

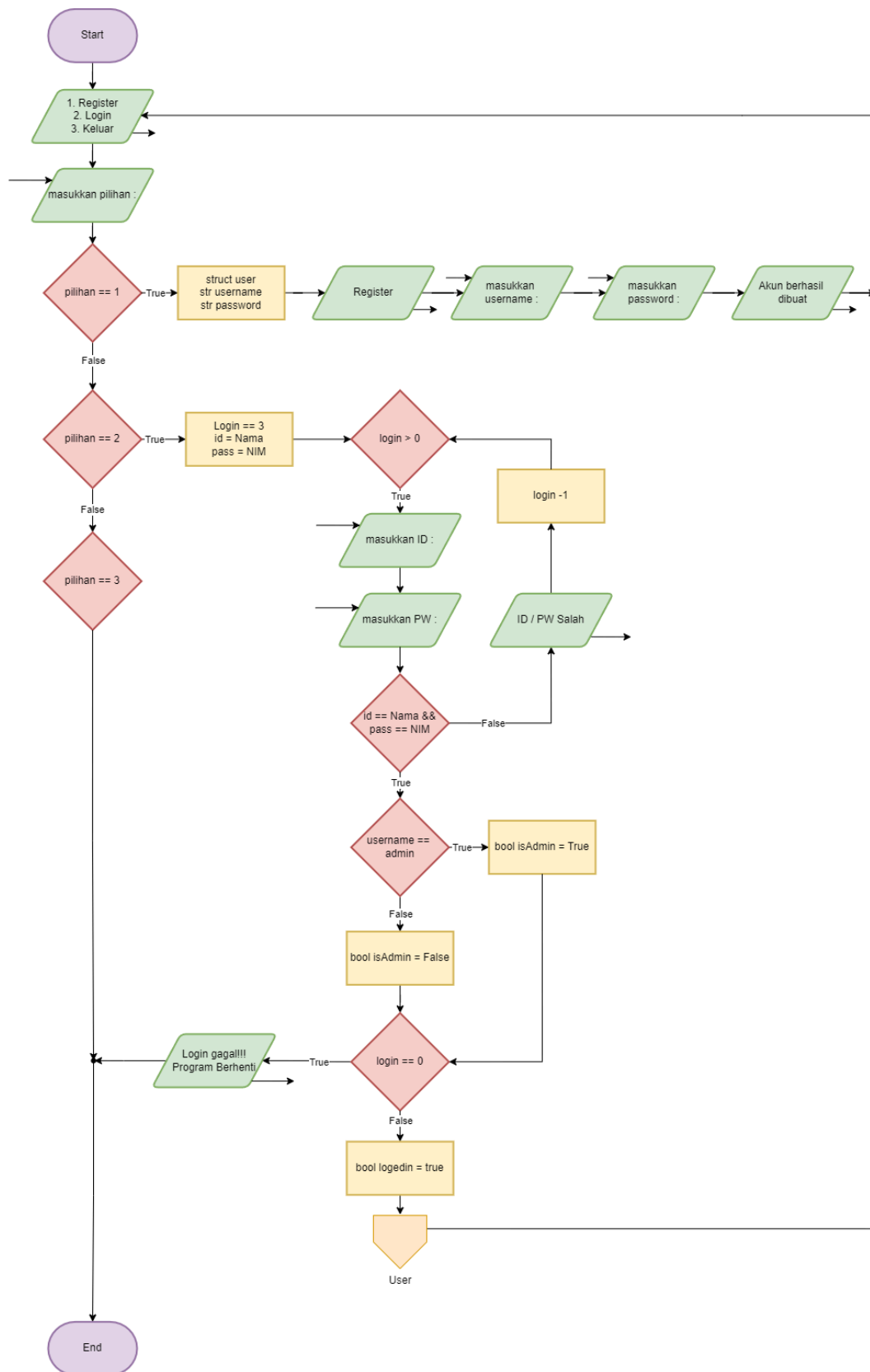
LAPORAN PRAKTIKUM
POSTTEST 3
ALGORITMA PEMROGRAMAN LANJUT



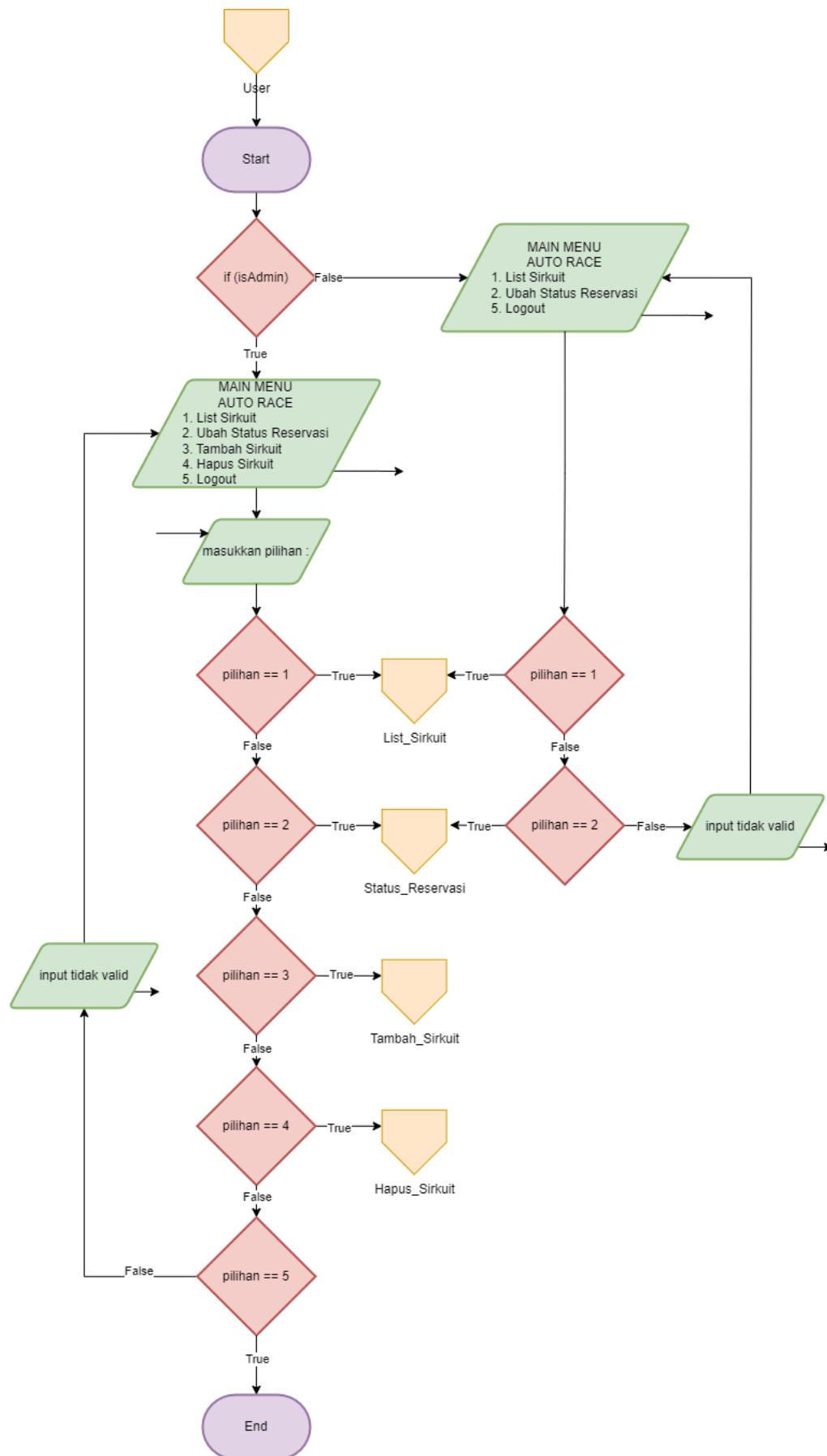
Disusun oleh:
Muhammad Ilma Yusrian Fahmi
Kelas B2 24

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

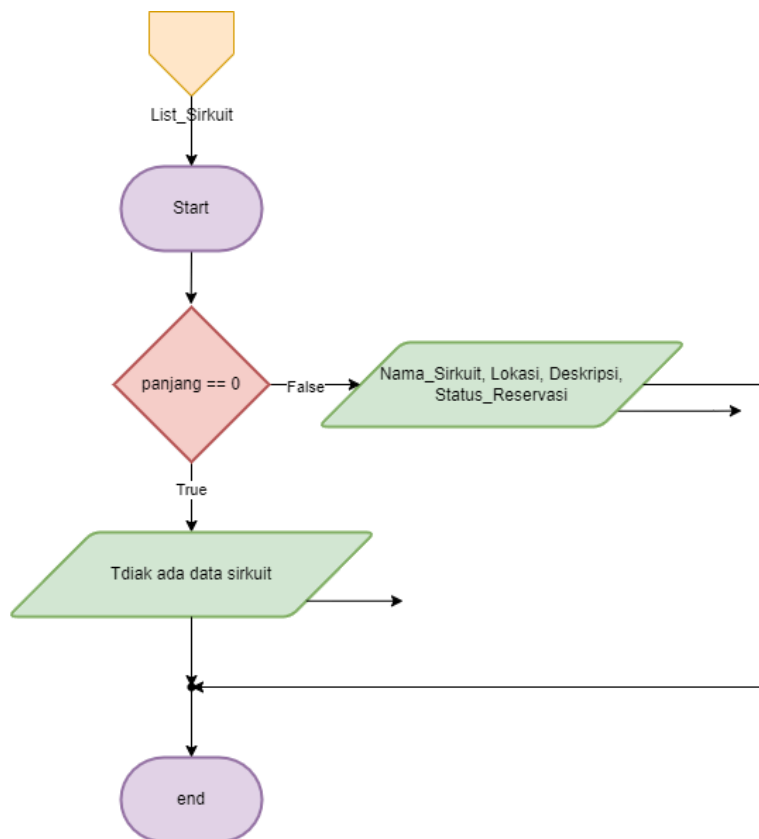
1. Flowchart



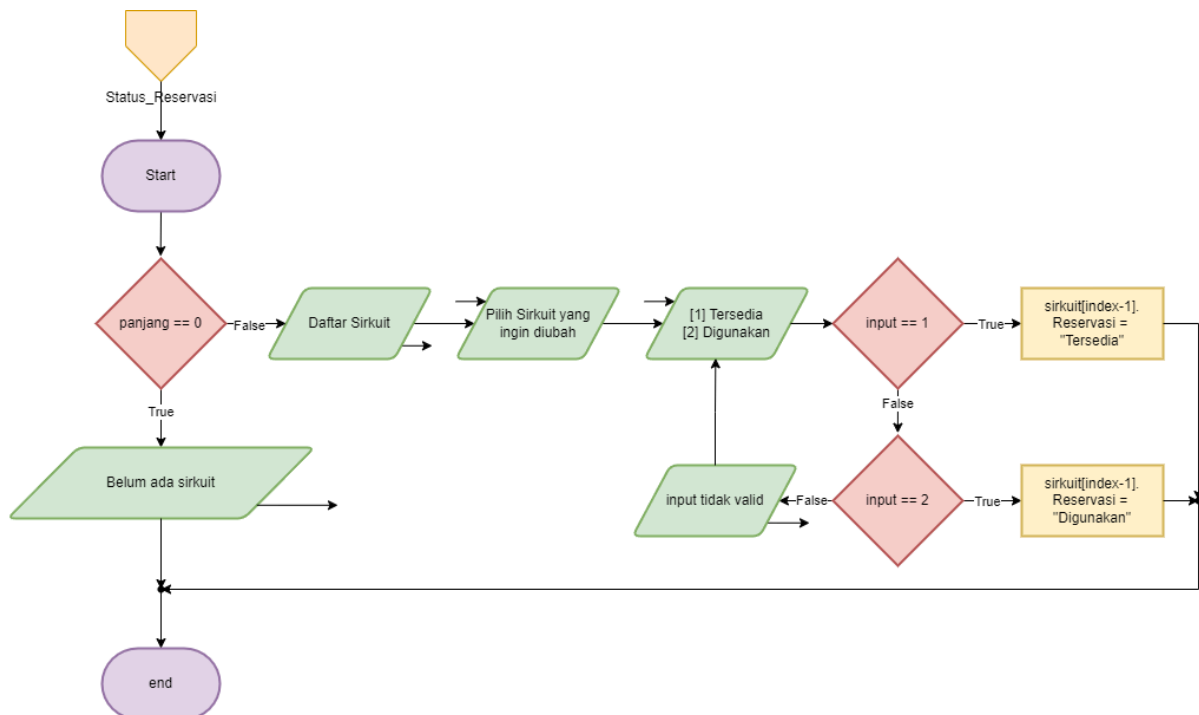
Gambar 1.1 Flowchart Main Menu



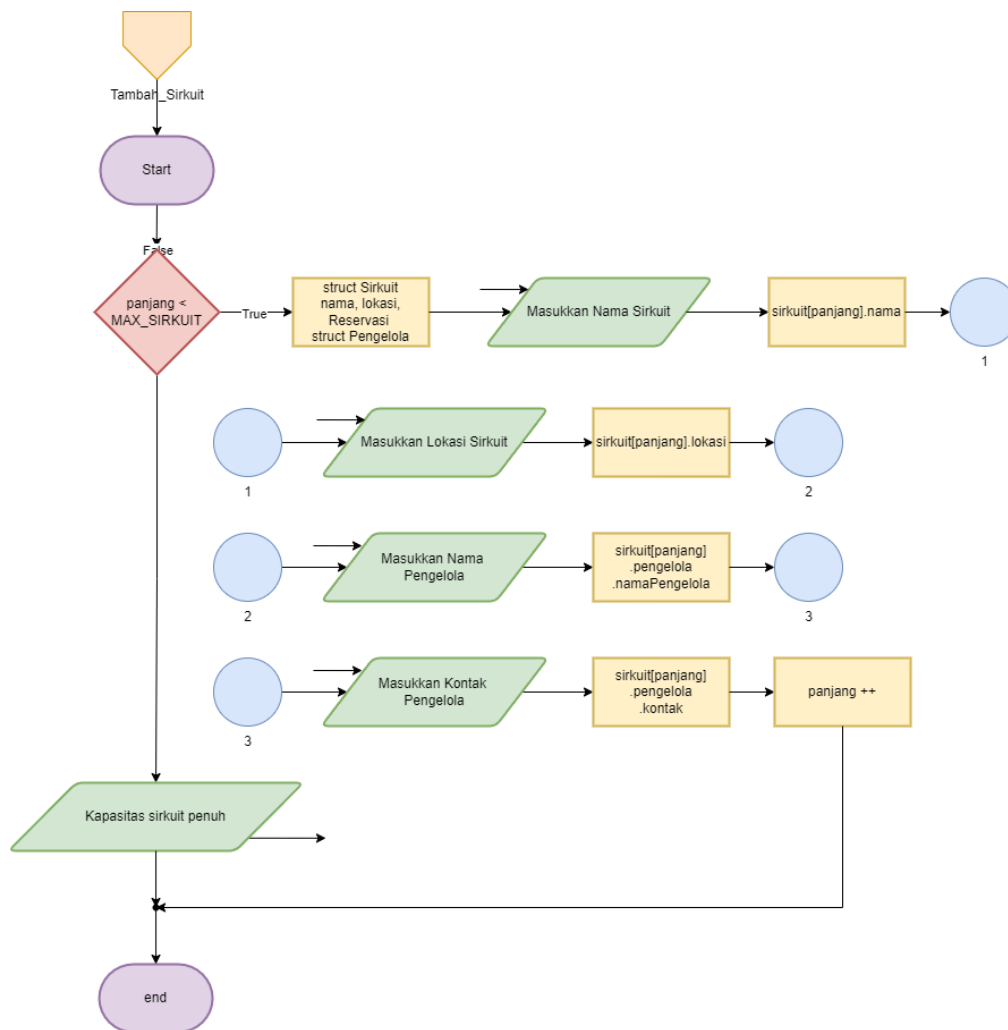
Gambar 1.2 Flowchart Menu User



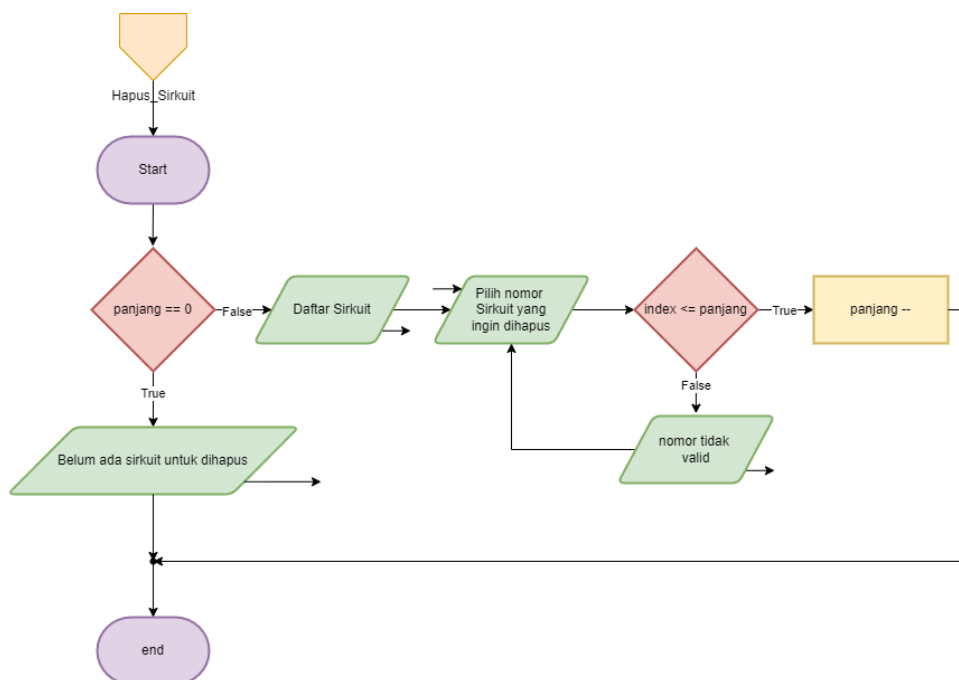
Gambar 1.3 Flowchart Menu List Sirkuit



Gambar 1.4 Flowchart Menu Reservasi



Gambar 1.5 Flowchart Menu Tambah Sirkuit



Gambar 1.6 Flowchart Menu Hapus Sirkuit

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.

```
struct User {
    string username;
    string password;
};

const int MAX_USERS = 100;
User users[MAX_USERS];
int jumlahPengguna = 0;
int pilihan, index;

string adminUsername = "Yusrian Fahmi";
string adminPassword = "2409106079";

if (pilihan == 1) {
    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++;
    }
}
```

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.

```
int login = 3;
string username, password;
bool isAdmin = false;

while (login > 0) {
    system("cls");
    cout << "===== " << endl;
    cout << "          LOGIN          " << endl;
    cout << "===== " << endl;
    cout << "Masukkan Username: ";
    getline(cin, username);
    cout << "Masukkan Password: ";
    getline(cin, password);

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

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

    if (!akun) {
        login--;
        cout << "\nGagal! Percobaan tersisa: " << login << "\n";
        system("pause");
    } else {
        break;
    }
}
```


C. Menu List Sirkuit

Ketika pengguna memilih salah satu 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"}}
};

system("cls");
cout <<
"\n+-----+-----+-----+-----+\n";
+-----+-----+-----+-----+
cout << "| No | Nama Sirkuit | Lokasi | Pengelola  
| Kontak | Status Reservasi |\n";
cout <<
"+-----+-----+-----+-----+\n";
+-----+-----+-----+-----+
for (int i = 0; i < panjang; i++) {
    cout << "| " << setw(1) << i+1 << " | "  

        << setw(22) << left << sirkuit[i].nama << " | "  

        << setw(14) << left << sirkuit[i].lokasi << " | "  

        << setw(22) << left << sirkuit[i].pengelola.namaPengelola << " | "  

        << setw(25) << left << sirkuit[i].pengelola.kontak << " | "  

        << setw(16) << left << sirkuit[i].statusReservasi << " |\n";
}
cout <<
"+-----+-----+-----+-----+\n";
+-----+-----+-----+-----+
system("pause");
```

4. Langkah-Langkah Git pada VSCode

```
=====
                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

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 "upload code"
[main 6f2fef2] upload code
9 files changed, 460 insertions(+), 104 deletions(-)
create mode 100644 .vscode/settings.json
create mode 100644 kelas/.vscode/settings.json
rename kelas/{ => pertemuan1}/pertemuan1.cpp (100%)
rename kelas/{ => pertemuan1}/pertemuan1.exe (100%)
create mode 100644 kelas/pertemuan2/pertemuan2.cpp
create mode 100644 kelas/pertemuan2/pertemuan2.exe
create mode 100644 post-test/post-test-3/multiuser-fixed-v1.cpp
create mode 100644 post-test/post-test-3/multiuser-fixed-v1.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: 21, done.
Counting objects: 100% (21/21), done.
Delta compression using up to 8 threads
Compressing objects: 100% (14/14), done.
Writing objects: 100% (16/16), 951.54 KiB | 3.11 MiB/s, done.
Total 16 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), completed with 2 local objects.
To https://github.com/MeFahme/post-test-apl.git
b6fff74..6f2fef2 main -> main
```

Gambar 5.3 Git Push