

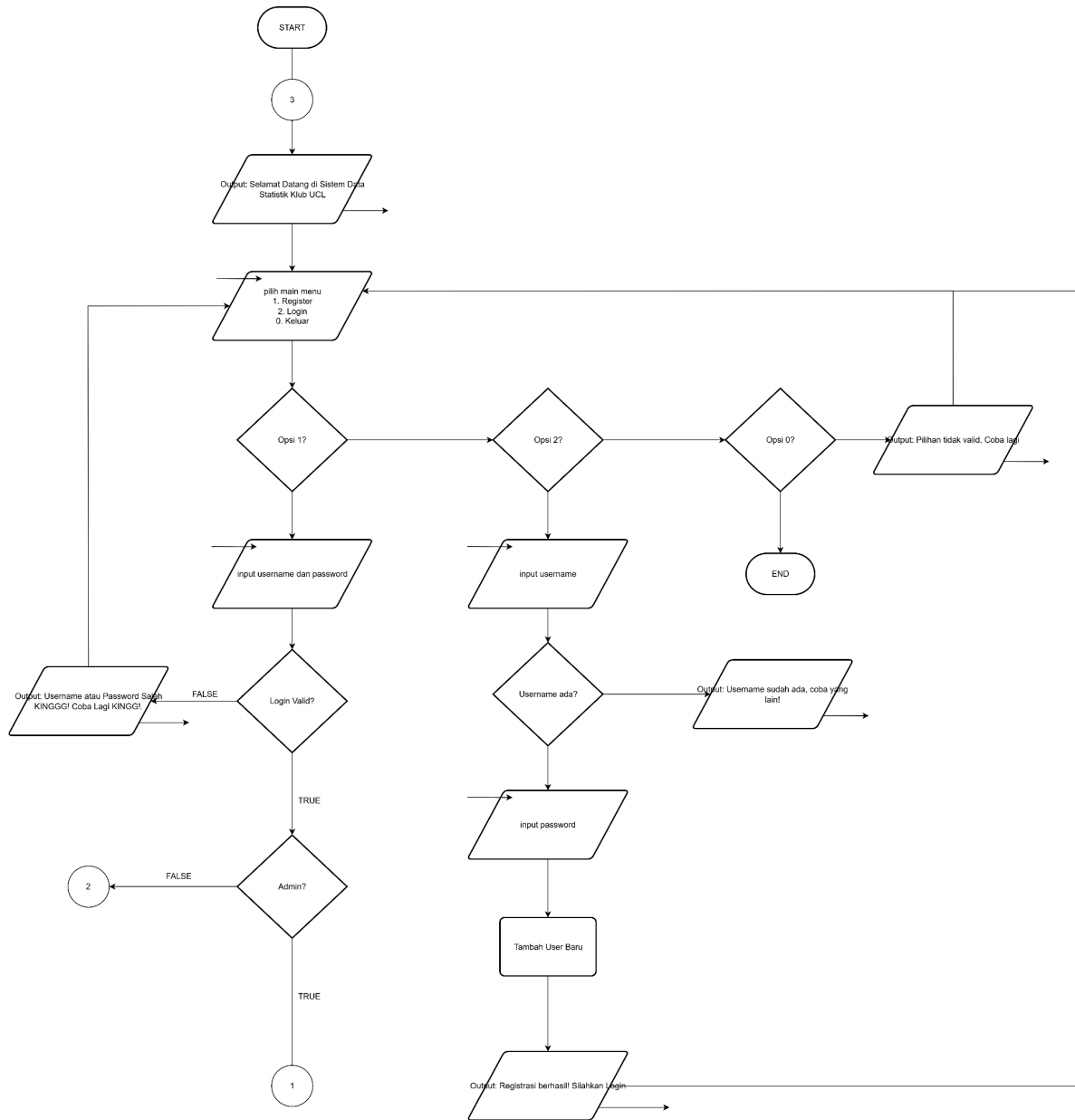
**LAPORAN PRAKTIKUM**  
**POSTTEST (7)**  
**ALGORITMA PEMROGRAMAN DASAR**

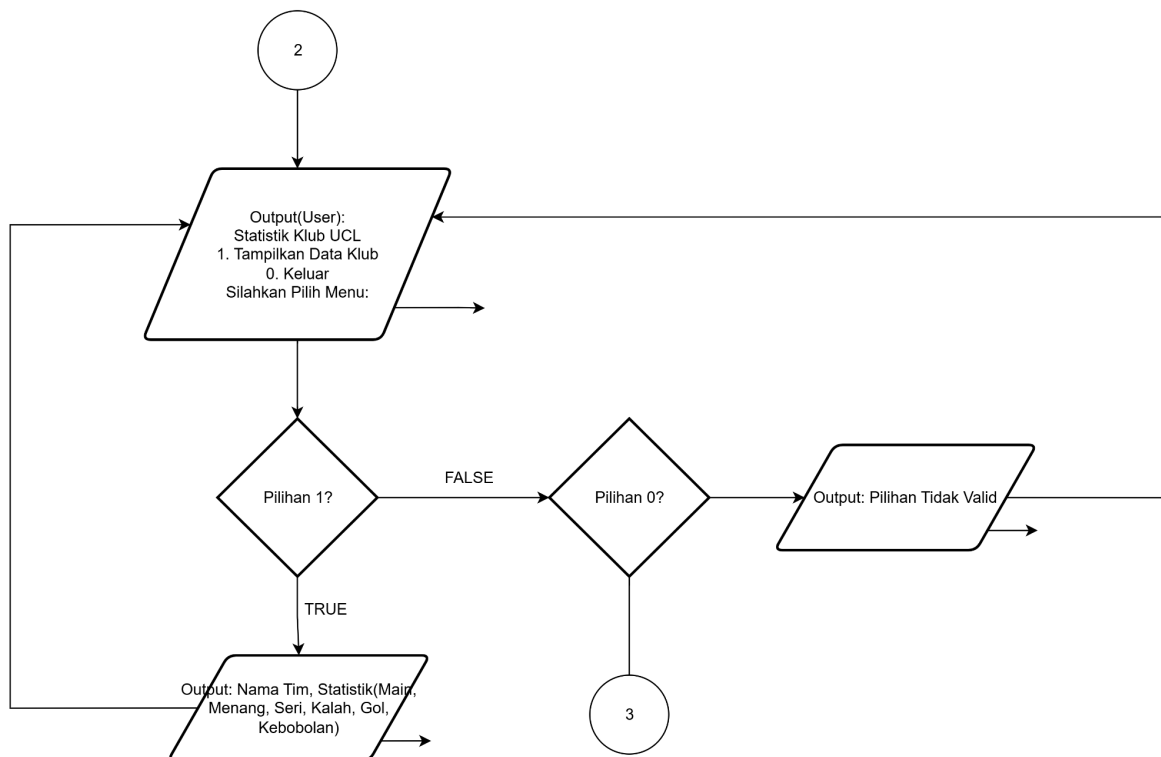
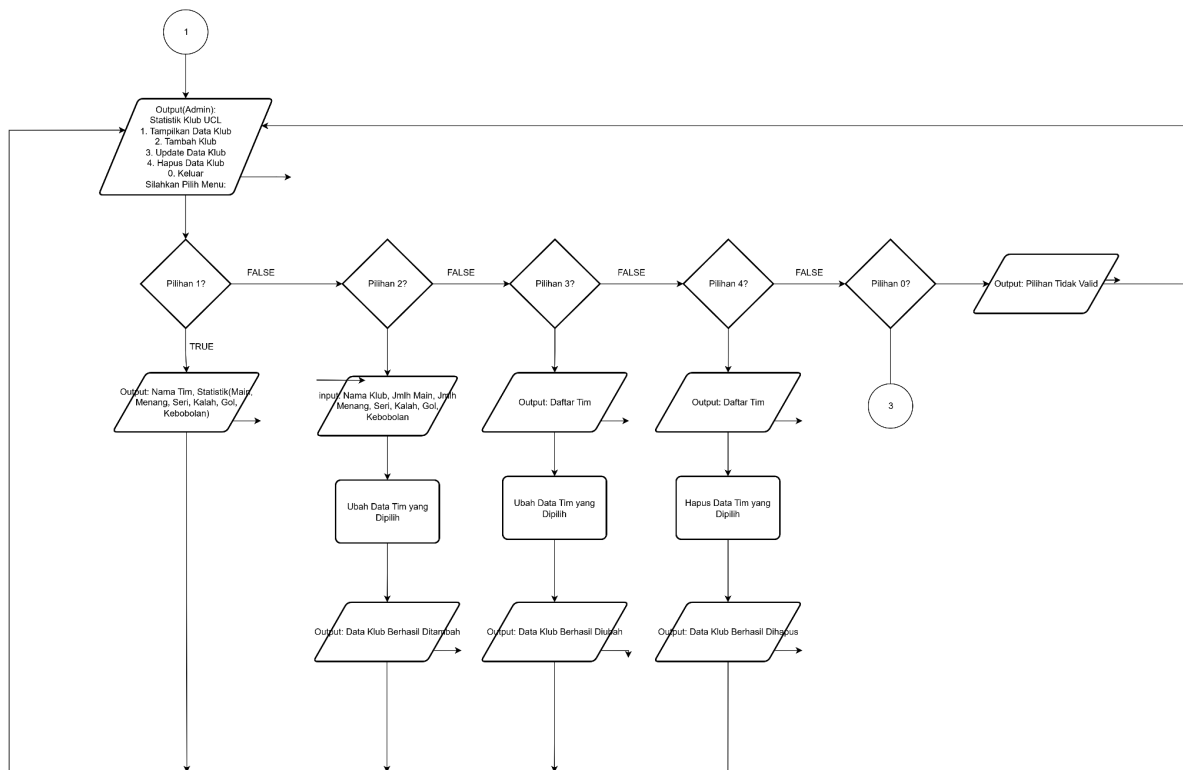


**Disusun oleh:**  
**Ulung Priskeaji Bintang Ramadhan (2509106051)**  
**Kelas (B1'25)**

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

# 1. Flowchart





Gambar <Flowchart draw.io>

Program ini adalah sistem manajemen statistik klub Liga Champions UEFA dengan fitur autentikasi pengguna (registrasi dan login) serta operasi CRUD (Create, Read, Update, Delete) pada data klub. Berikut adalah alur logika singkat dari input, proses, hingga output:

### 1. Input Awal (Menu Utama):

Program dimulai dengan menampilkan menu awal (`menuawal()`). Pengguna memasukkan pilihan (1: Registrasi, 2: Login, 0: Keluar).

### 2. Proses Input Awal:

- Jika pilih 1 (Registrasi): Pengguna memasukkan username dan password baru. Program memeriksa apakah username sudah ada di dictionary users. Jika belum, data ditambahkan; jika sudah, tampilkan pesan error.
- Jika pilih 2 (Login): Pengguna memasukkan username dan password. Program memverifikasi kecocokan dengan data di users. Jika valid, arahkan ke menu admin (jika username "admin") atau menu user; jika tidak, tampilkan pesan error.
- Jika pilih 0: Program berhenti.

### 3. Proses Menu Admin (jika login sebagai admin):

Menu admin menampilkan opsi (1: Tampilkan data, 2: Tambah data, 3: Update data, 4: Hapus data, 0: Keluar).

- Input: Pengguna memilih opsi dan memasukkan data sesuai (misal, untuk tambah: nama klub dan statistik numerik).
- Proses: Tampilkan: Iterasi dictionary klub\_ucl dan tampilkan semua data.
- Tambah: Tambahkan entri baru ke klub\_ucl dengan validasi input numerik.
- Update: Pilih klub dari daftar, masukkan data baru, ganti entri di klub\_ucl (hapus yang lama jika nama berubah).
- Hapus: Pilih klub dari daftar, hapus dari klub\_ucl.

Jika input invalid (misal bukan angka), tampilkan error dan ulangi.

### 4. Proses Menu User (jika login sebagai user biasa):

Menu user hanya menampilkan opsi (1: Tampilkan data, 0: Keluar).

- Input: Pilih opsi.
- Proses: Jika pilih 1, tampilkan semua data dari klub\_ucl (sama seperti admin).

#### 5. Output:

- Pesan konfirmasi (misal, "Registrasi berhasil", "Login berhasil", "Data berhasil ditambahkan").
- Tampilan data klub dalam format terstruktur (nama klub dan statistik).
- Pesan error untuk input salah atau operasi gagal.
- Program berakhir dengan pesan terima kasih jika keluar.

Alur ini berulang dalam loop menu hingga pengguna memilih keluar, dengan data disimpan sementara di dictionary global (users dan klub\_ucl). Program menggunakan input/output konsol untuk interaksi.

## 2. Deskripsi Singkat Program

Program ini bertujuan untuk mengelola data statistik klub Liga Champions UEFA secara terstruktur, dengan sistem autentikasi pengguna untuk mengontrol akses berdasarkan peran (admin atau user).

Fungsi/Manfaat Utama:

- Menyediakan sistem login dan register untuk pengguna baru, agar hanya pengguna yang berhak dapat mengakses data.
- Untuk user berfungsi melihat data klub.
- Sedangkan admin dapat melihat, menambah, mengubah, dan menghapus data klub secara dinamis

- Membantu dalam pengelolaan data statistik klub seperti jumlah pertandingan, kemenangan, hasil seri, kekalahan, serta jumlah gol yang dicetak dan kebobolan.

### 3. Source Code

Source Code:

```
# Data pengguna (username dan password)
users = {
    "admin": {"password": "admin123"},
    "user": {"password": "ggmu"},
}

# Data statistik klub UCL
klub_ucl = {
    "Paris Saint-Germain F.C.": {"main": 3, "menang": 3, "seri": 0, "kalah": 0, "gol_dicetak": 13, "gol_kebobolan": 3},
    "FC Bayern München": {"main": 3, "menang": 3, "seri": 0, "kalah": 0, "gol_dicetak": 12, "gol_kebobolan": 2},
    "Inter Milan": {"main": 3, "menang": 3, "seri": 0, "kalah": 0, "gol_dicetak": 9, "gol_kebobolan": 0},
    "Arsenal F.C.": {"main": 3, "menang": 3, "seri": 0, "kalah": 0, "gol_dicetak": 8, "gol_kebobolan": 0},
    "Real Madrid": {"main": 3, "menang": 3, "seri": 0, "kalah": 0, "gol_dicetak": 8, "gol_kebobolan": 1}
}

# Fungsi untuk registrasi pengguna baru
def register():
    global users
    username = input("Buat username: ")
    password = input("Buat password: ")
    if username in users:
        print("Username sudah ada, coba yang lain!")
    else:
```

```

        users[username] = {"password": password}
        print("Registrasi berhasil! Silahkan login.")

# Fungsi untuk login user dan admin
def login():
    global users
    username = input("Masukkan username: ")
    password = input("Masukkan password: ")

    if username in users and users[username]["password"] == password:
        print(f"Login berhasil! Selamat datang BROO {username}")
        if username == "admin":
            menuadmin()
        else:
            menuuser()
    else:
        print("Username atau Password Salah KINGGG! Coba Lagi KINGG!.")

# Fungsi menu awal
def menuawal():
    while True:
        print("\nPilih opsi:")
        print("1. Register")
        print("2. Login")
        print("0. Keluar")
        pilihan = input("Masukkan pilihan: ")

        if pilihan == "1":
            register()
        elif pilihan == "2":
            login()
        elif pilihan == "0":
            print("Terima kasih telah menggunakan program ini!")
            break
        else:
            print("Pilihan tidak valid, coba lagi!")

# Fungsi menu admin
def menuadmin():
    while True:
        print(

```

```

    """
    =====
    | Statistik Klub Liga Champions UEFA |
    =====
    | 1. Tampilkan Data Klub |
    | 2. Tambah Data Klub |
    | 3. Update Data Klub |
    | 4. Hapus Data Klub |
    | 0. Keluar |
    =====
    """
)
pilih = input("Silahkan Pilih Menu: ")
if pilih == "1":
    tampilkandata()
elif pilih == "2":
    klub = input("Nama Klub: ")
    try:
        main = int(input("Jumlah Main: "))
        menang = int(input("Jumlah Menang: "))
        seri = int(input("Jumlah Seri: "))
        kalah = int(input("Jumlah Kalah: "))
        gol_dicetak = int(input("Gol Dicetak: "))
        gol_kebobolan = int(input("Gol Kebobolan: "))
        tambahdata(klub, main, menang, seri, kalah, gol_dicetak,
gol_kebobolan)
    except ValueError:
        print("Input harus berupa angka KINGGGG, input yang benar.")
elif pilih == "3":
    print("Daftar Klub:")
    klub_list = list(klub_ucl.keys())
    for i in range(len(klub_list)):
        print(f"{i+1}. {klub_list[i]}")
    try:
        pilihan = int(input("Pilih nomor klub yang akan diubah: "))
        if 1 <= pilihan <= len(klub_list):
            klub_lama = klub_list[pilihan - 1]
            updatedata(klub_lama)
        else:
            print("Nomor tidak valid.")
    except ValueError:

```



```

        print("Input harus berupa angka.")
    elif pilih == "4":
        print("Daftar Klub:")
        klub_list = list(klub_ucl.keys())
        for i in range(len(klub_list)):
            print(f"{i+1}. {klub_list[i]}")
        try:
            pilihan = int(input("Pilih nomor klub yang ingin dihapus:
"))
            if 1 <= pilihan <= len(klub_list):
                klub_lama = klub_list[pilihan - 1]
                delete(klub_lama)
            else:
                print("Nomor tidak valid.")
        except ValueError:
            print("Input harus berupa angka.")
    elif pilih == "0":
        print("Kembali ke menu login.")
        break
    else:
        print("Pilihan tidak valid!")

# Fungsi menu user
def menuuser():
    while True:
        print(
            """
            =====
            | Statistik Klub Liga Champions UEFA |
            =====
            | 1. Tampilkan Data Klub |
            | 0. Keluar |
            =====
            """
        )
        pilih = input("Silahkan Pilih Menu: ")
        if pilih == "1":
            tampilkandata()
        elif pilih == "0":
            print("Kembali ke menu login.")
            break

```

```

        else:
            print("Pilihan tidak valid!")

# Fungsi tambah data klub
def tambahdata(klub, main, menang, seri, kalah, gol_dicetak, gol_kebobolan):
    klub_ucl[klub] = {
        "main": main,
        "menang": menang,
        "seri": seri,
        "kalah": kalah,
        "gol_dicetak": gol_dicetak,
        "gol_kebobolan": gol_kebobolan
    }
    print(f>Data klub {klub} berhasil ditambahkan.")

# Fungsi menampilkan data klub
def tampilkandata():
    for klub, stats in klub_ucl.items():
        print(f"\nKlub: {klub}")
        print(f>Main: {stats['main']}")
        print(f"Menang: {stats['menang']}")
        print(f"Seri: {stats['seri']}")
        print(f>Kalah: {stats['kalah']}")
        print(f>Gol Dicetak: {stats['gol_dicetak']}")
        print(f>Gol Kebobolan: {stats['gol_kebobolan']}")

# Fungsi update data klub
def updatedata(klub_lama):
    if klub_lama in klub_ucl:
        klub_baru = input("Nama Klub baru: ")
        try:
            main = int(input("Jumlah Main baru: "))
            menang = int(input("Jumlah Menang baru: "))
            seri = int(input("Jumlah Seri baru: "))
            kalah = int(input("Jumlah Kalah baru: "))
            gol_dicetak = int(input("Gol Dicetak baru: "))
            gol_kebobolan = int(input("Gol Kebobolan baru: "))
        except ValueError:
            print("Input harus berupa angka, gagal mengubah data.")
        return

```

```

        klub_ucl[klub_baru] = {
            "main": main,
            "menang": menang,
            "seri": seri,
            "kalah": kalah,
            "gol_dicetak": gol_dicetak,
            "gol_kebobolan": gol_kebobolan
        }
        if klub_baru != klub_lama:
            del klub_ucl[klub_lama]
            print(f>Data klub {klub_baru} berhasil diubah.")
        else:
            print("Klub tidak ditemukan.")

# Fungsi delete data klub
def delete(klub_lama):
    if klub_lama in klub_ucl:
        del klub_ucl[klub_lama]
        print(f>Klub {klub_lama} berhasil dihapus.")
    else:
        print("Klub tidak ditemukan.")

menuawal()

```

## A. Fitur Hapus Data

### Source Code:

```

# Fungsi delete data klub

def delete(klub_lama):

    if klub_lama in klub_ucl:

```

```
del klub_ucl[klub_lama]

print(f"Klub {klub_lama} berhasil dihapus.")

else:

    print("Klub tidak ditemukan.")
```

#### 4. Hasil Output

```
Pilih opsi:
1. Register
2. Login
0. Keluar
Masukkan pilihan: 1
Buat username: tes
Buat password: 123
Registrasi berhasil! Silahkan login.

Pilih opsi:
1. Register
2. Login
0. Keluar
Masukkan pilihan: 2
Masukkan username: tes
Masukkan password: 123

Login berhasil! Selamat datang BROO tes
```

Gambar <Hasil Output Register dan Login>

```
Pilih opsi:  
1. Register  
2. Login  
0. Keluar  
Masukkan pilihan: 0  
Terima kasih telah menggunakan program ini!  
PS C:\praktikum-apd> █
```

Gambar <Hasil Output Keluar>

```

=====
|  Statistik Klub Liga Champions UEFA  |
=====
|      1. Tampilkan Data Klub          |
|      0. Keluar                      |
=====

Silahkan Pilih Menu: 1

Klub: Paris Saint-Germain F.C.
Main: 3
Menang: 3
Seri: 0
Kalah: 0
Gol Dicetak: 13
Gol Kebobolan: 3

Klub: FC Bayern München
Main: 3
Menang: 3
Seri: 0
Kalah: 0
Gol Dicetak: 12
Gol Kebobolan: 2

Klub: Inter Milan
Main: 3
Menang: 3
Seri: 0
Kalah: 0
Gol Dicetak: 9
Gol Kebobolan: 0

```

Gambar <Hasil Output Menu 1(Lihat Data)>

```

=====
Menu Statistik Klub UCL
=====
1. Lihat Statistik Tim Klub
2. Tambah Data Klub
3. Update Data Klub
4. Hapus Data Klub
5. Keluar
Silahkan Pilih Menu (1-5): 1
=====
Statistik Tim Klub
=====

1. Klub: FC Bayern München
Main: 2,
Menang: 2,
Seri: 0,
Kalah: 0,
Gol Dicetak: 8,
Gol Kebobolan: 2
-----

2. Klub: Real Madrid
Main: 2,
Menang: 2,
Seri: 0,
Kalah: 0,
Gol Dicetak: 7,
Gol Kebobolan: 1
-----

```

Gambar <Hasil Output Menu 1(Lihat Data)>

```
=====
|  Statistik Klub Liga Champions UEFA  |
=====
|      1. Tampilkan Data Klub          |
|      0. Keluar                      |
=====

Silahkan Pilih Menu: 0
Kembali ke menu login.

Pilih opsi:
1. Register
2. Login
0. Keluar
```

Gambar <Hasil Output Menu 0(Kembali ke Menu Login)>



```
Pilih opsi:
1. Register
2. Login
0. Keluar
Masukkan pilihan: 2
Masukkan username: admin
Masukkan password: admin123

Login berhasil! Selamat datang BROO admin
```

```
=====
| Statistik Klub Liga Champions UEFA |
=====
| 1. Tampilkan Data Klub |
| 2. Tambah Data Klub |
| 3. Update Data Klub |
| 4. Hapus Data Klub |
| 0. Keluar |
=====
```

```
Silahkan Pilih Menu: █
```

Gambar <Hasil Output Menu Admin>

```
=====
Menu Statistik Klub UCL
=====
1. Lihat Statistik Tim Klub
2. Tambah Data Klub
3. Update Data Klub
4. Hapus Data Klub
5. Keluar
Silahkan Pilih Menu (1-5): 4
=====
Daftar Klub yang Bisa Dihapus:
1. FC Bayern München
2. Real Madrid
3. Paris Saint-Germain F.C.
4. Inter Milan
5. Arsenal F.C.
6. AFC Ajax
=====
Pilih Nomor Klub yang Ingin Dihapus: 6
Data klub Berhasil Dihapus KINGGG!!.
```

Gambar <Hasil Output Menu 2(Tambah Data)>

```
=====
| Statistik Klub Liga Champions UEFA |
=====
| 1. Tampilkan Data Klub |
| 2. Tambah Data Klub |
| 3. Update Data Klub |
| 4. Hapus Data Klub |
| 0. Keluar |
=====
```

Silahkan Pilih Menu: 3

Daftar Klub:

1. Paris Saint-Germain F.C.
2. FC Bayern München
3. Inter Milan
4. Arsenal F.C.
5. Real Madrid
6. Manchester City

Pilih nomor klub yang akan diubah: 1

Jumlah Main baru: 4

Jumlah Menang baru: 4

Jumlah Seri baru: 0

Jumlah Kalah baru: 0

Gol Dicitak baru: 18

Gol Kebobolan baru: 5

Data klub Paris Saint-Germain F.C. berhasil diubah.

Gambar <Hasil Output Menu 3(Update Data)>

```

=====
|  Statistik Klub Liga Champions UEFA  |
=====
|      1. Tampilkan Data Klub          |
|      2. Tambah Data Klub             |
|      3. Update Data Klub             |
|      4. Hapus Data Klub              |
|      0. Keluar                       |
=====

```

Silahkan Pilih Menu: 4

Daftar Klub:

1. Paris Saint-Germain F.C.
2. FC Bayern München
3. Inter Milan
4. Arsenal F.C.
5. Real Madrid
6. Manchester City

Pilih nomor klub yang ingin dihapus: 6

Klub Manchester City berhasil dihapus.

Gambar <Hasil Output Menu 4(Hapus Data)>

## 5. Langkah-langkah GIT

```
PS C:\praktikum-apd> git add .
PS C:\praktikum-apd> gti commit "apd7"
gti : The term 'gti' is not recognized as the name of a cmdlet, function, script file, or operable program. Check the
spelling of the name, or if a path was included, verify that the path is correct and try again.
At line:1 char:1
+ gti commit "apd7"
+ ~~~~
+ CategoryInfo          : ObjectNotFound: (gti:String) [], CommandNotFoundException
+ FullyQualifiedErrorId : CommandNotFoundException

PS C:\praktikum-apd> git commit -m "apd7"
[main 647e371] apd7
4 files changed, 736 insertions(+)
rename kelas/{tes list&tuple => tes list&tuple.py} (100%)
rename kelas/{tes set&dictionary => tes set&dictionary.py} (100%)
create mode 100644 post-test/post-test-apd-7/2509106051-UlungPriskeajiBintangRamadhan-PT-7.drawio
create mode 100644 post-test/post-test-apd-7/2509106051-UlungPriskeajiBintangRamadhan-PT-7.py
PS C:\praktikum-apd> git push
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 28 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (7/7), 5.90 KiB | 1.97 MiB/s, done.
Total 7 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
remote: This repository moved. Please use the new location:
remote:   https://github.com/UlungPriskeaji/praktikum-apd.git
To https://github.com/UlungPriskeaji/praktikumapd.git
   5ccec6c..647e371  main -> main
PS C:\praktikum-apd>
```

Gambar <Proses GIT>

### 5.1 GIT Add

GIT add merupakan perintah yang digunakan untuk menambahkan perubahan pada folder/file ke area staging sebelum melakukan commit. Area staging adalah area perantara yang menyimpan informasi tentang perubahan yang akan dimasukkan dalam commit berikutnya.

### 5.2 GIT Commit

GIT Commit adalah instruksi yang digunakan untuk menyimpan perubahan yang telah dimasukkan ke dalam staging area ke dalam repository Git lokal secara tetap. Setiap commit menyimpan cuplikan proyek pada saat itu dan disertai pesan deskriptif yang menjelaskan perubahan yang telah dilakukan.

### 5.3 GIT Push

GIT Push digunakan untuk mengirim commit dari repository lokal ke repository remote (misalnya GitHub).