

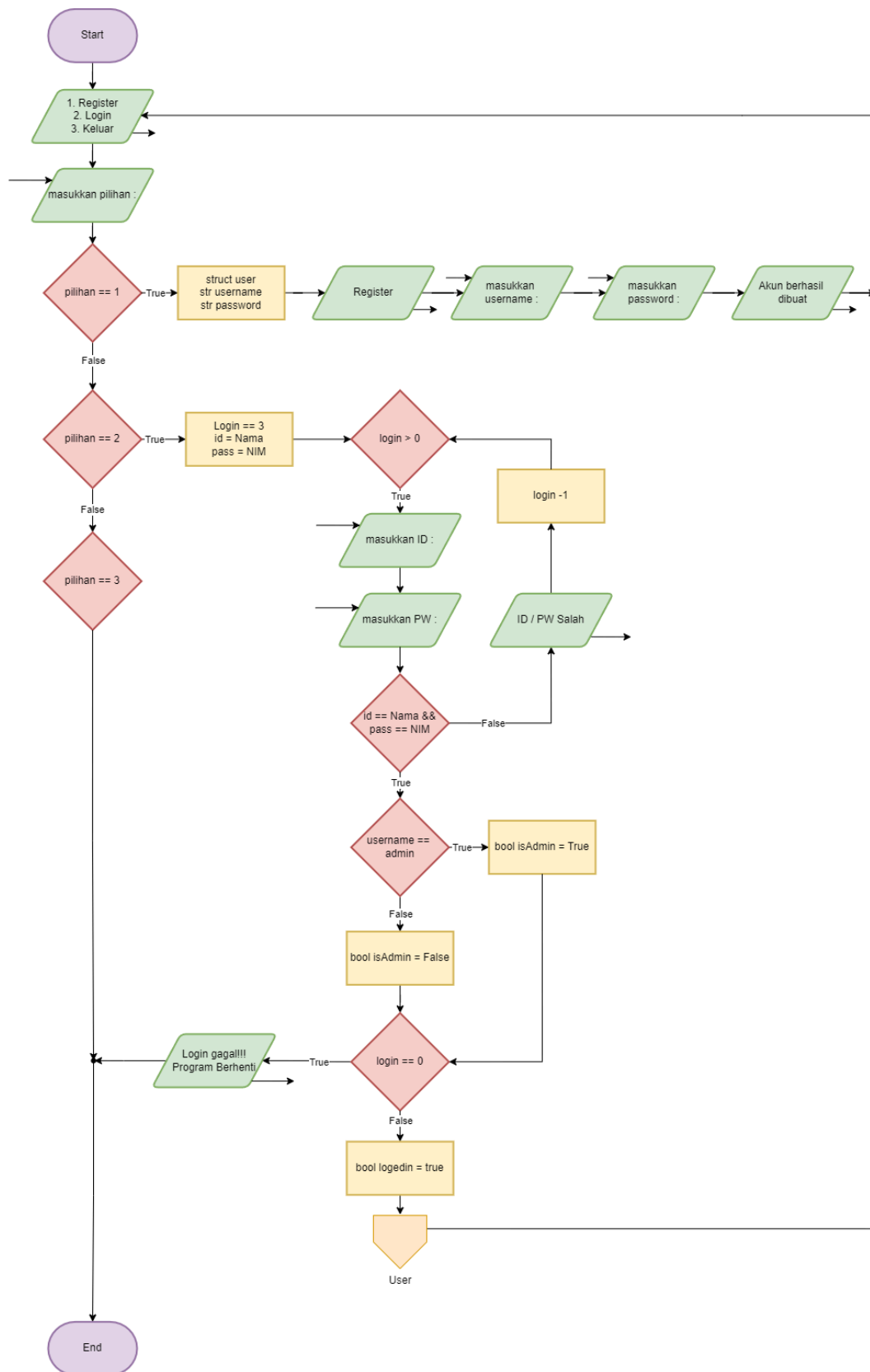
**LAPORAN PRAKTIKUM**  
**POSTTEST 6**  
**ALGORITMA PEMROGRAMAN LANJUT**



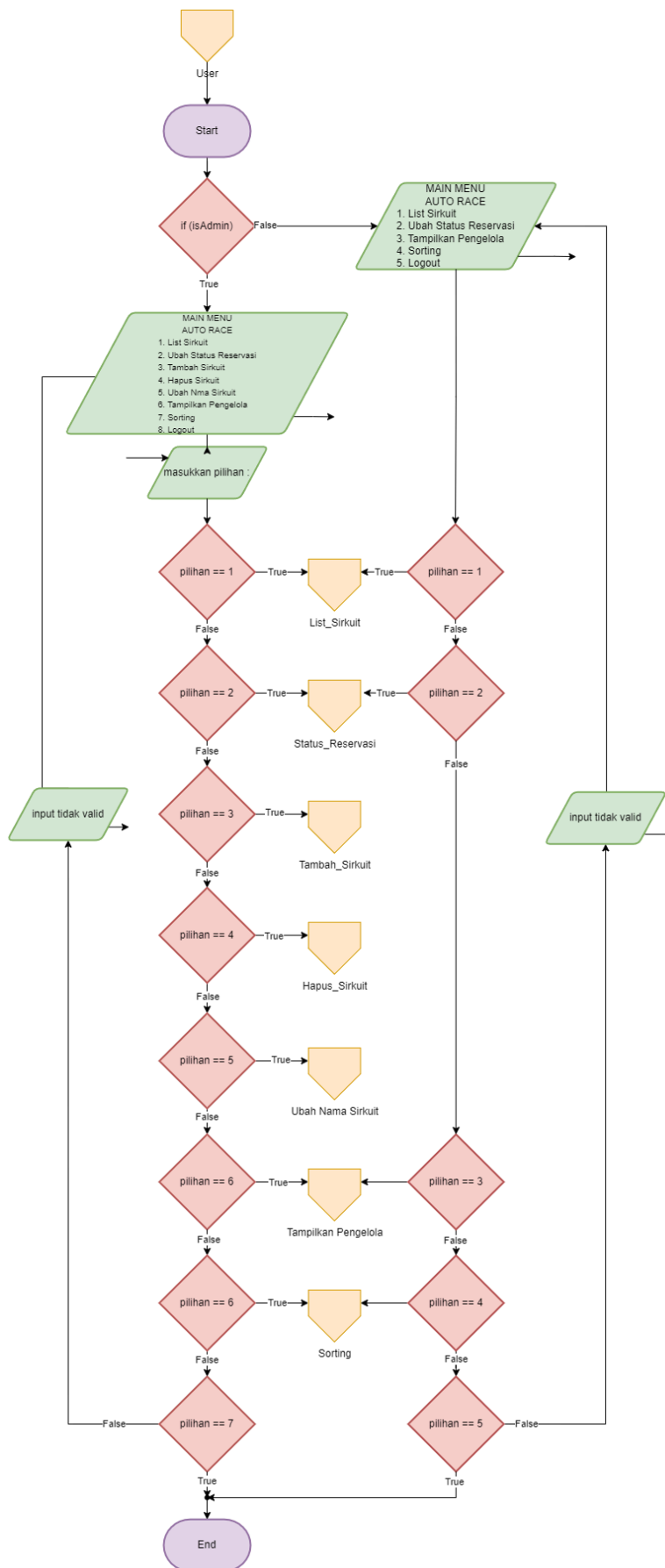
**Disusun oleh:**  
**Muhammad Ilma Yusrian Fahmi**  
**Kelas B2 24**

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

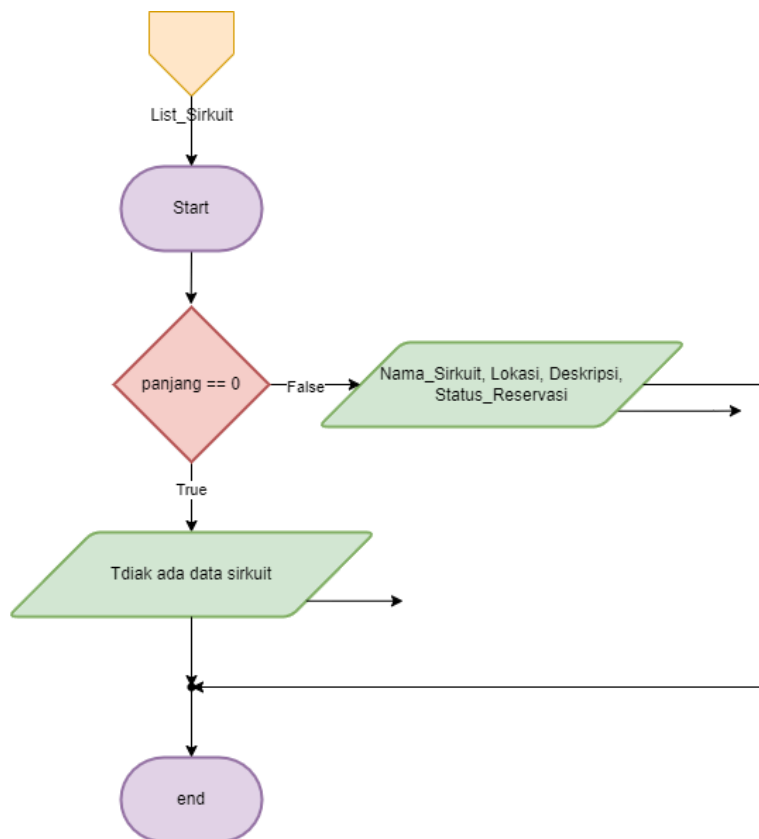
## 1. Flowchart



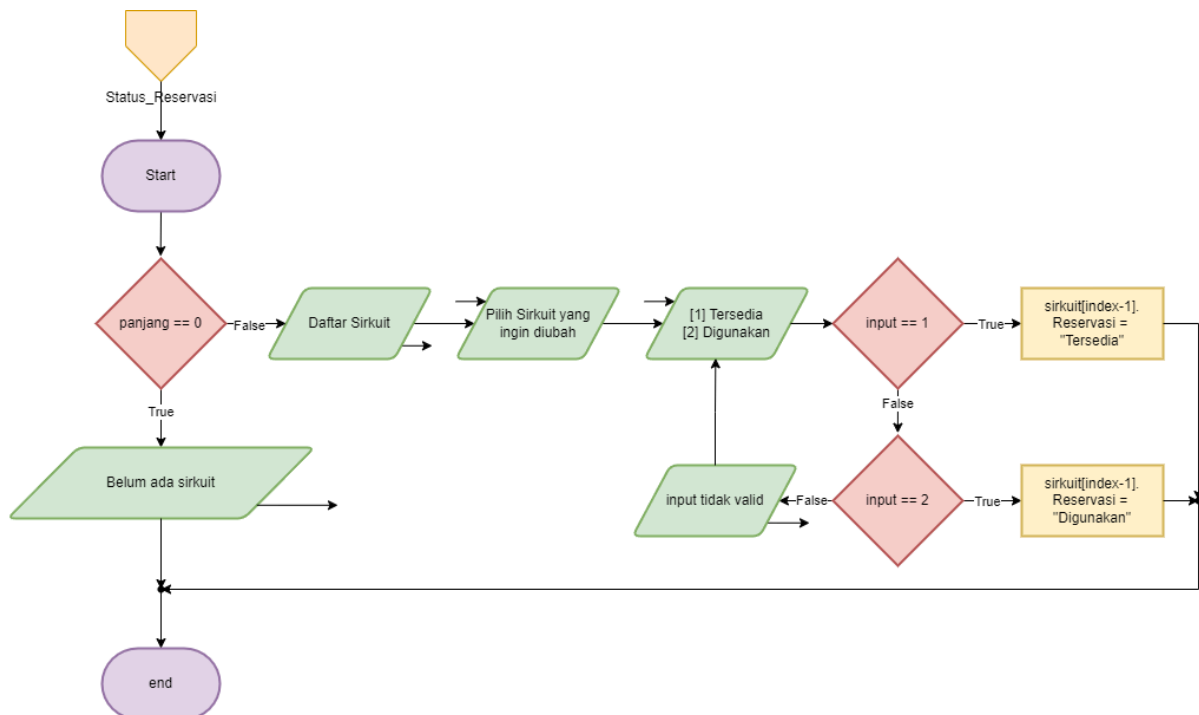
Gambar 1.1 Flowchart Main Menu



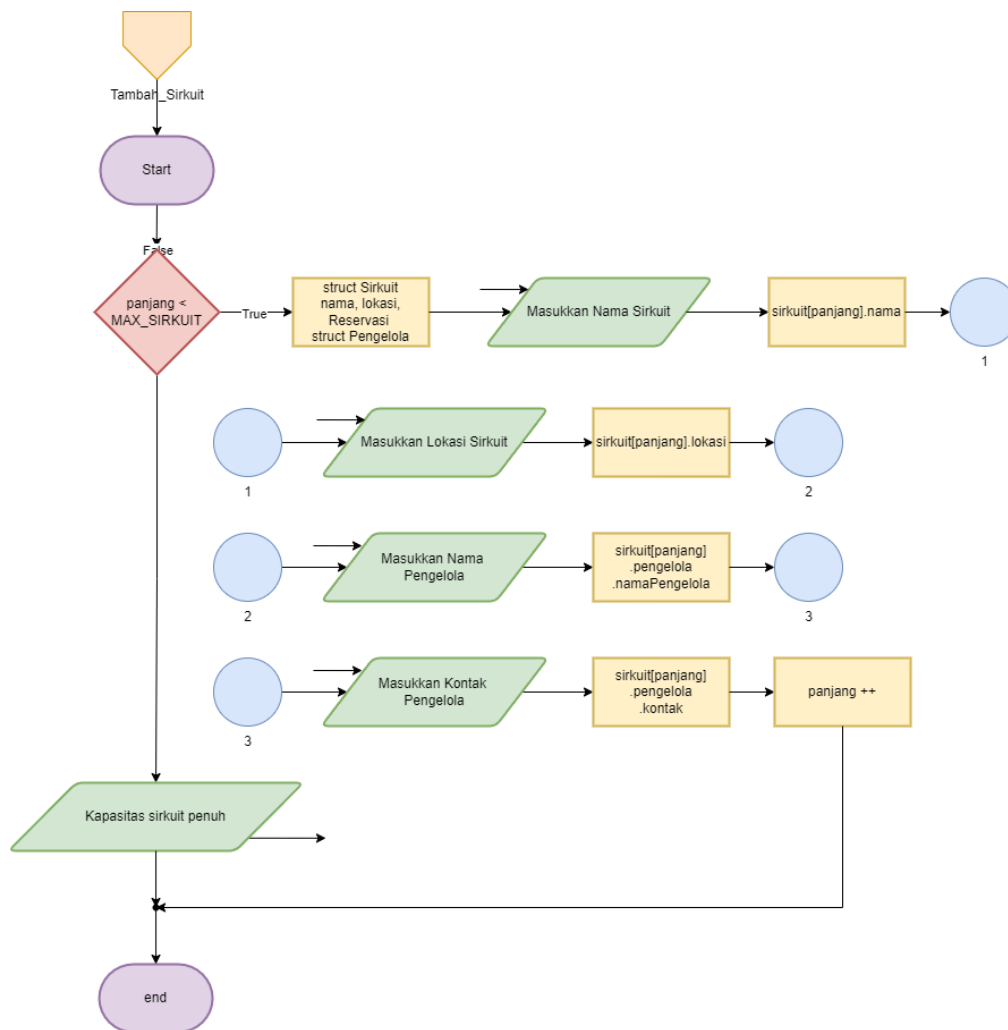
Gambar 1.2 Flowchart Menu User



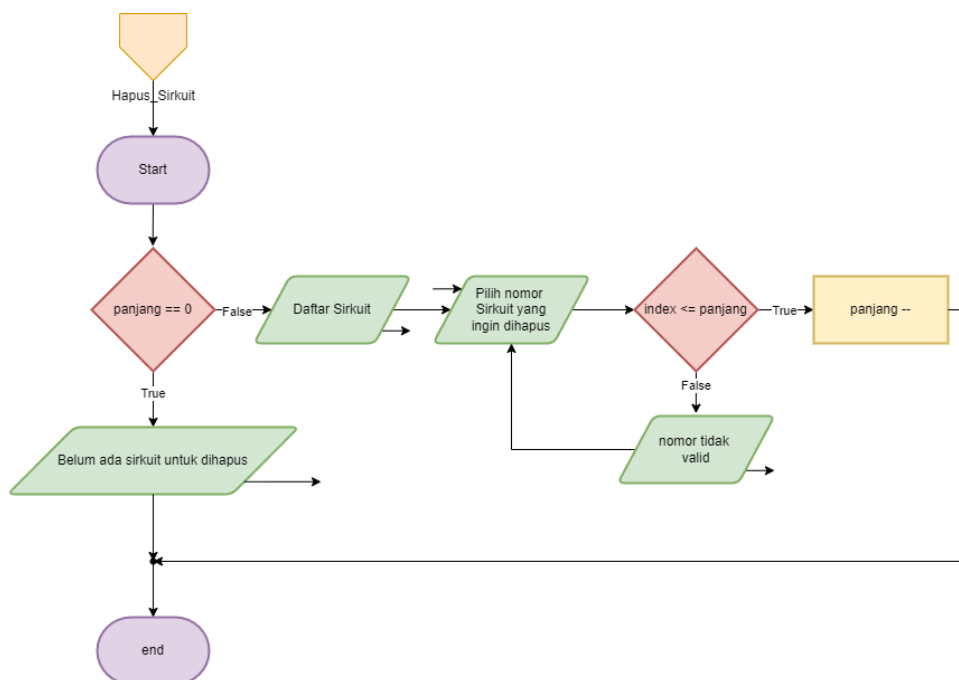
Gambar 1.3 Flowchart Menu List Sirkuit



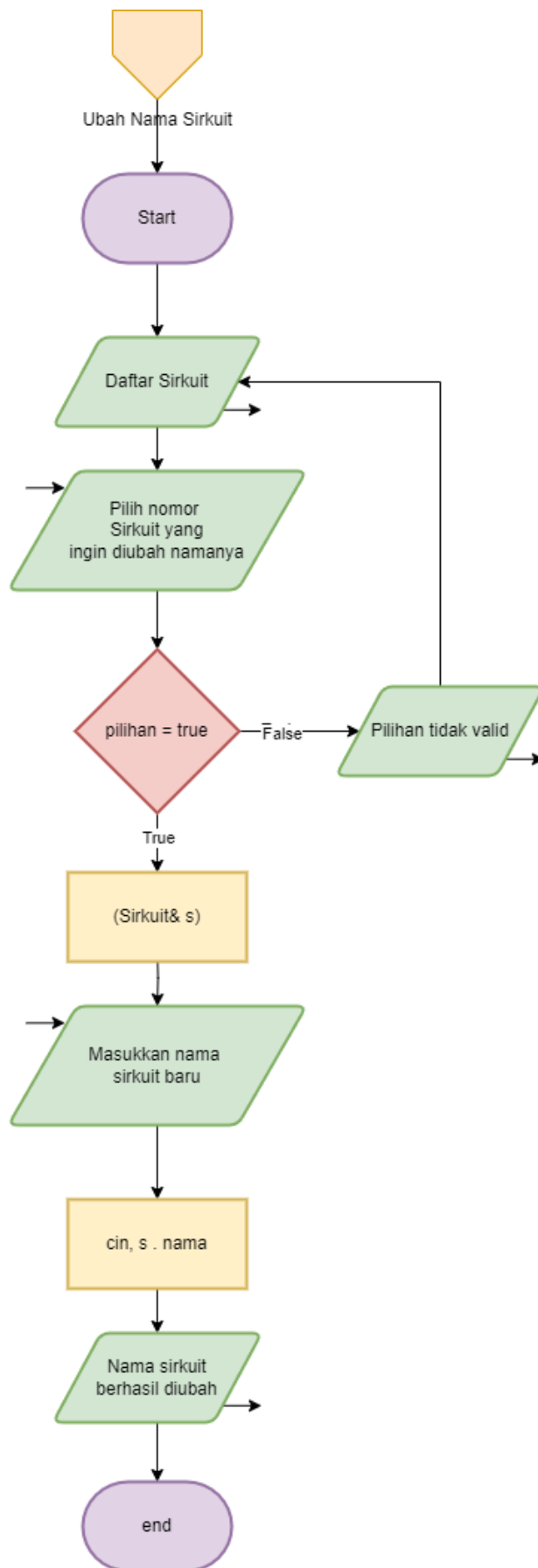
Gambar 1.4 Flowchart Menu Reservasi



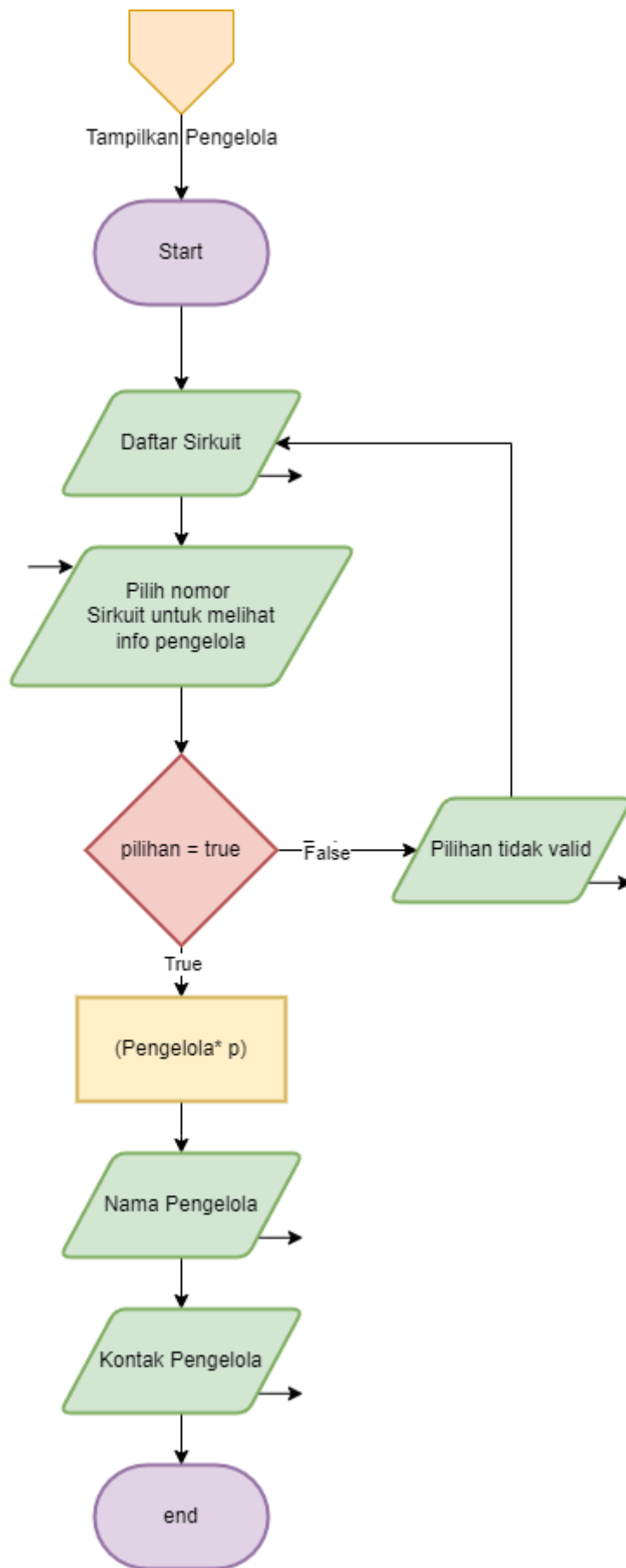
Gambar 1.5 Flowchart Menu Tambah Sirkuit



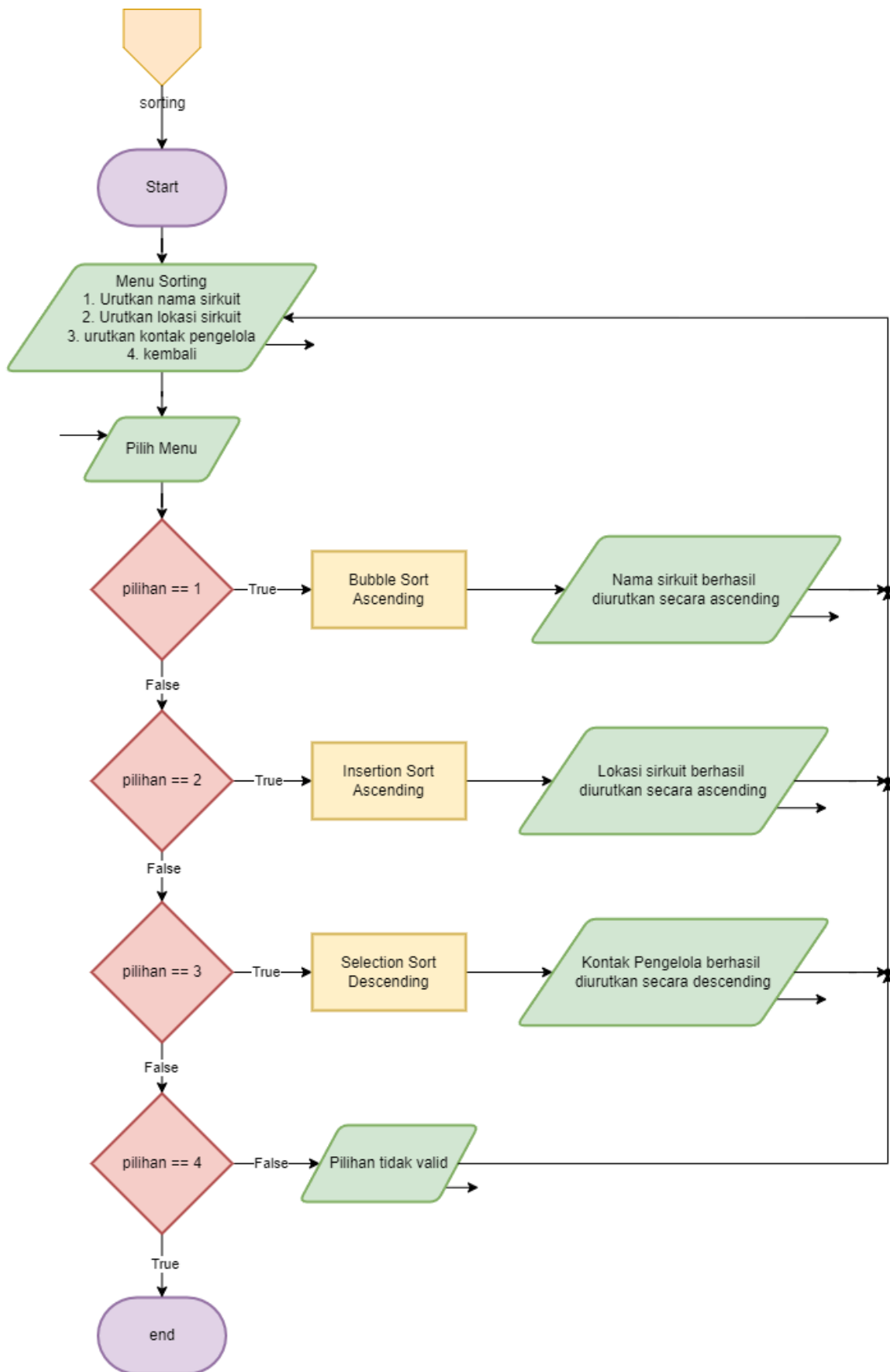
Gambar 1.6 Flowchart Menu Hapus Sirkuit



Gambar 1.7 Flowchart Menu Ubah Nama Sirkuit



Gambar 1.8 Flowchart Menu Tampilkan Pengelola



Gambar 1.9 Flowchart Menu Sorting



## **2. Analisis Program**

Program manajemen sirkuit ini dibuat untuk mempermudah pengguna dalam mengelola data sirkuit balap, seperti menambahkan, menampilkan, mengubah status reservasi, dan menghapus sirkuit. Program ini dilengkapi dengan fitur register dan login untuk memastikan hanya pengguna yang memiliki akses yang dapat mengelola data, serta akan menghentikan program jika gagal login sebanyak tiga kali. Selain itu, sistem memastikan validasi input agar data yang dimasukkan sesuai.

### 3. Source Code

#### A. Fitur Register

Fitur ini digunakan untuk melakukan proses registrasi bagi pengguna yang belum memiliki akun.

```
void registerUser() {  
    if (jumlahPengguna >= MAX_USERS) {  
        cout << "Batas pengguna tercapai!\n";  
    } else {  
        cout << "\n=== Register ===\n";  
        cout << "Masukkan Username: ";  
        cin >> users[jumlahPengguna].username;  
        cout << "Masukkan Password: ";  
        cin >> users[jumlahPengguna].password;  
        cout << "Registrasi Berhasil!\n";  
        jumlahPengguna++;  
    }  
    system("pause");  
}
```

## B. Fitur Login

Fitur ini digunakan untuk melakukan proses login, pengguna hanya memiliki kesempatan maksimal 3 kali, jika salah memasukkan username atau password sebanyak 3 kali maka program akan secara otomatis berhenti.

```
//kode admin/user
bool loginUser(bool &isAdmin) {
    int login = 3;
    string username, password;
    while (login > 0) {
        system("cls");
        cout << "=====\\n";
        cout << "                LOGIN                \\n";
        cout << "=====\\n";
        cout << "Masukkan Username: ";
        getline(cin, username);
        cout << "Masukkan Password: ";
        getline(cin, password);

        if (username == adminUsername && password == adminPassword) {
            isAdmin = true;
            return true;
        }

        for (int i = 0; i < jumlahPengguna; i++) {
            if (users[i].username == username && users[i].password ==
password) {
                return true;
            }
        }

        login--;
        cout << "\\nGagal! Percobaan tersisa: " << login << "\\n";
        system("pause");
    }
    return false;
}
```

### C. Menu List Sirkuit

Ketika pengguna memilih salah menu untuk menampilkan list sirkuit, program akan menampilkan list sirkuit yang ada, jika data sirkuit tidak tersedia maka program akan menampilkan output tidak ada data sirkuit, tetapi jika data tersedia maka program akan menampilkan dalam bentuk tabel.

```
struct Pengelola {
    string namaPengelola;
    string kontak;
};

struct Sirkuit {
    string nama;
    string lokasi;
    string statusReservasi;
    Pengelola pengelola;
};

Sirkuit sirkuit[MAX_SIRKUIT] = {
    {"Mandalika", "Pulau Lombok", "Tersedia", {"PT Mandalika Racing",
"mandalika@racing.com"}},
    {"Internasional Sentul", "Bogor", "Tersedia", {"Sentul Management",
"sentul@racing.com"}},
    {"Jaya Ancol", "Jakarta", "Tersedia", {"Ancol Circuit Ltd",
"ancol@racing.com"}}
};

//fungsi rekursif
void tampilkanSirkuitRekursif(Sirkuit sirkuit[], int index, int jumlahSirkuit)
{
    if (index >= jumlahSirkuit) return;
    cout << "| " << setw(1) << index + 1 << " | "
        << setw(22) << left << sirkuit[index].nama << " | "
        << setw(14) << left << sirkuit[index].lokasi << " | "
        << setw(22) << left << sirkuit[index].pengelola.namaPengelola << " | "
        << setw(25) << left << sirkuit[index].pengelola.kontak << " | "
        << setw(16) << left << sirkuit[index].statusReservasi << " |\n";
    tampilkanSirkuitRekursif(sirkuit, index + 1, jumlahSirkuit);
}
```



### E. Menu Tampilkan Pengelola

Ketika pengguna memilih menu untuk Tampilkan Pengelola sirkuit, program akan menampilkan list sirkuit yang ada terlebih dahulu lalu pengguna akan memilih nomor sirkuit yang ingin dilihat pengelolanya, menggunakan pointer dereference untuk mengakses data dari Pengelola.

```
void tampilkanPengelola(const Pengelola* p) {  
    cout << "Nama Pengelola   : " << p->namaPengelola << endl;  
    cout << "Kontak Pengelola : " << p->kontak << endl;  
}
```

## E. Menu Sorting

Ketika pengguna memilih menu sorting, terdapat beberapa pilihan seperti sorting nama sirkuit secara ascending, sorting lokasi sirkuit secara ascending dan sorting kontak pengelola secara descending.

```
void menuSorting() {
    int pilihSort;
    do {
        system("cls");
        cout << "=====\n";
        cout << "          MENU SORTING          \n";
        cout << "=====\n";
        cout << "1. Urutkan Nama Sirkuit (Ascending)\n";
        cout << "2. Urutkan Lokasi Sirkuit (Ascending)\n";
        cout << "3. Urutkan Kontak Pengelola (Descending)\n";
        cout << "4. Kembali\n";
        cout << "Pilih menu: ";
        cin >> pilihSort;
        cin.ignore();

        switch (pilihSort) {
            case 1:
                bubbleSortNama(sirkuit, panjang);
                break;
            case 2:
                insertionSortLokasi(sirkuit, panjang);
                break;
            case 3:
                selectionSortKontak(sirkuit, panjang);
                break;
            case 4:
                cout << "Kembali ke menu sebelumnya...\n";
                system("pause");
                break;
            default:
                cout << "Pilihan tidak valid!\n";
                system("pause");
        }
    } while (pilihSort != 4);
}
```

#### 4. Output Program

```
=====
MAIN MENU
AUTO RACE
=====
1. List Sirkuit
2. Ubah Status Reservasi
3. Tambah Sirkuit
4. Hapus Sirkuit
5. Logout
Pilih menu: 
```

Gambar 4.1 Menu Admin

```
=====
MAIN MENU
AUTO RACE
=====
1. List Sirkuit
2. Ubah Status Reservasi
5. Logout
Pilih menu: 
```

Gambar 4.2 Menu Pelanggan

No	Nama Sirkuit	Lokasi	Pengelola	Kontak	Status Reservasi
1	Mandalika	Pulau Lombok	PT Mandalika Racing	mandalika@racing.com	Tersedia
2	Internasional Sentul	Bogor	Sentul Management	sentul@racing.com	Tersedia
3	Jaya Ancol	Jakarta	Ancol Circuit Ltd	ancol@racing.com	Tersedia

Press any key to continue . . .

Gambar 4.3 List Sirkuit



```

+-----+-----+-----+
| 1 | Mandalika | Tersedia |
| 2 | Internasional Sentul | Tersedia |
| 3 | Jaya Ancol | Tersedia |
+-----+-----+-----+

Pilih nomor sirkuit: 1

[1] Tersedia
[2] Digunakan
Pilih Status Reservasi Baru (1/2): 2
Status reservasi berhasil diubah menjadi: Digunakan
Press any key to continue . . .

```

Gambar 4.4 Reservasi Sirkuit

```

+-----+-----+-----+
| No | Nama Sirkuit | Lokasi |
+-----+-----+-----+
| 1 | Mandalika | Pulau Lombok |
| 2 | Internasional Sentul | Bogor |
| 3 | Jaya Ancol | Jakarta |
+-----+-----+-----+

Pilih nomor sirkuit yang ingin dihapus: 3
Sirkuit berhasil dihapus!
Press any key to continue . . .

```

Gambar 4.5 Hapus Sirkuit

```

Pilih sirkuit yang ingin diubah namanya:
1. Mandalika
2. Internasional Sentul
3. Jaya Ancol
Pilih nomor: 1
Nama sirkuit saat ini: Mandalika
Masukkan nama baru: Likamanda
Nama sirkuit berhasil diubah menjadi: Likamanda
Press any key to continue . . .

```

Gambar 4.6 Ubah Nama Sirkuit

```
Pilih sirkuit untuk lihat info pengelola:
1. Likamanda
2. Internasional Sentul
3. Jaya Ancol
Pilih nomor: 1
Nama Pengelola   : PT Mandalika Racing
Kontak Pengelola : mandalika@racing.com
Press any key to continue . . .
```

Gambar 4.6 Tampilkan Pengelola

```
=====
                        MENU SORTING
=====
1. Urutkan Nama Sirkuit (Ascending)
2. Urutkan Lokasi Sirkuit (Ascending)
3. Urutkan Kontak Pengelola (Descending)
4. Kembali
Pilih menu: 
```

Gambar 4.7 Menu Sorting

```
=====
                        MENU SORTING
=====
1. Urutkan Nama Sirkuit (Ascending)
2. Urutkan Lokasi Sirkuit (Ascending)
3. Urutkan Kontak Pengelola (Descending)
4. Kembali
Pilih menu: 1
Data berhasil diurutkan berdasarkan Nama Sirkuit.
Press any key to continue . . .
```

Gambar 4.8 Memilih Menu 1. Urutkan Nama Sirkuit

No	Nama Sirkuit	Lokasi	Pengelola	Kontak	Status Reservasi
1	Bukit Peusar	Tasikmalaya	Peusar Motorsports	0852-7643-1059	Tersedia
2	Gokart Subang	Subang	Subang Racing Club	0898-2150-7386	Tersedia
3	Internasional Sentul	Bogor	Sentul Management	0812-3958-7264	Tersedia
4	Jaya Ancol	Jakarta	Ancol Circuit Ltd	0899-7321-4186	Tersedia
5	Mandalika	Pulau Lombok	PT Mandalika Racing	0874-2197-4638	Tersedia
6	Pancing Circuit	Medan	Medan Racing Team	0821-6394-5728	Tersedia
7	Sirkuit Balipat	Banjarmasin	Balipat Racing Club	0815-9730-6241	Tersedia
8	Sirkuit Manado	Manado	Manado Speed Track	0878-3542-6901	Tersedia
9	Sirkuit Skyland	Palembang	Skyland Motorsport	0860-4827-3195	Tersedia
10	e-Prix	Jakarta	e-Prix Inter	0831-5064-2987	Tersedia

Press any key to continue . . .

Gambar 4.9 Hasil Sorting Nama Sirkuit

## 5. Langkah-Langkah Git pada VSCode

### 5.1 Git Add

Git add digunakan untuk merubah status file yang diinginkan, file akan memiliki status Untracked apabila belum pernah di-add, dan akan berubah menjadi staged yang artinya file sudah di-add ke repository tetapi belum di-commit.

```
Lenovo@LAPTOP-1M0LURIM MINGW64 ~/Desktop/praktikum-apl (main)
$ git add .
```

Gambar 5.1 Git Add

### 5.2 Git Commit

Git commit berfungsi untuk menyimpan perubahan file di repository setelah ditambahkan, -m “...” untuk memberikan pesan yang diinginkan.

```
Lenovo@LAPTOP-1M0LURIM MINGW64 ~/Desktop/praktikum-apl (main)
$ git commit -m "up code"
[main 6ce68f0] up code
2 files changed, 461 insertions(+)
create mode 100644 post-test/post-test-6/2409106079-MuhammadIlmaYusrianFahmi-PT-6.cpp
create mode 100644 post-test/post-test-6/2409106079-MuhammadIlmaYusrianFahmi-PT-6.exe
```

Gambar 5.2 Git Commit

### 5.3 Git Push

git push -u origin main digunakan untuk meng-upload perubahan yang ada pada repository lokal ke repository github

```
Lenovo@LAPTOP-1M0LURIM MINGW64 ~/Desktop/praktikum-apl (main)
$ git push origin main
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 8 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 685.22 KiB | 4.31 MiB/s, done.
Total 6 (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/MeFahme/post-test-apl.git
a30250f..6ce68f0 main -> main
```

Gambar 5.3 Git Push