

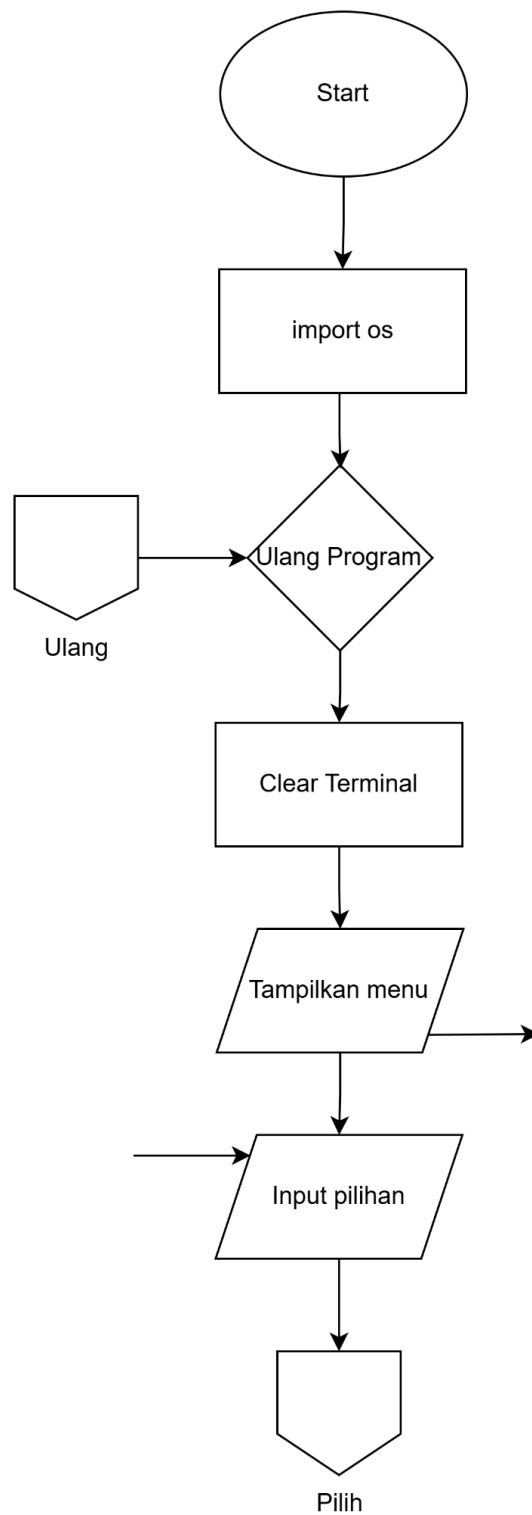
LAPORAN PRAKTIKUM
POSTTEST 5
ALGORITMA PEMROGRAMAN DASAR



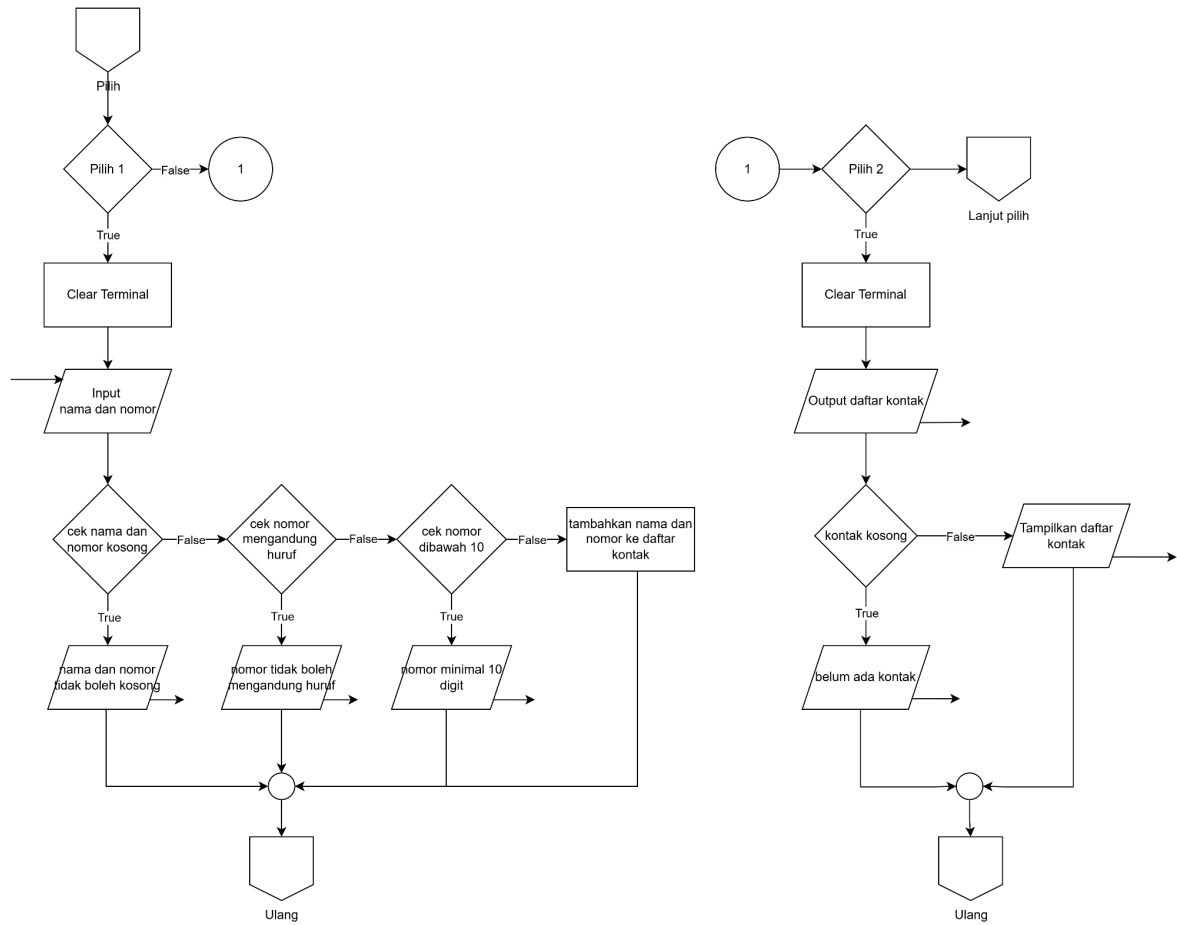
Disusun oleh:
Pirlo Syabila Hafuza (2509106008)
Kelas (A1 '25)

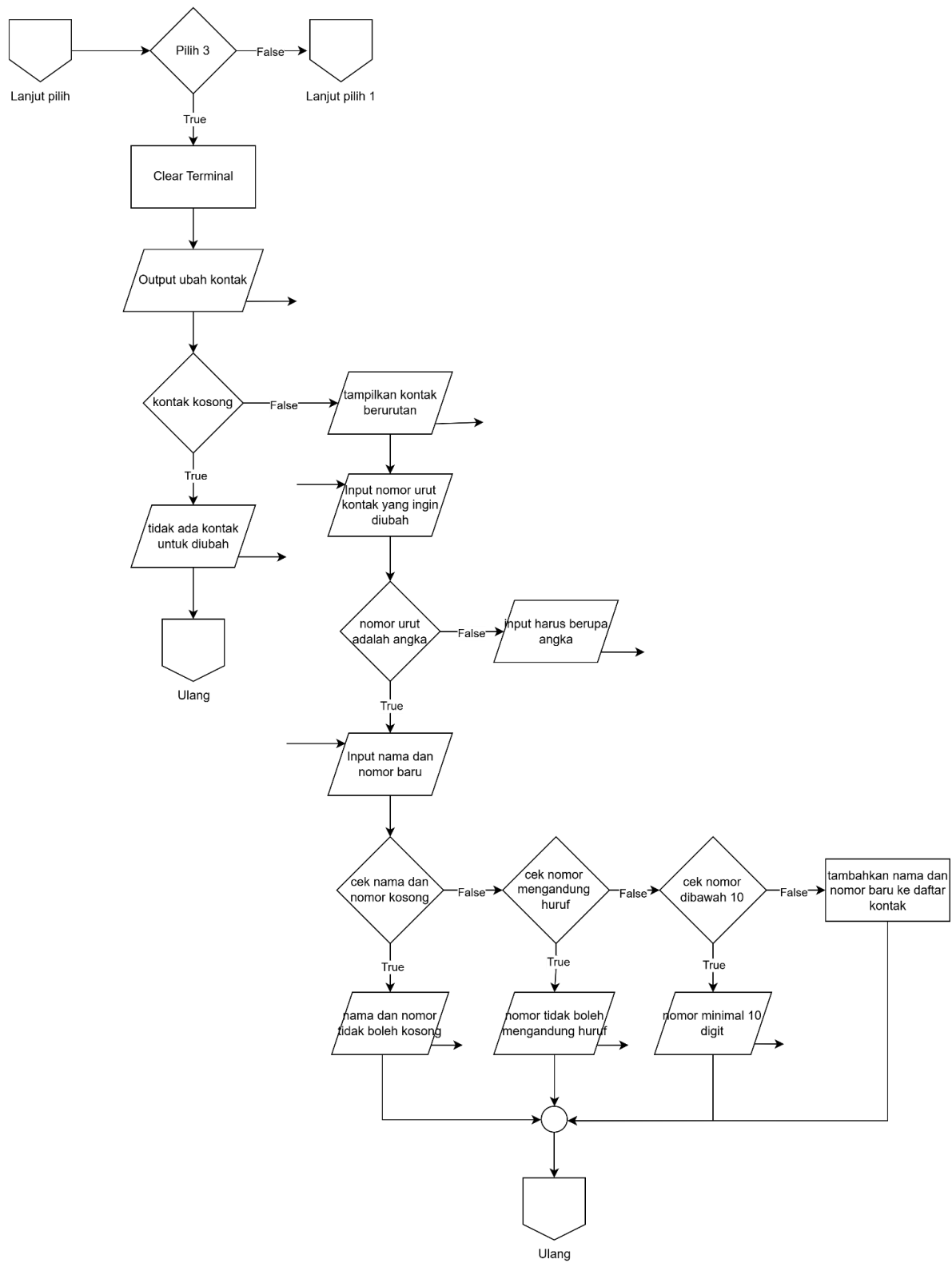
PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

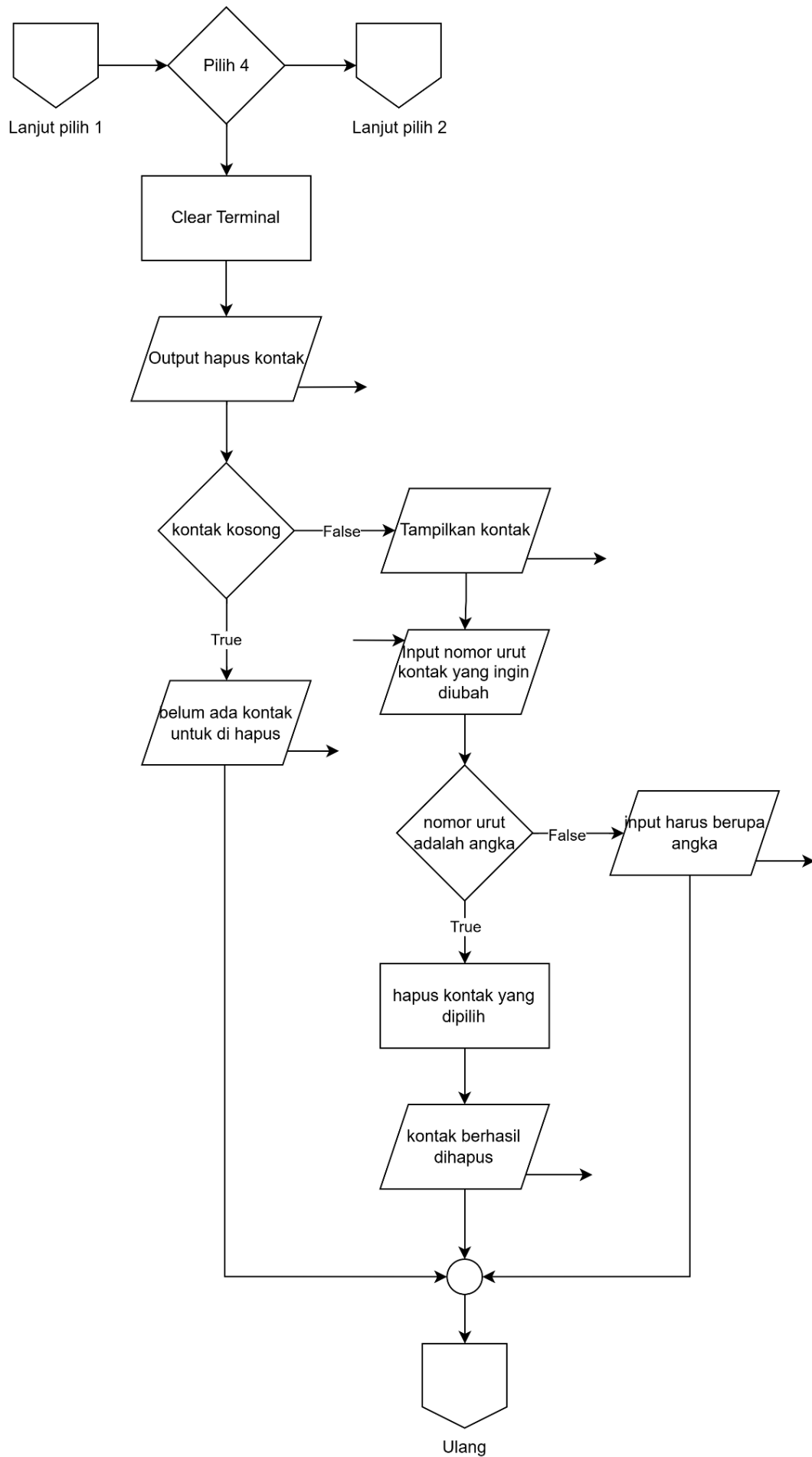
1. Flowchart

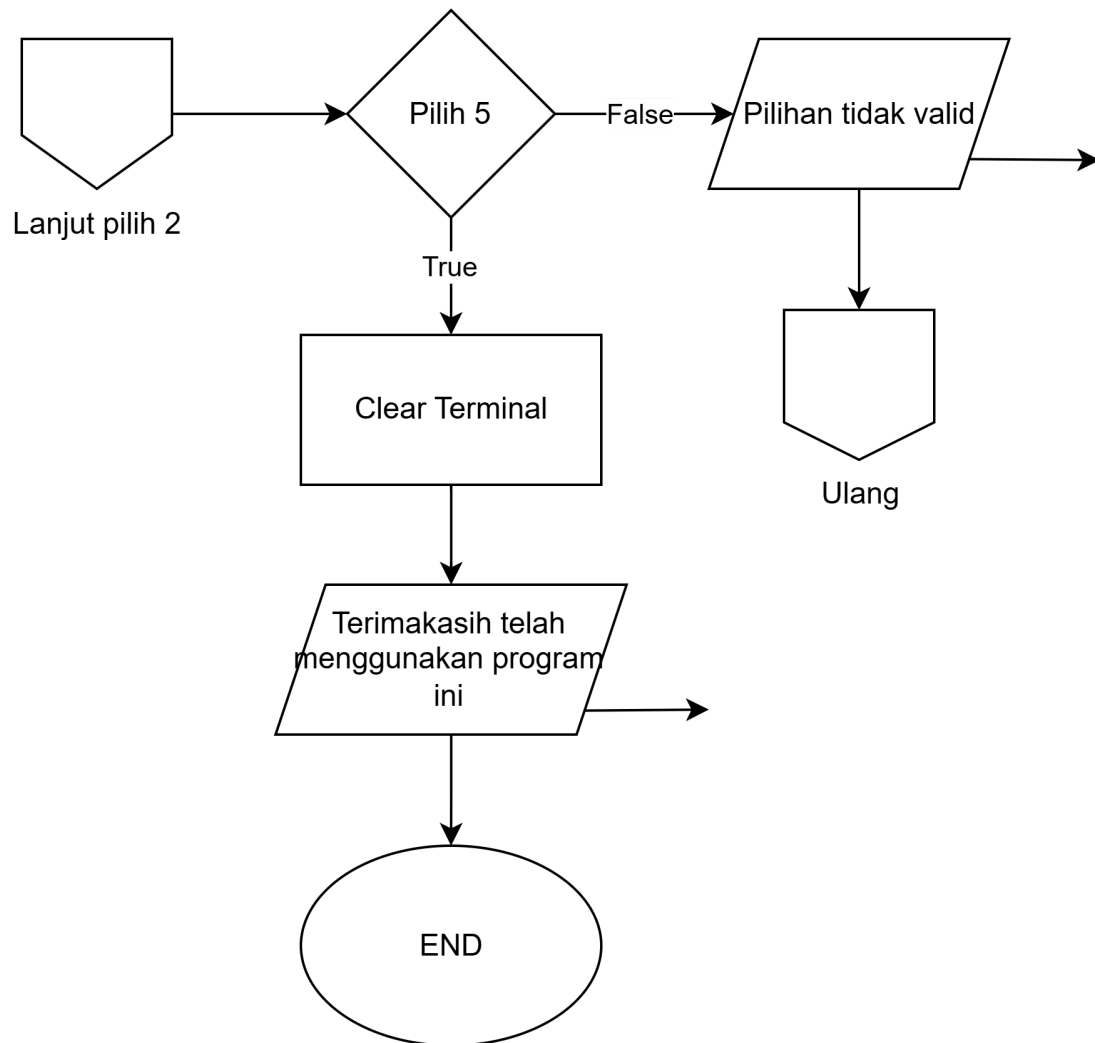


Gambar 1.1 Flowchart CRUD manajemen data kontak









2. Deskripsi Singkat Program

Program belanja ini dibuat untuk membantu pengguna mengelola kontak mereka. Dengan program ini pengguna dapat menambahkan kontak, melihat kontak yang sudah mereka dapat, mengubah data kontak jika pengguna mau dan menghapus data kontak yang sudah tidak dipakai lagi. Program ini juga menyediakan menu utama yang memandu pengguna dalam melakukan operasi dasar.

3. Source Code

A. Menu Utama

```
while True:
    os.system('cls' if os.name == 'nt' else 'clear')
    print("=== MANAJEMEN DATA KONTAK ===")
    print("1. Tambah Kontak")
    print("2. Lihat Semua Kontak")
    print("3. Ubah Kontak")
    print("4. Hapus Kontak")
    print("5. Keluar")
    pilihan = input("Pilih menu (1-5): ")
```

Di bagian ini sistem akan memberikan menu pilihan aksi atau fitur yang dapat di pilih oleh pengguna, pengguna dapat melakukan 5 aksi yang pertama menambah kontak, yang kedua melihat kontak yang ada atau sudah ditambahkan, yang ketiga mengubah data kontak yang diinginkan, keempat menghapus kontak dan yang terakhir berhenti atau keluar jika pengguna sudah tidak lagi ingin melakukan aksi atau sudah tidak ingin memakai program

B. Fitur Menambah Kontak

```
if pilihan == "1":
    os.system('cls' if os.name == 'nt' else 'clear')
    print("=== TAMBAH KONTAK ===")
    nama = input("Masukkan nama: ").strip()
    no_hp = input("Masukkan nomor HP: ").strip()

    if nama == "":
        print("Nama tidak boleh kosong!")
    elif no_hp == "":
        print("Nomor HP tidak boleh kosong!")
    elif not no_hp.isdigit():
        print("Nomor HP hanya boleh berisi angka (0-9)!")
    elif len(no_hp) < 10:
        print("Nomor HP minimal 10 digit!")
    else:
        kontak.append([nama, no_hp])
        print("Kontak berhasil ditambahkan!")
    input("Tekan Enter untuk kembali...")
```

Fitur ini digunakan saat pengguna memilih untuk menambahkan kontak pertama sistem akan membersihkan terminal lalu akan meminta pengguna untuk memasukkan nama dan nomor hp. Jika nama kosong maka sistem akan memberitahu bahwa nama tidak boleh kosong begitupun juga dengan nomor hp jika kosong sistem juga akan memberitahu bahwa nomor hp tidak boleh kosong, jika nama dan nomor hp tidak kosong selanjutnya sistem akan mengecek apakah nomor yang diinput menggunakan angka dan lebih dari 10 jika ya maka sistem akan menambahkan data input yang diberikan dan menyimpannya jika tidak sistem akan memberitahu bahwa nomor harus menggunakan digit (0-9) dan minimal 10 digit setelah selesai maka pengguna akan dibawa kembali ke menu fitur

C. Fitur Melihat Daftar Kontak

```
elif pilihan == "2":
    os.system('cls' if os.name == 'nt' else 'clear')
    print("=== DAFTAR KONTAK ===")
    if not kontak:
        print("Belum ada kontak.")
    else:
        for i in range(len(kontak)):
            print(f"{i + 1}. Nama: {kontak[i][0]}, No HP: {kontak[i][1]}")
        input("Tekan Enter untuk kembali ke menu...")
```

Fitur melihat daftar kontak jika pengguna memilih fitur ini maka sistem akan membersihkan terminal dan mengecek apakah ada kontak yang telah ditambahkan jika ada maka sistem menggunakan *for i in range* untuk mengakses nested list atau kontak yang telah ditambahkan atau di simpan jika ternyata tidak ada data kontak didalam list maka sistem akan memberi output belum ada kontak

D. Fitur Mengubah Data Kontak

```
elif pilihan == "3":
    os.system('cls' if os.name == 'nt' else 'clear')
    print("=== UBAH KONTAK ===")
    if len(kontak) == 0:
        print("Tidak ada kontak untuk diubah.")
        input("Tekan Enter untuk kembali...")
        continue

    for i in range(len(kontak)):
        print(f"{i + 1}. Nama: {kontak[i][0]}, No HP: {kontak[i][1]}")
    nomor = input("Pilih nomor kontak yang ingin diubah: ")

    if not nomor.isdigit():
        print("Input harus berupa angka!")
        input("Tekan Enter untuk kembali...")
        continue

    index = int(nomor) - 1
    if not (0 <= index < len(kontak)):
        print("Nomor kontak tidak valid!")
        input("Tekan Enter untuk kembali...")
        continue
```

```

nama = input("Masukkan nama baru: ").strip()
no_hp = input("Masukkan nomor HP baru: ").strip()

if nama == "":
    print("Nama tidak boleh kosong!")
elif no_hp == "":
    print("Nomor HP tidak boleh kosong!")
elif not no_hp.isdigit():
    print("Nomor HP hanya boleh berisi angka (0-9)!")
elif len(no_hp) < 10:
    print("Nomor HP minimal 10 digit!")
else:
    kontak[index][0] = nama
    kontak[index][1] = no_hp
    print("Kontak berhasil diubah!")
    input("Tekan Enter untuk kembali...")

```

Fitur ubah data kontak jika pengguna memilih fitur ini maka sistem akan membersihkan terminal lalu mengecek apakah ada daftar kontak yang dapat di ubah jika ada sistem akan memberikan list kontak yang tersedia pengguna akan diminta untuk memilih kontak mana yang ingin di ubah input harus berupa angka jika tidak maka sistem akan memberitahu bahwa input harus berupa angka dan program di ulang. Jika sudah memilih maka pengguna diminta untuk memasukkan nama dan nomor yang baru sistem akan mengecek ulang nama dan nomor jika kosong maka akan diberitahu sistem bahwa tidak boleh kosong dan mengubah kontak gagal program pun diulang, begitu pula jika nomor tidak berupa angka dan dibawah 10 digit sistem akan memberitahu dan program pun diulang kembali. Jika benar dan tidak ada kesalahan input maka kontak lama akan diubah menjadi kontak baru dan di simpan

E. Fitur Hapus Data Kontak

```
elif pilihan == "4":
    os.system('cls' if os.name == 'nt' else 'clear')
    print("=== HAPUS KONTAK ===")
    if not kontak:
        print("Tidak ada kontak untuk dihapus.")
        input("Tekan Enter untuk kembali...")
    else:
        for i in range(len(kontak)):
            print(f"{i + 1}. Nama: {kontak[i][0]}, No HP: {kontak[i][1]}")
        nomor = input("Pilih nomor kontak yang ingin dihapus: ")
        if nomor.isdigit():
            index = int(nomor) - 1
            if 0 <= index < len(kontak):
                del kontak[index]
                print("Kontak berhasil dihapus!")
            else:
                print("Nomor kontak tidak valid!")
        else:
            print("Input harus berupa angka!")
        input("Tekan Enter untuk kembali...")
```

Fitur hapus data kontak jika pengguna memilih fitur ini maka sistem akan membersihkan terminal lalu sistem akan mengecek apakah ada daftar atau data kontak yang dapat di hapus jika tidak sistem akan memberikan output bahwa tidak ada kontak yang dapat yang di hapus jika ada maka sistem akan mengakses list dan memberikan daftar kontak yang dapat dihapus pengguna akan diminta untuk memilih kontak mana yang ingin dihapus jika sudah memilih maka sistem akan langsung menghapus data kontak yang terpilih dan pengguna akan dibawa kembali ke menu utama

4. Hasil Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

=== MANAJEMEN DATA KONTAK ===
1. Tambah Kontak
2. Lihat Semua Kontak
3. Ubah Kontak
4. Hapus Kontak
5. Keluar
Pilih menu (1-5): █
```

Gambar 4.1 Output Menu Utama

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

=== TAMBAH KONTAK ===
Masukkan nama: Pirlo
Masukkan nomor HP: 082250557986
Kontak berhasil ditambahkan!
Tekan Enter untuk kembali...█
```

Gambar 4.2 Output Jika Pengguna Menambah Kontak

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

=== DAFTAR KONTAK ===
1. Nama: Yoga, No HP: 123456789101112
2. Nama: Pirlo, No HP: 082250557986
Tekan Enter untuk kembali ke menu...█
```

Gambar 4.3 Output Daftar Kontak

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

=== UBAH KONTAK ===
1. Nama: Yoga, No HP: 123456789101112
2. Nama: Pirlo, No HP: 082250557986
Pilih nomor kontak yang ingin diubah: 1
Masukkan nama baru: Nugrah
Masukkan nomor HP baru: 0876008765
Kontak berhasil diubah!
Tekan Enter untuk kembali...█
```

Gambar 4.4 Output Jika Pengguna Mengubah Data Kontak

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

=== HAPUS KONTAK ===
1. Nama: Nugrah, No HP: 0876008765
2. Nama: Pirlo, No HP: 082250557986
Pilih nomor kontak yang ingin dihapus: 1
Kontak berhasil dihapus!
Tekan Enter untuk kembali...█
```

Gambar 4.5 Output Jika Pengguna Menghapus Data Kontak

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

=== DAFTAR KONTAK ===
1. Nama: Pirlo, No HP: 082250557986
Tekan Enter untuk kembali ke menu...█
```

Gambar 4.6 Output Setelah Data Kontak Dihapus

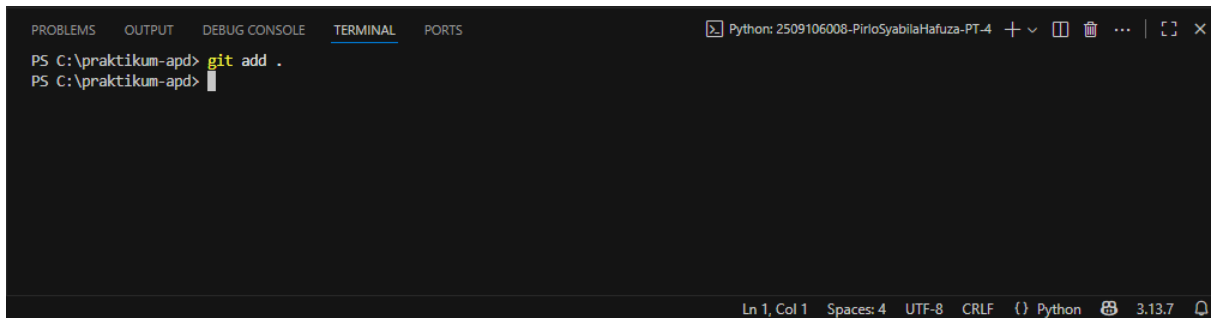
```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Terima kasih telah menggunakan program ini!
Tekan Enter untuk keluar...█
```

Gambar 4.7 Output Setelah Pengguna Selesai Menggunakan Program

5. Langkah-langkah GIT

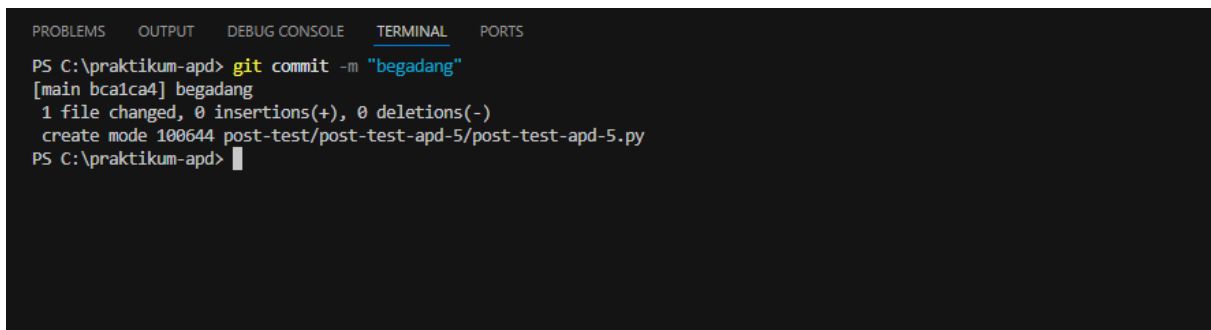
5.1 GIT Add



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Python: 2509106008-PirtoSyabilaHafuza-PT-4 + - ... | [ ] X
PS C:\praktikum-apd> git add .
PS C:\praktikum-apd>
```

Fungsi ini menandai atau menambahkan semua file yang ada di folder bisa juga menggunakan (git add namafile) jika ingin menambahkan file spesifik

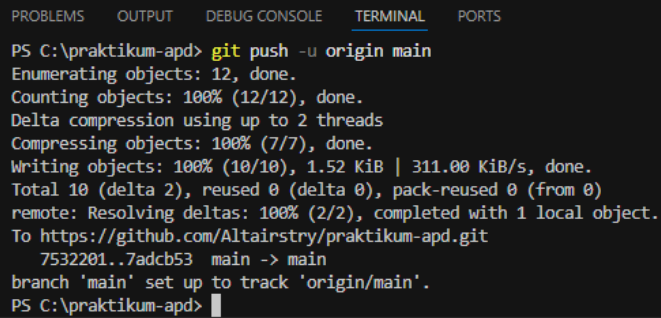
5.2 GIT Commit



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\praktikum-apd> git commit -m "begadang"
[main bca1ca4] begadang
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 post-test/post-test-apd-5/post-test-apd-5.py
PS C:\praktikum-apd>
```

Fungsi ini digunakan untuk *commit* atau menyimpan perubahan yang dilakukan untuk pesan commit sendiri tidak terpaksa apapun dan terserah ingin diisi apa saja asal bisa dipahami

5.3 GIT Push

A screenshot of a terminal window with a dark background. At the top, there are tabs labeled 'PROBLEMS', 'OUTPUT', 'DEBUG CONSOLE', 'TERMINAL' (which is selected and underlined), and 'PORTS'. The terminal shows the command 'git push -u origin main' being executed. The output includes progress bars for enumerating, counting, compressing, and writing objects, followed by a summary of the push operation and the URL of the repository.

```
PS C:\praktikum-apd> git push -u origin main
Enumerating objects: 12, done.
Counting objects: 100% (12/12), done.
Delta compression using up to 2 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (10/10), 1.52 KiB | 311.00 KiB/s, done.
Total 10 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 1 local object.
To https://github.com/Altairstry/praktikum-apd.git
   7532201..7adcb53  main -> main
branch 'main' set up to track 'origin/main'.
PS C:\praktikum-apd>
```

Fungsi ini digunakan untuk mengupload atau menambahkan file ke repository yang awalnya di folder komputer ke github langsung, jika push berhasil biasanya akan langsung dibawa ke halaman repository/github nya langsung