

SISTEM PEMESANAN TIKET BIOSKOP XYZ

Aji Sakti Saputra¹, Amelia Regina Permana², Najlia Intani³,
Rizwar Syefulloh⁴, Reva Anwar⁵, Riyan Putra Pratama⁶

Kelompok 2 | Mata Kuliah Pemrograman Dasar | Universitas Koperasi Indonesia

ajisakti554@gmail.com¹, ameliareginapermana16@gmail.com², najliaintani@gmail.com³,
rizwersaepuluh@gmail.com⁴, hi.revaanwar@gmail.com⁵, riyanputrapratama78@gmail.com⁶

ABSTRAK

Penelitian ini dilatarbelakangi oleh keterbatasan waktu dan jarak bagi pengunjung yang harus datang ke bioskop untuk memesan tiket jauh dari jam tayang, serta proses manual di loket tiket seringkali memakan waktu dan mengganggu pengalaman pengunjung. Untuk merespons hal itu, maka peneliti melakukan penelitian yang bertujuan untuk mengoptimalkan proses pemesanan tiket dengan menghadirkan sistem berbasis C++ yang mudah dan efisien. Tujuan yang hendak dicapai pada penelitian ini antara lain: (1) Membuat program sistem pemesanan tiket bioskop dengan konsep OOP (2) Membuat program yang mudah dipahami pengguna. Jenis penelitian ini adalah Studi Literatur dan Analisis Data, secara deskriptif menggunakan pendekatan kualitatif dimana data-data sekunder diperoleh melalui Jurnal dan sumber-sumber lainnya. Adapun hasil penelitian ini adalah: (1) Program sistem pemesanan tiket bioskop berbasis C++ dengan konsep OOP yang memudahkan pemeliharaan program, (2) Program yang telah dibuat dapat dengan mudah dipahami oleh pengguna.

Kata kunci: Sistem, C++, Bioskop.

1. PENDAHULUAN

Saat ini, film merupakan media hiburan yang paling digemari untuk dilakukan oleh banyak orang. Tidak jarang ketika film terbaru yang ditunggu-tunggu rilis, bioskop ramai dengan para pengunjung yang antusias. Hal ini membuat antrian panjang di loket tiket yang membuat pengunjung malas mengantri. Untuk mengatasi hal tersebut, dibuatlah sebuah sistem yang bertujuan untuk mengoptimalkan waktu dan biaya bagi pengunjung yang ingin memesan tiket bioskop. Beberapa masalah yang diidentifikasi adalah keterbatasan waktu dan jarak bagi pengunjung yang harus datang ke bioskop untuk memesan tiket jauh dari jam tayang, serta proses manual di loket tiket seringkali memakan waktu dan mengganggu pengalaman pengunjung. Tujuan utama dari sistem ini adalah mengoptimalkan proses pemesanan tiket dengan menghadirkan sistem berbasis C++ yang mudah dan efisien. Sistem ini diharapkan memberikan kenyamanan dan kemudahan bagi pengunjung dalam memesan tiket secara online yang hemat waktu dan biaya. Kemudahan dalam memilih film, jam tayang, dan tempat duduk dapat dilakukan kapan saja dimana saja. Dengan sistem ini, pengalaman pengunjung dalam memesan tiket menjadi lebih baik dan lebih modern.

2. TINJAUAN PUSTAKA

2.1. Konsep Dasar C++

Menurut buku Cepat Menguasai C++ terbitan Jubilee Enterprise (2015: 10), C++ adalah bahasa pemrograman hasil pengembangan bahasa C. Jika dilihat dari sejarahnya, bahasa C++ ini muncul di tahun 1970-an. Bahasa ini dikembangkan oleh seseorang bernama Bjarne Stroustrup. C++ merupakan subset dari bahasa C, maka pustaka yang ditulis dalam bahasa C dapat dipergunakan dalam bahasa C++.

Bahasa C++ digolongkan ke dalam Bahasa tingkat menengah (Middle Level Programming Language). Kode C++ bersifat case sensitive, artinya membedakan antara huruf kapital dengan huruf kecil. Keistimewaan dari bahasa C++ adalah karena bahasa ini mendukung pemrograman berarah objek atau yang lebih dikenal dengan istilah Objek Oriented Programming (OOP). Dengan Objek Oriented Programming (OOP), mempermudah programmer menulis program, mempercepat proses pembuatan program, dan mempermudah pemeliharaan program.

2.2. Aggregation

Atribut aggregatable digunakan untuk menunjukkan bahwa kelas mendukung agregasi. Ini berarti Objek COM dapat dibuat secara langsung atau dapat di agregasikan. Penggunaan agregasi dalam pemrograman berorientasi di C++. Dalam konteksnya, agregasi mengacu pada penggunaan objek lain sebagai bagian dari objek lain, memungkinkan juga pembuatan kode yang modular dan dapat digunakan Kembali. Fungsi di dalamnya termasuk fungsi Aggregate di excel, atribut aggregatable dan agregat di C++ COM, serta penggunaan agregasi dalam program berorientasi objek.

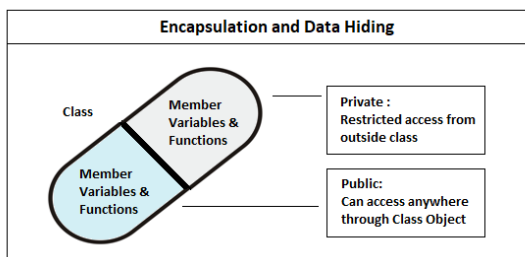
2.3. Exception Handling

Merupakan mekanisme yang paling diperlukan dalam menangani error yang terjadi pada saat runtime (program berjalan) atau yang lebih dikenal dengan sebutan runtime error. Secara umum, adanya kesalahan/error yang terjadi pada program pada saat runtime dapat menyebabkan program berhenti atau hang. Untuk itulah diperlukan mekanisme untuk memastikan bahwa program tetap dapat berjalan meskipun terdapat kesalahan yang terjadi. Tiga model ortogonal untuk penanganan exception handling sebagai berikut :

- Model penangkap, digunakan untuk menangkap kejadian yang menghasilkan pengecualian dalam model data pengecualian; Peristiwa semacam itu dapat dipicu oleh layanan jarak jauh atau oleh data internal (atau metadata) nilai.
- Model pemberitahuan, bertanggung jawab untuk memberitahukan pengecualian yang terjadi kepada pengguna di dalam model hypertext.
- Model penanganan, digunakan untuk menyelesaikan pengecualian, dengan menerapkan kebijakan pemulihan, yang biasanya berdampak pada model data itu sendiri.

2.4. Encapsulation

Menurut Indrajani dan Martin konsep OOP mengenai Enkapsulasi adalah “suatu mekanisme untuk menyembunyikan atau memproteksi suatu proses dari kemungkinan interferensi atau penyalahgunaan dari luar sistem sekaligus menyederhanakan penggunaan sistem sekaligus menyederhanakan penggunaan sistem itu sendiri”. Enkapsulasi adalah proses menghubungkan data dan fungsi atau metode yang beroperasi pada data tersebut menjadi satu unit yang terlindungi dari campur tangan dan penyalahgunaan pihak luar. Dengan menggunakan konsep pemrograman berorientasi objek dan menaruhnya kepada konsep OOP yang lain dikenal sebagai “Penyembunyian Data”. Enkapsulasi menyembunyikan data dan anggotanya, sedangkan abstraksi hanya mengungkap hal-hal spesifik yang diperlukan untuk berinteraksi dengan dunia luar.



Gambar 1 Konsep Encapsulation dan Data Hiding

2.5. Class

a. Private

Merupakan variabel yang dideklarasikan sebagai private hanya dapat diakses dan diubah dalam class yang sama. Private digunakan untuk melindungi variabel dari akses langsung oleh objek lain, sehingga memberikan control yang lebih baik terhadap data dan perilaku kelas.

b. Setter

Merupakan fungsi yang digunakan untuk mengubah nilai suatu variabel. Setter biasanya digunakan untuk melakukan operasi tambahan sebelum mengubah nilai variabel, seperti validasi input atau logging.

c. Getter

Merupakan fungsi yang digunakan untuk mengembalikan nilai suatu variabel. Getter biasanya digunakan untuk memberikan akses ke nilai variabel tanpa mengubah nilai tersebut, sehingga memberikan control yang lebih baik terhadap data dan perilaku kelas.

d. Public

Dalam C++ berarti, variabel yang dideklarasikan sebagai public dapat diakses dan diubah oleh objek lain. Public digunakan untuk memberikan akses langsung ke variabel, tetapi ini dapat membahayakan jika tidak dilakukan dengan hati hati, karena dapat menyebabkan data menjadi tidak aman.

2.6. Program Aplikasi

Menurut Kadir (2008:3) program aplikasi adalah program siap pakai atau program yang direka untuk melaksanakan suatu fungsi bagi pengguna atau aplikasi yang lain. Aplikasi juga diartikan sebagai penggunaan atau penerapan suatu konsep yang menjadi pokok pembahasan atau sebagai program komputer yang dibuat untuk menolong manusia dalam melaksanakan tugas tertentu. Aplikasi software yang dirancang untuk penggunaan praktisi khusus, klasifikasi luas ini dapat dibagi menjadi 2 (dua) yaitu:

- Aplikasi software spesialis, program dengan dokumentasi terdapat yang dirancang untuk menjalankan tugas tertentu.
- Aplikasi paket, suatu program dengan dokumentasi terdapat yang dirancang untuk jenis masalah tertentu.

Dari kedua pengertian di atas dapat disimpulkan bahwa aplikasi adalah sekumpulan perintah atau kode yang disusun secara sistematis untuk menjalankan suatu perintah yang diberikan oleh manusia melalui komponen atau hardware komputer yang digunakan oleh manusia dalam menjalankan program aplikasi, dengan demikian bisa membantu manusia untuk memberikan solusi dari apa yang diinginkan.

2.7. Method

Method adalah fungsi yang dimiliki kelas. Method adalah blok kode yang melakukan tugas tertentu dan dapat dipanggil dari bagian lain program (Malik, 2018). Ada dua jenis method dalam pemrograman bahasa C++:

a. Method Return

Memiliki nilai Kembali, contoh: int (sama seperti fungsi).

b. Method Void

Tanpa memiliki nilai kembalian (sama seperti prosedur).

2.8. Function

Function (dalam bahasa Indonesia adalah Fungsi). Function adalah sebuah struktur, Pengelompokan yang mengandung sekelompok pernyataan yang akan dilaksanakan oleh CPU jika nama function tersebut dipanggil untuk dieksekusi, kecuali untuk function utama yaitu `int main()` yang akan dieksekusi secara otomatis. Deklarasi function:

```
//Tanpa Parameter
void cetakPesan () {
    cout<<"Belajar Function C++";
}
//Dengan Parameter
Int tambah (int a, int b){
    return a+b;
}
```

2.9. Library

Dalam pemrograman, library adalah kumpulan kode yang telah ditulis sebelumnya (berisi function, class, modul) dan dapat digunakan kembali untuk menyelesaikan tugas-tugas tertentu. Tujuan utama dari library yaitu menyediakan solusi siap pakai, sehingga developer tidak perlu membangun segalanya dari awal. Library membantu meningkatkan efisiensi dalam pengembangan software dengan mengurangi waktu dan usaha yang diperlukan untuk mengembangkan aspek-aspek dasar sebuah aplikasi. Macam-macam library:

- | | | |
|---------------|----------------|-------------------|
| 1. iostream | 12. errno.h | 23. stdatomic.h |
| 2. Conio.h | 13. fenv.h. | 24. stdbool.h |
| 3. Stdio.h | 14. float.h | 25. stddef.h. |
| 4. String.h | 15. inttypes.h | 26. stdint.h |
| 5. Math.h | 16. iso646 | 27. stdnoreturn.h |
| 6. Windows | 17. llimits.h | 28. tgmath.h |
| 7. Iominp.h | 18. locale.h | 29. threads.h |
| 8. Stdlib.h | 19. setjmp.h | 30. time.h |
| 9. assert.h | 20. signal.h | 31. uchar.h |
| 10. complex.h | 21. stdalign.h | 32. wchar.h. |
| 11. ctype.h | 22. stdarg.h | 33. wctype.h |

2.10. Switch Case

Switch Case adalah percabangan kode program dimana kita membandingkan isi sebuah variabel dengan beberapa nilai. Jika proses perbandingan tersebut menghasilkan true, maka block kode program akan di proses.

2.11. Tipe Data

Tipe data adalah klasifikasi variabel untuk menentukan bentuk keadaan data yang akan disimpan dalam memori dan jenis operasi apa yang dapat dilakukan oleh variabel tersebut. variabel memiliki 3 jenis tipe data yaitu Variabel Numerik, Variabel Text dan Boolean. Tipe-tipe data dalam C++:

1. Tipe data Integer: Tipe data untuk angka bulat seperti 5, 7, atau 48.
2. Tipe data Float/Double: Tipe data untuk angka pecahan seperti 3.14, 5.55, atau 0.00024.
3. Tipe data Boolean: Tipe data yang berisi nilai true atau false.
4. Tipe data Char: Tipe data untuk 1 karakter, seperti 'a', 'Z' atau '%'.
5. Tipe data Void: Tipe data khusus yang menyatakan tidak ada data.

2.12. Atribut

Di C++11, atribut menyediakan cara standar untuk membuat anotasi konstruksi C++ (termasuk tetapi tidak terbatas pada kelas, fungsi, variabel, dan blok) dengan informasi tambahan. Atribut mungkin atau mungkin tidak spesifik vendor. Pengkompilasi dapat menggunakan informasi ini untuk menghasilkan pesan informasi, atau untuk menerapkan logika khusus saat mengkompilasi kode yang diatribusikan. Pengkompilasi mengabaikan atribut apa pun yang tidak dikenalnya, yang berarti tidak dapat menentukan atribut kustom menggunakan sintaks ini.

2.13. Object Oriented Programming

Object Oriented Programming atau disingkat menjadi OOP adalah paradigma pemrograman dalam melakukan pemrograman yang berorientasi kepada object, semua fungsi, data dan pengolahan data akan dibungkus dalam kelas-kelas dan object-object. Masing-masing object dapat memiliki sifat dan tugasnya. Pada paradigma ini, object-object tersebut dapat bekerja sendiri dan juga dapat saling bekerja sama dengan kemungkinan untuk saling berhubungan, seperti menerima, mengirim data kepada object lainnya dan memproses data. Jenis – Jenis OOP:

1. Bahasa OOP Murni adalah sebuah bahasa yang mengharuskan program ditulis hanya berupa object saja. Contoh – Eiffel, Smalltalk, Ruby, Jade dan lain-lain.
2. Bahasa OOP Hybrid adalah bahasa yang dirancang untuk pemrograman object dengan beberapa elemen procedural.
3. Bahasa OOP Hybrid dalam Web sama seperti Bahasa OOP Hybrid, yang berbeda hanyalah, konsep ini sering digunakan dalam pemrograman Web.

3. METODE PENELITIAN

Metode penelitian yang digunakan dalam proyek ini adalah studi literatur dan analisis data. Studi literatur digunakan untuk meninjau literatur tentang sistem pemesanan tiket bioskop, arsitektur program c++ berorientasi objek, dan praktik terbaik untuk mengembangkan program yang dibuat. Analisis data dilakukan dengan analisis model kelas untuk mengevaluasi desain kelas dan hubungan antar kelas dalam model berorientasi, analisis pengujian kasus penggunaan untuk menguji fungsionalitas sistem terhadap berbagai skenario pengguna, serta analisis kualitas program untuk mengevaluasi atribut kualitas program seperti kegunaan, keandalan, dan keamanan. Tatalaksana yang digunakan dapat diuraikan sebagai berikut:

1. Analisis dan Perencanaan
Perencanaan dimulai dengan pemahaman mendalam tentang fitur sistem yang dibutuhkan dan penggunaan kelas-kelas yang telah ditentukan.
2. Desain Struktur Sistem
 - a. Mendefinisikan kelas
 - b. Mendefinisikan hubungan antar kelas
 - c. Mengimplementasikan aggregation
3. Pembuatan Program
Program dibuat sesuai dengan ketentuan seluruh materi dan fitur yang dibutuhkan menjadi satu kesatuan sehingga menghasilkan output yang ingin dicapai.
4. Pengujian Awal
Pengujian dilakukan untuk memastikan semua fungsi berjalan dengan baik dan semua kasus pengecualian dihandle dengan benar.
5. Evaluasi
Hal ini dilakukan untuk mengidentifikasi tingkat keberhasilan dan juga segala macam error yang terjadi untuk dilakukan perbaikan dan pengembangan system menjadi lebih baik.
6. Pengujian Final
Setelah evaluasi dan perbaikan terhadap berbagai masalah, dilakukan pengujian final untuk hasil akhir yang sesuai.

4. HASIL PENELITIAN DAN PEMBAHASAN

Program Sistem Pemesanan Tiket Bioskop XYZ ini dibuat berbasis C++ dengan konsep Object Oriented Programming (OOP) untuk memudahkan pembuatan program dan pemeliharaan program. Direktif preprocessor yang digunakan dan beberapa berperan sebagai library yakni: (1) iostream untuk operasi input/output, (2) vector untuk mengolah koleksi penyimpanan data yang dapat berubah ukurannya, (3) string untuk mengolah string di C++, (4) stdexcept untuk mengolah kesalahan di C++, (5) algorithm untuk mengolah data, seperti sorting dan searching. Class yang digunakan dalam sistem ini yakni movie class, showtime class, ticket class, tikethistori class, cinema class, dan customer class. Atribut yang dimuat dari class yang digunakan terdiri dari title, duration, genre, time, availableSeats, seatNumber, dan customerName. Agregasi digunakan untuk merepresentasikan hubungan antar kelas. Misalnya, objek Showtime terdiri dari objek movie, dan objek ticket terdiri dari objek showtime dan objek customer. Function yang digunakan dalam program akan menjalankan perintah sesuai dengan pilihan yang dimasukkan oleh pengguna. Berikut adalah hasil dari pengujian program:

```

#####  ## #####  #####  ##  ## #####  #####  ##  ##  ## #####
##  ##  ##  ##  ##  ##  ##  ##  ##  ##  ##  ##  ##  ##  ##  ##
##  ##  ##  ##  ##  ##  ##  ##  ##  ##  ##  ##  ##  ##  ##
#####  ##  ##  ##  #####  ##  ##  ##  #####  ##  ##  ##
##  ##  ##  ##  ##  ##  ##  ##  ##  ##  ##  ##  ##  ##  ##
##  ##  ##  ##  ##  ##  ##  ##  ##  ##  ##  ##  ##  ##
#####  ##  #####  #####  ##  ##  #####  ##
#####  ##  #####  #####  ##  ##  #####  ##

=====
|          Sistem Pemesanan Tiket Bioskop          |
|          Selamat Datang di Bioskop XYZ            |
|=====|
|          DAFTAR MENU                             |
|=====|
| 1. | Menambahkan film                             |
| 2. | Menghapus film                               |
| 3. | Menambahkan jadwal tayang                     |
| 4. | Menghapus jadwal tayang                       |
| 5. | Melihat daftar film yang sedang tayang        |
| 6. | Melihat jadwal tayang untuk film tertentu    |
| 7. | Memesan tiket untuk jadwal tayang tertentu   |
| 8. | Melihat daftar tiket yang telah dipesan       |
| 9. | Melihat riwayat pemesanan oleh pelanggan     |
|10. | Keluar                                         |
|=====|
|          Masukkan pilihan Anda:                    |

```

Gambar 2 Output Program

Fungsi 'showMenu()' digunakan untuk menampilkan menu utama yang terdiri dalam 10 pilihan:

1. Menambahkan film
2. Menghapus film
3. Menambahkan jadwal tayang
4. Menghapus jadwal tayang
5. Melihat daftar film yang sedang tayang
6. Melihat jadwal tayang untuk film tertentu
7. Memesan tiket untuk jadwal tayang tertentu
8. Melihat daftar tiket yang telah dipesan
9. Melihat riwayat pemesanan oleh pelanggan
10. Keluar

Fungsi ini menggunakan perintah cout untuk menampilkan teks dan gambar yang membentuk menu. Gambar yang digunakan adalah sebuah logo bioskop yang dibuat menggunakan karakter ASCII. Pada fungsi 'main()' program akan memanggil fungsi 'showMenu()' secara berulang-ulang hingga pengguna memilih pilihan 10 (Keluar). Setiap kali pengguna memilih sebuah pilihan, program akan membersihkan layar menggunakan perintah 'system("cls")' dan menampilkan menu lagi. Pengguna dapat memilih pilihan dengan memasukkan angka yang sesuai dengan pilihan yang diinginkan. Setelah pengguna memasukkan pilihan, program akan menampilkan garis pembatas dan meminta pengguna untuk memasukkan input lagi.

```
Masukkan judul film: Dilan 1990
Masukkan durasi film (dalam menit): 120
Masukkan genre film: romance
```

Gambar 3 Output Menambahkan Film

Case 1: Menambahkan Film. Pengguna diminta memasukkan judul film, durasi film, dan genre film. Setelah itu, film baru akan ditambahkan ke dalam daftar film yang sedang tayang.

```
C:\Users\riyan\Desktop\PI x + v
Masukkan judul film yang akan dihapus: Dilan 1990
Film berhasil dihapus.
Press any key to continue . . .
```

Gambar 4 Output Menghapus Film

Case 2: Menghapus Film. Pengguna diminta memasukkan judul film yang akan dihapus. Setelah itu, film yang sesuai dengan judul yang dimasukkan oleh pengguna akan dihapus dari daftar film yang sedang tayang.

```
C:\Users\riyan\Desktop\PI x + v
Masukkan judul film: Dilan 1990
Masukkan waktu tayang (contoh: 10:00 AM): 12.00 am
Masukkan jumlah kursi tersedia: 50
Press any key to continue . . .
```

Gambar 5 Output Menambahkan Jadwal Tayang

Case 3: Menambahkan jadwal tayang. Pengguna diminta memasukkan judul film, waktu tayang, dan jumlah kursi tersedia. Setelah itu, jadwal tayang baru akan ditambahkan ke daftar jadwal tayang untuk film yang sesuai.

```
C:\Users\riyan\Desktop\PI x + v
Masukkan waktu tayang yang akan dihapus: Dilan 1990 Time 12.00 am
Jadwal tayang berhasil dihapus.
Press any key to continue . . .
```

Gambar 6 Output Menghapus Jadwal Tayang

Case 4: Menghapus jadwal tayang. Pengguna diminta memasukkan waktu tayang yang akan dihapus. Setelah itu, jadwal tayang yang sesuai dengan waktu tayang yang dimasukkan akan dihapus dari daftar jadwal tayang.

```
C:\Users\riyan\Desktop\PI x + v
Movie(title=Dilan 1990, duration=120, genre=romance)
Movie(title=Stars wars, duration=120, genre=action)
Press any key to continue . . .
```

Gambar 7 Output Melihat Daftar Film Yang Sedang Tayang

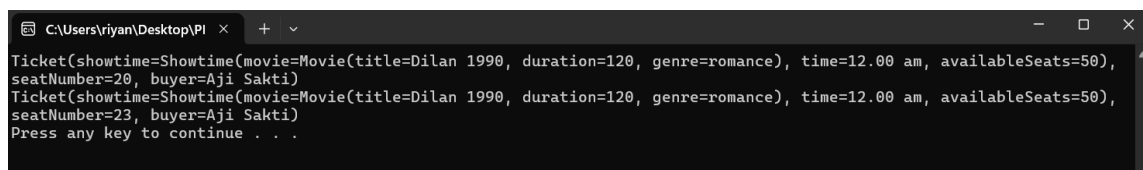
Case 5: Melihat daftar film yang sedang tayang. Program akan menampilkan daftar film yang sedang tayang dengan atribut title, duration, dan genre.

```
C:\Users\riyan\Desktop\PI x + v
Masukkan judul film: Dilan 1990
Showtime(movie=Movie(title=Dilan 1990, duration=120, genre=romance), time=12.00 am, availableSeats=50)
Press any key to continue . . .
```

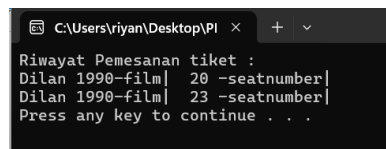
Gambar 8 Output Melihat Jadwal Tayang Untuk Film Tertentu

Case 6: Melihat jadwal tayang film tertentu. Pengguna diminta memasukkan judul film. Setelah itu, program akan menampilkan daftar jadwal tayang untuk film yang sesuai.

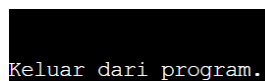
Case 7: Memesan tiket untuk jadwal tayang film tertentu. Pengguna diminta memasukkan judul film dan nomor kursi yang diinginkan. Setelah itu, program akan memesan tiket untuk jadwal tayang yang sesuai dan menampilkan konfirmasi pemesanan.



Case 8: Melihat daftar tiket yang telah dipesan. Program akan menampilkan daftar tiket yang telah dipesan.



Case 9: Melihat Riwayat pemesanan oleh pelanggan. Program akan menampilkan riwayat pemesanan.



Case 10: Keluar. Program akan keluar dari sistem pemesanan tiket bioskop.

5. KESIMPULAN

8 | Kelompok 2

6. DAFTAR PUSTAKA

- [1] Andre. 2020. *Tutorial Belajar C++ Part11: Jenis-jenis Tipe Data dalam Bahasa C++*. <https://www.duniailkom.com/tutorial-belajar-c-plus-plus-jenis-jenis-tipe-data-dalam-bahasa-c-plus-plus/>. Diakses pada 21 Juni 2024.
- [2] Andre. 2021. *Pengertian Setter dan Getter*. <https://www.duniailkom.com/tutorial-belajar-oop-c-plus-plus-pengertian-setter-dan-getter/>. Diakses pada 14 Juni 2024.
- [3] Enterprise, Jubilee. 2015. *Cepat Menguasai C++: Ringkas dan Tepat Mempelajari C++ untuk Programmer Pemula/Penulis*. Jakarta: Elex Media Komputindo.
- [4] Fajar. 2018. *Pengertian Function*. <https://www.belajarcpp.com/tutorial/cpp/function/>. Diakses pada 21 Juni 2024.
- [5] Fajar. 2019. *Penjelasan dan Konsep OOP (Object Oriented Programming)*. <https://www.belajarcpp.com/tutorial/cpp/penjelasan-oop/>. Diakses pada 21 Juni 2024.
- [6] Febrian, Ade. 2020. *Cinekom – Cinema Amikom*. <https://github.com/Adefebrian/Cinekom>. Diakses pada 14 Juni 2024.
- [7] Frigo. 2011. *Private Set / get Functions—why private and how to use?*. <https://stackoverflow.com/questions/7169492/private-set-get-functions-why-private-and-how-to-use>. Diakses pada 14 Juni 2024.
- [8] Indrajani dan Martin. 2007. *Pemrograman Berbasis Objek Dengan Bahasa Java*. Jakarta: PT Alex Media Komputindo.
- [9] Kadir, Abdul. 2008. *Tuntunan Praktis Belajar Database Menggunakan MySQL*. Yogyakarta: CV Andi Offset.
- [10] Brambilla, M., Comai S., and Tziviskou C. 2006. *Exception Management Within Web Applications Implementing Business Processes*. <https://www.springer.com/de/impressum/610232>. Diakses pada 14 Juni 2024.
- [11] Malik, Adam. 2018. *Pengantar Statistika Pendidikan*. Edited by Minan Chusni. Yogyakarta: Deepublish.
- [12] Microsoft team. 2022. *AGGREGATE (Fungsi AGGREGATE)*. <https://support.microsoft.com/id-id/office/aggregate-fungsi-aggregate-43b9278e-6aa7-4f17-92b6-e19993fa26df>. Diakses pada 14 Juni 2024.
- [13] Prasetyo, Andi. 2018. *Aplikasi Ticketing Bioskop Menggunakan Bahasa C++ Berbasis Code Block*. https://www.academia.edu/37995746/APLIKASI_TICKETING_BIOSKOP_MENGGUNAKAN_BAHASA_C_BERBASIS_CODE_BLOCK. Diakses pada 14 Juni 2024.
- [14] Reddit team. 2022. *What is the difference between a private variable and a getter and setter?* https://www.reddit.com/r/learnprogramming/comments/xf5fp3/what_is_the_difference_between_a_private_variable/?rdt=46763. Diakses pada 14 Juni 2024.
- [15] RevoU team. 2024. *Apa itu Bahasa C++? Pengertian dan contoh*. <https://revou.co/kosakata/bahasa-cpp>. Diakses pada 14 Juni 2024.
- [16] RevoU team. 2024. *Apa itu Library?* <https://revou.co/kosakata/library>. Diakses pada 21 Juni 2024.
- [17] Swatter, Potato. 2011. *Private Set / get Functions—why private and how to use?* <https://stackoverflow.com/questions/7169492/private-set-get-functions-why-private-and-how-to-use>. Diakses pada 14 Juni 2024.
- [18] Selfadjri, Dhima. 2019. *Tipe Data C++, Deklarasi Fungsi/Method pada C++, Macam-macam Library pada C++, Pengertian Object Oriented Programming*. <https://snxel.home.blog/2019/03/16/tipe-data-c-deklarasi-fungsi-method-pada-c-macam-maca-library-pada-c-pengertian-object-oriented-programing/>. Diakses pada 21 Juni 2024.
- [19] Wongso, Rini. 2018. *Exception Handling*. <https://socs.binus.ac.id/2018/12/05/exception-handling/>. Diakses pada 14 Juni 2024.