

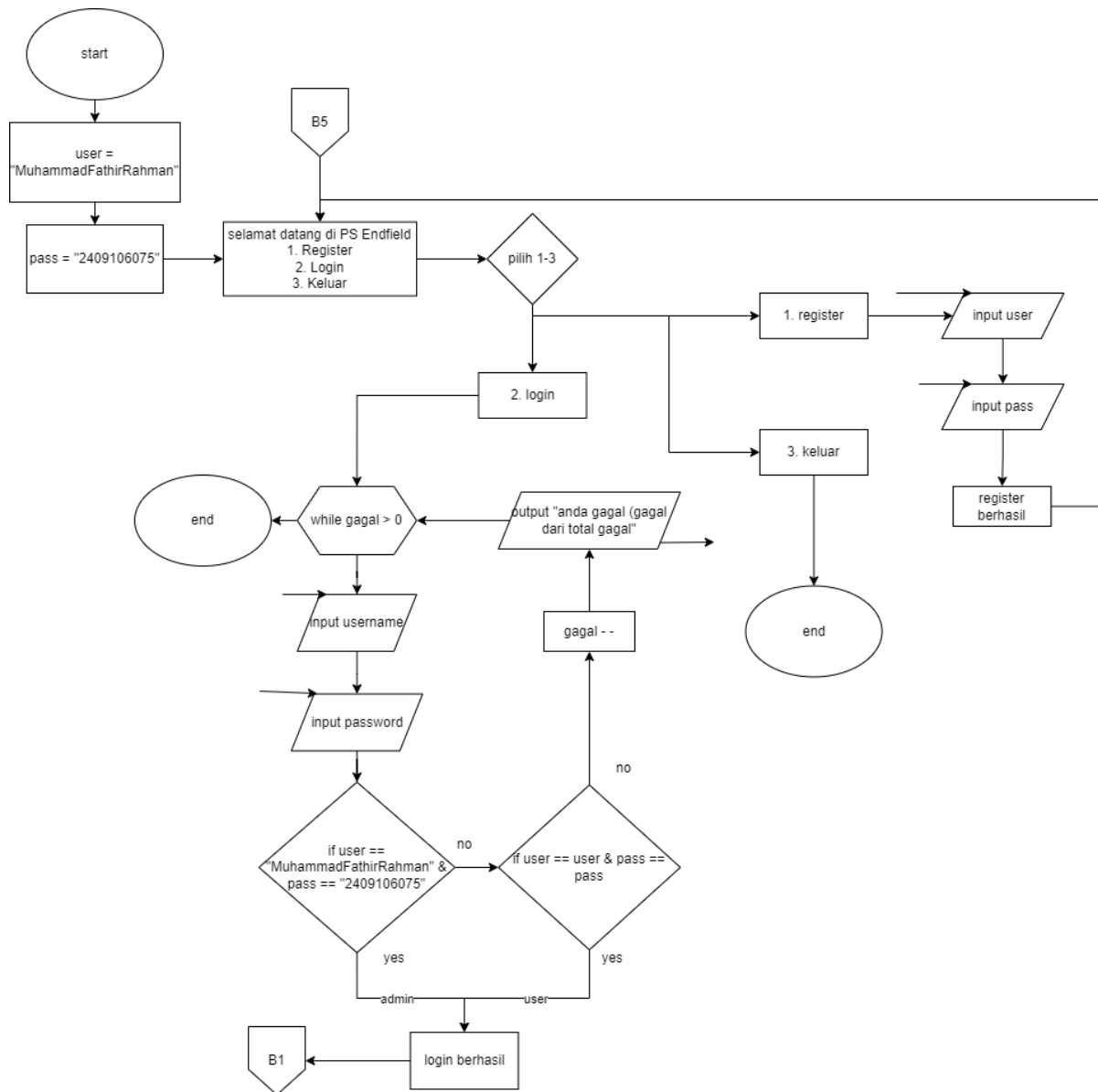
LAPORAN PRAKTIKUM
POSTTEST (5)
ALGORITMA PEMROGRAMAN LANJUT



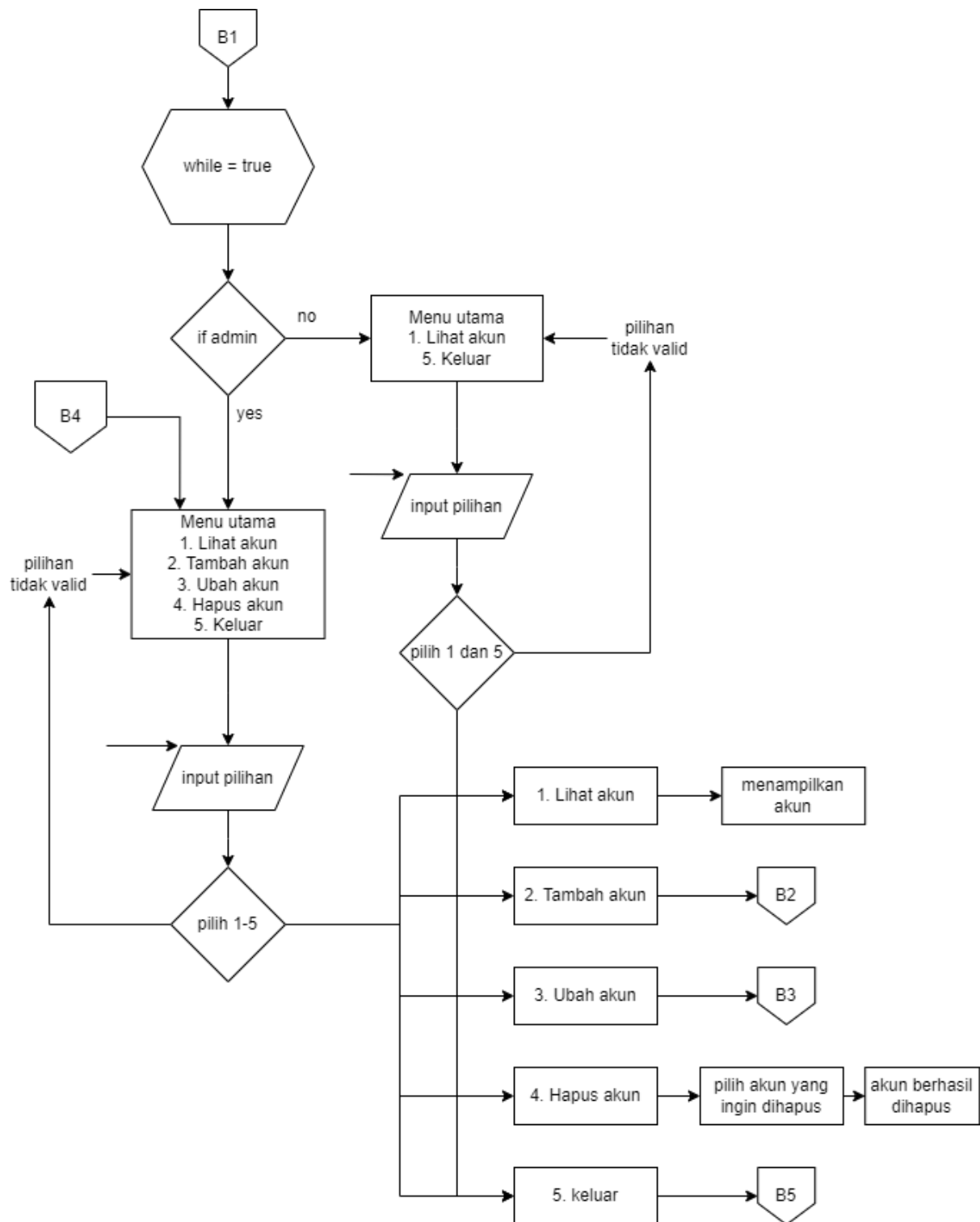
Disusun oleh:
Muhammad Fathir Rahman (2409106075) Kelas
(B2'24)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN SAMARINDA
2025

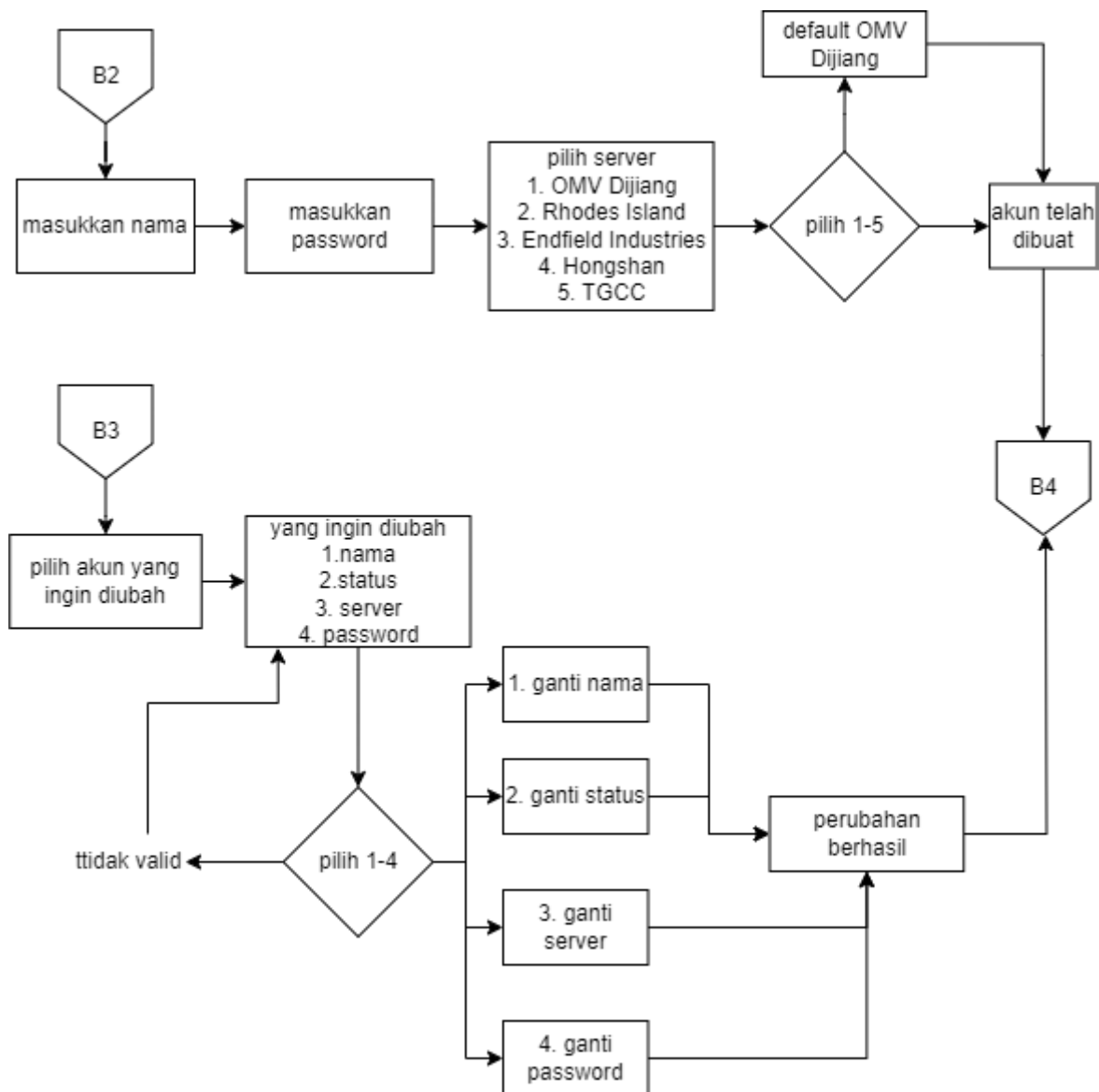
1. Flowchart



Gambar 1.1 Register dan Login



Gambar 1.2 Menu utama



Gambar 1.3 Tambah akun dan Ubah akun

2. Analisis Program

Pada posttest kali ini kita akan menambahkan pointer pada program kita sebelumnya yang dimana ada beberapa fungsi yang kita masukkan juga yakni fungsi dengan parameter address-of dan dereference.

3. Source Code

A. Fitur Register

Fitur ini digunakan untuk mendaftarkan user

Sourcecode:

```
void registerUser(User users[], int &userCount) {
    if (userCount < MAX_USER) {
        cout << "Masukkan username: ";
        cin >> users[userCount].username;
        cout << "Masukkan password: ";
        cin >> users[userCount].password;
        users[userCount].role = "Commoners";

        // Menggunakan fungsi dengan parameter pointer
        incrementAkunCount(&userCount);
        cout << "Registrasi berhasil!" << endl;
    } else {
        cout << "Kapasitas pengguna penuh!" << endl;
    }
}
```

Gambar 3.1 Register

B. Fitur Login

Fitur ini sendiri digunakan untuk melakukan login

Sourcecode:

```
User* login(User users[], int userCount) {
    string username, password;
    int percobaan = 0;
    const int maksPercobaan = 3;

    while (percobaan < maksPercobaan) {
        cout << "Username: ";
        cin >> username;
        cout << "Password: ";
        cin >> password;

        for (int i = 0; i < userCount; i++) {
            if (users[i].username == username && users[i].password ==
password) {
                cout << "Login berhasil! Selamat datang, " <<
users[i].role << "!" << endl;

                // Menggunakan fungsi dengan parameter pointer
                if (users[i].role == "Commoners") {
```

```

        updateUserRole(&users[i], "Verified Commoners");
    }

    this_thread::sleep_for(chrono::seconds(2));
    return &users[i];
}

}

    percobaan++;
    cout << "Username atau password salah! Percobaan ke-" <<
    percobaan << " dari " << maksPercobaan << "." << endl;
}

    cout << "Anda telah melebihi batas percobaan login." << endl;
    keluarProgram();
    return nullptr;
}

```

Gambar 3.2 Login

C. Menu utama

Dimenu utama sendiri ada 5 pilihan sebagai admin dan 2 pilihan sebagai user.

Sourcecode:

```

int tampilkanMenu(User* user) {
    tampilkanHeader("MENU UTAMA");
    int pilihan;
    cout << "1. Tampilkan Akun" << endl;
    if (user->role == "Endministrator") {
        cout << "2. Tambah Akun" << endl;
        cout << "3. Ubah Akun" << endl;
        cout << "4. Hapus Akun" << endl;
    }
    cout << "5. Keluar" << endl;
    cout << "Pilihan: ";
    cin >> pilihan;
    return pilihan;
}

```

Gambar 3.3 Menu utama

D. Tampilkan akun

Bagian ini berfungsi untuk menampilkan akun

Sourcecode:

```

int tampilkanAkun(Akun akun[], int akunCount) {
    if (akunCount == 0) {
        cout << "Belum ada akun" << endl;
    }
}

```



```

    akun[akunCount].serverTujuan = pilihServerTujuan();
    akun[akunCount].status = "Aktif";
    akun[akunCount].user = user; // Menyimpan pointer ke user

    // Menggunakan fungsi dengan parameter pointer
    incrementAkunCount(&akunCount);
    cout << "Akun berhasil ditambahkan" << endl;
    return 1;
} else {
    cout << "Kapasitas penuh!" << endl;
    return 0;
}
}

```

Gambar 3.5 Tambah akun

F. Ubah akun

Bagian ini berfungsi untuk mengubah akun yang telah dibuat

Sourcecode:

```

int ubahAkun(Akun akun[], int akunCount, User* user) {
    int index;
    if (akunCount == 0) {
        cout << "Belum ada akun untuk diubah." << endl;
        return 0;
    }

    tampilkanAkun(akun, akunCount);
    cout << "Masukkan nomor akun yang akan diubah: ";
    cin >> index;

    if (index > 0 && index <= akunCount) {
        if (user->role == "Endministrator" || akun[index - 1].user-
>username == user->username) {
            int pilihanUbah;
            cout << "Pilih yang ingin diubah:" << endl;
            cout << "1. Nama Akun" << endl;
            cout << "2. Status Akun" << endl;
            cout << "3. Server Tujuan" << endl;
            cout << "4. Password" << endl;
            cout << "Pilihan: ";
            cin >> pilihanUbah;
            cin.ignore();

            switch (pilihanUbah) {
                case 1:
                    cout << "Masukkan nama akun baru: ";
                    getline(cin, akun[index - 1].nama);
                    break;

```

```

        case 2: {
            string statusInput;
            do {
                cout << "Masukkan status akun (Aktif/Tidak
Aktif): ";
                getline(cin, statusInput);
                if (statusInput != "Aktif" && statusInput !=
"Tidak Aktif") {
                    cout << "Status tidak valid!" << endl;
                }
            } while (statusInput != "Aktif" && statusInput !=
"Tidak Aktif");
            akun[index - 1].status = statusInput;
            break;
        }
        case 3:
            akun[index - 1].serverTujuan = pilihServerTujuan();
            break;
        case 4: {
            string passwordInput;
            do {
                cout << "Masukkan password akun baru (minimal 8
karakter): ";
                getline(cin, passwordInput);
                if (passwordInput.length() < 8) {
                    cout << "Password harus minimal 8 karakter!"
<< endl;
                }
            } while (passwordInput.length() < 8);
            akun[index - 1].password = passwordInput;
            break;
        }
        default:
            cout << "Pilihan tidak valid!" << endl;
            return 0;
    }

    cout << "Akun berhasil diubah" << endl;
    return 1;
} else {
    cout << "Anda tidak memiliki izin untuk mengubah akun ini."
<< endl;
    return 0;
}
} else {
    cout << "Nomor akun tidak valid" << endl;
    return 0;
}
}
}

```

Gambar 3.6 Ubah akun

G. Hapus akun

Bagian ini berfungsi untuk menghapus akun yang sudah ada
Sourcecode:

```
int hapusAkun(Akun akun[], int &akunCount, User* user) {
    int index;
    if (akunCount == 0) {
        cout << "Belum ada akun untuk dihapus." << endl;
        return 0;
    }

    tampilkanAkun(akun, akunCount);
    cout << "Masukkan nomor akun yang akan dihapus: ";
    cin >> index;

    if (index > 0 && index <= akunCount) {
        if (user->role == "Endministrator" || akun[index - 1].user-
>username == user->username) {
            for (int i = index - 1; i < akunCount - 1; i++) {
                akun[i] = akun[i + 1];
            }
            akunCount--;
            cout << "Akun berhasil dihapus" << endl;
            return 1;
        } else {
            cout << "Anda tidak memiliki izin untuk menghapus akun ini."
<< endl;
            return 0;
        }
    } else {
        cout << "Nomor akun tidak valid" << endl;
        return 0;
    }
}
```

Gambar 3.7 Hapus akun

H. Keluar

Ini sendiri merupakan fungsi untuk keluar dari program CRUD ini sendiri

Sourcecode:

```
case 5:
    cout << "Keluar dari menu." << endl;
    break;
default:
    cout << "Pilihan tidak valid" << endl;
    break;
}
if (pilihanMenu != 5) {
```

```
        system("pause");
    }
    } while (pilihanMenu != 5);
}
break;
case 3:
    keluarProgram();
    return 0;
default:
    cout << "Pilihan tidak valid!" << endl;
    break;
}
if (pilihan != 3) {
    system("pause");
}
}

return 0;
}
```

Gambar 3.8 Keluar

4. Uji Coba dan Hasil Output

```
=====
PS ENDFIELD MANAGEMENT
=====
1. Register
2. Login
3. Keluar
Pilihan:
```

Gambar 4.1 Metode masuk

```
=====
PS ENDFIELD MANAGEMENT
=====
1. Register
2. Login
3. Keluar
Pilihan: 1
Masukkan username: Ajimu
Masukkan password: Ajimu
Akun count bertambah menjadi: 2
Registrasi berhasil!
Press any key to continue . . .
```

Gambar 4.2 Register

```
Pilihan: 2
Username: MuhammadFathirRahman
Password: 2409106075
Login berhasil! Selamat datang, Endadministrator!
```

Gambar 4.3 Login sebagai admin

```
Username: Ajimu
Password: Ajimu
Login berhasil! Selamat datang, Commoners!
Role user Ajimu diubah menjadi Verified Commoners
```

Gambar 4.4 Login sebagai user

```
Pilihan: 2
Username: a
Password: a
Username atau password salah! Percobaan ke-1 dari 3.
Username: a
```

Gambar 4.5 gagal kurang dari 3

```
Username: a
Password: a
Username atau password salah! Percobaan ke-2 dari 3.
Username: a
Password: a
Username atau password salah! Percobaan ke-3 dari 3.
Anda telah melebihi batas percobaan login.
Program akan berhenti dalam 3 detik...
Program akan berhenti dalam 2 detik...
Program akan berhenti dalam 1 detik...
Program berhenti.
PS E:\rusak ngini rek\rusak ngini rek\.vscode\output>
```

Gambar 4.6 gagal 3 kali

```
=====
                        DASHBOARD
=====
+-----+-----+
| Server Tujuan      | Jumlah Akun      |
+-----+-----+
+-----+-----+
=====
                        MENU UTAMA
=====
1. Tampilkan Akun
2. Tambah Akun
3. Ubah Akun
4. Hapus Akun
5. Keluar
Pilihan: █
```

Gambar 4.7 menu utama admin

```

=====
DASHBOARD
=====
+-----+-----+
| Server Tujuan | Jumlah Akun |
+-----+-----+
+-----+-----+
=====

MENU UTAMA
=====
1. Tampilkan Akun
5. Keluar
Pilihan: █

```

Gambar 4.8 menu utama user

```

Pilihan: 1
+-----+-----+-----+-----+-----+-----+
| No. | Nama Akun | Status | Server Tujuan | Password | Pemilik |
+-----+-----+-----+-----+-----+-----+
| 1 | Ajimu | Aktif | Rhodes Island | AngelinaAjimu | MuhammadFathirRahman |
| 2 | Leavatain | Aktif | Rhodes Island | Leavatcream | MuhammadFathirRahman |
+-----+-----+-----+-----+-----+-----+
Press any key to continue . . . █

```

4.9 Lihat akun

```

Pilihan: 2
Masukkan nama akun: Ajimu
Masukkan password akun (minimal 8 karakter): AngelinaAjimu
Pilih server tujuan:
1. OMV Dijiang
2. Rhodes Island
3. Endfield Industries
4. Hongshan
5. TGCC
Pilihan: 2
Akun count bertambah menjadi: 1
Akun berhasil ditambahkan
Press any key to continue . . . █

```

Gambar 4.10 Tambah akun

```
Pilihan: 3
+-----+-----+-----+-----+-----+
| No. | Nama Akun | Status | Server Tujuan | Password | Pemilik |
+-----+-----+-----+-----+-----+
| 1 | Ajimu | Aktif | Rhodes Island | AngelinaAjimu | MuhammadFathirRahman |
| 2 | Leavatain | Aktif | Rhodes Island | Leavatcream | MuhammadFathirRahman |
| 3 | Avywenna | Aktif | TGCC | TGCChamber | MuhammadFathirRahman |
+-----+-----+-----+-----+-----+
Masukkan nomor akun yang akan diubah: 1
Pilih yang ingin diubah:
1. Nama Akun
2. Status Akun
3. Server Tujuan
4. Password
Pilihan: 1
Masukkan nama akun baru: Angelina
Akun berhasil diubah
Press any key to continue . . .
```

Gambar 4.11 Ubah akun

```
Pilihan: 4
+-----+-----+-----+-----+-----+
| No. | Nama Akun | Status | Server Tujuan | Password | Pemilik |
+-----+-----+-----+-----+-----+
| 1 | Angelina | Aktif | Rhodes Island | AngelinaAjimu | MuhammadFathirRahman |
| 2 | Leavatain | Aktif | Rhodes Island | Leavatcream | MuhammadFathirRahman |
| 3 | Avywenna | Aktif | TGCC | TGCChamber | MuhammadFathirRahman |
+-----+-----+-----+-----+-----+
Masukkan nomor akun yang akan dihapus: 3
Akun berhasil dihapus
Press any key to continue . . .
```

Gambar 4.12 Hapus akun

```
=====
PS ENDFIELD MANAGEMENT
=====
1. Register
2. Login
3. Keluar
Pilihan: 3
Program akan berhenti dalam 3 detik...
Program akan berhenti dalam 2 detik...
Program akan berhenti dalam 1 detik...
Program berhenti.
PS E:\rusak ngini rek\rusak ngini rek\.vscode\output>
```

Gambar 4.13 keluar program

5. Langkah-Langkah Git pada VSCode

A. Git init, Git add Git commit

Perintah Git init berfungsi untuk menginisialisasikan repo git pada folder dan Git add untuk menambah file apa saja sebelum melakukan commit dan lakukan commit setelah sudah git init dan git add

```
PS E:\Praktikum-APL> git init
Reinitialized existing Git repository in E:/Praktikum-APL/.git/
PS E:\Praktikum-APL> git add .
PS E:\Praktikum-APL> git commit -m 'selesai'
[main b48e477] selesai
2 files changed, 453 insertions(+)
create mode 100644 Post-Test/post-test-apl-5/2409106075-MuhammadFathirRahman-PT-5.cpp
create mode 100644 Post-Test/post-test-apl-5/2409106075-MuhammadFathirRahman-PT-5.exe
PS E:\Praktikum-APL>
```

Gambar 5.1 Git init, Git add dan Git Commit

B. Git push

Karena sebelumnya sudah pernah mengremote jadi kita langsung ke Git push yang berguna untuk ngepush file ke repo github

```
PS E:\Praktikum-APL> git push
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 12 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (6/6), 745.04 KiB | 3.71 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/SeiSendai/Praktikum-APL.git
f5d2ddb..b48e477 main -> main
PS E:\Praktikum-APL>
```

Gambar 5.2 Git push