

# **LAPORAN TUGAS BESAR**

## **PROGRAM PEMBELIAN TIKET PESAWAT**

Mata Kuliah : Pemrograman (A & B)  
Dosen Pengampu : Kadek Suar Wibawa. M.T.



Disusun Oleh :  
Syaripatul Aini (2105551012)  
I Gusti Ayu Sri Sahana Putri (2105551036)

**TEKNOLOGI INFORMASI**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS UDAYANA**  
**2021**

## **KATA PENGANTAR**

Puji syukur dipanjatkan kehadiran Tuhan Yang Maha Esa karena berkat dan rahmat-Nya sehingga kami dapat menyelesaikan tugas besar yang berjudul “Program Pembelian Tiket Pesawat” ini tepat pada waktunya.

Adapun tujuan dari penulisan laporan ini adalah untuk memenuhi tugas dosen pada mata kuliah pemrograman. Selain itu, laporan ini juga bertujuan untuk menambah wawasan tentang pemrograman bahasa C/C++ dan penggunaan github sebagai layanan kolaborasi dalam pembuatan program ini..

Kami mengucapkan terima kasih kepada Bapak Kadek Suar Wibawa. M.T., selaku dosen mata kuliah pemrograman yang telah memberikan tugas ini sehingga dapat menambah pengetahuan dan wawasan sesuai dengan bidang studi yang saya tekuni.

Kami juga mengucapkan terima kasih kepada semua pihak yang telah membagi sebagian pengetahuannya sehingga kami dapat menyelesaikan laporan ini.

Kami menyadari, laporan dan program yang kami buat ini masih jauh dari kata sempurna. Oleh karena itu, kritik dan saran yang membangun akan kami nantikan demi kesempurnaan laporan dan program ini.

Gianyar, 29 November 2021

Penulis

# **DAFTAR ISI**

## [DAFTAR ISI](#)

### [BAB I PENDAHULUAN](#)

#### [1.1 Latar Belakang](#)

#### [1.2 Tujuan](#)

#### [1.3 Identifikasi Masalah](#)

### [BAB II LANDASAN TEORI](#)

### [BAB III HASIL DAN PEMBAHASAN](#)

#### [3.1 Penambahan Repository dan Commit di Github](#)

#### [3.3 Tampilan Program](#)

### [BAB IV PENUTUP](#)

#### [4.1 Kesimpulan](#)

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Pemesanan tiket adalah salah satu proses yang sering banyak orang lakukan sebelum melaksanakan suatu perjalanan ataupun suatu keberangkatan. Khususnya tiket untuk melakukan perjalanan udara dengan menggunakan pesawat. Berbagai cara yang dilakukan para calon penumpang dalam melakukan pemesanan tiket, baik dengan cara memesan langsung ke maskapai penerbangan maupun kepada agen terdekat dengan lokasi calon penumpang. Namun kedua proses pemesanan tersebut kurang efektif, baik dalam segi waktu maupun biaya. Serta dari beberapa web pemesanan tiket pesawat online yang sudah ada terlalu sulit untuk dioperasikan terutama bagi yang lanjut usia, sehingga tidak paham untuk mengoperasikannya. Untuk itulah perlu adanya proses pemesanan untuk lebih mengefektifkan baik dalam segi waktu ataupun biaya itu sendiri serta lebih memudahkan, lebih praktis sehingga dapat dioperasikan oleh semua kalangan dan lebih cepat tentunya dalam melakukan pemesanan tiket.

Maka atas dasar itulah, penulis merasa perlu untuk membuat suatu program untuk memberikan kemudahan masyarakat yang akan melakukan pemesanan tiket tanpa ada batasan ruang dan waktu dalam melakukan pemesanan tiket. Maka solusinya adalah membuat program yang berjudul: “Program Pemesanan Tiket Pesawat”.

### **1.2 Tujuan**

Program ini dibuat oleh penulis dengan tujuan :

1. Pemenuhan tugas besar mata kuliah Pemrograman
2. Untuk mempermudah konsumen melakukan pemesanan tiket pesawat
3. Agar konsumen dapat menghemat waktu dan biaya

### **1.3 Identifikasi Masalah**

Berdasarkan uraian pada latar belakang dapat diidentifikasi beberapa masalah sebagai berikut:

- a. Sulit mendapatkan informasi reservasi tiket pesawat
- b. Kurang efisiennya dalam waktu & biaya karena konsumen harus datang langsung ke tempat pembelian tiket.

## **BAB II**

### **LANDASAN TEORI**

C++ adalah sebuah pemrograman yang memiliki banyak dialek, seperti bahasa orang yang banyak memiliki dialek. Dalam C++, dialek bukan disebabkan oleh karena si pembicara berasal dari Jepang atau Indonesia, melainkan karena bahasa ini memiliki beberapa compiler yang berbeda. Ada empat compiler umum yaitu : C++ Borland, C++ Microsoft Visual, C/386 Watcom, dan DJGPP. Anda dapat mendownload DJGPP atau mungkin saja anda telah memiliki compiler lain.

Setiap compiler ini agak berbeda. Setiap compiler akan dapat menjalankan fungsi-fungsi standar C++ ANSI/ISO, tetapi masing-masing compiler juga akan dapat menjalankan fungsi-fungsi non standar (fungsi-fungsi ini, agak mirip dengan ucapan yang tidak standar yang diucapkan orang di berbagai pelosok negeri. Sebagai contoh, di New Orleans kata median disebut neutral ground). Kadang-kadang pemakaian fungsi non standar akan menimbulkan masalah pada saat anda hendak mengkompilasi kode sumber data (source code) (yaitu program berbahasa C++ yang ditulis oleh seseorang programmer) menggunakan compiler yang berbeda. Tutorial ini tidak lepas dari masalah seperti itu.

Bila anda belum mempunyai sebuah compiler, disarankan agar anda memiliki sebuah compiler. Sebuah compiler sederhana sudah cukup untuk dipergunakan oleh anda dalam mengikuti tutorial ini.

Bahasa programan C++ adalah bahasa yang amat berbeda. Untuk compiler C++ berbasis DOS, akan memerlukan beberapa kata kunci (keywords) ; keywords sendiri tidak cukup untuk difungsikan sebagai input dan output. Walau hampir semua fungsi dalam file library tampaknya biasa diakses oleh header filenya.

Tanda `#include` adalah sebuah prosesor pengarah yang mengatakan kepada compiler untuk meletakkan kode header file `iostream.h` kedalam program. Dalam menyertakan header file, anda dapat mengakses banyak fungsi-fungsi.

Dalam contoh diatas, fungsi `cout` memerlukan file `iostream.h` pada baris berikutnya yang penting adalah `int main ()`. Baris ini mengatakan kepada compiler bahwa ada sebuah fungsi bernama `main`, yang mana fungsi itu mengembalikan sebuah integer, sehingga diberi tanda `int`. Tanda kurung (`{` dan `}`) menandakan awal dan akhir fungsi dalam program diatas dan menghentikan kode lainnya. Jika anda memakai bahasa Pascal, anda akan menggunakan perintah `BEGIN` dan `END`. Baris berikutnya dari program diatas

agak aneh. Jika anda menulis sebuah program menggunakan bahasa lain, anda akan mengetahui bahwa perintah print akan menampilkan text dilayar monitor. Dalam bahasa C++ tidaklah demikian, pemakaian fungsi cout dipakai untuk menampilkan text dilayar monitor anda. Itu juga memakai tanda atau symbol <<, yang diketahui sebagai operator pemasukan (insertion operators).

Tanda tersebut mengatakan kepada compiler agar segera menghasilkan output sesuai dengan input anda. Tanda semicolon ditambahkan dalam bagian akhir dari semua fungsi yang di panggil dalam bahasa C++; tanda seterusnya memperlihatkan variable yang anda deklarasikan. Pada baris itu juga ada kode yang memerintahkan fungsi main kembali ke nol. Pada saat satu kali kembali di proses, itu dilakukan melalui system operasi. Sebagai catatan, pendeklarasian fungsi main yang tidak di inginkan memiliki proses kembali, ditambahkan void main()- dan biasanya itu berfungsi dengan baik ; namun cara ini merupakan cara yang kurang baik.

Tanda kurung terakhir berperan sebagai penutup agar fungsi di hentikan. Anda biasa mencoba mengoprasikan fungsi ini dalam sebuah compiler. Anda dapat melakukan cut dan paste kode diatas kedalam sebuah file, dan menyimpannya sebagai file. Cpp, dan kemudian anda buka file itu dari dalam compiler anda. Jika anda memakai baris perintah compiler seperti yang dalam bordlan C++ 5.5, sebaiknya anda membaca dahulu intruksi compiler tentang bagaimana cara melakukan kompilasi (how to compiler).

Command atau computer sering dipakai dalam program. Ketika anda mengatakan kepada compiler bahwa sebuah bagian dari program anda adalah text komentar, compiler tidak akan memasukan itu sebagai perintah program. Untuk membuat sebuah komentar dipergunakan tanda //, yang mengakan kepada compiler bahwa baris berikut adalah komentar, memakai tanda /\* dan kemudian diakhiri tanda \*/ untuk menandai bahwa segala yang ada antara kedua tanda ini adalah komentar. Beberapa compiler akan mengubah warna komentar, tetapi beberapa compiler lain tidak. Berhati-hatila dalam menulis program anda agar kode program anda tidak dianggap sebagai komentar oleh compiler karena dapat mempengaruhi output program yang anda buat. Selanjutnya anda harus dapat menuliskan program sederhana untuk menampilkan informasi yang anda ketikan kedalamnya. Selain itu, program juga bias dibuat untuk menerima input. Fungsi yang di pakai adalah cin, dan di ikuti dengan tanda >>.

Tentu saja anda sebelum anda mencoba menerima input, anda harus memiliki tempat penyimpanan input. Dalam pemrograman, pada saat anda ingin mengatakan kepada compiler sebuah variable yang anda deklarasikan, anda harus menyertakan tipe

data dan nama dari variable. Beberapa cara dasar adalah menuliskan include care, ini, dapat float.

Sebuah variable char menyimpan sebuah karakter tunggal; variable int akan menyimpan integer (bukan bilangan decimal), dan variable float akan menyimpan bilangan decimal. Setiap variable-char, int, dan float. Merupakan sebuah kata kunci (keyword) yang anda pergunakan pada saat anda mendeklarasikan sebuah variable. Untuk mendeklarasikan sebuah variable anda memakai syntax type. Ini di ijin untuk mendeklarasikan variable multiple dari jenis variable yang sama pada baris yang sama; masing-masing variabelnya di pisahkan dengan tanda koma. Deklarasi sebuah variable atau sekelompok variable dapat di ikuti tanda semicolon (catatan, tanda ini sama dengan yang diterapkan pada prosedur dimana anda akan memanggil sebuah fungsi).

Jika anda tidak memakai sebuah variable yang di deklarasi sebelumnya. Program anda tidak dapat dioperasikan (atau di run), dan anda akan menerima pesan error yang memberitahu anda bahwa anda telah melakukan kesalahan.

Platform Github adalah layanan web host untuk kebutuhan pengembangan perangkat lunak di kalangan back end developer dengan sistem kendali Git Version. Pada praktiknya, platform ini mampu memberikan akses kontrol dan kolaborasi antar tiap developer untuk dapat manajemen tugas, perbaikan bug, dan pembuatan dokumentasi pada proyek.

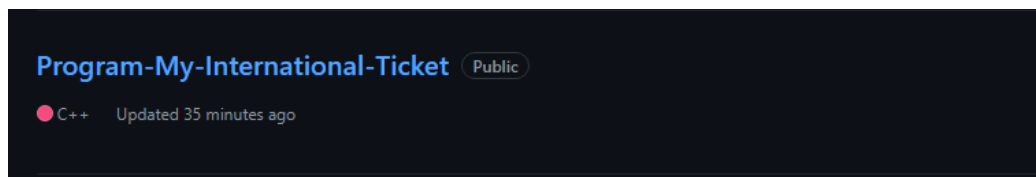
Github juga menawarkan paket repositori secara personal dan gratis pada penggunaan akun yang sama. Selain itu juga tersedia dengan paket open source untuk memudahkan pengguna dalam mengakses berbagai fitur di dalam situs tersebut.

Pada bulan April 2017 lalu, jumlah total pengguna dari layanan Github telah mencapai lebih dari 20 juta pengguna, serta lebih dari 57 juta repository. Sehingga, membuat platform ini menjadi penyedia source code terbesar di dunia.

## BAB III

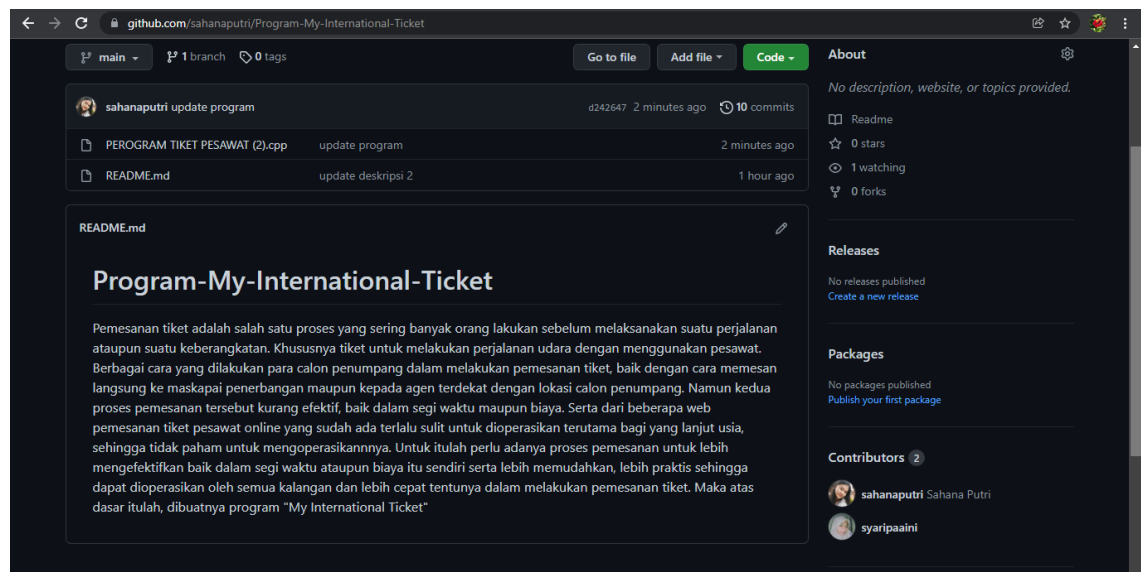
### HASIL DAN PEMBAHASAN

#### 3.1 Penambahan Repository dan Commit di Github



*gambar 1*

Pada gambar 1, adalah langkah awal kita yaitu, kita membuat repository baru di github dengan nama repository Program-My-International-Ticket



*gambar 2*

Pada gambar 2, kita terlebih dahulu menambahkan program awal yang sudah kita buat sebelumnya.



Berikut merupakan tampilan program yang telah kita buat dan commit di github :

```

1 #include <iostream> //header input output
2 #include <conio.h> //header getch
3 #include <windows.h> //header system cls
4 #include <fstream> //header membuka dan menutup file
5 #include <string.h> //header operasi string
6 #include <string> //header fungsi string
7 #include <iomanip> //header pemerataan io set flags
8 #include <time.h> //Header yang di gunakan untuk operasi pada bahasa C
9 time_t waktuuser;
10 using namespace std; //penganti std::
11 char namaFile[]="loginRecord.txt";
12
13 struct user{
14     char nama [50];
15     char username [32];
16     char password [30];
17     char email [50];
18 }u;
19
20 struct {
21     char *nama[50];
22     char *telepon[13];
23     char *alamat[70];
24 }Identitas;
25
26 void waktu(){
27     time(&waktuuser);
28     struct tm*waktu = localtime (&waktuuser);
29     printf ("\\n\\t\\t\\t||=====Tanggal: %d/%d/%d=====||\\n\\n", waktu -> tm_mon +1, waktu -> tm_year +1900);
30 }
31
32 void masuk(){

```

```

26 void waktu();
27     time(&waktu);
28     struct tm *waktu = localtime (&waktu);
29     printf ("\n\t\t\t\t\t-----Tanggal: %d/%d/%d-----\n\n", waktu -> tm_mday, waktu -> tm_mon +1, waktu -> tm_year +1900);
30 }
31
32 void masuk(){
33     void daftar();
34     /* Variabel username[12] digunakan untuk menyimpan input username
35     yang dilakukan oleh pengguna. Inputan tersebut akan divalidasi
36     dengan username yang sudah terdaftar pada file loginRecord.txt */
37     char username[12];
38
39     /* Variabel password[10] digunakan untuk menyimpan input password
40     yang dilakukan oleh pengguna. Inputan tersebut akan divalidasi
41     dengan password yang sudah terdaftar pada file loginRecord.txt */
42     char password[10];
43
44     // Variabel pilihan untuk pilihan user apakah ingin mendaftar, masuk ulang, atau keluar
45     int pilihan;
46
47     // Membuat pointer record untuk menunjuk pada file "loginRecord.txt"
48     FILE *record;
49
50     // Membuka file dengan mode "r"
51     record = fopen (namafile, "r");
52     if (record == NULL){
53         fputs ("Error, tidak terdapat file untuk menyimpan akun\n", stderr);
54         exit(1);
55     }
56
57     // Instruksi untuk login

```

```

195 void tampilanAwal () //membuat garis tabel awal tampilan
196 {
197     cout << "*****" << endl;
198     cout << "                                WELCOME TO MY INTERNATIONAL TICKET                               " << endl;
199     cout << "*****" << endl;
200 }
201 
202 void tampilanPesawat () //tabel pesawat yang tersedia
203 {
204     cout << endl;
205     cout<<endl;
206     cout<<"\n\t\t*"-----"*";
207     cout<<"\n\t\t| Nama Pesawat       | Jam Berangkat   | Harga Tiket    |";
208     cout<<"\n\t\t|-----|-----|-----|";
209     cout<<"\n\t\t| 1. Garuda             |      08.30     | Rp. 1.600.000,- |";
210     cout<<"\n\t\t|-----|-----|-----|";
211     cout<<"\n\t\t| 2. Airbus              |      10.00     | Rp. 1.700.000,- |";
212     cout<<"\n\t\t|-----|-----|-----|";
213     cout<<"\n\t\t| 3. Lion Air            |      12.30     | Rp. 1.400.000,- |";
214     cout<<"\n\t\t|-----|-----|-----|";
215     cout<<"\n\t\t| 4. Air Asia            |      14.00     | Rp. 1.500.000,- |";
216     cout<<"\n\t\t|-----|-----|-----|";
217     cout<<"\n\t\t*"-----"*";
218 }
219 
220 void orang () //memasukan nama pembeli tiket
221 {
222     cin.ignore();
223     cout << "\n Client * <a << endl;
224     cout << "    Nama Lengkap      : "; getline(cin, c1.nama[a]);
225     c1.org = &c1.nama[a];

```

```
#include <iostream> //header input output
#include <conio.h> //header getch
#include <windows.h> //header system cls
#include <fstream> //header membuka dan menutup file
#include <string.h> //header operasi string
#include <string> //header fungsi string
#include <iomanip> //header pemerataan io set flags
#include <time.h> //Header yang di gunakan untuk operasi
pada bahasa C
time_t waktuserver;
using namespace std; //pengganti std::
char namaFile[]="loginRecord.txt";

struct user{
    char nama [50];
    char username [12];
    char password [10];
    char email [50];
}u;

struct {
    char *nama[50];
    char *telepon[13];
    char *alamat[70];
}Identitas;

void waktu(){
    time(&waktuserver);
    struct tm*waktu = localtime (&waktuserver);
    printf ("\n\t\t\t\t\t|=====Tanggal:
%d/%d/%d=====||\n\n", waktu -> tm_mday, waktu ->tm_mon +1,
waktu -> tm_year +1900);
}

void masuk(){
    void daftar();
    /* Variabel username[12] digunakan untuk menyimpan input
username
yang dilakukan oleh pengguna. Inputan tersebut akan
divalidasi
dengan username yang sudah terdaftar pada file
loginRecord.txt */
    char username[12];

    /* Variabel password[10] digunakan untuk menyimpan input
password
yang dilakukan oleh pengguna. Inputan tersebut akan
divalidasi
dengan password yang sudah terdaftar pada file
loginRecord.txt */
    char password[10];

    // Variabel pilihan untuk pilihan user apakah ingin mendaftar,
masuk ulang, atau keluar
    int pilihan;
```

```

    // Membuat pointer record untuk menunjuk pada file
    "loginRecord.txt"
    FILE *record;

    // Membuka file dengan mode "r"
    record = fopen (namaFile, "r");
    if (record == NULL){
        fputs ("Error, tidak terdapat file untuk menyimpan
akun\n", stderr);
        exit(1);
    }

    // Instruksi untuk login
    printf ("\t\t\t\t|=====||\n");
    printf ("\t\t\t\t| LOGIN\n");
    printf ("\t\t\t\t|=====||\n");
    printf ("\t\t\t\t| Username: ");
    scanf ("%s", username);
    printf ("\t\t\t\t|=====||\n");
    printf ("\t\t\t\t| Password: ");
    scanf ("%s", password);
    printf ("\t\t\t\t|=====||\n");

    // Membersihkan layar terminal
    system ("cls");

    // Membeaca file "logRecord.txt"
    while (fread(&u, sizeof(u),1,record)){
        /* Mengkomparasi username serta password yang diinput
user ke dalam fungsi
        masuk() dengan username serta password yang sudah
terdaftar */
        if(strcmp(username, u.username)==0 && strcmp(password,
u.password)==0){
            printf ("\t\t\t\t|=====||\n");
            printf ("\t\t\t\t| Login Berhasil! ||\n");
            printf ("\t\t\t\t|=====||\n");
        }
        // Jika akun tidak terdaftar maka akan diberikan pilihan
untuk mendaftar, masuk ulang, atau keluar
        else {
            printf ("\t\t\t\t|=====||\n");
            printf ("\t\t\t\t| Akun Tidak Terdaftar! ||\n");
            printf ("\t\t\t\t|=====||\n");

            instruksi:
            printf ("\t\t\t\t|=====||\n");
            printf ("\n\t\t\t\t|Ketik 1 untuk Daftar");
            printf ("\n\t\t\t\t|Ketik 2 untuk Masuk Ulang");
            printf ("\n\t\t\t\t|Ketik 3 untuk Keluar\n");
            printf ("\n\t\t\t\t|=====||\n");
            printf ("\n\t\t\t\t|Pilihan : ");
            scanf ("%d", &pilihan);
            system ("cls");
            if (pilihan == 1){

```

```

        daftar();
    }
    else if (pilihan == 2){
        masuk();
    }
    else if (pilihan == 3){
        exit(1);
    }
    else {
        printf ("Error! Silahkan Masukkan Pilihan dengan
Benar \n");
        goto instruksi;
    }
}
}
fclose (record);
return;
}

void daftar(){
    // Membuat pointer regis untuk file "loginRecord.txt"
    FILE *regis;
    // Membuka file "loginRecord.txt" dengan mode "w"
    regis = fopen(namaFile, "w");
    if (regis == NULL){
        fputs ("ERROR, Tidak terdapat file untuk menyimpan
data!\n", stderr);
    }

    //Instruksi untuk melakukan registrasi akun
    printf ("\n\t\t\t\t\t Silahkan Registrasi Akun Anda! \n\n");
    getchar ();
    printf ("\n\t\t\t\t\t Nama Lengkap: ");
    gets (u.nama);
    printf ("\n\t\t\t\t\t Email: ");
    scanf ("%s", u.email);
    printf ("\n\t\t\t\t\t Hallo %s! \n", u.nama);
    printf ("\n\t\t\t\t\t Silahkan membuat username (max 12 karakter),
serta password (max 10 karakter) \n");

    // Instruksi untuk menginput username serta password
    printf ("\n\t\t\t\t\t Masukkan Username : ");
    scanf ("%s", u.username);
    printf ("\n\t\t\t\t\t Masukkan Password : ");
    scanf ("%s", u.password);

    // Menuliskan nama yang diinput oleh pengguna kedalam file
"loginRecord.txt"
    fwrite (&u, sizeof(u),1,regis);
    fclose (regis);
    printf ("\n\t\t\t\t\t Registrasi Selesai! ");
    printf ("\n\t\t\t\t\t Akun Anda Terdaftar, Silahkan Mencoba!");

    // Membersihkan layar terminal
    system("cls");
    // Memanggil fungsi masuk()

```

```

        masuk ();
    }

void pilmenu(){
    //variabel pilihan untuk menyimpan pilihan yang diinput user
    pada switch case
    int pilihan;

    instruksi:
    printf ("\t\t\t\t|=====||\n");
    printf ("\t\t\t\t|          PILIHAN MENU MASUK   ||\n");
    printf ("\t\t\t\t|=====||\n");
    printf ("\t\t\t\t|      No   |   Menu Masuk   ||\n");
    printf ("\t\t\t\t|=====||\n");
    printf ("\t\t\t\t|      1   |   Masuk         ||\n");
    printf ("\t\t\t\t|      2   |   Registrasi    ||\n");
    printf ("\t\t\t\t|      4   |   Keluar          ||\n");
    printf ("\t\t\t\t|=====||\n");
    printf ("\t\t\t\t|=====||\n");
    printf ("\t\t\t\t| Masukkan pilihan anda : ");
    scanf ("%d", &pilihan);
    printf ("\t\t\t\t|=====||\n");
    system ("cls");

    /* Percabangan Switch dengan kondisi nilai yang disimpan pada
    variabel
    pilihan untuk menuju ke pilihan menu masuk yang diinput
    user */
    switch (pilihan){
        case 1:
            masuk();
            break;
        case 2:
            daftar();
            break;
        case 3:
            exit(1);
            break;
        default:
            printf ("Error! Silahkan Masukkan Plihan dengan Benar
\n");
            break;
    }
}

int a, b, c, d, jml, jum;
int byr, kembalian;
int totPem = 0;

struct client
{
    string nama[30];
    string *org;
}cl;

```

```

struct atasNama
{
    string nama[30][30];
    int umur[10][10];
}an;

struct pesawat
{
    int tiket[10];
    float diskon[10];
    string tujuan[15][15], kelas, kPenerbangan;
    char pswt[10], pil[10][10], pil2[10][10];
}t;

void tampilanAwal () //membuat garis tabel awal tampilan
{
    cout <<
    "=====
=====
    cout << "                                WELCOME TO
MY INTERNATIONAL TICKET
    << endl;
    cout <<
    "=====
=====
    << endl;
}

void tampilanPesawat () //tabel pesawat yang tersedia
{
    cout << endl;
    cout<<endl;

    cout<<"\n\t\t*-----*-----*-----
-----*";
    cout<<"\n\t\t| Nama Pesawat          | Jam Berangkat          | Harga
Tiket          |";

    cout<<"\n\t\t|-----|-----|-----
-----|";
    cout<<"\n\t\t| 1. Garuda              |          08.30          |
Rp. 1.600.000,- |";

    cout<<"\n\t\t|-----|-----|-----
-----|";
    cout<<"\n\t\t| 2. Airbus              |          10.00          |
Rp. 1.700.000,- |";

    cout<<"\n\t\t|-----|-----|-----
-----|";
    cout<<"\n\t\t| 3. Lion Air            |          12.30          |
Rp. 1.400.000,- |";

    cout<<"\n\t\t|-----|-----|-----
-----|";

```

```

        cout<<"\n\t\t| 4. Air Asia          |          14.00          |
Rp. 1.500.000,- |";

cout<<"\n\t\t|-----|-----|-----
-----|";

cout<<"\n\t\t*-----*-----*-----
-----*";
}

void orang () //pemasukan nama pembeli tiket
{
    cin.ignore();
    cout << "\n    Client " << a << endl;
    cout << "    Nama Lengkap      : "; getline(cin, cl.nama[a]);
    cl.org = &cl.nama[a];
    cout << "    Alamat dari nama client : " << cl.org << endl;
    cout << "    Jumlah Penumpang : "; cin >> jum;
    if (jum > 1)
    {
        for (d = 1; d < jum; d++)
        {
            cin.ignore();
            cout << "    Nama Lengkap      : "; getline(cin,
an.nama[a][d]);
            cout << "    Umur" << endl;
            cout << "    1. Dewasa (Lebih dari 14 Tahun)" << endl;
            cout << "    2. Anak-Remaja    (7 Tahun sampai 14 Tahun)"
<< endl;
            cout << "    3. Balita (Kurang dari 7 Tahun)" << endl;
            cout << "    Pilihan Umur : "; cin >> an.umur[a][d];
        }
    }
}

void pilihanTujuan () //menampilkan list Tujuan tujuan
{
    cout << "    Tujuan Keberangkatan:\n";
    cout << "    1. BANJARMASIN-JAKARTA\n";
    cout << "    2. BANJARMASIN-MAKAASAR\n";
    cout << "    3. BANJARMASIN-BALI\n";
    cout << "    4. BANJARMASIN-SURABAYA\n";
    cout << "    5. BANJARMASIN-PONTIANAK\n";
}

string Tujuan () //untuk pengeluaran output hasil dari pemilihan
tujuan
{
    if (t.pil[a][b] == '1')
    {
        t.tujuan[a][b] = "BANJARMASIN-JAKARTA";
    }
    else if (t.pil[a][b] == '2')
    {
        t.tujuan[a][b] = "BANJARMASIN-MAKAASAR";
    }
}

```

```

else if (t.pil[a][b] == '3')
{
    t.tujuan[a][b] = "BANJARMASIN-BALI";
}
else if (t.pil[a][b] == '4')
{
    t.tujuan[a][b] = "BANJARMASIN-SURABAYA";
}
else if (t.pil[a][b] == '5')
{
    t.tujuan[a][b] = "BANJARMASIN-PONTIANAK";
}

return t.tujuan[a][b];
}

string kode () //kode penerbangan saat diprint out di teks
{
    if (t.pswt[a] == '1')
    {
        if (t.pil[a][b] == '1')
        {
            t.kPenerbangan = "SI-BJ";
        }
        else if (t.pil[a][b] == '2')
        {
            t.kPenerbangan = "SI-BM";
        }
        else if (t.pil[a][b] == '3')
        {
            t.kPenerbangan = "SI-BB";
        }
        else if (t.pil[a][b] == '4')
        {
            t.kPenerbangan = "SI-BS";
        }
        else if (t.pil[a][b] == '5')
        {
            t.kPenerbangan = "SI-BP";
        }
    }
    else if (t.pswt[a] == '2')
    {
        if (t.pil[a][b] == '1')
        {
            t.kPenerbangan = "YA-BJ";
        }
        else if (t.pil[a][b] == '2')
        {
            t.kPenerbangan = "YA-BM";
        }
        else if (t.pil[a][b] == '3')
        {
            t.kPenerbangan = "YA-BB";
        }
        else if (t.pil[a][b] == '4')

```



```

        {
            t.kPenerbangan = "YA-BS";
        }
        else if (t.pil[a][b] == '5')
        {
            t.kPenerbangan = "YA-BP";
        }
    }
    else if (t.pswt[a] == '3')
    {
        if (t.pil[a][b] == '1')
        {
            t.kPenerbangan = "SA-BJ";
        }
        else if (t.pil[a][b] == '2')
        {
            t.kPenerbangan = "SA-BM";
        }
        else if (t.pil[a][b] == '3')
        {
            t.kPenerbangan = "SA-BB";
        }
        else if (t.pil[a][b] == '4')
        {
            t.kPenerbangan = "SA-BS";
        }
        else if (t.pil[a][b] == '5')
        {
            t.kPenerbangan = "SA-BP";
        }
    }
    else if (t.pswt[a] == '4')
    {
        if (t.pil[a][b] == '1')
        {
            t.kPenerbangan = "LA-BJ";
        }
        else if (t.pil[a][b] == '2')
        {
            t.kPenerbangan = "LA-BM";
        }
        else if (t.pil[a][b] == '3')
        {
            t.kPenerbangan = "LA-BB";
        }
        else if (t.pil[a][b] == '4')
        {
            t.kPenerbangan = "LA-BS";
        }
        else if (t.pil[a][b] == '5')
        {
            t.kPenerbangan = "LA-BP";
        }
    }
    return t.kPenerbangan;
}

```

```

string tipe () //kelas pesawat
{
    if (t.pil2[a][b] == '1')
    {
        t.kelas = "Ekonomi";
    }
    else if (t.pil2[a][b] == '2')
    {
        t.kelas = "Bisnis";
    }
    return t.kelas;
}

void pilihanKelas () //tabel pilihan kelas pesawat ekonomi atau
ekslusif
{
    cout << endl;
    cout << "    Kelas Pesawat\n 1. Ekonomi\n 2. Bisnis\n";
}

void dataTiket () // pemasukan data pilihan tujuan penerbangan
{
    ulang:
    cout << "    Pilihan : "; cin >> t.pil2[a][b];
    if (t.pil2[a][b] == '1')
    {
        cout << "    Tiket Regular, tidak ada tambahan biaya"
<< endl;
    }
    else if (t.pil2[a][b] == '2')
    {
        cout << "    Tambahan biaya Rp. 500.000,-" << endl;
    }
    else
    {
        cout << "    Salah Input\n";
        goto ulang;
    }
}

int hargaTiket () //harga tiket
{
    if (t.pswt[a] == '1')
    {
        if (t.pil2[a][b] == '1')
        {
            t.tiket[a] = 1600000 * jum;
        }
        else if (t.pil2[a][b] == '2')
        {
            t.tiket[a] = 1600000 * jum + 500000;
        }
    }
    if (t.pswt[a] == '2')
    {

```

```

        if (t.pil2[a][b] == '1')
        {
            t.tiket[a] = 1700000 * jum;
        }
        else if (t.pil2[a][b] == '2')
        {
            t.tiket[a] = 1700000 * jum + 500000;
        }
    }
    if (t.pswt[a] == '3')
    {
        if (t.pil2[a][b] == '1')
        {
            t.tiket[a] = 1400000 * jum;
        }
        else if (t.pil2[a][b] == '2')
        {
            t.tiket[a] = 1400000 * jum + 500000;
        }
    }
    if (t.pswt[a] == '4')
    {
        if (t.pil2[a][b] == '1')
        {
            t.tiket[a] = 1500000 * jum;
        }
        else if (t.pil2[a][b] == '2')
        {
            t.tiket[a] = 1500000 * jum + 500000;
        }
    }
    return t.tiket[a];
}

float diskon()
{
    if (hargaTiket() > 15000000)
    {
        t.diskon[a] = hargaTiket() * 0.4;
    }
    else if (hargaTiket() > 7500000 && hargaTiket() <= 15000000)
    {
        t.diskon[a] = hargaTiket() * 0.25;
    }
    else if (hargaTiket() > 5000000 && hargaTiket() <= 7500000)
    {
        t.diskon[a] = hargaTiket() * 0.15;
    }
    else
    {
        t.diskon[a] = 0;
    }
    return t.diskon[a];
}

int hTiket() //rumus total harga tiket

```

```

{
    totPem = hargaTiket() - diskon();
    return totPem;
}

void tampilanAkhir () //menampilkan hasil dari proses
{
    for (a = 1; a <= jml; a++)
    {
        for (b = 1; b <= 1; b++)
        {
            cout << "    Client " << cl.nama[a] << " Membeli tiket
" << Tujuan() << " dengan pesawat " << kode();
            cout << " kelas " << tipe() << "\n    seharga Rp. " <<
hTiket() << ",-" << endl;
        }
    }
}

void printTabel () //print hasil proses
{
    ofstream file ("Retail.txt");
    cout << endl;
    if (!file.is_open()){
        cout << "Tidak bisa membuka file";
    }
    else {
        cout << endl;
        cout <<
"=====
"
<< endl;
        file <<
"=====
"
<< endl;
        cout << " No. |          Nama Client          |
Tujuan    |          Pesawat          |
Harga     |" << endl;
        file << " No. |          Nama Client          |
Tujuan    |          Pesawat          |
Harga     |" << endl;
        cout << endl;

        for (a = 1; a <= jml; a++)
        {
            cout << endl;
            cout << setiosflags(ios::right) << setw(2) << a <<
setiosflags(ios::right) << setw(26) << cl.nama[a];
            file << setiosflags(ios::right) << setw(2) << a <<
setiosflags(ios::right) << setw(26) << cl.nama[a];
            for (b = 1; b <= 1; b++)
            {
                totPem = totPem + hTiket();
                cout << setiosflags(ios::left) << setw(31) <<
Tujuan() << setiosflags(ios::left) << setw(13) << kode();

```

```

        file << setiosflags(ios::left) << setw(31) <<
Tujuan() << setiosflags(ios::left) << setw(13) << kode();
        cout << setiosflags(ios::left) << setw(20) <<
tipe() << setiosflags(ios::left) << setw(16) << "Rp. " << hTiket()
<< ",-" << setiosflags(ios::left) << setw(11) << "|" << endl;
        file << setiosflags(ios::left) << setw(20) <<
tipe() << setiosflags(ios::left) << setw(16) << "Rp. " << hTiket()
<< ",-" << setiosflags(ios::left) << setw(11) << "|" << endl;
    }
    cout <<

"=====
=====
"
<< endl;

        file <<

"=====
=====
"
<< endl;

        cout << setiosflags(ios::right) << setw(102) <<
"Total" << setiosflags(ios::left) << setw(6) << "Rp. " << totPem
<< ",-" << endl;
        file << setiosflags(ios::right) << setw(102) <<
"Total" << setiosflags(ios::left) << setw(6) << "Rp. " << totPem
<< ",-" << endl;
        cout << setiosflags(ios::right) << setw(102) <<
"Bayar" << setiosflags(ios::left) << setw(6) << "Rp. "; cin >>
byr;

        file << setiosflags(ios::right) << setw(102) <<
"Bayar" << setiosflags(ios::left) << setw(6) << "Rp. " << byr <<
",-" << endl;

        kembalian = byr - totPem;
        cout << setiosflags(ios::right) << setw(102) <<
"Kembalian" << setiosflags(ios::left) << setw(6) << "Rp. " <<
kembalian << ",-" << endl;
        file << setiosflags(ios::right) << setw(102) <<
"Kembalian" << setiosflags(ios::left) << setw(6) << "Rp. " <<
kembalian << ",-" << endl;
    }
    file.close();
    cout << endl << "    Rekap data sudah ditulis dan
dikirim ke pusat. Terima kasih telah memakai jasa kami" << endl;
}

}

void printTiket ()
{
    ofstream file ("Tiket.txt");
    cout << endl;
    if (!file.is_open()) {
        cout << "Tidak bisa membuka file";
    }
    else {
        for (a = 1; a <= jml; a++)
        {
            for (b = 1; b <= 1; b++)
            {

```

```

        file << "- - - - -";
        file << endl << endl;
        file << "    Kelas " << tipe() << endl << endl;
        file << "    Nama" << endl;
        file << "    " << cl.nama[a] << endl << endl;
        file << "    Penerbangan          Kelas" << endl;
        file << "    " << kode() <<
setiosflags(ios::right) << setw(19) << tipe() << endl << endl;
        file << "    Dari - Ke" << endl;
        file << "    " << Tujuan() << endl;
        file << "- - - - -";
        " << endl;

        if (jum > 1)
        {
            for (d = 1; d < jum; d++)
            {
                file << "- - - - -";
                file << "    Kelas " << tipe() << endl <<
endl;

                file << "    Nama" << endl;
                file << "    " << an.nama[a][d] << endl <<
endl;

                file << "    Penerbangan          Kelas" <<
endl;

                file << "    " << kode() <<
setiosflags(ios::right) << setw(19) << tipe() << endl << endl;
                file << "    Dari - Ke" <<
endl;

                file << "    " << Tujuan() << endl;
                file << "- - - - -";
                " << endl;
            }
        }
        cout << endl;
    }
}

main ()
{
    system("cls");
    tampilanAwal();
    tampilanPesawat();
    cout << endl;
    cout << "    Jumlah Client : "; cin >> jml; //untuk perulangan
jumlah client yang membeli tiket pesawat

    for (a = 1; a <= jml; a++) //perulangan jumlah client
    {
        orang();
        ulang1:
        cout << "    Pesawat : "; cin >> t.pswt[a]; //inputan
pilihan pesawat
    }
}

```

```

        for (b = 1; b <= 1; b++) //perulangan data pesawat client
        {
            switch (t.pswt[a]) //penyeleksian kondisi pemilihan
pesawat
            {
                case '1':
                {
                    cout << endl;
                    pilihanTujuan();
                    kembali1:
                    cout << "    Tujuan Pilihan : "; cin >>

t.pil[a][b];

                    switch (t.pil[a][b])
                    {
                        case '1':
                        {
                            cout << endl << "    Tujuan : " <<
Tujuan () << endl;
                            pilihanKelas();
                            dataTiket();
                            break;
                        }
                        case '2':
                        {
                            cout << endl << "    Tujuan : " <<
Tujuan () << endl;
                            pilihanKelas();
                            dataTiket();
                            break;
                        }
                        case '3':
                        {
                            cout << endl << "    Tujuan : " <<
Tujuan () << endl;
                            pilihanKelas();
                            dataTiket();
                            break;
                        }
                        case '4':
                        {
                            cout << endl << "    Tujuan : " <<
Tujuan () << endl;
                            pilihanKelas();
                            dataTiket();
                            break;
                        }
                        case '5':
                        {
                            cout << endl << "    Tujuan : " <<
Tujuan () << endl;
                            pilihanKelas();
                            dataTiket();
                            break;
                        }
                    }
                }
            }
        }
    }
}

```

```

    }
    default:
    {
        cout << "    Salah Inputan\n";
        cout << "    Tekan ENTER untuk
kembali\n";

        getch();
        goto kembali1;
        break;
    }
    }
    break;
}
case '2':
{
    cout << endl;
    pilihanTujuan();
    kembali2:
    cout << "    Tujuan Pilihan : "; cin >>

t.pil[a][b];

    switch (t.pil[a][b])
    {
        case '1':
        {
            cout << endl << "    Tujuan : " <<

Tujuan () << endl;

            pilihanKelas();
            dataTiket();
            break;
        }
        case '2':
        {
            cout << endl << "    Tujuan : " <<

Tujuan () << endl;

            pilihanKelas();
            dataTiket();
            break;
        }
        case '3':
        {
            cout << endl << "    Tujuan : " <<

Tujuan () << endl;

            pilihanKelas();
            dataTiket();
            break;
        }
        case '4':
        {
            cout << endl << "    Tujuan : " <<

Tujuan () << endl;

            pilihanKelas();
            dataTiket();
            break;
        }
        case '5':

```



```

        {
            cout << endl << "    Tujuan : " <<
Tujuan () << endl;
            pilihanKelas();
            dataTiket();
            break;
        }
        default:
        {
            cout << "    Salah Inputan\n";
            cout << "    Tekan ENTER untuk
kembali\n";

            getch();
            goto kembali2;
            break;
        }
    }
    break;
}
case '3':
{
    cout << endl;
    pilihanTujuan();
    kembali3:
    cout << "    Tujuan Pilihan : "; cin >>

t.pil[a][b];

    switch (t.pil[a][b])
    {
        case '1':
        {
            cout << endl << "    Tujuan : " <<
Tujuan () << endl;
            pilihanKelas();
            dataTiket();
            break;
        }
        case '2':
        {
            cout << endl << "    Tujuan : " <<
Tujuan () << endl;
            pilihanKelas();
            dataTiket();
            break;
        }
        case '3':
        {
            cout << endl << "    Tujuan : " <<
Tujuan () << endl;
            pilihanKelas();
            dataTiket();
            break;
        }
        case '4':
        {

```

```

        cout << endl << "    Tujuan : " <<
Tujuan () << endl;
        pilihanKelas();
        dataTiket();
        break;
    }
    case '5':
    {
        cout << endl << "    Tujuan : " <<
Tujuan () << endl;
        pilihanKelas();
        dataTiket();
        break;
    }
    default:
    {
        cout << "    Salah Inputan\n";
        cout << "    Tekan ENTER untuk
kembali\n";

        getch();
        goto kembali3;
        break;
    }
    }
    break;
}
case '4':
{
    cout << endl;
    pilihanTujuan();
    kembali4:
    cout << "    Tujuan Pilihan : "; cin >>
t.pil[a][b];

    switch (t.pil[a][b])
    {
        case '1':
        {
            cout << endl << "    Tujuan : " <<
Tujuan () << endl;
            pilihanKelas();
            dataTiket();
            break;
        }
        case '2':
        {
            cout << endl << "    Tujuan : " <<
Tujuan () << endl;
            pilihanKelas();
            dataTiket();
            break;
        }
        case '3':
        {
            cout << endl << "    Tujuan : " <<
Tujuan () << endl;

```

```

        pilihanKelas();
        dataTiket();
        break;
    }
    case '4':
    {
        cout << endl << "    Tujuan : " <<
Tujuan () << endl;

        pilihanKelas();
        dataTiket();
        break;
    }
    case '5':
    {
        cout << endl << "    Tujuan : " <<
Tujuan () << endl;

        pilihanKelas();
        dataTiket();
        break;
    }
    default:
    {
        cout << "    Salah Inputan\n";
        cout << "    Tekan ENTER untuk
kembali\n";

        getch();
        goto kembali4;
        break;
    }
    }
    break;
}
default: //jika penginputan salah, maka akan langsung
mengarah ke default untuk menyatakan kondisi salah
{
    cout << "Salah memasukan pilihan\n";
    goto ulang1;
    break;
}
}
cout << endl;

}
tampilanAkhir(); //menampilkan hasil dari proses dalam main
cout << endl;
getch();
system("cls"); //menghapus tampilan program
printTabel(); //menampilkan tabel rekap data pelanggan yang
membeli tiket
printTiket (); //menampilkan boarding pass
cout << endl;

return 0; //mengembalikan seluruhnya
}

```

### 3.3 Tampilan Program

Berikut hasil dari tampilan awal program pemesanan tiket pesawat :

```
||=====|| |
||          PILIHAN MENU MASUK          ||
||=====||
||      No      |      Menu Masuk      ||
||=====||
||      1      |      Masuk           ||
||      2      |      Registrasi      ||
||      4      |      Keluar          ||
||=====||
|| Masukkan pilihan anda : 1 ||
```

//User diminta untuk memilih terlebih dahulu pilihan menu

Berikut hasil dari tampilan pilihan kode kelas tiket pesawat :

```
Silahkan Registrasi Akun Anda!

Nama Lengkap: Putri Ayu
Email: ptayu00@gmail.com
Hallo Putri Ayu!

Silahkan membuat username (max 12 karakter), serta password (max 10 karakter)
Masukkan Username : ayuputri
Masukkan Password : putriayu

Registrasi Selesai!
Akun Anda Terdaftar, Silahkan Mencoba!           ||=====||
|| LOGIN ||
||=====||
|| Username: ayuputri ||
||=====||
|| Password: putriayu ||
```

```
=====
WELCOME TO MY INTERNATIONAL TICKET
=====

*-----*
| Nama Pesawat | Jam Berangkat | Harga Tiket |
|-----|-----|-----|
| 1. Garuda   | 08.30        | Rp. 1.600.000,- |
|-----|-----|-----|
| 2. Airbus   | 10.00        | Rp. 1.700.000,- |
|-----|-----|-----|
| 3. Lion Air | 12.30        | Rp. 1.400.000,- |
|-----|-----|-----|
| 4. Air Asia | 14.00        | Rp. 1.500.000,- |
|-----|-----|-----|
*-----*

Jumlah Client : 1

Client 1
Nama Lengkap : Putri Ayu
Alamat dari nama client : 0x49b144
Jumlah Penumpang : 1
```

```
Jumlah Client : 1

Client 1
Nama Lengkap      : Putri Ayu
Alamat dari nama client : 0x49b144
Jumlah Penumpang : 1
Pesawat : 3

Tujuan Keberangkatan:
1. BANJARMASIN-JAKARTA
2. BANJARMASIN-MAKAASAR
3. BANJARMASIN-BALI
4. BANJARMASIN-SURABAYA
5. BANJARMASIN-PONTIANAK
Tujuan Pilihan : 4

Tujuan : BANJARMASIN-SURABAYA

Kelas Pesawat
1. Ekonomi
2. Bisnis
Pilihan : 1
Tiket Regular, tidak ada tambahan biaya

Client Putri Ayu Membeli tiket BANJARMASIN-SURABAYA dengan pesawat SA-BS kelas Ekonomi
seharga Rp. 1400000,-
```

=====					
No.	Nama Client	Tujuan	Pesawat	Kelas	Harga
1	Putri Ayu	BANJARMASIN-SURABAYA	SA-BS	Ekonomi	Rp. 1400000,-
=====					
				Total	Rp. 1400000,-
				Bayar	Rp. 2000000

//Setelah login atau registrasi, user dapat memasukan jumlah pembelian, dan mengisi data yang diminta. Setelah itu, user bisa memilih pesawat, tujuan kota, dan kelas pesawat. Jika sudah selesai, user dapat melihat hasil transaksi dari pemesanan tiket pesawat

## **BAB IV**

### **PENUTUP**

#### **4.1 Kesimpulan**

Dari laporan tugas besar ini, maka dapat diambil kesimpulan yaitu dengan adanya program pemesanan tiket pesawat, dapat mempermudah konsumen dan konsumen dapat menghemat waktu dan biaya, dan dapat melakukan pemesanan tiket tanpa harus langsung ke tempat pembelian tiket.