

LAPORAN PROJECT AKHIR PBO

PEMBUATAN KALKUKATOR SUBNETTING DENGAN PBO

Mata Kuliah : Pemrograman Berorientasi Objek

Semester : III (Tiga) **Rombel :** I (Satu)



Penyusun :

Deska Zodanta (2310506014)

Akbar Indra Saputra (2310506015)

Zharifa Nur Majidah (2330506021)

Nabila Kartika Putri (2320506039)

PROGRAM STUDI S1 TEKNOLOGI INFORMASI
JURUSAN TEKNIK ELEKTRO FAKULTAS TEKNIK
UNIVERSITAS TIDAR
GANJIL 2023/2024

BAB I

PENDAHULUAN

1.1. Latar Belakang

Pengelolaan jaringan komputer, terutama subnetting, adalah aspek penting dalam teknologi informasi. Subnetting memungkinkan optimasi alamat IP, meningkatkan efisiensi komunikasi, dan mengurangi konflik jaringan. Namun, perhitungan subnet secara manual sering kali sulit dan rentan kesalahan, yang dapat memengaruhi alokasi IP dan konfigurasi jaringan. Untuk mengatasi hal ini, dibutuhkan alat yang dapat menghitung subnet secara otomatis dan akurat. Aplikasi kalkulator subnet ini membantu pengguna menghitung subnet mask, jumlah host, rentang alamat IP, alamat jaringan, dan alamat broadcast berdasarkan alamat IP dan subnet prefix. Dengan aplikasi ini, proses perhitungan menjadi lebih cepat dan efisien.

Aplikasi ini dikembangkan menggunakan pendekatan Pemrograman Berorientasi Objek (PBO) dengan bahasa Java, yang menawarkan modularitas, keterbacaan kode, dan kemudahan pemeliharaan. Konsep PBO seperti kelas, objek, enkapsulasi, pewarisan, dan polimorfisme diterapkan untuk memastikan struktur aplikasi yang fleksibel. Dirancang dengan antarmuka yang sederhana, aplikasi ini dapat digunakan oleh pelajar hingga profesional jaringan komputer. Selain sebagai alat perhitungan, aplikasi ini juga berfungsi sebagai sarana pembelajaran subnetting. Diharapkan aplikasi ini dapat meningkatkan efisiensi manajemen jaringan dan mendukung pembelajaran teknologi informasi, khususnya manajemen jaringan komputer.

1.2. Rumusan Masalah

1. Bagaimana cara mengoptimalkan proses perhitungan subnetting agar lebih cepat, akurat, dan minim kesalahan?
2. Apa saja fitur yang disajikan dalam aplikasi kalkulator subnet untuk mendukung kebutuhan pelajar hingga profesional jaringan komputer?

1.3. Tujuan Penulisan

1. Menerapkan konsep Pemrograman Berorientasi Objek (PBO) dalam pengembangan aplikasi yang terstruktur
2. Memberikan alternatif solusi untuk mempermudah pelajar hingga teknisi jaringan melalui aplikasi kalkulator komputer.

BAB II

LANDASAN TEORI

2.1. Konsep Subnetting

Subnetting adalah proses membagi jaringan besar menjadi subnet-subnet lebih kecil untuk mengoptimalkan penggunaan alamat IP, meningkatkan komunikasi antar perangkat, dan mengurangi konflik jaringan. Proses ini melibatkan alamat IP, subnet mask, dan perhitungan jumlah host per subnet. Namun untuk menerapkannya kita membutuhkan beberapa yang harus dipenuhi yaitu:

- a. Alamat IP, yang merupakan pengidentifikasi unik unik perangkat dalam jaringan.
- b. Subnet Mask, digunakan untuk membedakan Network ID dan Host ID
- c. CIDR (*Classless Inter-Domain Routing*), merupakan cara penulisan subnet mask dalam format singkat.

Subnetting digunakan untuk memecah jaringan besar menjadi subnet lebih kecil dengan menghitung beberapa parameter penting, antara lain:

- a. Jumlah subnet dihitung berdasarkan jumlah bit yang dipinjam dari Host ID.
- b. Jumlah Host per Subnet dihitung dengan rumus $2^h - 2$ (h adalah jumlah bit tersisa untuk Host ID).
- c. Range Alamat IP menunjukkan alamat yang tersedia untuk perangkat dalam setiap subnet dari alamat jaringan hingga alamat broadcast.

2.2. Pemrograman Berorientasi Objek (PBO)

Pemrograman Berorientasi Objek (Object-Oriented Programming atau OOP) adalah paradigma pemrograman yang berfokus pada penggunaan objek dan kelas untuk merepresentasikan data dan operasi. Berikut adalah beberapa hal penting dalam konsep dasar PBO:

- a. Kelas

Kelas adalah template atau blueprint untuk menciptakan objek. Kelas mendefinisikan atribut (variabel) dan metode (fungsi) yang dimiliki oleh objek.

- b. Objek

Objek adalah instance dari kelas yang memiliki atribut dan dapat menjalankan metode yang didefinisikan oleh kelas.

- c. Enkapsulasi

Konsep menyembunyikan detail implementasi dengan memberikan akses hanya melalui metode tertentu.

d. Pewarisan (Inheritance)

Kemampuan suatu kelas untuk mewarisi atribut dan metode dari kelas lain.

2.3. Teknologi yang Digunakan

1.1 Java

Java adalah bahasa pemrograman berbasis PBO yang populer, dikenal karena portabilitasnya (write once, run anywhere). Dalam pengembangan aplikasi ini, Java digunakan untuk menangani logika perhitungan subnetting.

2.1 Swing

Swing adalah library Java untuk membangun antarmuka grafis (GUI). Swing menyediakan elemen visual seperti tombol, label, dan input field, yang memudahkan pengembangan aplikasi desktop.

3.1 IDE

Menggunakan JavaNetbeans untuk meningkatkan efisiensi pengembangan aplikasi melalui fitur seperti debugging, auto-completion, dan integrasi library.

BAB II

METODOLOGI

3.1 Tahapan Pengembangan

1. Analisis Kebutuhan

Mengidentifikasi fitur utama aplikasi, seperti perhitungan subnet mask, alamat jaringan, dan alamat broadcast

2. Perancangan Struktur Program

Membuat diagram UML untuk menggambarkan hubungan antar kelas.

3. Pengembangan Aplikasi

Implementasi menggunakan Java dengan konsep PBO

4. Pengujian

Menguji aplikasi untuk memastikan semua fitur berfungsi dengan baik.

3.2 Fitur-Fitur Pada Aplikasi

1. Input Alamat IP

Pengguna dapat memasukkan alamat IP dalam format IPv4.

2. Input Prefix

Memungkinkan pengguna untuk menentukan prefix (seperti /24) untuk jaringan.

3. Hitung Subnet

Tombol Hitung untuk memproses data yang dimasukkan dan menghasilkan informasi jaringan.

4. Reset Data

Fitur tombol *Reset* pada aplikasi kalkulator subnet berfungsi untuk menghapus semua data yang telah dimasukkan pengguna (seperti alamat IP dan prefix) serta membersihkan hasil perhitungan yang telah ditampilkan. Dengan fitur ini, pengguna dapat dengan cepat mengosongkan semua kolom input dan output, sehingga aplikasi siap digunakan untuk melakukan perhitungan baru tanpa perlu menghapus data secara manual.

5. Menampilkan Subnet Mask, Network ID, Ip Broadcast, dan Jumlah Subnet.

Fitur ini secara otomatis menghasilkan informasi penting tentang konfigurasi jaringan berdasarkan alamat IP dan prefix/subnet mask yang dimasukkan. Aplikasi akan menampilkan subnet mask dalam format desimal (seperti 255.255.255.0) untuk menentukan batas jaringan, Network ID sebagai alamat unik yang merepresentasikan subnet tertentu, IP Broadcast sebagai alamat terakhir dalam

BAB IV

IMPLEMENTASI APLIKASI

4.1 Struktur Program

1. IPTidak Valid

Exception ini menangani kesalahan input alamat IP, seperti format yang salah atau nilai di luar rentang, untuk memastikan hanya data valid yang diproses.

2. SubnetTidakValid

Exception ini menangani kesalahan input subnet mask atau prefix yang tidak valid, mencegah perhitungan subnet dengan data yang salah.

3. IPAddress

Kelas ini merepresentasikan alamat IP, memastikan formatnya valid (empat oktet dalam rentang 0–255), serta menyediakan konversi antara format desimal dan biner.

4. Penghitungan

Kelas ini menghitung informasi jaringan, seperti Network ID, Broadcast Address, jumlah host, dan rentang IP berdasarkan alamat IP dan subnet mask yang diberikan.

5. Main

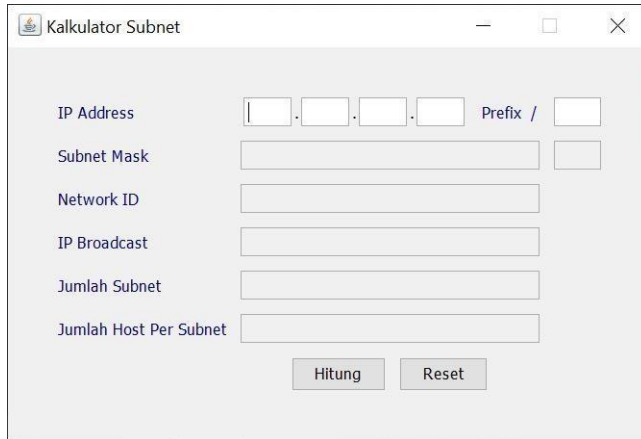
Kelas ini membagi jaringan besar menjadi subnet kecil, menghitung jumlah subnet yang tersedia, dan menampilkan informasi detail untuk setiap subnet, seperti Network ID dan Broadcast Address.

6. SubnetMask

Kelas ini merepresentasikan subnet mask dalam format desimal (contoh: 255.255.255.0) atau prefix (contoh: /24). SubnetMask memvalidasi input dan mendukung konversi antara format desimal dan CIDR.

4.2 Antarmuka Aplikasi

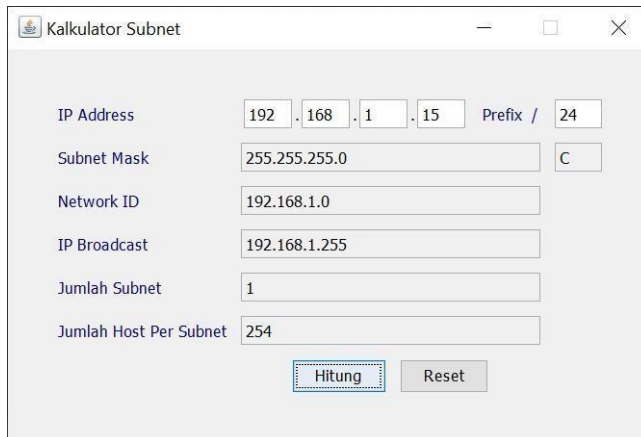
1. Tampilan Utama Aplikasi



The screenshot shows the 'Kalkulator Subnet' application window. It features a light gray background and a standard Windows-style title bar with a minimize button, a maximize button, and a close button. The main area contains several input fields and two buttons. The input fields are labeled 'IP Address', 'Subnet Mask', 'Network ID', 'IP Broadcast', 'Jumlah Subnet', and 'Jumlah Host Per Subnet'. The 'IP Address' field is split into four boxes for individual octets, followed by a 'Prefix /' label and a box for the prefix value. The 'Subnet Mask' field has a 'C' button next to it. The 'Hitung' and 'Reset' buttons are located at the bottom right of the input area.

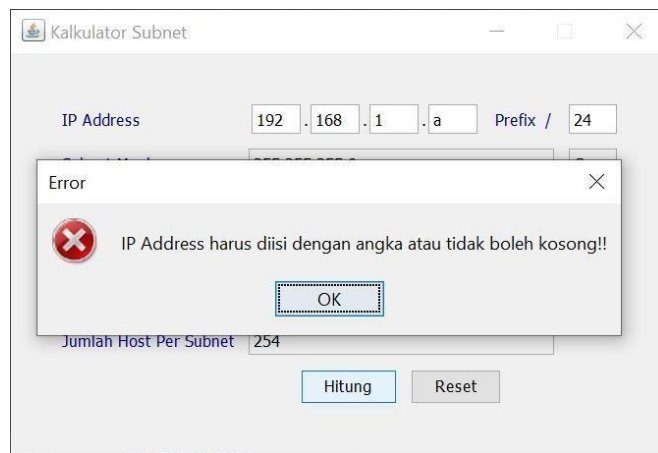
Tampilan tersebut adalah antarmuka aplikasi Kalkulator Subnet, yang digunakan untuk menghitung parameter subnetting berdasarkan IP Address dan prefix yang dimasukkan. Pengguna dapat mengisi IP Address dan prefix di bagian input, lalu menekan tombol Hitung untuk mendapatkan hasil seperti Subnet Mask, Network ID, IP Broadcast, Jumlah Subnet, dan Jumlah Host per Subnet. Tombol Reset digunakan untuk menghapus semua input dan output agar bisa memulai perhitungan baru. Aplikasi ini dirancang sederhana dan mudah digunakan.

2. Tampilan Penggunaan Aplikasi



This screenshot shows the same 'Kalkulator Subnet' application window after a calculation. The 'IP Address' field now contains '192', '168', '1', and '15'. The 'Prefix' field contains '24'. The 'Subnet Mask' field displays '255.255.255.0' with a 'C' button next to it. The 'Network ID' field displays '192.168.1.0'. The 'IP Broadcast' field displays '192.168.1.255'. The 'Jumlah Subnet' field displays '1'. The 'Jumlah Host Per Subnet' field displays '254'. The 'Hitung' button is highlighted with a dashed blue border, indicating it was the last active button.

3. Exception Handling



Gambar tersebut menunjukkan pesan error dari aplikasi Kalkulator Subnet. Pesan tersebut muncul karena input pada bagian IP Address tidak valid, yaitu terdapat karakter non-numerik ("a") yang dimasukkan. Aplikasi memberikan peringatan bahwa IP Address harus diisi dengan angka dan tidak boleh kosong.

Kotak dialog error ini membantu pengguna untuk memperbaiki kesalahan input sebelum melanjutkan perhitungan subnetting. Pengguna perlu mengganti nilai yang salah dengan angka valid dalam format IPv4 (xxx.xxx.xxx.xxx).

4.3 Evaluasi Aplikasi

Aplikasi ini memiliki kelebihan seperti antarmuka yang sederhana dan mudah untuk digunakan dengan perhitungan yang sudah akurat dan cepat. Namun, aplikasi tersebut juga memiliki kekurangan karena belum bisa mendukung IPV6.

BAB V

PENUTUP

5.1 Kesimpulan

Aplikasi kalkulator subnet berbasis Java berhasil dikembangkan untuk membantu pengguna dalam menghitung subnet mask, alamat jaringan, alamat broadcast, dan jumlah host secara cepat dan akurat. struktur program yang terdiri dari IPAddress, SubnetMask, NetworkCalculator, SubnetCalculator, InvalidIPEException, dan InvalidSubnetException dirancang untuk mengelola dan menghitung informasi subnet dengan efisien. IPAddress dan SubnetMask digunakan sebagai input untuk perhitungan, memastikan bahwa data yang dimasukkan valid dan sesuai. NetworkCalculator dan SubnetCalculator menangani perhitungan seperti Network ID, Broadcast Address, jumlah host, serta pembagian jaringan menjadi subnet yang lebih kecil. Sementara itu, InvalidIPEException dan InvalidSubnetException memberikan mekanisme penanganan kesalahan untuk memvalidasi input dan mencegah proses dengan data yang salah. Struktur ini memudahkan administrator jaringan dalam merancang, membagi, dan mengelola jaringan komputer secara optimal.

5.2 Saran

Perlu adanya tambahan fitur pada aplikasi agar bisa mendukung untuk penggunaan IPv6 dan juga agar bisa memvisualisasikan jaringan.

DOKUMEN REFLEKSI DIRI ANGGOTA

1. DESKA ZODANTA (2310506014)

Dalam proyek akhir PBO ini, saya pribadi ikut dalam tahap perencanaan awal hingga akhir proses pembuatan, dimulai dengan menentukan hingga memutuskan ide aplikasi yang ingin dibuat. Pada tahap awal, saya bersama teman-teman sekelompok memikirkan tentang ide yang relevan untuk dikembangkan menjadi sebuah aplikasi. Setelah itu akhirnya menemukan ide yang cocok yaitu mengembangkan aplikasi kalkulator subnet. Menurut saya memilih mengembangkan kalkulator subnet merupakan pilihan yang sangat baik dalam konteks pembelajaran dan penerapan jaringan komputer dipadukan dengan pembelajaran pemrograman. Pertama, kalkulator subnet menjadi alat yang sangat berguna bagi para teknisi jaringan, administrator sistem, dan mahasiswa dengan jurusan terkait. Dengan kalkulator subnet, mereka dapat dengan cepat dan akurat dalam menghitung berbagai parameter jaringan seperti jumlah subnet, jumlah host per subnet, alamat IP network, broadcast address, dan subnet mask. Kedua, melalui proses pengembangan kalkulator subnet, pemahaman konsep subnetting akan semakin mendalam. Proses perancangan algoritma, konsep pengalamatan jaringan, dan pengujian teknis lainnya akan memaksa kita untuk menggali lebih dalam mengenai teori di balik subnetting.

Ketiga, proyek ini juga dapat menjadi portofolio yang menarik untuk ditampilkan pada saat melamar pekerjaan, menunjukkan kemampuan dalam algoritma jaringan, pemecahan masalah, pemahaman konsep jaringan, hingga kemampuan teknis dalam mengimplementasikannya dalam kode program sampai aplikasi kalkulator subnet tersebut dapat berjalan dan menghasilkan hasil yang akurat. Terakhir, kalkulator subnet yang telah dikembangkan dapat dibagikan kepada komunitas open source, sehingga dapat bermanfaat bagi banyak orang dan berkontribusi pada pengembangan ilmu pengetahuan terutama yang terkait jaringan maupun pemrograman. Diluar pembahasan mengenai jaringan, dengan mengembangkan proyek akhir kalkulator subnetting ini juga akan melatih kemampuan kita dalam pemrograman terutama yang menggunakan bahasa pemrograman Java. Setelah mengetahui tujuan dan manfaat mengembangkan aplikasi tersebut, selanjutnya saya pribadi beserta teman-teman lainnya membuat struktur program sederhana yang sekaligus bisa menjalankan penghitungan subnet. Pada tahap awal pengembangan kode program ini hasil output dari kalkulatornya relatif masih sederhana, terutama belum bisa menangani error jika kita

memasukkan IP address yang tidak sesuai. Setelah aplikasi sederhana tersebut jadi dan dirasa sudah memiliki gambaran hingga cakupan luas untuk bisa menghitung beberapa cakupan konsep pengalamatan jaringan mulai dari prefix, subnetmask yang cocok, dan cakupan lainnya langkah selanjutnya adalah menambahkan beberapa fitur yang terkait dengan penghitungan IP, termasuk dalam penanganan error handling.

Menurut saya pribadi penanganan error menjadi sangat penting karena perlu diketahui dalam pengalamatan jaringan banyak algoritma logis yang mana jika suatu block IP atau block lain yang dimasukkan salah namun dieksekusi oleh program dengan benar, maka bisa menimbulkan kekeliruan dalam pengalamatan jaringan. Setelah perbaikan serta evaluasi semua pengembangan selesai, saya pribadi beserta kelompok saya melakukan finalisasi proyek serta pembuatan laporan.

2. Akbar Indra Saputra (2310506015)

Pada proyek akhir kali ini, kelompok kami mengembangkan sebuah aplikasi kalkulator subnetting yang bertujuan untuk mempermudah perhitungan subnetting pada jaringan dengan menggunakan IP address dan prefix tertentu. Aplikasi ini dirancang untuk membantu pengguna, terutama mahasiswa dan profesional di bidang jaringan, dalam melakukan perhitungan secara cepat dan akurat.

Kontribusi saya dalam proyek ini yaitu pengembangan kode program dan perbaikan diagram UML. Pada bagian pengembangan kode, saya berperan dalam membuat algoritma utama yang digunakan untuk menghitung subnet mask, jumlah host, serta pembagian jaringan berdasarkan subnet. Selain itu, saya memastikan bahwa kode yang saya tulis terstruktur dengan baik, mudah dipahami oleh anggota tim lain, dan sesuai dengan prinsip OOP

Di sisi lain, saya juga berkontribusi dalam memperbaiki diagram UML untuk memastikan bahwa desain aplikasi mencerminkan hubungan antar kelas secara tepat. Fokus utama saya adalah memperbaiki dan memvalidasi hubungan **komposisi** dan **agregasi** antar kelas dalam diagram. Hal ini penting untuk memastikan bahwa setiap kelas memiliki tanggung jawab yang jelas.

Melalui proyek ini, saya tidak hanya mengembangkan keterampilan teknis dalam pengkodean dan desain UML, tetapi juga memperdalam pemahaman saya tentang konsep **komposisi** dan **agregasi** dalam Pemrograman Berorientasi Objek (PBO). Komposisi menggambarkan hubungan "has-a" yang kuat, di mana keberadaan suatu objek bergantung pada objek lainnya, sementara agregasi

merepresentasikan hubungan "has-a" yang lebih lemah, di mana objek dapat berdiri sendiri meskipun memiliki hubungan.

Secara keseluruhan, proyek ini memberikan pengalaman yang sangat berharga, baik dalam pengembangan aplikasi jaringan maupun dalam penerapan konsep PBO secara mendalam. Hasil akhirnya diharapkan tidak hanya berguna bagi kami sebagai pengembang, tetapi juga bermanfaat bagi pengguna yang membutuhkan alat bantu untuk menghitung subnetting dengan efisien.

3. Zharifa Nur Majidah (2320506021)

Dalam proyek akhir ini, saya berkontribusi pada pembuatan diagram UML, penyusunan laporan, dan presentasi PowerPoint. Pada tahap perancangan, saya bertanggung jawab untuk membuat diagram UML, termasuk class diagram dan use-case diagram, yang menjadi dasar desain aplikasi kalkulator subnetting. Diagram ini dirancang untuk menggambarkan hubungan antar kelas secara detail, seperti hubungan komposisi dan agregasi, sehingga menciptakan desain aplikasi yang terstruktur dan mudah dipahami oleh seluruh tim. Saya juga memastikan bahwa diagram ini dapat digunakan sebagai panduan yang jelas dalam proses pengembangan kode program agar setiap anggota tim dapat bekerja dengan arah yang sama.

Selain itu, saya bersama satu anggota tim lainnya berfokus pada penyusunan laporan akhir dan PowerPoint. Laporan ini mencakup seluruh detail penting, seperti tujuan proyek, metodologi yang digunakan, rancangan aplikasi, hingga hasil pengujian, untuk memberikan dokumentasi yang lengkap dan komprehensif. PowerPoint dirancang untuk menyampaikan informasi secara visual, ringkas, dan menarik, sehingga mampu menjelaskan proyek ini dengan baik kepada audiens. Meskipun tugas laporan dan PowerPoint menjadi tanggung jawab utama kami berdua, proses pengerjaannya bersifat fleksibel, di mana seluruh anggota tim dapat memberikan masukan dan turut berkolaborasi untuk menghasilkan hasil akhir yang lebih optimal.

Melalui proyek ini, saya tidak hanya meningkatkan kemampuan teknis dalam pengembangan aplikasi dan pembuatan diagram UML, tetapi juga memperdalam pemahaman saya tentang konsep dasar Pemrograman Berorientasi Objek (PBO), seperti komposisi dan agregasi. Komposisi menggambarkan hubungan "has-a" yang erat, di mana keberadaan satu objek sangat bergantung pada objek lain, sedangkan agregasi merepresentasikan hubungan "has-a" yang lebih longgar, di mana objek tetap dapat berdiri sendiri meskipun saling terkait. Secara

keseluruhan, proyek ini memberikan pengalaman yang sangat bermakna, baik dalam membangun aplikasi berbasis jaringan maupun dalam mengaplikasikan konsep-konsep PBO secara mendalam.

4. Nabila Kartika Putri (2320506039)

Pada penyelesaian proyek akhir ini saya berkontribusi dalam beberapa peran yang pertama adalah ikut adil dalam ide pembuatan produk, pada awal nya kami memiliki beberapa opsi untuk produk akhir yang kami kumpulkan menjadi 1 dan akhirnya kami memutuskan untuk membuat kalkulator subnet karena dirasa produk ini dapat kami rancang dengan pemahaman mengenai PBO yang sudah kami dapat selama satu semester serta produk ini juga kami rasa sangat bermanfaat

Setelah penentuan pembuatan produk kami membagi focus sehingga 2 orang fokus dalam pembuatan kode program dan 2 lainnya membuat laporan dan powerpoint, namun hal ini sangat kondisional dimana tidak ada larangan untuk ikut mengerjakan yang bukan focus nya. Saya berperan dalam melakukan checking terhadap kode dan memastikan system berjalan sesuai fungsi nya, pada awal nya terdapat kesulitan dalam kode ini dan perlu perbaikan kode namun akhirnya dapat berjalan dengan baik.

Setelah saya merasa system berjalan dengan baik dan fungsi nya saya lanjut membuat laporan dan power point laporan ini juga dikerjakan bersama agar semua memahami teori dari pembuatan kalkulator ini

Meskipun kalkulator ini tergolong sederhana, saya tetap mengalami beberapa kesulitan, terutama dalam penulisan kode dan penerapan fungsi-fungsi dalam setiap kelas di PBO. Namun, tantangan tersebut justru menjadi pengalaman berharga bagi saya, karena saya belajar banyak dalam menyelesaikan error yang muncul selama pengembangan. Proyek ini mengajarkan saya tentang pentingnya kolaborasi, kesabaran dalam memecahkan masalah, dan penerapan konsep PBO dalam dunia nyata.