout dan kembali ke menu login. Dan jika admin menginput menu selain angka 1-6, program akan memberi output “pilihan tidak tersedia”.



Gambar 1.4 Flowchart Role User Biasa

Jika pengguna login sebagai user biasa, program akan menampilkan menu user. Pada menu ini, pengguna hanya bisa melakukan tahap read yaitu hanya melihat data pendonor dan keluar. Jika pengguna memilih menu 1, program akan menampilkan data para pendonor darah dalam bentuk tabel. Namun jika pengguna memilih menu 2, maka program akan logout dan kembali ke menu login.

2. Deskripsi Singkat Program

Program ini adalah sebuah sistem yang dirancang untuk mengelola data pendonor darah. Sistem ini memiliki CRUD (Create, Read, Update, Delete) yang digunakan untuk pendataan. Program ini juga mendukung sistem multi-user, di mana hak akses dibedakan antara peran Admin dan Pengguna Biasa.

3. Source Code

A. Fitur Login

Pada fitur ini, pengguna diberikan tiga kali kesempatan untuk mencoba login. Jika gagal, program akan kembali ke menu awal.

Source Code:

```
elif pilihan_awal == '1':
    coba_login = 0
    login_berhasil = False
    while coba_login < 3:
        os.system('cls' if os.name == 'nt' else 'clear')
        print('--- LOGIN ---')
        print(f'Percobaan ke-{coba_login + 1} dari 3')
        username = input('Username : ')
        password = input('Password : ')
        if username == "" or password == "":
            print('Username dan Password tidak boleh kosong')
            coba_login += 1
            continue
        if username in akun_user:
            if akun_user[username]['password'] == password:
                user_login = username
                role_user = akun_user[username]['role']
                login_berhasil = True
                print('Login berhasil!')

                break
        coba_login += 1
        sisa_coba = 3 - coba_login
        if sisa_coba > 0:
            print(f'Username atau Password salah! Sisa percobaan:
{sisacoba}')
```

```
if login_berhasil == False:
    print('Login Gagal, kembali ke menu awal...')
```

B. Fitur Register (Daftar Akun)

Fitur ini digunakan pengguna baru untuk mendaftarkan akun terlebih dahulu. Sistem akan memeriksa apakah username sudah ada untuk menghindari duplikasi.

Source Code:

```
elif pilihan_awal == '2':
    os.system('cls' if os.name == 'nt' else 'clear')
    print('--- REGISTER ---')
    username_baru = input('Masukkan Username Baru : ')
    password_baru = input('Masukkan Password Baru : ')
    if username_baru in akun_user:
        print('Username sudah terdaftar! Silahkan coba lagi.')
    elif username_baru == "" or password_baru == "":
        print('Username dan Password tidak boleh kosong')
    else:
        akun_user[username_baru] = {'password': password_baru,
        'role': 'user'}
        print('Akun berhasil dibuat! Silahkan login.')
```

C. Fitur Create (Menambah Data)

Fitur ini digunakan untuk Admin menambahkan data pendonor baru yang meliputi ID, Nama, Umur, Berat Badan, Golongan Darah, dan Nomor Telepon.

Source Code:

```
#c
if pilihan_menu == '1':
    os.system('cls' if os.name == 'nt' else 'clear')
    print('--- TAMBAH DATA PENDONOR BARU ---') #misal tambah
    data nur bru mau donor
    nama = input('Nama Lengkap      : ')
    umur_str = input('Umur          : ')
    bb_str = input('Berat Badan (kg) : ')
    goldar = input('Golongan Darah : ')
    telepon = input('Nomor Telepon   : ')
```

```

id_terakhir += 1
donor_baru_dict = {
    'nama': nama,
    'umur': int(umur_str),
    'bb': int(bb_str),
    'goldar': goldar,
    'telepon': telepon,
    'jumlah_donor': 0
}
data_donor[id_terakhir] = donor_baru_dict
print('Data berhasil ditambahkan')
input('< kembali(0)')

```

D. Fitur Read (Lihat Data)

Pada fitur ini, Admin dan Pengguna Biasa dapat melihat seluruh daftar data pendonor yang ditampilkan dalam bentuk tabel.

Source Code:

```

elif pilihan_menu == '2':
    os.system('cls' if os.name == 'nt' else 'clear')
    print('--- DAFTAR SEMUA PENDONOR ---')
    if data_donor == {}:
        print('Belum ada data pendonor yang tersimpan.')
    else:
        print('-' * 105)
        print(f"{'ID':<4} | {'Nama':<20} | {'Umur':<5} | {'BB(kg)':<6} | {'Goldar':<8} | {'Telepon':<15} | {'Jumlah Donor':<13} | {'Penghargaan':<15}")
        print('-' * 105)
        for donor_id, donor_info in data_donor.items():
            jumlah_donor = donor_info['jumlah_donor']
            penghargaan = '-'
            if jumlah_donor >= 10: penghargaan = "Satya Lencana"
            elif jumlah_donor >= 5: penghargaan = "Medali Emas"
            elif jumlah_donor >= 1: penghargaan = "Sertifikat"
            print(f"{'donor_id':<4} | {'donor_info['nama']':<20} | {'donor_info['umur']':<5} | {'donor_info['bb']':<6} | {'donor_info['goldar']':<8} | {'donor_info['telepon']':<15} | {str(jumlah_donor) + 'x':<13} | {penghargaan:<15}")
        print('-' * 105)
        input('< kembali(0)')

```


E. Fitur Update (Ubah Data)

Di fitur ini, Admin dapat mengubah informasi dari data pendonor yang sudah ada berdasarkan ID. Jika admin tidak memasukkan input baru, data lama tidak berubah.

Source Code:

```
#u
    elif pilihan_menu == '3':
        os.system('cls' if os.name == 'nt' else 'clear')
        print('--- UBAH DATA PENDONOR ---') #misal data michael bb
        nya salah
        if data_donor == {}:
            print('Belum ada data untuk diubah.')
        else:
            print('-' * 105)
            print(f"{'ID':<4} | {'Nama':<20} | {'Umur':<5} | {'BB(kg)':<6} | {'Goldar':<8} | {'Telepon':<15} | {'Jumlah Donor':<13} | {'Penghargaan':<15}")
            print('-' * 105)
            for donor_id, donor_info in data_donor.items():
                jumlah_donor = donor_info['jumlah_donor']
                penghargaan = '-'
                if jumlah_donor >= 10: penghargaan = "Satya Lencana"
                elif jumlah_donor >= 5: penghargaan = "Medali Emas"
                elif jumlah_donor >= 1: penghargaan = "Sertifikat"
                print(f"{'donor_id':<4} | {'donor_info['nama']':<20} | {'donor_info['umur']':<5} | {'donor_info['bb']':<6} | {'donor_info['goldar']':<8} | {'donor_info['telepon']':<15} | {str(jumlah_donor) + 'x':<13} | {penghargaan:<15}")
            print('-' * 105)
            id_input = input('\nMasukkan ID data yang ingin diubah: ')

            if not id_input.isdigit():
                print('ID harus berupa angka')
            else:
                id_edit = int(id_input)
                if id_edit not in data_donor:
                    print(f'ID {id_edit} tidak ditemukan.')
                else:
                    pendonor_ditemukan = data_donor[id_edit]
                    print('(Kosongi jika tidak ingin diubah)')

                    nama_baru = input(f>Nama baru ({pendonor_ditemukan['nama']}): ")
                    umur_baru_str = input(f"Umur baru ({pendonor_ditemukan['umur']}): ")
```

```

        bb_baru_str = input(f"BB baru
({pendonor_ditemukan['bb']}): ")
        goldar_baru = input(f"Goldar baru
({pendonor_ditemukan['goldar']}): ")
        telepon_baru = input(f"Telepon baru
({pendonor_ditemukan['telepon']}): ")

        update_berhasil = True
        if update_berhasil:
            print('Data berhasil diubah!')
        input('< kembali(0)')

```

F. Fitur Delete (Hapus Data)

Admin dapat menghapus data pendonor dari sistem berdasarkan ID setelah memberikan konfirmasi ("y/n").

```

#d
elif pilihan_menu == '4':
    os.system('cls' if os.name == 'nt' else 'clear')
    print('--- HAPUS DATA PENDONOR ---') #misal hapus fred
    karna sdh meninggal (pendonor non aktif)
    if data_donor == {}:
        print('Belum ada data untuk dihapus.')
    else:
        print('-' * 105)
        print(f'{'ID':<4} | {'Nama':<20} | {'Umur':<5} |
{'BB(kg)':<6} | {'Goldar':<8} | {'Telepon':<15} | {'Jumlah Donor':<13} |
{'Penghargaan':<15}')
```

```

        print('-' * 105)
        for donor_id, donor_info in data_donor.items():
            jumlah_donor = donor_info['jumlah_donor']
            penghargaan = '-'
            if jumlah_donor >= 10: penghargaan = "Satya Lencana"
            elif jumlah_donor >= 5: penghargaan = "Medali Emas"
            elif jumlah_donor >= 1: penghargaan = "Sertifikat"
            print(f"{'donor_id':<4} | {'donor_info['nama']':<20} |
{'donor_info['umur']':<5} | {'donor_info['bb']':<6} | {'donor_info['goldar']':<8}
| {'donor_info['telepon']':<15} | {str(jumlah_donor) + 'x':<13} |
{'penghargaan':<15}")
            id_input = input('\nMasukkan ID data yang ingin dihapus:
')

            if not id_input.isdigit():
                print('ID harus berupa angka!')
            else:

```

```

        id_hapus = int(id_input)
        if id_hapus not in data_donor:
            print(f'ID {id_hapus} tidak ditemukan.')
        else:
            nama_hapus = data_donor[id_hapus]['nama']
            konfirmasi = input(f'Yakin ingin menghapus data
{nama_hapus} (ID: {id_hapus})? (y/n): ')
            if konfirmasi == 'y':
                del data_donor[id_hapus]
                print('Data berhasil dihapus')
            else:
                print('Penghapusan dibatalkan.')
        input('< kembali(0)')

```

G. Fitur Clear Terminal

Program menggunakan perintah ‘clear’ untuk membersihkan terminal setiap kali menu ditampilkan, sehingga menampilkan output yang rapi dan mudah dibaca.

Source Code:

```

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

```

4. Hasil Output

```

=====
                SISTEM CEK DONOR DARAH
=====
1. Login
2. Register
3. Keluar
-----
Pilih Opsi (1/2/3): █

```

Gambar 4.1 Output Menu Utama

```

--- HALAMAN LOGIN ---
Percobaan ke-3 dari 3
Username : nrs
Password : nrs

```

Gambar 4.2 Output Halaman Login

```

--- TAMBAH DATA PENDONOR BARU ---
Nama Lengkap      : Nur Azizah
Umur              : 18
Berat Badan (kg)  : 51
Golongan Darah    : B+
Nomor Telepon     : 081256745684

Data berhasil ditambahkan!
< kembali(0)

```

Gambar 4.3 Output Fitur Create

```

--- DAFTAR SEMUA PENDONOR ---
-----
ID   | Nama                | Umur | BB(kg) | GolDar | Telepon          | Jumlah Donor | Penghargaan
-----
1   | Fred Brailey        | 93   | 71     | O+     | 081258016876    | 135x         | Satya Lencana
2   | Cristiano Ronaldo   | 40   | 83     | AB+    | 081987654321    | 34x          | Satya Lencana
3   | Hrithik Roshan      | 51   | 77     | B-     | 081987654321    | 41x          | Satya Lencana
4   | Michael Octaviano   | 50   | 88     | O+     | 081987654321    | 11x          | Satya Lencana
5   | Diko Saputra        | 22   | 55     | B+     | 081987654321    | 4x           | Sertifikat
6   | Nur Azizah          | 18   | 51     | B+     | 081256745684    | 0x           | -
-----
< kembali(0)

```

Gambar 4.4 Output Fitur Read

```
--- UBAH DATA PENDONOR ---
-----
ID   | Nama                | Umur | BB(kg) | Goldar | Telepon          | Jumlah Donor | Penghargaan
-----
1   | Fred Brailey        | 93   | 71     | O+     | 081258016876    | 135x         | Satya Lencana
2   | Cristiano Ronaldo   | 40   | 83     | AB+    | 081987654321    | 34x          | Satya Lencana
3   | Hrithik Roshan      | 51   | 77     | B-     | 081987654321    | 41x          | Satya Lencana
4   | Michael Octaviano   | 50   | 88     | O+     | 081987654321    | 11x          | Satya Lencana
5   | Diko Saputra        | 22   | 55     | B+     | 081987654321    | 4x           | Sertifikat
6   | Nur Azizah          | 18   | 51     | B+     | 081256745684    | 0x           | -
-----

Masukkan ID data yang ingin diubah: 4

(Kosongi jika tidak ingin diubah)
Nama baru (Michael Octaviano):
Umur baru (50):
BB baru (88): 85
Goldar baru (O+):
Telepon baru (081987654321):

Data berhasil diubah!
< kembali(0)
```

Gambar 4.5 Output Fitur Update

```
--- HAPUS DATA PENDONOR ---
-----
ID   | Nama                | Umur | BB(kg) | Goldar | Telepon          | Jumlah Donor | Penghargaan
-----
1   | Fred Brailey        | 93   | 71     | O+     | 081258016876    | 135x         | Satya Lencana
2   | Cristiano Ronaldo   | 40   | 83     | AB+    | 081987654321    | 34x          | Satya Lencana
3   | Hrithik Roshan      | 51   | 77     | B-     | 081987654321    | 41x          | Satya Lencana
4   | Michael Octaviano   | 50   | 85     | O+     | 081987654321    | 11x          | Satya Lencana
5   | Diko Saputra        | 22   | 55     | B+     | 081987654321    | 4x           | Sertifikat
6   | Nur Azizah          | 18   | 51     | B+     | 081256745684    | 0x           | -
-----

Masukkan ID data yang ingin dihapus: 1
Yakin ingin menghapus Fred Brailey? (y/n): y

Data berhasil dihapus!
< kembali(0)
```

Gambar 4.6 Output Fitur Delete

```

--- CATAT RIWAYAT DONOR ---
ID   | Nama                      | Jumlah Donor Saat Ini
-----
2    | Cristiano Ronaldo        | 34x
3    | Hrithik Roshan           | 41x
4    | Michael Octaviano        | 11x
5    | Diko Saputra             | 4x
6    | Nur Azizah               | 0x
-----

ID pendonor: 6

Berhasil! Riwayat donor Nur Azizah kini menjadi 1 kali.
< kembali(0)

```

Gambar 4.7 Output Pencatatan Riwayat Donor

```

--- HALAMAN REGISTER ---
Masukkan Username Baru : nurs
Masukkan Password Baru : 000

Akun berhasil dibuat! Silahkan login.

```

Gambar 4.8 Output Halaman Register

```
=====
                        SISTEM CEK DONOR DARAH
=====

1. Login
2. Register
3. Keluar
-----
Pilih Opsi (1/2/3): 3
Anda telah keluar
```

Gambar 4.9 Output Menu Keluar

5. Langkah-langkah GIT

5.1 GIT Add

```
ACER@nrs MINGW64 ~/Downloads/INFORMATIKA .doc/vscode/praktikum-apd (main)
$ git add .
```

Gambar 5.1 Git add

Git add berfungsi untuk menambah file dan folder.

5.2 GIT Commit

```
ACER@nrs MINGW64 ~/Downloads/INFORMATIKA .doc/vscode/praktikum-apd (main)
$ git commit -m "pt 6"
[main 6b9bdf7] pt 6
3 files changed, 81 insertions(+)
create mode 100644 post-test/post-test-apd-6/2509106018-NUR AZIZAH ISLAMIYAH-PT-6.pdf
create mode 100644 post-test/post-test-apd-6/2509106018-NUR AZIZAH ISLAMIYAH-PT-6.py
```

Gambar 5.2 Git Commit

Git commit berfungsi untuk menyimpan perubahan di repository, dan pesan commit untuk menjelaskan perubahan.

5.3 GIT Push

```
ACER@nrs MINGW64 ~/Downloads/INFORMATIKA .doc/vscode/praktikum-apd (main)
$ git push
Enumerating objects: 13, done.
Counting objects: 100% (13/13), done.
Delta compression using up to 22 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (8/8), 807.20 KiB | 12.05 MiB/s, done.
Total 8 (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/nrszzh/praktikum-apd.git
a3ab3e6..6b9bdf7 main -> main
```

Gambar 5.3 Git Push

Git push berfungsi untuk mengupload file yang ada di repository komputer ke repository Github.