

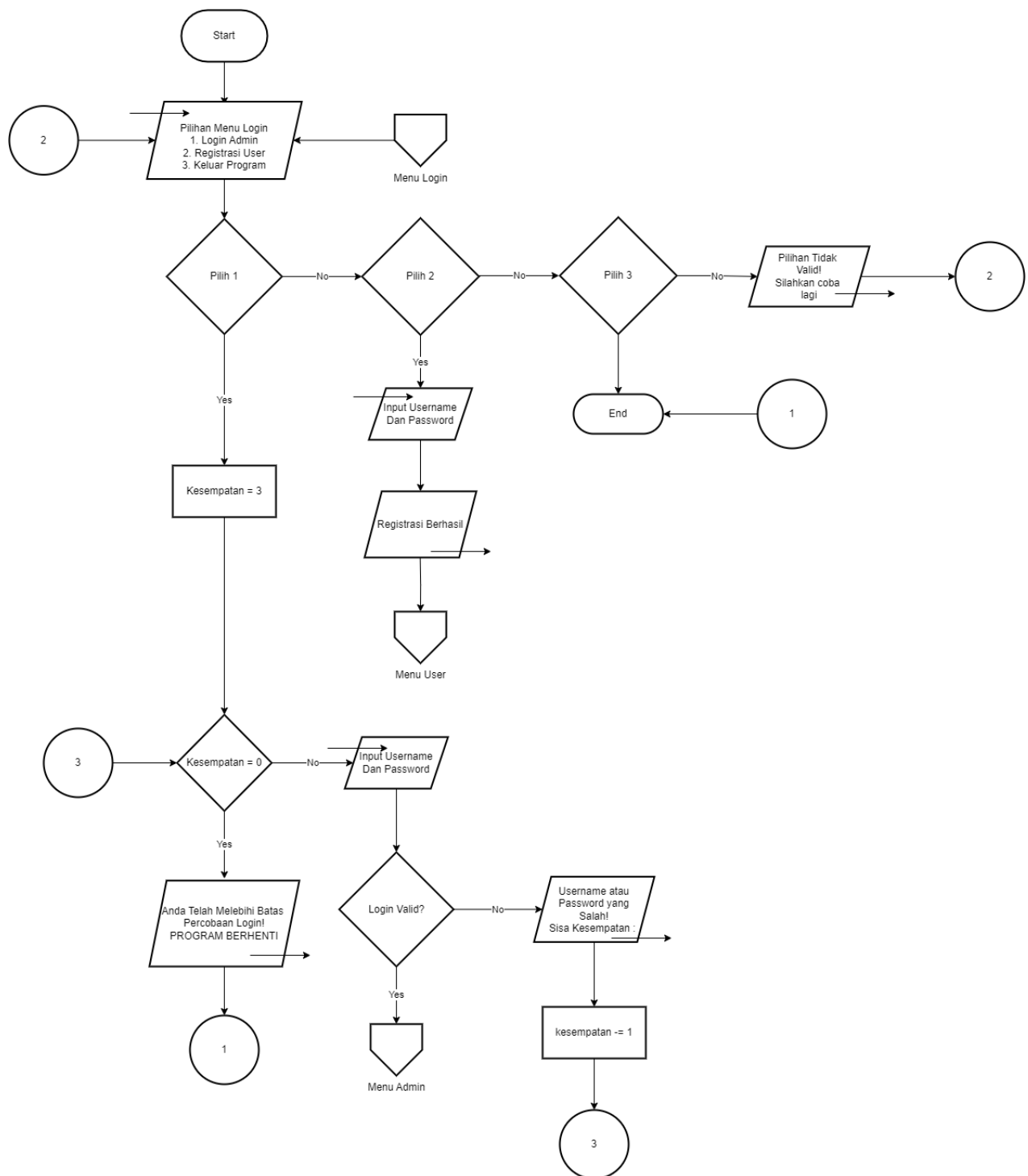
LAPORAN PRAKTIKUM
POSTTEST 6
ALGORITMA PEMROGRAMAN LANJUT



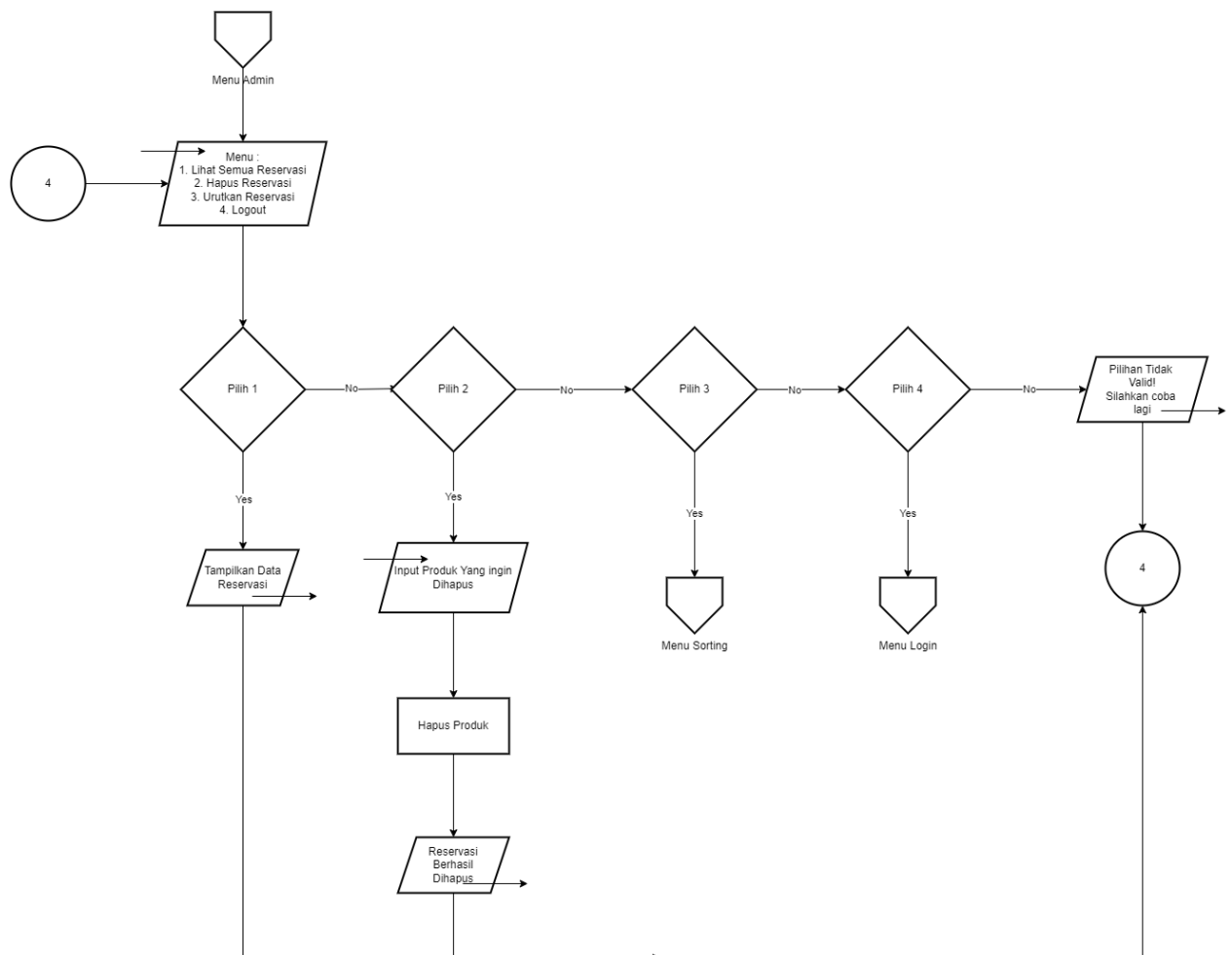
Disusun oleh:
Anindtya Puji Astari (2409106063)
Kelas B1 '24

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

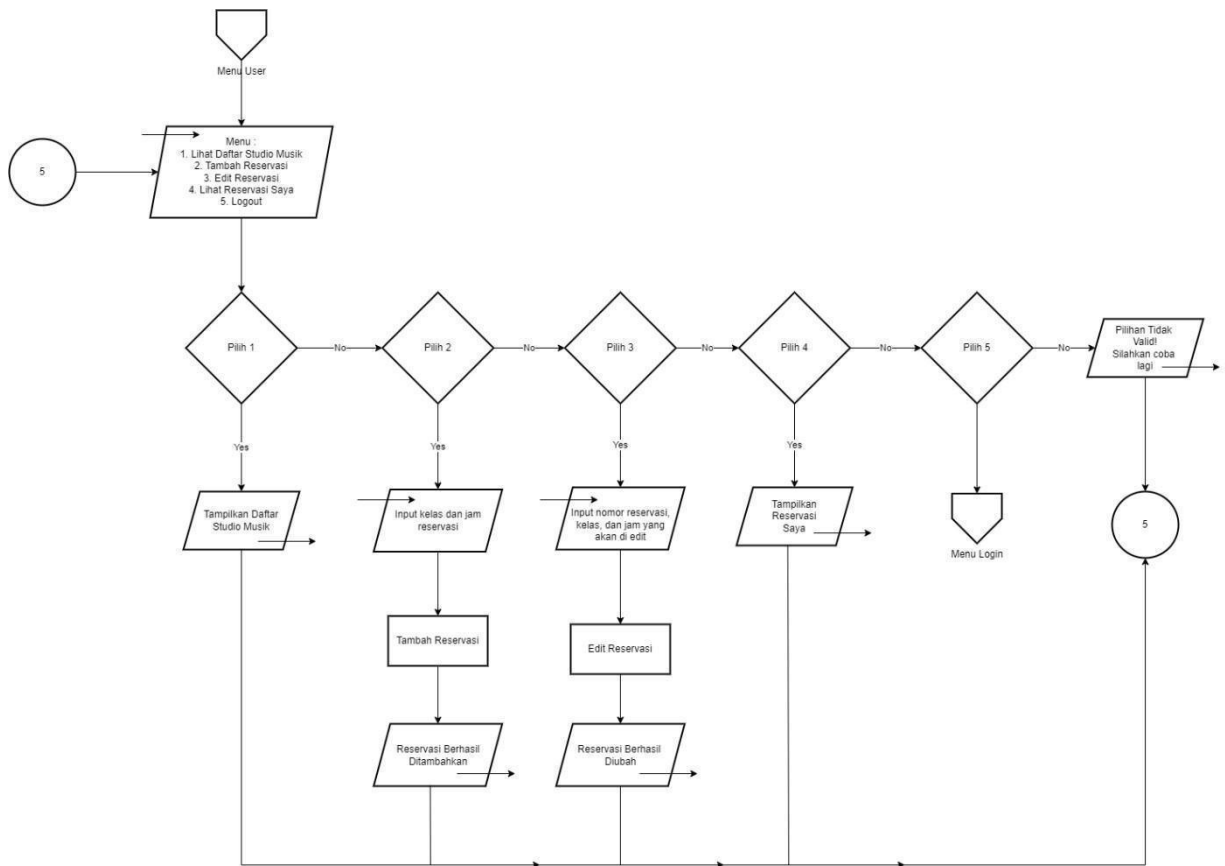
1. Flowchart



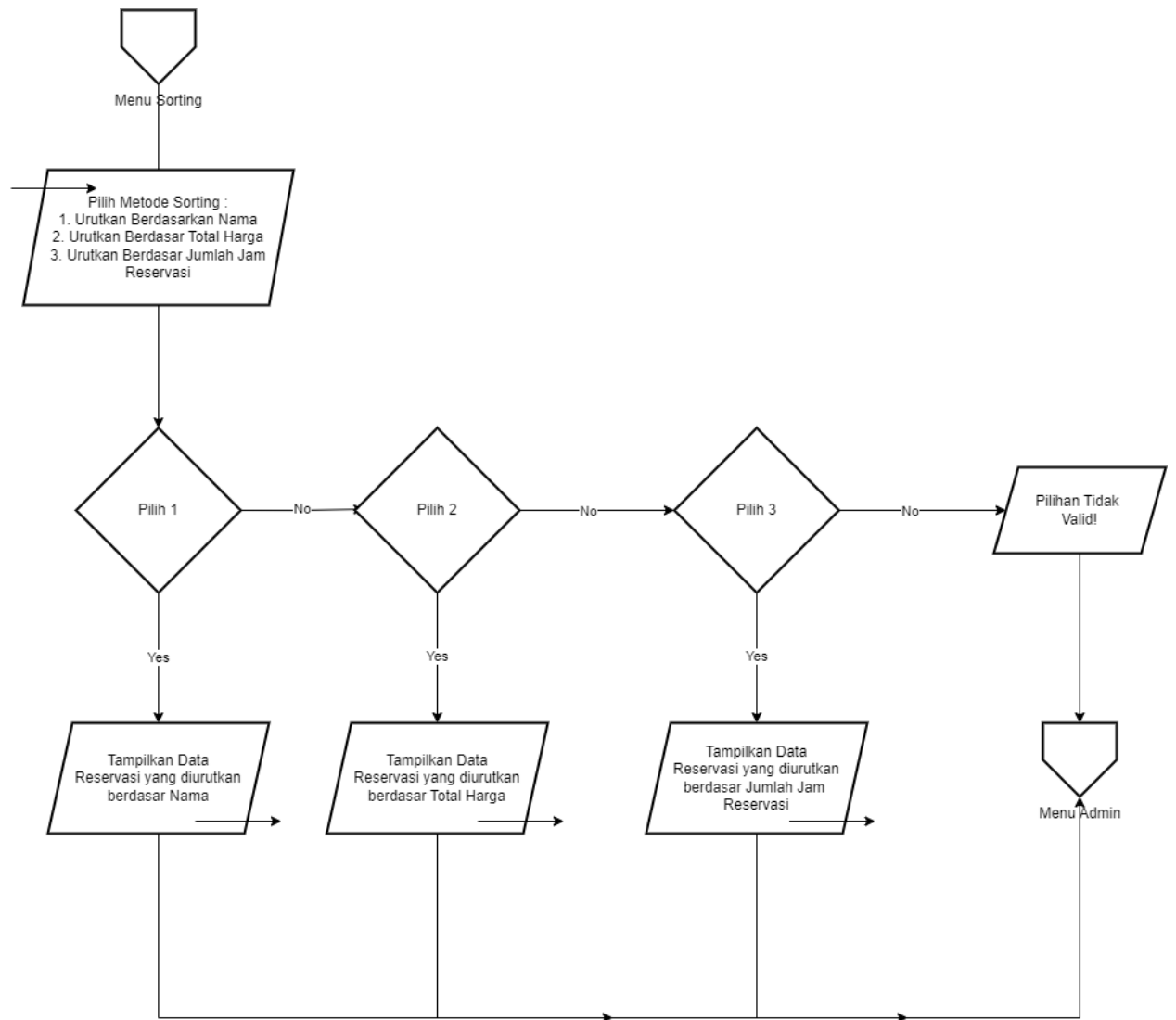
Gambar 1.1 Flowchart Menu Login



Gambar 1.2 Flowchart Menu Admin



Gambar 1.3 Flowchart Menu User



Gambar 1.4 Flowchart Menu Sorting

2. Deskripsi Singkat Program

Program ini merupakan sistem reservasi studio musik yang memungkinkan pengguna untuk melakukan pemesanan studio musik. Program menyediakan dua jenis akses, yaitu admin dan user. Menu admin terbatas pada dapat melihat seluruh reservasi dan menghapus reservasi, sedangkan user dapat melihat daftar studio, melakukan reservasi, mengedit reservasi, serta melihat reservasi pribadi yang sudah dibuat. Program ini memberikan kemudahan akses, mempermudah proses pemesanan studio music, mempermudah pengelolaan data reservasi karena lebih sistematis, lebih aman, dan membantu efisiensi waktu dan tenaga.

3. Source Code

A. Fungsi Login Admin

Fitur ini digunakan untuk login admin dengan batas percobaan 3 kali.

Source Code:

```
bool loginAdmin()
{
    string username, password;
    int kesempatan = 3;
    while (kesempatan > 0)
    {
        cout << "\n--===LOGIN ADMIN===--\nMasukkan Username: ";
        cin >> username;
        cout << "Masukkan Password: ";
        cin >> password;
        if (username == "tea" && password == "063") return true;
        kesempatan--;
        cout << "Username atau Password Salah!\nSisa kesempatan: " <<
kesempatan << endl;
    }
    return false;
}
```

B. Fungsi Registrasi User

Fitur ini digunakan untuk registrasi sekaligus login user.

Source Code:

```
string registrasiUser(string *nama, string *nim)
{
    cout << "\n--===REGISTRASI USER===--\nMasukkan Nama: ";
    cin.ignore();
    getline(cin, *nama);
    cout << "Masukkan NIM: ";
    getline(cin, *nim);
    return "Registrasi Berhasil!\n\n";
}
```

C. Fungsi Tampilkan Daftar Studio

Fitur ini digunakan untuk menampilkan Daftar Studio Musik mulai dari kelas, alat musik, dan harga per jamnya.

Source Code:

```
string tampilkanDaftarStudio()
{
    return tampilkanDaftarStudio(daftarStudio, alatMusik);
}

string tampilkanDaftarStudio(Studio *daftarStudio, string (*alatMusik)[7])
{
    string output =
    "\n=====
=====\\n";    output +=
    "| No | Kelas   | Alat Musik
    | Harga Per Jam |\\n";    output +=
    "=====
=====\\n";    for (int i = 0; i < 2; i++) {
    output += "| " + to_string(i + 1) + " | " + (*(daftarStudio +
    i)).namaKelas + " | ";    for (int j = 0; j < 7; j++) {
    output += (*(alatMusik + i) + j);    if (j < 6) output += ",
    ";
        }    output += " | Rp " + to_string(*(daftarStudio +
    i)).hargaPerJam) + "
    |\\n";
    }
    output +=
    "=====
=====\\n";
    return output;
}
```

D. Fungsi Tambah Reservasi

Fitur ini digunakan untuk Reservasi Studio Musik oleh User.

Source Code:

```
string tambahReservasiUser(string nama, string nim, Reservasi *daftar, int
*jumlah, Studio *daftarStudio)
{
    if (*jumlah >= MAX_RESERVASI) return "Reservasi penuh!\n";

    Reservasi r;
    r.nama = nama;
    r.nim = nim;

    cout << tampilkanDaftarStudio();
    int pilihan;    cout << "Pilih Kelas
Studio (1/2): ";    cin >> pilihan;

    if (pilihan < 1 || pilihan > 2) return "Pilihan tidak valid!\n";

    r.studio = *(daftarStudio + pilihan - 1);
    cout << "Masukkan jumlah jam: ";    cin >>
    r.jam;

    int *jamPtr = &r.jam;    int
    *hargaPtr = &r.studio.hargaPerJam;
    r.totalHarga = (*jamPtr) * (*hargaPtr);

    *(daftar + (*jumlah)++) = r;    return
    "Reservasi berhasil ditambahkan!\n"; }
```

E. Fungsi Edit Reservasi

Fitur ini digunakan untuk Mengedit Reservasi Studio Musik oleh User.

Source Code:

```
string editReservasiUser(string nim, Reservasi daftar[], int jumlah, Studio
daftarStudio[])
{
```

```

    int urutan = 0;

    cout << "\nReservasi Anda:\n";    cout <<
    "=====\n";    cout << "| No | Kelas
    | Jam | Total Harga      |\n";    cout <<
    "=====\n";    for (int i = 0; i <
    jumlah; i++) {        if (daftar[i].nim == nim) {            urutan++;
    cout << "| " << urutan << " | " << daftar[i].studio.namaKelas
        << " | " << daftar[i].jam
        << " | Rp " << daftar[i].totalHarga << "      |\n";
    }
    }

    if (urutan == 0) {        cout << "|            Tidak ada
    reservasi      |\n";        cout <<
    "=====\n";        return
    "Tidak ada reservasi yang bisa diedit.\n";    }

    cout << "=====\n";

    int no, pilihan, jam;    cout << "Masukkan nomor
    reservasi yang akan diedit: ";    cin >> no;

    urutan = 0;    for (int i = 0; i < jumlah; i++) {        if
    (daftar[i].nim == nim) {            urutan++;            if (urutan ==
    no) {                cout << "\nPilih Kelas Studio Baru:\n";
    cout << "1. Kelas A (Rp " << daftarStudio[0].hargaPerJam <<
    "/jam)\n";                cout << "2. Kelas B (Rp " <<
    daftarStudio[1].hargaPerJam <<
    "/jam)\n";                cout <<
    "Pilihan(1/2): ";                cin >>
    pilihan;

                if (pilihan < 1 || pilihan > 2)
    return "Pilihan Tidak Valid!\n";

                daftar[i].studio = daftarStudio[pilihan - 1];
    cout << "Masukkan jumlah jam baru: ";                cin >>
    jam;                daftar[i].jam = jam;                int
    *jamBaru = &jam;

```

```

        int *hargaBaru = &daftarStudio[pilihan - 1].hargaPerJam;
daftar[i].totalHarga = (*jamBaru) * (*hargaBaru);

        return "Reservasi Berhasil Diperbarui!\n";
    }
}

return "Reservasi Tidak Ditemukan!\n";
}

        daftar[i].studio = daftarStudio[pilihan - 1];
cout << "Masukkan jumlah jam baru: ";        cin >> jam;
daftar[i].jam = jam;        daftar[i].totalHarga = jam *
daftarStudio[pilihan -
1].hargaPerJam;        return "Reservasi Berhasil
Diperbarui!\n";
    }
}

return "Reservasi Tidak Ditemukan!\n";
}

```

F. Fungsi Hapus Reservasi

Fitur ini digunakan untuk Hapus Reservasi Studio Musik oleh Admin.

Source Code:

```

string hapusReservasiAdmin(Reservasi daftar[], int &jumlah)
{
    if (jumlah == 0) return "Tidak Ada Reservasi Yang Bisa Dihapus.\n";
    tampilkanReservasiAdmin(daftar, jumlah);
    int index;
    cout << "Masukkan nomor reservasi yang akan dihapus: ";
    cin >> index;
    if (index < 1 || index > jumlah) return "Nomor Reservasi Tidak
Valid!\n";
    for (int i = index - 1; i < jumlah - 1; i++)

```

```
        daftar[i] = daftar[i + 1];  
    jumlah--;  
    return "Reservasi Berhasil Dihapus!\n";  
}  
  
jumlah--;  
return "Reservasi Berhasil Dihapus!\n";  
}
```

G. Fungsi Tampilkan Reservasi User

Fitur ini digunakan untuk Menampilkan Reservasi Studio Musik oleh User.

Source Code:

```

void tampilkanReservasiUser(Reservasi daftarReservasi[], int jumlah,
string nim, int i = 0, int no = 1, bool ada = false)
{
    if (i == 0) {        cout << "\nReservasi Anda:\n";
cout << "=====\n";
cout << "| No | Kelas | Jam | Total Harga | \n";
cout << "=====\n";    }

    if (i >= jumlah) {        if (!ada) cout << " |          Tidak ada
reservasi | \n";        cout <<
"=====\n";        return;
    }

    if (daftarReservasi[i].nim == nim) {
cout << " | " << no << " | " <<
daftarReservasi[i].studio.namaKelas
        << " | " << daftarReservasi[i].jam
        << " | Rp " << daftarReservasi[i].totalHarga << "
| \n";        tampilkanReservasiUser(daftarReservasi, jumlah, nim, i + 1,
no + 1, true);    } else {        tampilkanReservasiUser(daftarReservasi,
jumlah, nim, i + 1, no, ada);    }
}

```

H. Fungsi Tampilkan Reservasi Admin

Fitur ini digunakan untuk Menampilkan Reservasi Studio Musik oleh Admin.

Source Code:

```

void tampilkanReservasiAdmin(Reservasi daftar[], int jumlah)
{
    cout <<
"\n=====\n"
;
    cout << " | No | Nama | NIM | Kelas | Jam | Total

```

```

Harga |\n";
cout <<
"=====|\n";

    for (int i = 0; i < jumlah; i++) {          cout << "| " << i + 1
<< " | " << daftar[i].nama;          cout << string(21 -
daftar[i].nama.length(), ' ') << "| " << daftar[i].nim
        << " | " << daftar[i].studio.namaKelas << " | " << daftar[i].jam
        << " | Rp " << daftar[i].totalHarga << " | \n";
    }

    if (jumlah == 0)          cout << "|          Tidak
Ada Reservasi Saat Ini          | \n";

    cout <<
"=====|\n"; }

```

I. Fungsi Sorting Nama

Fitur ini digunakan untuk Mengurutkan Daftar Reservasi Berdasar Nama secara Ascending menggunakan Bubble Sort.

Source Code:

```

void bubbleSortNama(Reservasi daftar[], int jumlah)
{
    for (int i = 0; i < jumlah - 1; i++)
    {
        for (int j = 0; j < jumlah - i - 1; j++)
        {
            if (daftar[j].nama > daftar[j+1].nama)
            {
                Reservasi temp = daftar[j];
                daftar[j] = daftar[j+1];
                daftar[j+1] = temp;
            }
        }
    }
}

```

J. Fungsi Sorting Total Harga

Fitur ini digunakan untuk Mengurutkan Daftar Reservasi Berdasar Total Harga secara Descending menggunakan Insertion Sort.

Source Code:

```
void insertionSortTotalHargaDesc(Reservasi daftar[], int jumlah)
{
    for (int i = 1; i < jumlah; i++)
    {
        Reservasi key = daftar[i];
        int j = i - 1;
        while (j >= 0 && daftar[j].totalHarga < key.totalHarga)
        {
            daftar[j+1] = daftar[j];
            j--;
        }
        daftar[j+1] = key;
    }
}
```

K. Fungsi Sorting Total Harga

Fitur ini digunakan untuk Mengurutkan Daftar Reservasi Berdasar Jumlah Jam Reservasi secara Ascending menggunakan Selection Sort.

Source Code:

```
void selectionSortJam(Reservasi daftar[], int jumlah)
{
    for (int i = 0; i < jumlah - 1; i++)
    {
        int minIdx = i;
        for (int j = i + 1; j < jumlah; j++)
        {
            if (daftar[j].jam < daftar[minIdx].jam)
            {

```

```

        minIdx = j;
    }
}
Reservasi temp = daftar[i];
daftar[i] = daftar[minIdx];
daftar[minIdx] = temp;
}
}

```

L. Menu Sorting

Source Code:

```

void menuSorting(Reservasi daftar[], int jumlah)
{
    int pilihSort;
    cout << "\nPilih Metode Sorting:\n";
    cout << "1. Urutkan Berdasarkan Nama\n";
    cout << "2. Urutkan Berdasarkan Total Harga\n";
    cout << "3. Urutkan Berdasarkan Jumlah Jam Reservasi\n";
    cout << "Pilihan: ";
    cin >> pilihSort;
    switch(pilihSort)
    {
        case 1:
            bubbleSortNama(daftar, jumlah);
            cout << "\nData berhasil diurutkan berdasarkan Nama
secara Ascending!\n";
            tampilkanReservasiAdmin(daftar, jumlah);
            break;
        case 2:
            insertionSortTotalHargaDesc(daftar, jumlah);
            cout << "\nData berhasil diurutkan berdasarkan Total
Harga secara Descending!\n";
            tampilkanReservasiAdmin(daftar, jumlah);
            break;
        case 3:
            selectionSortJam(daftar, jumlah);
            cout << "\nData berhasil diurutkan berdasarkan Jumlah

```



```

Jam secara Ascending!\n";
    tampilkanReservasiAdmin(daftar, jumlah);
    break;
default:
    cout << "\nPilihan tidak valid!\n";
}
}

```

M. Menu Utama

Source Code:

```

int main()
{
    int pilihan;
    bool isAdmin = false;
    string namaUser, nimUser;
    tambahReservasiAwal(daftarReservasi, &jumlahReservasi,
daftarStudio);
    while (true)
    {
        cout << "\n\nHELLO";
        cout << "\nSelamat Datang di Sistem Reservasi Studio
Musik\n";
        cout << "1. Login Admin\n2. Registrasi User\n3. Keluar
Program\nPilih: ";
        cin >> pilihan;
        if (pilihan == 1)
        {
            isAdmin = loginAdmin();
            if (!isAdmin)
            {
                cout << "Anda Telah Melebihi Batas Percobaan
Login!\nPROGRAM BERHENTI.\n";
                return 0;
            }
        }
    }
}

```

```

    }
    else if (pilihan == 2)
    {
        cout << registrasiUser(&namaUser, &nimUser);
        isAdmin = false;
    }
    else if (pilihan == 3)
    {
        cout << "\nTerima kasih telah menggunakan sistem
ini.\n";
        return 0;
    }
    else
    {
        cout << "\nPilihan Tidak Valid! Silakan coba
lagi.\n\n";
        continue;
    }

    while (true)
    {
        cout << "\nMENU:\n";
        if (isAdmin)
        {
            cout << "1. Lihat Semua Reservasi\n2. Hapus
Reservasi\n3. Urutkan Reservasi\n4. Logout\n";
        }
        else
        {
            cout << "1. Lihat Daftar Studio Musik\n2. Tambah
Reservasi\n3. Edit Reservasi\n4. Lihat Reservasi Saya\n5.
Logout\n";
        }
        cout << "Pilih menu: ";
        cin >> pilihan;

        if (isAdmin)
        {
            if (pilihan == 1)
tampilkanReservasiAdmin(daftarReservasi, jumlahReservasi);
            else if (pilihan == 2) cout <<
hapusReservasiAdmin(daftarReservasi, jumlahReservasi);

```

```
else if (pilihan == 3) menuSorting(daftarReservasi,
jumlahReservasi);
    else if (pilihan == 4)
    {
        cout << "\nAnda Berhasil Logout\n\n";
        break;
    }
    else
    {
        cout << "\nPilihan Tidak Valid!\nSilakan coba
lagi.\n\n";
    }
}
else
{
    if (pilihan == 1) cout << tampilkanDaftarStudio();
    else if (pilihan == 2) cout <<
tambahReservasiUser(namaUser, nimUser, daftarReservasi,
&jumlahReservasi, daftarStudio);
    else if (pilihan == 3) cout <<
editReservasiUser(nimUser, daftarReservasi, jumlahReservasi,
daftarStudio);
    else if (pilihan == 4)
tampilkanReservasiUser(daftarReservasi, jumlahReservasi, nimUser);
    else if (pilihan == 5)
    {
        cout << "\nAnda Berhasil Logout\n\n";
        break;
    }
    else
    {
        cout << "\nPilihan Tidak Valid!\nSilakan coba
lagi.\n\n";
    }
}
}
}
```

4. Output

```
HELLO
Selamat Datang di Sistem Reservasi Studio Musik
1. Login Admin
2. Registrasi User
3. Keluar Program
Pilih: 1

====LOGIN ADMIN====
Masukkan Username: tea
Masukkan Password: 063

MENU:
1. Lihat Semua Reservasi
2. Hapus Reservasi
3. Urutkan Reservasi
4. Logout
Pilih menu: 1

=====
| No | Nama          | NIM    | Kelas | Jam | Total Harga |
=====
| 1  | ipeh          | 202101 | Kelas A | 3  | Rp 240000   |
| 2  | aryo          | 202102 | Kelas B | 2  | Rp 200000   |
| 3  | araa          | 202103 | Kelas A | 5  | Rp 400000   |
| 4  | inun          | 202104 | Kelas B | 4  | Rp 400000   |
=====
```

Gambar 4.1 lihat semua reservasi

```
HELLO
Selamat Datang di Sistem Reservasi Studio Musik
1. Login Admin
2. Registrasi User
3. Keluar
Pilih: 2

====REGISTRASI USER====
Masukkan Nama: tya
Masukkan NIM: 1
Registrasi Berhasil!

MENU:
1. Lihat Daftar Studio Musik
2. Tambah Reservasi
3. Edit Reservasi
4. Lihat Reservasi Saya
5. Logout
Pilih menu: 2

=====
| No | Kelas | Alat Musik                                     | Harga Per Jam |
=====
| 1  | Kelas A | Drum, Keyboard, Gitar Akustik, Gitar Elektrik, Bass, Cajon, Mic | Rp 80000       |
| 2  | Kelas B | Piano, Violin, Cello, Flute, Terompet, Saxofon, Harpa          | Rp 100000      |
=====
Pilih Kelas Studio (1/2): 1
Masukkan jumlah jam: 5
Reservasi Berhasil Ditambahkan!
```

Gambar 4.2 tambah reservasi

```

Pilih menu: 3

Reservasi Anda:
=====
| No | Kelas   | Jam | Total Harga |
=====
| 5  | Kelas A | 5   | Rp 400000   |
=====
Masukkan nomor reservasi yang akan diedit: 5

Pilih Kelas Studio Baru:
1. Kelas A (Rp 80000/jam)
2. Kelas B (Rp 100000/jam)
Pilihan: 2
Masukkan jumlah jam baru: 8
Reservasi Berhasil Diperbarui!

```

Gambar 4.3 edit reservasi

```

Pilih menu: 4

Reservasi Anda:
=====
| No | Kelas   | Jam | Total Harga |
=====
| 5  | Kelas B | 8   | Rp 800000   |
=====

```

Gambar 4.4 reservasi pribadi

```

MENU:
1. Lihat Semua Reservasi
2. Hapus Reservasi
3. Urutkan Reservasi
4. Logout
Pilih menu: 2

=====
| No | Nama           | NIM      | Kelas | Jam | Total Harga |
=====
| 1  | ipeh           | 202101   | Kelas A | 3  | Rp 240000   |
| 2  | aryo           | 202102   | Kelas B | 2  | Rp 200000   |
| 3  | araa           | 202103   | Kelas A | 5  | Rp 400000   |
| 4  | inun           | 202104   | Kelas B | 4  | Rp 400000   |
=====

Masukkan nomor reservasi yang akan dihapus: 3
Reservasi Berhasil Dihapus!

MENU:
1. Lihat Semua Reservasi
2. Hapus Reservasi
3. Urutkan Reservasi
4. Logout
Pilih menu: 1

=====
| No | Nama           | NIM      | Kelas | Jam | Total Harga |
=====
| 1  | ipeh           | 202101   | Kelas A | 3  | Rp 240000   |
| 2  | aryo           | 202102   | Kelas B | 2  | Rp 200000   |
| 3  | inun           | 202104   | Kelas B | 4  | Rp 400000   |
=====

```

Gambar 4.6 hapus reservasi

```

MENU:
1. Lihat Semua Reservasi
2. Hapus Reservasi
3. Urutkan Reservasi
4. Logout
Pilih menu: 3

Pilih Metode Sorting:
1. Urutkan Berdasarkan Nama
2. Urutkan Berdasarkan Total Harga
3. Urutkan Berdasarkan Jumlah Jam Reservasi
Pilihan: 1

Data berhasil diurutkan berdasarkan Nama secara Ascending!

=====
| No | Nama           | NIM      | Kelas | Jam | Total Harga |
=====
| 1  | araa           | 202103   | Kelas A | 5  | Rp 400000   |
| 2  | aryo           | 202102   | Kelas B | 2  | Rp 200000   |
| 3  | inun           | 202104   | Kelas B | 4  | Rp 400000   |
| 4  | ipeh           | 202101   | Kelas A | 3  | Rp 240000   |
=====

```

Gambar 4.7 sort berdasar nama

```

MENU:
1. Lihat Semua Reservasi
2. Hapus Reservasi
3. Urutkan Reservasi
4. Logout
Pilih menu: 3

Pilih Metode Sorting:
1. Urutkan Berdasarkan Nama
2. Urutkan Berdasarkan Total Harga
3. Urutkan Berdasarkan Jumlah Jam Reservasi
Pilihan: 2

Data berhasil diurutkan berdasarkan Total Harga secara Descending!

=====
| No | Nama          | NIM      | Kelas | Jam | Total Harga |
=====
| 1  | araa          | 202103   | Kelas A | 5  | Rp 400000   |
| 2  | inun          | 202104   | Kelas B | 4  | Rp 400000   |
| 3  | ipeh          | 202101   | Kelas A | 3  | Rp 240000   |
| 4  | aryo          | 202102   | Kelas B | 2  | Rp 200000   |
=====

```

Gambar 4.8 sort berdasar total harga

```

MENU:
1. Lihat Semua Reservasi
2. Hapus Reservasi
3. Urutkan Reservasi
4. Logout
Pilih menu: 3

Pilih Metode Sorting:
1. Urutkan Berdasarkan Nama
2. Urutkan Berdasarkan Total Harga
3. Urutkan Berdasarkan Jumlah Jam Reservasi
Pilihan: 3

Data berhasil diurutkan berdasarkan Jumlah Jam secara Ascending!

=====
| No | Nama          | NIM      | Kelas | Jam | Total Harga |
=====
| 1  | aryo          | 202102   | Kelas B | 2  | Rp 200000   |
| 2  | ipeh          | 202101   | Kelas A | 3  | Rp 240000   |
| 3  | inun          | 202104   | Kelas B | 4  | Rp 400000   |
| 4  | araa          | 202103   | Kelas A | 5  | Rp 400000   |
=====

```

Gambar 4.9 sort berdasar jumlah jam

5. Git

5.1 Git Add

```
Lenovo@LAPTOP-2R0GRAM4 MINGW64 ~/Desktop/Praktikum-APL (main)
$ git add .
```

Git add berfungsi untuk memasukkan file ke staging area agar siap untuk commit.

5.2 Git Commit

```
Lenovo@LAPTOP-2R0GRAM4 MINGW64 ~/Desktop/Praktikum-APL (main)
$ git commit -m "coba upload"
[main 5b02069] coba upload
3 files changed, 390 insertions(+)
create mode 100644 Post-test/Post-test-APL-6/2409106063-AnindtyaPujiAstari-PT-6.cpp
create mode 100644 Post-test/Post-test-APL-6/2409106063-AnindtyaPujiAstari-PT-6.exe
```

Git commit berfungsi menyimpan perubahan di repository lokal. Pesan di -m "..." menjelaskan perubahan yang dilakukan.

5.3 Git push origin main

```
Lenovo@LAPTOP-2R0GRAM4 MINGW64 ~/Desktop/Praktikum-APL (main)
$ git push origin main
Enumerating objects: 12, done.
Counting objects: 100% (12/12), done.
Delta compression using up to 16 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (8/8), 789.89 KiB | 3.87 MiB/s, done.
Total 8 (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/Yyak135/Praktikum-APL.git
 eae8c04..5b02069 main -> main
```

Git push origin main mengunggah commit yang ada di branch main ke repository GitHub. Jika branch belum ada di GitHub, jalankan git push -u origin main agar branch main menjadi default untuk push berikutnya.