

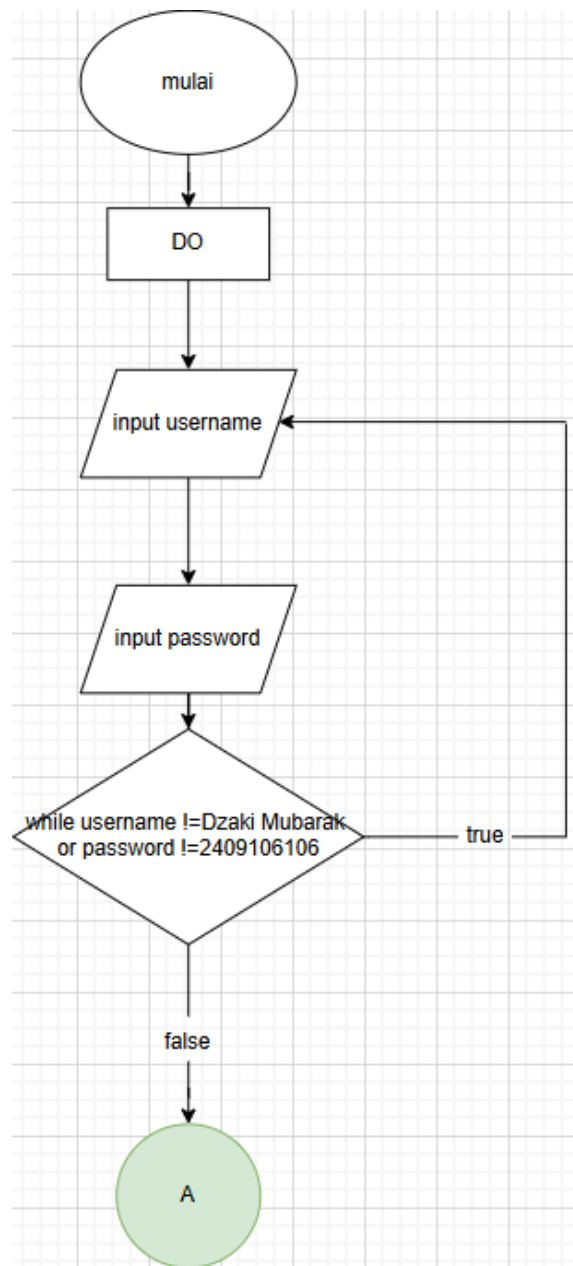
LAPORAN PRAKTIKUM
POSTTEST (3)
ALGORITMA PEMROGRAMAN LANJUT



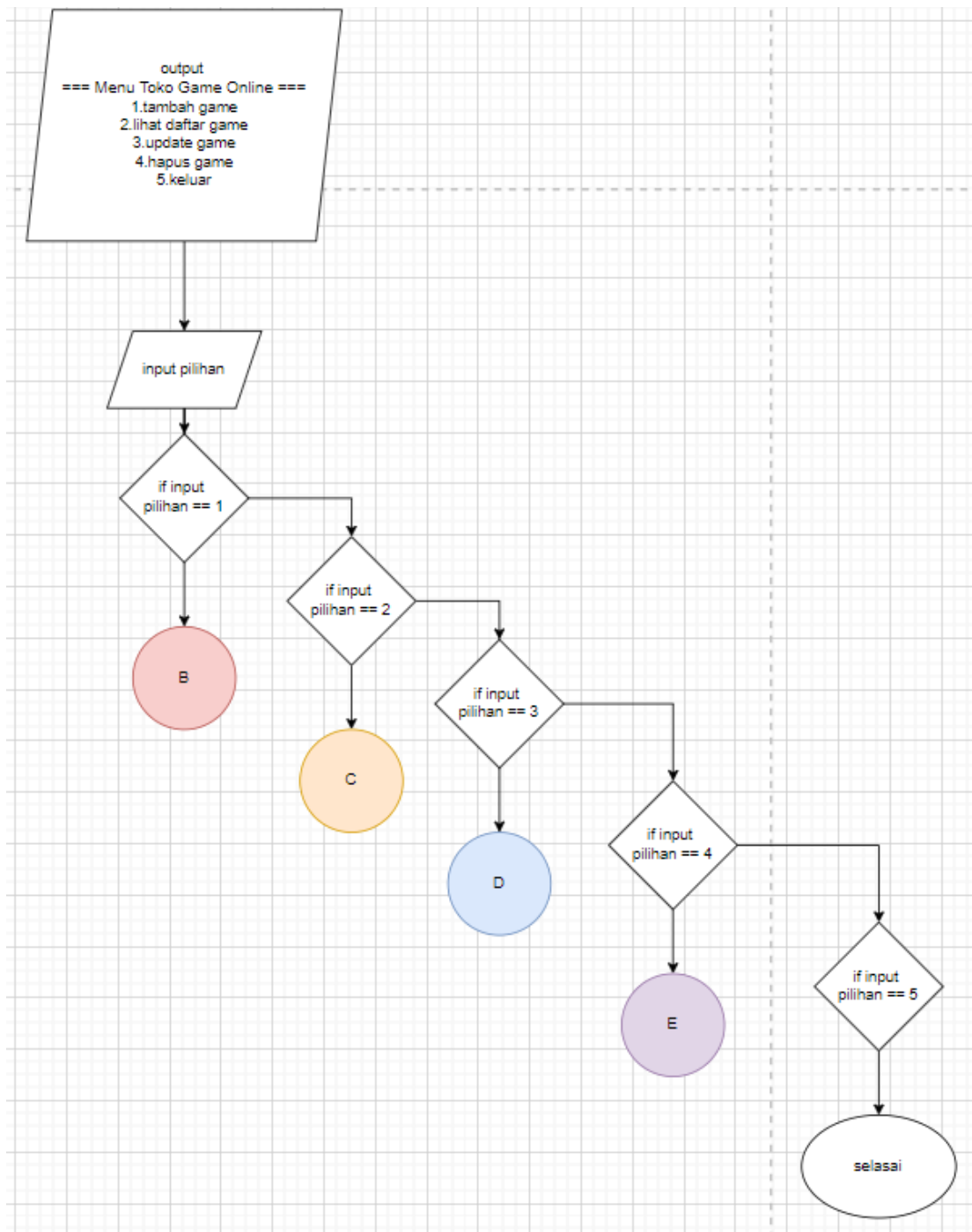
Disusun oleh:
Dzaki mubarak (2409106106)
Kelas (C '24)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

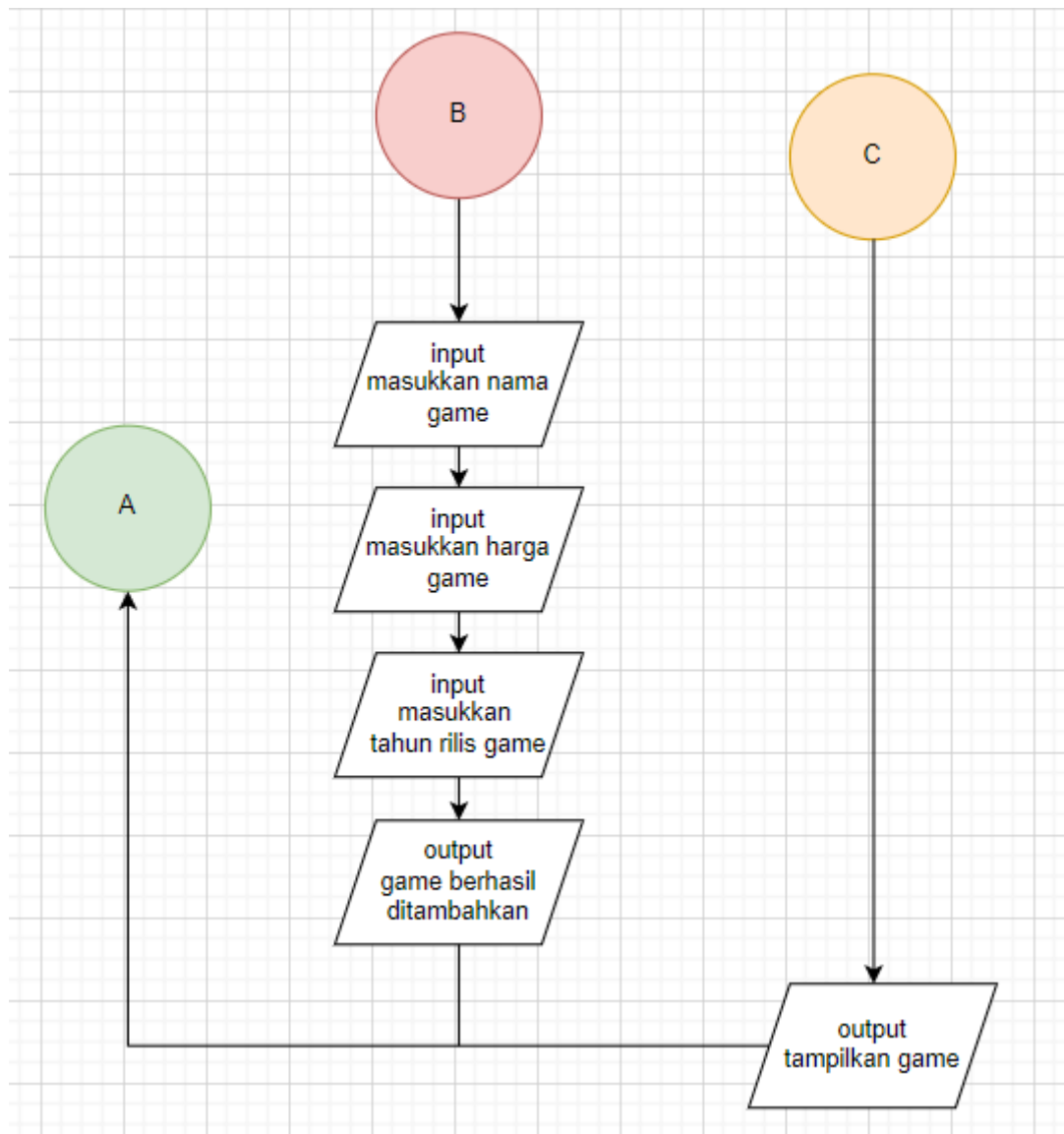
1. Flowchart



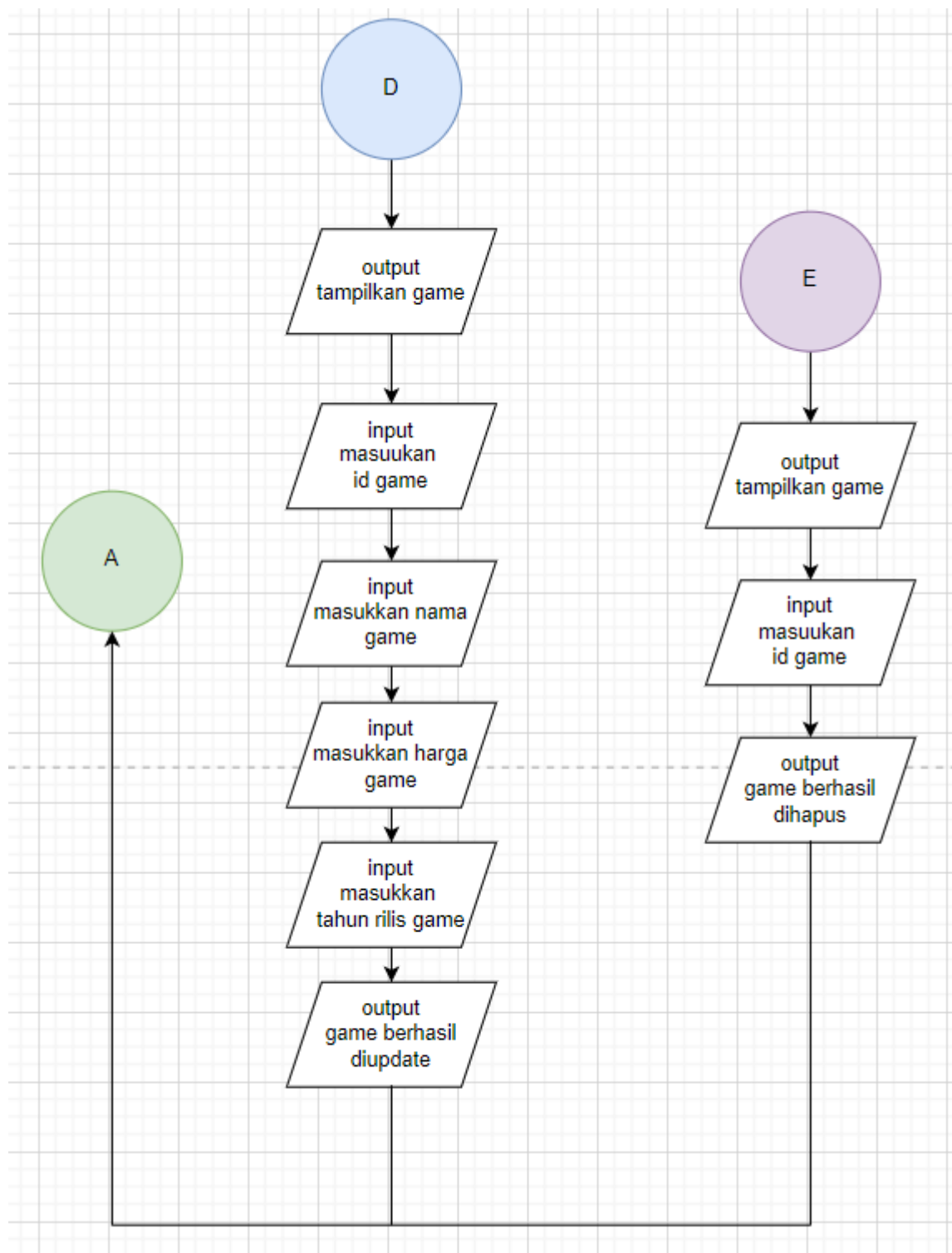
gambar 1.1 flowchart manajemen online shop



gambar 1.2 flowchart manajemen online shop



gambar 1.3 flowchart manajemen online shop



gambar 1.4 flowchart manajemen online shop

2. Analisis Program

2.1 Deskripsi Singkat Program

program ini bekerja menjadi manajemen sebuah online game dimana bisa mengatur online game shop sesuai kemaun pengguna. pengguna dapat menambah game, lihat game yang ada, update game yang ada, dan menghapus game yang ada .

3. Source Code

A.login

Fitur ini digunakan pengguna untuk login ke program menggunakan cara memasukkan username dan password pengguna, jika salah satu dari username dan password salah maka login akan gagal dan program akan meminta pengguna mengulangi login, pengguna hanya dapat login 3 kali saja jika lebih program akan berhenti.

Source Code:

```
bool login() {
    string nama, nim;
    int percobaan = 0;

    while (percobaan < 3) {
        cout << "Masukkan Nama: ";
        getline(cin, nama);
        cout << "Masukkan NIM: ";
        getline(cin, nim);

        if (nama == "user" && nim == "123456") {
            cout << "Login berhasil!\n";
            return true;
        } else {
            percobaan++;
            cout << "Login gagal. Percobaan tersisa: " << 3 - percobaan << endl;
        }
    }

    cout << "Anda telah melebihi batas percobaan login. Program berhenti.\n";
    return false;
}
```

B.menu

Fitur ini adalah dimana pengguna dapat untuk memilih salah satu opsi yang ada untuk pengguna mau memilih fitur yang mau digunakan.

Source Code:

```
int tampilkanMenu() {
```

```

int pilihan;
cout << "\n=== Menu Toko Game Online ===\n";
cout << "1. Tambah Game\n";
cout << "2. Lihat Daftar Game\n";
cout << "3. Update Game\n";
cout << "4. Hapus Game\n";
cout << "5. Keluar\n";
cout << "Pilih menu: ";
cin >> pilihan;
cin.ignore();
return pilihan;
}

```

C. tambah game

Fitur ini akan menambah game ke dalam program dengan mengirim game yang ditambah ke daftar game

Source Code:

```

bool tambahGame(Game daftarGame[], int &jumlahGame) {
    if (jumlahGame < 100) {
        cout << "\nMasukkan Nama Game: ";
        getline(cin, daftarGame[jumlahGame].nama);
        cout << "Masukkan Harga Game: ";
        cin >> daftarGame[jumlahGame].harga;
        cout << "Masukkan Tahun Rilis Game: ";
        cin >> daftarGame[jumlahGame].tahunRilis;
        jumlahGame++;
        cout << "Game berhasil ditambahkan!\n";
        return true;
    } else {
        cout << "Kapasitas game penuh!\n";
        return false;
    }
}

```


D. lihat daftar game

Fitur ini akan menunjukkan game yang ada di dalam program

Source Code:

```
int tampilkanDaftarGame(Game daftarGame[], int jumlahGame) {
    if (jumlahGame == 0) {
        cout << "\nTidak ada game yang tersedia.\n";
    } else {
        cout << "\n=== Daftar Game ===\n";
        for (int i = 0; i < jumlahGame; i++) {
            cout << "ID: " << i + 1 << endl;
            cout << "Nama Game: " << daftarGame[i].nama << endl;
            cout << "Harga: Rp" << daftarGame[i].harga << endl;
            cout << "Tahun Rilis: " << daftarGame[i].tahunRilis << endl;
            cout << "-----\n";
        }
    }
    return jumlahGame;
}
```

E. Update game

Fitur ini akan mengubah game yang sudah ada dan dapat mengubah nama, harga, dan tahun rilis

Source Code:

```
int jumlah = tampilkanDaftarGame(daftarGame, jumlahGame);
    if (jumlah > 0) {
        int id;
        cout << "Masukkan ID Game yang ingin diupdate: ";
        cin >> id;
        cin.ignore();

        if (id > 0 && id <= jumlahGame) {
            cout << "Masukkan Nama Game Baru: ";
            getline(cin, daftarGame[id - 1].nama);
            cout << "Masukkan Harga Game Baru: ";
            cin >> daftarGame[id - 1].harga;
            cout << "Masukkan Tahun Rilis Game Baru: ";
            cin >> daftarGame[id - 1].tahunRilis;
            cout << "Game berhasil diupdate!\n";
        } else {
            cout << "ID tidak valid!\n";
        }
    }
    break;
}
```

E. Hapus game

Fitur ini akan menghapus game yang ada

Source Code:

```
case 4: {
    int jumlah = tampilkanDaftarGame(daftarGame, jumlahGame);
    if (jumlah > 0) {
        int id;
        cout << "Masukkan ID Game yang ingin dihapus: ";
        cin >> id;

        if (id > 0 && id <= jumlahGame) {
            for (int i = id - 1; i < jumlahGame - 1; i++) {
                daftarGame[i] = daftarGame[i + 1];
            }
            jumlahGame--;
            cout << "Game berhasil dihapus!\n";
        } else {
            cout << "ID tidak valid!\n";
        }
    }
    break;
}
```

E. Keluar

Fitur ini akan menghapus game yang ada

Source Code:

```
case 5:
    cout << "Terima kasih telah menggunakan program ini!\n";
    return 0;
```

4. Uji Coba dan Hasil Output

4.1 Hasil Output

1. login gagal

```
Masukkan username: dzaki  
Masukkan password: 123  
Login gagal. Percobaan tersisa: 2
```

Gambar 4.2.1 login gagal (masih ada percobaan)

```
Masukkan username: dzaki  
Masukkan password: 123  
Login gagal. Percobaan tersisa: 0  
Anda telah melebihi batas percobaan login. Program berhenti.
```

Gambar 4.2.2 login gagal (percobaan habis)

2. login berhasil

```
Masukkan username: Dzaki Mubarak  
Masukkan password: 2409106106  
Login berhasil!
```

Gambar 4.2.3 login berhasil

3. menu

```
=== Menu Toko Game Online ===  
1. Tambah Game  
2. Lihat Daftar Game  
3. Update Game  
4. Hapus Game  
5. Keluar  
Pilih menu: █
```

Gambar 4.2.4 menu program

4.tambah game

```
Masukkan Nama Game: terraria
Masukkan Harga Game: 98000
Masukkan Tahun Rilis Game: 2010
Game berhasil ditambahkan!
```

Gambar 4.2.5 tambah game

5.lihat daftar game

```
=== Daftar Game ===
ID: 1
Nama Game: terraria
Harga: Rp98000
Tahun Rilis: 2010
-----
```

Gambar 4.2.6 jika ada game

```
Tidak ada game yang tersedia.
```

Gambar 4.2.7 jika tidak ada game

6.update game

```
=== Daftar Game ===
ID: 1
Nama Game: terraria
Harga: Rp98000
Tahun Rilis: 2010
-----
Masukkan ID Game yang ingin diupdate: 1
Masukkan Nama Game Baru: minecraft
Masukkan Harga Game Baru: 100000
Masukkan Tahun Rilis Game Baru: 2011
Game berhasil diupdate!
```

Gambar 4.2.8 jika ada game

```
Tidak ada game yang tersedia untuk diupdate.
```

Gambar 4.2.9 jika tidak ada game

```
=== Daftar Game ===  
ID: 1  
Nama Game: minecraft  
Harga: Rp100000  
Tahun Rilis: 2011  
-----
```

Gambar 4.2.10 game diupdate

7.hapus game

```
=== Daftar Game ===  
ID: 1  
Nama Game: minecraft  
Harga: Rp100000  
Tahun Rilis: 2011  
-----  
Masukkan ID Game yang ingin dihapus: 1  
Game berhasil dihapus!
```

Gambar 4.2.11 game berhasil dihapus

```
Tidak ada game yang tersedia untuk dihapus.
```

Gambar 4.2.12 tidak ada game

8.keluar

```
Terima kasih telah menggunakan program ini!
```

Gambar 4.2.13 keluar program

9.input tidak valid

```
Pilih menu: 6  
Pilihan tidak valid. Silakan coba lagi.
```

Gambar 4.2.14 input tidak valid/pilihan salah

5.git

```
ASUS@LAPTOP-2J48I9FG MINGW64 ~/Documents/pratikum-apl (main)
$ git init
Reinitialized existing Git repository in C:/Users/ASUS/Documents/pratikum-apl/.git/

ASUS@LAPTOP-2J48I9FG MINGW64 ~/Documents/pratikum-apl (main)
$ git add .

ASUS@LAPTOP-2J48I9FG MINGW64 ~/Documents/pratikum-apl (main)
$ git commit -m "finish pos test 2"
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

ASUS@LAPTOP-2J48I9FG MINGW64 ~/Documents/pratikum-apl (main)
$ git remote add origin https://github.com/Dzaki106/pratikum-apl.git
error: remote origin already exists.

ASUS@LAPTOP-2J48I9FG MINGW64 ~/Documents/pratikum-apl (main)
$ git push -u origin main
branch 'main' set up to track 'origin/main'.
Everything up-to-date

ASUS@LAPTOP-2J48I9FG MINGW64 ~/Documents/pratikum-apl (main)
$ |
```

1.git init

Perintah ini digunakan untuk menginisialisasi repository Git baru di direktori lokal

2.git add

Perintah ini digunakan untuk menambahkan file ke staging area agar bisa di komit.

3. git commit -m "Finish Post Test 2"

Perintah untuk menyimpan perubahan dalam repository dengan commit dan pesan “finish post test 2

4. git remote add origin https://github.com/Dzaki106/pratikum-apl.git

Perintah digunakan untuk menambahkan repository dengan nama origin

5.git push -u origin main

Perintah ini digunakan untuk mengirim perubahan yang telah di-commit dari repository lokal ke remote repository.