

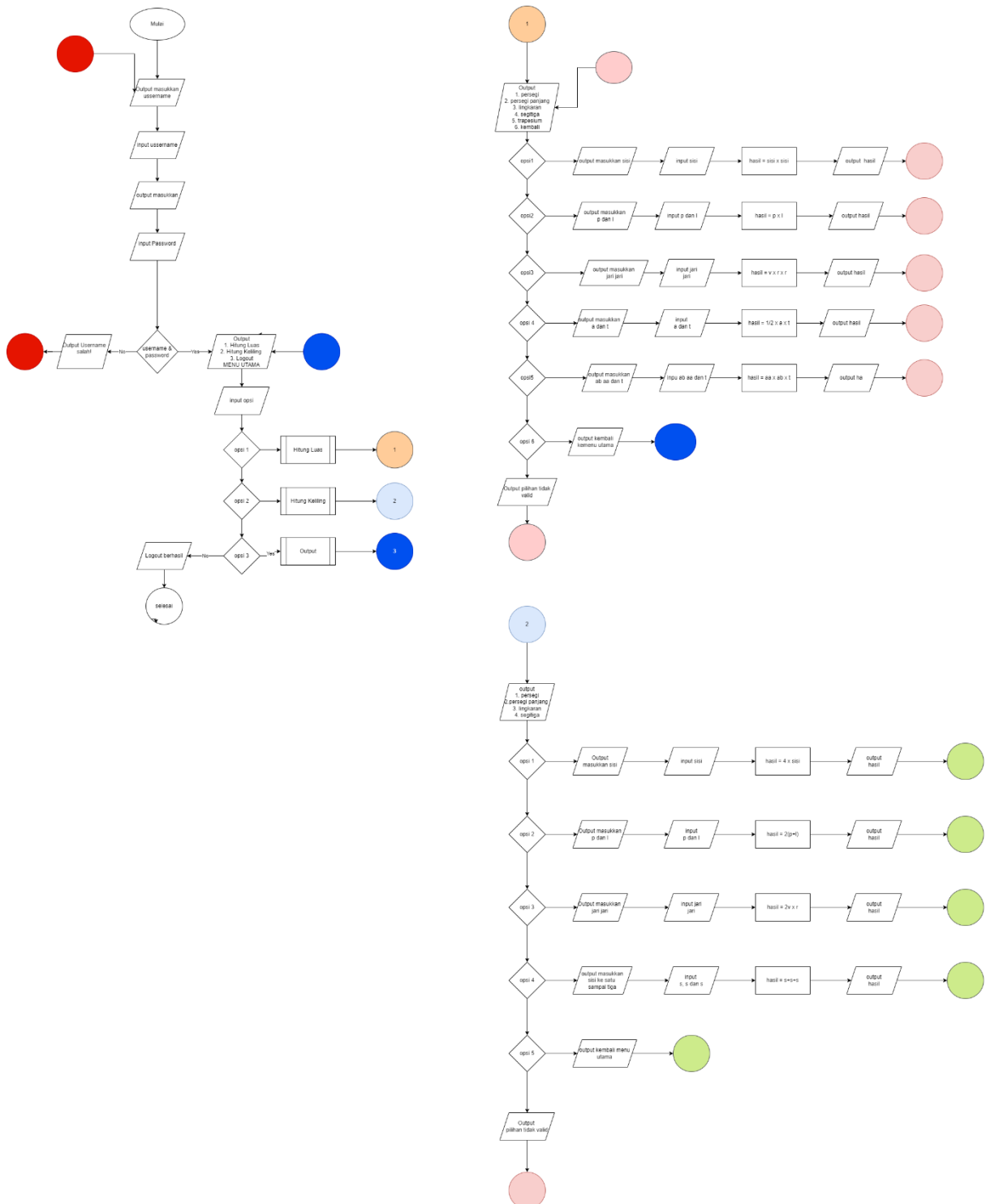
LAPORAN PRAKTIKUM
POSTTEST 1
ALGORITMA PEMROGRAMAN LANJUT



Disusun oleh:
Muhammad Fizriannur
2409106105
Kelas C1`24

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

1. Flowchart



Gambar 1.1 Flowchart

2. Analisis Program

2.1 Deskripsi Singkat Program

Program ini adalah aplikasi berbasis C++ untuk menghitung luas dan keliling berbagai bentuk geometris (persegi, persegi panjang, lingkaran, segitiga, dan trapesium).

Fitur Utama:

1. **Sistem login** – Pengguna harus memasukkan username dan password yang benar sebelum mengakses menu utama.
2. **Perhitungan Luas** – Menghitung luas berbagai bentuk berdasarkan input pengguna.
3. **Perhitungan Keliling** – Menghitung keliling berbagai bentuk dengan input sisi atau jari-jari.
4. **Menu Interaktif** – Pengguna dapat memilih untuk menghitung luas, keliling, atau logout.

2.2 Penjelasan Alur & Algoritma

Alur Jalannya Program

a. Program Dimulai

- a. Program pertama kali dijalankan melalui `main()`, yang memanggil fungsi `login()`.

b. Sistem Login

- a. Pengguna diminta memasukkan **username** dan **password**.
- b. Jika benar, program menampilkan pesan "Login berhasil!" dan melanjutkan ke `menuUtama()`.
- c. Jika salah, pengguna harus mencoba lagi sampai memasukkan kredensial yang benar.

c. Menu Utama

- a. Setelah login, pengguna diberikan tiga pilihan:

1. **Hitung Luas**
2. **Hitung Keliling**
3. **Logout**

d. Perhitungan Luas

- a. Jika pengguna memilih "Hitung Luas", maka akan muncul submenu:
 1. **Persegi** → Menghitung luas dengan rumus $\text{sisi} \times \text{sisi}$.
 2. **Persegi Panjang** → Menghitung luas dengan rumus $\text{panjang} \times \text{lebar}$.
 3. **Lingkaran** → Menghitung luas dengan rumus $\pi \times r^2$.

1. **Segitiga** → Menghitung luas dengan rumus $\frac{1}{2} \times \text{alas} \times \text{tinggi}$.
 2. **Trapesium** → Menghitung luas dengan rumus $\frac{1}{2} \times (\text{alas bawah} + \text{alas atas}) \times \text{tinggi}$.
- b. Hasil perhitungan ditampilkan ke layar.
- e. **Perhitungan Keliling**
- a. Jika pengguna memilih "Hitung Keliling", maka akan muncul submenu:
 1. **Persegi** → Menghitung keliling dengan rumus $4 \times \text{sisi}$.
 2. **Persegi Panjang** → Menghitung keliling dengan rumus $2 \times (\text{panjang} + \text{lebar})$.
 3. **Lingkaran** → Menghitung keliling dengan rumus $2 \times \pi \times r$.
 4. **Segitiga** → Menghitung keliling dengan menjumlahkan ketiga sisi.
 - b. Hasil perhitungan ditampilkan ke layar.
- f. **Logout dan Keluar**
- a. Jika pengguna memilih "Logout", program akan menampilkan pesan perpisahan dan keluar.
 - b. Program berhenti dan kembali ke sistem.

3. Source Code

A. Menu login di program ini percobaan maksimal 3 kali,jika gagal akan terkeluar dari program

```
int login() {
    string username, password;
    string userBenar = "Muhammad Fizriannur";
    string passBenar = "2409106105";
    int attempts = 3;

    while (attempts > 0) {
        cout << "\n==== LOGIN =====\n";
        cout << "Masukkan Username (atau ketik 'exit' untuk keluar): ";
        cin >> ws;
        getline(cin, username);

        if (username == "exit") {
            cout << "Keluar dari program.\n";
            return 0;
        }

        cout << "Masukkan Password: ";
        getline(cin, password);

        if (username == userBenar && password == passBenar) {
            cout << "Login berhasil! Selamat datang, " << userBenar << "!\n";
            return menuUtama();
        } else {
            attempts--;
            cout << "Username atau password salah! Sisa percobaan: " << attempts << "\n";
            if (attempts == 0) {
                cout << "Anda telah gagal login 3 kali. Program akan keluar.\n";
                return 0;
            }
        }
    }
}
```

Gambar 3.1 Menu Login

B. fungsi code ini untuk memastikan pengguna hanya memasukkan angka positif atau nol, angka akan terus meminta input sampai mendapatkan nilai yang benar

```
float cekInput(string pesan) {
    float nilai;
    while (true) {
        cout << pesan;
        cin >> nilai;

        if (cin.fail() || nilai < 0) {
            cout << "Input tidak valid! Masukkan angka yang benar.\n";
            cin.clear();
            cin.ignore(numeric_limits<streamsize>::max(), '\n');
        } else {
            cin.ignore(numeric_limits<streamsize>::max(), '\n');
            return nilai;
        }
    }
}
```

Gambar 3.2 meminta input angka dari pengguna dengan validasi

C. Kode ini adalah fungsi dalam C++ yang menghitung luas bangun datar berdasarkan pilihan pengguna. Menggunakan **menu** dan switch-case, fungsi ini menghitung luas

```
float hitungLuas() {
    while (true) {
        cout << "\n===== MENU LUAS =====\n";
        cout << "1. Persegi\n2. Persegi Panjang\n3. Lingkaran\n4. Segitiga\n5. Trapesium\n6. ";
        cout << "Pilih: ";
        int pilihan;
        cin >> pilihan;
        cin.ignore();

        switch (pilihan) {
            case 1: return pow(cekInput("Masukkan sisi: "), 2);
            case 2: return cekInput("Masukkan panjang: ") * cekInput("Masukkan lebar: ");
            case 3: return M_PI * pow(cekInput("Masukkan jari-jari: "), 2);
            case 4: return 0.5 * cekInput("Masukkan alas: ") * cekInput("Masukkan tinggi: ");
            case 5: return 0.5 * (cekInput("Masukkan alas bawah: ") + cekInput("Masukkan alas atas: ")) * cekInput("Masukkan tinggi: ");
            case 6: return 0;
            default: cout << "Pilihan tidak tersedia.\n";
        }
    }
}
```

Gambar 3.3 menghitung luas berbagai bangun datar

D. Fungsi ini berjalan dalam loop while (**true**), sehingga akan terus meminta input sampai pengguna memilih opsi "Kembali".

```
float hitungKeliling() {
    while (true) {
        cout << "\n==== MENU KELILING =====\n";
        cout << "1. Persegi\n2. Persegi Panjang\n3. Lingkaran\n4. Segitiga\n5. Kembali\n";
        cout << "Pilih: ";
        int pilihan;
        cin >> pilihan;
        cin.ignore();

        switch (pilihan) {
            case 1: return 4 * cekInput("Masukkan sisi: ");
            case 2: return 2 * (cekInput("Masukkan panjang: ") + cekInput("Masukkan lebar: "));
            case 3: return 2 * M_PI * cekInput("Masukkan jari-jari: ");
            case 4: return cekInput("Masukkan sisi pertama: ") + cekInput("Masukkan sisi kedua: ") + cekInput("Masukkan sisi ketiga: ");
            case 5: return 0;
            default: cout << "Pilihan tidak tersedia.\n";
        }
    }
}
```

Gambar 3.4 menghitung keliling bangun datar

E. Program ini terstruktur dengan baik, menggunakan **modularisasi** (fungsi hitungLuas(), hitungKeliling(), dan login()), serta memiliki **validasi input** untuk mencegah kesalahan pengguna.

```
int menuUtama() {
    while (true) {
        cout << "\n==== MENU UTAMA =====\n";
        cout << "1. Hitung Luas\n2. Hitung Keliling\n3. Logout\n";
        cout << "Pilih: ";
        int pilihan;
        cin >> pilihan;
        cin.ignore();

        switch (pilihan) {
            case 1:
                cout << "Hasil Luas: " << hitungLuas() << "\n";
                break;
            case 2:
                cout << "Hasil Keliling: " << hitungKeliling() << "\n";
                break;
            case 3:
                cout << "Terima kasih telah menggunakan program ini!\n";
                return 0;
            default:
                cout << "Pilihan tidak tersedia.\n";
        }
    }
}

int main() {
    return login();
}
```

Gambar 3.5 menampilkan menu utama untuk menghitung luas dan keliling bangun datar

4. Uji Coba dan Hasil Output

4.1 Uji Coba

Skenario 1 = Nama saya Muhammad Fizirannur dengan Nim 2409106105 saya akan masuk ke program dan ternyata berhasil

Skenario 2 = Saya ingin masuk dengan menggunakan nama fizri dengan Nim 2409106105 dan ternyata tidak bisa masuk

Skenario 3 = saya coba lagi dengan nama Muhammad tapi dengan Nim yang berbeda namun tetap gagal, sampai 3 kali dan saya pun dikeluarkan dari program.

4.2 Hasil Output

Jika login berhasil

```
===== LOGIN =====  
Masukkan Username (atau ketik 'exit' untuk keluar): Muhammad Fizriannur  
Masukkan Password: 2409106105  
Login berhasil! Selamat datang, Muhammad Fizriannur!
```

Gambar 4.1

Jika login gagal

```
===== LOGIN =====  
Masukkan Username (atau ketik 'exit' untuk keluar): fizri  
Masukkan Password: 2409106105  
Username atau password salah! Sisa percobaan: 2  
  
===== LOGIN =====  
Masukkan Username (atau ketik 'exit' untuk keluar): Muhammad  
Masukkan Password: 2409106105  
Username atau password salah! Sisa percobaan: 1  
  
===== LOGIN =====  
Masukkan Username (atau ketik 'exit' untuk keluar): Muhammad Fizriannur  
Masukkan Password: 242409106100  
Username atau password salah! Sisa percobaan: 0  
Anda telah gagal login 3 kali. Program akan keluar.  
PS C:\Users\fizri\OneDrive\Attachments\praktikumAPLJ\posttest1>
```

Gambar 4.2

5. Uji coba dan hasil output

5.1 Uji coba

Skenario 1 = Saya masuk ke menu utama,lalu memilih 1.Hitung luas dan lalu ada pilihan 1 – 6. Jika saya memilih no.1 maka masukkan sisi dan jumlahnya 529.

Skenario 2 = Saya masuk ke menu utama,lalu memilih 1.Hitung luas dan lalu ada pilihan 1 – 6. Jika saya memilih no.2 maka masukkan panjang misal 12, dan lebar 12 maka hasilnya 144.

Skenario 3 = Saya masuk ke menu utama,lalu memilih 1.Hitung luas dan lalu ada pilihan 1 – 6. Jika saya memilih no.3 maka masukkan jari” misalkan 10 mak hasilnya 314.159.

Skenario 4 = Saya masuk ke menu utama,lalu memilih 1.Hitung luas dan lalu ada pilihan 1 – 6. Jika meilih no.4 maka masukkan alas misalkan 100 dan tinggi 15 maka hasilnya 750.

Skenario 5 = Saya masuk ke menu utama,lalu memilih 1.Hitung luas dan lalu ada pilihan 1 – 6. Jika memilih no.5 maka masukkan alas bawah misal 56, alat atas 79, dan tinggi 14 makas hasilnya 945.

Skenario 6 = Saya masuk ke menu utama,lalu memilih 1.Hitung luas dan lalu ada pilihan 1 – 6. Jika memilih no.6 maka program kembali ke menu utama.

5.2 Hasil Output

Jika memilih persegi

```
===== MENU UTAMA =====
1. Hitung Luas
2. Hitung Keliling
3. Logout
Pilih: 1
Hasil Luas:
===== MENU LUAS =====
1. Persegi
2. Persegi Panjang
3. Lingkaran
4. Segitiga
5. Trapesium
6. Kembali
Pilih: 1
Masukkan sisi: 23
529

===== MENU UTAMA =====
1. Hitung Luas
2. Hitung Keliling
3. Logout
Pilih: 
```

Gambar 5.3 persegi

jika memilih persegi panjang

```
===== MENU UTAMA =====
1. Hitung Luas
2. Hitung Keliling
3. Logout
Pilih: 1
Hasil Luas:
===== MENU LUAS =====
1. Persegi
2. Persegi Panjang
3. Lingkaran
4. Segitiga
5. Trapesium
6. Kembali
Pilih: 2
Masukkan panjang: 12
Masukkan lebar: 12
144

===== MENU UTAMA =====
1. Hitung Luas
2. Hitung Keliling
3. Logout
Pilih: 
```

Gambar 5.4 persegi panjang

Jika memilih lingkaran

```
===== MENU UTAMA =====
1. Hitung Luas
2. Hitung Keliling
3. Logout
Pilih: 1
Hasil Luas:
===== MENU LUAS =====
1. Persegi
2. Persegi Panjang
3. Lingkaran
4. Segitiga
5. Trapesium
6. Kembali
Pilih: 3
Masukkan jari-jari: 10
314.159
```

Gambar 5.5 lingkaran

Jika memilih segitiga

```
===== MENU UTAMA =====
1. Hitung Luas
2. Hitung Keliling
3. Logout
Pilih: 1
Hasil Luas:
===== MENU LUAS =====
1. Persegi
2. Persegi Panjang
3. Lingkaran
4. Segitiga
5. Trapesium
6. Kembali
Pilih: 4
Masukkan alas: 100
Masukkan tinggi: 15
750
```

Gambar 5.6 segitiga

Jika memilih trapesium

```
===== MENU UTAMA =====
1. Hitung Luas
2. Hitung Keliling
3. Logout
Pilih: 1
Hasil Luas:
===== MENU LUAS =====
1. Persegi
2. Persegi Panjang
3. Lingkaran
4. Segitiga
5. Trapesium
6. Kembali
Pilih: 5
Masukkan alas bawah: 56
Masukkan alas atas: 79
Masukkan tinggi: 14
945
```

Gambar 5.7 trapesium

jika memilih kembali

```
===== MENU UTAMA =====
1. Hitung Luas
2. Hitung Keliling
3. Logout
Pilih: 1
Hasil Luas:
===== MENU LUAS =====
1. Persegi
2. Persegi Panjang
3. Lingkaran
4. Segitiga
5. Trapesium
6. Kembali
Pilih: 6
0

===== MENU UTAMA =====
1. Hitung Luas
2. Hitung Keliling
3. Logout
Pilih: 
```

Gambar 5.8 kembali

6. Uji coba dan hasil output

6.1 Uji coba

Skenario 1 = Saya masuk ke menu utama,lalu memilih 1.Hitung luas dan lalu ada pilihan 1 – 5. Jika memilih no.1 dan masukkan sisi misal 88 maka hasilnya 352.

Skenario 2 = Saya masuk ke menu utama,lalu memilih 1.Hitung luas dan lalu ada pilihan 1 – 5. Jika memilih no.2 dan masukkan panjang misal 67 dan lebar 45 maka hasilnya 224.

Skenario 3 = Saya masuk ke menu utama,lalu memilih 1.Hitung luas dan lalu ada pilihan 1 – 5. Jika saya memilih no.3 dan memasukkan jari” misal 55 maka hasilnya 3435.575.

Skenario 4 = Saya masuk ke menu utama,lalu memilih 1.Hitung luas dan lalu ada pilihan 1 – 5. Jika saya memilih no.4 dan masukkan sisi pertama misal 56, sisi kedua 12, dan sisi ketiga 17 maka hasilnya 85.

Skenario 5 = Saya masuk ke menu utama,lalu memilih 1.Hitung luas dan lalu ada pilihan 1 – 5. Jika saya memilih no.5

6.2 Hasil Output

Jika memilih persegi

```
===== MENU UTAMA =====
1. Hitung Luas
2. Hitung Keliling
3. Logout
Pilih: 2
Hasil Keliling:
===== MENU KELILING =====
1. Persegi
2. Persegi Panjang
3. Lingkaran
4. Segitiga
5. Kembali
Pilih: 1
Masukkan sisi: 88
352
```

Gambar 6.3 persegi

jika memilih persegi panjang

```
===== MENU UTAMA =====
1. Hitung Luas
2. Hitung Keliling
3. Logout
Pilih: 2
Hasil Keliling:
===== MENU KELILING =====
1. Persegi
2. Persegi Panjang
3. Lingkaran
4. Segitiga
5. Kembali
Pilih: 2
Masukkan panjang: 67
Masukkan lebar: 45
224
```

Gambar 6.4 persegi panjang

Jika memilih lingkaran

```
===== MENU UTAMA =====
1. Hitung Luas
2. Hitung Keliling
3. Logout
Pilih: 2
Hasil Keliling:
===== MENU KELILING =====
1. Persegi
2. Persegi Panjang
3. Lingkaran
4. Segitiga
5. Kembali
Pilih: 3
Masukkan jari-jari: 55
345.575
```

Gambar 6.5 lingkaran

jika memilih segitiga

```
===== MENU UTAMA =====
1. Hitung Luas
2. Hitung Keliling
3. Logout
Pilih: 2
Hasil Keliling:
===== MENU KELILING =====
1. Persegi
2. Persegi Panjang
3. Lingkaran
4. Segitiga
5. Kembali
Pilih: 4
Masukkan sisi pertama: 56
Masukkan sisi kedua: 12
Masukkan sisi ketiga: 17
85
```

Gambar 6.7 segitiga

Jika memilih kembali

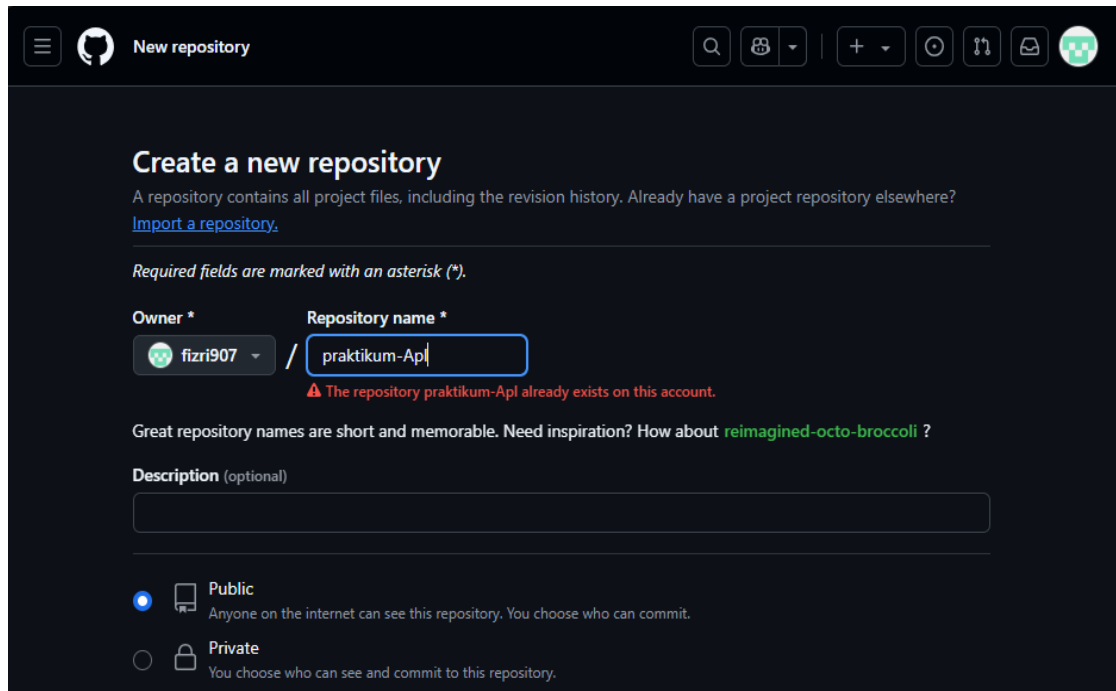
```
===== MENU UTAMA =====
1. Hitung Luas
2. Hitung Keliling
3. Logout
Pilih: 2
Hasil Keliling:
===== MENU KELILING =====
1. Persegi
2. Persegi Panjang
3. Lingkaran
4. Segitiga
5. Kembali
Pilih: 5
0
```

Gambar 6.8 kembali

7. Sistem GitHub

1. Membuat Repository

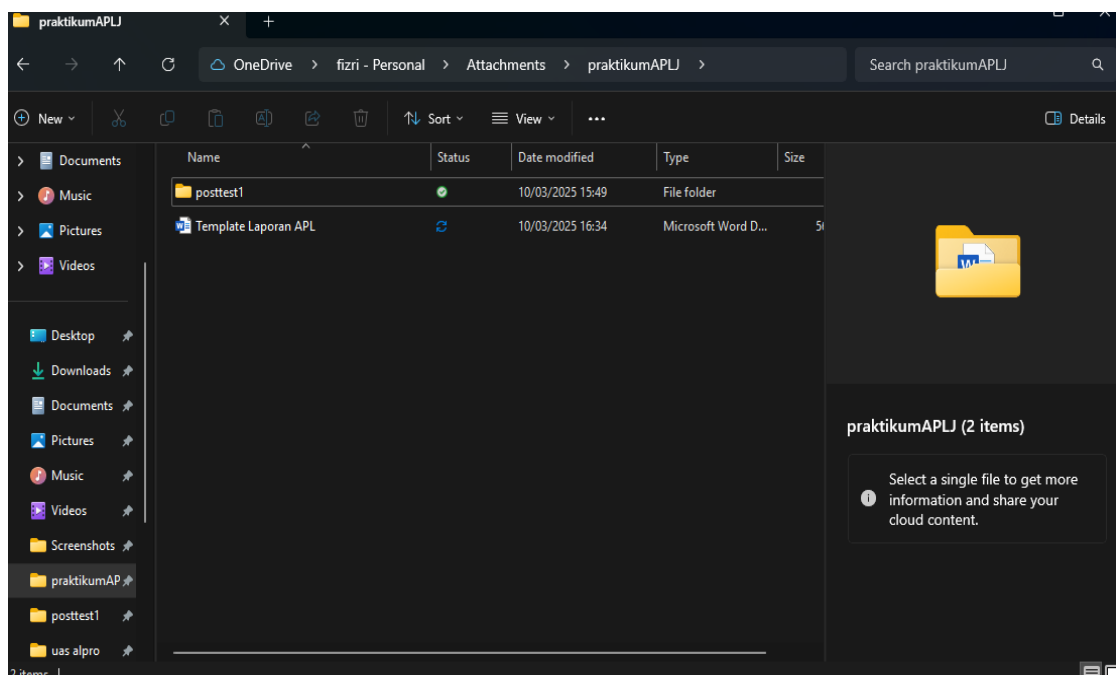
Membuat repository baru untuk menyimpan folder



Gambar 7.1 Repository

2. Membuat Folder

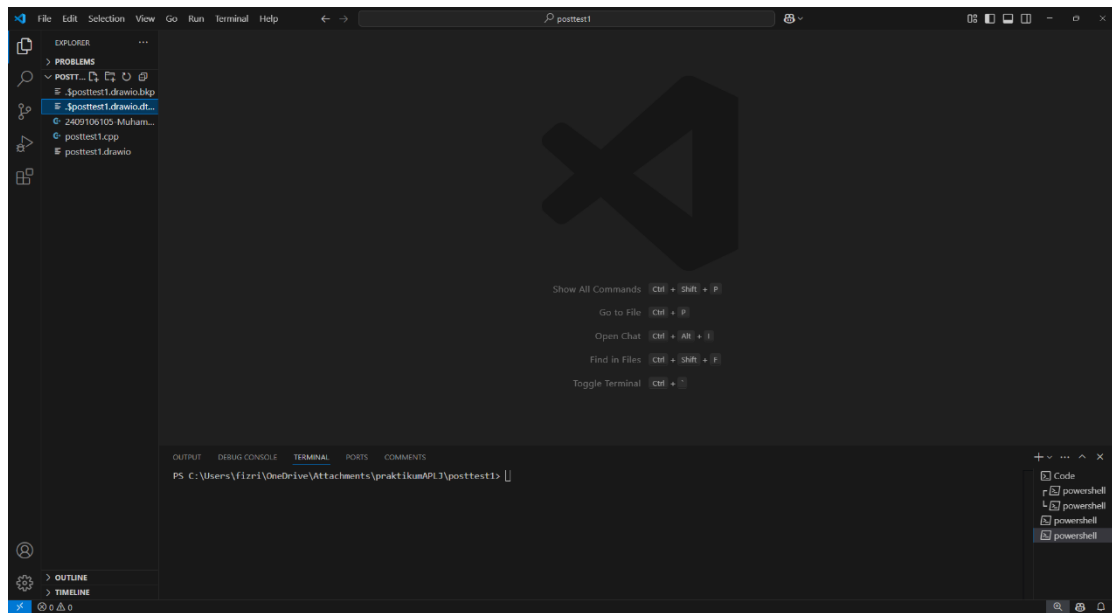
Buat folder eksplor dengan nama praktikum-Apl yang sesuai dengan repository github, dan di dalam folder buat 2 folder lagi dengan nama kelas dan posttest.



Gambar 7.2 Folder

3. Membuat Folder di vscode dan terminal

Buka folder yang sudah di buat di visualcode lalu kiln new terminal, pastikan path diterminal sesuai dengan folder utama.



Gambar 7.3 Vscode

4. Langkah Langkah Git

Pertama klik kanan pada folder lalu pilih show open with lalu pilih open git bash

Kedua ketika sudah di terminal ketik `git config --global user.email`”sesuaikan dengan email yang ada di akun github”

Ketiga `git init` untuk menginisiasi repository git.

Keempat ketik `git add` untuk menambah file yang akan di komit dan `git commit -m “Update”` untuk membuat check point

Kelima ketik `git remote` untuk menghubungkan repository yang ada di lokal computer dengan cara mengcopy link yang ada di github.

```
MINGW64:/c/Github/Praktikum-Apl

fizri@LAPTOP-2D98KBHL MINGW64 /c/Github/Praktikum-Apl
$ git config --global user.email "fizri907@gmail.com"

fizri@LAPTOP-2D98KBHL MINGW64 /c/Github/Praktikum-Apl
$ git init
Initialized empty Git repository in C:/Github/Praktikum-Apl/.git/

fizri@LAPTOP-2D98KBHL MINGW64 /c/Github/Praktikum-Apl (master)
$ git add .

fizri@LAPTOP-2D98KBHL MINGW64 /c/Github/Praktikum-Apl (master)
$ git commit -m "update fizri"
[master (root-commit) 6211ddf] update fizri
1 file changed, 167 insertions(+)
create mode 100644 Post-test/Post-test-1/2409106105-Muhammad Fizriannur-PT-1.cp
p

fizri@LAPTOP-2D98KBHL MINGW64 /c/Github/Praktikum-Apl (master)
$ git branch -M main

fizri@LAPTOP-2D98KBHL MINGW64 /c/Github/Praktikum-Apl (main)
$ git remote add origin https://github.com/fizri907/Praktikum-Apl.git

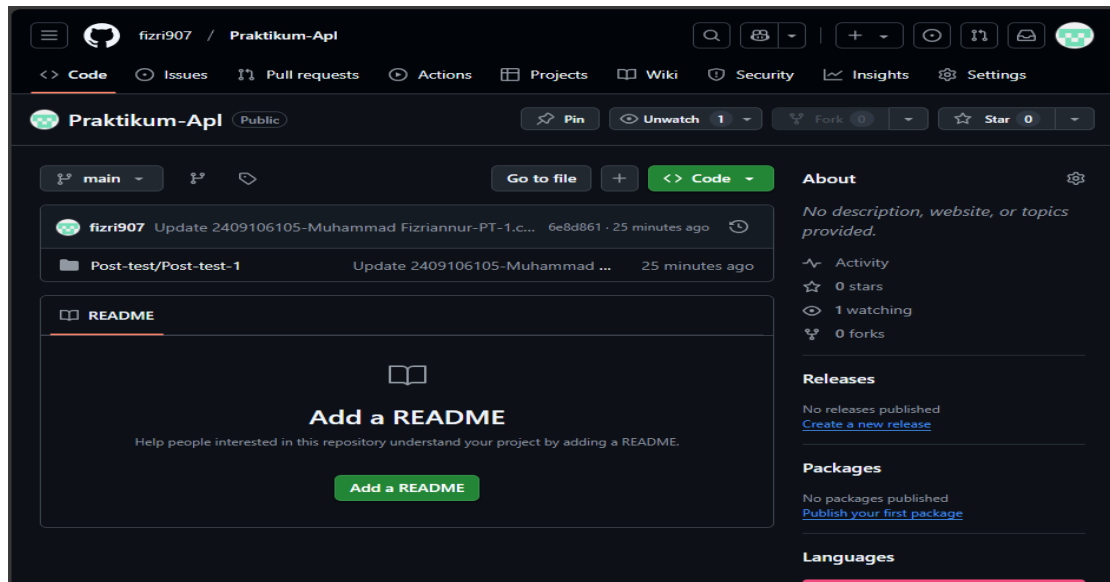
fizri@LAPTOP-2D98KBHL MINGW64 /c/Github/Praktikum-Apl (main)
$ git push -u origin main
info: please complete authentication in your browser...
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (5/5), 1.43 KiB | 1.43 MiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/fizri907/Praktikum-Apl.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.

fizri@LAPTOP-2D98KBHL MINGW64 /c/Github/Praktikum-Apl (main)
$ |
```

Gambar 7.4 langkah Langkah Git dan Gitpush

5. GitHub

Reload github kalian pada browser atau google choreme kalian



Gambar 7.6 Reload