

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



Disusun oleh:

Mahdi Sarwan

Abdullah

(2509106067)

Kelas (B1 '25)

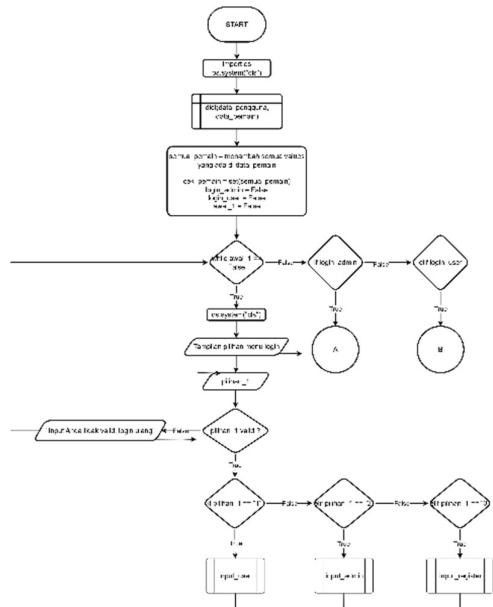
PROGRAM STUDI INFORMATIKA

UNIVERSITAS MULAWARMAN

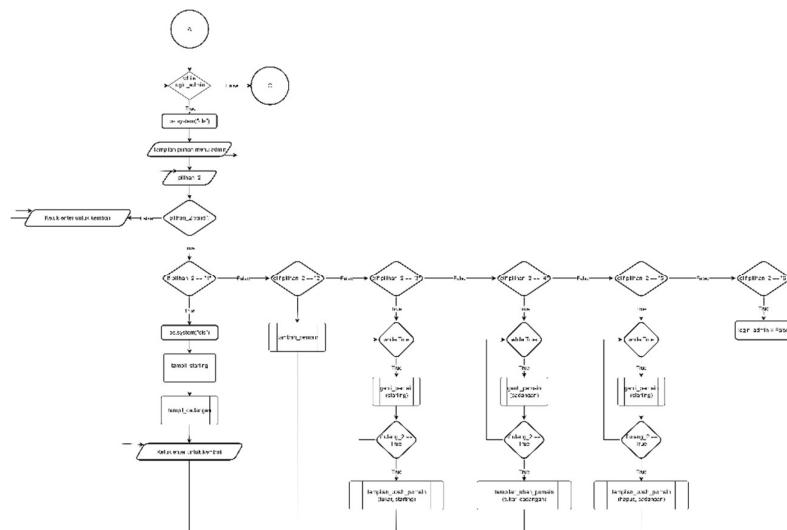
SAMARINDA

2025

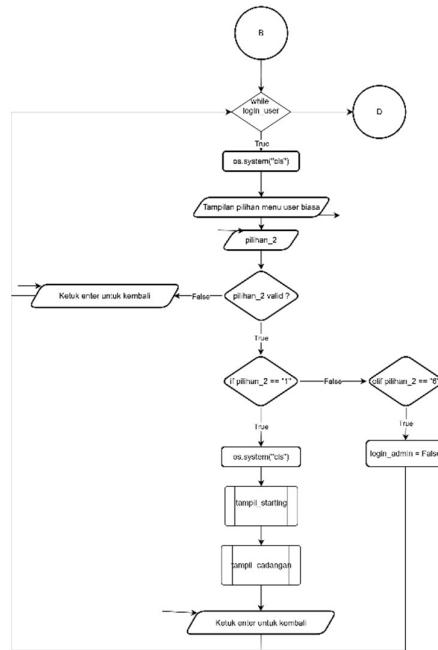
1. Flowchart



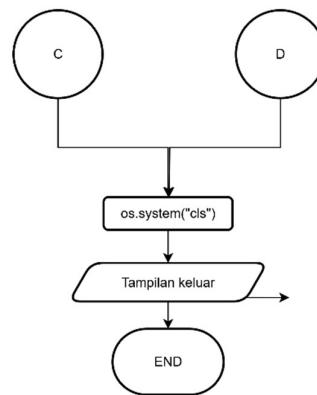
Gambar 1.1 Program Utama



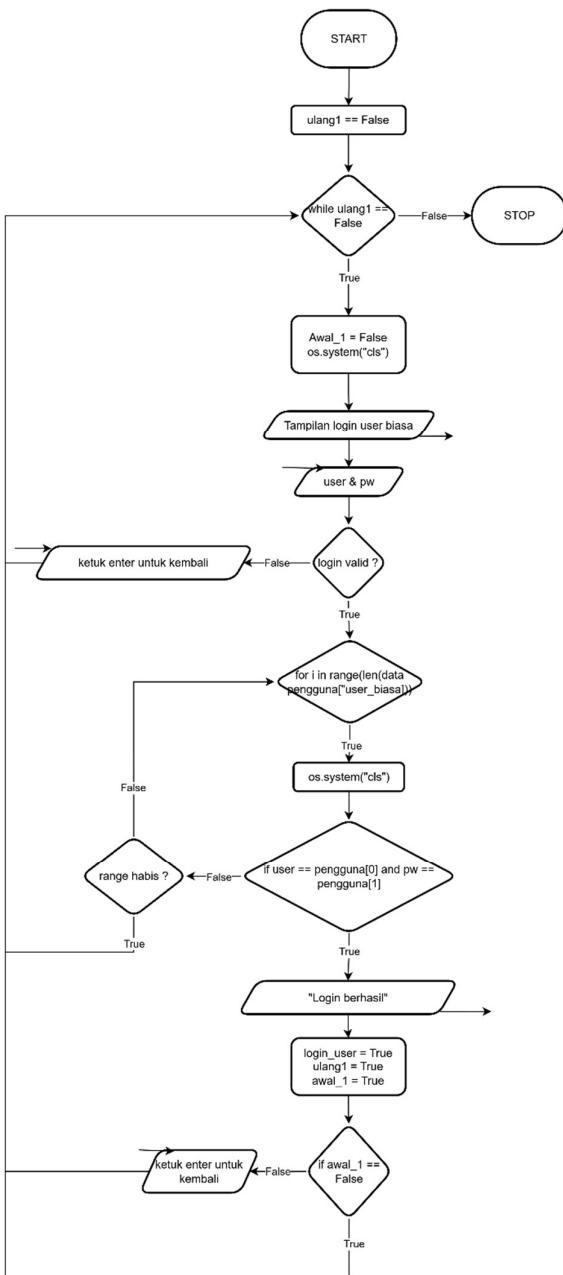
Gambar 1.2 Program Utama



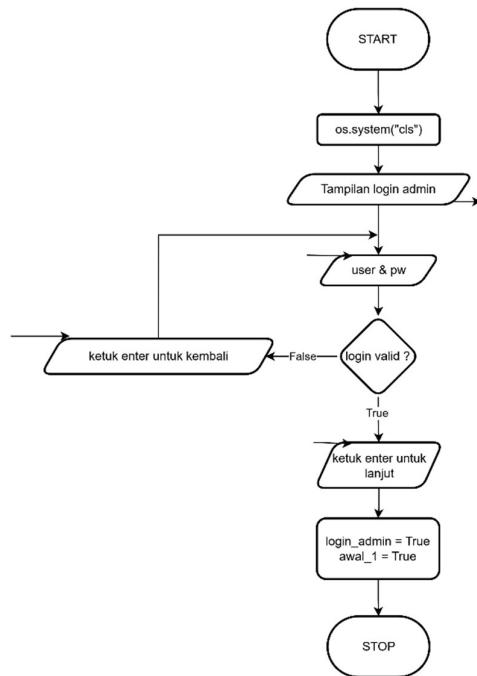
Gambar 1.3 Program Utama



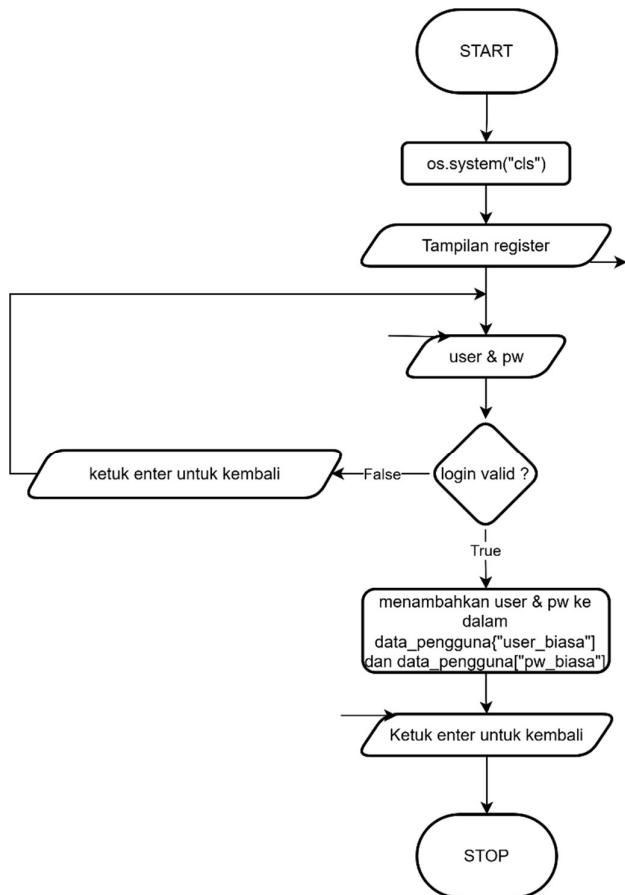
Gambar 1.4 Program Utama



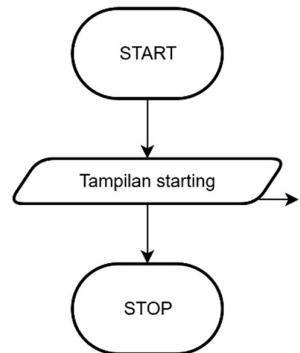
Gambar 1.5 Fungsi input_user



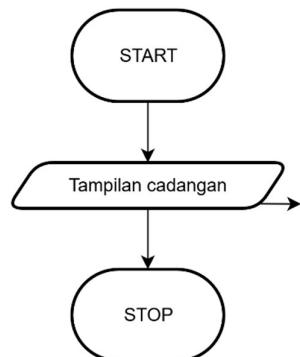
Gambar 1.6 Fungsi input_admin



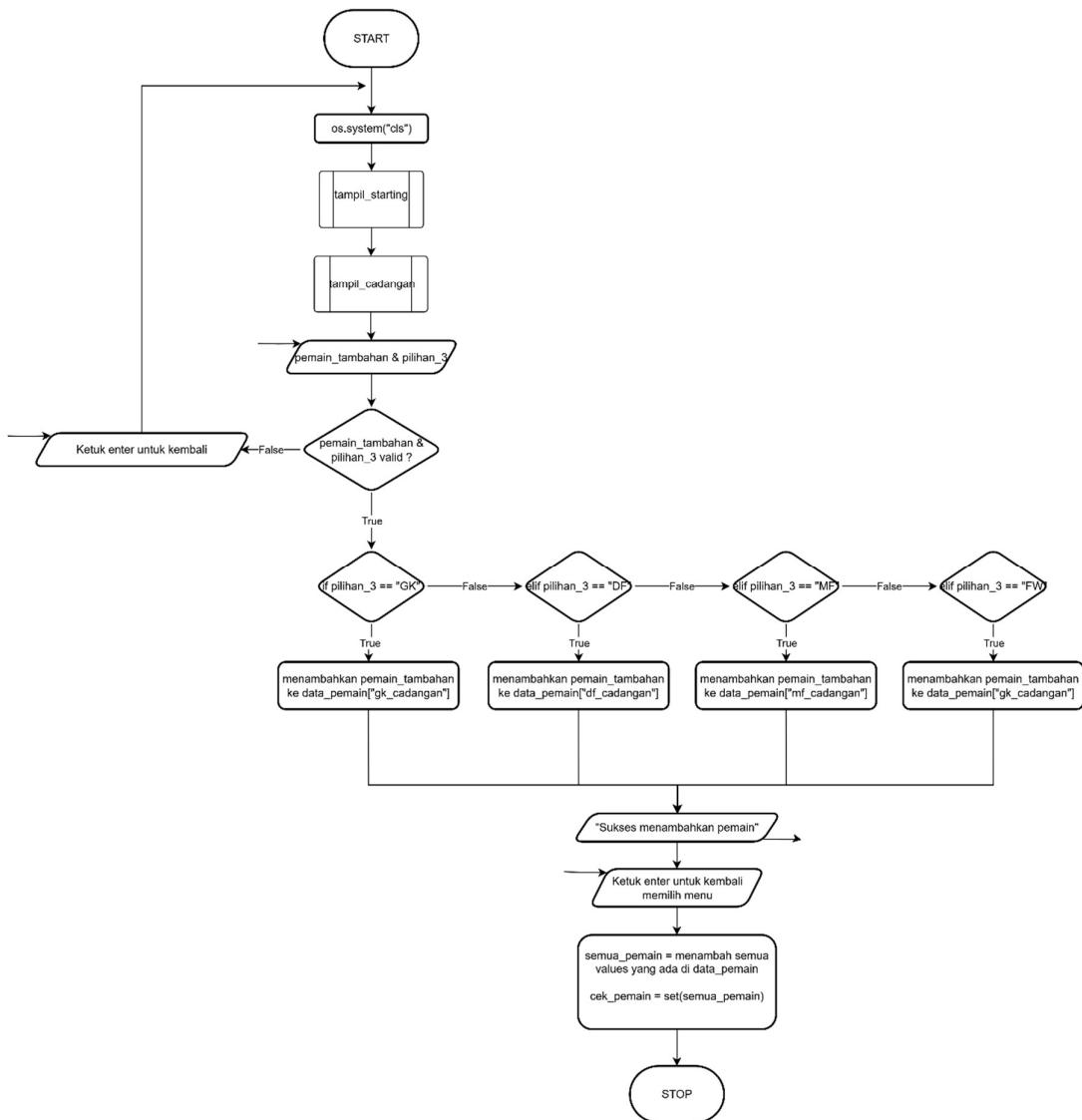
Gambar 1.7 Fungsi input_register



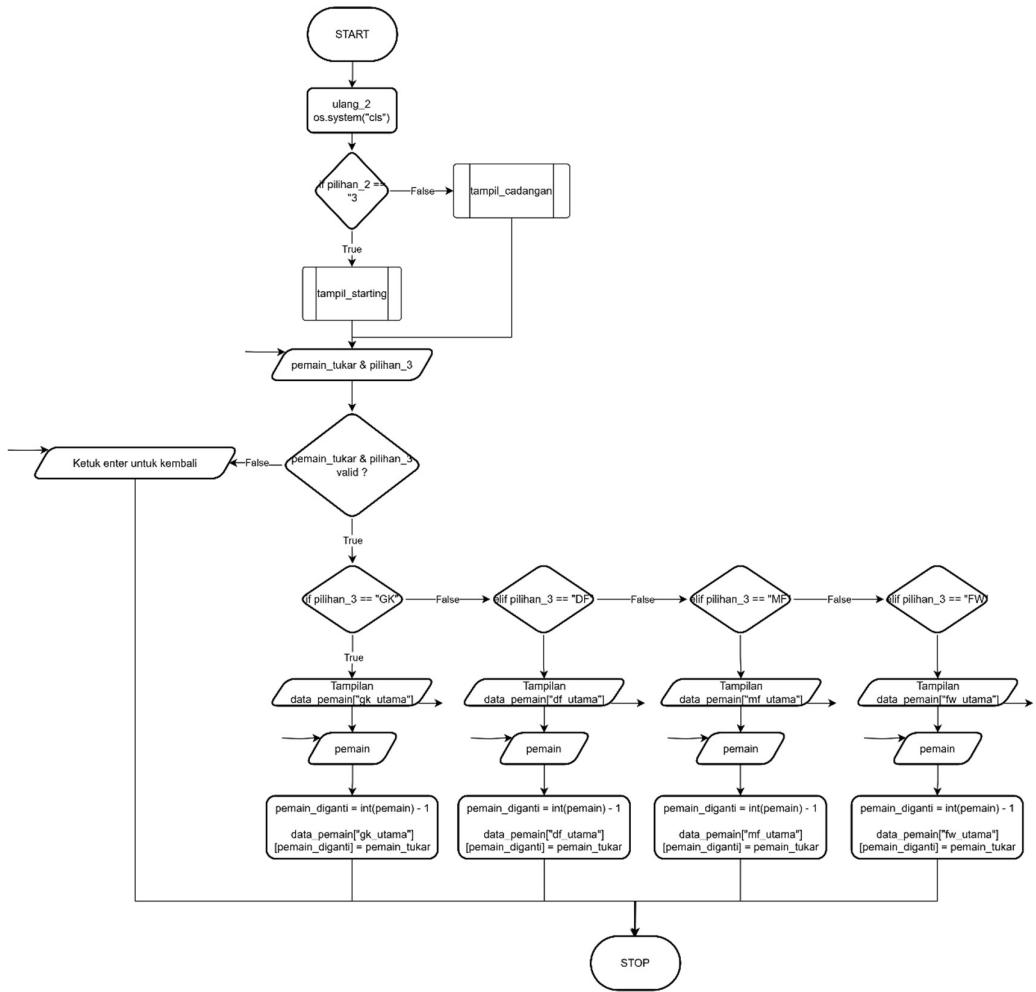
Gambar 1.8 Fungsi tampil_starting



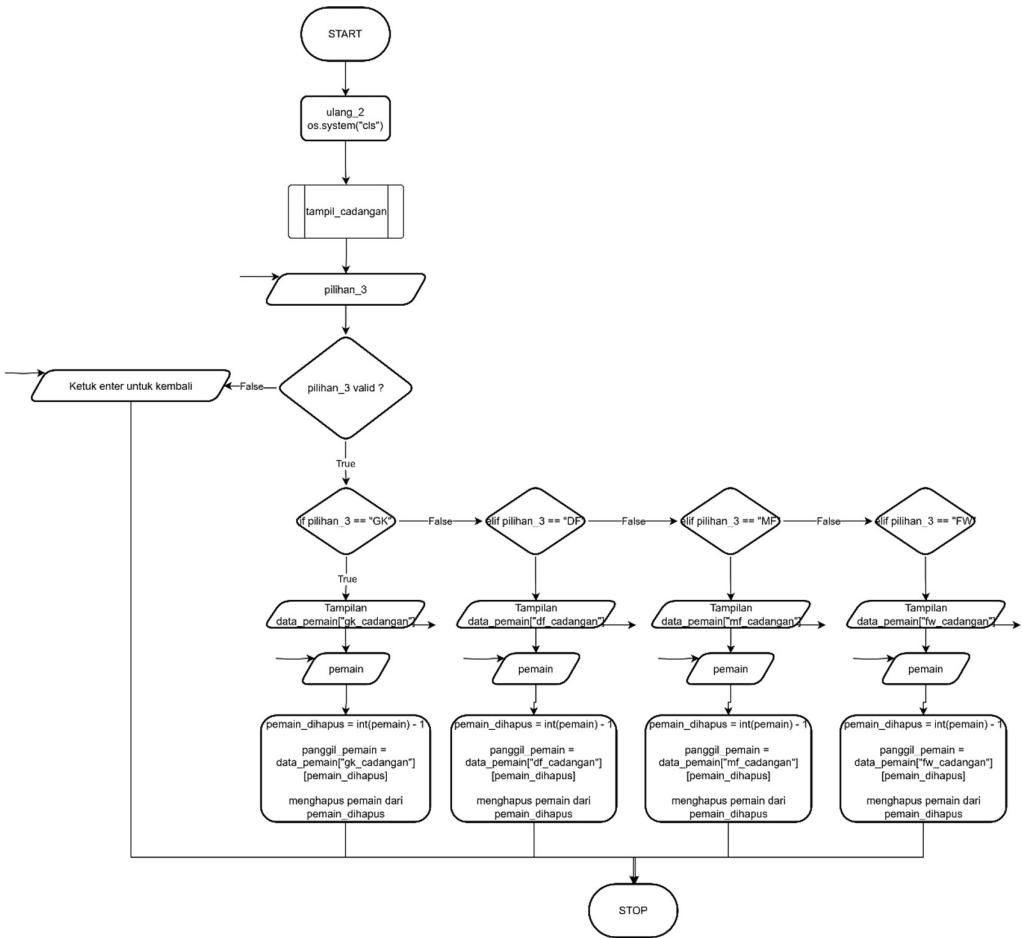
Gambar 1.9 Fungsi tampil_cadangan



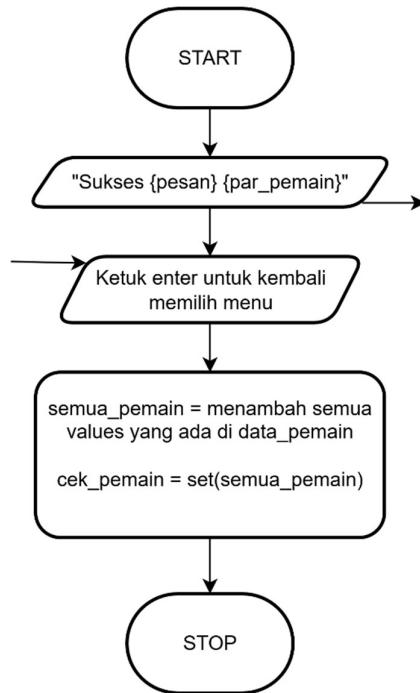
Gambar 1.10 Fungsi tambah_pemain



Gambar 1.11 Fungsi Ganti_pemain



Gambar 1.12 Fungsi hapus_pemain



Gambar 1.13 Fungsi tampilan_ubah_pemain

2. Deskripsi Singkat Program

1. Struktur Data Utama

* Akses Pengguna: Program memuat dua set data utama: data_pengguna (berisi username dan password Admin dan Pengguna Biasa) dan data_pemain (daftar pemain inti dan cadangan, dikelompokkan berdasarkan posisi GK, DF, MF, FW).

* Otentikasi: Pengguna harus melewati proses login (input_user() atau input_admin()) atau mendaftar (user_register()) sebelum mengakses menu.

2. Fitur Pengguna Biasa (User)

* Hak Akses Terbatas: Pengguna Biasa hanya memiliki satu opsi utama setelah login, yaitu melihat daftar lengkap Starting Line Up dan pemain cadangan (tampil_starting() dan tampil_cadangan()).

3. Fitur Admin

* Hak Akses Penuh: Admin dapat melihat daftar pemain dan melakukan modifikasi data pemain:

* Menambah pemain baru ke daftar cadangan (tambah_pemain()).

* Mengganti pemain yang sudah ada, baik di Starting Line Up maupun di daftar Cadangan (ganti_pemain()).

* Menghapus pemain dari daftar cadangan (hapus_pemain()).

4. Elemen Kunci Pemrograman

* Fungsi (def): Seluruh alur program dipisahkan ke dalam fungsi-fungsi (misalnya, def input_user(), def tambah_pemain()) untuk membuat kode modular dan mudah dibaca.

* Penanganan Kesalahan (try except): Digunakan secara luas di hampir semua fungsi input untuk mencegah program crash. Blok ini memastikan bahwa input kosong, password salah, atau pemilihan nomor pemain yang tidak valid dapat ditangani dan diinformasikan kepada pengguna tanpa menghentikan program.

* Kontrol Alur (return): Digunakan di dalam fungsi (terutama dalam blok try except) untuk menghentikan eksekusi fungsi saat ini atau, dalam kasus error, untuk memanggil ulang fungsi tersebut agar pengguna dapat mencoba input kembali.

5. Penggunaan Library

Program menggunakan os dan datetime untuk tugas-tugas sistem dan waktu, prettytable untuk tampilan tabel yang rapi, dan modul lokal (login dan crud) untuk logika bisnis utama (otentikasi dan manajemen data pemain).

3. Source Code

```
● ● ●
1 import os
2 from login import input_user, input_admin, input_register
3 from crud import tampil_starting, tampil_cadangan, tambah_pemain, ganti_pemain, hapus_pemain, tampilan_ubah_pemain
4 from prettytable import PrettyTable
5 import datetime as dt
6
7 login_admin = False
8 login_user = False
9
10 awal_1 = False
11 while awal_1 == False:
12     os.system("cls")
13     tabel_menu = PrettyTable()
14     tabel_menu.title = "ANDA INGIN LOGIN SEBAGAI:"
15     tabel_menu.field_names = ["kiri", "kanan"]
16     tabel_menu.header = False
17     tabel_menu.add_rows([
18         ["1.", "Pengguna Biasa"],
19         ["2.", "Sebagai Admin"],
20         ["3.", "Daftar Sebagai Pengguna Baru"]
21     ])
22     print(tabel_menu)
23
24 pilihan_1 = input("Pilih menu (1-3) = ").strip()
25
26 if pilihan_1 == "":
27     input("\n(Masukkan karakter, ketuk enter untuk memilih kembali)")
28     continue
29 elif not pilihan_1.isdigit() or pilihan_1 == "0":
30     input("\n(Masukkan angka sesuai pilihan, ketuk enter untuk memilih kembali)")
31     continue
32
33 elif pilihan_1 == "1":
34     user, login_user, awal_1 = input_user()
35
36 elif pilihan_1 == "2":
37     user, login_admin, awal_1 = input_admin()
38
39 elif pilihan_1 == "3":
40     input_register()
41
42 else:
43     input("\n(Input tidak valid, ketuk enter untuk memilih kembali)")
44
```

```

1
2 if login_admin:
3     while login_admin:
4         os.system("cls")
5         print("== Selamat Datang Tuan Admin ===\n")
6         tabel_menu_admin = PrettyTable()
7         tabel_menu_admin.title = "Mau ngapain hari ini?"
8         tabel_menu_admin.field_names = ["kiri", "kanan"]
9         tabel_menu_admin.header = False
10        tabel_menu_admin.add_rows([
11            "[[1], "Lihat Daftar Line Up"],
12            "[[2], "Tambah Pemain Cadangan"],
13            "[[3], "Ganti Starting Line Up"],
14            "[[4], "Ganti Pemain Cadangan"],
15            "[[5], "Hapus Pemain Cadangan"],
16            "[[6], "Keluar"]
17        ])
18        print(tabel_menu_admin)
19
20    pilihan_2 = input("Pilih menu (1-6) = ").strip()
21
22    if pilihan_2 == "1":
23        data_waktu = dt.datetime.now()
24        os.system("cls")
25        print(F"Daftar Line Up Timnas Indonesia ({data_waktu.strftime("%A")}, {data_waktu.day} - {data_waktu.month} - {data_waktu.year})\n")
26        tampil_starting()
27        print()
28        tampil_cadangan()
29        input("\n(Ketuk enter untuk kembali memilih menu)")
30
31    elif pilihan_2 == "2":
32        tambah_pemain()
33
34    elif pilihan_2 == "3":
35        while True:
36            os.system("cls")
37            tampil_starting()
38            pemain_tukar, ulang_2 = ganti_pemain("gk_utama", "df_utama", "mf_utama", "fw_utama")
39            if ulang_2 == True:
40                break
41
42        tampilan_ubah_pemain(pemain_tukar, "mengganti pemain starting dengan")
43
44    elif pilihan_2 == "4":
45        while True:
46            os.system("cls")
47            tampil_cadangan()
48            pemain_tukar, ulang_2 = ganti_pemain("gk_cadangan", "df_cadangan", "mf_cadangan", "fw_cadangan")
49            if ulang_2 == True:
50                break
51

```

```
1     tampilan_ubah_pemain(pemain_tukar, "mengganti pemain cadangan dengan")
2
3     elif pilihan_2 == "5":
4         while True:
5             panggil_pemain, ulang_2 = hapus_pemain()
6             if ulang_2 == True:
7                 break
8
9             tampilan_ubah_pemain(panggil_pemain, "menghapus")
10
11    elif pilihan_2 == "6":
12        login_admin = False
13
14    else:
15        input("\n(Input tidak valid, ketuk enter untuk kembali)")
16
17 elif login_user:
18     while login_user:
19         os.system("cls")
20         print(f"==> Selamat Datang Tuan Muda {user} ==>\n")
21         tabel_menu_user = PrettyTable()
22         tabel_menu_user.title = "Mau ngapain hari ini?"
23         tabel_menu_user.field_names = ["kiri", "kanan"]
24         tabel_menu_user.header = False
25         tabel_menu_user.add_rows([
26             ["[1]", "Lihat Daftar Line Up"],
27             ["[2]", "Keluar"]
28         ])
29         print(tabel_menu_user)
30
31         pilihan_2 = input("Pilih menu (1-2) = ").strip()
32
33         if pilihan_2 == "1":
34             data_waktu = dt.datetime.now()
35             os.system("cls")
36             print("Daftar Line Up Timnas Indonesia ((data_waktu.strftime("%A")), (data_waktu.day) - (data_waktu.month) - (data_waktu.year))\n")
37             tampil_starting()
38             print()
39             tampil_cadangan()
40             input("\n(Ketuk enter untuk kembali memilih menu)")
41
42
```

```
1
2     elif pilihan_2 == "2":
3         login_user = False
4
5     else:
6         input("\n(Input tidak valid, ketuk enter untuk kembali)")
7
8 os.system("cls")
9 print(f"◆ Terima kasih atas waktunya, {user}. Sampai jumpa di lain kesempatan! Selamat tinggal. 🌟")
```

```
 1 data_pengguna = {
 2     "user_id": ["1admin"],
 3     "user_utama": ["user0123"],
 4     "user_bilas": ["xandri1"],
 5     "pw_biasa": ["06/'"]
 6 }
 7
 8 data_pemasin = {
 9     "pk_utama": ["Marten Paes"],
10     "sf_utama": ["Jay Jozes",
11                   "Ricky Ridho",
12                   "Merry Huber",
13                   "Calvin Vebond",
14                   "Kevin Dilks"],
15     ],
16     "mf_utama": ["Thom Hope",
17                   "Ade Paligassy",
18                   "Marcelino"],
19     ],
20     "fw_utama": ["Nafael Struick",
21                   "Ole Rommy"],
22     ],
23     "gk_cadangan": ["Neill Astero",
24                   "Nedeno Argawinata"],
25     ],
26     "df_cadangan": ["Freddy Hilman",
27                   "Sandy Walsh",
28                   "Yakob Sayuri",
29                   "Vance Sayuri",
30                   "Justin Huber"],
31     ],
32     "mf_cadangan": ["Ivar Jenner",
33                   "Ricky Kambuya"],
34     ],
35     "fw_cadangan": ["Ragnar Oratmanggoen",
36                   "Milano Jonathans",
37                   "Aggi Maulana Vilkey"],
38     ],
39   }
40
41 sum_pemasin = data_pemasin["pk_utama"] + data_pemasin["sf_utama"] + data_pemasin["mf_utama"] + data_pemasin["fw_utama"] + data_pemasin["pk_cadangan"] + data_pemasin["df_cadangan"] + data_pemasin["mf_cadangan"] + data_pemasin["fw_cadangan"]
42 cek_pemasin = set(sum_pemasin)
43
```

```

1 import os
2 from data import data_pengguna, data_pemain, semua_pemain, cek_pemain
3
4 def input_user():
5     global awal_1
6     global login_user
7     global user
8
9     ulang_1 = False
10    while ulang_1 == False:
11        awal_1 = False
12        os.system("cls")
13        print("==== Login Sebagai User Biasa ===\n")
14
15    try:
16        user = input("Masukkan username Anda = ").strip()
17        pw = input("Masukkan password Anda = ").strip()
18
19        if user == "" or pw == "":
20            raise ValueError("\n(Masukkan karakter, tekan enter untuk login kembali)")
21
22        for i in range(len(data_pengguna["user_biasa"])):
23            if user == data_pengguna["user_biasa"][i] and pw == data_pengguna["pw_biasa"][i]:
24                input("\n(Login berhasil, ketuk enter untuk lanjut)")
25                login_user = True
26                ulang_1 = True
27                awal_1 = True
28                return user, login_user, awal_1
29            elif user == data_pengguna["user_biasa"][i]:
30                awal_1 = True
31                raise ValueError("\n>Password salah, ketuk enter untuk kembali")
32
33        if awal_1 == False:
34            raise ValueError("\n(User tidak ditemukan, ketuk enter untuk login kembali)")
35
36    except Exception as e:
37        input(e)
38        continue
39
40 def input_admin():
41     global user
42     global login_admin
43     global awal_1
44
45     while True:
46         os.system("cls")
47         print("==== Login Sebagai Admin ===\n")
48
49     try:
50         user = input("Masukkan username Anda = ").strip()
51         pw = input("Masukkan password Anda = ").strip()
52

```

```
1      if user != data_pengguna["user_admin"][0] or pw != data_pengguna["pw_admin"][0]:
2          if user == "" or pw == "":
3              raise ValueError("\nMasukkan karakter, ketuk enter untuk kembali")
4          elif user == data_pengguna["user_admin"][0]:
5              raise ValueError("\nPassword salah, ketuk enter untuk kembali")
6          elif pw == data_pengguna["pw_admin"][0]:
7              raise ValueError("\nUsername salah, ketuk enter untuk kembali")
8          else:
9              raise ValueError("\nUsername dan password salah, ketuk enter untuk kembali")
10
11
12     except Exception as e:
13         input(e)
14         continue
15
16     input("\n(Login berhasil, ketuk enter untuk lanjut)")
17     login_admin = True
18     awal_1 = True
19     return user, login_admin, awal_1
20
21 def input_register():
22     global data_pengguna
23     global user
24
25     while True:
26         os.system("cls")
27         print("== Menu Register ==\n")
28
29         try:
30             user = input("Masukkan username Anda = ").strip()
31             pw = input("Masukkan password Anda = ").strip()
32
33             if user == "" or pw == "":
34                 raise ValueError("\nMasukkan karakter, ketuk enter untuk kembali")
35             elif user in data_pengguna["user_biasa"]:
36                 raise ValueError("\nPengguna sudah ada, harap ganti username Anda")
37
38         except Exception as e:
39             input(e)
40             continue
41
42         data_pengguna["user_biasa"].append(user)
43         data_pengguna["pw_biasa"].append(pw)
44         input("\n(Register berhasil, ketuk enter untuk login ulang)")
45         break
46
```

```

1  from data import data_pemain, cek_pemain, semua_pemain
2  from prettytable import PrettyTable
3  import os
4
5  def tampil_starting():
6      data_starting = []
7      for i in range(len(data_pemain["gk_utama"])):
8          nomor = i + 1
9          pemain_starting = data_pemain["gk_utama"][i]
10         data_starting.append([f"{nomor}.", pemain_starting, "GK"])
11     for i in range(len(data_pemain["df_utama"])):
12         nomor = i + 2
13         pemain_starting = data_pemain["df_utama"][i]
14         data_starting.append([f"{nomor}.", pemain_starting, "DF"])
15     for i in range(len(data_pemain["mf_utama"])):
16         nomor = i + 7
17         pemain_starting = data_pemain["mf_utama"][i]
18         data_starting.append([f"{nomor}.", pemain_starting, "MF"])
19     for i in range(len(data_pemain["fw_utama"])):
20         nomor = i + 10
21         pemain_starting = data_pemain["fw_utama"][i]
22         data_starting.append([f"{nomor}.", pemain_starting, "FW"])
23
24     tabel_starting = PrettyTable()
25     tabel_starting.title = "STARTING"
26     tabel_starting.field_names = ["NO.", "Nama Pemain", "Posisi"]
27     tabel_starting.add_rows(data_starting)
28     print(tabel_starting)
29
30 def tampil_cadangan():
31     data_cadangan = []
32     for i in range(len(data_pemain["gk_cadangan"])):
33         nomor = i + 1
34         pemain_cadangan = data_pemain["gk_cadangan"][i]
35         data_cadangan.append([f"{nomor}.", pemain_cadangan, "GK"])
36     for i in range(len(data_pemain["df_cadangan"])):
37         nomor = i + 1 + len(data_pemain["gk_cadangan"])
38         pemain_cadangan = data_pemain["df_cadangan"][i]
39         data_cadangan.append([f"{nomor}.", pemain_cadangan, "DF"])
40     for i in range(len(data_pemain["mf_cadangan"])):
41         nomor = i + 1 + len(data_pemain["gk_cadangan"] + data_pemain["df_cadangan"])
42         pemain_cadangan = data_pemain["mf_cadangan"][i]
43         data_cadangan.append([f"{nomor}.", pemain_cadangan, "MF"])
44

```

```

1     for i in range(len(data_pemain["fw_cadangan"])):
2         nomor = i + 1 + len(data_pemain["gk_cadangan"]) + data_pemain["df_cadangan"] + data_pemain["mf_cadangan"])
3         pemain_cadangan = data_pemain["fw_cadangan"][i]
4         data_cadangan.append([f"[nomor] ", pemain_cadangan, "FW"])
5
6     tabel_cadangan = PrettyTable()
7     tabel_cadangan.title = "CADANGAN"
8     tabel_cadangan.field_names = ["NO.", "Nama Pemain", "Posisi"]
9     tabel_cadangan.add_rows(data_cadangan)
10    print(tabel_cadangan)
11
12 def tambah_pemain():
13     global data_pemain
14     global semua_pemain
15     global cek_pemain
16
17     os.system("cls")
18     tampil_starting()
19     print()
20     tampil_cadangan()
21     try:
22         pemain_tambahan = input("\nMasukkan nama pemain = ").strip()
23
24         if pemain_tambahan == "":
25             raise ValueError("\n(Masukkan karakter, ketuk enter untuk kembali)")
26         elif pemain_tambahan in cek_pemain:
27             raise ValueError("\nPemain sudah ada di line up, ketuk enter untuk menambah kembali") # butuh perbaikan ini
28
29         pilihan_3 = input("Pilih lini yang ingin ditambah (GK/DF/MF/FW) = ").strip().upper()
30         if pilihan_3 == "GK":
31             data_pemain["gk_cadangan"].append(pemain_tambahan)
32         elif pilihan_3 == "DF":
33             data_pemain["df_cadangan"].append(pemain_tambahan)
34         elif pilihan_3 == "MF":
35             data_pemain["mf_cadangan"].append(pemain_tambahan)
36         elif pilihan_3 == "FW":
37             data_pemain["fw_cadangan"].append(pemain_tambahan)
38         else:
39             raise ValueError("\n(Input tidak valid, ketuk enter untuk kembali)")
40

```

```

1     except Exception as e:
2         input(e)
3         return tambah_pemain()
4
5     print("(*Success menambahkan {pemain_tambahan} ke dalam lini {pilihan_3})")
6     input("\n(Ketuk enter untuk keluar menu)")
7     semua_pemain = data_pemain["gk_utama"] + data_pemain["df_utama"] + data_pemain["mf_utama"] + data_pemain["fw_utama"] + data_pemain["gk_cadangan"] + data_pemain["df_cadangan"] + data_pemain["mf_cadangan"] + data_pemain["fw_cadangan"]
8     cek_pemain = set(someall_pemain)
9
10    def ganti_pemain(gk_status, df_status, mf_status, fw_status):
11        global data_pemain
12        global pemain_tukar
13        global ulang_2
14
15        ulang_2 = False
16        pemain_tukar = None
17
18        try:
19            pemain_tukar = input("\nMasukkan nama pemain - ").strip()
20
21            if pemain_tukar == "":
22                raise ValueError("\n(Masukkan karakter, ketuk enter untuk kembali)")
23            elif pemain_tukar in cek_pemain:
24                raise ValueError("\nPemain sudah ada di line up, ketuk enter untuk menambah kembali") # butuh perbaikan ini
25
26            pilihan_3 = input("Pilih lini yang ingin diganti (GK/DF/MF/FW) = ").strip().upper()
27            if pilihan_3 == "GK":
28                for i in range(len(data_pemain[gk_status])):
29                    print(f"({i+1}), {data_pemain[gk_status][i]} ({gk})")
30
31                pemain_diganti = input("Masukkan angka pemain yang ingin diganti - ").strip()
32                if not pemain_diganti.isdigit():
33                    raise ValueError("\n(Masukkan input berupa angka, ketuk enter untuk kembali)")
34                pemain_diganti = int(pemain_diganti) - 1
35
36                if pemain_diganti > len(data_pemain[gk_status]) - 1 or pemain_diganti < 0:
37                    raise IndexError("\nPemain tidak ditemukan, ketuk enter untuk kembali")
38
39                data_pemain[gk_status][pemain_diganti] = pemain_tukar
40                ulang_2 = True
41            else:
42                raise IndexError("\nPemain tidak ditemukan, ketuk enter untuk kembali")
43
44            elif pilihan_3 == "DF":
45                for i in range(len(data_pemain[df_status])):
46                    print(f"({i+1}), {data_pemain[df_status][i]} ({df})")
47
48                pemain_diganti = input("Masukkan angka pemain yang ingin diganti - ").strip()
49                if not pemain_diganti.isdigit():
50                    raise ValueError("\n(Masukkan input berupa angka, ketuk enter untuk kembali)")
51                pemain_diganti = int(pemain_diganti) - 1
52
53                if pemain_diganti > len(data_pemain[df_status]) - 1 or pemain_diganti < 0:
54                    raise IndexError("\nPemain tidak ditemukan, ketuk enter untuk kembali")
55
56                data_pemain[df_status][pemain_diganti] = pemain_tukar
57                ulang_2 = True
58
59            return pemain_tukar, ulang_2
60

```

```

1     elif pilihan_3 == "MF":
2         for i in range(len(data_pemain[mf_status])):
3             print(f" {i+1}. {data_pemain[mf_status][i]} (MF)")
4             pemain = input("\nMasukkan angka pemain yang ingin diganti = ").strip()
5             if not pemain.isdigit():
6                 raise ValueError("\nMasukkan input berupa angka, ketuk enter untuk kembali")
7             pemain_diganti = int(pemain) - 1
8             if pemain_diganti > len(data_pemain[mf_status]) - 1 or pemain_diganti < 0:
9                 raise IndexError("\nPemain tidak ditemukan, ketuk enter untuk kembali")
10            else:
11                data_pemain[mf_status][pemain_diganti] = pemain_tukar
12                ulang_2 = True
13                return pemain_tukar, ulang_2
14        elif pilihan_3 == "FW":
15            for i in range(len(data_pemain[fw_status])):
16                print(f" {i+1}. {data_pemain[fw_status][i]} (FW)")
17                pemain = input("\nMasukkan angka pemain yang ingin diganti = ").strip()
18                if not pemain.isdigit():
19                    raise ValueError("\nMasukkan input berupa angka, ketuk enter untuk kembali")
20                pemain_diganti = int(pemain) - 1
21                if pemain_diganti > len(data_pemain[fw_status]) - 1 or pemain_diganti < 0:
22                    raise IndexError("\nPemain tidak ditemukan, ketuk enter untuk kembali")
23                else:
24                    data_pemain[fw_status][pemain_diganti] = pemain_tukar
25                    ulang_2 = True
26                    return pemain_tukar, ulang_2
27            else:
28                raise ValueError("\nInput tidak valid, ketuk enter untuk kembali")
29        except Exception as e:
30            input(e)
31            return pemain_tukar, ulang_2
32
33    def hapus_pemain():
34        global data_pemain
35        global ulang_2
36        global panggil_pemain
37
38        ulang_2 = False
39        panggil_pemain = None
40
41        os.system("cls")
42        tampil_cadangan()
43        try:
44            pilihan_3 = input("\nPilih lini yang ingin dihapus (GK/DF/MF/FW) = ").strip().upper()
45
46            if pilihan_3 == "GK":
47                for i in range(len(data_pemain["gk_cadangan"])):
48                    print(f" {i+1}. {data_pemain["gk_cadangan"][i]} (GK)")
49                    pemain = input("\nMasukkan angka pemain yang ingin dihapus = ").strip()
50                    if not pemain.isdigit():
51                        raise ValueError("\nMasukkan input berupa angka, ketuk enter untuk kembali")

```

```

1         pemain_dihapus = int(pemain) -1
2         if pemain_dihapus > len(data_pemain["gk_cadangan"]) -1 or pemain_dihapus < 0:
3             raise IndexError("\n(Pemain tidak ditemukan, ketuk enter untuk kembali)")
4         else:
5             panggil_pemain = data_pemain["gk_cadangan"][pemain_dihapus]
6             del data_pemain["gk_cadangan"][pemain_dihapus]
7             ulang_2 = True
8             return panggil_pemain, ulang_2
9     elif pilihan_3 == "DF":
10        for i in range(len(data_pemain["df_cadangan"])):
11            print(f" {i+1}. {data_pemain["df_cadangan"][i]} DF")
12            pemain = input("\nMasukkan angka pemain yang ingin dihapus = ").strip()
13            if not pemain.isdigit():
14                raise ValueError("\n(Masukkan input berupa angka, ketuk enter untuk kembali)")
15            pemain_dihapus = int(pemain) -1
16            if pemain_dihapus > len(data_pemain["df_cadangan"]) -1 or pemain_dihapus < 0:
17                raise IndexError("\n(Pemain tidak ditemukan, ketuk enter untuk kembali)")
18            else:
19                panggil_pemain = data_pemain["df_cadangan"][pemain_dihapus]
20                del data_pemain["df_cadangan"][pemain_dihapus]
21                ulang_2 = True
22                return panggil_pemain, ulang_2
23    elif pilihan_3 == "MF":
24        for i in range(len(data_pemain["mf_cadangan"])):
25            print(f" {i+1}. {data_pemain["mf_cadangan"][i]} (MF)")
26            pemain = input("\nMasukkan angka pemain yang ingin dihapus = ").strip()
27            if not pemain.isdigit():
28                raise ValueError("\n(Masukkan input berupa angka, ketuk enter untuk kembali)")
29            pemain_dihapus = int(pemain) -1
30            if pemain_dihapus > len(data_pemain["mf_cadangan"]) -1 or pemain_dihapus < 0:
31                raise IndexError("\n(Pemain tidak ditemukan, ketuk enter untuk kembali)")
32            else:
33                panggil_pemain = data_pemain["mf_cadangan"][pemain_dihapus]
34                del data_pemain["mf_cadangan"][pemain_dihapus]
35                ulang_2 = True
36                return panggil_pemain, ulang_2
37    elif pilihan_3 == "FW":
38        for i in range(len(data_pemain["fw_cadangan"])):
39            print(f" {i+1}. {data_pemain["fw_cadangan"][i]} (FW)")
40            pemain = input("\nMasukkan angka pemain yang ingin dihapus = ").strip()
41            if not pemain.isdigit():
42                raise ValueError("\n(Masukkan input berupa angka, ketuk enter untuk kembali)")
43            pemain_dihapus = int(pemain) -1
44            if pemain_dihapus > len(data_pemain["fw_cadangan"]) -1 or pemain_dihapus < 0:
45                raise IndexError("\n(Pemain tidak ditemukan, ketuk enter untuk kembali)")
46

```

```

1         else:
2             panggil_pemain = data_pemain["fw_cadangan"][pemain_dihapus]
3             del data_pemain["fw_cadangan"][pemain_dihapus]
4             ulang_2 = True
5             return panggil_pemain, ulang_2
6         else:
7             raise ValueError("\n(Input tidak valid, ketuk enter untuk kembali)")
8     except Exception as e:
9         input()
10        return panggil_pemain, ulang_2
11
12 def tampilan_ubah_pemain(par_pemain, pesan):
13     global semua_pemain
14     global cek_pemain
15
16     print(f"\n(Sukses {pesan}) (par_pemain)")
17     input("\n(Ketuk enter untuk kembali menilai menu)")
18     semua_pemain = data_pemain["gk_utana"] + data_pemain["df_utana"] + data_pemain["mf_utana"] + data_pemain["gk_cadangan"] + data_pemain["df_cadangan"] + data_pemain["mf_cadangan"] + data_pemain["fw_cadangan"]
19     cek_pemain = set(semua_pemain)
20

```

4. Hasil Output

```
+-----+
|      ANDA INGIN LOGIN SEBAGAI:   |
+-----+
| 1. | Pengguna Biasa           |
| 2. | Sebagai Admin            |
| 3. | Daftar Sebagai Pengguna Baru |
+-----+
Pilih menu (1-3) = █
```

==== Login Sebagai Admin ===

Masukkan username Anda = admin

Masukkan password Anda = admin123

(Login berhasil, ketuk enter untuk lanjut)█

==== Selamat Datang Tuan Admin ===

```
+-----+
|      Mau ngapain hari ini?   |
+-----+
| [1] | Lihat Daftar Line Up  |
| [2] | Tambah Pemain Cadangan |
| [3] | Ganti Starting Line Up |
| [4] | Ganti Pemain Cadangan  |
| [5] | Hapus Pemain Cadangan  |
| [6] |          Keluar        |
+-----+
Pilih menu (1-6) = █
```

Daftar Line Up Timnas Indonesia (Tuesday, 4 - 11 - 2025)

STARTING		
NO.	Nama Pemain	Posisi
1.	Marten Paes	GK
2.	Jay Idzes	DF
3.	Rizky Ridho	DF
4.	Justin Hubner	DF
5.	Calvin Verdonk	DF
6.	Kevin Diks	DF
7.	Thom Haye	MF
8.	Joel Pelupessy	MF
9.	Marselino	MF
10.	Rafael Struick	FW
11.	Ole Romeny	FW

CADANGAN		
NO.	Nama Pemain	Posisi
1.	Emil Audero	GK
2.	Nadeo Argawinata	GK
3.	Mees Hilgers	DF
4.	Sandy Walsh	DF
5.	Yakob Sayuri	DF
6.	Yance Sayuri	DF
7.	Justin Hubner	DF
8.	Ivar Jenner	MF
9.	Ricky Kambuaya	MF
10.	Ragnar Oratmangoen	FW
11.	Miliano Jonathans	FW
12.	Egy Maulana Vikry	FW

(Ketuk enter untuk kembali memilih menu) █

★ Terima kasih atas waktunya, admin. Sampai jumpa di lain kesempatan! Selamat tinggal. 🌟

5. Langkah-Langkah GIT

5.1 GIT Add

```
PS D:\praktikum\praktikum_apd> git add .
```

Gambar 5.1 GIT Add

Fungsinya untuk menambah semua file yang ada di directory tersebut.

5.2 GIT Commit

```
PS D:\praktikum\praktikum_apd> git commit -m "hampir final"
[main c75e94f] hampir final
23 files changed, 2507 insertions(+)
create mode 100644 code 1.png
create mode 100644 code 10.png
create mode 100644 code 11.png
create mode 100644 code 12.png
create mode 100644 code 2.png
create mode 100644 code 3.png
create mode 100644 code 4.png
create mode 100644 code 5.png
create mode 100644 code 6.png
create mode 100644 code 7.png
create mode 100644 code 8.png
create mode 100644 code 9.png
create mode 100644 code.png
create mode 100644 kelas/Pertemuan 8.py
create mode 100644 kelas/math.pyi
create mode 100644 post-test/post-test-apd-8/2509106067-Mahdi Sarwan Abdullah-PT- 8/__pycache__/crud.cpython-313.pyc
create mode 100644 post-test/post-test-apd-8/2509106067-Mahdi Sarwan Abdullah-PT- 8/__pycache__/data.cpython-313.pyc
create mode 100644 post-test/post-test-apd-8/2509106067-Mahdi Sarwan Abdullah-PT- 8/__pycache__/login.cpython-313.pyc
create mode 100644 post-test/post-test-apd-8/2509106067-Mahdi Sarwan Abdullah-PT- 8/crud.py
create mode 100644 post-test/post-test-apd-8/2509106067-Mahdi Sarwan Abdullah-PT- 8/data.py
create mode 100644 post-test/post-test-apd-8/2509106067-Mahdi Sarwan Abdullah-PT- 8/login.py
create mode 100644 post-test/post-test-apd-8/2509106067-Mahdi Sarwan Abdullah-PT- 8/main.py
create mode 100644 post-test/post-test-apd-8/2509106067-Mahdi Sarwan Abdullah-PT-8.drawio
```

Gambar 5.2 GIT Commit

Fungsinya untuk melakukan commit atau konfirmasi perubahan yang terjadi pada repository kita.

5.3 GIT Push

```
PS D:\praktikum\praktikum_apd> git push -u origin main
Enumerating objects: 32, done.
Counting objects: 100% (32/32), done.
Delta compression using up to 16 threads
Compressing objects: 100% (29/29), done.
Writing objects: 100% (29/29), 4.74 MiB | 168.00 KiB/s, done.
Total 29 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/msarabd/praktikum_apd.git
  34cb9b9..c75e94f  main -> main
branch 'main' set up to track 'origin/main'.
```

Gambar 5.3 GIT Push

Kita bisa melakukan upload file yang tadinya hanya berada pada komputer kita ke Github.