

LAPORAN PRAKTIKUM
POSTTEST 3
ALGORITMA PEMROGRAMAN LANJUT



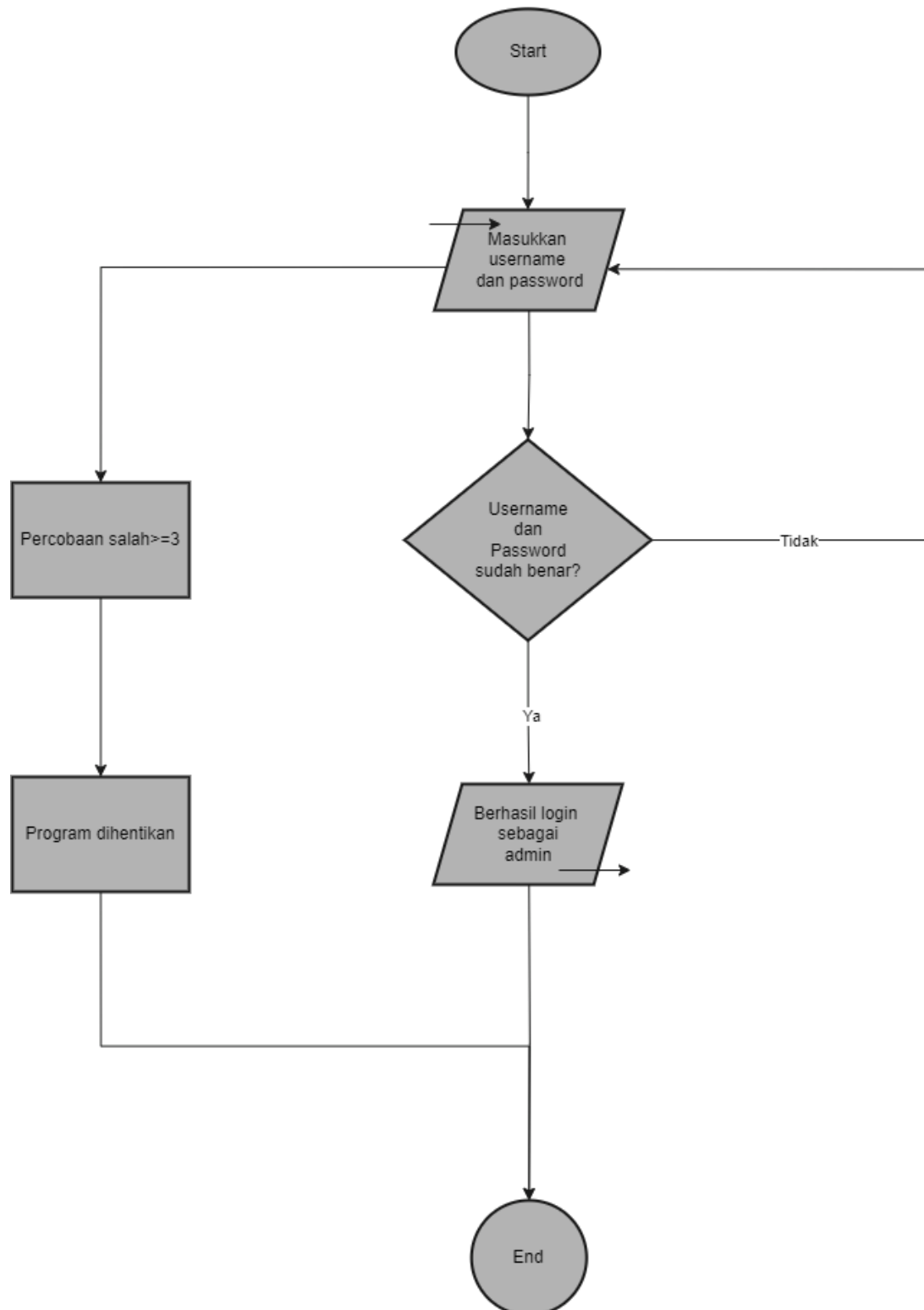
Disusun oleh:

Nama (2409106013)

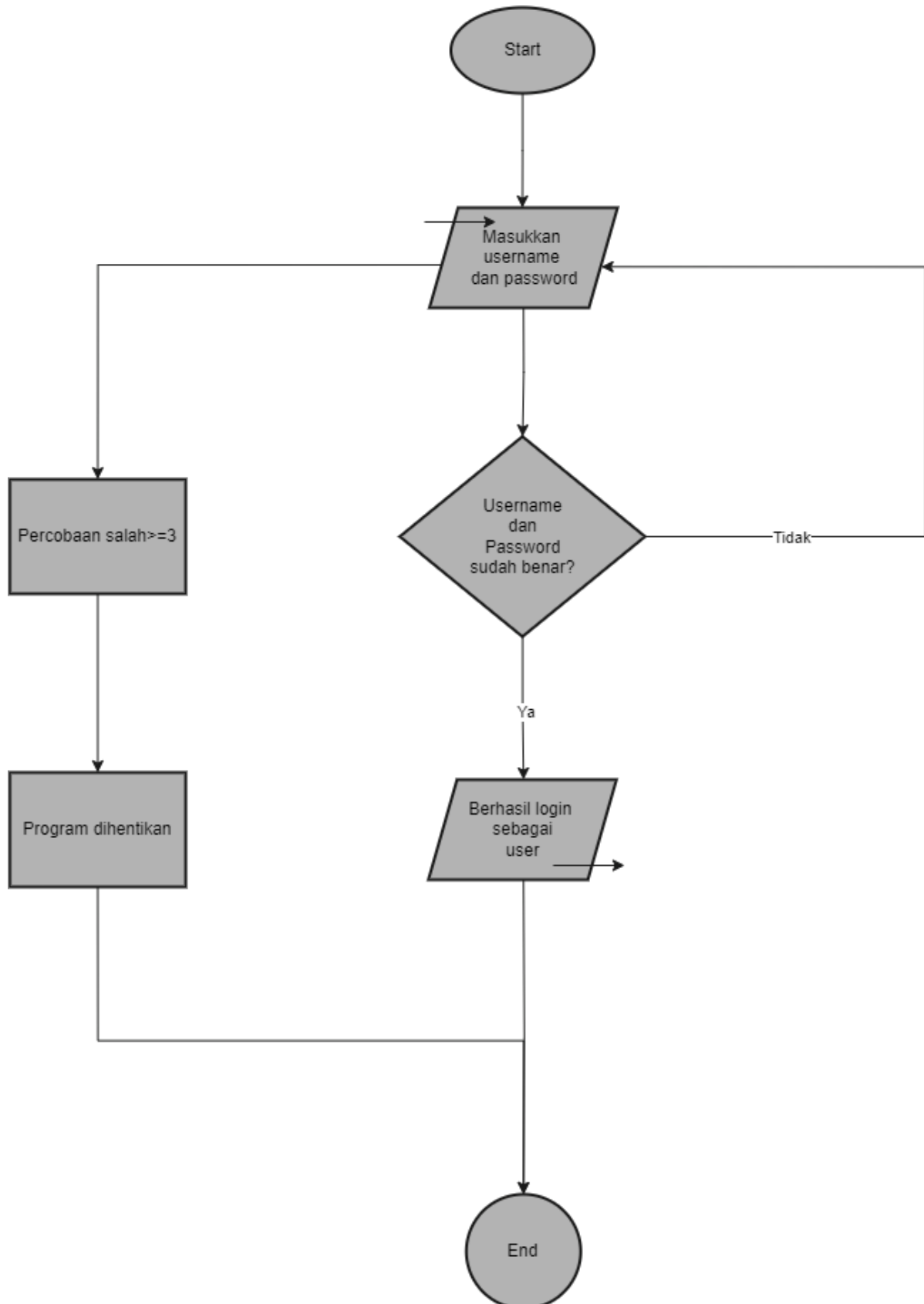
Kelas (A1 '24)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

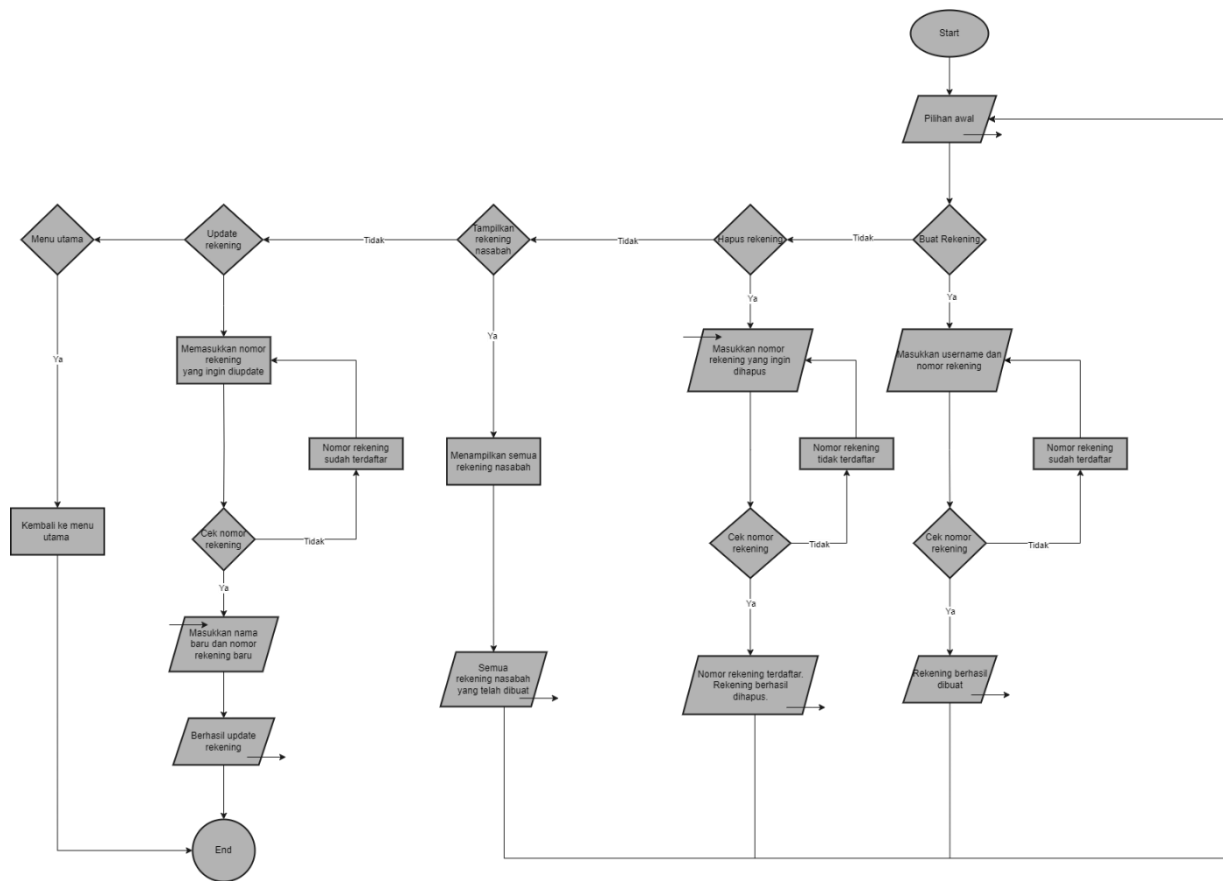
1. Flowchart



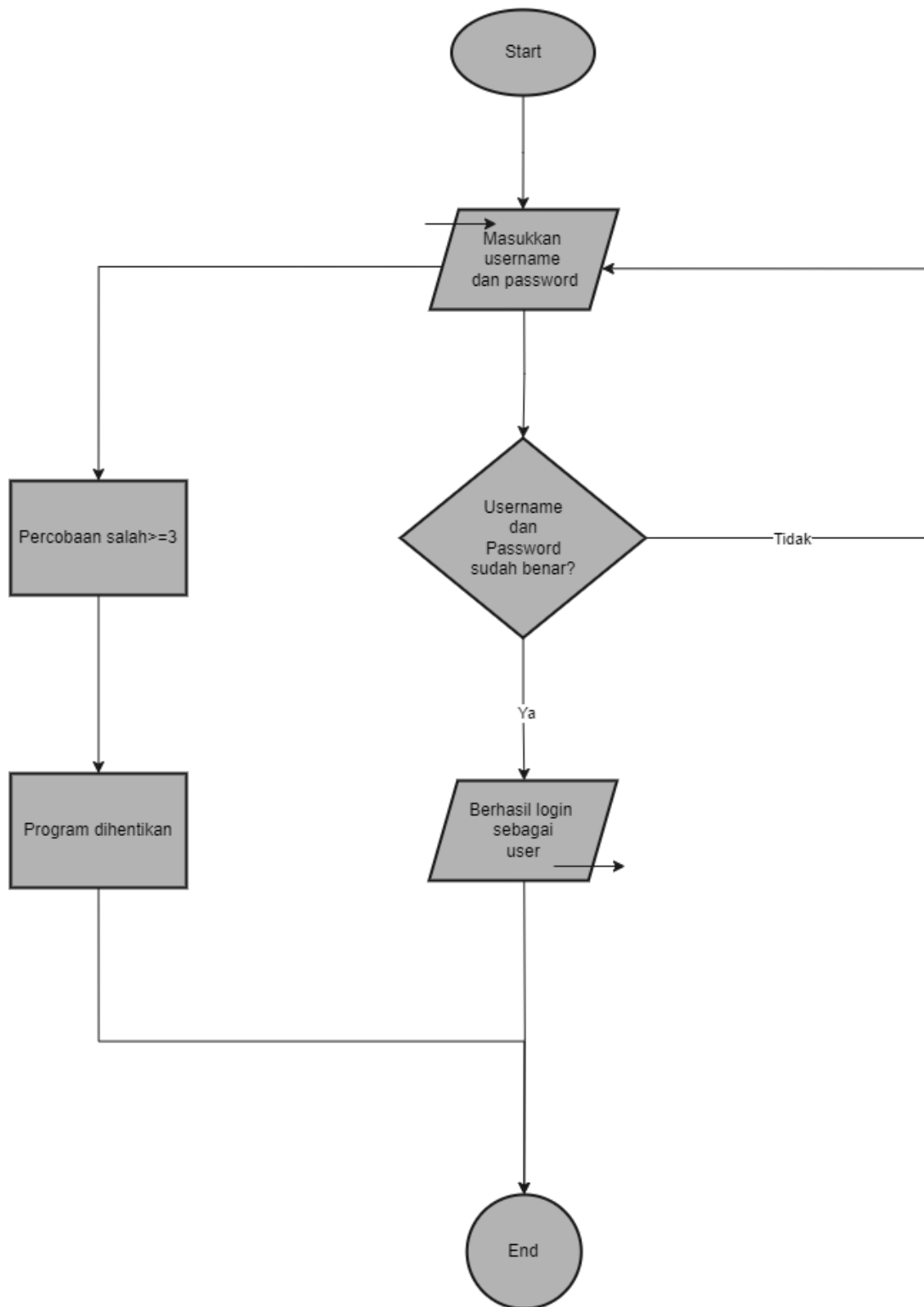
Gambar 1.1 Menu utama



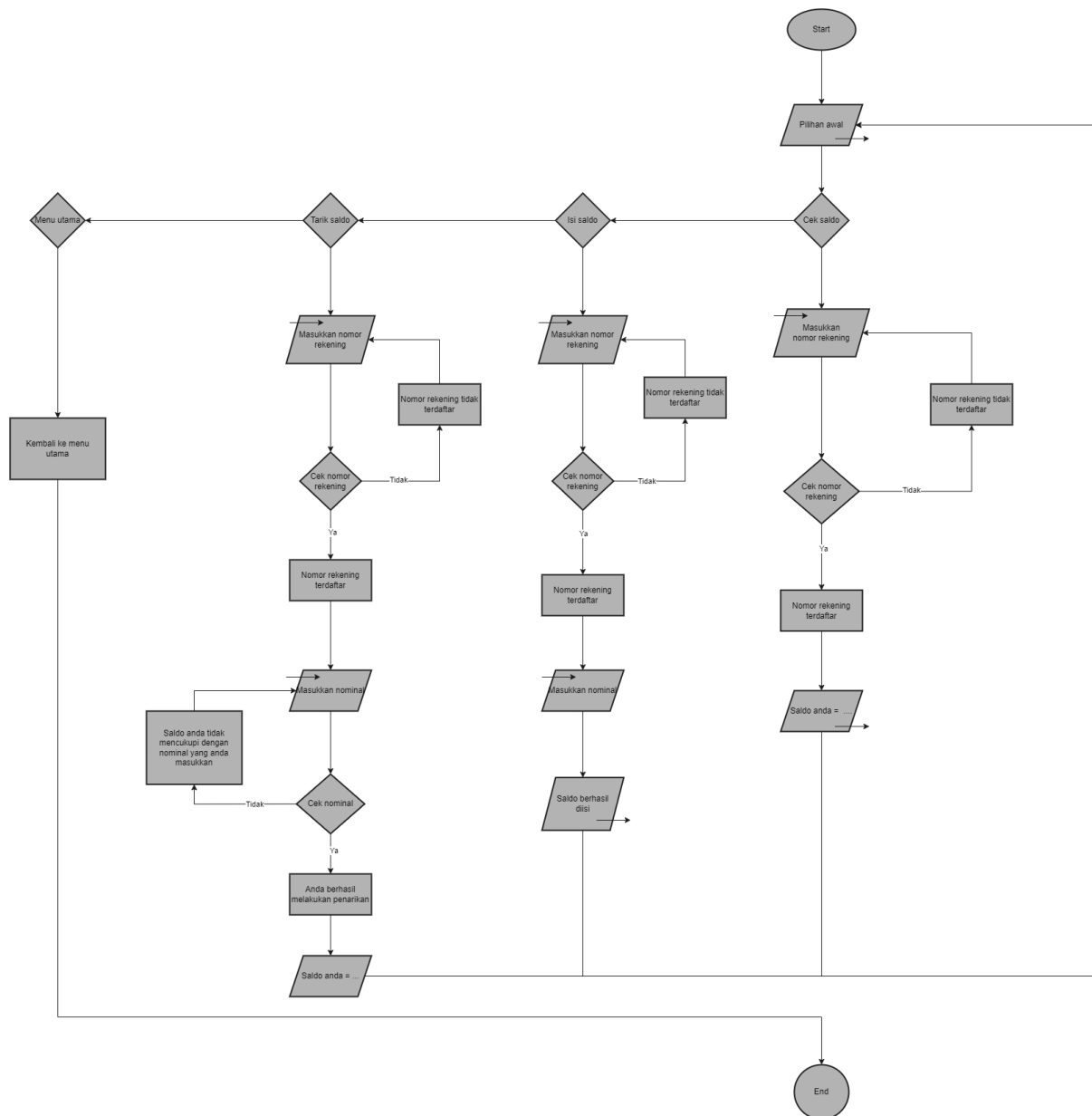
Gambar 1.2 Login sebagai admin



Gambar 1.3 Menu admin



Gambar 1.4 Login sebagai user



Gambar 1.5 Menu user

2. Analisis Program

Program ini merupakan simulasi sistem perbankan sederhana yang memungkinkan user atau pengguna untuk melakukan operasi dasar seperti pembuatan rekening, update 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, update, dan menampilkan data rekening.

Sedangkan user dapat mengakses informasi saldo, mengisi saldo, dan menarik saldo.

3. Source Code

1) Create

```

                                case 1: //Membuat rekening//
if (panjang >= MAX_NASABAH) {                                cout <<
"Maaf jumlah nasabah sudah penuh.
Tidak dapat menambahkan rekening baru." << endl;
                                } else {
                                    string nama, nomorRekening;
cout << "Masukkan nama nasabah : " << endl;
cin >> nama;
                                    cout << "Masukkan nomor rekening : "
<< endl;
                                    cin >> nomorRekening;

bool nomorSudahDigunakan = false;                                for
(int i = 0; i < panjang; i++) {                                if
(dataNasabah[i].nomorRekening == nomorRekening) {
nomorSudahDigunakan = true;
break;
                                }
                                }

                                if (nomorSudahDigunakan) {
cout << "Nomor rekening sudah digunakan. Tidak dapat membuat rekening baru." <<
endl;
                                } else {
dataNasabah[panjang].nama = nama;
dataNasabah[panjang].nomorRekening = nomorRekening;
dataNasabah[panjang].saldo = 0;
panjang++;
dibuat." << endl;
                                cout << "Rekening berhasil
                                }
}
break;

```

Gambar 3.1 Membuat rekening

2) Read

```

                                case 3: //Menampilkan daftar rekening//
if (panjang == 0) {                                cout << "Belum ada
rekening yang terdaftar.";
                                } else {
cout <<
"<=====>" <<

```

```

endl;
                                cout << "      |
DAFTAR NASABAH                |      " << endl;
cout <<
"<=====>" <<
endl;                                for (int i = 0; i < panjang; i++) {
cout << "      | " << i + 1 << ". Nama :
" << dataNasabah[i].nama
                                << " | Nomor Rekening : " <<
dataNasabah[i].nomorRekening
                                << " | Saldo : Rp." <<
dataNasabah[i].saldo << endl;
                                }
cout <<
"<=====>"
<< endl;
                                }
break;

```

Gambar 3.2 Menampilkan daftar rekening

```

                                case 1: //Cek saldo yg ada didalam rekening//
if (panjang == 0) {                                cout << "Belum ada rekening
yang terdaftar." << endl;
                                } else {
string nomorRekening;                                cout <<
"Masukkan nomor rekening : ";                                cin >>
nomorRekening;

                                bool ditemukan = false;
for (int i = 0; i < panjang; i++) {
if (dataNasabah[i].nomorRekening == nomorRekening) {
cout << "Nama : " << dataNasabah[i].nama << endl;
cout << "Nomor Rekening : " << dataNasabah[i].nomorRekening << endl;
cout << "Saldo : " << dataNasabah[i].saldo << endl;
ditemukan = true;                                break;
                                }
                                }

                                if (!ditemukan) {
cout << "Nomor rekening tidak
ditemukan." << endl;

```



```

break;
}
}

```

Gambar 3.3 Cek saldo dalam rekening

3) Update

```

case 4: //Pilihan untuk mengupdate rekening, seperti
mengubah nama dan nomor rekening yang sudah terdaftar sebelumnya//
if (panjang == 0) {
    cout << "Belum ada rekening
yang terdaftar." << endl;
} else {
    string nomorRekening;
    cout << "Masukkan nomor rekening yg ingin anda update : ";
    cin >> nomorRekening;

    bool ditemukan = false;
    for (int i = 0; i < panjang; i++) {
        if (dataNasabah[i].nomorRekening == nomorRekening) {
            cout << "Masukkan nama baru : ";
            cin >> dataNasabah[i].nama;
            cout << "Masukkan nomor rekening baru : ";
            cin >> dataNasabah[i].nomorRekening;
            cout << "Rekening berhasil diupdate." << endl;
            ditemukan = true;
            break;
        }
    }

    if (!ditemukan) {
        cout << "Nomor rekening tidak ditemukan." << endl;
    }
}
break;

```

Gambar 3.4 Mengupdate rekening

```

                                case 2: //Mengisi saldo//                                if
(panjang == 0) {                                cout << "Belum ada rekening
yang terdaftar." << endl;
                                } else {
                                    string nomorRekening;
                                    cout << "Masukkan nomor
                                    cin >> nomorRekening;
                                    cout << "Masukkan jumlah saldo yg ingin
                                    cin >> jumlah;

                                    bool    ditemukan    =    false;
for (int i= 0; i < panjang; i++){                                if
(dataNasabah[i].nomorRekening == nomorRekening) {
                                    dataNasabah[i].saldo += jumlah;
cout << "Saldo berhasil diisi! Saldo anda sekarang : Rp." <<
dataNasabah[i].saldo << endl;
ditemukan = true;                                break;

                                    }
                                }

                                if    (!ditemukan)    {
cout << "Nomor rekening tidak ditemukan." << endl;
                                }
}
break;

```

Gambar 3.5 Isi saldo

```

                                case 3: //Menarik saldo//                                if
(panjang == 0) {                                cout << "Belum ada rekening
yang terdaftar." << endl;
                                } else {
                                    string nomorRekening;                                int jumlah;
                                    cout << "Masukkan nomor rekening :";                                cin >>
nomorRekening;                                cout << "Masukkan jumlah saldo
yg ingin anda tarik : Rp.";
                                    cin >> jumlah;

                                    bool ditemukan = false;

```

```

                                for (int i = 0; i < panjang; i++) {
                                    if (dataNasabah[i].nomorRekening == nomorRekening) {
                                        if (dataNasabah[i].saldo >= jumlah) {
                                            dataNasabah[i].saldo -= jumlah;
                                            cout << "Saldo berhasil ditarik!
Saldo anda sekarang : Rp." << dataNasabah[i].saldo << endl;
                                        } else {
                                            cout << "Saldo tidak
cukup." << endl;
                                        }
                                    }
                                }
ditemukan = true;
break;

```

Gambar 3.6 Menarik saldo

4) Delete

```

                                case 2: //Menghapus rekening//
if (panjang == 0) {
                                cout << "Belum ada rekening yang terdaftar."
<< endl;
                                } else {
                                    string
nomorRekening;
                                    cout << "Masukkan nomor
rekening yang ingin dihapus : ";
                                    cin >> nomorRekening;

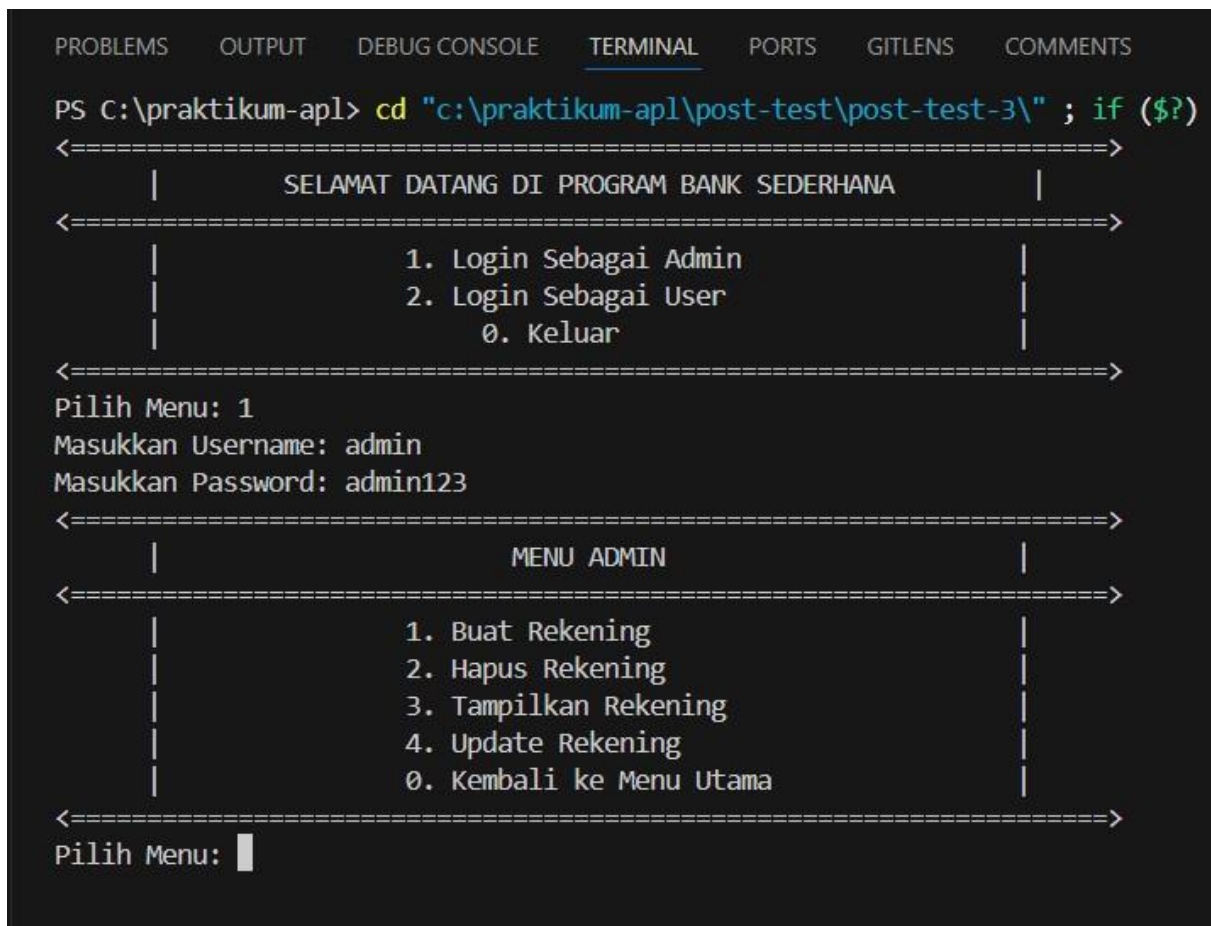
                                    bool ditemukan = false;
for (int i = 0; i < panjang; i++) {
                                if
                                (dataNasabah[i].nomorRekening == nomorRekening) {
                                    for (int j = i; j < panjang - 1; j++) {
                                        dataNasabah[j] = dataNasabah[j + 1];
                                    }
                                    panjang--;
                                    ditemukan = true;
                                    cout << "Rekening berhasil dihapus."
<< endl;
                                    break;
                                }
                            }

                                if (!ditemukan) {
                                    cout << "Nomor rekening tidak ditemukan." << endl;
                                }
                            }
break;

```

Gambar 3.7 Menghapus rekening

4. Uji Coba dan Hasil Output



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  GITLENS  COMMENTS

PS C:\praktikum-apl> cd "c:\praktikum-apl\post-test\post-test-3\" ; if ($?)
<=====>
|          SELAMAT DATANG DI PROGRAM 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          |
|          4. Update Rekening          |
|          0. Kembali ke Menu Utama          |
<=====>
Pilih Menu: █
```

Gambar 4.1 Login sebagai admin

```
<=====>
|           MENU ADMIN           |
<=====>
|           1. Buat Rekening      |
|           2. Hapus Rekening    |
|           3. Tampilkan Rekening|
|           4. Update Rekening   |
|           0. Kembali ke Menu Utama |
<=====>
Pilih Menu: 1
Masukkan nama nasabah : Ahnap
Masukkan nomor rekening : 1234
Rekening berhasil dibuat.
<=====>
|           MENU ADMIN           |
<=====>
|           1. Buat Rekening      |
|           2. Hapus Rekening    |
|           3. Tampilkan Rekening|
|           4. Update Rekening   |
|           0. Kembali ke Menu Utama |
<=====>
Pilih Menu: 3
<=====>
|           DAFTAR NASABAH       |
<=====>
| 1. Nama : Ahnap | Nomor Rekening : 1234 | Saldo : Rp.0 |
<=====>
```

Gambar 4.2 Membuat rekening dan menampilkan rekening nasabah.

```
<=====>
|                               |
|                               | MENU ADMIN                               |
|                               |                               |
<=====>
|                               |                               |
|                               | 1. Buat Rekening                |
|                               | 2. Hapus Rekening              |
|                               | 3. Tampilkan Rekening          |
|                               | 4. Update Rekening             |
|                               | 0. Kembali ke Menu Utama      |
|                               |                               |
<=====>
Pilih Menu: 4
Masukkan nomor rekening yg ingin anda update : 1234
Masukkan nama baru : Melchi
Masukkan nomor rekening baru : 4321
Rekening berhasil diupdate.
<=====>
|                               |
|                               | MENU ADMIN                               |
|                               |                               |
<=====>
|                               |                               |
|                               | 1. Buat Rekening                |
|                               | 2. Hapus Rekening              |
|                               | 3. Tampilkan Rekening          |
|                               | 4. Update Rekening             |
|                               | 0. Kembali ke Menu Utama      |
|                               |                               |
<=====>
Pilih Menu: 3
<=====>
|                               |                               |
|                               | DAFTAR NASABAH                 |
|                               |                               |
<=====>
| 1. Nama : Melchi | Nomor Rekening : 4321 | Saldo : Rp.0 |
|                               |                               |
<=====>
<=====>
|                               |
|                               | MENU ADMIN                               |
|                               |                               |
<=====>
|                               |                               |
|                               | 1. Buat Rekening                |
|                               | 2. Hapus Rekening              |
|                               | 3. Tampilkan Rekening          |
|                               | 4. Update Rekening             |
|                               | 0. Kembali ke Menu Utama      |
|                               |                               |
<=====>
Pilih Menu: █
```

Gambar 4.3 Update rekening dan menampilkan daftar rekening

```
<=====>
Pilih Menu: 0
<=====>
|          SELAMAT DATANG DI PROGRAM 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 4.4 Kembali ke menu utama, login sebagai user

```
<=====>
|          MENU USER          |
<=====>
|          1. Cek Saldo          |
|          2. isi Saldo          |
|          3. Tarik Saldo          |
|          0. Kembali Ke Menu Utama          |
<=====>
Pilih Menu User :2
Masukkan nomor rekening : 4321
Masukkan jumlah saldo yg 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          |
<=====>
```

Gambar 4.5 Isi saldo


```
<=====>
|                MENU USER                |
<=====>
|                1. Cek Saldo                |
|                2. isi Saldo                |
|                3. Tarik Saldo              |
|                0. Kembali Ke Menu Utama   |
<=====>
Pilih Menu User :1
Masukkan nomor rekening : 3
Nomor rekening tidak ditemukan.
```

Gambar 4.6 Tidak menemukan nomor rekening yang tidak terdaftar.

```
<=====>
Pilih Menu User :3
Masukkan nomor rekening :4321
Masukkan jumlah saldo yg ingin anda tarik : Rp.56000
Saldo tidak cukup.
<=====>
|                MENU USER                |
<=====>
|                1. Cek Saldo                |
|                2. isi Saldo                |
|                3. Tarik Saldo              |
|                0. Kembali Ke Menu Utama   |
<=====>
```

Gambar 4.7 Tidak dapat ditarik, karena nominal melebihi saldo yang ada direkening.

```
<=====>
Pilih Menu User :3
Masukkan nomor rekening :4321
Masukkan jumlah saldo yg ingin anda tarik : Rp.50000
Saldo berhasil ditarik! Saldo anda sekarang : Rp.5000
<=====>
|                MENU USER                |
<=====>
|                1. Cek Saldo                |
|                2. isi Saldo                |
|                3. Tarik Saldo              |
|                0. Kembali Ke Menu Utama   |
<=====>
Pilih Menu User :█
```

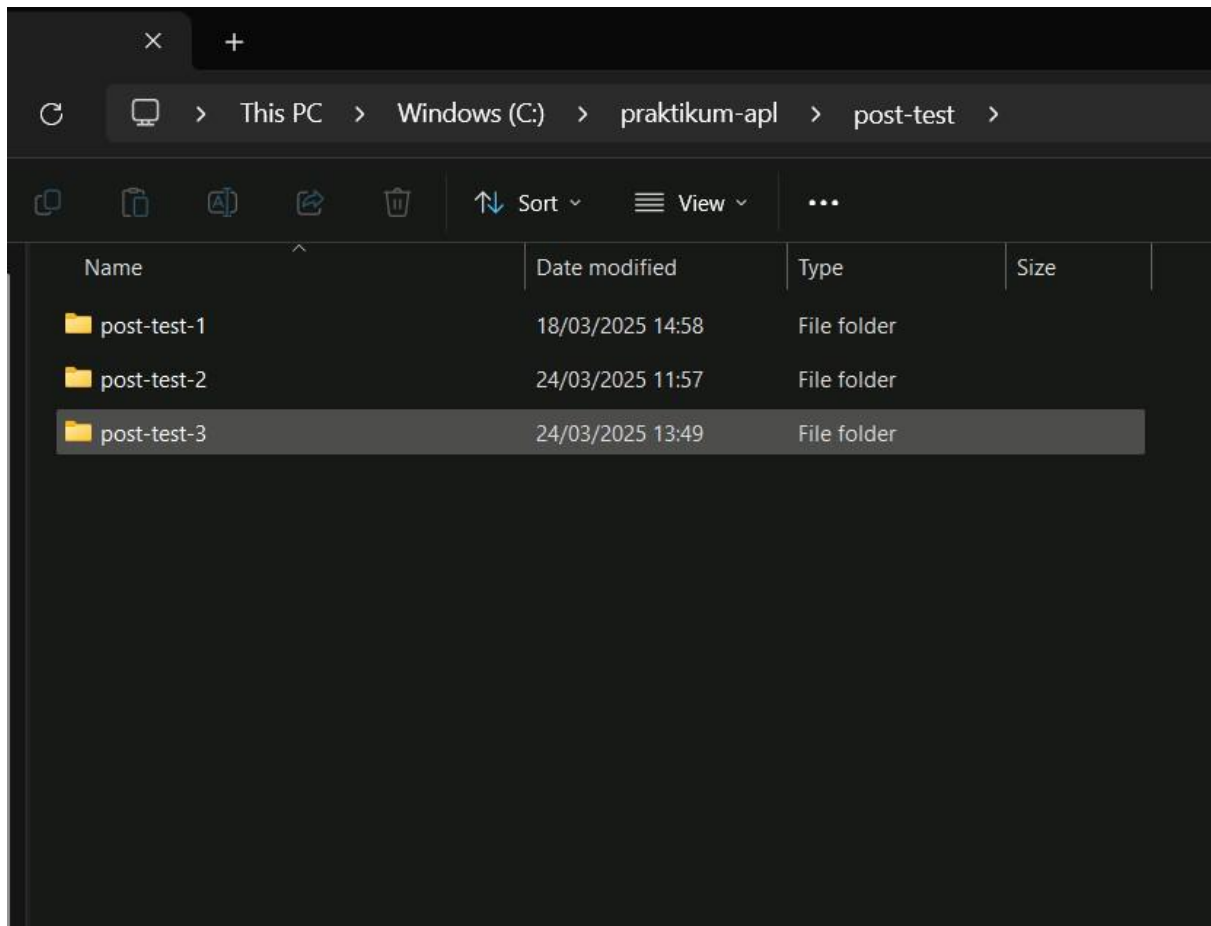
Gambar 4.8 Tarik saldo berhasil.


```
<=====>
|                MENU USER                |
<=====>
|                1. Cek Saldo                |
|                2. isi Saldo                |
|                3. Tarik Saldo              |
|                0. Kembali Ke Menu Utama    |
|                |                          |
<=====>
Pilih Menu User :0
<=====>
|                SELAMAT DATANG DI PROGRAM BANK SEDERHANA                |
<=====>
|                1. Login Sebagai Admin                |
|                2. Login Sebagai User                |
|                0. Keluar                            |
|                |                                    |
<=====>
Pilih Menu: 0
Terimakasih telah menggunakan program ini.
```

Gambar 4.9 Keluar dari menu user dan program (berhenti dari program)

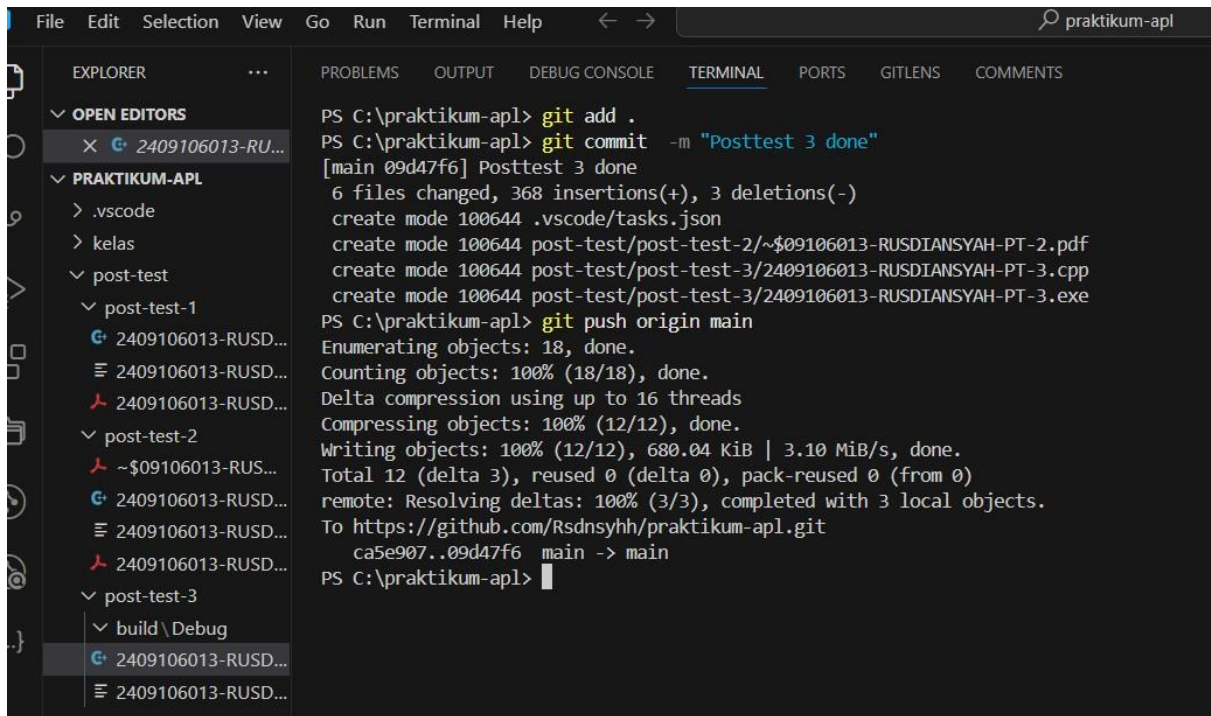
5. Langkah-Langkah Git pada VSCode.

- a) 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-3.



Gambar 5.1 Membuat folder post-test-3.

- b) Buka VS Code kemudian buka terminal. 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.



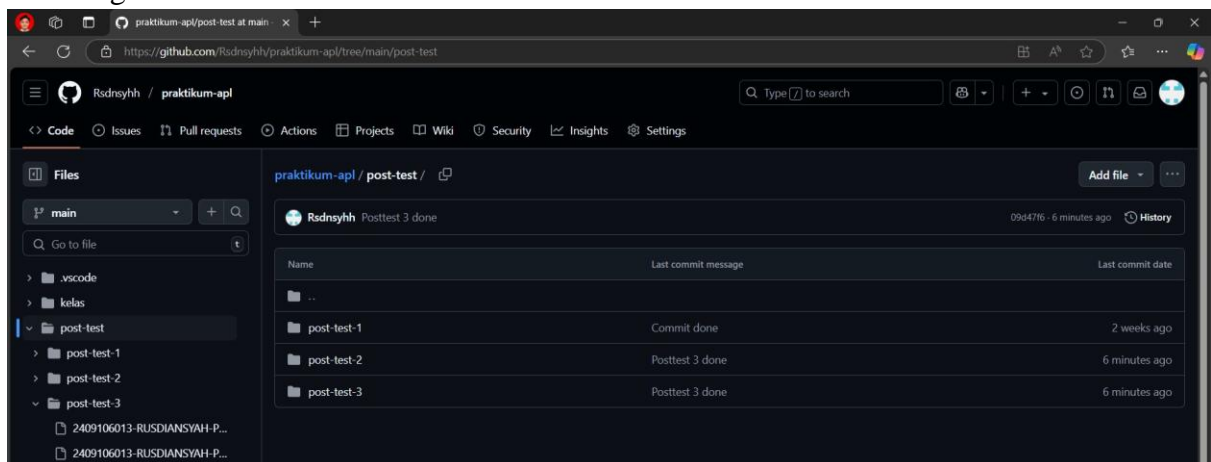
```
File Edit Selection View Go Run Terminal Help ← → praktikum-apl

EXPLORER
OPEN EDITORS
2409106013-RU...
PRAKTIKUM-APL
.vscode
kelas
post-test
post-test-1
2409106013-RUSD...
2409106013-RUSD...
2409106013-RUSD...
post-test-2
~$09106013-RUS...
2409106013-RUSD...
2409106013-RUSD...
post-test-3
build\Debug
2409106013-RUSD...
2409106013-RUSD...

TERMINAL
PS C:\praktikum-apl> git add .
PS C:\praktikum-apl> git commit -m "Posttest 3 done"
[main 09d47f6] Posttest 3 done
6 files changed, 368 insertions(+), 3 deletions(-)
create mode 100644 .vscode/tasks.json
create mode 100644 post-test/post-test-2/~$09106013-RUSDIANSYAH-PT-2.pdf
create mode 100644 post-test/post-test-3/2409106013-RUSDIANSYAH-PT-3.cpp
create mode 100644 post-test/post-test-3/2409106013-RUSDIANSYAH-PT-3.exe
PS C:\praktikum-apl> git push origin main
Enumerating objects: 18, done.
Counting objects: 100% (18/18), done.
Delta compression using up to 16 threads
Compressing objects: 100% (12/12), done.
Writing objects: 100% (12/12), 680.04 KiB | 3.10 MiB/s, done.
Total 12 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To https://github.com/Rsdnsyhh/praktikum-apl.git
ca5e907..09d47f6 main -> main
PS C:\praktikum-apl>
```

Gambar 5.2 Git init, Git add, commit, dan Git push

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



Gambar 5.3 Refresh tab Github