

LAPORAN PRAKTIKUM
POSTTEST 3
ALGORITMA PEMROGRAMAN LANJUT



Disusun oleh:

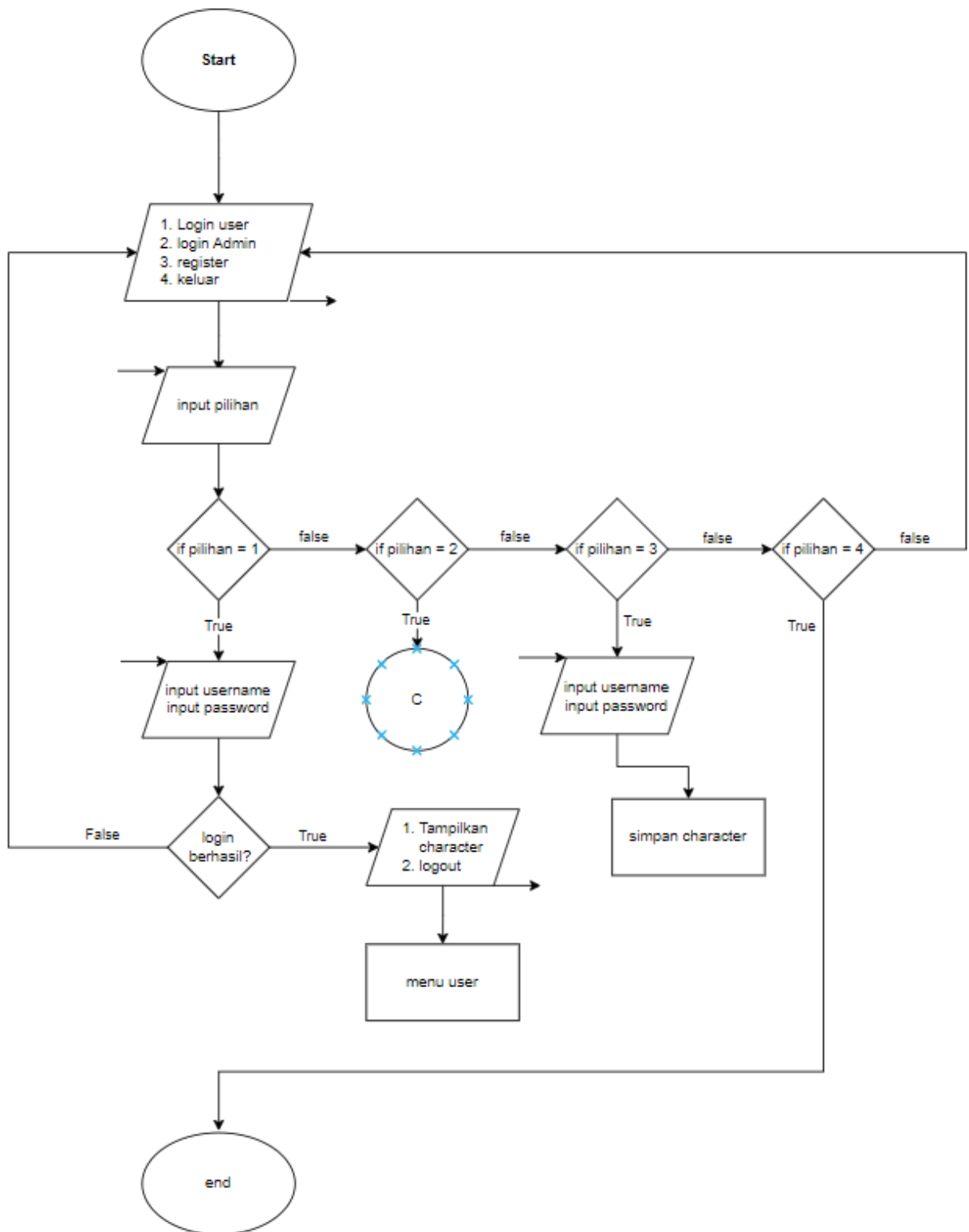
Nama : 2409106039 - Muhammad Rizal Alfath

Kelas : Informatika A2'24

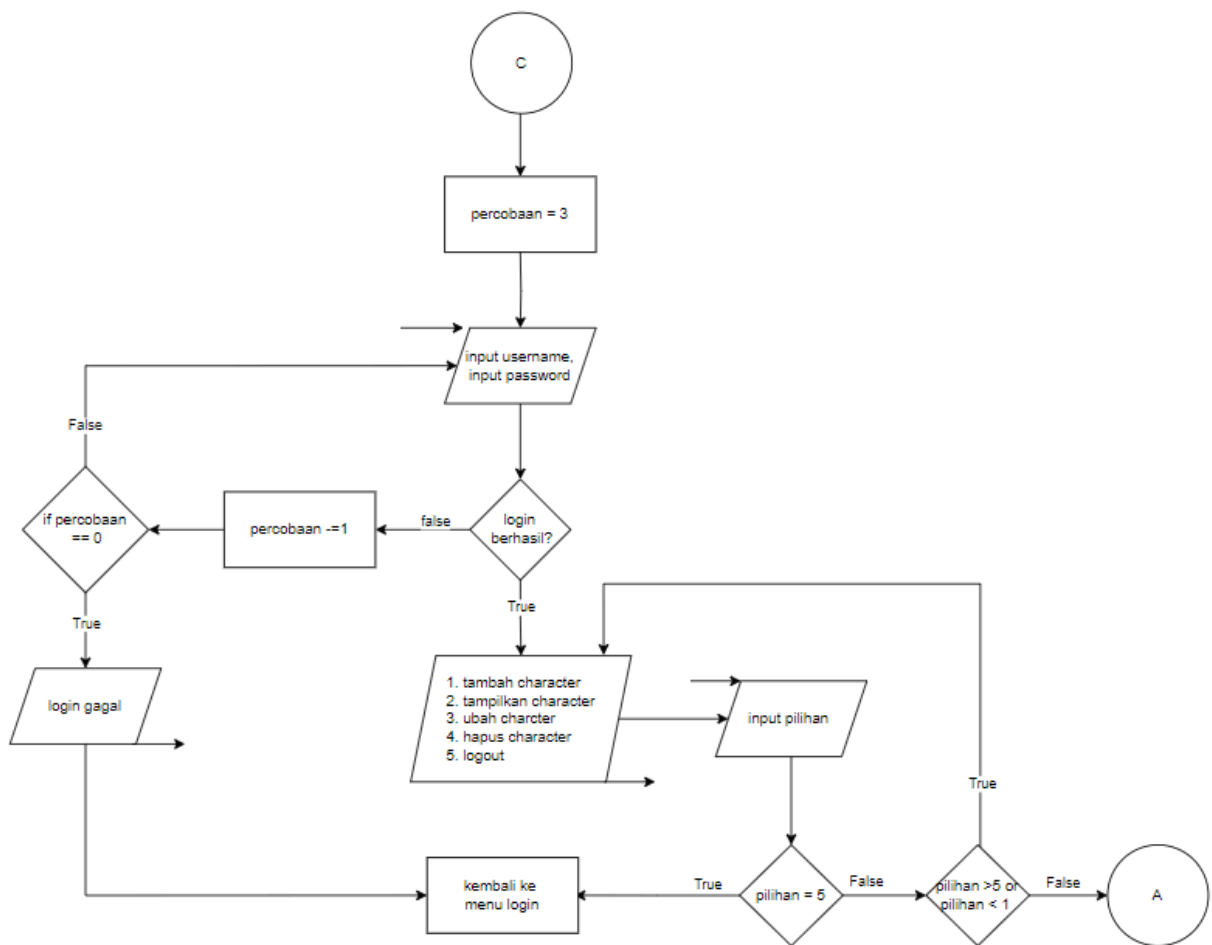
PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA

2025

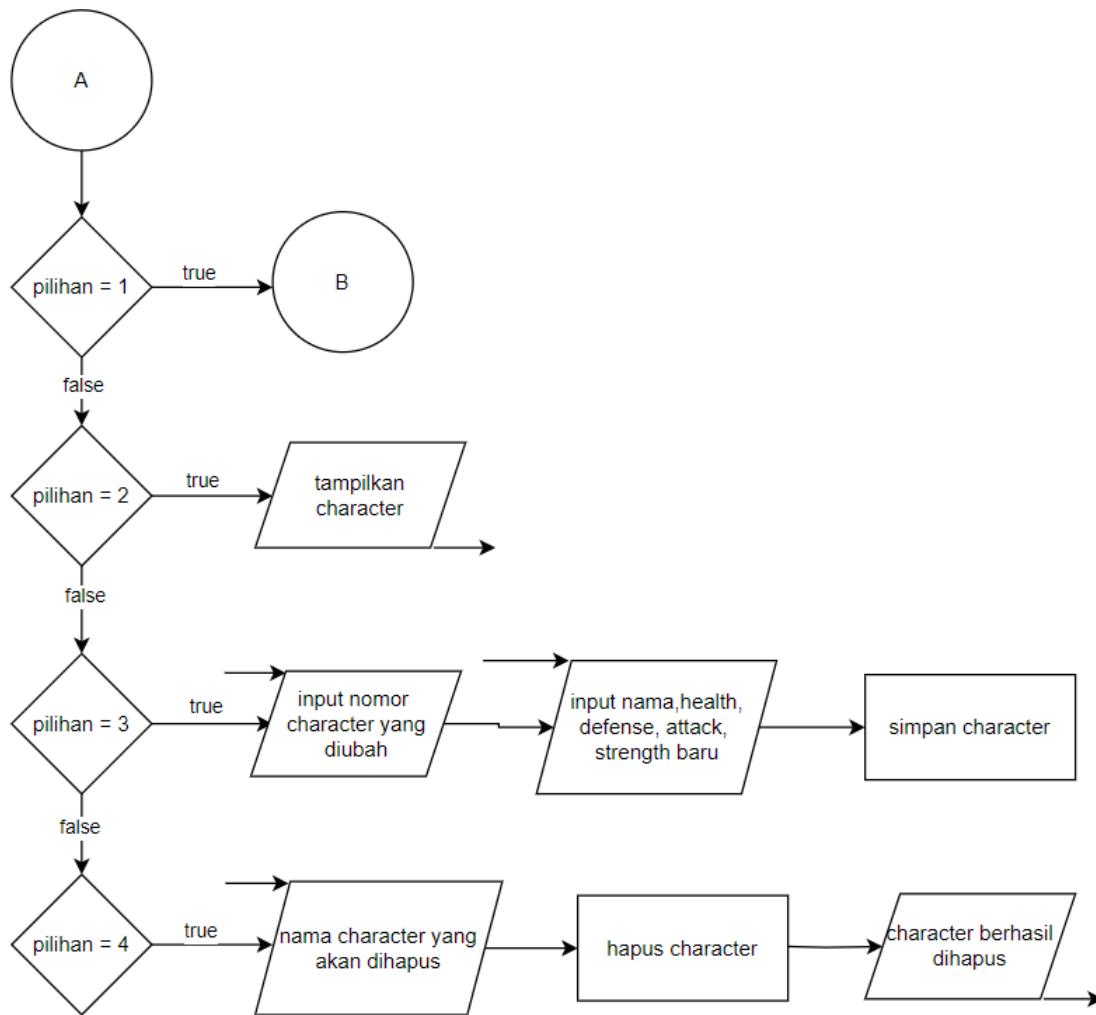
1. Flowchart



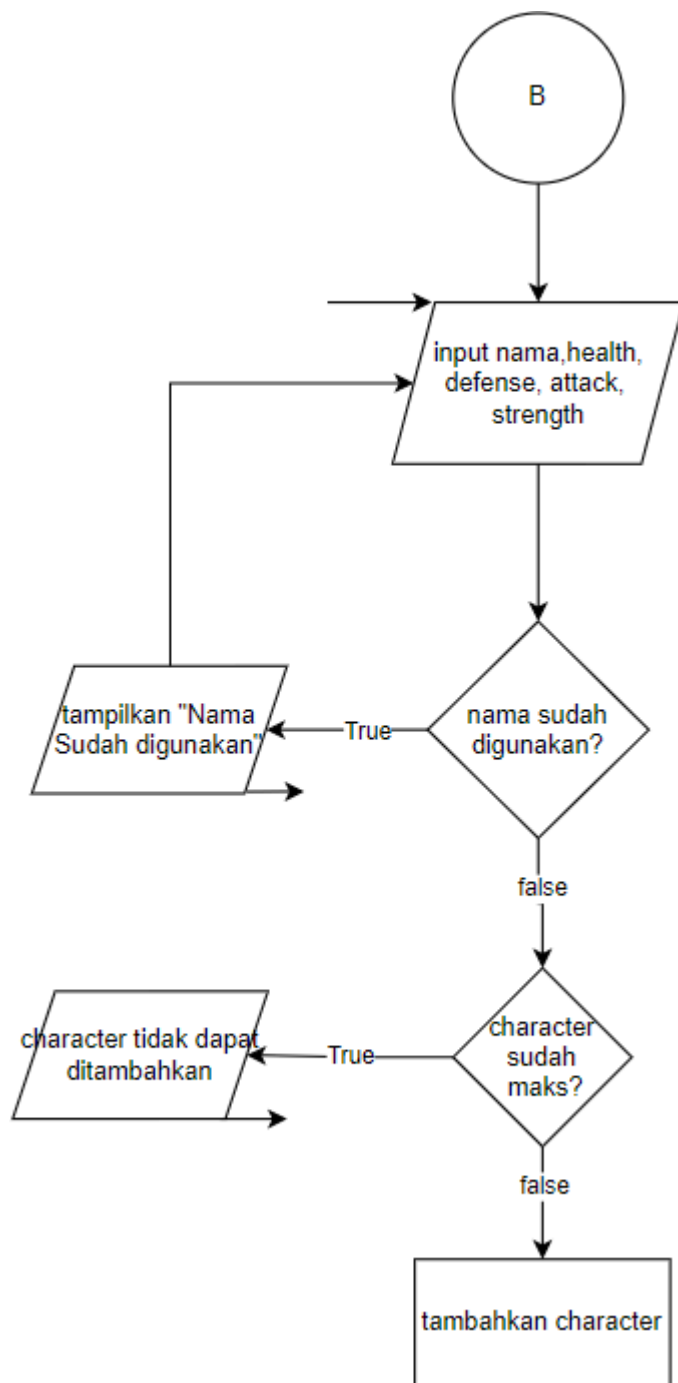
Gambar 1.1, contoh 1 FLOWchart



Gambar 1.1, contoh 2 FLOWchart



Gambar 1.1, contoh 1 FFlowchart



Gambar 1.1, contoh 1 FLOWchart

2. Analisis Program

Program ini bertujuan untuk menyimpan dan mengelola data character dengan mudah dan lebih terorganisir, memungkinkan user untuk menambahkan, mengedit / mengubah, melihat dan menghapus character. Program memastikan bahwa tidak ada character dengan nama yang sama, sehingga data tetap konsisten dan terorganisir. Program ini juga diberikan sistem login sederhana yang memberikan keamanan, dan dapat mengakses data karakter

Program ini merupakan sistem CRUD (create, read, update, delete) untuk mengelola data character. berikut adalah alur kerja program :

1. Melakukan login Admin

Program meminta user untuk memasukan input username dan password dengan 3x kesempatan mengulang, jika user memasukan username ("rizal") dan password ("039") maka user bisa masuk ke menu admin dari programnya, tetapi jika user salah memasukan usernam dan password 3x maka program tidak akan berjalan.

2. Register

Program akan meminta untuk melakukan register sebelum masuk ke login user, dengan menginput username dan password

3. Login user

Masuk dengan memasukan username dan password yang dibuat melewati register terlebih dahulu, pastikan username dan password sama dengan yang telah dibuat

4. Menu Utama

program menampilkan menu utama dengan 5 pilihan:

- Tambah Karakter
- Tampilkan Karakter
- Ubah Karakter
- Hapus Karakter
- Keluar dari program

5. Fitur CRUD Program

- Tambah Character (create)
 - o Program memiliki batas maksimum dalam membuat character yaitu hanya 10 character yang dapat dibuat
 - o User akan memasukan nama character, tetapi tidak boleh sama dengan nama character yang sudah ada, memasukan nilai dari attack, health, strength, dan defense character
- Tampilkan Character (read)
 - o Jika ada character yang tersimpan maka program akan menampilkan character yang ada dengan menampilkan informasi nama, attack, defense, health, strength character
- Ubah character (update)
 - o User dapat memilih character mana yang ingin diubah, character dapat diubah nama ataupun statistiknya menjadi nilai baru
 - o Program memastikan tidak ada nama yang sama / kembar saat memasukan nama baru untuk character
- Hapus Character (delete)
 - o User memilih nomor character yang ingin dihapus, program akan menghapus character tersebut dan menggeser semua data setelahnya untuk mengisi celah dari character yang sudah dihapus, jumlah character akan berkurang 1
- Keluar dari program
 - o Mengakhiri program / keluar dari program

3. Source Code

1. Fitur Login admin \

Fitur ini digunakan untuk melakukan login sebagai admin yang dimana semua crud dipegang oleh admin, gunakan username = "rizal" dan password = "039" untuk masuk ke menu admin

```
int percobaan = 3;
bool login_admin = false;

while (percobaan > 0 && !login_admin) {
    string username, password;
    cout << "\n=== LOGIN ADMIN ===" << endl;
    cout << "Percobaan tersisa: " << percobaan << endl;
    cout << "Username: ";
    cin >> username;
    cout << "Password: ";
    cin >> password;

    bool found = false;
    for (int i = 0; i < jumlahUser; i++) {
        if (users[i].username == username && users[i].password
== password) {
            if (username == "rizal") {
                found = true;
                login_admin = true;
                logged_in = true;
                current_user = username;
                is_admin = true;
                cout << "Login admin berhasil!" << endl;
                break;
            }
        }
    }
}
```

2. Fitur Tambah character

fitur ini digunakan untuk membuat character baru dengan maksimal character adalah 10

```
    karakter[jumlahKarakter].nama = namaBaru;
    cout << "Masukkan Attack: ";
    cin >> karakter[jumlahKarakter].att_str.attack;
    cout << "Masukkan Strength: ";
    cin >> karakter[jumlahKarakter].att_str.strength;
    cout << "Masukkan Defense: ";
    cin >> karakter[jumlahKarakter].defense;
    cout << "Masukkan Health: ";
    cin >> karakter[jumlahKarakter].health;
```



```

        jumlahKarakter++;
        cout << endl << "Karakter berhasil ditambahkan!\n";
        cout << "\nTekan Enter untuk kembali ke menu...";
        cin.ignore(1000, '\n');
        cin.get();
    } else {
        cout << "Data penuh! Tidak bisa menambahkan
karakter baru.\n";
    }
}

```

3. Fitur tampilkan character

menampilkan semua character yang sudah dibuat beserta statistiknya

```

        for (int i = 0; i < jumlahKarakter; i++) {
            cout << left << setw(5) << i+1
                << setw(15) << karakter[i].nama
                << setw(10) << karakter[i].att_str.attack
                << setw(10) <<
karakter[i].att_str.strength
                << setw(10) << karakter[i].defense
                << setw(10) << karakter[i].health <<
endl;
        }
    }

```

4. Mengubah character

Fitur ini digunakan untuk mengganti nama dan statistik character

```

        karakter[index].nama = namaBaru;
        cout << "Masukkan Attack Baru: ";
        cin >> karakter[index].att_str.attack;
        cout << "Masukkan Strength Baru: ";
        cin >> karakter[index].att_str.strength;
        cout << "Masukkan Defense Baru: ";
        cin >> karakter[index].defense;
        cout << "Masukkan Health Baru: ";
        cin >> karakter[index].health;

        cout << endl << "Data karakter berhasil
diperbarui!\n";

        cout << "\nTekan Enter untuk kembali ke menu...";
        cin.ignore(1000, '\n');
        cin.get();
    } else {
        cout << "Nomor karakter tidak valid!\n";
    }
}

```

5. Fitur hapus character

fitur ini digunakan untuk menghapus character

```
int index;

cout << "Masukkan nomor karakter yang ingin dihapus: ";

cin >> index;
index--;

if (index >= 0 && index < jumlahKarakter) {
    for (int i = index; i < jumlahKarakter - 1; i++) {
        karakter[i] = karakter[i + 1];
    }
    jumlahKarakter--;
    cout << endl << "Karakter berhasil dihapus!\n";
    cout << "\nTekan Enter untuk kembali ke menu...";

    cin.ignore(1000, '\n');
    cin.get();
} else {
    cout << endl << "Nomor karakter tidak valid!\n";
}
```

6. Fitur login user

Fitur ini digunakan untuk login sebagai user yang dapat melihat character, sebelum itu buat akun terlebih dahulu dengan cara register

```
if (menu_utama == 1) { // Login User
    string username, password;
    cout << "\n=== LOGIN USER ===" << endl;
    cout << "Username: ";
    cin >> username;
    cout << "Password: ";
    cin >> password;

    bool found = false;
    for (int i = 0; i < jumlahUser; i++) {
        if (users[i].username == username && users[i].password ==
password) {
            found = true;
            logged_in = true;
            current_user = username;
            is_admin = false;
            cout << "Login berhasil!" << endl;
            break;
        }
    }
}
```

7. Fitur register

Fitur ini digunakan untuk membuat akun untuk dapat login ke menu user

```
        for (int i = 0; i < jumlahUser; i++) {
            if (users[i].username == username) {
                username_exists = true;
                break;
            }
        }

        if (username_exists) {
            cout << "Username sudah digunakan!" << endl;
        } else {
            cout << "Password: ";
            cin >> password;

            users[jumlahUser].username = username;
            users[jumlahUser].password = password;
            jumlahUser++;

            cout << "Registrasi berhasil! Silakan login." << endl;
        }
    }
    else if (menu_utama == 4) {
        cout << "Keluar dari program." << endl;
        return 0;
    }
    else {
        cout << "Pilihan tidak valid!" << endl;
    }
}
```

4. Uji Coba dan Hasil Output

```
Masukan Username :rizal
Masukan Password :039

=== CRUD Statistik Karakter ===
1. Tambah Karakter
2. Tampilkan Karakter
3. Ubah Karakter
4. Hapus Karakter
5. Keluar

Pilih menu: █
```

4.2 contoh. 1, Login admin Berhasil

```
Masukan Username :udin
Masukan Password :w

Maaf Username atau Password salah, coba lagi
percobaan : 2

Masukan Username :404
Masukan Password :not found

Maaf Username atau Password salah, coba lagi
percobaan : 1

Masukan Username :Masukan Password :w

Maaf Username atau Password salah, coba lagi
percobaan : 0

maaf anda sudah tidak dapat login, percobaan habis
```

4.2 contoh. 2, Login Gagal

```

=== CRUD Statistik Karakter ===
1. Tambah Karakter
2. Tampilkan Karakter
3. Ubah Karakter
4. Hapus Karakter
5. Keluar

Pilih menu: 2

===== Daftar Karakter =====

No    Nama           Attack    Strength    Defense    Health
-----
1     kratos          9999      9999        8532       6000
-----

Tekan Enter untuk kembali ke menu...

```

4.2 contoh. 4, Tampilkan character - menu 2

```

=== CRUD Statistik Karakter ===
1. Tambah Karakter
2. Tampilkan Karakter
3. Ubah Karakter
4. Hapus Karakter
5. Keluar

Pilih menu: 3

===== Daftar Karakter =====

No    Nama           Attack    Strength    Defense    Health
-----
1     kratos          9999      9999        8532       6000
-----

Masukkan nomor karakter yang ingin diubah: 1
Masukkan Nama Baru: mario
Masukkan Attack Baru: 23
Masukkan Strength Baru: 90
Masukkan Defense Baru: 44
Masukkan Health Baru: 10

Data karakter berhasil diperbarui!

Tekan Enter untuk kembali ke menu...

```

4.2 contoh. 5, Ubah character – menu 3

```

=== CRUD Statistik Karakter ===
1. Tambah Karakter
2. Tampilkan Karakter
3. Ubah Karakter
4. Hapus Karakter
5. Keluar

Pilih menu: 4

===== Daftar Karakter =====

No    Nama      Attack  Strength  Defense  Health
-----
1     mario      23      90        44       10
-----

Masukkan nomor karakter yang ingin dihapus: 1

Karakter berhasil dihapus!

Tekan Enter untuk kembali ke menu...

```

4.2 contoh. 6, Hapus character – menu 4

```

=== CRUD Statistik Karakter ===
1. Tambah Karakter
2. Tampilkan Karakter
3. Ubah Karakter
4. Hapus Karakter
5. Keluar

Pilih menu: 5
Keluar dari program.

```

4.2 contoh. 7, Keluar – menu 5

```
=== MENU UTAMA ===
1. Login User
2. Login Admin
3. Register User
4. Keluar
Pilihan: 3

=== REGISTER USER ===
Username: guy
Password: sensei
Registrasi berhasil! Silakan login.
```

4.2 contoh. 8, Register

```
=== MENU UTAMA ===
1. Login User
2. Login Admin
3. Register User
4. Keluar
Pilihan: 1

=== LOGIN USER ===
Username: guy
Password: sensei
Login berhasil!

=== MENU USER (guy) ===
1. Lihat Karakter
2. Logout
Pilihan: █
```

4.2 contoh. 8, Login user

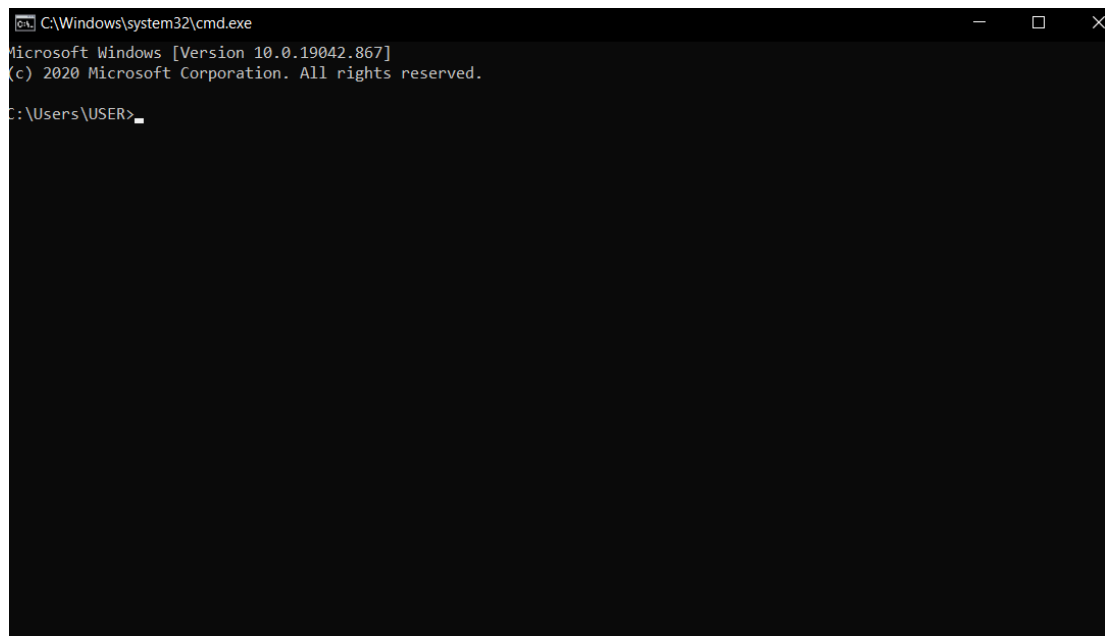
5. Langkah-Langkah Git pada VSCode

5.1 Git init

git init adalah perintah untuk menginisialisasi repository Git di dalam sebuah folder proyek.

1. Buka Terminal

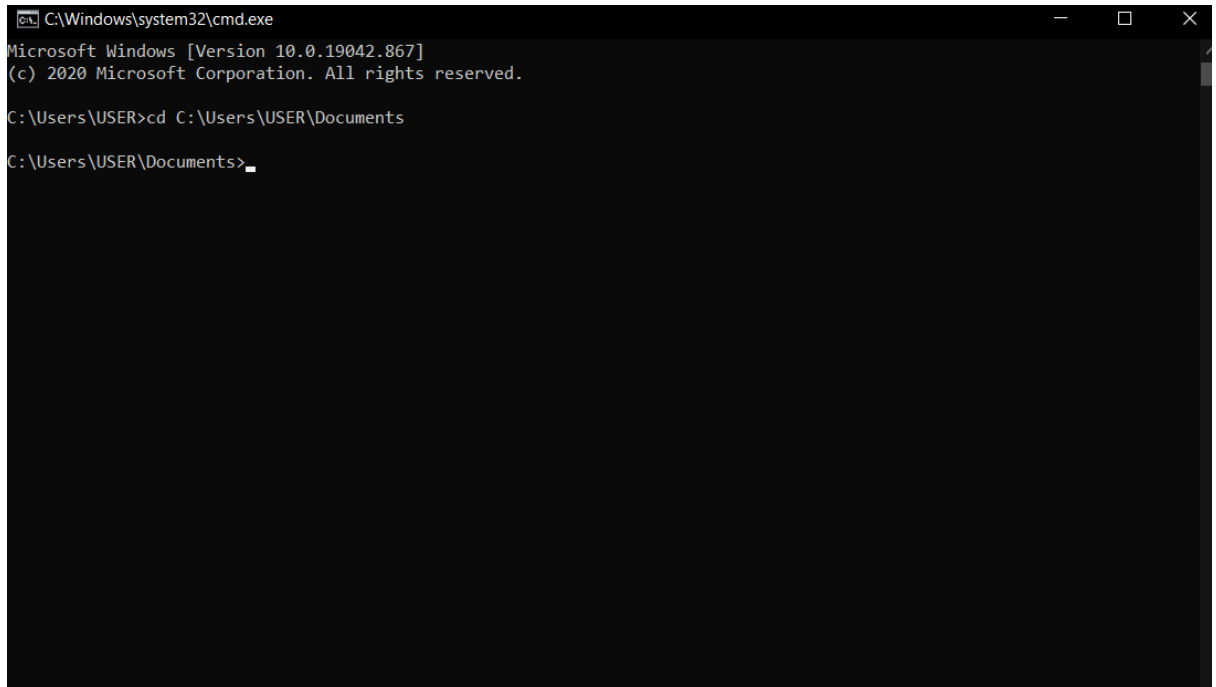
Jika di Windows, bisa menggunakan Git Bash atau Command Prompt (cmd)



Gambar 5.1, 1. Buka Terminal

2. Pindah ke directory folder

Masukan path directory folder praktikum-apl dengan mengetikan perintah di command prompt

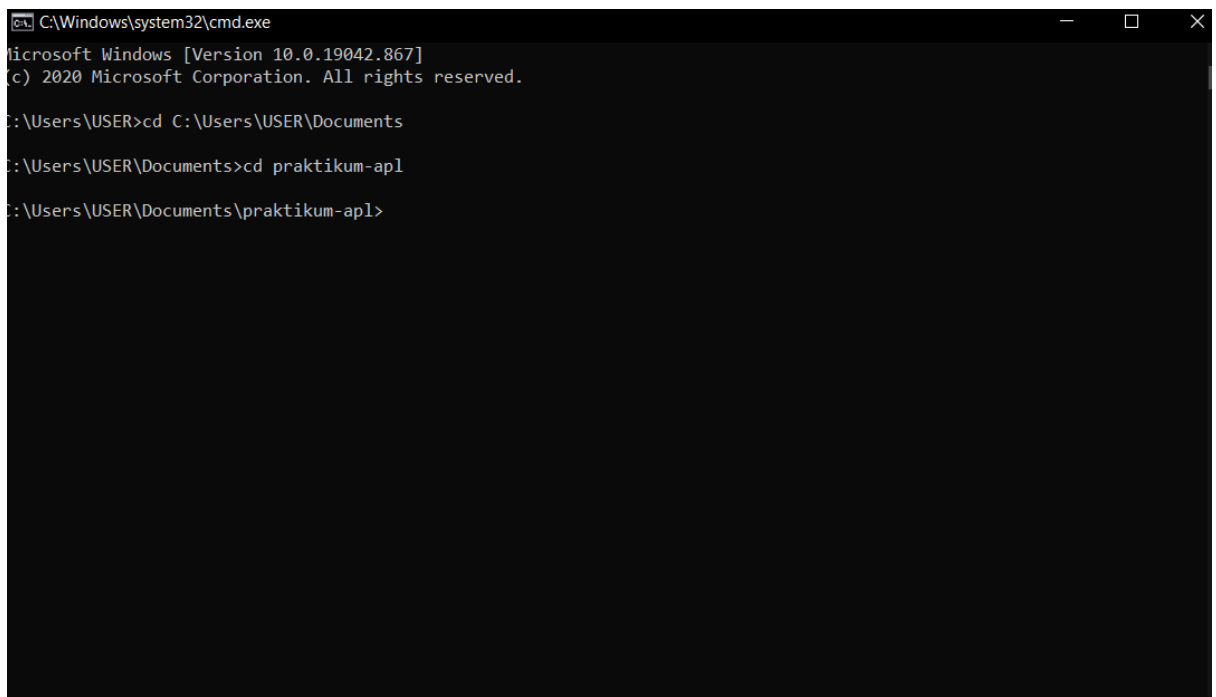


```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.19042.867]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\USER>cd C:\Users\USER\Documents

C:\Users\USER\Documents>
```

Gambar 5.1, 2. Path / direct folder



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.19042.867]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\USER>cd C:\Users\USER\Documents

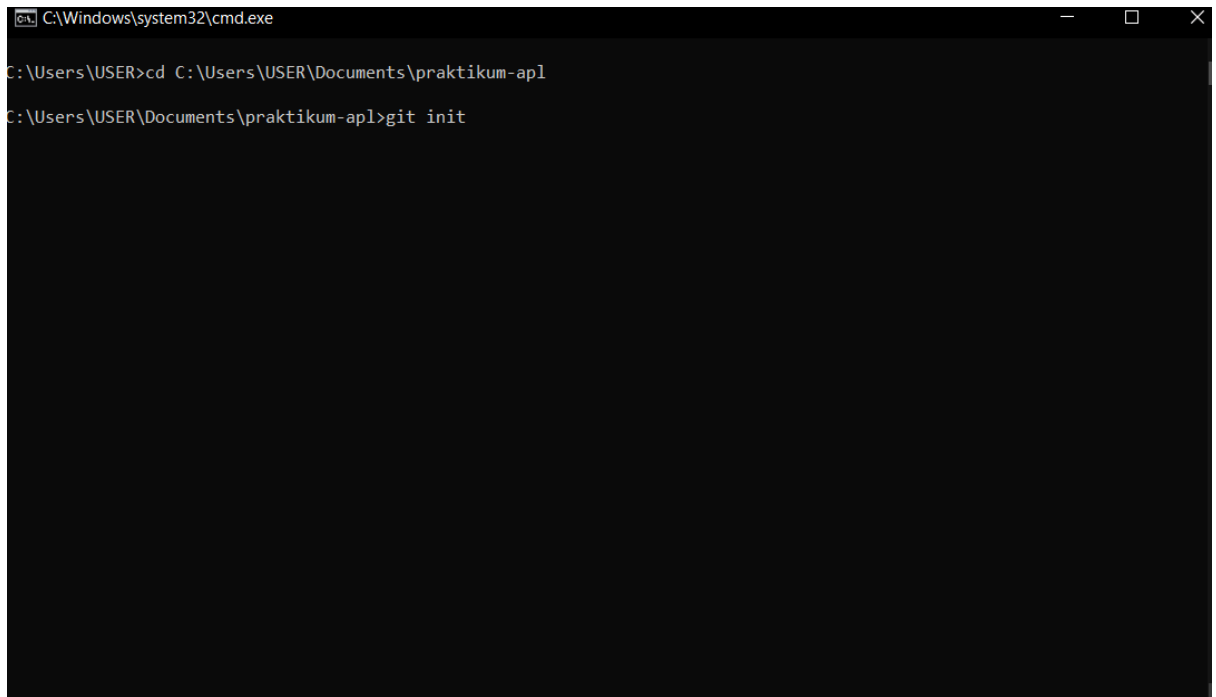
C:\Users\USER\Documents>cd praktikum-apl

C:\Users\USER\Documents\praktikum-apl>
```

Gambar 5.1, 3. Path / direct folder

3. Inisialisasi Git

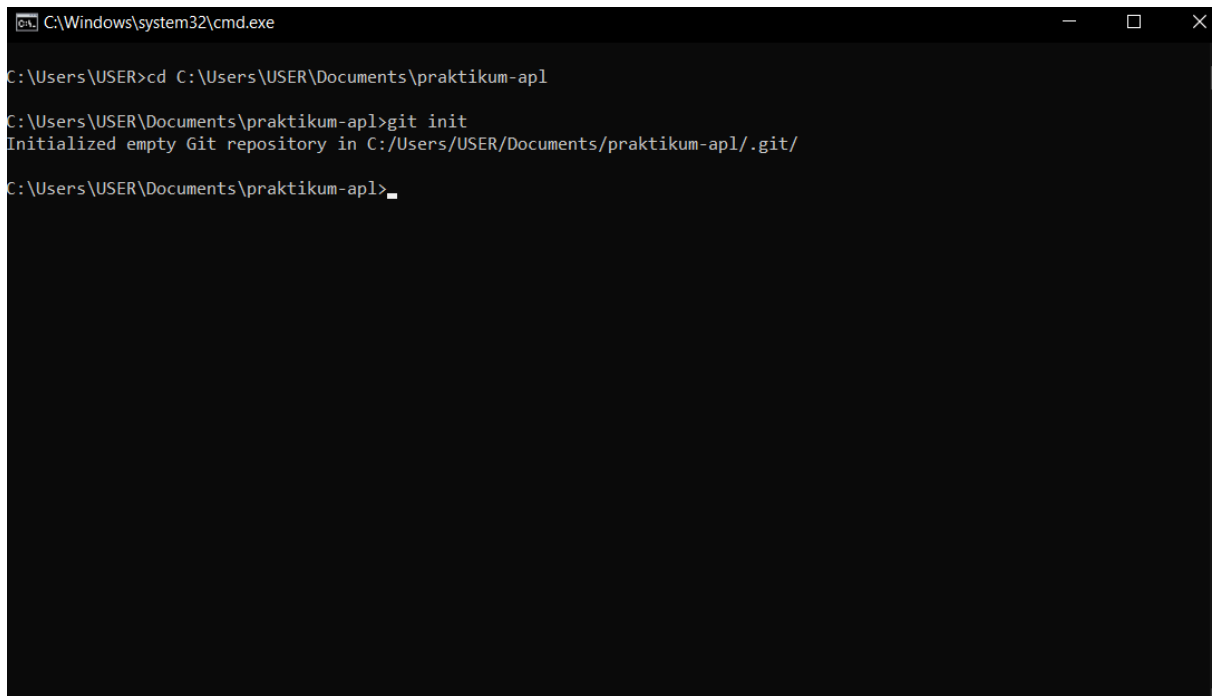
Jadikan folder sebagai repository Git, ketik



```
C:\Windows\system32\cmd.exe
C:\Users\USER>cd C:\Users\USER\Documents\praktikum-apl
C:\Users\USER\Documents\praktikum-apl>git init
```

Gambar 5.1, 3. Inisialisasi git init

Kalau berhasil, akan muncul pesan :

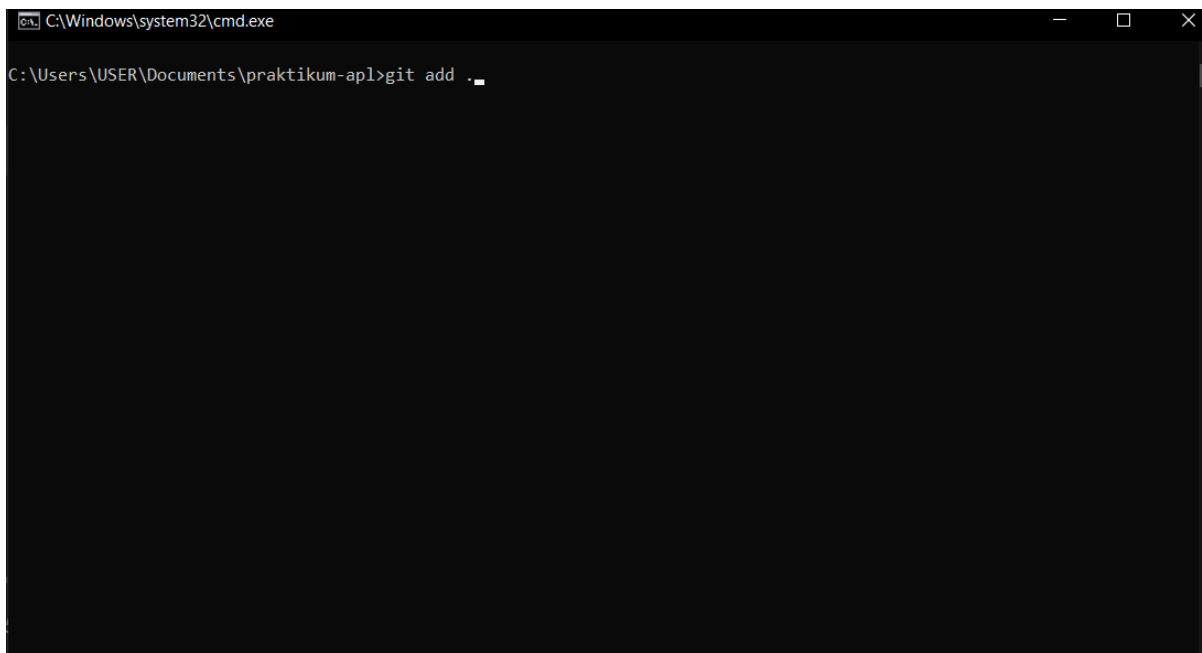


```
C:\Windows\system32\cmd.exe
C:\Users\USER>cd C:\Users\USER\Documents\praktikum-apl
C:\Users\USER\Documents\praktikum-apl>git init
Initialized empty Git repository in C:/Users/USER/Documents/praktikum-apl/.git/
C:\Users\USER\Documents\praktikum-apl>
```

Gambar 5.1, 4. Git init

5.2 Git Add

Setelah repository Git dibuat, Tambahkan file atau folder yang ingin dilacak oleh Git.

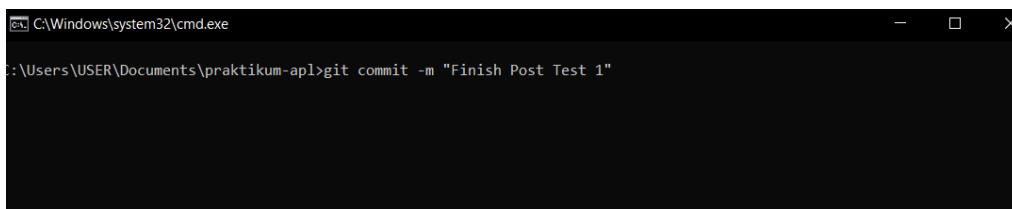


```
C:\Windows\system32\cmd.exe
C:\Users\USER\Documents\praktikum-apl>git add .
```

Gambar 5.2, 1. Add file/folder

5.3 Git Commit

Setelah menjalankan git add, langkah selanjutnya adalah git commit untuk menyimpan perubahan ke dalam repository Git lokal. Di terminal, ketik perintah berikut:



```
C:\Windows\system32\cmd.exe
C:\Users\USER\Documents\praktikum-apl>git commit -m "Finish Post Test 1"
```

Gambar 5.3, 1. Commit

Jika sudah berhasil akan menampilkan :

```
C:\Users\USER\Documents\praktikum-apl>git commit -m "Finish Post Test 1"
[main (root-commit) a6f3dfb] Finish Post Test 1
7 files changed, 575 insertions(+)
create mode 100644 kelas/Pertemuan-1/2409106039-MuhammadRizalAlfath-pertemuan-1.cpp
create mode 100644 post-test/post-test-1/.$2409106039_MuhammadRizalAlfath_PT-1.drawio.bkp
create mode 100644 post-test/post-test-1/2409106039-MuhammadRizalAlfath-PT-1.cpp
create mode 100644 post-test/post-test-1/2409106039-MuhammadRizalAlfath-PT-1.drawio
create mode 100644 post-test/post-test-1/2409106039_MuhammadRizalAlfath_PT_1.docx
create mode 100644 post-test/post-test-1/~$09106039_MuhammadRizalAlfath_PT_1.docx
create mode 100644 post-test/post-test-1/~WRL0003.tmp

C:\Users\USER\Documents\praktikum-apl>git status
On branch main
nothing to commit, working tree clean

C:\Users\USER\Documents\praktikum-apl>
```

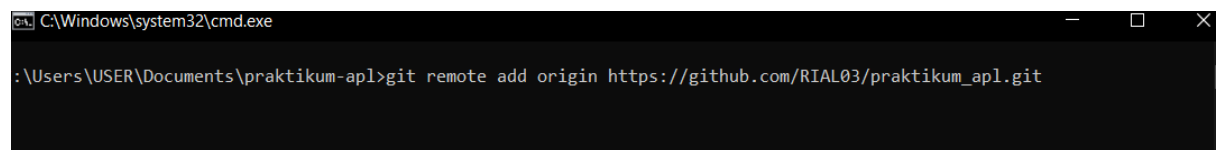
Gambar 5.3, 2. Commit berhasil

5.4 Git Remote

Setelah melakukan git commit, langkah selanjutnya adalah menghubungkan repository lokal ke repository di GitHub menggunakan git remote. git remote add origin, Perintah ini digunakan untuk menautkan repository lokal dengan repository di GitHub

1. Buat repository di github
2. Tambahkan Remote Repository di Git Lokal

Buka terminal di folder (praktikum-apl), lalu jalankan:

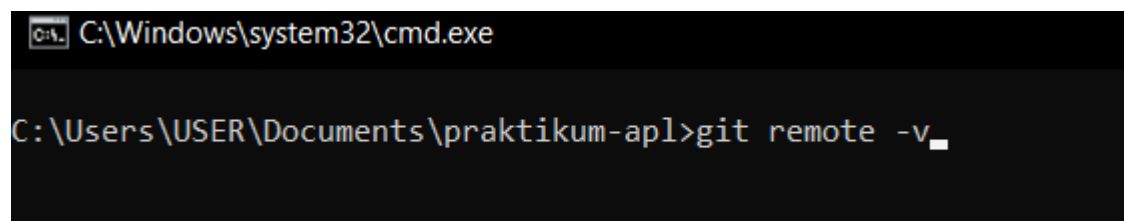


```
C:\Windows\system32\cmd.exe

C:\Users\USER\Documents\praktikum-apl>git remote add origin https://github.com/RIAL03/praktikum_apl.git
```

Gambar 5.4, 1. Git remote

Cek apakah remote sudah ditambahkan dengan perintah :

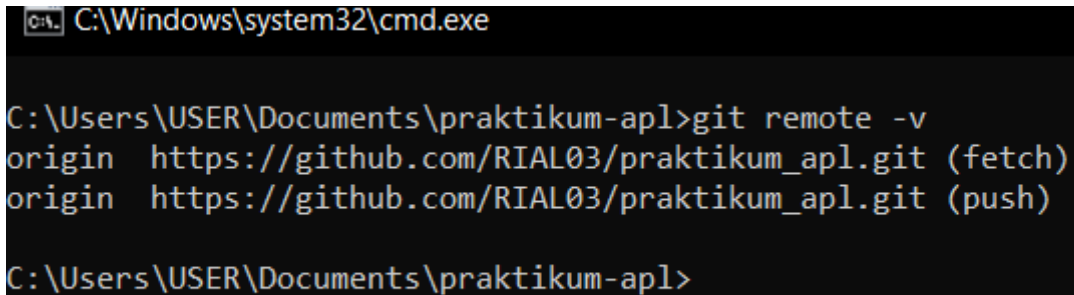


```
C:\Windows\system32\cmd.exe

C:\Users\USER\Documents\praktikum-apl>git remote -v
```

Gambar 5.4, 2. Cek remote

Jika berhasil, outputnya akan seperti ini :



```
C:\Windows\system32\cmd.exe

C:\Users\USER\Documents\praktikum-apl>git remote -v
origin  https://github.com/RIAL03/praktikum_apl.git (fetch)
origin  https://github.com/RIAL03/praktikum_apl.git (push)

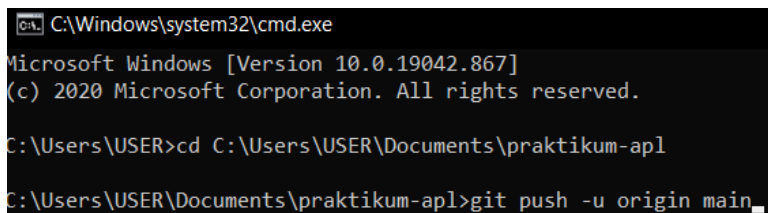
C:\Users\USER\Documents\praktikum-apl>
```

Gambar 5.4, 3. Output

5.5 Git push

git push digunakan untuk mengunggah perubahan dari lokal ke repository GitHub.

Ketik perintah berikut :



```
C:\Windows\system32\cmd.exe

Microsoft Windows [Version 10.0.19042.867]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\USER>cd C:\Users\USER\Documents\praktikum-apl

C:\Users\USER\Documents\praktikum-apl>git push -u origin main_
```

Gambar 5.4, 1. Psuh origin