

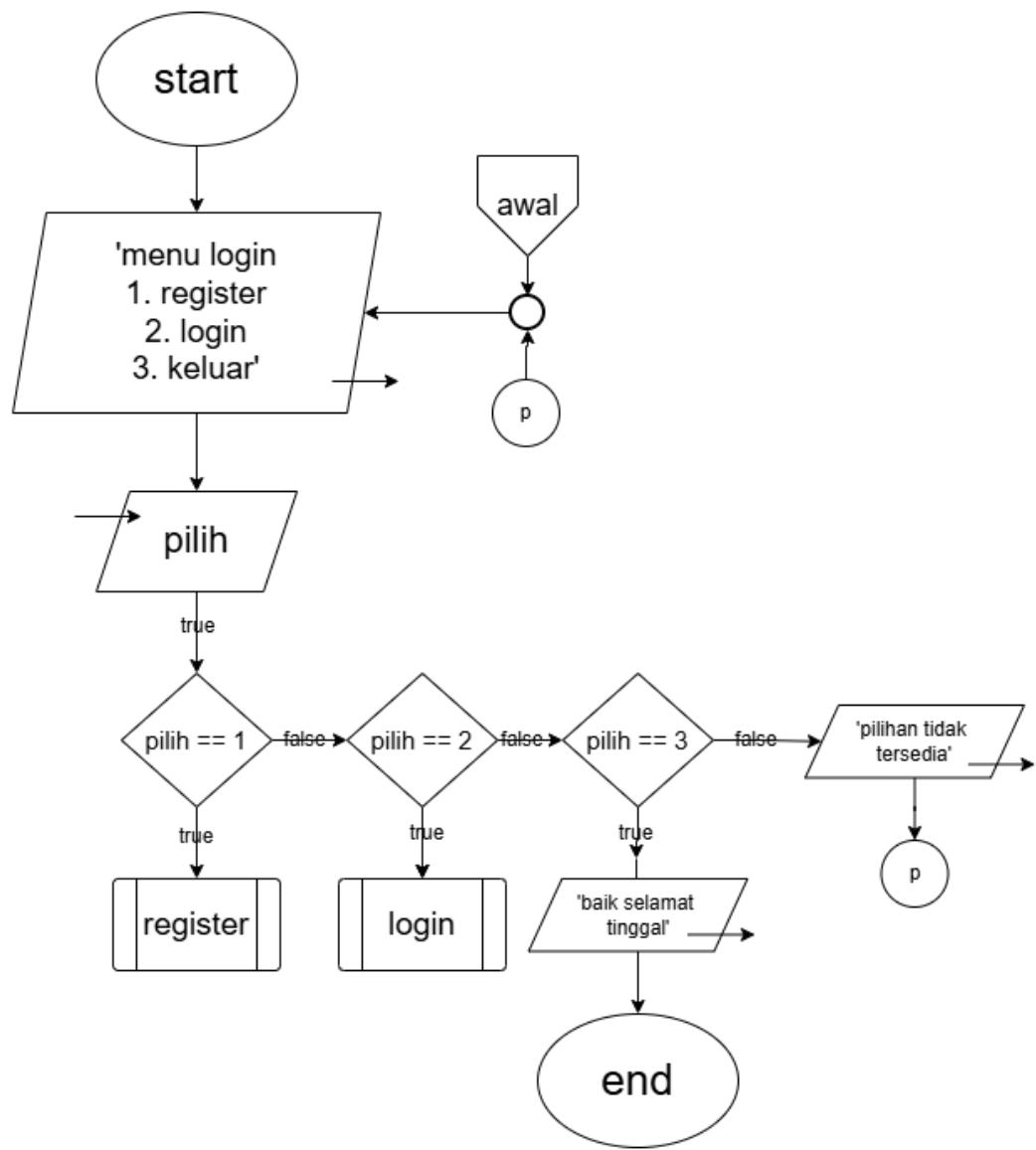
**LAPORAN PRAKTIKUM  
POSTTEST (8)  
ALGORITMA PEMROGRAMAN DASAR**



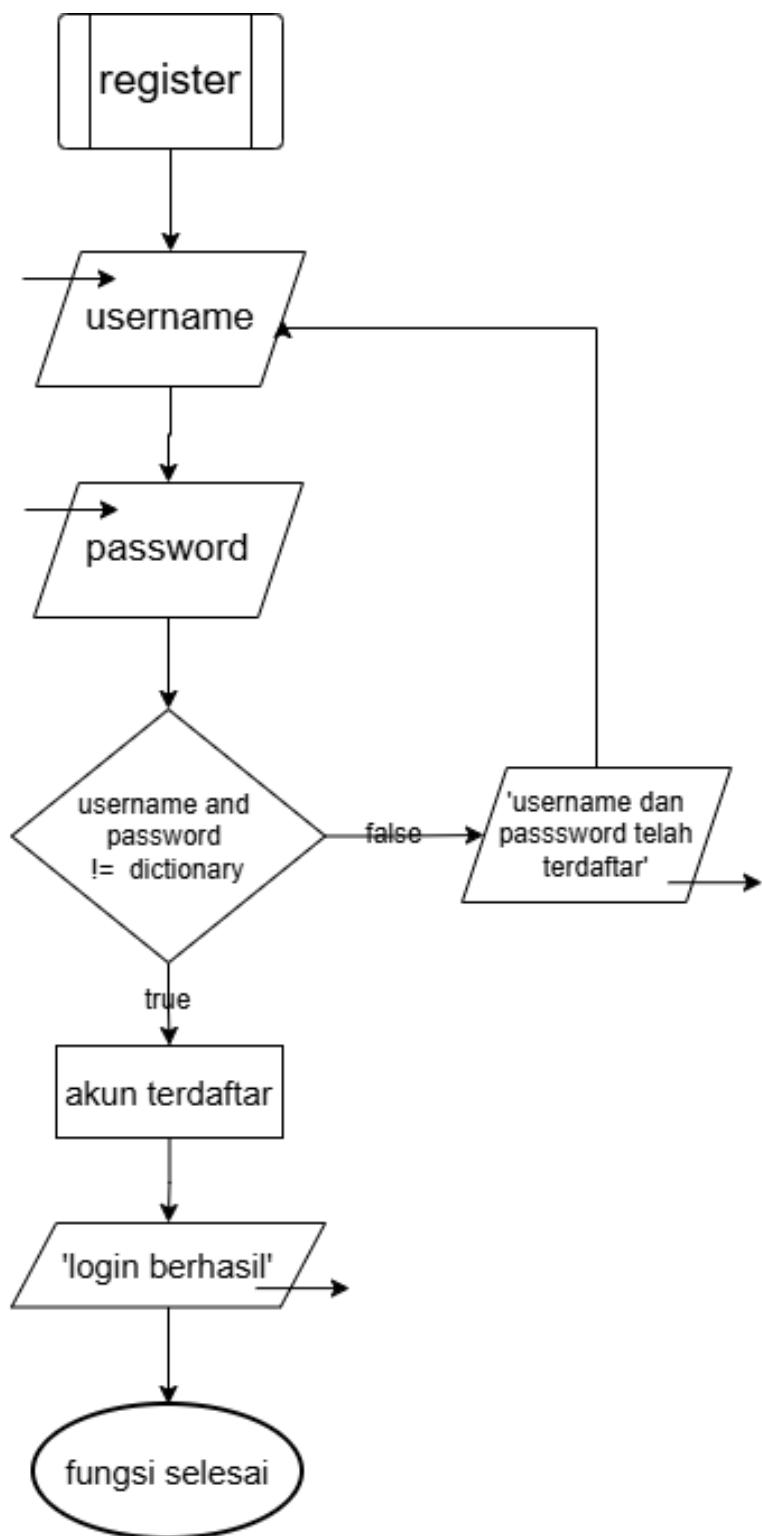
**Disusun oleh:**  
**Muhammad Rizky Budianto (2509106060)**  
**Kelas (B1'25)**

**PROGRAM STUDI INFORMATIKA  
UNIVERSITAS MULAWARMAN  
SAMARINDA  
2025**

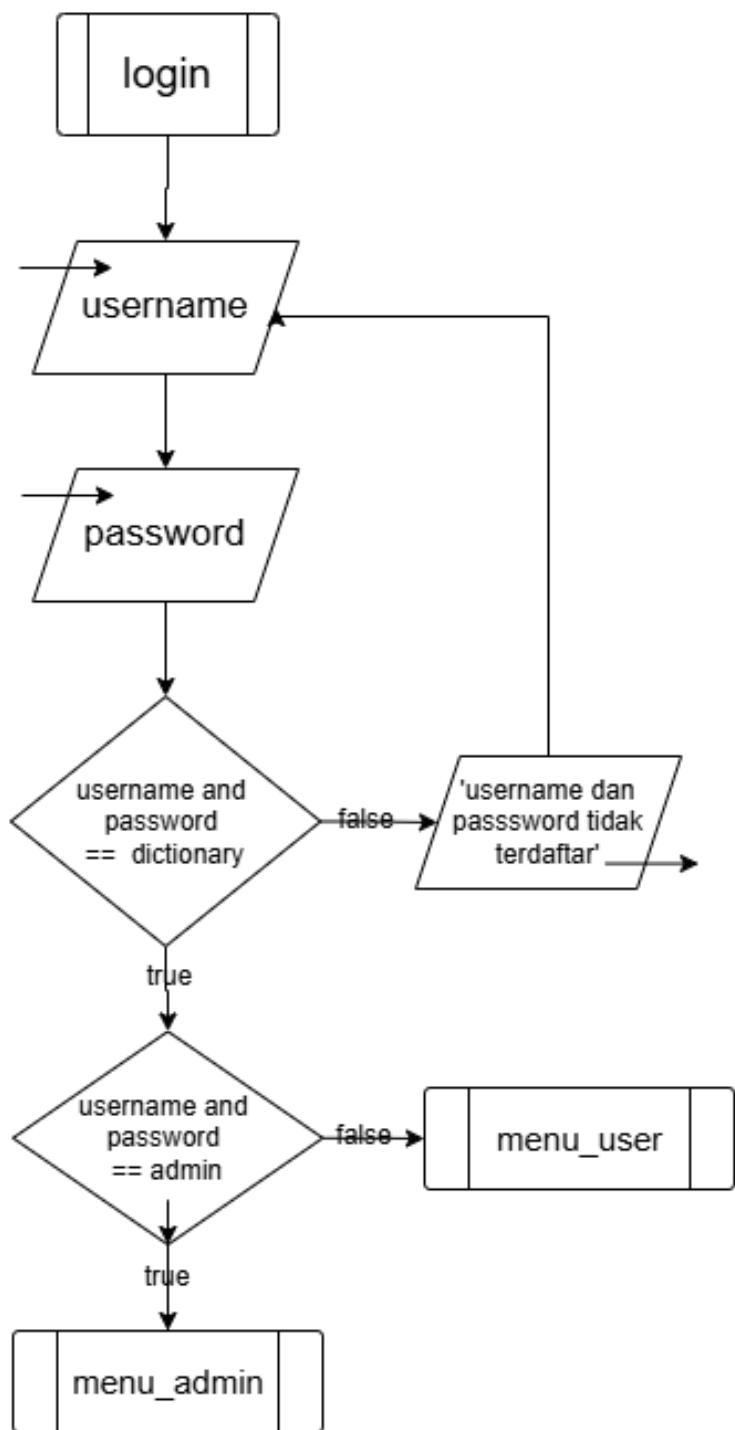
## 1. Flowchart



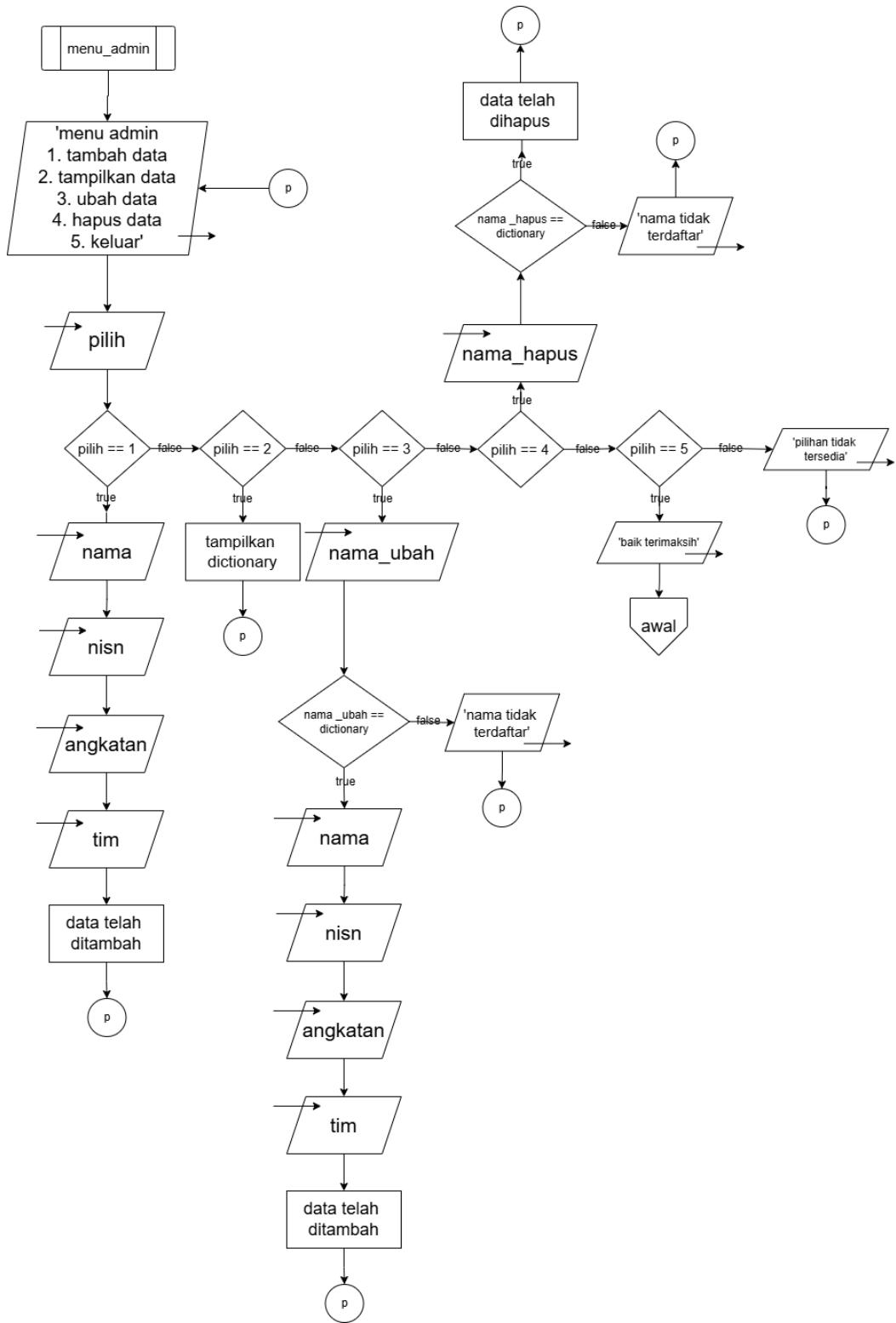
Gambar 1.1 menu awal



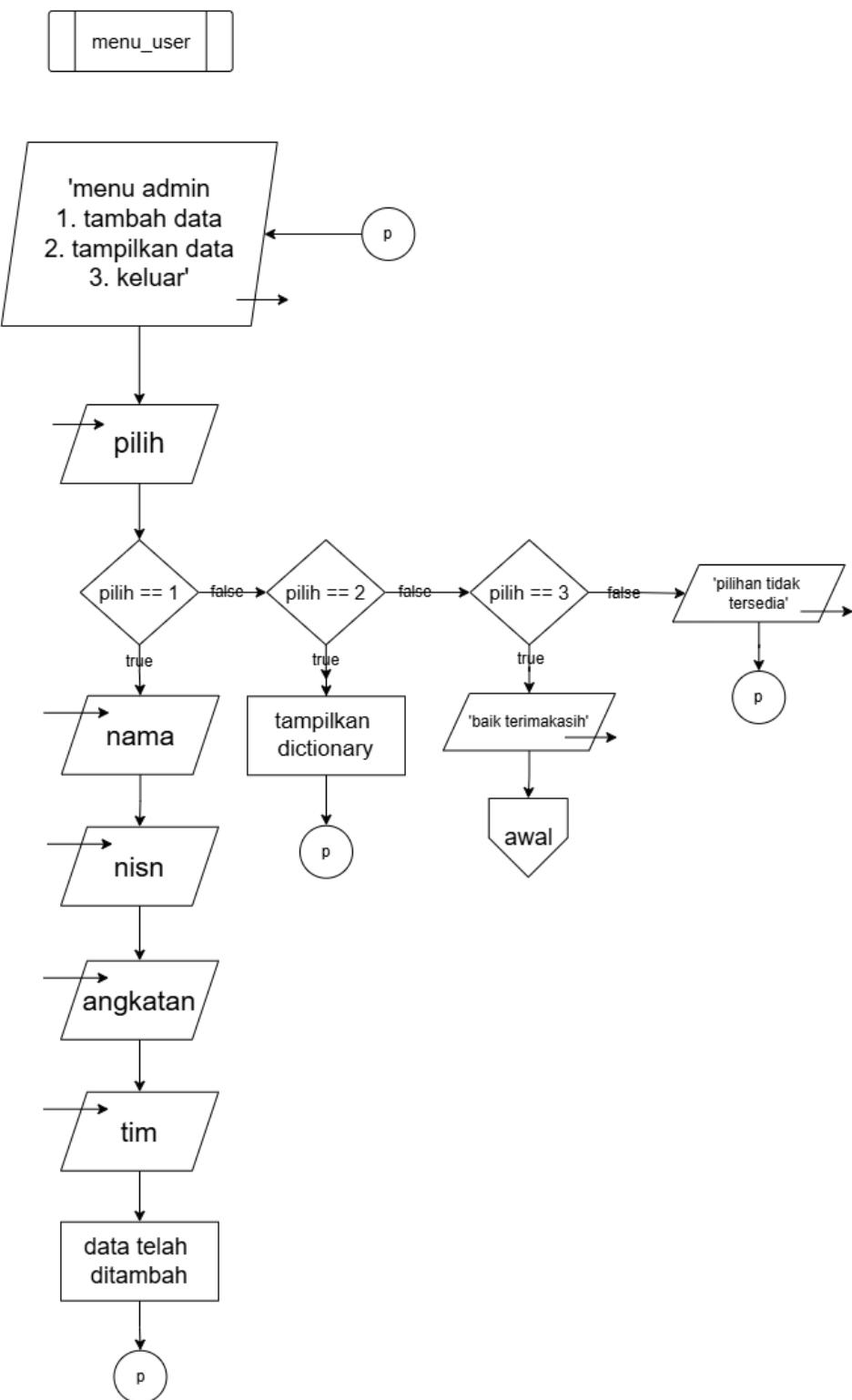
Gambar 1.2 fungsi register



Gambar 1.3 fungsi login



Gambar 1.4 fungsi menu\_admin



Gambar 1.5 fungsi menu\_user

## 2. Deskripsi Singkat Program

Program ini dibuat/diciptakan agar organisasi paskibra indonesia terkhusus PASKIBSMADA MANAGEMENT dapat mengatur dengan mudah dan cepat arsip keanggotaan mereka, dengan tujuan agar arsip keanggotaan bisa disimpan dengan tersusun rapih.

## 3. Source Code

Source Code :

Main :

```
import inquirer
import os
from prog_adser import menu_admin, menu_user
data={}
user= {
    'admin':{'pass':'dewa'},
    'user':{'pass':'rakyat'},
}

def register():
    usr=input('masukkan username = ')
    pas=input('masukkan password = ')
    if usr in user:
        print('Username sudah ada, silahkan coba lagi')
    else:
        user[usr] = {'pass': pas}
        print('Selamat akun berhasil didaftarkan! ')

def login():
    usr=input('masukkan username = ')
    pas=input('masukkan password = ')
    if usr in user and user[usr]['pass']==pas:
        print('=====login berhasil=====')
        if usr == 'admin':
            menu_admin(data)
        else : menu_user(data)
    else : print('username atau password salah silahkan coba lagi')

print('Login module loaded successfully.')
```

```

def menu_login():
    while True:
        os.system('cls')
        pertanyaan = [
            inquirer.List('pilihan',
                          message="Pilih opsi:",
                          choices=['1. REGISTER', '2. LOGIN', '3. KELUAR'],
                          )
        ]

        try:
            answers = inquirer.prompt(pertanyaan)
            pilih = int(answers['pilihan'].split('.')[0])
            if pilih == 1:
                register()
            elif pilih == 2:
                login()
            elif pilih == 3:
                os.system('cls')
                print('Baik! Selamat tinggal')
                break
            else:
                print('Pilihan tidak tersedia!')
                input('Tekan Enter untuk lanjut...')
        except ValueError:
            print('(ERROR) Masukkan angka, bukan huruf/kosong/simbol!')
            input('Tekan Enter untuk lanjut...')

menu_login()

```

### Admin dan user :

```

import inquirer
import os
from prettytable import PrettyTable

def tampil_data(n):
    tabel = PrettyTable()
    tabel.field_names = ['NAMA', 'NISN', 'ANGKATAN', 'TIM']
    if len(n)==0:
        print('data masih belum ada')
    else :
        for key, value in n.items():
            tabel.add_row([key, value['nisn'], value['angkatan'],
value['tim']])

```

```


| DATA ANGGOTA      |                |
|-------------------|----------------|
| 1. TAMBAH DATA    | TAMBAH DATA    |
| 2. TAMPILKAN DATA | TAMPILKAN DATA |
| 3. UBAH DATA      | UBAH DATA      |
| 4. HAPUS DATA     | HAPUS DATA     |
| 5. KELUAR         | KELUAR         |


```

def menu\_admin(data):

```

    while True :
        pertanyaan = [
            inquirer.List('pilihan',
                           message="Pilih opsi:",
                           choices=['1. TAMBAH DATA', '2. TAMPILKAN
DATA', '3. UBAH DATA', '4. HAPUS DATA', '5. KELUAR'],
                           )
        ]
        try:
            answers = inquirer.prompt(pertanyaan)
            pilih = int(answers['pilihan'].split('.')[0])
            if pilih == 1:
                nama=input('nama = ')
                nisn=input('nisn = ')
                angkatan=input('angkatan = ')
                tim=input('tim = ')
                data[nama] = {
                    'nisn': nisn,
                    'angkatan': angkatan,
                    'tim': tim
                }
            elif pilih == 2:
                tampil_data(data)

            elif pilih == 3:
                nama_lama = input('Nama anggota yang akan diubah =
')
                if nama_lama in data:
                    nama_baru = input('Nama = ')
                    nisn_baru = input('NISN = ')
                    angkatan_baru = input('Angkatan = ')
                    tim_baru = input('Tim = ')
                    data[nama_baru] = {
                        'nisn': nisn_baru,
                        'angkatan': angkatan_baru,
                        'tim': tim_baru
                    }
                    if nama_baru != nama_lama:
                        del data[nama_lama]
                    print('Data anggota berhasil diubah')
                else:
                    print('Nama tidak ditemukan.')
```
```

```

        elif pilih == 4:
            hapus = input('Nama anggota yang ingin dihapus = ')
            if hapus in data:
                del data[hapus]
                print('Data berhasil dihapus')
            else:
                print('Anggota tidak ditemukan.')

        elif pilih == 5:
            print('baik terimakasih telah menggunakan program ini')
            break
        else:
            print('Pilihan tidak tersedia!')
            input('Tekan Enter untuk lanjut...')
    except ValueError:
        print('(ERROR) Masukkan angka, bukan huruf/kosong/simbol!')
        input('Tekan Enter untuk lanjut...')

def menu_user(data):
    while True :
        pertanyaan = [
            inquirer.List('pilihan',
                          message="Pilih opsi:",
                          choices=['1. TAMBAH DATA', '2. TAMPILKAN DATA', '3. KELUAR'],
                          )
        ]
        try:
            answers = inquirer.prompt(pertanyaan)
            pilih = int(answers['pilihan'].split('.')[0])
            if pilih == 1:
                nama=input('nama = ')
                nisn=input('nisn = ')
                angkatan=input('angkatan = ')
                tim=input('tim = ')
                data[nama] = {
                    'nisn': nisn,
                    'angkatan': angkatan,
                    'tim': tim
                }
                continue

            elif pilih == 2:
                tampil_data(data)
                continue

```

```
        elif pilih == 3:  
            os.system('cls')  
            print('baik! kembali ke halaman login')  
            break  
        else:  
            print('Pilihan tidak tersedia!')  
            input('Tekan Enter untuk lanjut...')  
    except ValueError:  
        print('(ERROR) Masukkan angka, bukan  
huruf/kosong/simbol!')  
        input('Tekan Enter untuk lanjut...')  
        continue  
    return data
```

#### 4. Hasil Output

```
[?] Pilih opsi::  
> 1. REGISTER  
2. LOGIN  
3. KELUAR
```

Gambar 4.1 halaman login

```
=====login berhasil=====  
[?] Pilih opsi::  
> 1. TAMBAH DATA  
2. TAMPILKAN DATA  
3. UBAH DATA  
4. HAPUS DATA  
5. KELUAR
```

Gambar 4.2 menu admin

```
=====login berhasil=====  
[?] Pilih opsi::  
> 1. TAMBAH DATA  
2. TAMPILKAN DATA  
3. KELUAR
```

Gambar 4.3 manu user

## 5. Langkah-langkah GIT

```
PS C:\praktikum-apd\post-test> git add .
PS C:\praktikum-apd\post-test> git commit -m "PT 6"
[main 61958fe] PT 6
 2 files changed, 139 insertions(+)
 hammad Rizky Budianto-PT-6.pdf
    create mode 100644 post-test/post-test-apd-6/2509106060-Muhammad Rizky Budianto-PT-6.py
PS C:\praktikum-apd\post-test> git push -u origin main
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 2 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 614.53 KiB | 6.83 MiB/s, done.
Total 6 (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/duatigaikanle/praktikum-apd.git
  f1bd33e..61958fe main -> main
branch 'main' set up to track 'origin/main'.
```

### 5.1 GIT Add

**Berfungsi untuk menandai data data yang ingin dikirimkan ke dalam repository.**

### 5.2 GIT Commit

**Untuk merekam/menyimpan semua data yang telah di "git add."**

### 5.3 GIT Push

**Mengirim data yang telah di "git commit" ke dalam repository.**