Membuat aplikasi CRUD (Create, Read, Update, Dalate) menggunakan C++

Rukman Gaffar

Teknik Komputer, Politeknik Negeri Samarinda, 7542, Kota Samarinda

Nur Aini Fauziyah

Teknik Komputer, Politeknik Negeri Samarinda, 7542, Kota Samarinda

Email: rukmangaffar@gmail.com

nurainifauziyah992@gmail.com

Abstrak

C++ adalah bahasa pemrograman komputer yang dibuat oleh Bjarne Stroustrup merupakan perkembangan dari bahasa C yang dikembangkan di Bell Labs (Dennis Ritchie). Pada awal tahun 1970-an, bahasa itu adalah penambahan dari bahasa sebelumnya, yaitu C.[1] C++ dalam bahasa pemrograman yang berorientasi objek (OOP) terdapat istilah Class Library. Class library merupakan suatu fungsi yang dibuat untuk mempermudah programmer dalam membuat sebuah aplikasi. Dalam eksperimen, Class Library CRUD dibagi menjadi 4 bagian, yaitu: Class Create, Class Read, Class Update, Class Delete. Hasil dari penelitian ini, Class Library dapat mempermudah pembuatan aplikasi CRUD.

Kata kunci: C++, class library, CRUD, Object Oriented Programming

1. PENDAHULUAN

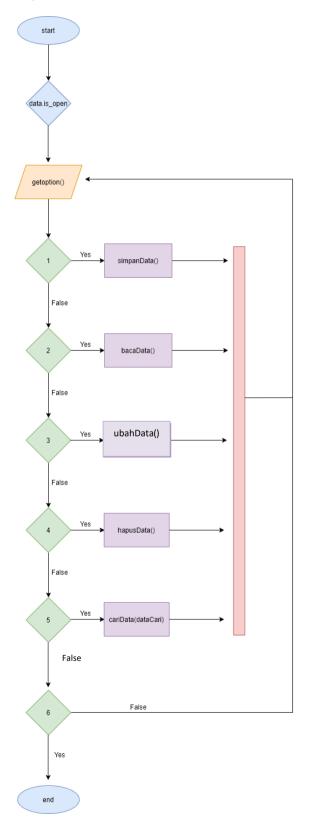
Pemrograman komputer di zaman sekarang semakin menjadi kepentingan bagi masyarakat dalam semua lini kehidupan. Semenjak perkembangan IT yang semakin melaju, hampir semuanya telah didigitalisasi dan penggunaan software/aplikasi merupakan aktivitas yang umum untuk semua kalangan.[2]

Pemrograman Berorientasi Objek atau Object Oriented Programming (OOP) adalah sebuah metode pembuatan program (programming paradigm) dengan menggunakan konsep "objek" yang memiliki data dan code. Data ini tersedia dalam bentuk atribut atau properti, dan code dalam bentuk prosedur.[3]

Kelebihan C++ yaitu, Proses kinerja lebih cepat, proteksi terhadap pustaka fungsi dan kelas yang banyak untuk memungkinkan pembuatan aplikasi makro, bersifat reusable yaitu bisa dimanfaatkan kembali pada project lain menggunakan file header dan library, bahasa pemrograman yang multi – paradigma atau dinamis, penulisan kode singkat dibandingkan bahasa pemrograman yang lain, memiliki sifat conveyable yang bisa digunakan untuk semua jenis komputer, Bahasa umumnya adalah ANSI, oleh karena itu C++ dapat digunakan diberbagai platform, Salah satu keunggulan yang dimiliki adalah sangat potensial jika dapat membuat aplikasi graphic processor dengan kualitas tinggi, kompiler bahasa C++ kebanyakan open source, lebih sistematis sehingga dapat menunjang OOP (objected – oriented – programming).[4]

2. METODOLOGI

Dalam peneletian kali ini, penulis membuat program CRUD berdasarkan topologi yang dapat anda lihat pada gambar 1:



Gambar 1. Topologi CRUD

Disini kita memerlukan header file <fstream> untuk melakukan operasi terhadap file eksternal.

Pada awal program kita akan disajikan menu dalam bentuk seperti :

Gambar 2. MENU

Proses tersebut terjadi pada bagian fungsi



Gambar 3. Getoption() pada topologi

Lalu kita harus memasukkan inputan berdasarkan menu dan akan memproses fungsi-fungsi yang telah kita buat. Jika kita memasukkan inputan dengan nilai 1, maka akan memproses fungsi simpanData(), seperti gambar dibawah:

```
Program CRUD data mahasiswa

1. Tambah data mahasiswa

2. Tampilkan data mahasiswa

3. Ubah data mahasiswa

4. Hapus data mahasiswa

5. Search data by NIM

6. Selesai

-------
pilih [1-6]? : 1

Menambah data mahasiswa

NIM : 206

Nama : Rukman

Tugas : 80

UTS : 90

UAS : 100
```

Gambar 3. Input data

```
Code untuk fungsi simpanData():

void simpanData(){

NilaiMahasiswa nilai;

nilai.inputNilai();

//simpan objek ke file

ofstream f;

f.open(namaFile, ios::binary|ios::app);

f.write( (char*) &nilai, sizeof(nilai));

f.close();
}
```

Proses yang terjadi pada fungsi itu adalah, awalnya kita memasukkan data kita pada nilai.inputNilai(); lalu data akan disimpan ke file dengan menggunakan ofstream dengan mode ios::binary untuk menulis dalam bentuk binary, dan ios::app untuk menulis diakhir baris.

Jika kita memasukkan inputan dengan nilai 2 akan memproses fungsi bacaData() untuk menampilkan data.

```
Program CRUD data mahasiswa
-----
1. Tambah data mahasiswa
Tampilkan data mahasiswa
3. Ubah data mahasiswa
4. Hapus data mahasiswa
Search data by NIM
Selesai
               _____
pilih [1-6]? : 2
Tampilkan data mahasiswa
NIM
                                    Nilai Akhir
                                                   Nilai Huruf
       Nama
              Tugas
                             UAS
                     UTS
206
       Rukman
              80
                      90
                             100
                                    90
```

Gambar 4. Input data

```
Code untuk fungsi bacaData():

void bacaData(){

NilaiMahasiswa nilai;

ifstream f;

f.open(namaFile, ios::binary);

cout << "NIM\t" << "Nama\t" << "Tugas\t" << "UTS\t" << "UAS\t" << "Nilai;

cout << Akhir\t" << "Nilai Huruf\t" << endl;

while (f.read((char*)&nilai, sizeof(nilai))){

nilai.outputNilai();

}

f.close();
```

Pada proses ini akan dilakukan looping dengan kondisi selama data dibaca sampai akhir, lalu setelah data dibaca akan ditampilkan.

Jika kita memilih 3, maka akan memproses untuk mengubah data. Seperti gambar dibawah:

```
Ubah data mahasiswa
Masukkan NIM dari Mahasiswa : 206
                                         Nilai Akhir
                                                          Nilai Huruf
        Nama
                Tugas
                        UTS
                                 UAS
206
        Rukman
                80
                         90
                                 100
                                                          Α
Silahkan masukkan data baru
NIM: 206
Nama : Rukman
Tugas: 70
UTS : 80
UAS : 70
Program CRUD data mahasiswa
1. Tambah data mahasiswa
2. Tampilkan data mahasiswa
Ubah data mahasiswa
4. Hapus data mahasiswa
5. Search data by NIM
Selesai
pilih [1-6]? : 2
Tampilkan data mahasiswa
                                         Nilai Akhir
                                                         Nilai Huruf
NIM
        Nama
                Tugas
                        UTS
                                 UAS
206
        Rukman
                70
                         80
                                 70
```

Gambar 5. Ubah data

```
Code untuk fungsi ubahData():
void ubahData(int nimDicari){
        NilaiMahasiswa nilai;
        fstream f;
        f.open(namaFile, ios::in|ios::out);
        while (f.read((char*) &nilai, sizeof(nilai))) {
                if(nilai.getNim() == nimDicari){
                         //tampil data sebelum update
                         cout << "NIM \backslash t" << "Nama \backslash t" << "Tugas \backslash t" << "UTS \backslash t" << "UAS \backslash t";
                         cout << "Nilai Akhir\t" << "Nilai Huruf\t" <<endl;</pre>
                         nilai.outputNilai();
                         //input data baru
                         cout << "Silahkan masukkan data baru" << endl;</pre>
                         nilai.inputNilai();
                         int pos = -1 * sizeof(nilai);
                         f.seekp(pos, ios::cur);
                         f.write ((char*) &nilai, sizeof(nilai));
                         break;
                 }
        }
f.close();
}
```

Pada proses ini, awalnya data di cek ada atau tidak, jika ada akan ditampilkan sebelum diubah, lalu data baru ditampung pada nilai.inputNilai(), lalu data dihapus sesuai tempat dari NIM dari yang kita masukkan.

Jika kita memasukkan nilai 4 akan menghapus data.

Gambar 6. Hapus data

```
void hapusData(int nimDicari){
       NilaiMahasiswa nilai;
       ifstream fi;
       fi.open(namaFile, ios::binary);
       ofstream fo;
       fo.open("tmp.dat", ios::out|ios::binary);
       int ketemu = 0;
       while(fi.read((char*) &nilai, sizeof(nilai))) {
          if(nilai.getNim() != nimDicari){
               fo.write((char*) &nilai, sizeof(nilai));
          }
               else {
                       ketemu = 1;
               }
       }
       fi.close();
       fo.close();
```

```
remove(namaFile);
rename("tmp.dat", namaFile);
if (ketemu == 1){
      cout << "Data dengan NIM " << nimDicari << " telah dihapus!" << endl;
} else {
      cout << "Data dengan NIM " << nimDicari << " Tidak ditemukan" << endl;
}
</pre>
```

Pada proses ini, kita perlu menggunakan ifstream dan ofstream. Disini akan dilakukan proses loping dengan kondisi selama mebaca data sampai selesai. Pada looping terjadi proses decision, apabila nilai dari data NIM pada index tidak sama dengan NIM yang kita cari, maka data asli akan ditulis ke dalam file sementara, disini saya menggunakan "tmp.dat" untuk menampung data baru, lalu data yang asli akan dihapus, lalu data "tmp.dat" akan diganti namanya dengan nama dari file asli.

Jika kita memasukkan inputan dengan nilai 5, maka akan menampilkan data berdasarkan NIM.

```
Program CRUD data mahasiswa

    Tambah data mahasiswa

Tampilkan data mahasiswa
Ubah data mahasiswa
4. Hapus data mahasiswa
Search data by NIM
6. Selesai
pilih [1-6]? : 5
Search data by NIM
Masukkan NIM dari Mahasiswa : 207
Nim 207 ditemukan
                                        Nilai Akhir
MIM
                        UTS
                                UAS
                                                         Nilai Huruf
        Nama
                Tugas
207
        Aini
                80
                        80
                                80
                                         80
```

Gambar 7. Cari data

```
Code untuk fungsi cariData();
void cariData(int nimDicari){
       NilaiMahasiswa nilai, nilaiketemu;
       int ketemu = 0;
       ifstream f:
       f.open(namaFile, ios::binary);
       while(f.read((char*)&nilai, sizeof(nilai))){
               if(nilai.getNim() == nimDicari){
                       nilaiketemu = nilai;
                       ketemu = 1;
                       break;
               }
        }
       if (\text{ketemu} == 0)
                      cout << "\nNim " << nimDicari << " tidak ditemukan" << endl;</pre>
               }else {
                      cout << "\nNim " << nimDicari << " ditemukan" << endl << endl;</pre>
                       cout << "NIM\t" << "Nama\t" << "Tugas\t" << "UTS\t" << "UAS\t";
                       cout << "Nilai Akhir\t" << "Nilai Huruf\t" << endl;
                       nilaiketemu.outputNilai();
               }
       f.close();
}
```

Pada proses ini kita memiliki suatu variable sebagai nilai untuk decision nanti, disini saya memiliki variable int ketemu dengan nilai 0 sebagai penanda kalau data saat ini belum ditemuan. Lalu terjadi looping untuk mengecek satu-satu pada data, apakah pada index nilai NIM-nya sesuai dengan NIM yang kita cari, jika iya maka nilai dari variable ketemu akan diubah menjadi 1, jika tidak akan tetap 0. Lalu setelah looping, terjadi decision dengan kondisi jika nilai dari ketemu itu 0 maka akan memberitahukan bahwa data yang dicari tidak ada, jika nilai dari ketemu bernilai 1 maka akan menampilkan data.

3. HASIL DAN PEMBAHASAN

Fungsi-fungsi dapat kita gunakan pada fungsi utama dengan mengkombinasikan dengan WHILE agar program tidak berlangsung sekali jalan saja, melainkan dapat digunakan terus menerus selama kita ingin kita gunakan, dan agar program berjalan sesuai dengan perintah.

```
hile(pilihan != FINISH){
   switch(pilihan){
        case CREATE:
            cout << "\nMenambah data mahasiswa" << endl;</pre>
            simpanData();
        break;
case READ:
            cout << "\nTampilkan data mahasiswa" << endl;</pre>
            bacaData();
        case UPDATE:
            cout << "\nUbah data mahasiswa" << endl;</pre>
            cout << "\nMasukkan NIM dari Mahasiswa : ";</pre>
            cin >> dataCari;
            ubahData(dataCari);
        case DELETE:
            cout << "\nHapus data mahasiswa" << endl;</pre>
            cout << "\nMasukkan NIM dari Mahasiswa : ";</pre>
            cin >> dataCari;
            hapusData(dataCari);
        case SEARCH:
            cout << "\nSearch data by NIM" << endl;
cout << "\nMasukkan NIM dari Mahasiswa : ";</pre>
            cin >> dataCari;
            cariData(dataCari);
            cout << "\nPilihan tidak ditemukan" << endl;</pre>
```

Gambar 7. Decision pada fungsi main()

Pada program ini akan menghasilkan program CRUD (Create, Read, Update, Delete) dengan kasus memasukkan nilai dari Mahasiswa, dengan metode ini kita juga dapat membuat program dengan kasus-kasus lainnya.

4. KESIMPULAN

Dari hasil penelitian dan uji project, telah dihasilkan suatu program CRUD yand dapat dijalankan pada Terminal ataupun Commnd Promp. Dengan pembuatan beberapa fungsi

Library ini dapat menjadikan proses pembuatan aplikasi CRUD menjadi lebih mudah dan cepat

dikarenakan programmer tidak perlu memakan waktu lama terhapa suatu code yang akan digunakan secara berulang. Projek kali ini dapat menjadi bahan latihan untuk program baca tulis kedalam file berupa CRUD. Adapun kekurangan program kali ini adalah terbatasnya table yang dibuat, disini kami hanya membuat table untuk mahasiswa. Diharapkan di kesempatan selanjutnya dapat dikembangkan lagi, misalnya ditambahkan beberapa table lain.

Daftar Pustaka

- [1] "C++ Wikipedia bahasa Indonesia, ensiklopedia bebas." https://id.wikipedia.org/wiki/C%2B%2B (accessed Jul. 14, 2021).
- [2] N. G. A. P. Harry Saptarini, R. A. Hidayat, and P. I. Ciptayani, "AJARINCODE: APLIKASI PEMBELAJARAN BAHASA PEMROGRAMAN BERBASIS WEB," *Just TI (Jurnal Sains Terap. Teknol. Informasi)*, vol. 10, no. 2, p. 21, Jul. 2019, doi: 10.46964/JUSTTI.V10I2.106.
- [3] "Pengertian Pemrograman Berorientasi Objek C++ | Duniailkom." https://www.duniailkom.com/tutorial-belajar-oop-c-plus-pengertian-pemrograman-berorientasi-objek/ (accessed Jul. 14, 2021).
- [4] "Kekurangan dan Kelebihan Yang Dimiliki C++ Josi Kie." https://josikie.com/kekurangan-dan-kelebihan-yang-dimiliki-c/ (accessed Jul. 14, 2021).
- [5] F. File, "File & Stream," vol. 2, no. 5, pp. 1–5.