

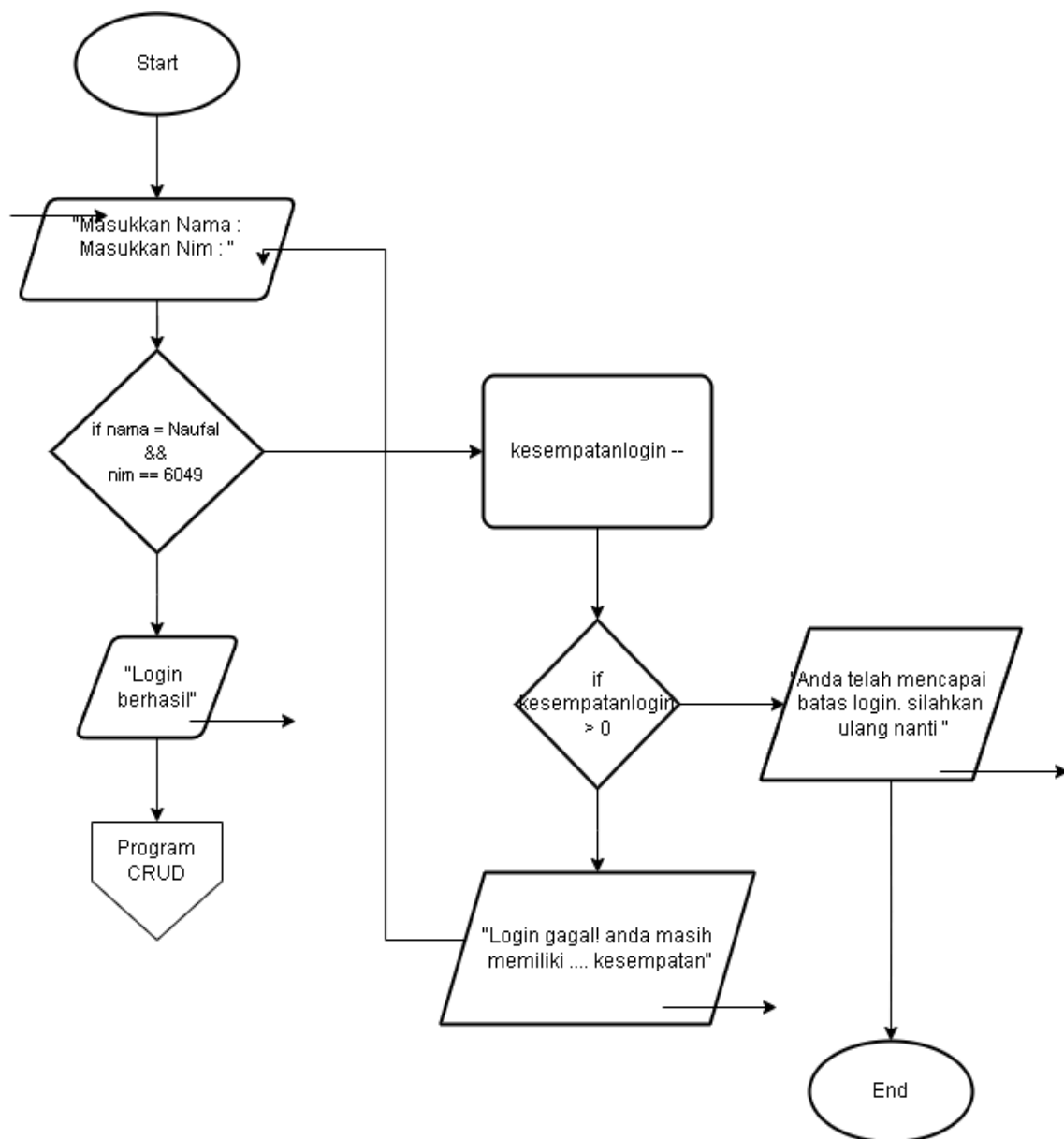
LAPORAN PRAKTIKUM
POSTTEST 2
ALGORITMA PEMROGRAMAN LANJUT



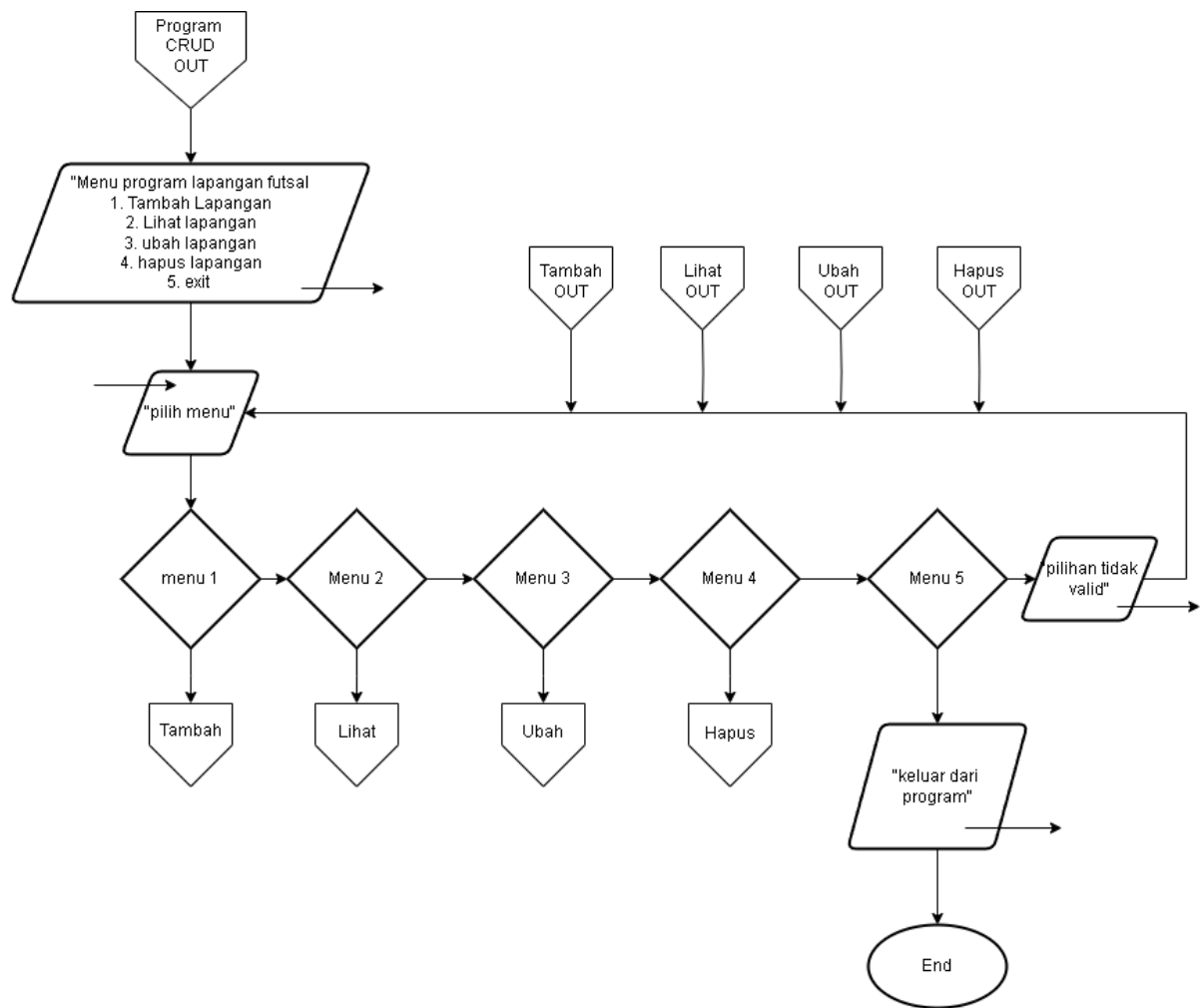
Disusun oleh:
Muhammad Naufal Adi Brata Putra Suharizman Poerwo (2409106049)
Kelas (B1 '24)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

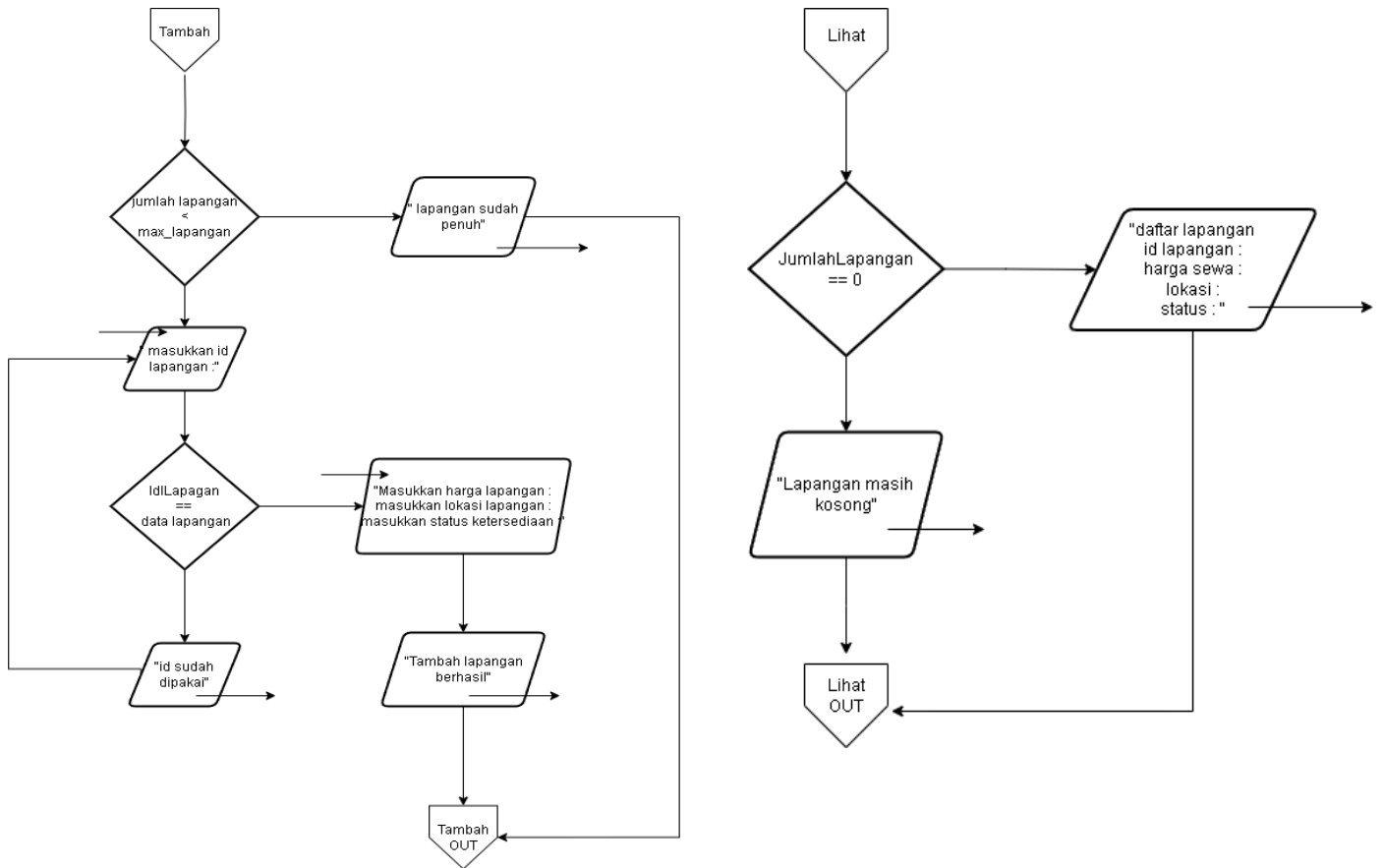
1. Flowchart



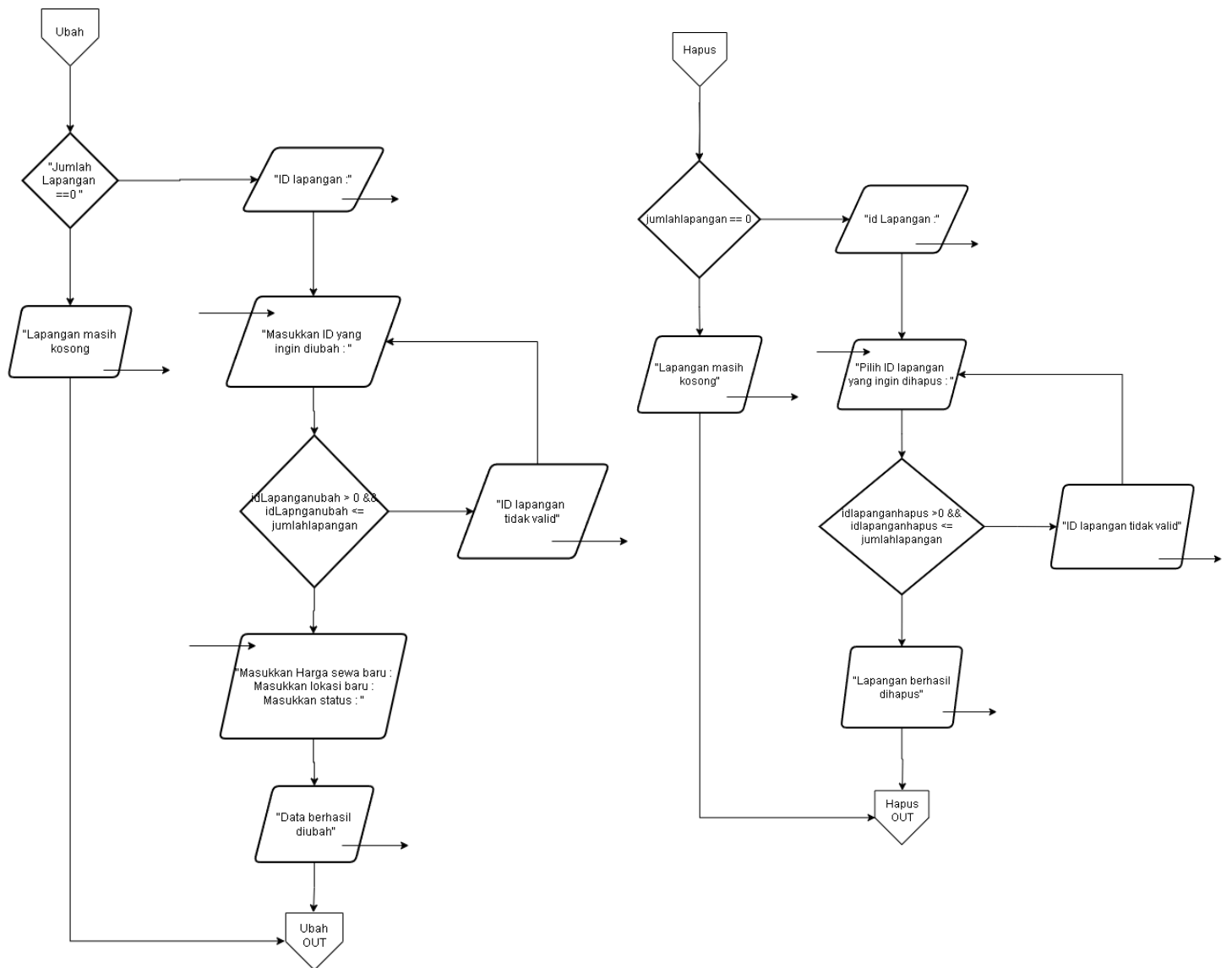
Gambar 1.1 Flow Login



Gambar 1.2 Flow CRUD



Gambar 1.3Flow CRUD (Tambah dan Lihat)



Gambar 1.4 (Ubah dan Hapus)

2. Analisis Program

2.1 Deskripsi Singkat Program

jadi program ini merupakan program Manajemen Lapangan Futsal yang dimana bisa diakses menggunakan Nama dan NIM kemudian program ini bisa melakukan Tambah Lapangan, Lihat Lapangan, dan Ubah data lapangan.

2.2 Penjelasan Alur & Algoritma

pada fitur login user diminta untuk memasukkan Nama dan NIM yang sudah ditentukan dan user diberi 3 kesempatan. Ketika user memasukan Nama dan NIM dengan benar maka akan muncul output “Anda Berhasil login !!”. Jika user memasukkan Nama atau NIM yang salah maka kesempatan akan berkurang 1 dan muncul output “Login gagal !! , anda masih memiliki ... kesempatan lagi.” jika si user sudah tidak punya kesempatan maka akan muncul output “Anda telah mencapai batas kesempatan login. Silakan ulang nanti.

kemudian kita masuk ke program Manajemen Lapangan Futsal, program ini dapat diakses ketika user memasukkan Nama dan NIM dengan benar. Di dalam program ini terdapat 5 menu / fitur yang pertama ada tambah lapangan. jika user memilih fitur yang pertama maka user diminta untuk memasukkan ID lapangan kemudian id lapangan ini akan di check apakah ID ini sudah dipakai atau belum jika sudah dipakai maka akan muncul output “ ID Lapangan sudah digunakan ! silahkan masukkan id yang berbeda.” Jika belum ada maka akan lanjut untuk memasukkan Harga sewa, lokasi, status ketersediaan.” Kemudian akan muncul output “Tambah Lapangan Berhasil !.” kemudian jika jumlah lapangan lebih kecil daripada jumlah max lapangan maka akan muncul output “ Lapangan sudah penuh !. “ kemudian fitur yang kedua ada fitur lihat lapangan . Jika user memilih fitur kedua ini maka user akan mendapatkan output dari data yang sudah dimasukkan di fitur pertama tadi. kemudian jika belum ada data lapangan maka akan muncul output “ Lapangan masih kosong”. Jika user memilih fitur yang ketiga maka user diminta untuk memasukkan ID lapangan yang ingin diubah dan akan ditampung dalam variabel idLapanganUbah kemudian jika variabel ini lebih besar dari 0 dan kurang dari sama dengan jumlahlapangan maka user akan memasukkan data yang baru dan akan muncul output “Lapangan berhasil diubah”. kemudian jika user memasukkan nomor/id yang tidak sesuai maka akan muncul output “Nomor lapangan tidak valid”kemudian yang keempat ada fitur hapus data lapangan ketika user memilih fitur ini

maka user akan memasukkan ID yang ingin dihapus dan akan ditampung di variabel `idLapanganHapus` kemudian jika variabel ini lebih besar daripada 0 dan kurang dari sama dengan variabel `jumlahlapangan` maka data akan terhapus dan akan muncul output “lapangan berhasil dihapus !”. kemudian jika user memasukkan ID/nomor yang tidak sesuai maka akan muncul output “nomor lapangan tidak valid”.

3. Source Code

A. Fitur Login

```
while (kesempatanLogin > 0) {
    cout << "Masukkan Nama : ";
    cin >> userLogin[0];
    cout << "Masukkan Nim : ";
    cin >> userLogin[1];

    if (userLogin[0] == "Naufal" && userLogin[1] == "6049") {
        cout << "Login berhasil!" << endl;
        break;
    } else {
        kesempatanLogin--;
        if (kesempatanLogin > 0) {
            cout << "Login gagal! Anda memiliki " << kesempatanLogin << "
kesempatan lagi." << endl;
        } else {
            cout << "Anda telah mencapai batas kesempatan login. Silakan
ulang nanti." << endl;
            return 0;
        }
    }
}
```

fitur ini berfungsi untuk mengakses sistem CRUD

B. Fitur Tambah Lapangan

```
case 1:
    if (jumlahLapangan < MAX_LAPANGAN) {
        string idLapangan;
        cout << "Masukkan ID lapangan : ";
        cin >> idLapangan;

        bool idUsed = false;
        for (int i = 0; i < jumlahLapangan; i++) {
```

```

        if (dataLapangan[i].id == idLapangan) {
            idUsed = true;
            break;
        }
    }

    if (idUsed) {
        cout << "ID lapangan sudah digunakan! Silakan
masukkan ID yang berbeda." << endl;
    } else {
        dataLapangan[jumlahLapangan].id = idLapangan;
        cout << "Masukkan harga sewa lapangan : ";
        cin >> dataLapangan[jumlahLapangan].harga;
        cout << "Masukkan lokasi lapangan : ";
        cin >> dataLapangan[jumlahLapangan].lokasi;
        cout << "Masukkan status ketersediaan lapangan: ";
        cin >> dataLapangan[jumlahLapangan].status;
        cout << "Tambah Lapangan Berhasil !!" << endl;
        jumlahLapangan++;
    }
} else {
    cout << "Lapangan sudah penuh!" << endl;
}
break;

```

Fitur ini berfungsi untuk menambah data lapangan

C. Fitur Lihat Lapangan

```

case 2:
    if (jumlahLapangan == 0) {
        cout << "Lapangan masih kosong!" << endl;
    } else {
        cout << "\n=== DAFTAR LAPANGAN FUTSAL ===" << endl;

        cout << "+" << setfill('-') << setw(13) << "+" << setw(13)
<< "+" << setw(15) << "+" << setw(13) << "+" << endl;
        cout << "| " << setfill(' ') << left << setw(11) << "ID
Lapangan" << "| "
            << setw(11) << "Harga Sewa" << "| "
            << setw(13) << "Lokasi" << "| "
            << setw(11) << "Status" << "|" << endl;
        cout << "+" << setfill('-') << setw(13) << "+" << setw(13)
<< "+" << setw(15) << "+" << setw(13) << "+" << endl;

        for (int i = 0; i < jumlahLapangan; i++) {
            cout << "| " << setfill(' ') << left << setw(11) <<

```



```

dataLapangan[i].id << "| "
                                << setw(11) << dataLapangan[i].harga << "| "
                                << setw(13) << dataLapangan[i].lokasi << "| "
                                << setw(11) << dataLapangan[i].status << "|" <<
endl;
    }

    cout << "+" << setfill('-') << setw(13) << "+" << setw(13)
<< "+" << setw(15) << "+" << setw(13) << "+" << endl;
}
break;

```

Fitur ini berfungsi untuk Melihat data lapangan

D. Fitur Ubah Data Lapangan

```

case 3:
    if (jumlahLapangan == 0) {
        cout << "Lapangan masih kosong!" << endl;
    } else {
        cout << "\n== DAFTAR ID LAPANGAN ==" << endl;
        cout << "+" << setfill('-') << setw(7) << "+" <<
setw(25) << "+" << endl;
        cout << "| " << setfill(' ') << left << setw(5) << "No"
<< "| "
                                << setw(23) << "ID Lapangan" << "|" << endl;
        cout << "+" << setfill('-') << setw(7) << "+" <<
setw(25) << "+" << endl;

        for (int i = 0; i < jumlahLapangan; i++) {
            cout << "| " << setfill(' ') << left << setw(5) << i
+ 1 << "| "
                                << setw(23) << dataLapangan[i].id << "|" <<
endl;
        }

        cout << "+" << setfill('-') << setw(7) << "+" <<
setw(25) << "+" << endl;

        int idLapanganUbah;
        cout << "Masukkan nomor lapangan yang akan diubah: ";
        cin >> idLapanganUbah;

        if (idLapanganUbah > 0 && idLapanganUbah <=
jumlahLapangan) {
            cout << "Masukkan harga sewa lapangan baru: ";

```

```

        cin >> dataLapangan[idLapanganUbah - 1].harga;
        cout << "Masukkan lokasi lapangan baru: ";
        cin >> dataLapangan[idLapanganUbah - 1].lokasi;
        cout << "Masukkan status ketersediaan lapangan baru: ";

        cin >> dataLapangan[idLapanganUbah - 1].status;
        cout << "Lapangan berhasil diubah!" << endl;
    } else {
        cout << "Nomor lapangan tidak valid!" << endl;
    }
}
break;

```

Fitur ini berfungsi untuk megubah data lapangan

E. Fitur Hapus Data Lapangan

```

case 4:
    if (jumlahLapangan == 0) {
        cout << "Lapangan masih kosong!" << endl;
    } else {
        cout << "\n=== DAFTAR ID LAPANGAN ===" << endl;
        cout << "+" << setfill('-') << setw(7) << "+" <<
setw(25) << "+" << endl;
        cout << "| " << setfill(' ') << left << setw(5) <<
"No" << "| "
        << setw(23) << "ID Lapangan" << "|" << endl;
        cout << "+" << setfill('-') << setw(7) << "+" <<
setw(25) << "+" << endl;

        for (int i = 0; i < jumlahLapangan; i++) {
            cout << "| " << setfill(' ') << left << setw(5)
<< i + 1 << "| "
            << setw(23) << dataLapangan[i].id << "|" <<
endl;
        }

        cout << "+" << setfill('-') << setw(7) << "+" <<
setw(25) << "+" << endl;

        int idLapanganHapus;
        cout << "Pilih nomor lapangan yang ingin dihapus: ";
        cin >> idLapanganHapus;
    }
}

```

```

        if (idLapanganHapus > 0 && idLapanganHapus <=
jumlahLapangan) {
            for (int i = idLapanganHapus - 1; i <
jumlahLapangan - 1; i++) {
                dataLapangan[i].id = dataLapangan[i + 1].id;
                dataLapangan[i].harga = dataLapangan[i +
1].harga;
                dataLapangan[i].lokasi = dataLapangan[i +
1].lokasi;
                dataLapangan[i].status = dataLapangan[i +
1].status;
            }
            jumlahLapangan--;
            cout << "Lapangan berhasil dihapus!" << endl;
        } else {
            cout << "Nomor lapangan tidak valid!" << endl;
        }
    }
    break;

```

Fitur ini digunakan untuk menghapus data lapangan

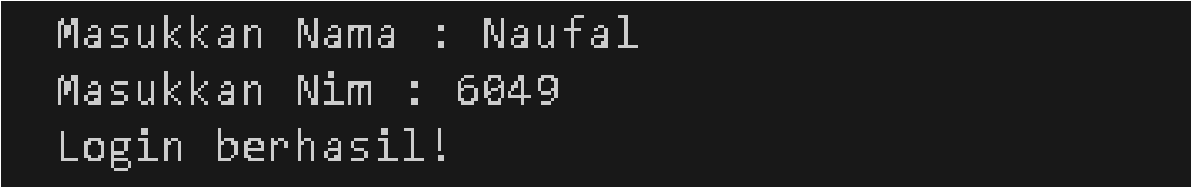
4. Uji Coba dan Hasil Output

4.1 Uji Coba

1. "Tambah dan Lihat Lapangan"
2. "Ubah dan Hapus Lapangan"

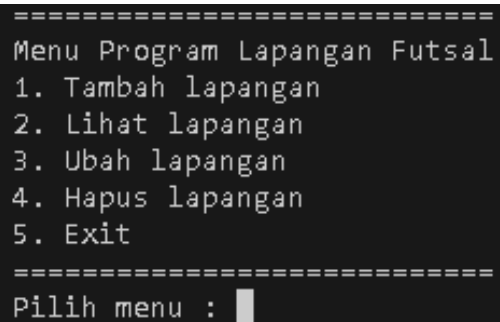
4.2 Hasil Output

A.Skenario 1



```
Masukkan Nama : Naufal
Masukkan Nim : 6049
Login berhasil!
```

Gambar 4.1 Login berhasil (skenario 1)



```
=====
Menu Program Lapangan Futsal
1. Tambah lapangan
2. Lihat lapangan
3. Ubah lapangan
4. Hapus lapangan
5. Exit
=====
Pilih menu : █
```

Gambar 4.2 Menu Utama (skenario 1)

```
=====
Menu Program Lapangan Futsal
1. Tambah lapangan
2. Lihat lapangan
3. Ubah lapangan
4. Hapus lapangan
5. Exit
=====
Pilih menu : 1
Masukkan ID lapangan : F01
Masukkan harga sewa lapangan : 15000
Masukkan lokasi lapangan : lantai-1
Masukkan status ketersediaan lapangan: TERSEDIA
Tambah Lapangan Berhasil !!

Menu Program Lapangan Futsal
1. Tambah lapangan
2. Lihat lapangan
3. Ubah lapangan
4. Hapus lapangan
5. Exit
=====
Pilih menu : 1
Masukkan ID lapangan : F02
Masukkan harga sewa lapangan : 15000
Masukkan lokasi lapangan : lantai-2
Masukkan status ketersediaan lapangan: TERSEDIA
Tambah Lapangan Berhasil !!
```

Gambar 4.3 Pengguna memilih menu 1 dan memasukkan Data lapangan (skenario 1)

```
=====
Menu Program Lapangan Futsal
1. Tambah lapangan
2. Lihat lapangan
3. Ubah lapangan
4. Hapus lapangan
5. Exit
=====
Pilih menu : 2

=== DAFTAR LAPANGAN FUTSAL ===
+-----+-----+-----+-----+
| ID Lapangan | Harga Sewa | Lokasi      | Status      |
+-----+-----+-----+-----+
| F01         | 15000      | lantai-1    | TERSEDIA    |
| F02         | 15000      | lantai-2    | TERSEDIA    |
+-----+-----+-----+-----+
```

Gambar 4.4 Pengguna memilih menu 2 dan melihat data lapangan yang tersedia (skenario 1)

```
=====
Menu Program Lapangan Futsal
1. Tambah lapangan
2. Lihat lapangan
3. Ubah lapangan
4. Hapus lapangan
5. Exit
=====
Pilih menu : 5
Keluar dari program.
```

Gambar 4.5 Pengguna memilih menu 5 untuk keluar dari program (skenario 1)

B. Skenario 2

```
=====
Menu Program Lapangan Futsal
1. Tambah lapangan
2. Lihat lapangan
3. Ubah lapangan
4. Hapus lapangan
5. Exit
=====
Pilih menu : 3

=== DAFTAR ID LAPANGAN ===
++-----+-----+
| No  | ID Lapangan |
++-----+-----+
| 1   | F01         |
| 2   | F02         |
++-----+-----+
Masukkan nomor lapangan yang akan diubah: 1
Masukkan harga sewa lapangan baru: 20000
Masukkan lokasi lapangan baru: Lantai-1
Masukkan status ketersediaan lapangan baru: Booked
Lapangan berhasil diubah!
```

Gambar 4.5 Pengguna memilih menu 3 untuk memilih id yang ingin diubah (skenario 2)

```
=====
Menu Program Lapangan Futsal
1. Tambah lapangan
2. Lihat lapangan
3. Ubah lapangan
4. Hapus lapangan
5. Exit
=====
Pilih menu : 2

=== DAFTAR LAPANGAN FUTSAL ===
++-----+-----+-----+-----+
| ID Lapangan | Harga Sewa | Lokasi      | Status      |
++-----+-----+-----+-----+
| F01         | 20000      | Lantai-1    | Booked      |
| F02         | 15000      | lantai-2    | TERSEDIA    |
++-----+-----+-----+-----+
```

Gambar 4.6 Pengguna kembali ke menu 2 untuk memastikan data berubah(skenario 2)

```
=====
Menu Program Lapangan Futsal
1. Tambah lapangan
2. Lihat lapangan
3. Ubah lapangan
4. Hapus lapangan
5. Exit
=====
Pilih menu : 4

=== DAFTAR ID LAPANGAN ===
++-----+-----+
| No  | ID Lapangan |
++-----+-----+
| 1   | F01         |
| 2   | F02         |
++-----+-----+
Pilih nomor lapangan yang ingin dihapus: 2
Lapangan berhasil dihapus!
=====
```

Gambar 4.7 Pengguna memilih menu 4 dan memasukkan id yang ingin dihapus(skenario 2)

```

=====
Menu Program Lapangan Futsal
1. Tambah lapangan
2. Lihat lapangan
3. Ubah lapangan
4. Hapus lapangan
5. Exit
=====
Pilih menu : 2

=== DAFTAR LAPANGAN FUTSAL ===
++-----+-----+-----+-----+
| ID Lapangan| Harga Sewa | Lokasi      | Status      |
++-----+-----+-----+-----+
| F01        | 20000      | Lantai-1    | Booked      |
++-----+-----+-----+-----+

```

Gambar 4.8 Pengguna kembali ke menu 2 untuk melihat apakah data sudah dihapus (skenario 2)

```

=====
Menu Program Lapangan Futsal
1. Tambah lapangan
2. Lihat lapangan
3. Ubah lapangan
4. Hapus lapangan
5. Exit
=====
Pilih menu : 5
Keluar dari program.

```

Gambar 4.9 Pengguna memilih menu 5 untuk keluar dari program (skenario 2)

5. Langkah - Langkah Git

A. Git Init

```
USER@DESKTOP-3005TS7 MINGW64 /d/Praktikum APL (main)
• $ git init
Reinitialized existing Git repository in D:/Praktikum APL/.git/
```

Gambar 5.1 Git init

Digunakan untuk menginisialisasi sebuah repository Git dalam folder yang sedang aktif.

B. Git Add

```
USER@DESKTOP-3005TS7 MINGW64 /d/Praktikum APL (main)
• $ git add .
```

Gambar 5.2 Git add

Digunakan untuk menambahkan file apa saja yang ingin kita commit selanjutnya.

C. Git Commit

```
USER@DESKTOP-3005TS7 MINGW64 /d/Praktikum APL (main)
• $ git commit -m "posttest 1 selesai"
[main e84023a] posttest 1 selesai
6 files changed, 390 insertions(+), 299 deletions(-)
delete mode 100644 Kelas/New Text Document.txt
rename Posttest/Posttest-1/{2409106049-MuhammadNaufalAdiBrata-PT-1 .cpp => 2409106049-MuhammadNaufalAdiBrata-PT-1.cpp} (88%
create mode 100644 Posttest/Posttest-1/2409106049-MuhammadNaufalAdiBrata-PT-1.drawio
create mode 100644 Posttest/Posttest-1/2409106049-MuhammadNaufalAdiBrata-PT-1.pdf
delete mode 100644 Posttest/Posttest-1/main.exe
delete mode 100644 Posttest/Posttest-1/posttest 1.drawio
```

Gambar 5.3 Git commit

Digunakan untuk membuat checkpoint pada file

D. Git Remote

```
USER@DESKTOP-3005TS7 MINGW64 /d/Praktikum APL (main)
• $ git remote add origin https://github.com/Naufal-oboy/Praktikum-APL.git
```

Gambar 5.4 Git remote

Digunakan untuk menghubungkan repository yang ada di lokal komputer kita dengan repository cloud pada Github.

E Git Push


```
USER@DESKTOP-3005TS7 MINGW64 /d/Praktikum APL (main)
● $ git push -u origin main
info: please complete authentication in your browser...
Enumerating objects: 16, done.
Counting objects: 100% (16/16), done.
Delta compression using up to 16 threads
Compressing objects: 100% (11/11), done.
Writing objects: 100% (16/16), 1.04 MiB | 709.00 KiB/s, done.
Total 16 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/Naufal-oboy/Praktikum-APL.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
```

Gambar 5.5 Git push

Digunakan untuk mengupload semua hal yang ada pada repository lokal kita ke Github.