

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



**Disusun oleh:**

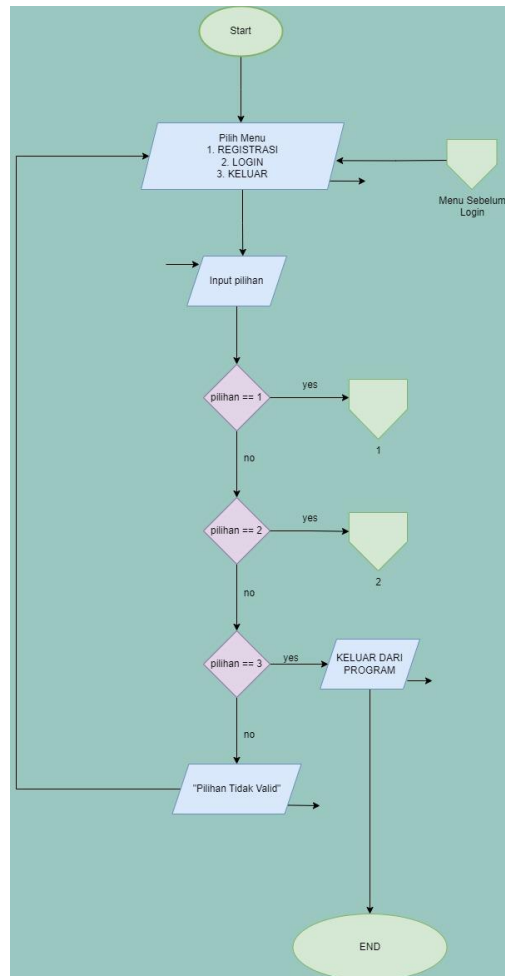
**Rusdiana**

**(2409106021)**

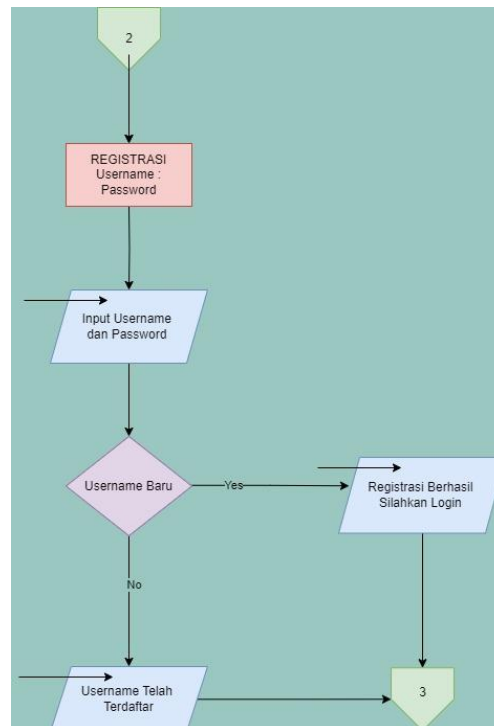
**Kelas A1'24**

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

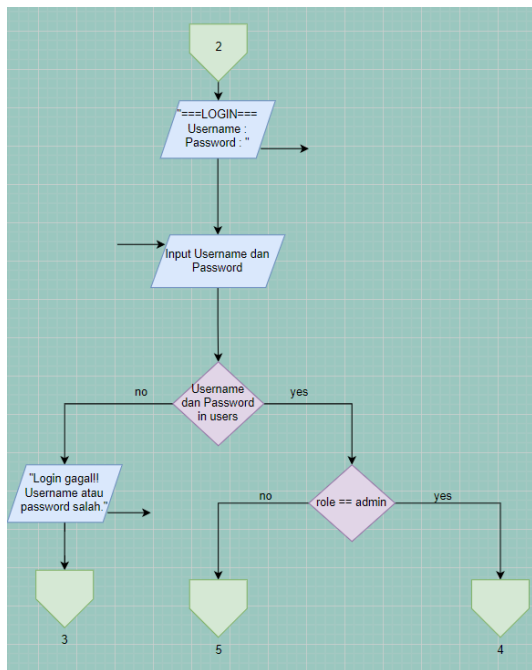
## 1. Flowchart



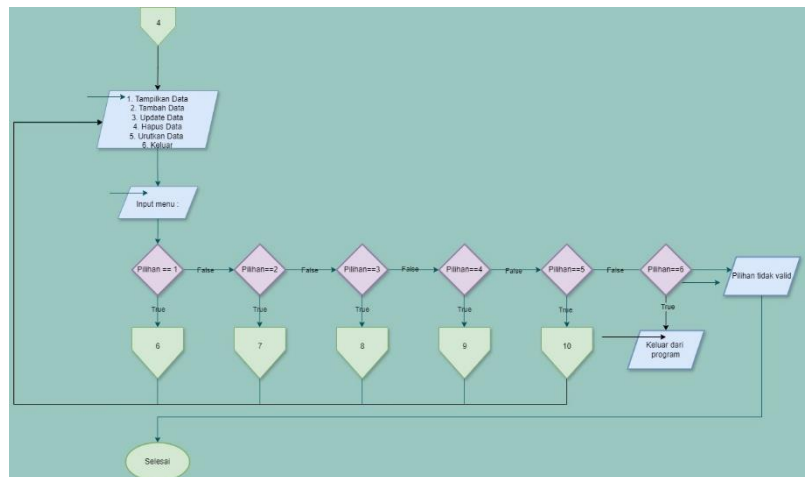
Gambar 1.1 Menu Sebelum Login



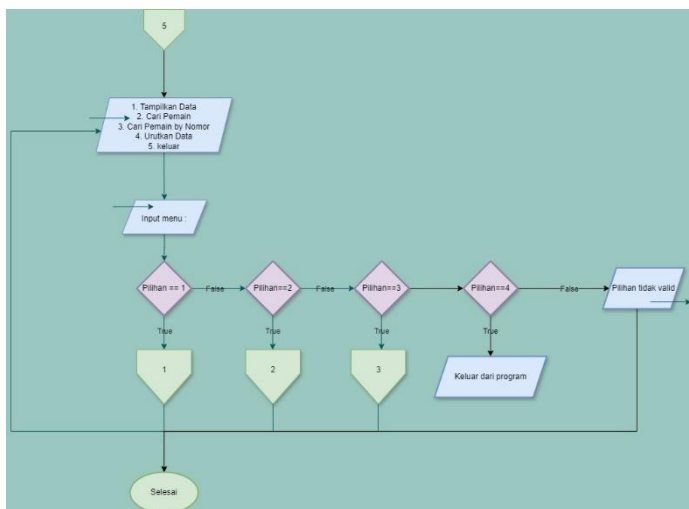
Gambar 1.2 Menu Registrasi



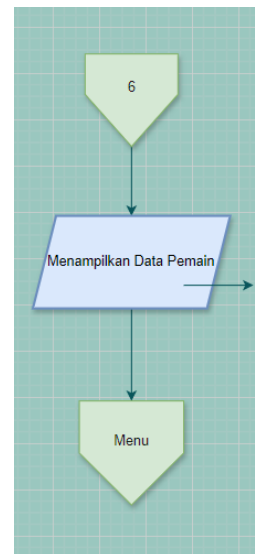
Gambar 1.3 Menu Login



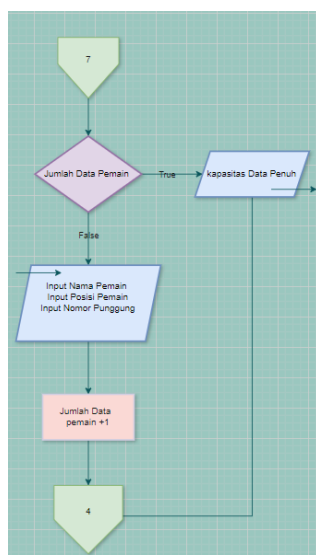
Gambar 1.4 Menu Admin



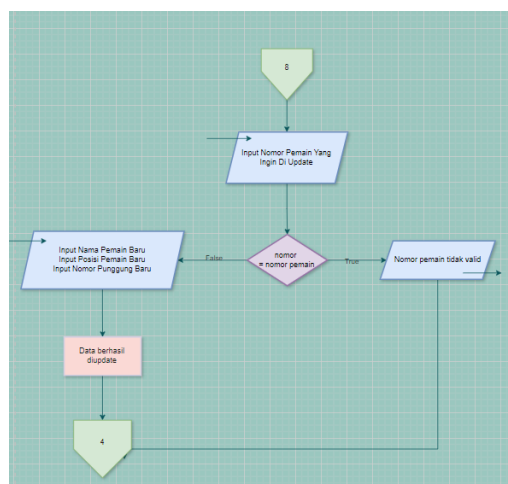
Gambar 1.5 Menu User



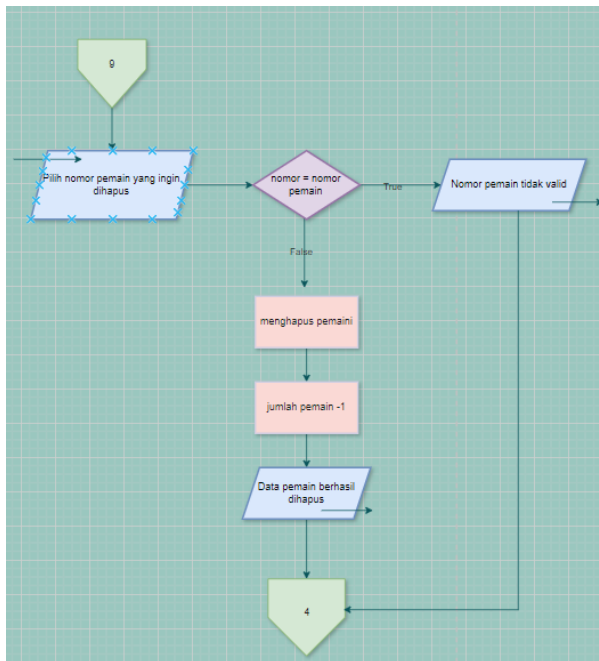
Gambar 1.6 Tampilan Data



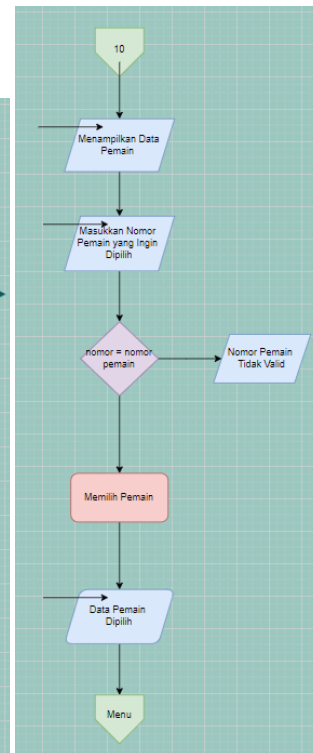
Gambar 1.7 Tambah Data



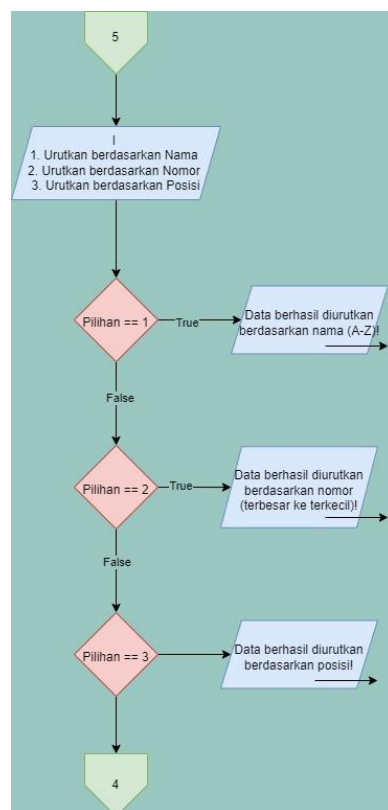
Gambar 1.8 Update Data



Gambar 1.9 Hapus Data



Gambar 1.10 Pilih Data Pemain



1.11 Gambar Menu Urutkan Data

## 2. Analisis Program

### 2.1 Deskripsi Singkat Program

Program ini adalah sebuah manajemen “Data Pemain Bola” yang ditulis dalam bahasa C++. Pada program ini terdapat menu sebagai user baru dan menu sebagai admin, di mana menu user sendiri dapat mengakses tampilan, pilihan, serta keluar. Lalu untuk menu pada admin dapat mengakses bermacam - macam pilihan yang dapat dipilih memungkinkan pengguna admin untuk melakukan berbagai operasi CRUD (Create, Read, Update, Delete) pada data pemain bola, seperti menambahkan, menampilkan, mengupdate, dan menghapus data pemain. Selain itu, program ini juga dilengkapi dengan sistem login untuk mengamankan akses ke aplikasi.

### 2.2 Penjelasan Alur & Algoritma

Program ini terdiri dari beberapa tahap utama, yang mencakup proses login, navigasi menu, serta operasi CRUD (Create, Read, Update, Delete) pada data produk. Berikut adalah alur kerja program :

#### 1. Login Pengguna

- ☐ Pengguna harus memasukkan username dan 3 digit terakhir NIM untuk masuk ke sistem.
- ☐ Jika pengguna salah memasukkan data login sebanyak tiga kali, program akan berhenti.
- ☐ Jika login berhasil, pengguna diarahkan ke menu utama.

#### 2. Menampilkan Menu Utama

- ☐ Program menampilkan opsi yang bisa dipilih oleh pengguna:
  1. Tampilkan Data Pemain
  2. Tambah Data Pemain
  3. Update Data Pemain
  4. Hapus Data Pemain
  5. Urutkan data
  6. Keluar
- ☐ Pengguna memasukkan nomor pilihan menu, yang akan menentukan langkah selanjutnya.

#### 3. CRUD Data Produk

- **Tampilan Data Pemain:** Program mencetak daftar data pemain dalam format tabel agar lebih rapi dan mudah dibaca.
- **Tambah Data Pemain:** Pengguna menginputkan nama pemain, posisi pemain, dan nomor punggung pemain, yang kemudian disimpan dalam array

**Update Data Pemain:** Pengguna memilih Pemain berdasarkan nomor urutnya, lalu memasukkan data baru untuk mengganti informasi yang ada.

- **Hapus Data Pemain :** Pengguna memilih nomor pemain yang ingin dihapus, lalu program menggeser data dalam array agar tetap terstruktur.
- **Urutkan Data :** Pengguna memilih nomor pilihan untuk memilih berdasarkan apa

#### 4. Keluar dari Program

- Jika pengguna memilih opsi Keluar, program akan menampilkan pesan perpisahan dan berhenti.

##### **Penjelasan Fungsi dalam Program:**

- **TampilkanData( )** Menampilkan daftar pilihan menu utama.
- **TambahData( )** Menambahkan data pemain baru ke dalam array jika kapasitas belum penuh.
- **TampilkanData( )** Menampilkan semua data pemain yang telah disimpan dalam format tabel.
- **UbahData( )** Mengubah informasi data pemain tertentu berdasarkan input pengguna.
- **HapusData( )** Menghapus data pemain yang dipilih dan menyesuaikan susunan array.
- **Urutkan data ( )** Mengurutkan data pemain dan menyusunnya ke dalam array
- **Main( )** Mengontrol seluruh proses eksekusi program dari login hingga interaksi dengan menu.

### 3. Source Code

#### A. Login

Pada Fitur ini menu yang digunakan untuk memvalidasi user yang ingin menggunakan aplikasi

##### Source Code :

```
void login() {
    string username, password;
    cout << "\nUsername: ";
    getline(cin, username);
    cout << "Password: ";
    getline(cin, password);

    for (int i = 0; i < jumlahUser; i++) {
        if (users[i].username == username && users[i].password == password) {
            if (users[i].isAdmin) {
                tampilkanMenuAdmin();
            } else {
                tampilkanMenuUser();
            }
            return;
        }
    }

    percobaanLogin++;
    cout << "\nLogin gagal! Percobaan " << percobaanLogin << "/3\n";
    if (percobaanLogin >= 3) {
        cout << "Anda telah 3x salah login. Program berhenti.\n";
        exit(0);
    }
}
```



## B. Registrasi

```
void registerUser() {
    if (jumlahUser >= MAX_USER) {
        cout << "\nKapasitas user penuh!\n";
        return;
    }

    User baru;
    cout << "\nUsername: ";
    getline(cin, baru.username);
    cout << "Password: ";
    getline(cin, baru.password);
    baru.isAdmin = false;

    users[jumlahUser++] = baru;
    cout << "\nRegistrasi berhasil!\n";
}

void tampilkanMenuAdmin() {
    while (true) {
        cout << "\n=== MENU ADMIN ===";
        cout << "\n1. Tampilkan Data Pemain";
        cout << "\n2. Tambah Pemain";
        cout << "\n3. Edit Pemain";
        cout << "\n4. Hapus Pemain";
        cout << "\n5. Cari Pemain";
        cout << "\n6. Urutkan Data Pemain";
        cout << "\n7. Logout";
        cout << "\nPilihan: ";

        int pilihan;
        cin >> pilihan;
        cin.ignore();

        switch (pilihan) {
            case 1: tampilkanDataPemain(); break;
            case 2: tambahPemain(); break;
            case 3: editPemain(); break;
            case 4: hapusPemain(); break;
            case 5: cariPemain(); break;
            case 6: {
                cout << "\n1. Urutkan berdasarkan Nama (A-Z)";
                cout << "\n2. Urutkan berdasarkan Nomor (Descending)";
                cout << "\n3. Urutkan berdasarkan Posisi";
                cout << "\nPilihan: ";
                int sortChoice;
                cin >> sortChoice;
                cin.ignore();
                switch (sortChoice) {
                    case 1: sortNamaAscending(); break;
                    case 2: sortNomorDescending(); break;
                    case 3: selectionSortPosisi(); break;
                    default: cout << "Pilihan tidak valid!\n";
                }
            }
        }
    }
}
```

```
        }  
        break;  
    }  
    case 7: return;  
    default: cout << "Pilihan tidak valid!\n";  
}  
}  
}
```

## B. Menu

Terdapat menu pada user yang dapat memilih fitur pada aplikasi

### Source Code

```
cout << "\n=== MENU USER ===";
cout << "\n1. Tampilkan Data Pemain";
cout << "\n2. Cari Pemain";
cout << "\n3. Urutkan Data Pemain";
cout << "\n4. Logout";
cout << "\nPilihan: ";
```

Pada menu Admin terdapat beberapa fitur pada aplikasi yang dapat di akses

### Source Code

```
cout << "\n=== MENU ADMIN ===";
cout << "\n1. Tampilkan Data Pemain";
cout << "\n2. Tambah Pemain";
cout << "\n3. Edit Pemain";
cout << "\n4. Hapus Pemain";
cout << "\n5. Cari Pemain";
cout << "\n6. Urutkan Data Pemain";
cout << "\n7. Logout";
cout << "\nPilihan: ";
```

## C. CRUD Manajemen Data Pemain Bola

Fitur CRUD dalam pemrograman ini mengacu pada empat operasi utama yaitu Create (membuat/menambahkan data baru), Read (Mengambil dan menampilkan data yang tersimpan), Update (Mengubah atau memperbarui data yang sudah ada.), Delet (Menghapus data dari Array.) yang digunakan untuk mengelola data produk dalam Array.

### Source Code ( Admin ):

```
int pilihan;
cin >> pilihan;
cin.ignore();

switch (pilihan) {
    case 1: tampilkanDataPemain(); break;
    case 2: tambahPemain(); break;
    case 3: editPemain(); break;
    case 4: hapusPemain(); break;
    case 5: cariPemain(); break;
    case 6: {
        cout << "\n1. Urutkan berdasarkan Nama (A-Z)";
        cout << "\n2. Urutkan berdasarkan Nomor (Descending)";
        cout << "\n3. Urutkan berdasarkan Posisi";
        cout << "\nPilihan: ";
        int sortChoice;
        cin >> sortChoice;
```

```
        cin.ignore();
        switch (sortChoice) {
            case 1: sortNamaAscending(); break;
            case 2: sortNomorDescending(); break;
            case 3: selectionSortPosisi(); break;
            default: cout << "Pilihan tidak valid!\n";
        }
        break;
    }
    case 7: return;
    default: cout << "Pilihan tidak valid!\n";
}
}
```

### Source Code ( User Baru )

```
cout << "\n=== MENU USER ===";
cout << "\n1. Tampilkan Data Pemain";
cout << "\n2. Cari Pemain";
cout << "\n3. Urutkan Data Pemain";
cout << "\n4. Logout";
cout << "\nPilihan: ";

int pilihan;
cin >> pilihan;
cin.ignore();

switch (pilihan) {
    case 1: tampilkanDataPemain(); break;
    case 2: cariPemain(); break;
    case 3: {
        cout << "\n1. Urutkan berdasarkan Nama (A-Z)";
        cout << "\n2. Urutkan berdasarkan Nomor (Descending)";
        cout << "\n3. Urutkan berdasarkan Posisi";
        cout << "\nPilihan: ";
        int sortChoice;
        cin >> sortChoice;
        cin.ignore();
        switch (sortChoice) {
            case 1: sortNamaAscending(); break;
            case 2: sortNomorDescending(); break;
            case 3: selectionSortPosisi(); break;
            default: cout << "Pilihan tidak valid!\n";
        }
        break;
    }
    case 4: return;
    default: cout << "Pilihan tidak valid!\n";
}
}
```

## 4. Hasil Output

### 1. Menu Admi

```
=====
|         SISTEM MANAJEMEN PEMAIN BOLA         |
=====
| 1. Login                                     |
| 2. Register                                |
| 3. Keluar                                   |
=====
Pilihan: 1

===== LOGIN =====
Username: nana
Password: 12

Username atau password salah!
Percobaan tersisa: 2

===== LOGIN =====
Username: nana
Password: 12

Username atau password salah!
Percobaan tersisa: 1

===== LOGIN =====
Username: nana
Password: 12

Username atau password salah!
Percobaan tersisa: 0

Anda telah 3x salah login. Program berhenti.
PS C:\Users\ASUS\OneDrive\PDF\PRAKTIKUM-APL\PostTest\PostTest-3>
```

Gambar 4.1.1 Kondisi Gagal Login

```
=====
|         SISTEM MANAJEMEN PEMAIN BOLA         |
=====
| 1. Login                                     |
| 2. Register                                |
| 3. Keluar                                   |
=====
Pilihan: 1

===== LOGIN =====
Username: Rusdiana
Password: 2409106021

Login berhasil!
```

Gambar 4.1.2 Tampilan Login Admin

```
=== MENU ADMIN ===
1. Tampilkan Data Pemain
2. Tambah Pemain
3. Edit Pemain
4. Hapus Pemain
5. Cari Pemain
6. Urutkan Data Pemain
7. Logout
Pilihan: 2

Nama: Messi
Posisi: Gelandang
Nomor Punggung: 12
```

Gambar 4.1.3 Menu Tambah Pemain

```
=== MENU ADMIN ===
1. Tampilkan Data Pemain
2. Tambah Pemain
3. Edit Pemain
4. Hapus Pemain
5. Cari Pemain
6. Urutkan Data Pemain
7. Logout
Pilihan: 1

=== DAFTAR PEMAIN ===
No  Nama      Posisi    Nomor
-----
1   Neymar    Bek       11
2   Messi     Gelandang 12
```

Gambar 4.1.4 Menu Tampilan Data

```
=== MENU ADMIN ===
1. Tampilkan Data Pemain
2. Tambah Pemain
3. Edit Pemain
4. Hapus Pemain
5. Cari Pemain
6. Urutkan Data Pemain
7. Logout
Pilihan: 3

=== DAFTAR PEMAIN ===
No  Nama      Posisi    Nomor
-----
1   Neymar    Bek       11
2   Messi     Gelandang 12

Pilih nomor pemain yang akan diedit: 2
Nama [Messi]: Mbapee
Posisi [Gelandang]: 10
Nomor Punggung [12]: 10

Data pemain berhasil diupdate!
```

Gambar 4.1.5 Menu Edit Pemain

```
=== MENU ADMIN ===
1. Tampilkan Data Pemain
2. Tambah Pemain
3. Edit Pemain
4. Hapus Pemain
5. Cari Pemain
6. Urutkan Data Pemain
7. Logout
Pilihan: 4

=== DAFTAR PEMAIN ===
No  Nama      Posisi    Nomor
-----
1   Neymar    Bek       11
2   Mbapee    Gelandang 10
```

Gambar 4.1.6 Menu Hapus Pemain

```

=== MENU ADMIN ===
1. Tampilkan Data Pemain
2. Tambah Pemain
3. Edit Pemain
4. Hapus Pemain
5. Cari Pemain
6. Urutkan Data Pemain
7. Logout
Pilihan: 5

Masukkan nama pemain: Neymar

=== HASIL PENCARIAN ===
No  Nama          Posisi    Nomor
-----
1   Neymar        Bek       11

```

Gambar 4.1.7 Menu Cari Pemain

```

=== MENU ADMIN ===
1. Tampilkan Data Pemain
2. Tambah Pemain
3. Edit Pemain
4. Hapus Pemain
5. Cari Pemain
6. Urutkan Data Pemain
7. Logout
Pilihan: 6

1. Urutkan berdasarkan Nama (A-Z)
2. Urutkan berdasarkan Nomor (Descending)
3. Urutkan berdasarkan Posisi
Pilihan: 1

Data berhasil diurutkan berdasarkan nama (A-Z)!

```

Gambar 4.1.8 Menu Urutkan Data Pemain

```

=== MENU ADMIN ===
1. Tampilkan Data Pemain
2. Tambah Pemain
3. Edit Pemain
4. Hapus Pemain
5. Cari Pemain
6. Urutkan Data Pemain
7. Logout
Pilihan: 7

=== SISTEM MANAJEMEN PEMAIN BOLA ===
1. Login
2. Register
3. Keluar
Pilihan: 3

PS C:\Users\ASUS\OneDrive\PDF\PRAKTIKUM-APL\PostTest\PostTest-6>

```

Gambar 4.1.8 Menu Logout

## 2. Menu User Baru

```

=====
|          SISTEM MANAJEMEN PEMAIN BOLA          |
=====
| 1. Login |
| 2. Register |
| 3. Keluar |
=====
Pilihan: 2

===== REGISTER =====
Username: nana
Password: 12

Registrasi berhasil!

```

Gambar 4.2.1 Registrasi Username Baru

```

=====
|          SISTEM MANAJEMEN PEMAIN BOLA          |
=====
| 1. Login |
| 2. Register |
| 3. Keluar |
=====
Pilihan: 1

===== LOGIN =====
Username: nana
Password: 0211

Login berhasil!

===== MENU USER =====
1. Lihat Data Pemain
2. Cari Pemain
3. Cari Pemain by Nomor (Pointer)
4. Logout
=====
Pilihan:

```

Gambar 4.2.2 Login Sebagai User Baru

```
=== MENU USER ===
1. Tampilkan Data Pemain
2. Cari Pemain
3. Urutkan Data Pemain
4. Logout
Pilihan: 1

=== DAFTAR PEMAIN ===
No  Nama                Posisi    Nomor
-----
1   Neymar              Bek       11
```

Gambar 4.2.3 Tampilan Lihat Data Pemain

```
=== MENU USER ===
1. Tampilkan Data Pemain
2. Cari Pemain
3. Urutkan Data Pemain
4. Logout
Pilihan: 2

Masukkan nama pemain: Neymar

=== HASIL PENCARIAN ===
No  Nama                Posisi    Nomor
-----
1   Neymar              Bek       11
```

Gambar 4.2.3 Menu Cari Pemain

```
=== MENU USER ===
1. Tampilkan Data Pemain
2. Cari Pemain
3. Urutkan Data Pemain
4. Logout
Pilihan: 3

1. Urutkan berdasarkan Nama (A-Z)
2. Urutkan berdasarkan Nomor (Descending)
3. Urutkan berdasarkan Posisi
Pilihan: 2

Data berhasil diurutkan berdasarkan nomor (terbesar ke terkecil)!
```

Gambar 4.2.4 Tampilan Urutkan Data Pemain

```
=== MENU USER ===
1. Tampilkan Data Pemain
2. Cari Pemain
3. Urutkan Data Pemain
4. Logout
Pilihan: 4

=== SISTEM MANAJEMEN PEMAIN BOLA ===
1. Login
2. Register
3. Keluar
Pilihan: 3

PS C:\Users\ASUS\OneDrive\PDF\PRAKTIKUM-APL\PostTest\PostTest-6>
```

Gambar 4.2.5 Menu Logout





## 4. Langkah-Langkah GIT

### 1. Git add

Git add adalah perintah Git yang digunakan untuk menambahkan perubahan pada file ke staging area sebelum dilakukan commit. Perintah ini memungkinkan Git mengetahui file mana saja yang akan dimasukkan dalam commit berikutnya

```
PS C:\Users\ASUS\OneDrive\PDF\PRAKTIKUM-APL> git add .  
PS C:\Users\ASUS\OneDrive\PDF\PRAKTIKUM-APL>
```

Gambar 5. 1 Git\_add

### 2. Git Commit

Git commit adalah perintah Git yang digunakan untuk menyimpan perubahan yang telah ditambahkan ke staging area ke dalam repository. Setiap commit akan memiliki hash unik, pesan commit, dan menyimpan snapshot dari perubahan yang dilakukan.

```
PS C:\Users\ASUS\OneDrive\PDF\PRAKTIKUM-APL> git commit -m "finish post test 6"  
[main 4e39da5] finish post test 6  
4 files changed, 625 insertions(+), 361 deletions(-)  
create mode 100644 PostTest/PostTest-6/2409106021_Rusdiana_PostTest-6.cpp  
create mode 100644 PostTest/PostTest-6/2409106021_Rusdiana_PostTest-6.exe  
PS C:\Users\ASUS\OneDrive\PDF\PRAKTIKUM-APL>
```

Gambar 5. 2 Git\_Commit

### 3. Git Push

Git push adalah perintah Git yang digunakan untuk mengirim (mengunggah) perubahan dari repository lokal ke repository remote (seperti GitHub, GitLab, atau Bitbucket). Perintah ini memastikan bahwa perubahan yang sudah dikomit di lokal tersedia di repository jarak jauh sehingga bisa diakses oleh orang lain atau untuk cadangan.

```
PS C:\Users\ASUS\OneDrive\PDF\PRAKTIKUM-APL> git push origin main --force  
Enumerating objects: 14, done.  
Counting objects: 100% (14/14), done.  
Delta compression using up to 8 threads  
Compressing objects: 100% (9/9), done.  
Writing objects: 100% (9/9), 272.50 KiB | 2.67 MiB/s, done.  
Total 9 (delta 7), reused 0 (delta 0), pack-reused 0 (from 0)  
remote: Resolving deltas: 100% (7/7), completed with 5 local objects.  
remote: This repository moved. Please use the new location:  
remote: https://github.com/RusdianaNana/praktikum-apl.git  
To https://github.com/RusdianaNana/Praktikum-Apl.git  
9827f93..4e39da5 main -> main  
PS C:\Users\ASUS\OneDrive\PDF\PRAKTIKUM-APL>
```

Gambar 5. 3 Git\_Push

