

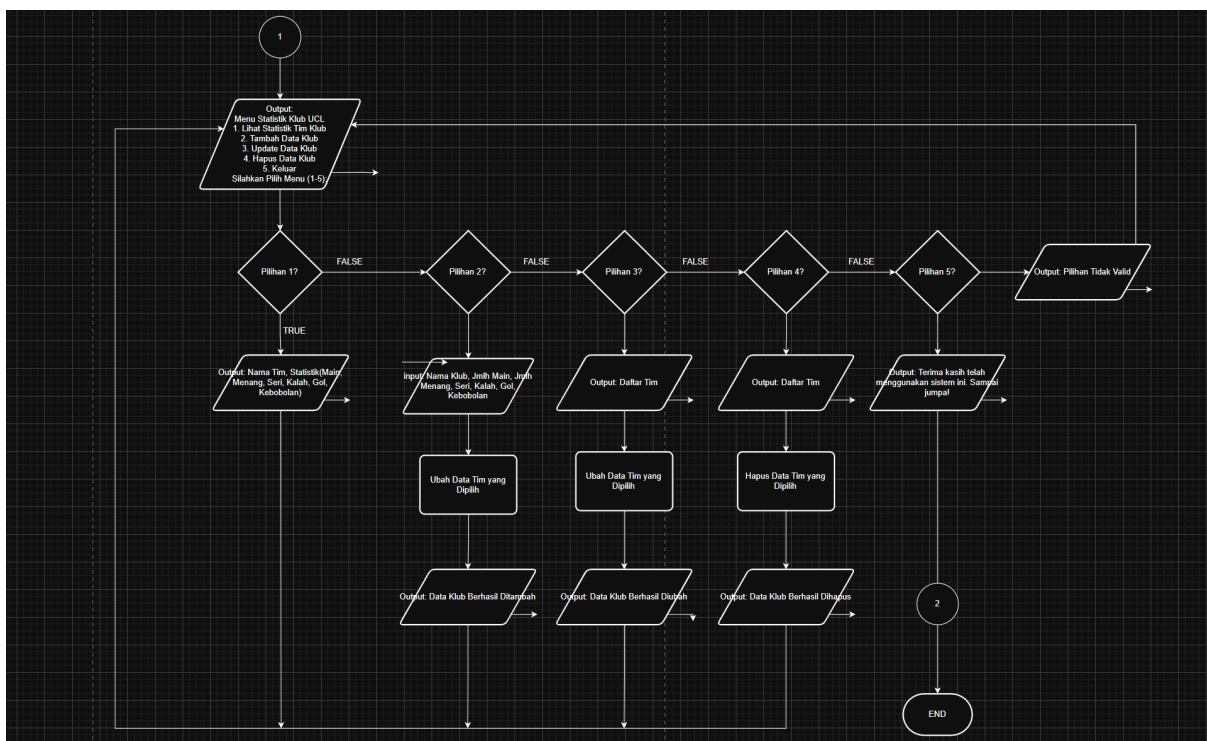
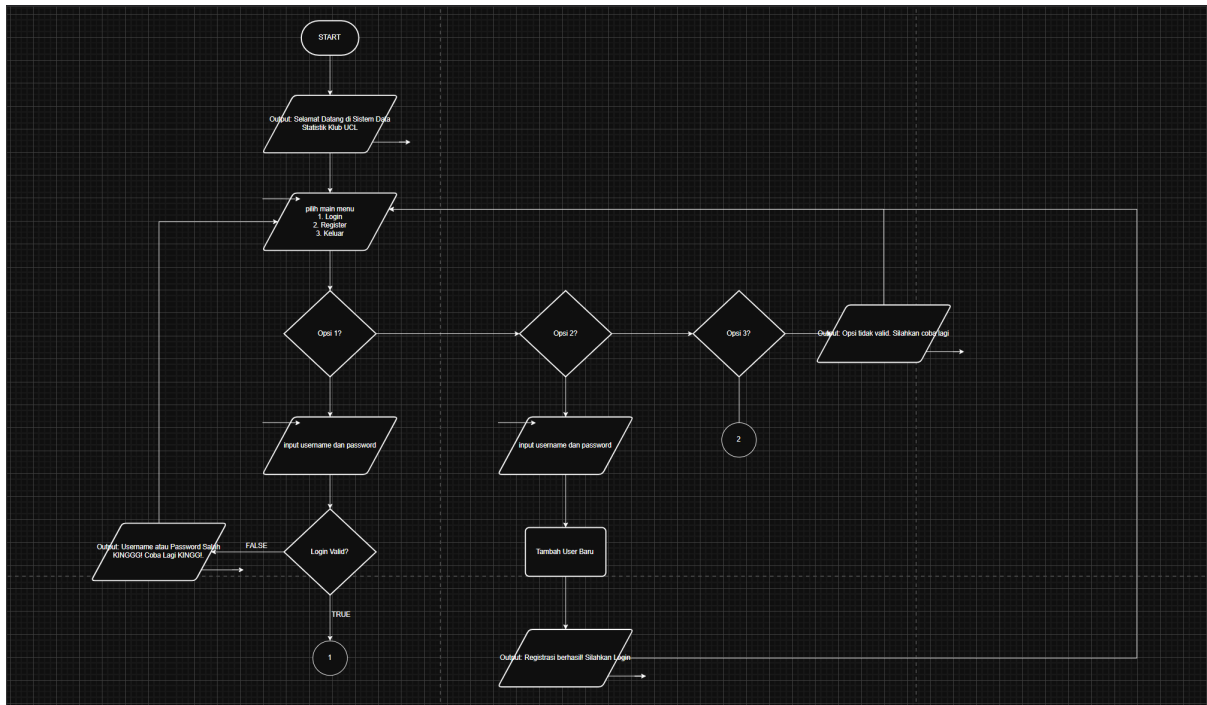
LAPORAN PRAKTIKUM
POSTTEST (6)
ALGORITMA PEMROGRAMAN DASAR



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

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

1. Flowchart



Gambar <Flowchart draw.io>

Dalam program ini, langkah pertama yang dilakukan adalah login dengan memasukkan username dan password. Jika username dan password yang dimasukkan sesuai dengan data yang benar (ggmu dan treble), maka program

akan menampilkan pesan “Login berhasil!” dan melanjutkan ke tahap berikutnya. Namun, jika salah, program akan menampilkan pesan “Username atau Password Salah KINGGG! Coba Lagi KINGG!” dan meminta pengguna untuk menginput ulang hingga data login benar. Selain itu, pengguna juga dapat memilih opsi register untuk membuat akun baru: jika memilih register, program akan meminta input username dan password baru, menambahkannya ke dalam sistem, lalu menampilkan pesan “Registrasi berhasil! Silahkan login.” dan kembali ke menu awal untuk login.

Setelah berhasil login, pengguna akan masuk ke menu utama sistem data statistik klub UCL. Pada tahap ini, pengguna dapat memilih beberapa opsi melalui input angka (1–5), yaitu:

1. Lihat Statistik Tim Klub: Menampilkan daftar seluruh klub beserta datanya, seperti jumlah main, menang, seri, kalah, gol dicetak, dan gol kebobolan.
2. Tambah Data Klub: Meminta input data klub baru (nama, jumlah main, menang, seri, kalah, gol dicetak, dan gol kebobolan), lalu menambahkannya ke dalam daftar klub_ucl.
3. Update Data Klub: Menampilkan daftar klub yang ada, lalu pengguna dapat memilih nomor klub yang ingin diubah dan memasukkan data baru untuk menggantikan data lama.
4. Hapus Data Klub: Meminta pengguna memilih nomor klub yang ingin dihapus dari daftar. Data klub tersebut kemudian akan dihapus permanen dari sistem.
5. Keluar: Mengakhiri program dengan menampilkan pesan penutup “Terima kasih telah menggunakan sistem ini. Sampai jumpa lagi KINGGG!!”.

Selama program berjalan, sistem akan melakukan pengecekan input agar pengguna tidak memasukkan data yang salah, misalnya memastikan input nomor klub berupa angka dan indeksnya valid. Program akan terus menampilkan menu secara berulang menggunakan perulangan while, hingga pengguna memilih opsi (5) keluar. Setelah itu, program akan berhenti sepenuhnya.

2. Deskripsi Singkat Program

Program ini adalah sistem manajemen data statistik klub sepak bola Liga Champions UEFA (UCL) berbasis Python, dengan fitur autentikasi login/register dan menu interaktif untuk melihat, menambah, memperbarui, atau menghapus data klub (main, menang, seri, kalah, gol dicetak, gol kebobolan). Program menggunakan dictionary untuk penyimpanan data dan loop while untuk navigasi menu, memastikan akses terbatas dan validasi input.

Fungsi/Manfaat Utama:

- Memudahkan pengguna untuk melihat, menambah, mengubah, dan menghapus data klub secara dinamis.
- Menyediakan sistem login dan register untuk pengguna baru, agar hanya pengguna yang berhak dapat mengakses data.
- 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:

```
# dictionary berisi data klub sepak bola di Liga Champions UEFA
klub_ucl = {
    "FC Bayern München": {"main": 2, "menang": 2, "seri": 0, "kalah": 0,
    "gol_dicetak": 8, "gol_kebobolan": 2},
    "Real Madrid": {"main": 2, "menang": 2, "seri": 0, "kalah": 0,
    "gol_dicetak": 7, "gol_kebobolan": 1},
    "Paris Saint-Germain F.C.": {"main": 2, "menang": 2, "seri": 0, "kalah":
    0, "gol_dicetak": 6, "gol_kebobolan": 1},
    "Inter Milan": {"main": 2, "menang": 2, "seri": 0, "kalah": 0,
    "gol_dicetak": 5, "gol_kebobolan": 0},
    "Arsenal F.C.": {"main": 2, "menang": 2, "seri": 0, "kalah": 0,
    "gol_dicetak": 4, "gol_kebobolan": 0}
}

# dictionary untuk menyimpan username dan password
data_user = {
    "ggmu": "treble"
}

print("\nSelamat Datang di Sistem Data Statistik Klub UCL")
print("="*50)
```

```

# login bisa memakai usn: ggmu dan pw: treble, atau register yahhhhhh
login_ok = False
while login_ok == False:
    print("1. Login")
    print("2. Register")
    print("3. Keluar")
    pilih_opsi = input("Pilih menu (1-3): ")

    if pilih_opsi == "1":
        username = input("Masukkan Username: ")
        password = input("Masukkan Password: ")

        if username in data_user and data_user[username] == password:
            login_ok = True
            print("Login berhasil!")
        else:
            print("Username atau Password Salah KINGGG! Coba Lagi KINGGG!.\n")

    elif pilih_opsi == "2":
        usn_baru = input("Masukkan Username Baru: ")
        pass_baru = input("Masukkan Password Baru: ")
        data_user[usn_baru] = pass_baru
        print("Registrasi berhasil! Silahkan login.\n")

    elif pilih_opsi == "3":
        print("Terima kasih telah menggunakan sistem ini. Sampai jumpa!")
        exit()

    else:
        print("Opsi tidak valid. Silahkan coba lagi.\n")

while True:
    print("")
    print("="*50)
    print("Menu Statistik Klub UCL")
    print("="*50)
    print("1. Lihat Statistik Tim Klub")
    print("2. Tambah Data Klub")
    print("3. Update Data Klub")
    print("4. Hapus Data Klub")

```

```

print("5. Keluar")

pilihan = input("Silahkan Pilih Menu (1-5): ")

if pilihan == "1":
    print("="*50)
    print("Statistik Tim Klub")
    print("="*50)
    list_klub = list(klub_ucl.keys())
    indeks = 1
    for klub in list_klub:
        stats = klub_ucl[klub]
        print(f"\n{indeks}. Klub: {klub}\nMain:
{stats['main']},\nMenang: {stats['menang']},\nSeri: {stats['seri']},\nKalah:
{stats['kalah']},\nGol Dicletak: {stats['gol_dicletak']},\nGol Kebobolan:
{stats['gol_kebobolan']}")
        indeks += 1
    print("-"*50)

elif pilihan == "2":
    print("="*50)
    print("Silahkan Tambah Data Klub")
    print("="*50)
    nama = input("Masukkan Nama Klub: ")
    main = int(input("Masukkan Jumlah Main: "))
    menang = int(input("Masukkan Jumlah Menang: "))
    seri = int(input("Masukkan Jumlah Seri: "))
    kalah = int(input("Masukkan Jumlah Kalah: "))
    goldicletak = int(input("Masukkan Jumlah Gol Dicletak: "))
    golkebobolan = int(input("Masukkan Jumlah Gol Kebobolan: "))
    klub_ucl[nama] = {
        "main": main,
        "menang": menang,
        "seri": seri,
        "kalah": kalah,
        "gol_dicletak": goldicletak,
        "gol_kebobolan": golkebobolan
    }
    print("Data klub Berhasil Ditambahkan.")

elif pilihan == "3":

```

```

print("="*50)
print("Daftar Klub yang Bisa Diupdate:")
list_klub = list(klub_ucl.keys())
indeks = 1
for klub in list_klub:
    print(f"{indeks}. {klub}")
    indeks += 1
print("="*50)
input_klub = input("Pilih Nomor Klub yang Ingin Diupdate: ")
if input_klub.isdigit():
    index_klub = int(input_klub) - 1 # Karena indeks list mulai
    dari 0 di Output, dan mempermudah user
    if index_klub >= 0 and index_klub < len(list_klub):
        klub_terpilih = list_klub[index_klub]
        print(f"Update Data untuk {klub_terpilih}")
        main = int(input("Masukkan Jumlah Main: "))
        menang = int(input("Masukkan Jumlah Menang: "))
        seri = int(input("Masukkan Jumlah Seri: "))
        kalah = int(input("Masukkan Jumlah Kalah: "))
        goldicetak = int(input("Masukkan Jumlah Gol Dicetak: "))
        golkebobolan = int(input("Masukkan Jumlah Gol Kebobolan: "))
        klub_ucl[klub_terpilih] = {
            "main": main,
            "menang": menang,
            "seri": seri,
            "kalah": kalah,
            "gol_dicetak": goldicetak,
            "gol_kebobolan": golkebobolan
        }
        print("Data klub Berhasil Diupdate KINGGG!!.")
    else:
        print("Indeks tidak valid.")
else:
    print("Indeks Harus Berupa Angka KINGGG!!.")

elif pilihan == "4":
    print("="*50)
    print("Daftar Klub yang Bisa Dihapus:")
    list_klub = list(klub_ucl.keys())
    indeks = 1
    for klub in list_klub:

```

```

        print(f"{indeks}. {klub}")
        indeks += 1
    print("="*50)
    input_klub = input("Pilih Nomor Klub yang Ingin Dihapus: ")
    if input_klub.isdigit():
        index_klub = int(input_klub) - 1 # Karena indeks list mulai
        dari 0 di Output, dan mempermudah user
        if index_klub >= 0 and index_klub < len(list_klub):
            klub_terpilih = list_klub[index_klub]
            del klub_ucl[klub_terpilih]
            print("Data klub Berhasil Dihapus KINGGG!!.")
        else:
            print("Indeks tidak valid.")
    else:
        print("Indeks Harus Berupa Angka KINGGG!!.")

elif pilihan == "5":
    print("Terima kasih telah menggunakan sistem ini. Sampai jumpa!")
    print("\nKING MU < LIVERPOOL 🤩 (19/10/2025)")
    print("kapan lagi mu menang di anfield aowmkoakowka")
    break

```

A. Fitur Hapus Data

Source Code:

```

elif pilihan == "4":

    print("="*50)

    print("Daftar Klub yang Bisa Dihapus:")

    list_klub = list(klub_ucl.keys())

    indeks = 1

    for klub in list_klub:

        print(f"{indeks}. {klub}")

```



```

        indeks += 1

    print("="*50)

    input_klub = input("Pilih Nomor Klub yang Ingin Dihapus: ")

    if input_klub.isdigit():

        index_klub = int(input_klub) - 1 # Karena indeks list mulai dari
0 di Output, dan mempermudah user

        if index_klub >= 0 and index_klub < len(list_klub):

            klub_terpilih = list_klub[index_klub]

            del klub_ucl[klub_terpilih]

            print("Data klub Berhasil Dihapus KINGGG!!.")

        else:

            print("Indeks tidak valid.")

    else:

        print("Indeks Harus Berupa Angka KINGGG!!.")

```

4. Hasil Output

```
Selamat Datang di Sistem Data Statistik Klub UCL
=====
1. Login
2. Register
3. Keluar
Pilih menu (1-3): 1
Masukkan Username: ggmu
Masukkan Password: treble
Login berhasil!
```

Gambar <Hasil Output Login>

```
Selamat Datang di Sistem Data Statistik Klub UCL
=====
1. Login
2. Register
3. Keluar
Pilih menu (1-3): 2
Masukkan Username Baru: tes
Masukkan Password Baru: 123
Registrasi berhasil! Silahkan login.

1. Login
2. Register
3. Keluar
Pilih menu (1-3): 1
Masukkan Username: tes
Masukkan Password: 123
Login berhasil!
```

Gambar <Hasil Output Register>

```
Selamat Datang di Sistem Data Statistik Klub UCL
=====
1. Login
2. Register
3. Keluar
Pilih menu (1-3): 3
Terima kasih telah menggunakan sistem ini. Sampai jumpa!
PS C:\praktikum-apd> █
```

Gambar <Hasil Output Keluar(Opsi 3)>

```

=====
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)>

```
=====
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): 2
=====
Silahkan Tambah Data Klub
=====
Masukkan Nama Klub: AFC Ajax
Masukkan Jumlah Main: 2
Masukkan Jumlah Menang: 0
Masukkan Jumlah Seri: 0
Masukkan Jumlah Kalah: 2
Masukkan Jumlah Gol Dicetak: 0
Masukkan Jumlah Gol Kebobolan: 6
Data klub Berhasil Ditambahkan.
```

Gambar <Hasil Output Menu 2(Tambah 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): 3
=====
Daftar Klub yang Bisa Diupdate:
1. FC Bayern München
2. Real Madrid
3. Paris Saint-Germain F.C.
4. Inter Milan
5. Arsenal F.C.
=====
Pilih Nomor Klub yang Ingin Diupdate: 1
Update Data untuk FC Bayern München
Masukkan Jumlah Main: 3
Masukkan Jumlah Menang: 3
Masukkan Jumlah Seri: 0
Masukkan Jumlah Kalah: 0
Masukkan Jumlah Gol Dicetak: 11
Masukkan Jumlah Gol Kebobolan: 3
Data klub Berhasil Diupdate KINGGG!!.
```

Gambar <Hasil Output Menu 3(Update 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): 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 4(Hapus 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): 5
Terima kasih telah menggunakan sistem ini. Sampai jumpa!

KING MU < LIVERPOOL 🤪 (19/10/2025)
kapan lagi mu menang di anfield aowmkoakowka
PS C:\praktikum-apd>
```

Gambar <Hasil Output Menu 5(Keluar Program)>

5. Langkah-langkah GIT

```
PS C:\praktikum-apd>
PS C:\praktikum-apd> git add .
PS C:\praktikum-apd> git commit -m "apd6"
[main fe8228a] apd6
 3 files changed, 646 insertions(+)
 create mode 100644 kelas/tes set&dictionary
 create mode 100644 post-test/post-test-apd-6/2509106051-UlungPriskeajiBintangRamadhan-PT-6.drawio
 create mode 100644 post-test/post-test-apd-6/2509106051-UlungPriskeajiBintangRamadhan-PT-6.py
PS C:\praktikum-apd> git push
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 28 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (8/8), 5.86 KiB | 2.93 MiB/s, done.
Total 8 (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
   5d3144e..fe8228a  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).