

HOLOTIX – TICKETING

LAPORAN STUDI KASUS

MATA KULIAH COMP6360 – ALGORITHM AND PROGRAMMING

KELAS BC20



Oleh :

2440060502 – Acmad Aqil Susanto

2440018854 – Gerry William Nanlohy

2440066405 – Muhammad Zulham Firdaus

Semester Ganjil – 2020/2021

MALANG

KATA PENGANTAR

Puji syukur kehadiran Allah Yang Maha Esa, yang telah memberikan rahmat dan karunia-Nya sehingga kami dapat menyelesaikan laporan studi kasus ini yang berjudul HOLOTIX – TICKETING. Laporan ini disusun untuk memenuhi tugas akhir semester 1 mata kuliah COMP6360 - Algorithm and Programming. Laporan ini pula berisi mengenai hasil dari semua proses pengerjaan Case Study Lab Algo 2020 pada Case Study C – Ticketing. Terima kasih kami ucapkan kepada dosen pembimbing dalam mata kuliah COMP6360 - Algorithm and Programming, pihak penyusun dan teman - teman seperjuangan yang telah mendukung kami, sehingga kami bisa mengerjakan dan menyelesaikan dengan lancar tugas ini secara tepat waktu.

Kami menyadari, bahwa laporan yang kami buat ini masih jauh dari kata sempurna baik segi penyajian, penyusunan bahasa, maupun penulisannya. Oleh karena itu, kami sangat mengharapkan kritik dan saran yang membangun dari semua pembaca guna menjadi acuan agar tim penulis bisa menjadi lebih baik lagi di masa mendatang.

Semoga laporan ini bisa menambah wawasan para pembaca dan bisa bermanfaat untuk perkembangan dan peningkatan ilmu pengetahuan Algorithm and Programming.

Malang, 05 Januari 2021

Tim Penulis

DAFTAR ISI

Kata Pengantar	i
Daftar Isi	ii
BAB I. GAMBARAN UMUM	1
A. Latar Belakang	1
B. Tampilan Program	2
C. Fitur Program	6
D. Proses dalam Program	9
BAB II. HASIL.....	10
A. Screenshot Program	10
B. Source Code	15
BAB III. PEMBAGIAN TUGAS	
A. Acmad Aqil Susanto	27
B. Gerry William Nanlohy	27
C. Muhammad Zulham Firdaus	28

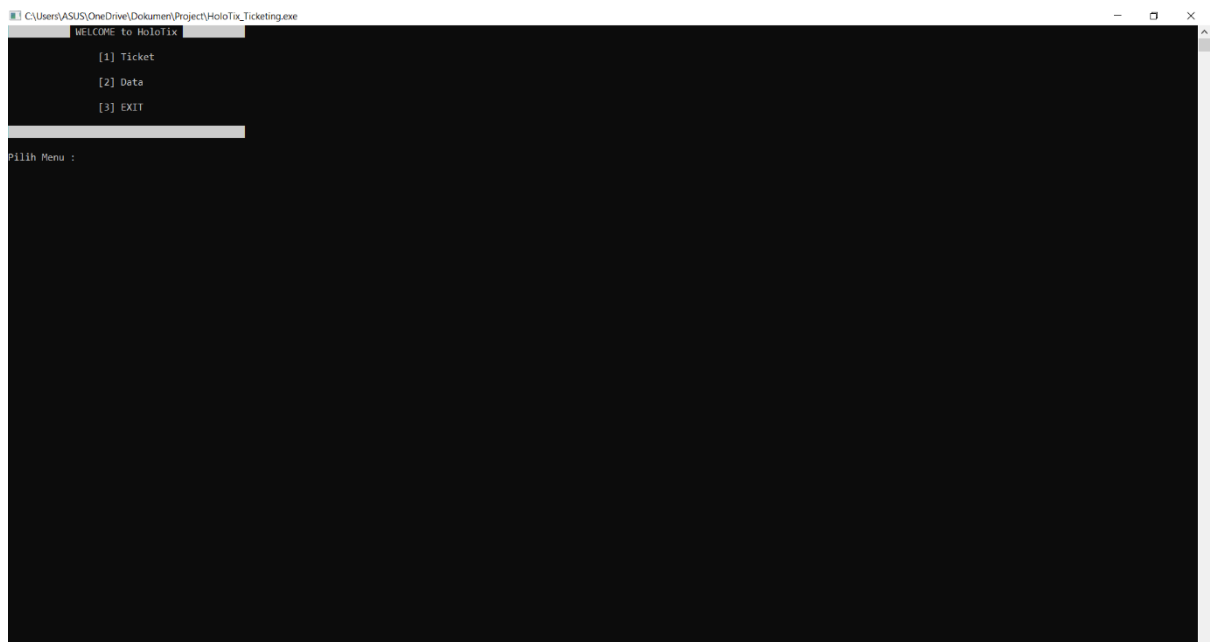
ⁱⁱ BAB I

GAMBARAN UMUM

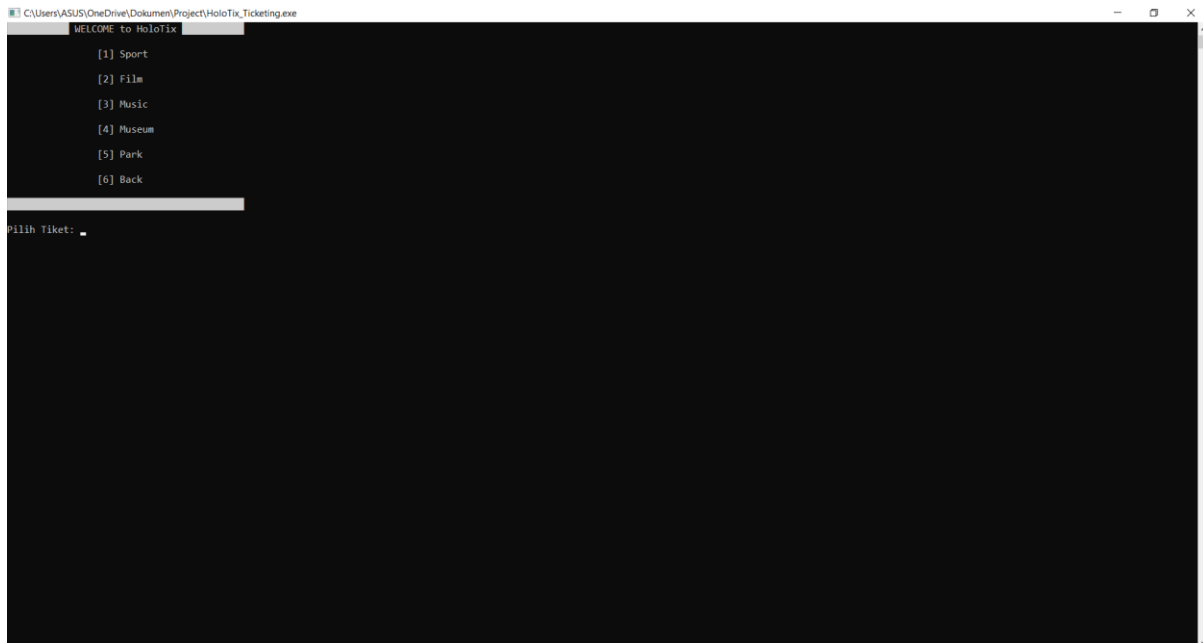
A. Latar Belakang

Ticketing merupakan salah satu dari 20 studi kasus dalam mata kuliah COMP6360 Algorithm & Programming Computer Science @BINUS MALANG, dimana studi kasus ditandai dengan abjad C. Dalam studi kasus ini program yang dibuat berfungsi untuk membantu suatu perusahaan yang berjalan di bidang ticketing dan menginginkan sebuah program yang dapat melakukan pendataan data per hari, di mana data diinput oleh pengguna. Data yang telah diinput oleh pengguna akan tersimpan dan dapat dilihat dalam bentuk file dengan ekstensi .txt yang mencakup tanggal transaksi, nama pelanggan, jenis event, jumlah tiket, harga tiket, dan total pembayaran. Adapun program yang diminta adalah yang pertama dapat melakukan pendataan ticket yang terjual, kedua melakukan pencetakan laporan berdasarkan pengurutan tanggal secara *ascending* (pengurutan dari yang terkecil ke yang terbesar), ketiga melakukan pencetakan laporan pengurutan tiket berdasarkan penjualan tiket paling banyak secara *descending* (pengurutan dari yang terbesar ke yang terkecil).

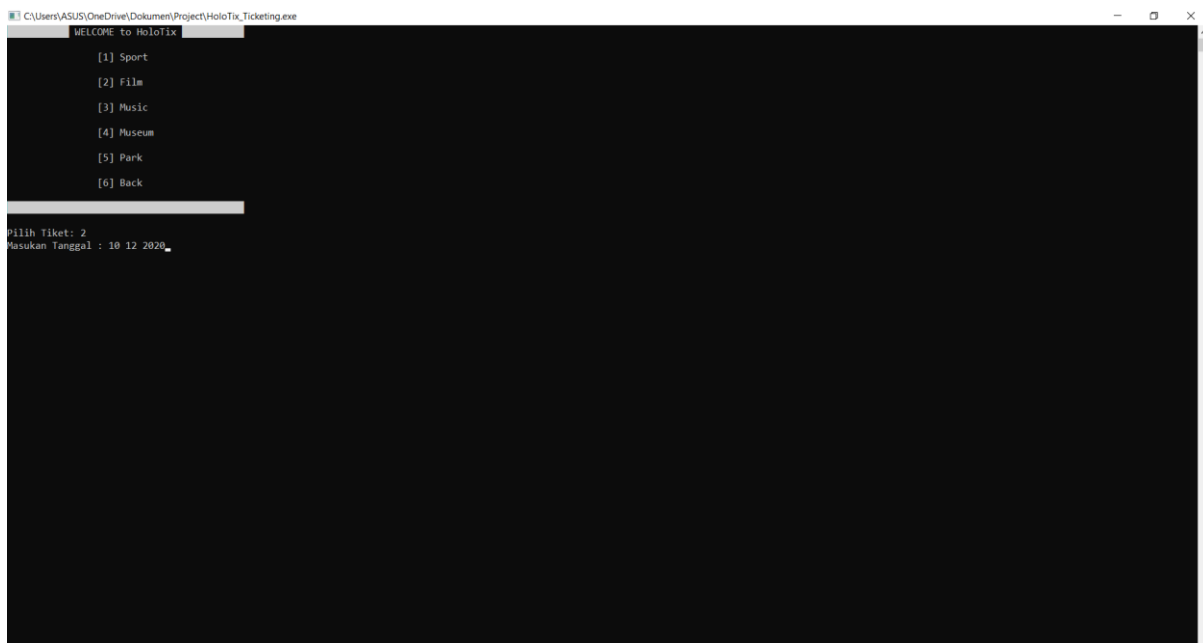
B. Tampilan Program



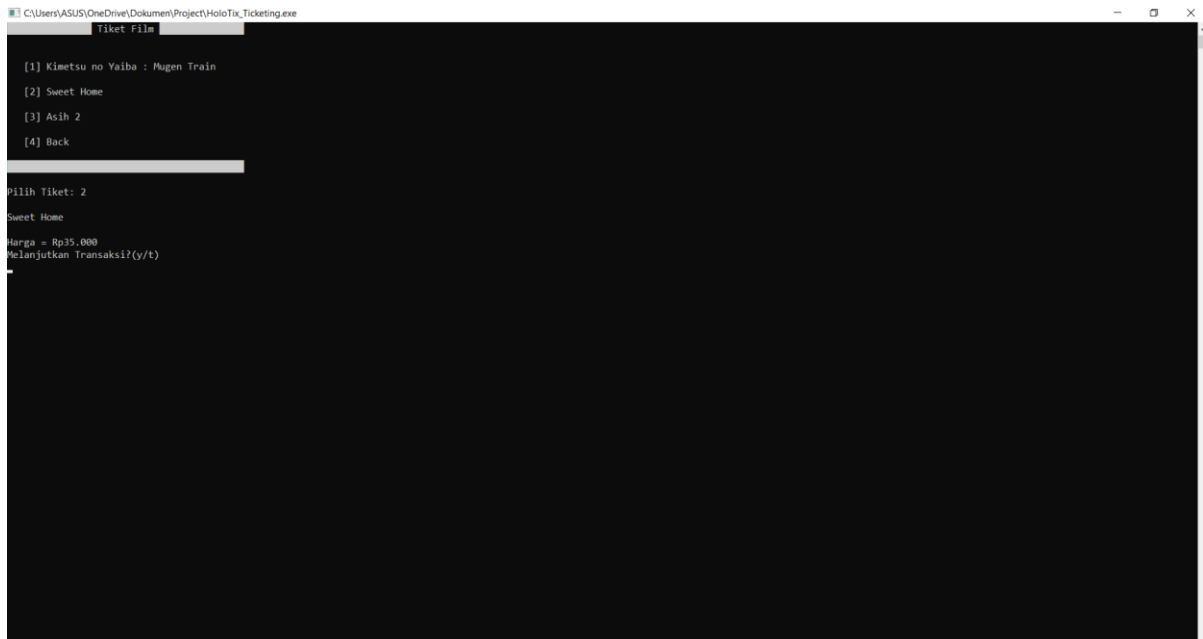
Tampilan awal aplikasi ini terdiri dari judul aplikasi (WELCOME to HoloTix) yang selalu muncul di seluruh tampilan menu lainnya, daftar angka sebanyak 3 beserta keterangan pada masing – masing angka : ([1] Ticket ; [2] Data ; [3] EXIT) dan baris (Pilih Menu :) sebagai tempat untuk pengguna dapat menginput angka yang diinginkan sesuai daftar menu yang telah disediakan. Dimana pada Menu [1] berfungsi untuk melakukan pendataan dan transaksi penjualan tiket, lalu pada Menu [2] berfungsi untuk menampilkan data secara terurut berdasarkan tanggal penjualan tiket dan penjualan tiket terbanyak, yang terakhir Menu [3] berfungsi untuk keluar dari aplikasi tersebut jika dirasa pengguna ingin membatalkan /sudah melakukan pendataan dan transaksi pembelian tiket.



Lalu apabila pengguna menginput angka (1) pada baris (Pilih Menu :) maka akan muncul daftar angka sebanyak 6 beserta keterangan pada masing – masing angka berupa jenis tiket yang berpeluang ingin dipesan oleh pengguna : ([1] Sport ; [2] Film ; [3] Music ; [4] Museum ; [5] Park ; [6] Back) dan baris (Pilih Tiket :) sebagai tempat untuk pengguna dapat menginput angka yang diinginkan sesuai daftar jenis tiket yang telah disediakan.



Lalu apabila pengguna aplikasi telah menginput angka sesuai jenis tiket yang ingin dipesan maka akan muncul baris (Masukkan Tanggal :) sebagai tempat pengguna melakukan pendataan untuk tanggal berapa tiket tersebut dipesan dengan format (dd mm yy) (seperti pada contoh gambar di atas).



Selanjutnya pengguna disediakan pilihan angka kembali yang berisikan jenis event sesuai jenis tiket yang dipilih (seperti pada contoh gambar di atas), terdapat juga baris (Pilih Tiket :) sebagai tempat untuk pengguna dapat menginput angka yang diinginkan sesuai daftar jenis event dari jenis tiket yang telah dipilih sebelumnya. Tidak hanya itu, terdapat baris pemberitahuan jenis event yang telah dipilih sesuai inputan angka pada baris (Pilih Tiket :), baris harga jenis event tersebut dan juga baris pertanyaan (Melanjutkan Transaksi?(y/t)) dimana apabila pengguna menginput abjad y (Ya) maka program akan melanjutkan transaksi, namun jika pengguna menginput abjad t (Tidak) maka program akan berhenti melakukan transaksi dan kembali ke menu utama.

```
C:\Users\ASUS\OneDrive\Project\HoloTix\Ticketing.exe
Tiket Film

[1] Kimetsu no Yaiba : Mugen Train
[2] Sweet Home
[3] Asih 2
[4] Back

Pilih Tiket: 2
Sweet Home
Harga = Rp35.000
Melanjutkan Transaksi?(y/t)
y
Banyak Tiket : 5
Nama : gwen
-----
Process exited after 1.658e+004 seconds with return value 0
Press any key to continue . . .
```

Terakhir, apabila pengguna menginput abjad y (Ya) maka akan muncul baris (Banyak Tiket :) sebagai tempat untuk pengguna dapat menginput jumlah tiket yang akan di pesan dan baris (Nama :) sebagai tempat untuk pengguna dapat menginput atas nama siapakah tiket tersebut dipesan.

C. Fitur Program

1. Fitur Pendataan Penjualan Tiket

a. Tanggal Transaksi

Tanggal transaksi berfungsi untuk menentukan kapan tanggal terjadinya transaksi yang akan dicetak pada Menu : Data Collection dengan format (dd mm yy) setelah transaksi.

b. Nama Pelanggan

Nama pelanggan berfungsi untuk menentukan atas nama siapakah tiket tersebut dipesan yang dapat dilihat pada Menu : Data Collection setelah transaksi.

c. Jenis Tiket

Jenis tiket berfungsi untuk menentukan jenis tiket apa yang berpeluang akan dipesan oleh pengguna. Ini dapat dilihat pada Menu : Data Collection setelah transaksi.

d. Nama Event

Nama event berfungsi untuk menentukan jenis event apa yang berpeluang akan dipesan oleh pengguna sesuai dengan jenis tiket apa yang telah dipilih sebelumnya oleh pengguna. Ini dapat dilihat pada Menu : Data Collection setelah transaksi.

e. Jumlah Tiket

Jumlah tiket berfungsi untuk menentukan berapa tiket yang hendak dipesan oleh pengguna. Ini dapat dilihat pada Menu : Data Collection setelah transaksi.

f. Harga Tiket

Harga tiket berfungsi untuk menentukan berapa harga tiket per tiketnya sesuai yang telah dipesan oleh pengguna. Ini dapat dilihat pada Menu : Data Collection setelah transaksi.

g. Total Pembayaran

Total pembayaran berfungsi untuk menentukan jumlah total harga tiket (harga tiket per tiket x jumlah tiket) yang telah dipesan oleh pengguna. Ini dapat dilihat pada Menu : Data Collection setelah transaksi.

h. Stok Tiket.

Stok Tiket berfungsi untuk menentukan dan mendata jumlah keseluruhan dari seluruh jenis event sesuai jenis tiket beserta sisa stok tiketnya (pengurangan stok = stok - banyak tike). Ini dapat dilihat pada Menu : Data Collection setelah transaksi.

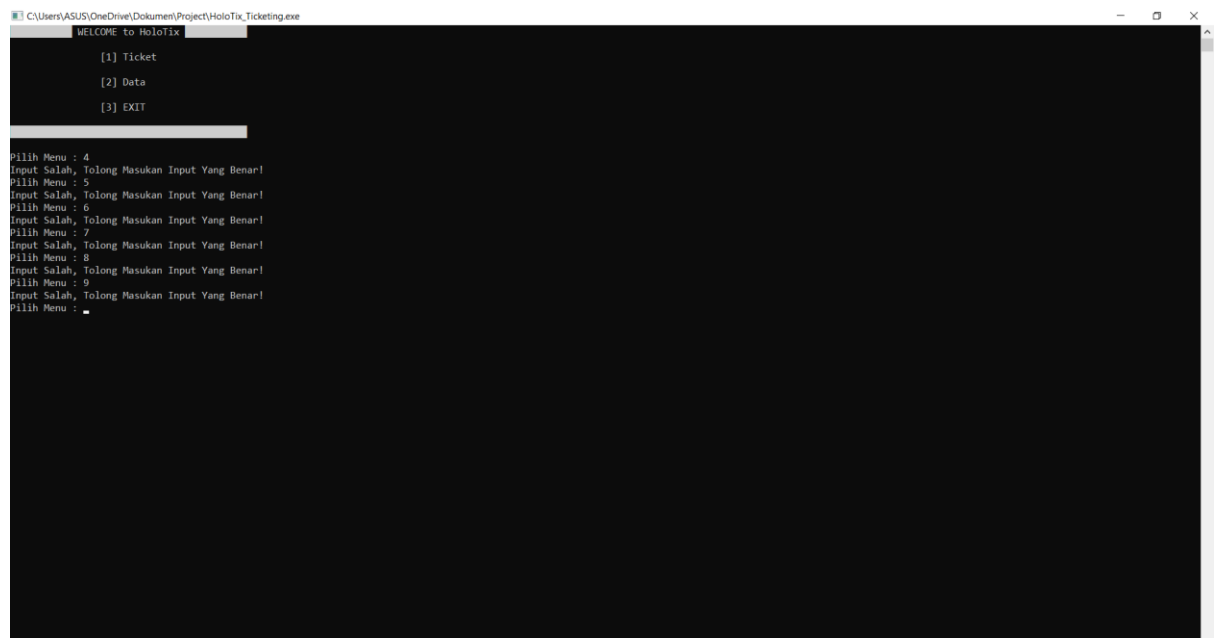
2. Fitur Pencetakan Laporan Berdasarkan Tanggal Transaksi

Fitur ini menjalankan aplikasi untuk membaca file berekstensi .txt, kemudian dengan fungsi tertentu aplikasi menerjemahkan serta menampilkan kepada pengguna ke dalam bentuk penyajian list terurut sesuai dengan tanggal transaksi yang paling awal dilakukan.

3. Fitur Pencetakan Laporan Berdasarkan Penjualan Tiket Paling Banyak

Fitur ini menjalankan aplikasi untuk membaca file berekstensi .txt, kemudian dengan fungsi tertentu aplikasi menerjemahkan serta menampilkan kepada pengguna ke dalam bentuk penyajian list terurut sesuai dengan tiket yang terjual paling banyak.

4. Fitur Validasi Inputan



Fitur ini berfungsi apabila data yang diinput oleh pengguna tidak sesuai dengan daftar data yang telah tersedia (seperti pada contoh gambar di atas).

5. Fitur Exit Menu



Fitur ini berfungsi supaya aplikasi dapat kembali atau menutup ke menu ucapan (TERIMA KASIH TELAH MENGGUNAKAN HoloTix) jika pengguna dirasa pengguna ingin membatalkan /sudah melakukan pendataan dan transaksi pembelian tiket.

D. Proses dalam Program

Proses yang terdapat pada program ini menggunakan metode penghitungan matematika. Pada menu utama, pengguna diminta untuk menentukan Menu yang akan mereka pilih dengan menginputkan sebuah angka yang telah tersedia di daftar angka beserta keterangannya. Apabila pengguna melakukan kesalahan input angka di luar daftar angka beserta keterangannya, maka program akan melakukan validasi inputan ulang.

Ketika pengguna memilih Menu [1] Tiket dengan menginput angka (1) pada baris (Pilih Menu :), maka pengguna akan diarahkan pada daftar jenis tiket. Terdapat 5 jenis tiket yang tersedia, yaitu [1] Sport, [2] Film, [3] Music, [4] Museum, [5] Park serta terdapat pilihan ke [6] jika pengguna ingin kembali ke Menu Utama. Lalu pengguna diminta untuk menginput sebuah angka yang telah tersedia di daftar jenis tiket, sesuai jenis tiket yang hendak dipesan. Selanjutnya ketika pengguna telah menginput sebuah angka, pengguna diminta untuk menginput tanggal untuk pendataan tanggal transaksi

Kemudian, pengguna diarahkan pada daftar jenis event sesuai jenis tiket yang telah dipilih sebelumnya. Setelah pengguna melakukan input angka yang telah tersedia di daftar jenis event, akan muncul harga tiket beserta baris pertanyaan yang ditujukan bagi pengguna apakah pengguna ini melanjutkan transaksi? Pengguna cukup menjawab hanya dengan menginput abjad y (Ya) atau t (Tidak). Apabila pengguna menginput abjad y (Ya) maka program akan melanjutkan transaksi dan program akan meminta pengguna untuk menginput jumlah tiket yang hendak dipesan beserta nama pengguna/nama orang yang diatasnamakan pada pemesanan tiket tersebut, namun jika pengguna menginput abjad t (Tidak) maka program akan berhenti melakukan transaksi dan kembali ke menu utama.

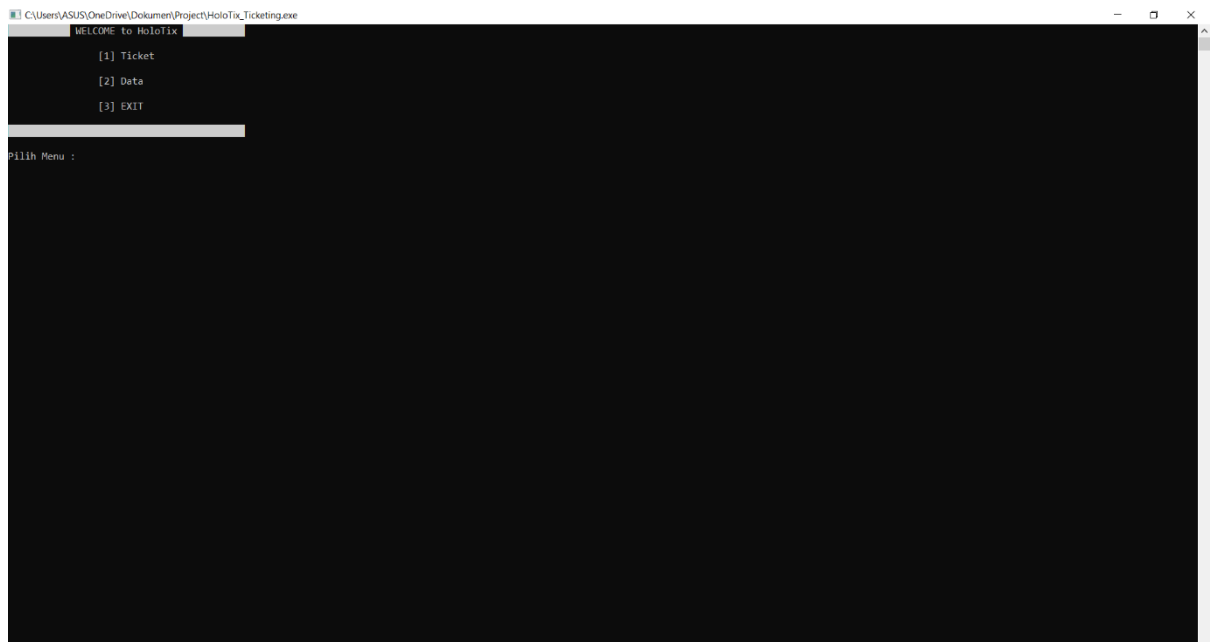
Pada Menu Utama apabila pengguna memilih Menu [2] Data dengan menginput angka (2) pada baris (Pilih Menu :), maka pengguna akan diarahkan pada Menu : Data Sales. Terdapat 3 pilihan bagi pengguna untuk dapat melihat Menu : Data Collection, jika pengguna memilih [1] maka pengguna akan diarahkan pada pendataan data transaksi, bila pengguna memilih [2] maka pengguna akan diarahkan pada laporan berdasarkan tanggal, dan bila pengguna memilih [3] maka pengguna akan diarahkan pada laporan berdasarkan produk, sedangkan pada pilihan [4] pengguna akan diarahkan kembali pada Menu Utama. Proses sorting yang digunakan pada pengurutan data transaksi berdasarkan tanggal dan jumlah tiket paling banyak yang terjual menggunakan BubbleSort.

BAB II

HASIL

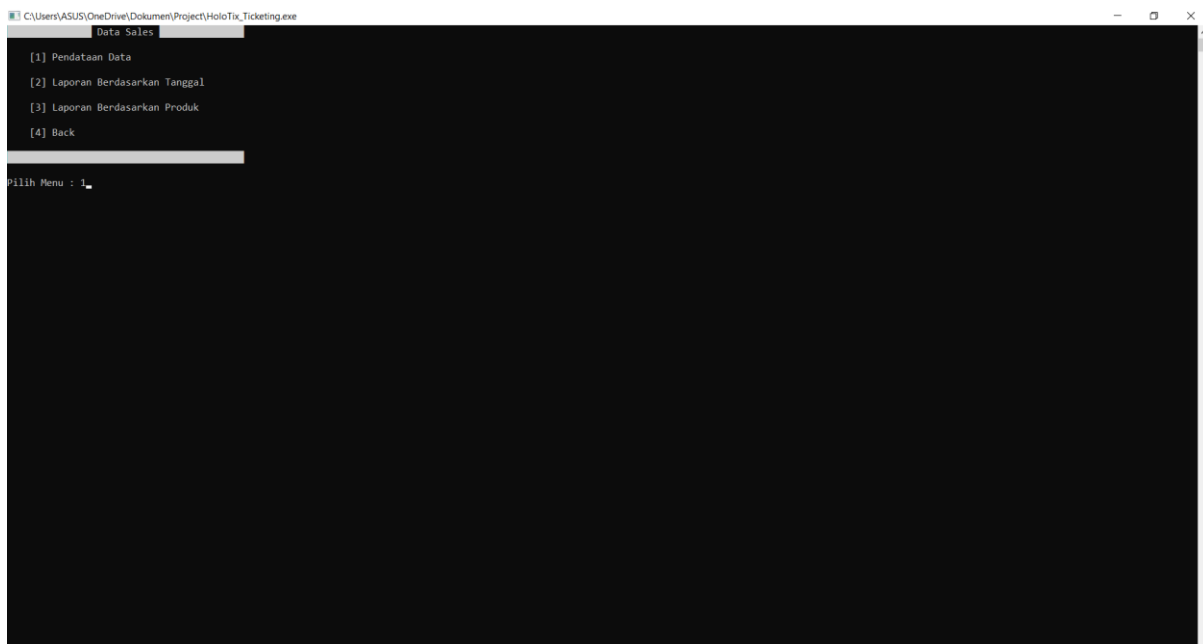
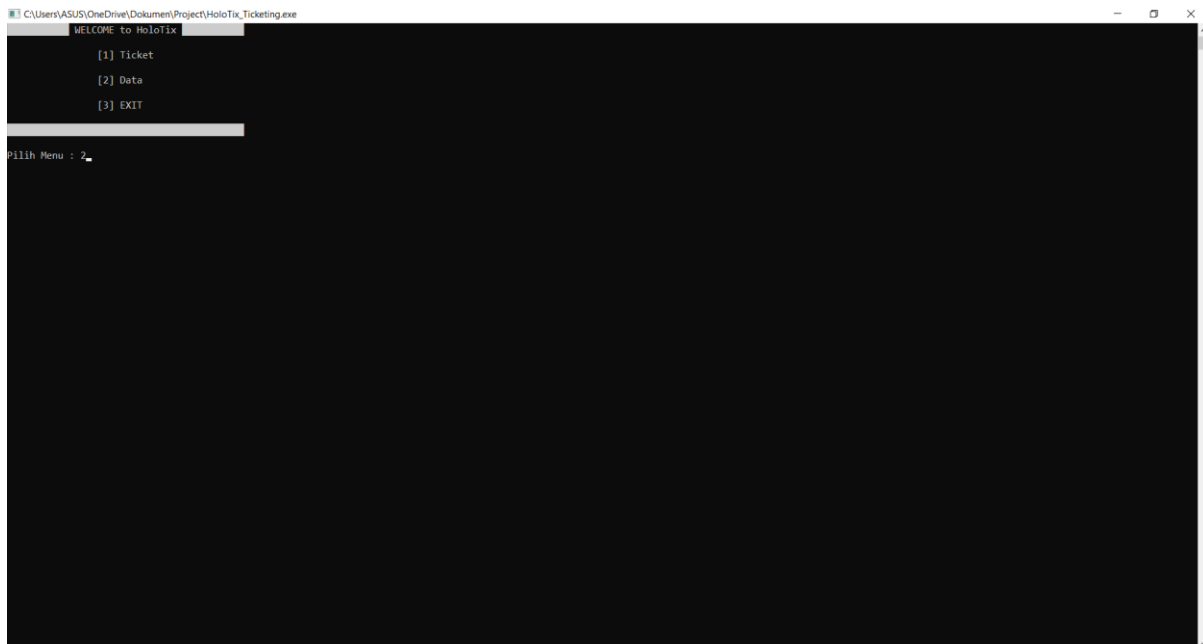
1. Screenshot Menu Program

a. Tampilan Menu Utama



Tampilan awal aplikasi ini terdiri dari judul aplikasi (WELCOME to HoloTix) yang selalu muncul di seluruh tampilan menu lainnya, daftar angka sebanyak 3 beserta keterangan pada masing – masing angka : ([1] Ticket ; [2] Data ; [3] EXIT) dan baris (Pilih Menu :) sebagai tempat untuk pengguna dapat menginput angka yang diinginkan sesuai daftar menu yang telah disediakan. Dimana pada Menu [1] berfungsi untuk melakukan pendataan dan transaksi penjualan tiket, lalu pada Menu [2] berfungsi untuk menampilkan data secara terurut berdasarkan tanggal penjualan tiket dan penjualan tiket terbanyak, yang terakhir Menu [3] berfungsi untuk keluar dari aplikasi tersebut jika dirasa pengguna ingin membatalkan /sudah melakukan pendataan dan transaksi pembelian tiket.

b. Pendataan Penjualan Tiket



Menu pendataan penjualan tiket dapat dibuka, dimana pengguna harus menginput angka (2) pada baris (Pilih Menu :) pada Menu Utama sehingga aplikasi dapat menjalankan fungsinya untuk menuju ke Menu : Data Sales seperti gambar pertama. Lalu pengguna dapat menginput angka (1) seperti pada gambar kedua pada baris (Pilih Menu :) sehingga aplikasi dapat menjalankan fungsinya untuk menuju ke Menu : Data Collection.

C:\Users\ASUS\OneDrive\Documents\Project\HoloTix_Ticketing.exe

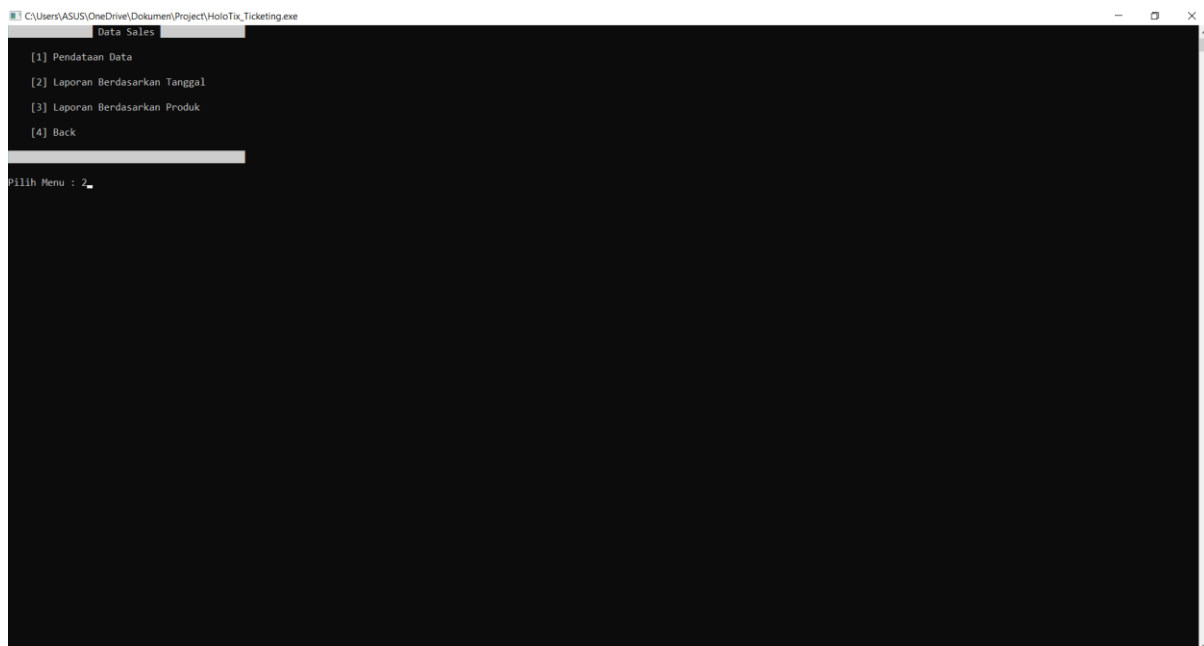
Data Collection

Tanggal	Nama	Jenis	Tiket	Banyak	Sisa Stock	Harga	Total
05 12 2020	agil	Music	AKB48	10	90	Rp. 500000	Rp. 5000000
20 12 2020	zul	Film	Sweet Home	15	85	Rp. 50000	Rp. 750000
10 12 2020	gwen	Film	Sweet Home	5	95	Rp. 35000	Rp. 175000

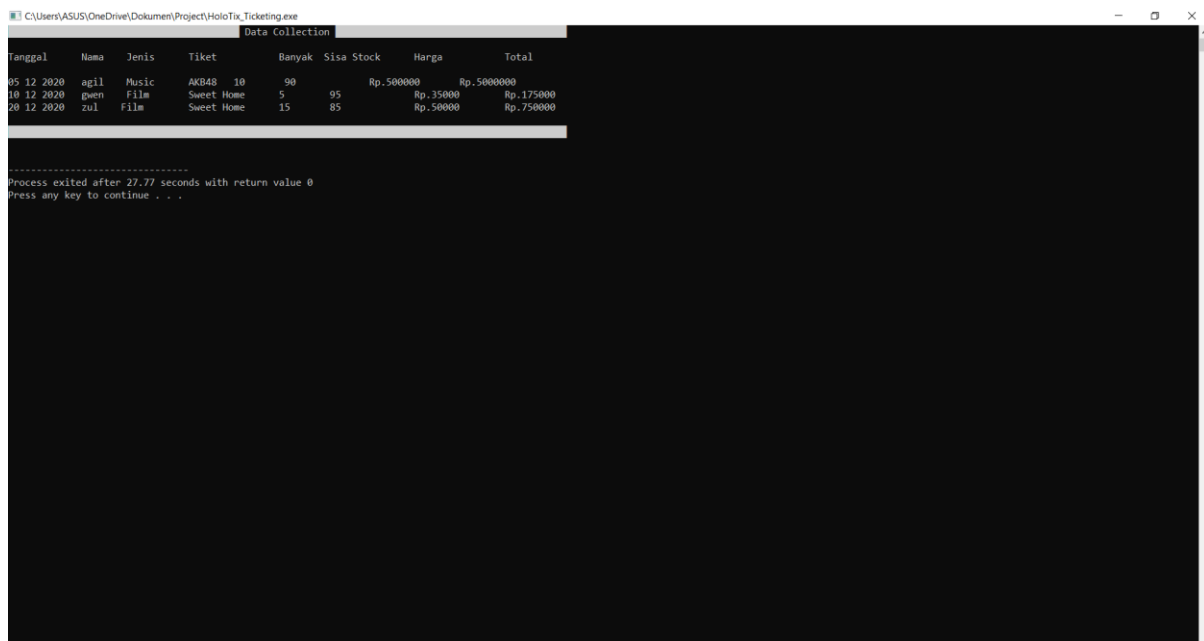
Process exited after 4.197 seconds with return value 0
Press any key to continue . . .

Ketika Menu : Data Collection telah muncul, pengguna dapat melihat data transaksi pembelian tiket yang telah dipesan sebelumnya seperti pada gambar ketiga. Tampilan Menu : Data Collection terdiri dari tanggal transaksi, nama orang yang melakukan transaksi atau orang yang mengatassnamakan tiket yang dipesan, jenis tiket, jenis event, jumlah keseluruhan tiket, sisa tiker yang tersedia, harga tiket dan total harga tiket.

c. Pengurutan Berdasarkan Tanggal Penjualan Tiket

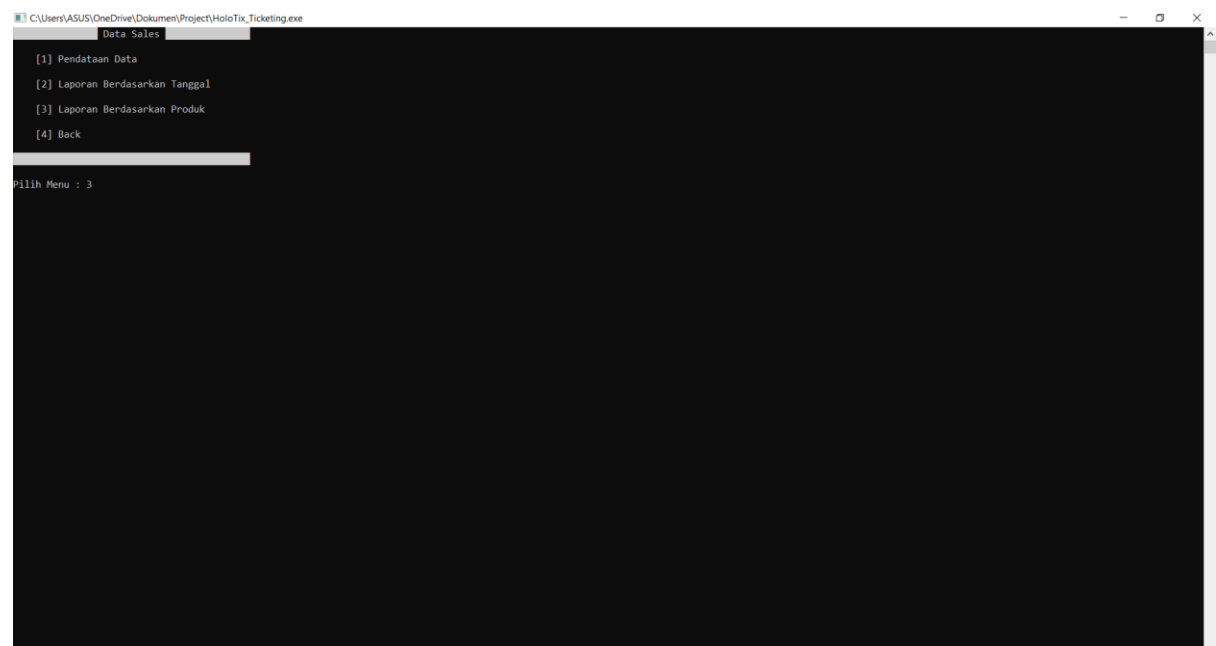


Menu pendataan tiket berdasarkan pengurutan tanggal penjualan tiket dapat dibuka, dimana pengguna harus menginput angka (2) pada baris (Pilih Menu :) pada Menu : Data Sales sehingga aplikasi dapat menjalankan fungsinya untuk menuju ke Menu : Data Collection.

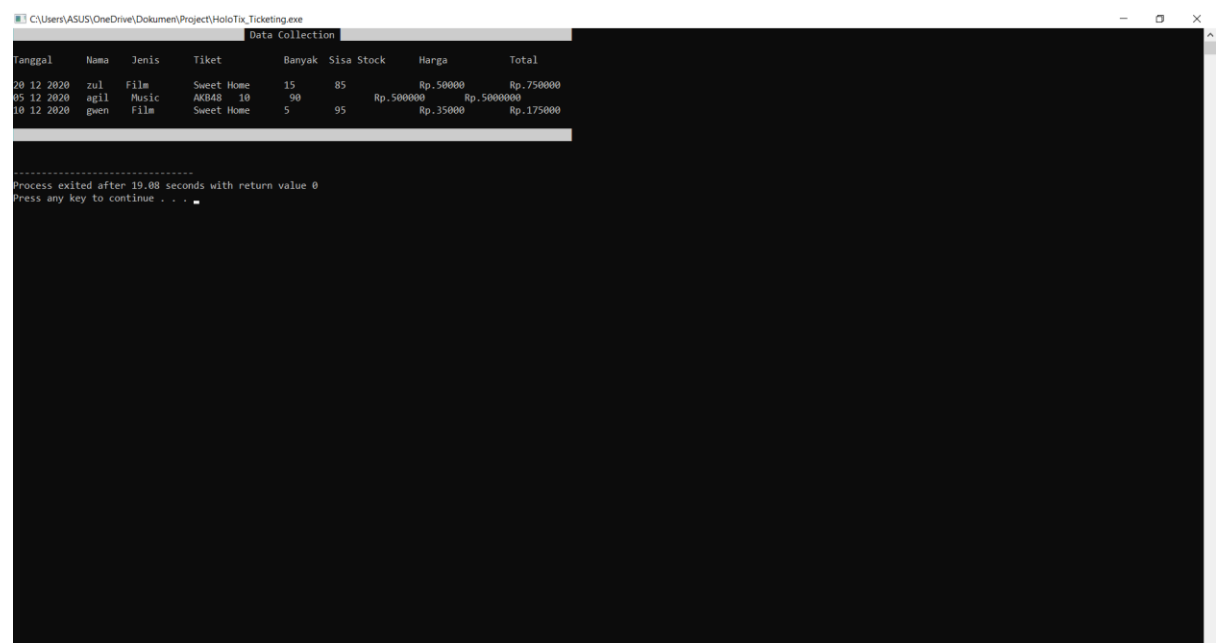


Ketika Menu : Data Collection telah muncul, maka pengguna dapat melihat data transaksi pembelian tiket yang telah dipesan sebelumnya berdasarkan pengurutan tanggal penjualan tiket (dimulai dari transaksi yang paling awal dilakukan, hingga transaksi yang paling akhir).

d. Pengurutan Berdasarkan Penjualan Tiket Paling Banyak



Menu pendataan tiket berdasarkan pengurutan tanggal penjualan tiket dapat dibuka, dimana pengguna harus menginput angka (3) pada baris (Pilih Menu :) pada Menu : Data Sales sehingga aplikasi dapat menjalankan fungsinya untuk menuju ke Menu : Data Collection.



Ketika Menu : Data Collection telah muncul, maka pengguna dapat melihat data transaksi pembelian tiket yang telah dipesan sebelumnya berdasarkan pengurutan tiket yang paling banyak terjual.

2. Source Code

```
1. #include<stdio.h>
2. #include<string.h>
3. #include<stdlib.h>
4. #include<windows.h>
5. struct data{
6.     int hari;
7.     int bulan;
8.     int tahun;
9.     int banyak;
10.    int harga;
11.    int total;
12.    char nama[50];
13.    char jenis[50];
14.    char event[50];
15.    int stok;
16.    int jumlah;
17.    int sisa;
18. };
19. void swap( struct data *p1,struct data *p2){
20.     struct data temp=*p1;
21.     *p1=*p2;
22.     *p2=temp;
23. }
24. void datesort(struct data tgl[],int t){
25.     struct data swap;
26.     int i,j;
27.     for (i = 0; i < t - 1; i++) {
28.         for (j = 0; j < t - 1 - i; j++) {
29.             if (tgl[j].tahun > tgl[j + 1].tahun) {
30.                 swap = tgl[j];
31.                 tgl[j] = tgl[j + 1];
32.                 tgl[j + 1] = swap;
33.             }
34.             else if(tgl[j].tahun==tgl[j+1].tahun){
35.                 if (tgl[j].bulan > tgl[j + 1].bulan) {
36.                     swap = tgl[j];
37.                     tgl[j] = tgl[j + 1];
38.                     tgl[j + 1] = swap;
39.                 }else if(tgl[j].bulan == tgl[j + 1].bulan){
40.                     if (tgl[j].hari > tgl[j + 1].hari) {
41.                         swap = tgl[j];
42.                         tgl[j] = tgl[j + 1];
43.                         tgl[j + 1] = swap;
44.                     }
45.                 }
46.             }
47.         }
48.     }
49. }
50. void JumlahSort(struct data list[], int n){
51.     int i,j;
52.     struct data temp;
53.     for(i = 0;i < n-1;i++)
54.     {
55.         for(j = 0;j < n-1-i;j++)
56.         {
57.             if(list[j].banyak < list[j+1].banyak){
58.                 temp = list[j];
```

[illegible]

[illegible]

```

157.         }
158.         else{
159.             printf("Input Salah, Tolong Masukkan Inp
ut Yang Benar!\n");
160.             goto salahbs;
161.         }
162.         if(s == 0){
163.             printf("\nHarga = Rp50.000\n");
164.             harga = 50000;
165.         }
166.         else{
167.             printf("\nHarga = Rp35.000\n");
168.             harga = 35000;
169.         }
170.     }
171.     else if(strcmp(select,"3")==0){
172.         system("cls");
173.
174.         printf("\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb
\b\xdb\xdb\xdb\xdb\xdb\xdb \xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb
\b\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\n\n\n");
175.         printf("    [1] Tiket Konser AKB48\n\n");
176.         printf("    [2] Tiket Holofest 2nd Stage\n\n
");
177.         printf("    [3] Tiket Konser Black Pink\n\n
");
178.         printf("    [4] Back\n\n");
179.         printf("\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb
\b\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb
\b\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\n\n\n");
180.         printf("Pilih Tiket: ");
181.         scanf("%d",&plhkn);
182.         printf("\n");
183.         if(plhkn == 1){
184.             printf("\nHarga = Rp500.000\n");
185.             harga = 500000;
186.         }
187.         else if(plhkn == 2){
188.             system("cls");
189.             printf("\n");
190.             menuholo :
191.             printf("\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb
\b\xdb\xdb\xdb\xdb\xdb\xdb \xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb
\b\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\n\n\n");
192.             printf("   \t[1] Stage 1 (day 1)\n\n");
193.             printf("   \t[2] Stage 2 (day 2)\n\n");
194.             printf("   \t[3] Stage 1-2 (day 1-
2)\n\n");
195.             printf("   \t[4] Back\n\n");
196.             printf("\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb
\b\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb
\b\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\n\n\n");
197.             salahst :
198.             printf("Pilih Menu : ");
199.             scanf("%d",&stage);
200.             if(stage == 1){
201.                 printf("\nHarga = Rp800.000\n");

```

```

202.             harga = 800000;
203.         }
204.         else if(stage == 2){
205.             printf("\nHarga = Rp800.000\n");
206.             harga = 800000;
207.         }
208.         else if(stage == 3){
209.             printf("\nHarga = Rp1.400.000\n");
210.             harga = 1400000;
211.         }
212.         else if(stage == 4){
213.             system("cls");
214.             goto menuholo;
215.         }
216.         else{
217.             printf("Input Salah, Tolong Masukan
Input Yang Benar!\n");
218.             goto salahst;
219.         }
220.     }
221.     else if(plhkn == 3){
222.         printf("Harga = Rp2.000.000\n");
223.         harga = 2000000;
224.     }
225.     else if(plhkn == 4){
226.         system("cls");
227.         goto atas;
228.     }
229. }
230. else if(strcmp(select,"4")==0){
231.     system("cls");
232.     printf("\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb
\b\xdb\xdb\xdb\xdb\xdb\xdb Tiket Museum Malang \xdb\xdb\xdb\xdb\xdb\xdb
\b\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\n\n\n");
233.     printf("\t[1] Museum Brawijaya\n\n");
234.     printf("\t[2] Back\n\n");
235.     printf("\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb
\b\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb
\b\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb
\b\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\xdb\n\n
");
236.     salah4 :
237.     printf("Pilih Tiket: ");
238.     scanf("%d",&plhms);
239.     if(plhms == 1){
240.         if(s == 0){
241.             printf("\nHarga = Rp50.000\n");
242.             harga = 50000;
243.         }
244.         else{
245.             printf("\nHarga = Rp20.000\n");
246.             harga = 20000;
247.         }
248.     }
249.     else if(plhms == 2){
250.         system("cls");
251.         goto atas;
252.     }
253.     else{
254.         printf("Input Salah, Tolong Masukan Inpu
ut Yang Benar!\n");

```

[illegible]

```

309.         printf("\nNama : ");
310.         scanf("\n%[^\\n]s", &inp.nama);
311.         fprintf(f, "%d#%d#%d", inp.hari, inp.bulan, in
    p.tahun);
312.         fprintf(f, "%s", inp.nama);
313.         if(strcmp(select, "1")==0) {
314.             fprintf(f, "#Sport");
315.         }
316.         else if(strcmp(select, "2")==0) {
317.             fprintf(f, "#Film");
318.         }
319.         else if(strcmp(select, "3")==0) {
320.             fprintf(f, "#Music");
321.         }
322.         else if(strcmp(select, "4")==0) {
323.             fprintf(f, "#Museum");
324.         }
325.         else if(strcmp(select, "5")==0) {
326.             fprintf(f, "#Park");
327.         }
328.         if(strcmp(select, "1")==0) {
329.             if(plhbl == 1) {
330.                 fprintf(f, "#Arema vs Persija");
331.             }
332.             else if(plhbl == 2) {
333.                 fprintf(f, "#Persib vs Persibaya");
334.             }
335.         }
336.         else if(strcmp(select, "2")==0) {
337.             if(plhbs == 1) {
338.                 fprintf(f, "#Kimetsu no Yaiba");
339.             }
340.             else if(plhbs == 2) {
341.                 fprintf(f, "#Sweet Home");
342.             }
343.             else if(plhbs == 3) {
344.                 fprintf(f, "#Asih 2");
345.             }
346.         }
347.         else if(strcmp(select, "3")==0) {
348.             if(plhkn == 1) {
349.                 fprintf(f, "#AKB48");
350.             }
351.             else if(plhkn == 2) {
352.                 if(stage == 1) {
353.                     fprintf(f, "#Holofest 2nd1");
354.                 }
355.                 else if(stage == 2) {
356.                     fprintf(f, "#Holofest 2nd2");
357.                 }
358.                 else if(stage == 3) {
359.                     fprintf(f, "#Holofest 2nd1-
    2");
360.                 }
361.             }
362.             else if(plhkn == 3) {
363.                 fprintf(f, "#Black Pink");
364.             }
365.         }

```



```

366.         else if(strcmp(select,"4")==0){
367.             fprintf(f,"#Museum Malang");
368.         }
369.         else if(strcmp(select,"5")==0){
370.             if(park == 1){
371.                 fprintf(f,"#Jatim Park 1");
372.             }
373.             else if(park == 2){
374.                 fprintf(f,"#Jatim Park 2");
375.             }
376.             else if(park == 3){
377.                 fprintf(f,"#Jatim Park 3");
378.             }
379.             else if(park == 4){
380.                 fprintf(f,"#Hawaii");
381.             }
382.         }
383.         fprintf(f,"%d",banyak);
384.         if(strcmp(select,"1")==0){
385.             if(plhbl == 1){
386.                 fprintf(f,"#200000");
387.             }
388.             else if(plhbl == 2){
389.                 fprintf(f,"#200000");
390.             }
391.         }
392.         else if(strcmp(select,"2")==0){
393.             if(s == 0){
394.                 fprintf(f,"#50000");
395.             }
396.             else{
397.                 fprintf(f,"#35000");
398.             }
399.         }
400.         else if(strcmp(select,"3")==0){
401.             if(plhkn == 1){
402.                 fprintf(f,"#500000");
403.             }
404.             else if(plhkn == 2){
405.                 if(stage == 1){
406.                     fprintf(f,"#800000");
407.                 }
408.                 else if(stage == 2){
409.                     fprintf(f,"#800000");
410.                 }
411.                 else if(stage == 3){
412.                     fprintf(f,"#1400000");
413.                 }
414.             }
415.             else if(plhkn == 3){
416.                 fprintf(f,"#2000000");
417.             }
418.         }
419.         else if(strcmp(select,"4")==0){
420.             if(s == 0){
421.                 fprintf(f,"#50000");
422.             }
423.             else{
424.                 fprintf(f,"#20000");

```

[illegible]

[illegible]

[illegible]

BAB III

PEMBAGIAN TUGAS

A. Acmad Aqil Susanto

Bertugas untuk membuat interface menu atau user interface pada bagian main source code dan fitur input data pengguna. Aqil mengawasi pencodingan untuk membuat program sesuai studi kasus dengan membuat variabel – variabel sesuai dengan jenis tipe data di dalam struct data, yang terdiri dari variabel penampung tanggal transaksi (hari, bulan, tahun), nama pelanggan (nama), jenis tiket (jenis), nama event (event), jumlah tiket (banyak), harga tiket (harga), dan total pembayaran (total). Tidak lupa juga Aqil memberikan variabel sesuai tipe data untuk fitur stok tiket yang tersedia (stok, jumlah, sisa). Lalu Aqil memberikan ide – ide dalam program, membantu Zulham dalam mencari solusi terhadap permasalahan mengenai sorting pada program, serta merapikan dan melakukan pengecekan ulang secara berkali – kali dikarenakan adanya beberapa bug - bug yang muncul dalam program tersebut.

B. Gerry William Nanlohy

Bertugas untuk membuat laporan mengenai penjelasan segala proses pengerjaan Case Study Lab Algo 2020 pada Case Study C – Ticketing, mulai dari awal cover (judul laporan), content (isi laporan) yang terdiri dari bab gambaran umum laporan (penjelasan tentang latar belakang studi kasus yang dipilih, penjelasan tentang seluruh tampilan program, semua fitur program beserta penjelasan masing – masing fitur, dan penjelasan semua proses dalam pemrograman), lalu bab hasil laporan (screenshot atau bukti gambar program beserta penjelasan setiap gambar dan menyisipkan source code sesuai apa yang ada pada compiler C program), terakhir bab pembagian tugas (job description). Laporan Gerry kerjakan sesuai format yang telah ditentukan pada keterangan di file Study Case Lab Algo 2020.pdf. Selain itu Gerry melakukan pengecekan ulang source code dan menjalankan program untuk melakukan screenshot terhadap semua tampilan program. Untuk tambahan Gerry memberikan nomor baris dan warna baris yang berbeda pada source code yang disisipkan pada laporan ini.

C. Muhammad Zulham Firdaus

Bertugas untuk membuat pendataan ticket yang terjual, membuat sorting untuk melakukan pencetakan laporan berdasarkan pengurutan tanggal secara ascending (pengurutan dari yang terkecil ke yang terbesar), ketiga melakukan pencetakan laporan pengurutan tiket berdasarkan penjualan tiket paling banyak secara descending (pengurutan dari yang terbesar ke yang terkecil). Selain itu Zulham juga membuat dan merapikan file processing pada bagian main source code, mencari solusi terhadap permasalahan mengenai sorting pada program, membuat seluruh data yang akan diinput dalam mendemostrasikan inputan pengguna aplikasi tersebut. Zulham juga memberikan ide – ide dalam program, membantu Aqil dalam membuat dan merapikan file processing pada bagian main source code seperti tidak sinkronnya bentuk tabel, terkadang menu tidak dapat muncul dan lain sebagainya, serta memecahkan permasalahan pada program.

Untuk tambahan seperti ini lah contoh inputan pengguna yang di buat oleh Aqil dan Zulham saat mendemonstrasikan program ini kepada Gerry :

#10#12#2020#gwen#Film#Sweet Home#5#35000#175000

#5#12#2020#agil#Music#AKB48#10#500000#5000000

#20#12#2020#zul#Film#Sweet Home#15#50000#750000