

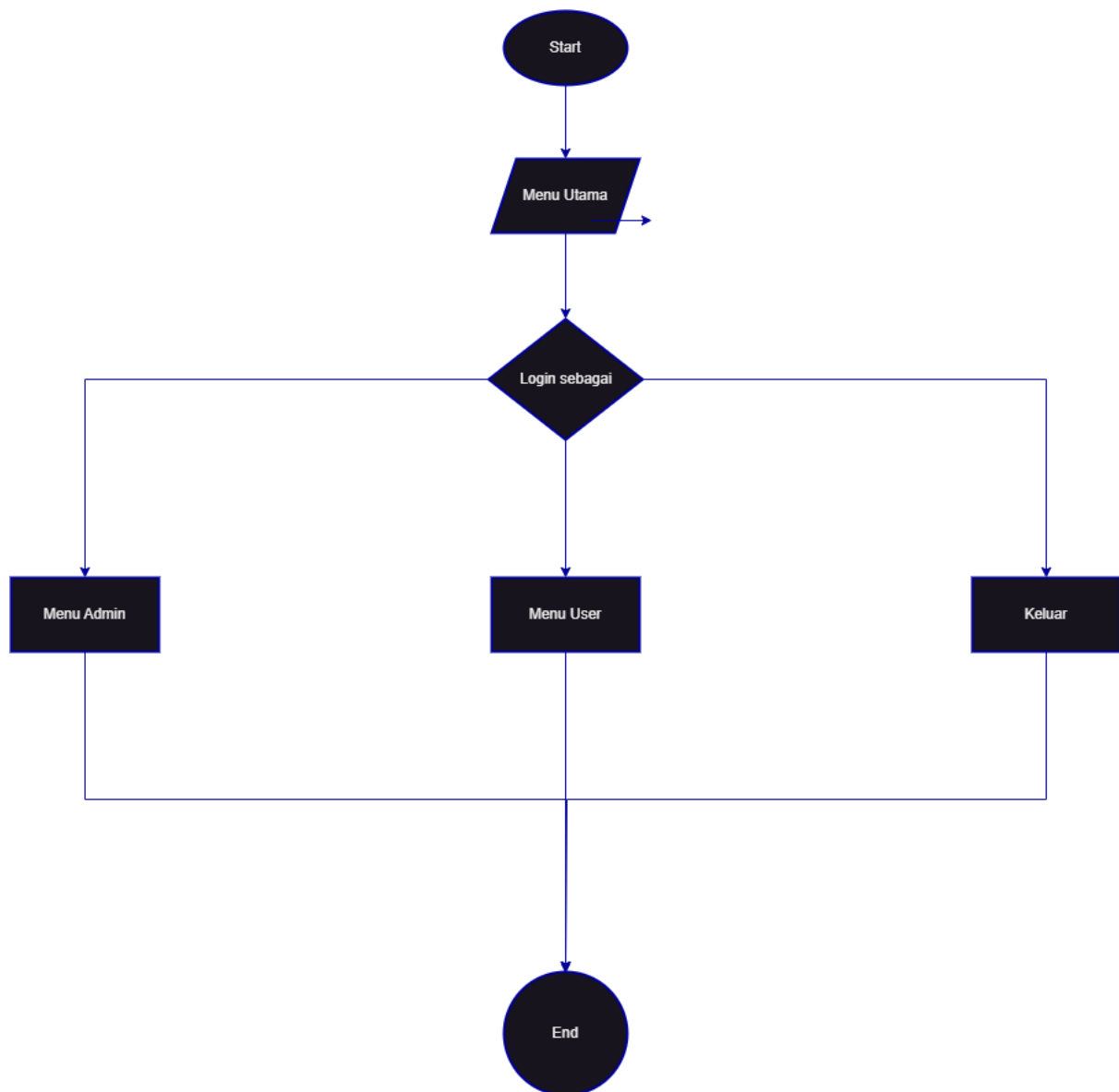
**LAPORAN PRAKTIKUM**  
**POSTTEST 2**  
**ALGORITMA PEMROGRAMAN LANJUT**



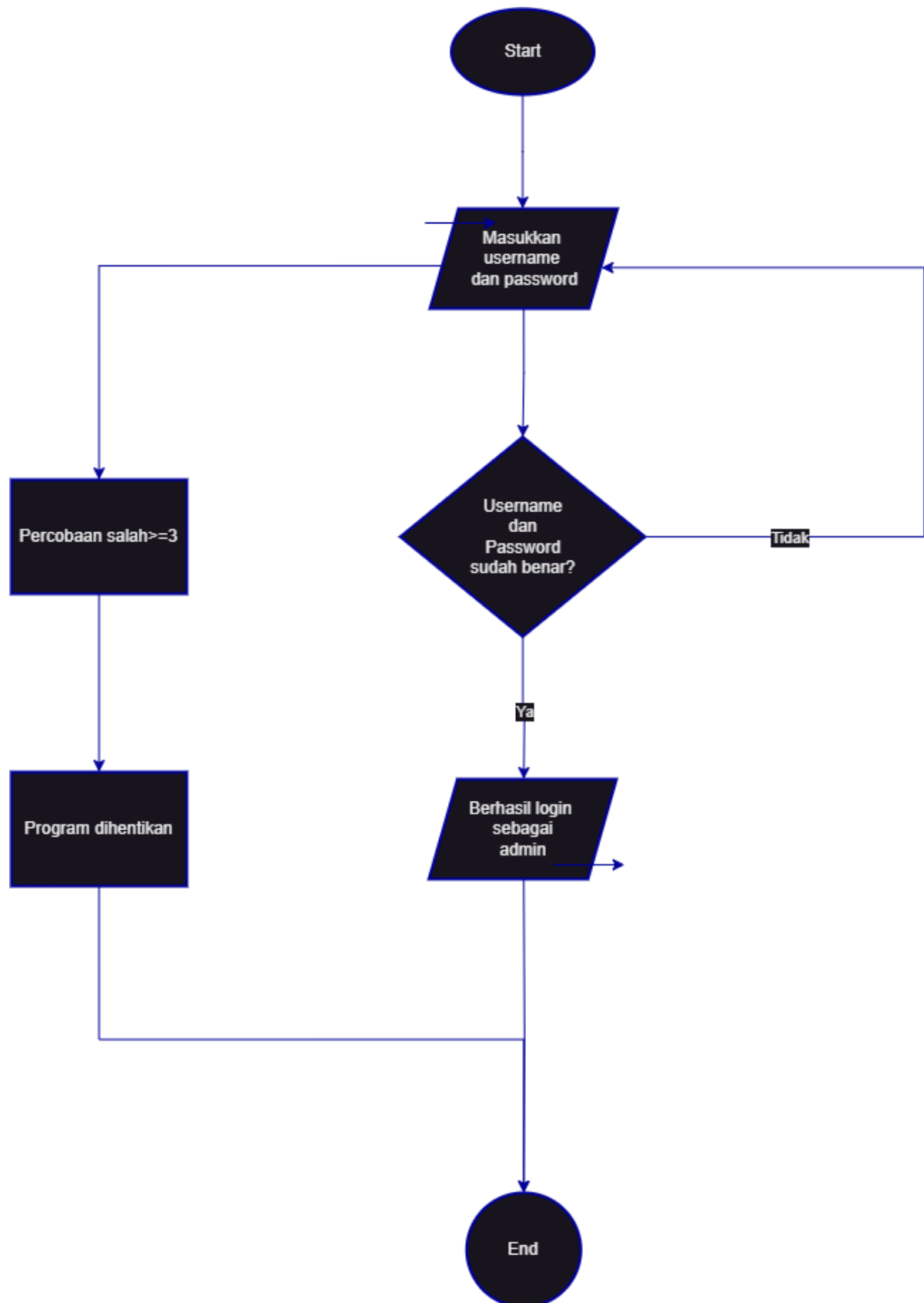
**Disusun oleh:**  
**Rusdiansyah (2409106013)**  
**Kelas (A1 '24)**

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

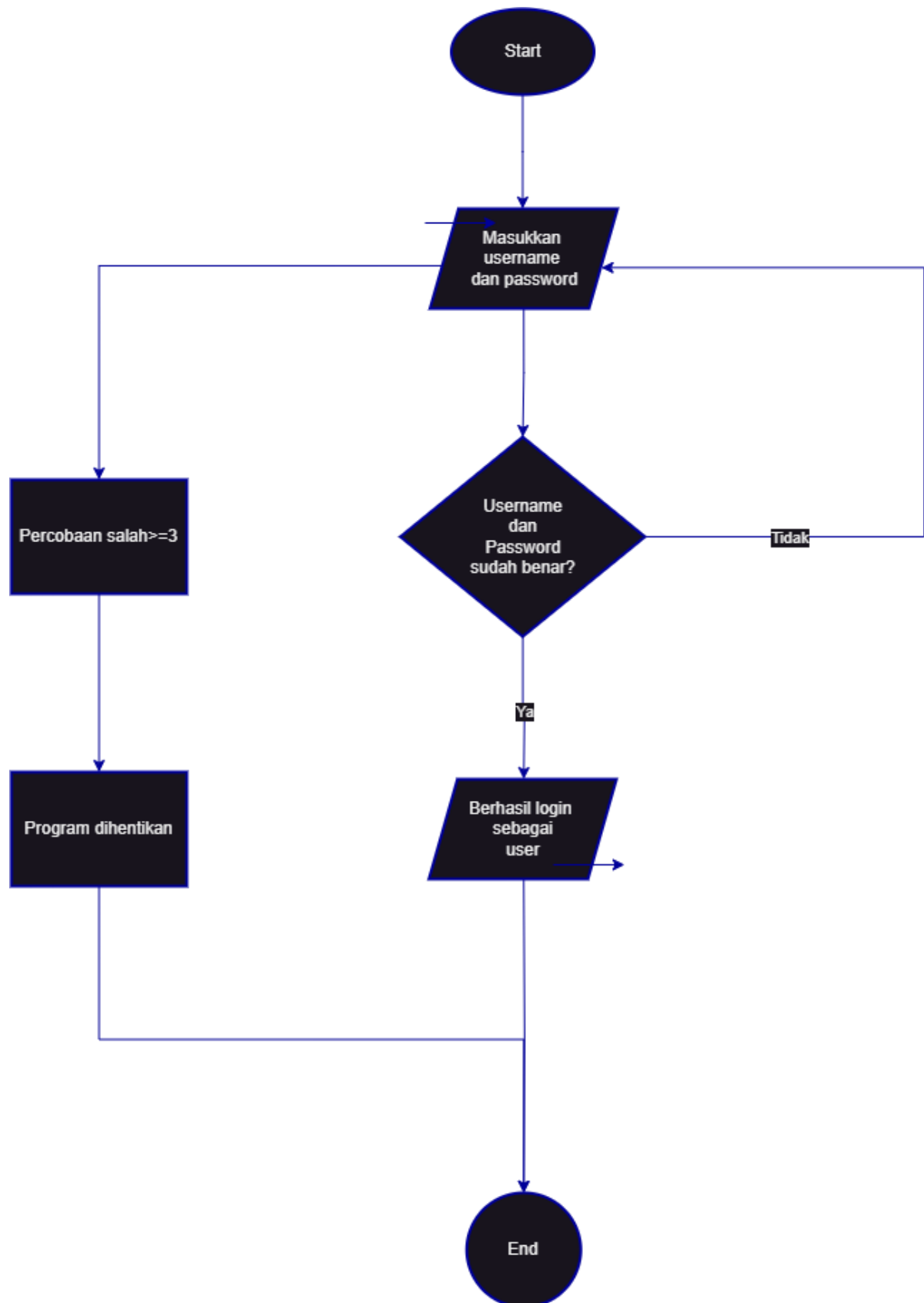
## 1. Flowchart



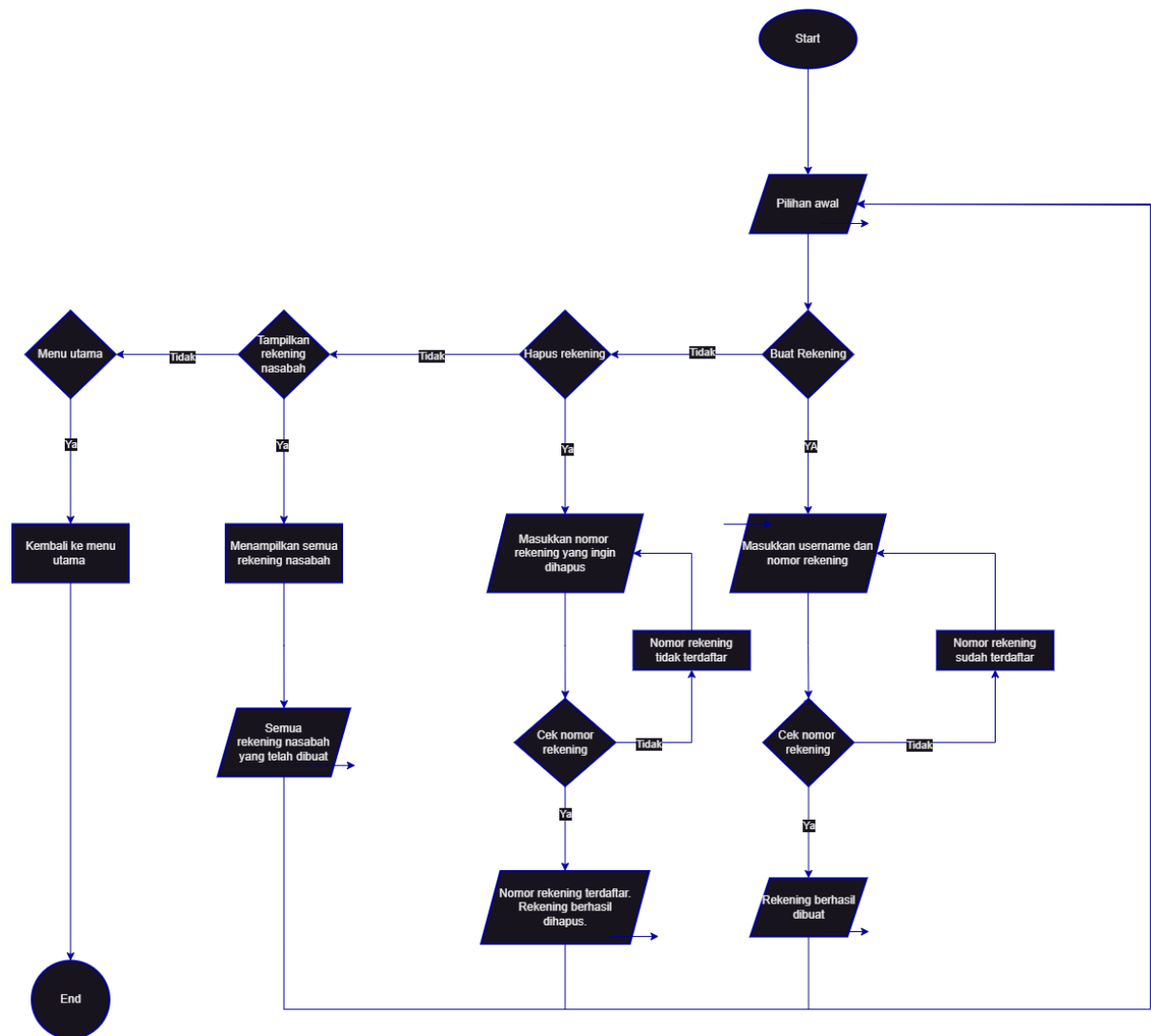
Gambar 1.1 Menu Utama



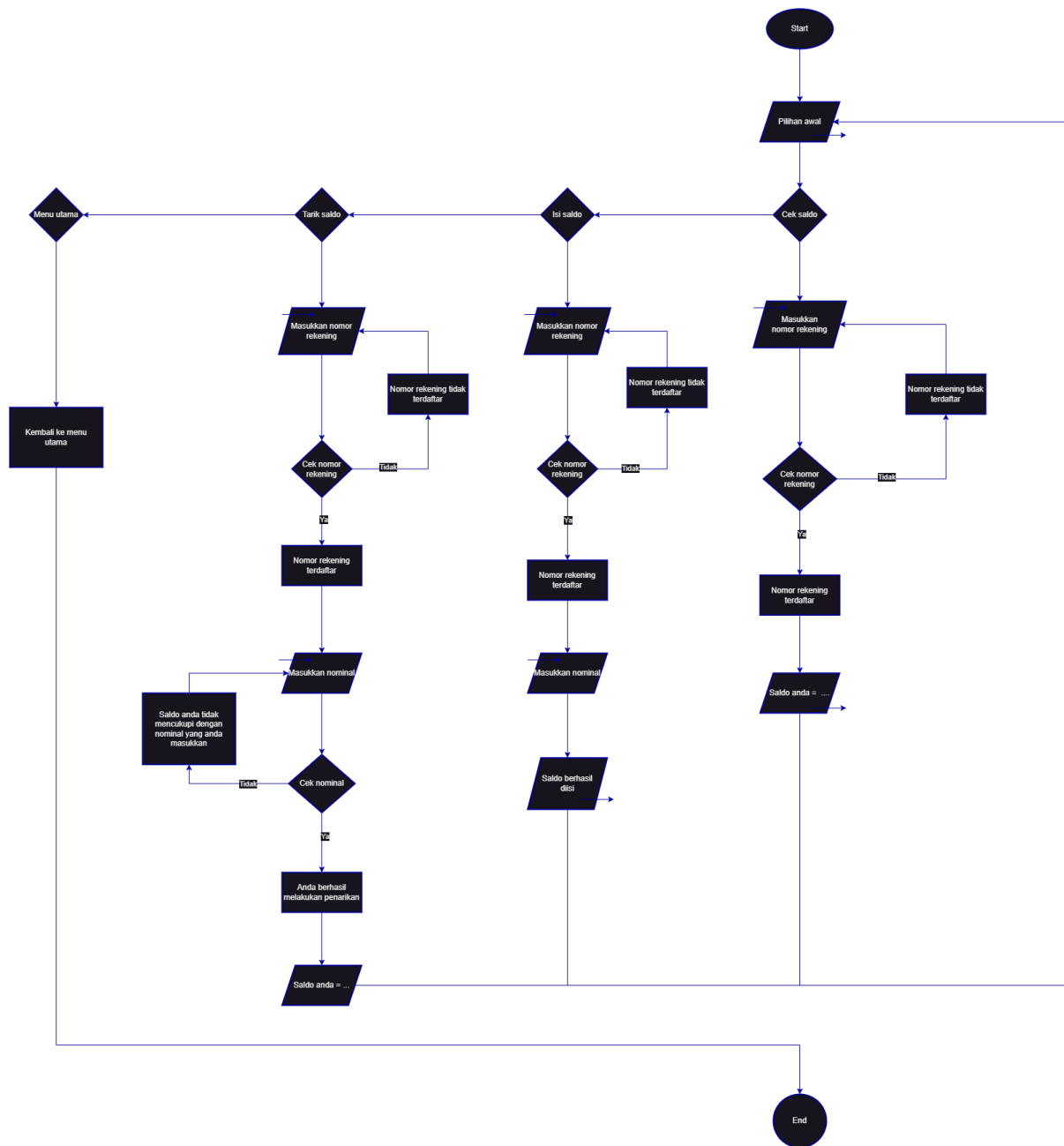
Gambar 1.2 Login sebagai admin



Gambar 1.3 Login sebagai user



Gambar 1.4 Menu admin



Gambar 1.5 Menu user

## **2. Analisis Program**

### **2.1 Deskripsi Singkat Program**

Program ini merupakan simulasi sistem perbankan sederhana yang memungkinkan user atau pengguna untuk melakukan operasi dasar seperti pembuatan rekening, penghapusan rekening, pengecekan saldo, pengisian saldo, dan penarikan saldo. Terdapat dua jenis pengguna yakni admin dan user. Yang dimana, admin bertugas mengelola rekening nasabah seperti membuat, menghapus, dan menampilkan data rekening. Sedangkan user dapat mengakses informasi saldo, mengisi saldo, dan menarik saldo.

### **2.2 Penjelasan Alur & Algoritma**

#### **1.Menu utama**

Program menampilkan pilihan login sebagai admin, user atau keluar.

#### **2.Login**

Pada saat login, user atau admin diminta untuk memasukkan username dan password. Jika login berhasil, pengguna diarahkan ke menu sesuai peran (admin/user). Jika gagal sebanyak 3 kali, program akan dihentikan.

#### **3.Menu admin**

- Buat rekening

Input nama dan nomor rekening, lalu simpan data.

- Hapus rekening

Hapus rekening berdasarkan nomor rekening yang diinput.

- Tampilkan nasabah

Menampilkan daftar rekening beserta saldo.

- Kembali ke menu utama

Keluar dari menu admin.

#### 4.Menu user

- Cek saldo

Menampilkan saldo berdasarkan nomor rekening yang diinput.

- Isi saldo

Menambahkan saldo ke dalam rekening.

- Tarik saldo

Melakukan penarikan saldo sesuai pada nomor rekening yang diinput.

- Kembali ke menu utama

Keluar dari menu user.

- Keluar

Program berhenti jika pengguna memilih keluar.

#### 5.Penjelasan fungsi penting

- Fungsi menu

Fungsi menu tentunya untuk menampilkan MenuUtama, MenuAdmin,dan menampilkan MenuUser.

- Fungsi buatRekening

Membuat rekening baru dengan menyimpan nama dan nomor rekening ke array Data\_Rekening\_Nasabah.

- Fungsi hapusRekening

Menghapus rekening berdasarkan nomor rekening yang dimasukkan.

- Fungsi tampilkanNasabah

Menampilkan daftar semua rekening nasabah beserta saldo.



- Fungsi cekSaldo, isiSaldo, tarikSaldo

Mengelola saldo rekening user berdasarkan nomor rekening yang diinput.

- Fungsi login

Untuk memverifikasi username dan password. Jika sesuai, pengguna akan diarahkan ke menu yang sesuai.

- Fungsi main

Yakni mengontrol alur program secara keseluruhan, termasuk menampilkan menu dan memproses pilihan pengguna.

### 3. Source Code

#### a) Array

Data\_Rekening\_Nasabah : menggunakan array 2D untuk menyimpan data nasabah (nama dan nomor rekening).

Saldo : array untuk menyimpan saldo setiap nasabah.

```
string Data_Rekening_Nasabah[MAX_NASABAH][10];  
int Saldo[MAX_NASABAH] = {0};
```

Gambar 3.1 Array

#### b) Fungsi Buat rekening

Membuat rekening baru dan memeriksa apakah masih ada slot tersedia dan nomor rekening belum digunakan. Menyimpan data nasabah (nama dan nomor rekening) serta menginisialisasi saldo ke 0.

```
void buatRekening() {  
    if (panjang >= MAX_NASABAH) {  
        cout << "Maaf, jumlah nasabah sudah penuh. Tidak dapat menambahkan  
rekening baru." << endl;  
        return;  
    }  
  
    string Nama_Nasabah, Nomor_Rekening;  
    cout << "Masukkan nama nasabah: ";  
    cin >> Nama_Nasabah;  
    cout << "Masukkan nomor rekening: ";  
    cin >> Nomor_Rekening;  
  
    for (int i = 0 ; i < panjang; i++) {  
        if (Data_Rekening_Nasabah[i][1] == Nomor_Rekening) {  
            cout << "Nomor rekening sudah digunakan. Tidak dapat membuat  
rekening baru." << endl;  
            return;  
        }  
    }  
  
    Data_Rekening_Nasabah[panjang][0] = Nama_Nasabah;  
    Data_Rekening_Nasabah[panjang][1] = Nomor_Rekening;  
    Saldo[panjang] = 0;  
    panjang++;  
    cout << "Rekening berhasil dibuat." << endl;  
}
```

Gambar 3.2 Fungsi buat rekening

c) Fungsi hapus rekening

Menghapus rekening berdasarkan nomor rekening yang dimasukkan. Menggeser data nasabah setelah rekening dihapus untuk menjaga urutan array.

```
void hapusRekening() {
    if (panjang == 0) {
        cout << "Belum ada rekening yang terdaftar.\n";
    } else {
        string Nomor_Rekening;
        cout << "Masukkan nomor rekening yang akan dihapus: ";
        cin >> Nomor_Rekening;

        int ditemukan = 0;
        for (int i = 0; i < panjang; i++) {
            if (Data_Rekening_Nasabah[i][1] == Nomor_Rekening) {
                for (int j = i; j < panjang - 1; j++) {
                    Data_Rekening_Nasabah[j][0] = Data_Rekening_Nasabah[j +
1][0];
                    Data_Rekening_Nasabah[j][1] = Data_Rekening_Nasabah[j +
1][1];
                    Saldo[j] = Saldo[j + 1];
                }
                panjang--;
                ditemukan = 1;
                cout << "Rekening berhasil dihapus." << endl;
                break;
            }
        }

        if (ditemukan == 0) {
            cout << "Nomor rekening tidak ditemukan.\n";
        }
    }
}
```

Gambar 3.3 Fungsi hapus rekening

d) Fungsi cek saldo

Menampilkan saldo nasabah berdasarkan nomor rekening dan jumlah yang dimasukkan. Memeriksa apakah nomor rekening valid.

```
void cekSaldo() {
    if (panjang == 0) {
        cout << "Belum ada rekening yang terdaftar.\n";
    } else {
        string Nomor_Rekening;
        cout << "Masukkan nomor rekening: ";
        cin >> Nomor_Rekening;

        int ditemukan = 0;
        for (int i = 0; i < panjang; i++) {
            if (Data_Rekening_Nasabah[i][1] == Nomor_Rekening) {
                cout << "Nama nasabah: " << Data_Rekening_Nasabah[i][0] << "\n";
                cout << "Nomor rekening: " << Data_Rekening_Nasabah[i][1] <<
"\n";

                cout << "Saldo: Rp." << Saldo[i] << "\n";
                ditemukan = 1;
                break;
            }
        }
        if (!ditemukan) {
            cout << "Nomor rekening tidak ditemukan.\n";
        }
    }
}
```

Gambar 3.4 Fungsi cek saldo

e)Fungsi isi saldo

Menambahkan saldo ke rekening nasabah berdasarkan nomor rekening dan jumlah yang dimasukkan. Memeriksa apakah nomor rekening valid.

```
void isiSaldo() {
    if (panjang == 0) {
        cout << "Belum ada rekening yang terdaftar.\n";
    } else {
        string Nomor_Rekening;
        int jumlah;
        cout << "Masukkan nomor rekening: ";
        cin >> Nomor_Rekening;
        cout << "Masukkan jumlah saldo yang ingin diisi: Rp. ";
        cin >> jumlah;

        int ditemukan = 0;
        for (int i = 0; i < panjang; i++) {
            if (Data_Rekening_Nasabah[i][1] == Nomor_Rekening) {
                Saldo[i] += jumlah;
                cout << "Saldo berhasil diisi! Saldo anda sekarang: Rp." <<
Saldo[i] << "\n";
                ditemukan = 1;
                break;
            }
        }
        if (ditemukan == 0) {
            cout << "Nomor rekening tidak ditemukan.\n";
        }
    }
}
```

Gambar 3.5 Fungsi isi saldo

f) Fungsi tarik saldo

Mengurangi saldo nasabah berdasarkan nomor rekening dan jumlah yang dimasukkan. Juga memeriksa apakah saldo mencukupi dan nomor rekening valid.

```
void tarikSaldo() {
    if (panjang == 0) {
        cout << "Belum ada rekening yang terdaftar.\n";
    } else {
        string Nomor_Rekening;
        int jumlah;
        cout << "Masukkan nomor rekening: ";
        cin >> Nomor_Rekening;
        cout << "Masukkan jumlah saldo yang ingin ditarik: Rp. ";
        cin >> jumlah;

        int ditemukan = 0;
        for (int i = 0; i < panjang; i++) {
            if (Data_Rekening_Nasabah[i][1] == Nomor_Rekening) {
                if (Saldo[i] >= jumlah) {
                    Saldo[i] -= jumlah;
                    cout << "Saldo berhasil ditarik! Saldo anda sekarang: Rp."
<< Saldo[i] << "\n";

                    } else {
                        cout << "Saldo tidak cukup.\n";
                    }
                    ditemukan = 1;
                    break;
                }
            }
        }
        if (ditemukan == 0) {
            cout << "Nomor rekening tidak ditemukan.\n";
        }
    }
}
```

Gambar 3.6 Fungsi tarik saldo

#### g) Fungsi login

Yang terakhir fungsi login, yakni memeriksa username dan password untuk login sebagai admin ataupun user. Memberikan 3 kesalahan sebelum menghentikan program.

```
int login(string role) {
    string username, password;
    int kesalahanLogin = 0;

    while (kesalahanLogin < 3) {
        cout << "Masukkan Username: ";
        cin >> username;
        cout << "Masukkan Password: ";
        cin >> password;

        if (role == "admin" && username == "admin" && password == "admin123") {
            return 1;
        } else if (role == "user" && username == "user" && password ==
"user123") {
            return 1;
        } else {
            kesalahanLogin++;
            cout << "Username atau password salah. Kesalahan ke-" <<
kesalahanLogin << ".\n";
        }
    }

    cout << "Anda telah melakukan 3 kali kesalahan login. Program
dihentikan.\n";
    return 0;
}
```

Gambar 3.7 Fungsi login

## 4. Uji Coba dan Hasil Output

### 4.1 Uji Coba

#### 1) Login sebagai admin :

Input

- Pilih menu 1 login sebagai admin.
- Masukkan username : admin.
- Masukkan password : admin123.

Output yang diharapkan

- Berhasil login dan menampilkan menu admin.

Uji fitur admin :

- Buat rekening.
- Input nama nasabah dan nomor rekening.
- Output “Rekening berhasil dibuat.”.

Hapus rekening :

- Input nomor rekening.
- Output “Rekening berhasil dihapus”.

Tampilkan rekening nasabah :

- Output : Menampilkan daftar nasabah yang terdaftar.

#### 2) Login sebagai user :

Input

- Pilih menu 2 untuk login sebagai user.
- Masukkan username : user.
- Masukkan password : user123.

Output yang diharapkan

- Berhasil login dan menampilkan menu user.

Uji fitur user:

- Cek saldo.
- Input : nomor rekening
- Output : menampilkan saldo nasabah
- Isi saldo.
- Input : nomor rekening dan jumlah saldo.
- Output : “Saldo berhasil ditarik! Saldo anda sekarang : Rp....”.

#### 3) Input yang tidak valid

Login dengan password yang salah

- Input : username = admin, password = salah.
- Output : “Username atau password salah. Kesalahan ke -1.”
- Ulangi hingga 3 kali kesalahan.
- Output : “Anda telah melakukan 3 kali kesalahan login. Program dihentikan.”



4) Batas maksimal nasabah

Input

- Buat rekening hingga mencapai batas maksimal
- Contoh:  
Rekening 1 : Nama = Rusdy, nomor rekening = 1234  
Rekening 2 : Nama = Ahnaf, nomor rekening = 4321
- Buat rekening ke-3.
- Output yang diharapkan “Maaf jumlah nasabah sudah penuh. Tidak dapat menambahkan rekening baru.”

5) Nomor rekening yang sudah ada

Input

- Buat rekening dengan nomor rekening yang sudah digunakan.
- Contoh:  
Rekening 1 : Nama = Rusdy, nomor rekening = 1234  
Coba buat rekening baru dengan nomor rekening = 1234
- Output yang diharapkan “Nomor rekening sudah digunakan. Tidak dapat membuat rekening baru.”

6) Penarikan saldo melebihi saldo yang ada

Input

- Buat rekening : Nama = Rusdy, nomor rekening = 1234, saldo awal Rp.55000
- Tarik saldo dengan jumlah = Rp.65000
- Output yang diharapkan “Saldo tidak cukup”

7) Menampilkan daftar nasabah kosong

Input

- Hapus semua rekening yang sudah dibuat.
- Kemudian pilih menu “Tampilkan rekening nasabah”
- Output yang diharapkan “Belum ada rekening yang terdaftar.”

8) Keluar dari program

Input

- Pilih menu 0 untuk keluar.
- Output yang diharapkan “Terimakasih telah menggunakan program bank sederhana ini.”
- Program berhenti.

## 4.2 Hasil Output

### 1. Login sebagai admin

```
PS C:\praktikum-apl\post-test\post-test-2> cd "c:\praktikum-apl\post-test\post-test-2"
if ($?) { .\2409106013-RUSDIANSYAH-PT-2 }

<=====>
|          SELAMAT DATANG DIPROGRAM BANK SEDERHANA          |
<=====>
|          1. Login Sebagai Admin          |
|          2. Login Sebagai User          |
|          0. Keluar          |
<=====>
Pilih Menu: 1
Masukkan Username: admin
Masukkan Password: admin123
<=====>
|          MENU ADMIN          |
<=====>
|          1. Buat Rekening          |
|          2. Hapus Rekening          |
|          3. Tampilkan Rekening Nasabah          |
|          0. Kembali Ke Menu Utama          |
<=====>
Pilih Menu Admin: █
```

Gambar 4.1 Login sebagai

a) Opsi Buat rekening

```
<=====>
|               MENU ADMIN               |
<=====>
|               1. Buat Rekening          |
|               2. Hapus Rekening         |
|               3. Tampilkan Rekening Nasabah |
|               0. Kembali Ke Menu Utama  |
<=====>
Pilih Menu Admin: 1
Masukkan nama nasabah: Rusdy
Masukkan nomor rekening: 1234
Rekening berhasil dibuat.
<=====>
|               MENU ADMIN               |
<=====>
|               1. Buat Rekening          |
|               2. Hapus Rekening         |
|               3. Tampilkan Rekening Nasabah |
|               0. Kembali Ke Menu Utama  |
<=====>
Pilih Menu Admin: 1
Masukkan nama nasabah: Ahnaf
Masukkan nomor rekening: 4321
Rekening berhasil dibuat.
```

Gambar 4.2 Buat rekening

b) Opsi Hapus rekening

```

<=====>
|                               |
|                               | MENU ADMIN                               |
|                               |                               |
<=====>
|                               |                               |
|                               | 1. Buat Rekening                |
|                               | 2. Hapus Rekening              |
|                               | 3. Tampilkan Rekening Nasabah |
|                               | 0. Kembali Ke Menu Utama      |
|                               |                               |
<=====>
Pilih Menu Admin: 2
Masukkan nomor rekening yang akan dihapus: 1234
Rekening berhasil dihapus.
<=====>
|                               |
|                               | MENU ADMIN                               |
|                               |                               |
<=====>
|                               |                               |
|                               | 1. Buat Rekening                |
|                               | 2. Hapus Rekening              |
|                               | 3. Tampilkan Rekening Nasabah |
|                               | 0. Kembali Ke Menu Utama      |
|                               |                               |
<=====>
Pilih Menu Admin: █

```

Gambar 4.3 Hapus rekening

c) Opsi Tampilkan rekening nasabah

```

<=====>
|                               |
|                               | MENU ADMIN                               |
|                               |                               |
<=====>
|                               |                               |
|                               | 1. Buat Rekening                |
|                               | 2. Hapus Rekening              |
|                               | 3. Tampilkan Rekening Nasabah |
|                               | 0. Kembali Ke Menu Utama      |
|                               |                               |
<=====>
Pilih Menu Admin: 3
<=====>
|                               |
|                               | DAFTAR NASABAH                               |
|                               |                               |
<=====>
| 1. Nama: Ahnaf          | Nomor Rekening: 4321          | Saldo : Rp. 0
|                               |                               |
<=====>
|                               |
|                               | MENU ADMIN                               |
|                               |                               |
<=====>
|                               |                               |
|                               | 1. Buat Rekening                |
|                               | 2. Hapus Rekening              |
|                               | 3. Tampilkan Rekening Nasabah |
|                               | 0. Kembali Ke Menu Utama      |
|                               |                               |
<=====>
Pilih Menu Admin: █

```

Gambar 4.4 Menampilkan rekening nasabah

## 2. Login sebagai user

```
PS C:\praktikum-apl\post-test\post-test-2> cd "c:\praktikum-apl\post-test\
if ($?) { .\2409106013-RUSDIANSYAH-PT-2 }
<=====>
|          SELAMAT DATANG DIPROGRAM BANK SEDERHANA          |
<=====>
|          1. Login Sebagai Admin          |
|          2. Login Sebagai User          |
|          0. Keluar          |
<=====>
Pilih Menu: 2
Masukkan Username: user
Masukkan Password: user123
<=====>
|          MENU USER          |
<=====>
|          1. Cek Saldo          |
|          2. Isi Saldo          |
|          3. Tarik Saldo          |
|          0. Kembali Ke Menu Utama          |
<=====>
Pilih Menu User: █
```

Gambar 4.5 Login sebagai user

a) Opsi Cek saldo

[illegible]

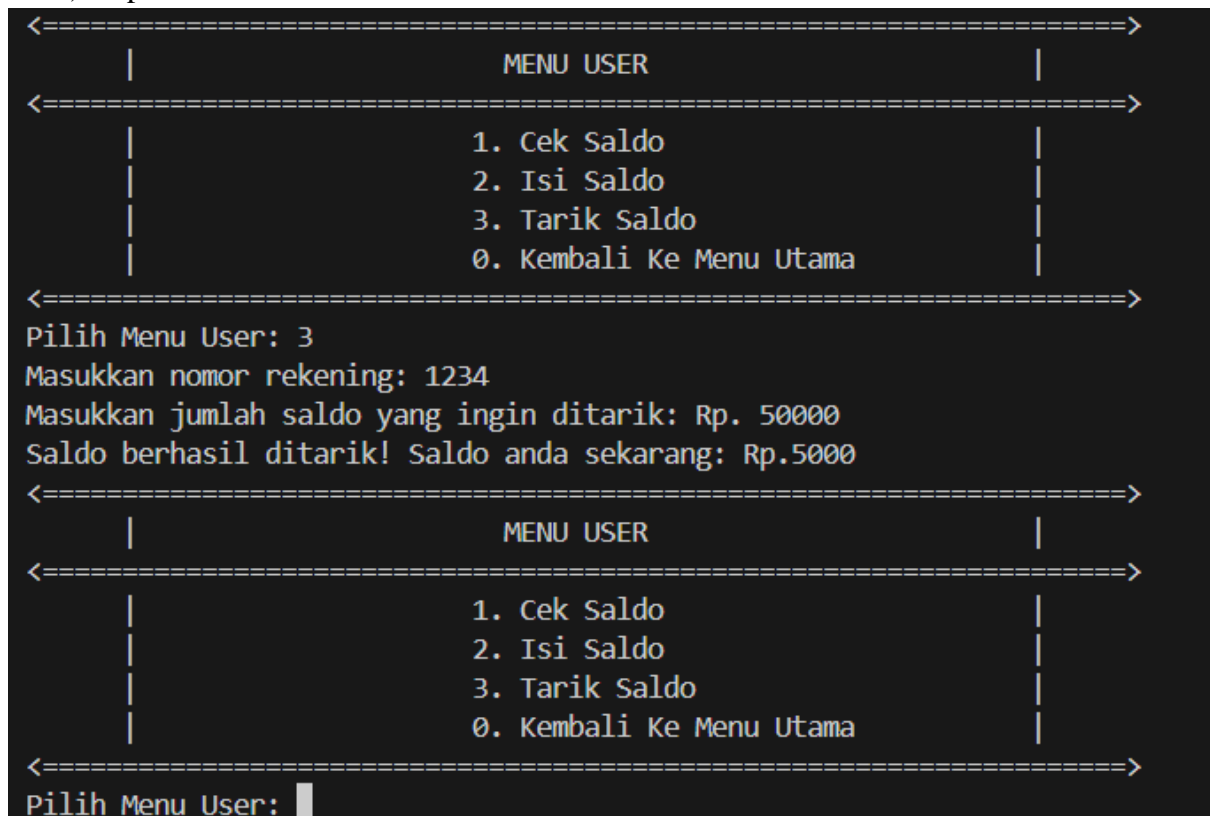
Gambar 4.6 Cek saldo

b) Opsi Isi saldo

```
<=====>  
| MENU USER |  
  
<=====>  
|          |  
| 1. Cek Saldo          |  
| 2. Isi Saldo         |  
| 3. Tarik Saldo       |  
| 0. Kembali Ke Menu Utama   |  
|          |  
  
<=====>  
Pilih Menu User: 2  
Masukkan nomor rekening: 1234  
Masukkan jumlah saldo yang ingin diisi: Rp. 55000  
Saldo berhasil diisi! Saldo anda sekarang: Rp.55000  
  
<=====>  
| MENU USER |  
  
<=====>  
|          |  
| 1. Cek Saldo          |  
| 2. Isi Saldo         |  
| 3. Tarik Saldo       |  
| 0. Kembali Ke Menu Utama   |  
|          |  
  
<=====>  
Pilih Menu User: █
```

Gambar 4.7 Isi saldo

c) Opsi Tarik saldo



Gambar 4.8 Tarik saldo

### 3. Login dengan username atau password yang salah

#### a) Login sebagai admin

```
PS C:\praktikum-apl\post-test\post-test-2> cd "c:\praktikum-apl\post-test-2"
if ($?) { .\2409106013-RUSDIANSYAH-PT-2 }

<=====>
|          SELAMAT DATANG DIPROGRAM BANK SEDERHANA          |
<=====>
|          1. Login Sebagai Admin          |
|          2. Login Sebagai User           |
|          0. Keluar                        |
<=====>
Pilih Menu: 1
Masukkan Username: admin
Masukkan Password: salah
Username atau password salah. Kesalahan ke-1.
Masukkan Username: admin
Masukkan Password: salah
Username atau password salah. Kesalahan ke-2.
Masukkan Username: admin
Masukkan Password: salah
Username atau password salah. Kesalahan ke-3.
Anda telah melakukan 3 kali kesalahan login. Program dihentikan.
PS C:\praktikum-apl\post-test\post-test-2> █
```

Gambar 4.9 Login admin gagal hingga 3 kali percobaan.



b) Login sebagai user

```
PS C:\praktikum-apl\post-test\post-test-2> cd "c:\praktikum-apl\post-test\
if ($?) { .\2409106013-RUSDIANSYAH-PT-2 }
<=====>
|          SELAMAT DATANG DIPROGRAM BANK SEDERHANA          |
<=====>
|          1. Login Sebagai Admin          |
|          2. Login Sebagai User           |
|          0. Keluar                       |
<=====>
Pilih Menu: 2
Masukkan Username: user
Masukkan Password: salah
Username atau password salah. Kesalahan ke-1.
Masukkan Username: user
Masukkan Password: salah
Username atau password salah. Kesalahan ke-2.
Masukkan Username: user
Masukkan Password: salah
Username atau password salah. Kesalahan ke-3.
Anda telah melakukan 3 kali kesalahan login. Program dihentikan.
PS C:\praktikum-apl\post-test\post-test-2> █
```

Gambar 4.10 Login user gagal hingga 3 kali percobaan

#### 4. Batas maksimal nasabah

```
<=====>
|          1. Buat Rekening          |
|          2. Hapus Rekening         |
|          3. Tampilkan Rekening Nasabah |
|          0. Kembali Ke Menu Utama  |
|                                     |
<=====>
Pilih Menu Admin: 1
Masukkan nama nasabah: Rusdy
Masukkan nomor rekening: 1234
Rekening berhasil dibuat.
<=====>
|          MENU ADMIN          |
|                                     |
<=====>
|          1. Buat Rekening          |
|          2. Hapus Rekening         |
|          3. Tampilkan Rekening Nasabah |
|          0. Kembali Ke Menu Utama  |
|                                     |
<=====>
Pilih Menu Admin: 1
Masukkan nama nasabah: Ahnaf
Masukkan nomor rekening: 4321
Rekening berhasil dibuat.
<=====>
|          MENU ADMIN          |
|                                     |
<=====>
|          1. Buat Rekening          |
|          2. Hapus Rekening         |
|          3. Tampilkan Rekening Nasabah |
|          0. Kembali Ke Menu Utama  |
|                                     |
<=====>
Pilih Menu Admin: 1
Maaf, jumlah nasabah sudah penuh. Tidak dapat menambahkan rekening baru.
```

Gambar 4.11 Batas maksimal menambahkan rekening 2 kali

5. Login dengan rekening yang sudah

```
<=====>
|                               |
|                               | MENU ADMIN                               |
|                               |                               |
<=====>
|                               |
|                               | 1. Buat Rekening                               |
|                               | 2. Hapus Rekening                               |
|                               | 3. Tampilkan Rekening Nasabah                   |
|                               | 0. Kembali Ke Menu Utama                       |
|                               |                               |
<=====>
Pilih Menu Admin: 1
Masukkan nama nasabah: Rusdy
Masukkan nomor rekening: 1234
Rekening berhasil dibuat.
<=====>
|                               |
|                               | MENU ADMIN                               |
|                               |                               |
<=====>
|                               |
|                               | 1. Buat Rekening                               |
|                               | 2. Hapus Rekening                               |
|                               | 3. Tampilkan Rekening Nasabah                   |
|                               | 0. Kembali Ke Menu Utama                       |
|                               |                               |
<=====>
Pilih Menu Admin: 1
Masukkan nama nasabah: Ahnaf
Masukkan nomor rekening: 1234
Nomor rekening sudah digunakan. Tidak dapat membuat rekening baru.
<=====>
|                               |
|                               | MENU ADMIN                               |
|                               |                               |
<=====>
|                               |
|                               | 1. Buat Rekening                               |
|                               | 2. Hapus Rekening                               |
|                               | 3. Tampilkan Rekening Nasabah                   |
|                               | 0. Kembali Ke Menu Utama                       |
|                               |                               |
<=====>
Pilih Menu Admin: █
```

Gambar 4.12 Mencoba login dengan nomor rekening yang sudah terdaftar.

6. Penarikan saldo melebihi saldo yang ada

[illegible]

Gambar 4.13 Penarikan saldo melebihi saldo yang ada.

## 7. Menampilkan daftar nasabah kosong

```
<=====>
|                               |
|                               | MENU ADMIN                               |
|                               |                               |
<=====>
|                               |                               |
|                               | 1. Buat Rekening                |
|                               | 2. Hapus Rekening              |
|                               | 3. Tampilkan Rekening Nasabah  |
|                               | 0. Kembali Ke Menu Utama      |
|                               |                               |
<=====>
Pilih Menu Admin: 2
Masukkan nomor rekening yang akan dihapus: 1234
Rekening berhasil dihapus.
<=====>
|                               |                               |
|                               | MENU ADMIN                               |
|                               |                               |
<=====>
|                               |                               |
|                               | 1. Buat Rekening                |
|                               | 2. Hapus Rekening              |
|                               | 3. Tampilkan Rekening Nasabah  |
|                               | 0. Kembali Ke Menu Utama      |
|                               |                               |
<=====>
Pilih Menu Admin: 3
Belum ada rekening yang terdaftar.
<=====>
|                               |                               |
|                               | MENU ADMIN                               |
|                               |                               |
<=====>
|                               |                               |
|                               | 1. Buat Rekening                |
|                               | 2. Hapus Rekening              |
|                               | 3. Tampilkan Rekening Nasabah  |
|                               | 0. Kembali Ke Menu Utama      |
|                               |                               |
<=====>
Pilih Menu Admin: █
```

Gambar 4.14 Opsi menampilkan rekening nasabah.

## 8. Keluar dari program

```
<=====>
|          SELAMAT DATANG DIPROGRAM BANK SEDERHANA          |
<=====>
|          1. Login Sebagai Admin          |
|          2. Login Sebagai User          |
|          0. Keluar          |
<=====>
Pilih Menu: 0
Terima kasih telah menggunakan program ini.
PS C:\praktikum-apl\post-test\post-test-2>
```

Gambar 4.15 Opsi keluar dari program.

## 5.Langkah Git

### A.Membuat repository


Membuat repository baru untuk menyimpan folder.

**Create a new repository**

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)


*Required fields are marked with an asterisk (\*).*


**Owner \*** **Repository name \***

 Rsdnsyhh /

Great repository names are short and memorable. Need inspiration? How about **refactored-system** ?

**Description** (optional)

☒  **Public**  
Anyone on the internet can see this repository. You choose who can commit.

☐  **Private**  
You choose who can see and commit to this repository.

**Initialize this repository with:**

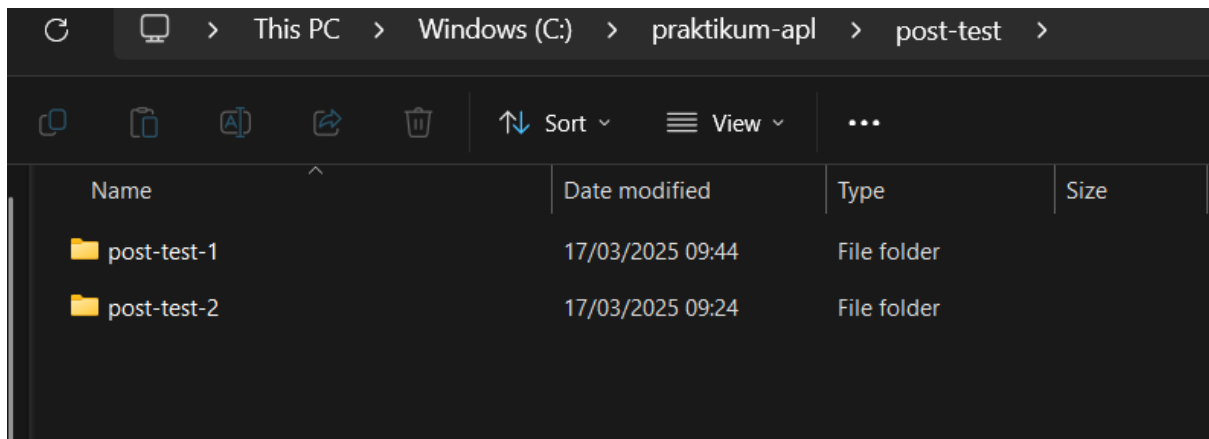
☐ **Add a README file**  
This is where you can write a long description for your project. [Learn more about READMEs.](#)

**Add .gitignore**

Gambar 5.1 New Repository

## B. Membuat folder

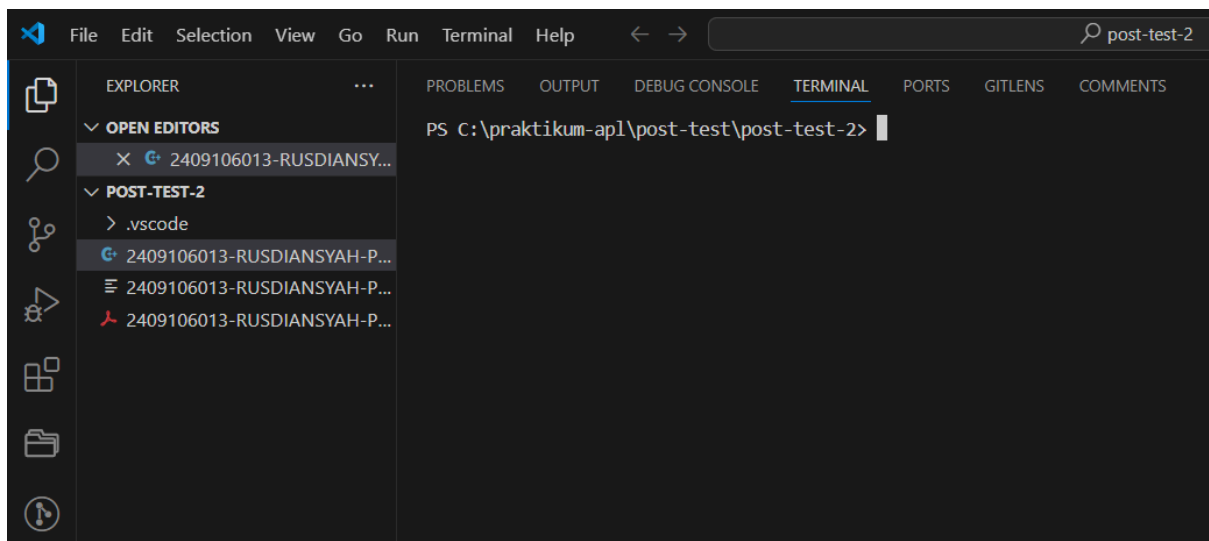
Membuat folder praktikum-apl, kemudian didalam folder praktikum-apl buat folder kelas dan post-test, dan yang terakhir didalam file post-test buat lagi folder post-test-2.



Gambar 5.2 Membuat folder post-test-2

## C. Membuka VS Code dan terminal

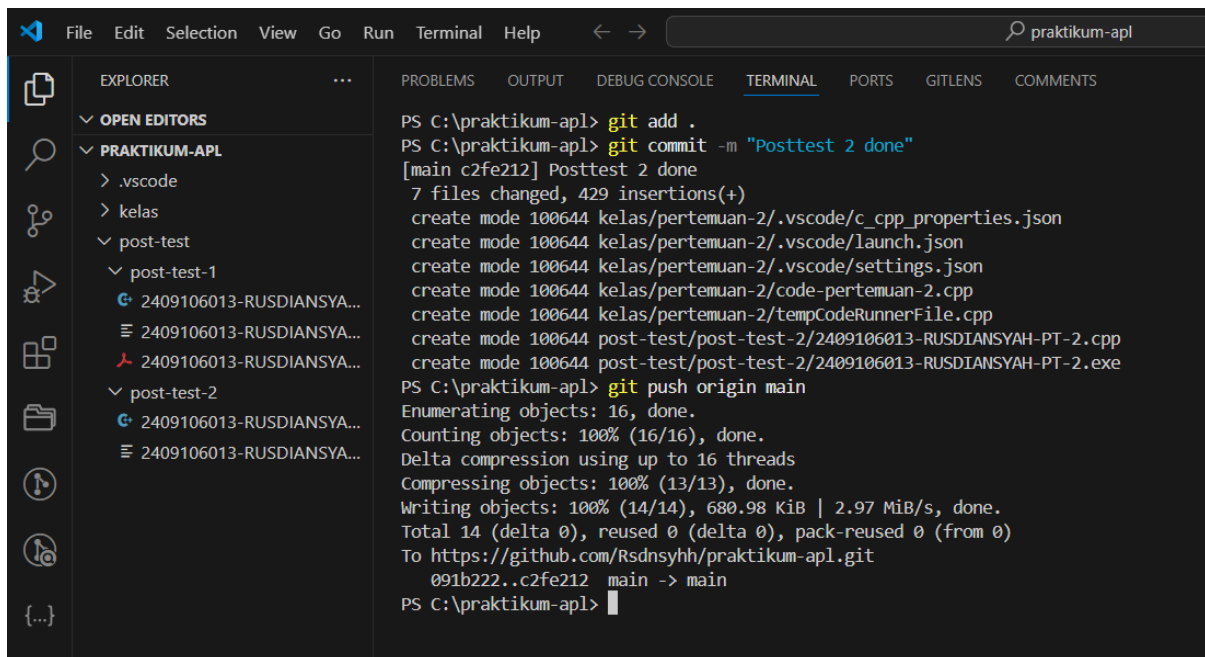
Buka VS Code dan terminal untuk membuat perintah Git.



Gambar 5.3 Terminal VS Code

#### D. Git init, Git add, dan commit

Git init merupakan perintah untuk menginisialisasi repositori Git baru di direktori proyek user. Git add merupakan perintah digunakan untuk menambahkan file ke staging area (area persiapan sebelum perubahan tersebut di commit). Commit adalah perintah yang digunakan untuk menyimpan perubahan yang telah di staging ke dalam riwayat repositori Git. Git push adalah perintah yang digunakan untuk mengirim perubahan yang telah di commit di repositori lokal ke repositori remote.

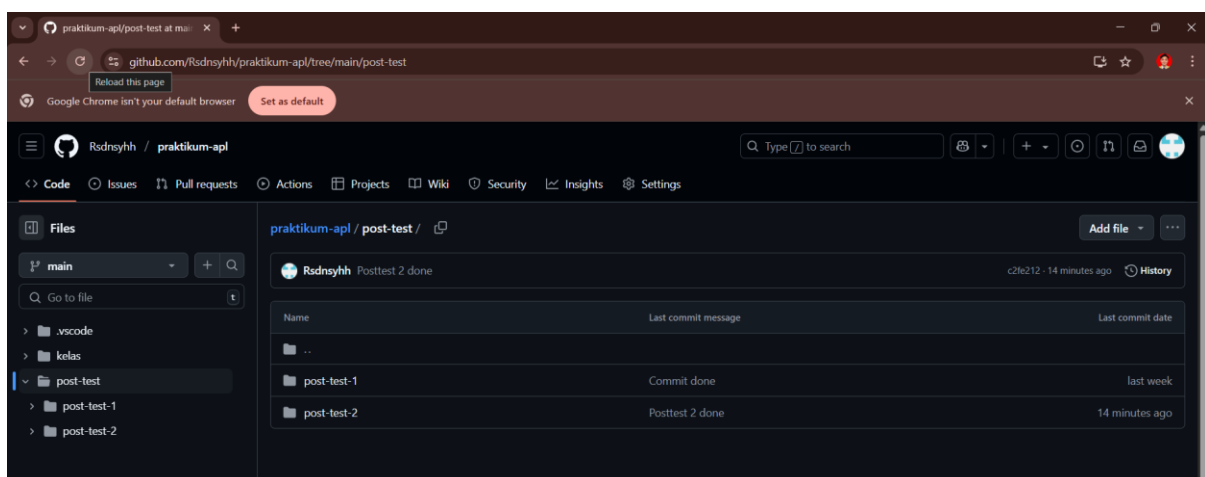


```
PS C:\praktikum-apl> git add .
PS C:\praktikum-apl> git commit -m "Posttest 2 done"
[main c2fe212] Posttest 2 done
7 files changed, 429 insertions(+)
create mode 100644 kelas/pertemuan-2/.vscode/c_cpp_properties.json
create mode 100644 kelas/pertemuan-2/.vscode/launch.json
create mode 100644 kelas/pertemuan-2/.vscode/settings.json
create mode 100644 kelas/pertemuan-2/code-pertemuan-2.cpp
create mode 100644 kelas/pertemuan-2/tempCodeRunnerFile.cpp
create mode 100644 post-test/post-test-2/2409106013-RUSDIANSYAH-PT-2.cpp
create mode 100644 post-test/post-test-2/2409106013-RUSDIANSYAH-PT-2.exe
PS C:\praktikum-apl> git push origin main
Enumerating objects: 16, done.
Counting objects: 100% (16/16), done.
Delta compression using up to 16 threads
Compressing objects: 100% (13/13), done.
Writing objects: 100% (14/14), 680.98 KiB | 2.97 MiB/s, done.
Total 14 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/Rsdnsyhh/praktikum-apl.git
   091b222..c2fe212  main -> main
PS C:\praktikum-apl>
```

Gambar 5.4 Git init, Git add, commit, dan Git push

#### E. Refresh/Reload pada Github

Lakukan refresh atau reload pada tab browser atau chrome Github yang digunakan kemudian selesai.



Gambar 5.5 Refresh tab Github