

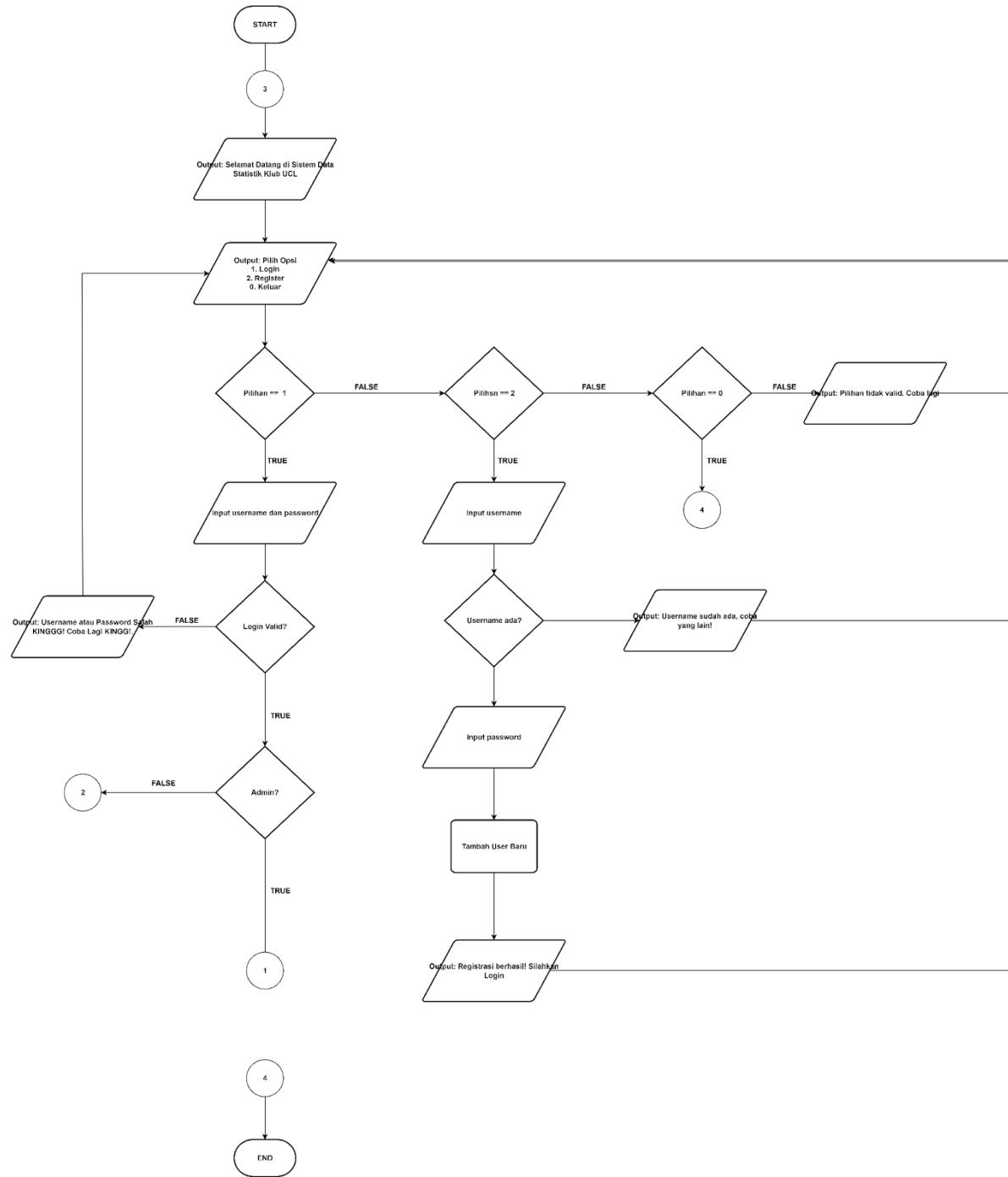
**LAPORAN PRAKTIKUM
POSTTEST (8)
ALGORITMA PEMROGRAMAN DASAR**

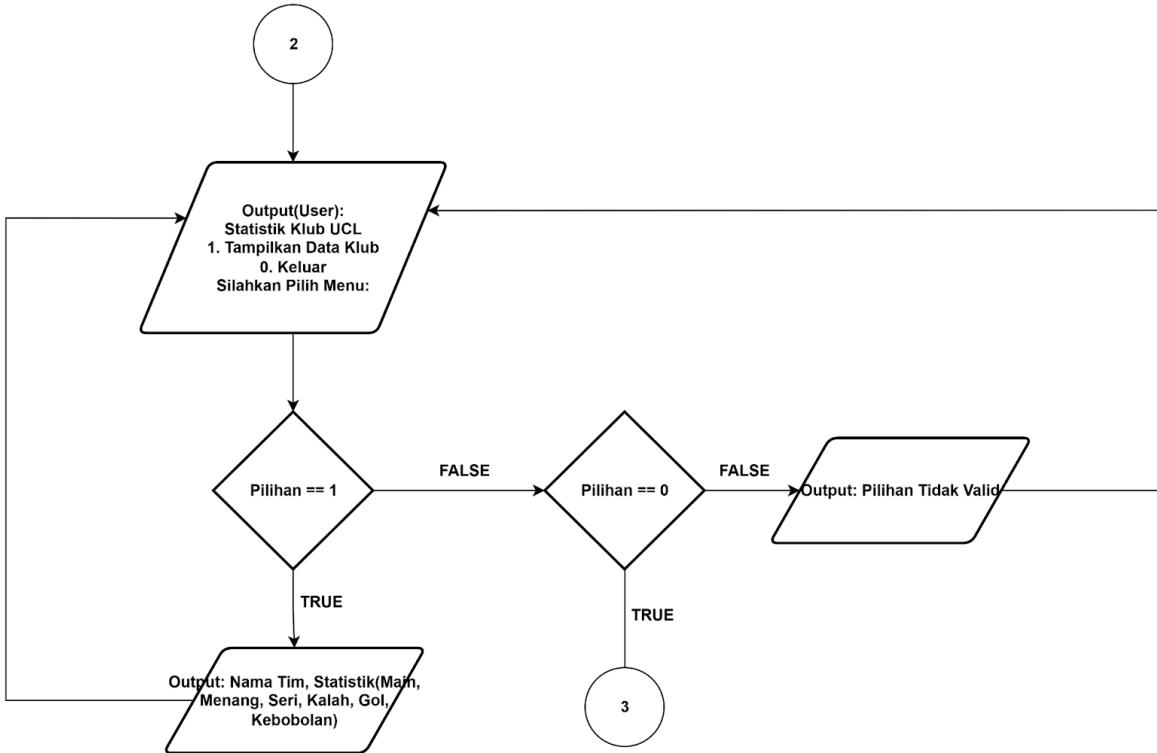
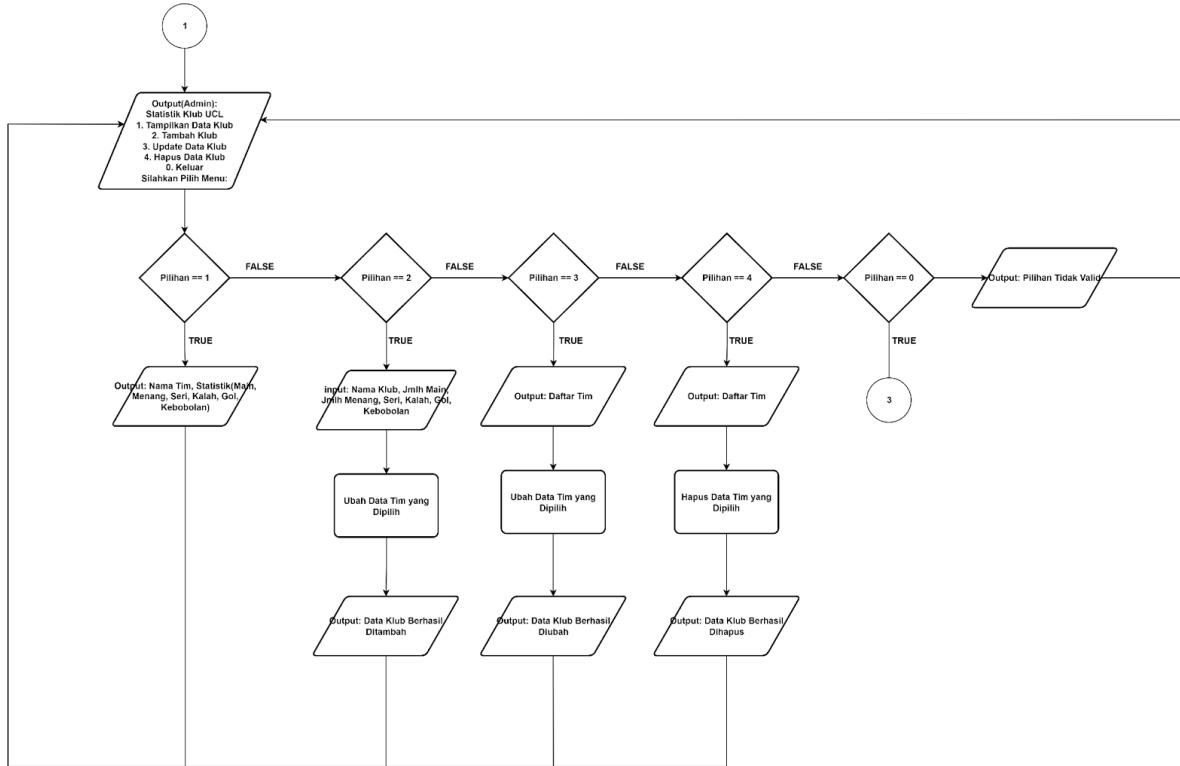


Disusun oleh:
Ulung Priskeaji Bintang Ramadhan (2509106051)
Kelas (B1'25)

**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 adalah **aplikasi berbasis Python** yang mensimulasikan sistem manajemen data klub sepak bola di Liga Champions UEFA. Program memiliki **fitur login, registrasi, dan manajemen data klub**, dengan peran pengguna yang dibedakan antara **admin** dan **user biasa**.

Program terdiri dari tiga modul utama:

1. **main.py**

Menyediakan menu utama untuk pengguna:

- Login
- Register
- Keluar dari program

Setelah login berhasil, pengguna diarahkan ke menu sesuai perannya (admin atau user).

2. **user.py**

Mengatur proses autentikasi dan menu setelah login:

- **Login:** memeriksa username & password dari data pengguna.
- **Register:** menambahkan akun baru ke data.
- **Menu Admin:** bisa melihat, menambah, mengubah, dan menghapus data klub.
- **Menu User:** hanya bisa melihat data klub.

3. **data.py**

► Menyimpan dan mengelola data:

- **users** → data login pengguna (admin dan user).
- **klub_uc1** → data statistik klub (main, menang, seri, kalah, gol dicetak, gol kebobolan).
- Fungsi CRUD (Create, Read, Update, Delete) untuk data klub.
- Menggunakan **PrettyTable** agar tampilan tabel rapi di terminal.

3. Source Code

A. main.py

Source Code:

```
from prettytable import PrettyTable

from auth import login, register

def menuawal():

    while True:

        table = PrettyTable()

        table.field_names = ["No", "Program Liga Champions UEFA"]

        table.align = "l"

        table.add_row(["1", "Login"])

        table.add_row(["2", "Register"])

        table.add_row(["0", "Keluar"])

        print()

        print(table)

pilihan = input("\n Silahkan Pilih Menu: ")

if pilihan == "1":

    login()

elif pilihan == "2":

    register()

elif pilihan == "0":
```

```
        print("Terima kasih telah menggunakan program ini!")

        break

    else:

        print("Pilihan tidak valid, coba lagi!")

if __name__ == "__main__":
    menuawal()
```

B. auth.py

Source Code:

```
import data

from prettytable import PrettyTable


def login():

    username = input("Masukkan username: ")

    password = input("Masukkan password: ")

    if username in data.users and data.users[username]["password"] == password:

        print()

        print(f"Login berhasil! Selamat datang BROO {username}")

    if username == "admin":
```

```

menuadmin()

else:

    menuuser()

else:

    print("Username atau Password Salah KINGGG! Coba Lagi KINGG!.")
```



```

def register():

    username = input("Buat username: ")

    password = input("Buat password: ")

    if username in data.users:

        print("Username sudah ada, coba yang lain!")

    else:

        data.users[username] = {"password": password}

        print("Registrasi berhasil! Silahkan login.")
```



```

def menuadmin():

    while True:

        table = PrettyTable()

        table.field_names = ["No", "Statistik Klub Liga Champions UEFA"]

        table.align = "l"

        table.add_row(["1", "Tampilkan Data Klub"])

        table.add_row(["2", "Tambah Data Klub"])

        table.add_row(["3", "Update Data Klub"])
```

```



```

```

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]

        data.updatedata(klub_lama)

    else:

        print("Nomor tidak valid.")

except ValueError:

    print("Input harus berupa angka.")

elif pilih == "4":

    print("Daftar Klub:")

    klub_list = list(data.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]

            data.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!")

def menuuser():
    while True:

        table = PrettyTable()

        table.field_names = ["No", "Statistik Klub Liga Champions UEFA"]
        table.align = "l"

        table.add_row(["1", "Tampilkan Data Klub"])
        table.add_row(["0", "Keluar"])

        print()
        print(table)

    pilih = input("\n Silahkan Pilih Menu: ")

    if pilih == "1":
        data.tampilkandata()

    elif pilih == "0":
        print("Kembali ke menu login.")

```

```
        break

    else:

        print("Pilihan tidak valid!")
```

C. data.py

Source Code:

```
from prettytable import PrettyTable

users = {

    "admin": {"password": "admin123"},

    "user": {"password": "ggmu"},

}

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}
```

```

}

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.")


def tampilkandata():

    table = PrettyTable()

        table.field_names = ["Klub", "Main", "Menang", "Seri", "Kalah", "Gol Dicetak", "Kebobolan"]

    for klub, stats in klub_ucl.items():

        table.add_row([
            klub,
            stats['main'],
            stats['menang'],
            stats['seri'],
            stats['kalah'],

```

```

        stats['gol_dicetak'],
        stats['gol_kebobolan']
    )

print(table)

def updatedata(klub_lama):

    if klub_lama in klub_ucl:
        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_lama] = {

        "main": main,
        "menang": menang,
        "seri": seri,
        "kalah": kalah,
        "gol_dicetak": gol_dicetak,
        "gol_kebobolan": gol_kebobolan
    }
}

```


4. Hasil Output

```
+-----+  
| No | Program Liga Champions UEFA |  
+-----+  
| 1 | Login |  
| 2 | Register |  
| 0 | Keluar |  
+-----+  
  
Silahkan Pilih Menu: 2  
Buat username: tes1  
Buat password: tes123  
Registrasi berhasil! Silahkan login.  
  
+-----+  
| No | Program Liga Champions UEFA |  
+-----+  
| 1 | Login |  
| 2 | Register |  
| 0 | Keluar |  
+-----+  
  
Silahkan Pilih Menu: 1  
Masukkan username: tes1  
Masukkan password: tes123  
  
Login berhasil! Selamat datang BROO tes1  
  
+-----+  
| No | Statistik Klub Liga Champions UEFA |  
+-----+  
| 1 | Tampilkan Data Klub |  
| 0 | Keluar |  
+-----+  
  
Silahkan Pilih Menu: █
```

Gambar <Hasil Output Register dan Login>

No	Program Liga Champions UEFA
1	Login
2	Register
0	Keluar

Silahkan Pilih Menu: 0

Terima kasih telah menggunakan program ini!

○ PS C:\praktikum-apd> █

Gambar <Hasil Output Keluar>

No	Statistik Klub Liga Champions UEFA
1	Tampilkan Data Klub
0	Keluar

Silahkan Pilih Menu: 1

Klub	Main	Menang	Seri	Kalah	Gol Dicetak	Kebobolan
Paris Saint-Germain F.C.	3	3	0	0	13	3
FC Bayern München	3	3	0	0	12	2
Inter Milan	3	3	0	0	9	0
Arsenal F.C.	3	3	0	0	8	0
Real Madrid	3	3	0	0	8	1

Gambar <Hasil Output Menu 1(Lihat Data)>

No	Statistik Klub Liga Champions UEFA
1	Tampilkan Data Klub
0	Keluar

Silahkan Pilih Menu: 0
Kembali ke menu login.

No	Program Liga Champions UEFA
1	Login
2	Register
0	Keluar

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

No	Program Liga Champions UEFA
1	Login
2	Register
0	Keluar

Silahkan Pilih Menu: 1

Masukkan username: admin

Masukkan password: admin123

Login berhasil! Selamat datang BROO admin

No	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>

No	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: 2

Nama Klub: Newcastle United

Jumlah Main: 3

Jumlah Menang: 2

Jumlah Seri: 0

Jumlah Kalah: 1

Gol Dicetak: 8

Gol Kebobolan: 2

Data klub Newcastle United berhasil ditambahkan.

Gambar <Hasil Output Menu 2(Tambah Data)>

No	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. Newcastle United

Pilih nomor klub yang akan diubah: 1

Jumlah Main baru: 4

Jumlah Menang baru: 4

Jumlah Seri baru: 0

Jumlah Kalah baru: 0

Gol Dicetak baru: 17

Gol Kebobolan baru: 5

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

Gambar <Hasil Output Menu 3(Update Data)>

No	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. Newcastle United

Pilih nomor klub yang ingin dihapus: 6

Klub Newcastle United berhasil dihapus.

Gambar <Hasil Output Menu 4(Hapus Data)>

5. Langkah-langkah GIT

```
● PS C:\praktikum-apd> git add .
warning: in the working copy of 'post-test/post-test-apd-8/2509106051-UlungPriskeajiBintangRamadhan-PT-8.drawio', LF will be replaced by CRLF the next time Git touches it
● PS C:\praktikum-apd> git commit -m "apd8"
[main c5578c4] apd8
 6 files changed, 741 insertions(+)
create mode 100644 post-test/post-test-apd-8/2509106051-UlungPriskeajiBintangRamadhan-PT-8.drawio
create mode 100644 post-test/post-test-apd-8/2509106051-UlungPriskeajiBintangRamadhan-PT-8/auth.py
create mode 100644 post-test/post-test-apd-8/2509106051-UlungPriskeajiBintangRamadhan-PT-8/data.py
create mode 100644 post-test/post-test-apd-8/2509106051-UlungPriskeajiBintangRamadhan-PT-8/main.py
create mode 100644 post-test/post-test-apd-8/_pycache__auth.cpython-313.pyc
create mode 100644 post-test/post-test-apd-8/_pycache__data.cpython-313.pyc
● PS C:\praktikum-apd> git push
Enumerating objects: 14, done.
Counting objects: 100% (14/14), done.
Delta compression using up to 28 threads
Compressing objects: 100% (12/12), done.
Writing objects: 100% (12/12), 10.16 KiB | 2.03 MiB/s, done.
Total 12 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
remote: This repository moved. Please use the new location:
remote:   https://github.com/UlungPriskeaji/praktikum-apd.git
To https://github.com/UlungPriskeaji/praktikumapd.git
 f9fb01a..c5578c4 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).