

SISTEM PENGELOLAAN EVENT KONSER

LAPORAN



Disusun oleh:
KELOMPOK 6

DWI PEBRIYANTO PRADANA (2409116012)
ISRINA LUTHFIAH (2409116003)
MUHAMMMAD SADIKIN SAMIR (2409116031)

Asisten Laboratorium:

Muhammad Nabil Ma'ruf
(2309116035)

Muhammad Rezky Setiawan
(2309116039)

PROGRAM STUDI SISTEM INFORMASI

FAKULTAS TEKNIK

UNIVERSITAS MULAWARMAN

2024

KATA PENGANTAR

Dengan memanjatkan puji syukur kepada Tuhan Yang Maha Esa yang telah memberikan nikmat dan karunia-Nya, sehingga kami dapat menyelesaikan tugas praktikum Dasar-dasar Pemrograman yang berjudul “Sistem Pengelolaan Event Konser” tepat pada waktunya.

Dalam penyusunan laporan ini melibatkan pemahaman konsep mendalam tentang Pemrograman coding, dan flowchart alur diagram. Selain itu, laporan ini juga bertujuan untuk menambah wawasan tentang pembuatan program untuk studi kasus sistem pengelolaan event konser.

Kami menyadari bahwa penyusunan laporan ini tidak terlepas dari berbagai kendala, namun dengan kerja keras dan semangat untuk terus belajar, laporan ini berhasil diselsaikan. Kami juga ingin mengucapkan terimakasih kepada pihak-pihak yang telah membantu. Pihak-pihak yang terkait itu di antaranya sebagai berikut:

Abang Muhammad Nabil Ma’ruf selaku asisten laboratorium yang telah memberikan bimbingan, nasihat, bantuan serta arahan kepada kami selama praktikum,

Abang Muhammad Rezky Setiawan selaku asistem laboratorium yang memberikan bimbingan kepada kami selama pertemuan praktikum berlangsung.

Laporan ini menjelaskan program yang telah kami buat dengan menerapkan semua modul yang telah diberikan selama praktikum. Semoga laporan ini dapat memberikan manfaat bagi pembacanya. Kami menyadari bahwa dalam proses pembuatan laporan ini masih terdapat banyak kesalahan, oleh karena itu kami mengharapkan segala bentuk kritik dan saran yang bersifat membangun demi perbaikan laporan kami selanjutnya.

Samarinda,6 November 2024

Penyusun

DAFTAR ISI

KATA PENGANTAR	i
DAFTAR ISI.....	ii
DAFTAR GAMBAR	iii
DAFTAR TABEL.....	v
BAB I	1
PENDAHULUAN	1
1.1 Deskripsi Masalah	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan.....	2
1.5 Manfaat.....	3
BAB II.....	4
PERANCANGAN	4
2.1 Analisis Program.....	4
2.2 Flowchart	4
BAB III HASIL DAN PEMBAHASAN.....	6
3.1 Implementasi Program.....	6
Gambar 3.7 Implementasi: Update data konser	10
3.2 Alur Program	14
.....	15
3.3 Source Code.....	17
BAB IV	57
PENUTUP.....	57
DAFTAR PUSTAKA	58
LAMPIRAN.....	59

DAFTAR GAMBAR

Gambar 2.1 Flowchart 1.1	14
Gambar 2.2 Flowchart 1.2	15
Gambar 2.3 Flowchart 1.3	16
Gambar 2.4 Flowchart 1.4	17
Gambar 2.5 Flowchart 1.5	18
Gambar 2.6 Flowchart 1.6	19
Gambar 3.1 Implementasi: Python	20
Gambar 3.2 Implementasi: Dictionary akun admin dan user	21
Gambar 3.3 Implementasi: Dictionary data barang dan sewa untuk export	21
Gambar 3.4 Implementasi: Dictionary kasir pemesanan	21
Gambar 3.5 Implementasi: Dictionary kasir penyewaan	21
Gambar 3.6 Implementasi: Dictionary data pelanggan sewa	21
Gambar 3.7 Implementasi: While True loop untuk mengulang proses login	22
Gambar 3.8 Implementasi: While True loop untuk mengulang input	22
Gambar 3.9 Implementasi: While dengan suatu kondisi	23
Gambar 3.10 Implementasi: Library PrettyTable	24
Gambar 3.11 Implementasi: Import modul pandas	24
Gambar 3.12 Implementasi: Penggunaan pandas	25
Gambar 3.13 Implementasi: Create data 1	25
Gambar 3.14 Implementasi: Create data 2	26
Gambar 3.15 Implementasi: Read data 1	26
Gambar 3.16 Implementasi: Read data 2	26
Gambar 3.17 Implementasi: Update data 1	27
Gambar 3.18 Implementasi: Update data 2	27
Gambar 3.19 Implementasi: Delete data	28
Gambar 3.20 Implementasi: Data akun admin dan user	28
Gambar 3.21 Implementasi: Pilihan menu login admin dan user	28
Gambar 3.22 Implementasi: Proses login admin	29
Gambar 3.23 Implementasi: Proses login user dan registrasi user	29
Gambar 3.24 Implementasi: pwininput	29
Gambar 3.25 Implementasi: pwininput pada input password admin	29
Gambar 3.26 Implementasi: pwininput pada input password user	30
Gambar 3.27 Implementasi: Export data ke Excel/CSV data barang-sewa	30
Gambar 3.28 Implementasi: Export data ke Excel/CSV data pemesanan barang	30
Gambar 3.29 Implementasi: Export data ke Excel/CSV data penyewaan ..	30
Gambar 3.30 Alur Program: Menu login 1	31
Gambar 3.31 Alur Program: Menu login 2	31
Gambar 3.32 Alur Program: Menu login 3	32
Gambar 3.33 Alur Program: Menu login 4	32

Gambar 3.34 Alur Program: Menu Admin	33
Gambar 3.35 Alur Program: Menu User	34
Gambar 3.36 Tampilan Program: Tampilan Awal	53
Gambar 3.37 Tampilan Program: Menu Login	54
Gambar 3.38 Tampilan Program: Login Admin	54
Gambar 3.39 Tampilan Program: Login User	54
Gambar 3.40 Tampilan Program: Registrasi User	55
Gambar 3.41 Tampilan Program: Kontak Admin	55
Gambar 3.42 Tampilan Program: Menu Admin	55
Gambar 3.43 Tampilan Program: Menu edit data barang	56
Gambar 3.44 Tampilan Program: Tambah data barang	57
Gambar 3.45 Tampilan Program: Ubah data barang	58
Gambar 3.46 Tampilan Program: Hapus data barang	59
Gambar 3.47 Tampilan Program: Tampilkan data barang	60
Gambar 3.48 Tampilan Program: Kembali ke menu admin	60
Gambar 3.49 Tampilan Program: Menu edit data sewa	61
Gambar 3.50 Tampilan Program: Tambah data penyewaan	62
Gambar 3.51 Tampilan Program: Ubah data penyewaan	63
Gambar 3.52 Tampilan Program: Hapus data sewa	64
Gambar 3.53 Tampilan Program: Tampilkan data sewa	64
Gambar 3.54 Tampilan Program: Export data	64
Gambar 3.55 Tampilan Program: Hasil export data	65
Gambar 3.56 Tampilan Program: Logout admin	65
Gambar 3.57 Tampilan Program: Menu user	65
Gambar 3.58 Tampilan Program: List harga barang	66
Gambar 3.59 Tampilan Program: Pemesanan barang	67
Gambar 3.60 Tampilan Program: Export data pemesanan	67
Gambar 3.61 Tampilan Program: List harga sewa	68
Gambar 3.62 Tampilan Program: Penyewaan barang	69
Gambar 3.63 Tampilan Program: Export data penyewaan barang	69
Gambar 3.64 Tampilan Program: Logout user	70

DAFTAR TABEL

Tabel 3.1 Source Code	
Tabel 1.1 Tabel Kontribusi	56

BAB I

PENDAHULUAN

1.1 Deskripsi Masalah

Event adalah kegiatan yang dirancang untuk mendatangkan orang-orang ke suatu tempat agar mereka mendapatkan suatu informasi atau pengalaman penting serta tujuan lain yang diselenggarakan oleh penyelenggara. Event dilaksanakan berdasarkan rencana yang telah disusun secara terarah, dan tidak dilakukan secara spontan. Pada laporan kali ini kami berfokus pada event konser musik.

Event konser musik merupakan pertunjukkan musik secara langsung yang dibuat dan dipersembahkan untuk penggemarnya melalui karya lagu dari sang penyanyi. Pada masa lalu, pendaftaran untuk konser dilakukan secara langsung, sedangkan saat ini kemajuan teknologi memungkinkan proses pemesanan dan pembelian tiket dilakukan secara online melalui berbagai perangkat seperti smartphone, computer dan gadget lainnya.

Penggunaan teknologi oleh manusia dalam membantu menyelesaikan pekerjaan menjadi lebih mudah dan efisien. Ketika ingin melakukan pemesanan suatu tiket, maka pengguna harus memiliki aplikasi khusus konser untuk bisa registrasi tanpa harus datang langsung ke tempat serta memudahkan pengguna untuk melihat jadwal konser, melihat harga tiket, dan melihat menu - menu lain yang tersedia pada aplikasi.

Kami menganalisis dan membuat sebuah program sistem pengelolaan event konser yang tujuannya untuk memudahkan admin, pengguna dan para pengunjung, yang di dalamnya terdapat tiga role mode untuk admin yang bisa melakukan CRUD pada data, role pengguna yang dapat mengakses untuk membeli tiket konser dan pengunjung dapat melihat list konser dengan menggunakan aplikasi sederhana dengan bahasa pemrograman python.

1.2 Rumusan Masalah

Berdasarkan deskripsi masalah yang telah dipaparkan diatas, maka rumusan masalah dalam laporan ini adalah sebagai berikut:

1. Bagaimana analisis dari program aplikasi sederhana pengelolaan event konser?
2. Bagaimana flowchart program aplikasi pengelolaan event konser diatas?
3. Bagaimana implementasi program aplikasi sederhana Pengelolaan Event Konser terhadap user?
4. Bagaimana alur dan tampilan program aplikasi sederhana event Konser mudah dipahami oleh user?

1.3 Batasan Masalah

Berdasarkan rumusan masalah di atas, dapat diidentifikasi bahwa diperlukannya batasan-batasan masalah dengan tujuan untuk memfokuskan analisis, implementasi, alur dan tampilan program agar mudah dipahami dan digunakan oleh user. Berikut batasan masalah yang bisa diambil, yaitu:

1. Implementasi menggunakan bahasa pemrograman python
2. Implementasi CRUD (create, read, update, delete) untuk role admin
3. Implementasi 2 menu login untuk role admin dan pengguna
4. Tampilan data konser dalam library Prettytable
5. Penyimpanan username dan password untuk admin atau pengguna dalam database json, namun format file nya terpisah.
6. Pembelian tiket hanya bisa dilakukan 1 kali, jika ingin membeli tiket lagi pengguna harus menghapus tiket sebelumnya
7. Opsi hapus akun untuk admin dan pengguna tidak ada
8. Username bisa menginputkan huruf, angka maupun kode

1.4 Tujuan

Berdasarkan rumusan masalah di atas, maka tujuan yang dapat dipaparkan sebagai berikut:

1. Menjelaskan mengenai analisis program untuk sistem pengelolaan event konser

2. Mengimplementasikan program sistem pengelolaan event konser kepada pengguna.
3. Menjelaskan alur dan tampilan program sistem pengelolaan event konser agar mudah dipahami oleh pengguna.

1.5 Manfaat

Berdasarkan tujuan yang dijelaskan di atas, adapun yang menjadi manfaat dari projek ini sebagai berikut:

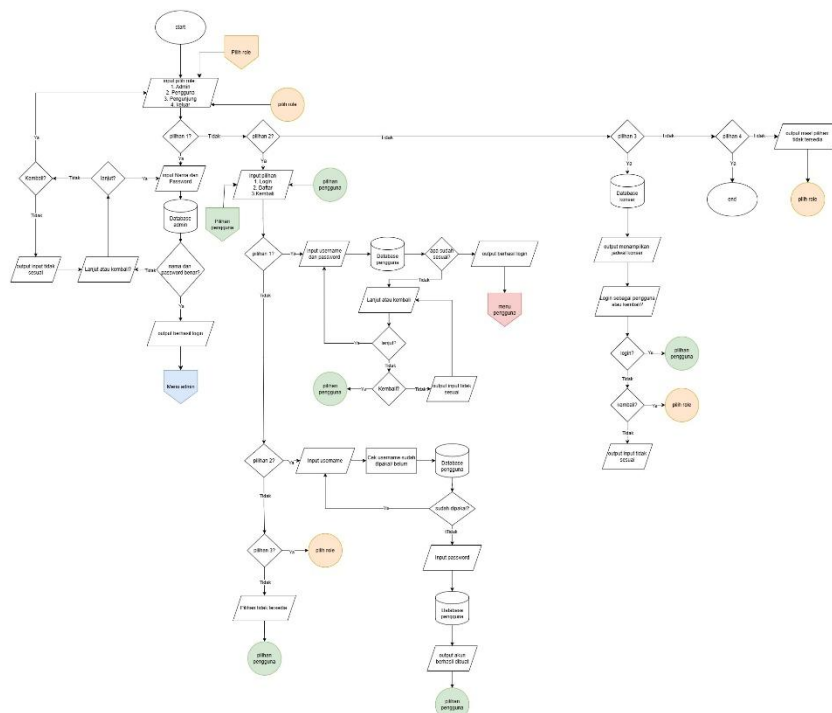
1. Meningkatkan kemampuan untuk menganalisis program yang telah dipelajari bagi mahasiswa untuk studi kasus yang berbeda kedepannya.
2. Untuk mengembangkan kemampuan dari pengimplementasian studi kasus program bagi mahasiswa
3. Dapat menerapkan pemahaman teori selama praktikum dan menjadikannya lebih nyata.

BAB II PERANCANGAN

2.1 Analisis Program

Program ini kami buat untuk mempermudah sistem pengelolaan event konser untuk mendata pengguna sesuai dengan cara melibatkan pemanfaatan teknologi informasi dalam setiap aktivitasnya. Rancangan dari program ini terdapat menu yang dapat diakses khusus oleh admin seperti penambahan data konser, pengecekan data konser, mengupdate data konser dan menghapus data konser, sementara rancangan dari program yang dapat diakses khusus oleh pengguna terdapat menu daftar konser yang berisi lihat event konser, beli tiket konser, top up e-money, dan pengunjung dapat mengakses list jadwal konser sehingga program ini dapat mengefisienkan pengerjaan sistem pengelolaan konser.

2.2 Flowchart



Gambar 2. 1 Flowchart Tampilan menu utama

https://drive.google.com/file/d/1yk62_QaeT4BcOA_QYOnMHYcGPhcPIEGe/view?usp=sharing

BAB III

HASIL DAN PEMBAHASAN

3.1 Implementasi Program

Adapun implementasi program ke dalam program aplikasi sederhana sistem pengelolaan event konser adalah sebagai berikut:

1. PrettyTable

PrettyTable adalah library Python yang digunakan untuk membuat atau mengeluarkan data dalam bentuk tabel yang memberikan kemudahan dalam membaca data, selain itu untuk mempermudah dalam mengetahui perubahan yang terjadi pada suatu hal baik menaik ataupun menurun.

2. Import module

Import untuk membuat kode dalam satu modul tersedia di modul lain. Import dalam Python penting untuk menyusun kode secara efektif.

3. While

Perulangan while pada python adalah proses pengulangan suatu blok kode program selama sebuah kondisi terpenuhi. Penentuan suatu blok kode akan diulang-ulang ditinjau dari (True or False) sampai kondisi yang tidak diinginkan tidak lagi terpenuhi.

4. Create

Create (menambah data) adalah cara bagaimana kita menambah data baru ke suatu database.

5. Read

Read (menampilkan data) adalah cara kita menampilkan data yang tadi telah ditambahkan pada proses sebelumnya yaitu Create (menambah data).

6. Update

Update (mengubah data) adalah proses untuk mengubah data yang sudah ada dalam database.

7. Delete

Delete (menghapus data) adalah proses menghapus data yang telah diinput sebelumnya dari daftar.

8. List

List tipe data yang paling serbaguna yang tersedia dalam bahasa Python, yang dapat ditulis sebagai daftar nilai yang dipisahkan koma (item) antara tanda kurung siku. Hal penting tentang daftar adalah item dalam list tidak boleh sama jenisnya.

9. If Elif Else

Pada python ada beberapa kondisi diantaranya adalah if, else dan elif. Kondisi if digunakan untuk mengeksekusi kode jika kondisi bernilai benar True. Jika kondisi bernilai salah False maka kondisi if tidak akan di eksekusi.

Kondisi If Else adalah kondisi dimana jika pernyataan benar True maka kode dalam if akan dieksekusi, tetapi jika bernilai salah False maka akan mengeksekusi kode di dalam else.

Elif merupakan lanjutan atau percabangan logika dari “kondisi if”. Dengan elif kita bisa membuat kode program yang akan menyeleksi beberapa kemungkinan yang bisa terjadi. Hampir sama dengan kondisi “else”, bedanya kondisi “elif” bisa banyak dan tidak hanya satu

10. For

Pengulangan for pada Python memiliki kemampuan untuk mengulangi item dari urutan apapun, seperti list atau string.

11. PwInput

PwInput berfungsi agar saat user atau admin input password, maka tampilan passwordnya berupa **** disembunyikan tidak diperlihatkan.

12. Dictionary

Dictionary adalah tipe data pada python yang berfungsi untuk menyimpan kumpulan data/nilai dengan pendekatan “key-value”. Setiap urutannya berisi key dan value. Setiap key dipisahkan dari value-nya oleh titik dua (:), item dipisahkan oleh koma, dan semuanya tertutup dalam

kurung kurawal. Dictionary kosong tanpa barang ditulis hanya dengan dua kurung kurawal, seperti ini: {}.

13. Def

Def digunakan untuk mendefinisikan fungsi dalam Python. Fungsi adalah blok kode yang dapat digunakan kembali untuk mengelompokkan kode, sehingga memudahkan debugging dan pemeliharaan kode

14. Try-except

Try except adalah konsep dalam pemrograman Python yang digunakan untuk menangani error atau eksepsi. Contohnya ValueError dan KeyboardInterrupt

15. Json

Json (JavaScript Object Notation) adalah sebuah format data yang digunakan untuk pertukaran dan menyimpan data antar pengguna dan server. File JSON memiliki ekstensi.json serta menggunakan teks yang sama-sama dapat dibaca oleh manusia dan dipahami oleh komputer.

16. Break

Break digunakan untuk menghentikan blok kode yang melakukan perulangan saat kondisi eksternal terpenuhi.

17. Print

Print digunakan untuk menampilkan atau mencetak data ke layar serta dapat menampilkan berbagai tipe data, seperti string, angka dan boolean

18. Shorting

Digunakan untuk mengurutkan data dalam urutan tertentu, baik dari yang terendah maupun tertinggi.

Berikut penggunaan kode implementasi pada bagian program, tertentu:

```
from prettytable import PrettyTable
import json
import pwinput
```

Gambar 3. 1 Implementasi: Import modul

```
#prettytable
def tampilkan_tabel(data_konser):
    tabel = PrettyTable()
    tabel.field_names = ["ID", "Guest Star", "Jadwal", "Harga"]
    for konser_id, data_konser_item in data_konser["konser"].items():
        jadwal = f"{data_konser_item['tanggal']} {data_konser_item['bulan']} {data_konser_item['tahun']}"
        tabel.add_row([konser_id, data_konser_item["gs"], jadwal, f"Rp.{data_konser_item['harga']}"])
    return tabel
```

Gambar 3. 2 Implementasi: Prettytable pada data konser

```
#login admin
def login_admin():
    while True :
        try:
            print(f"\n{"="*5} LOGIN ADMIN {"="*5}")
```

Gambar 3. 3 Implementasi: While True pada login Admin

```
#login akun pengguna
def login_pengguna():
    data_pengguna = memuat_data(json_path)
    while True :
```

Gambar 3. 4 Implementasi: While True pada login pengguna

```
#Menambah konser
def tambah_konser():
    data_konser = memuat_data(json_path3)
    print(f"\n{"="*21} MENAMBAH KONSER {"="*21}")
    print(tampilkan_tabel(data_konser))
    while True:
        try:
            no = (input("masukan angka untuk ID (Contoh:002) = "))
            if no.isdigit() and 3<=len(no)<4 :
                no = no.zfill(3)
                id_konser = f"DRS{no}"
                if id_konser in data_konser["konser"]:
                    print("ID sudah digunakan.")
                else :
                    break
            else:
                print("Harus memasukan 3 angka")
```

Gambar 3. 5 Implementasi: Create data konser

```
#lihat konser admin
def lihat_konser():
    print(f"\n{'='*21} LIST KONSER {'='*21}")
    data_konser = memuat_data(json_path3)
    print(tampilkan_tabel(data_konser))
    while True:
        try:
            print(f"\n{'='*5} PILIH KRITERIA SHORTING {'='*5}")
            print("[1]. Harga terendah ke tertinggi")
            print("[2]. Harga tetinggi ke terendah")
            print("[3]. Kembali")
            pilihan = input("Masukkan nomor pilihan Anda = ").strip()
            if pilihan == "1":
                kriteria = "harga1"
                update_table(data_konser, kriteria)
            elif pilihan == "2":
                kriteria = "harga2"
                update_table(data_konser, kriteria)
            elif pilihan == "3":
                menu_admin()
            else:
                print("Pilihan tidak tersedia.")
```

Gambar 3. 6 Implementasi: Read data konser

```
#Perbarui Konser
def perbarui_konser():
    data_konser = memuat_data(json_path3)
    print(f"\n{'='*21} MEMPERBARUI KONSER {'='*21}")
    print(tampilkan_tabel(data_konser))
    while True :
        try :
            id_konser = input("Masukkan ID konser yang ingin di ubah = ")
            if id_konser in data_konser["konser"]:
                while True:
                    try:
                        gs = input("Masukkan Guest Star = ").strip().lower().capitalize()
                        if 4 <= len(gs) <= 15:
                            if not gs:
                                raise ValueError("Nama Guest Star tidak boleh kosong.")
                            else :
                                break
                    else:
                        print("Nama harus antara 4 hingga 15 karakter.")
```

Gambar 3.7 Implementasi: Update data konser

```
#hapus konser
def hapus_konser():
    data_konser = memuat_data(json_path3)
    print(f"\n{'='*21} HAPUS KONSER {'='*21}")
    print(tampilkan_tabel(data_konser))
    while True :
        try:
            id_konser = input("Masukkan nomor konser yang ingin dihapus = ")
            if id_konser in data_konser["konser"]:
                del data_konser["konser"][id_konser]
                simpan_data(json_path3, data_konser)
                print(f"Konser dengan ID {id_konser} berhasil dihapus.")
                print(f"\n{'='*21} LIST KONSER {'='*21}")
                print(tampilkan_tabel(data_konser))
                while True :
                    yt = input("Ingin menghapus konser lagi(Ya/Tidak)? = ").lower()
                    if yt == "ya":
                        hapus_konser()
                    elif yt == "tidak":
```

Gambar 3.8 Implementasi: Delete data konser


```

data_konser["konser"][str(no)] = {"gs": gs, "tanggal": tanggal, "bulan":
simpan_data(json_path3, data_konser)
print(f'Konser baru berhasil ditambahkan dengan NO {no}.')
print(f"\n{"="*21} LIST KONSER {"="*21}")
print(update_table(data_konser))

```

Gambar 3.9 Implementasi: List perbarui data konser

```

login admin
def login_admin():
    while True :
        try:
            print(f"\n{"="*5} LOGIN ADMIN {"="*5}")
            nama = input("Username = ")
            pw = pwinput.pwinput("Password = ")
            if nama in data_admin and data_admin[nama]==pw:
                menu_admin()
            elif nama in data_admin and data_admin[nama]!=pw:
                print("Password salah!")
                while True :
                    try:
                        kl = input("apakah anda ingin kembali atau lanjut?(kembali/lanjut) = ").lower()
                        if kl == "kembali":
                            pilih_role()
                        elif kl == "lanjut":
                            break
                        else :
                            print("Input tidak sesuai!")
                    except KeyboardInterrupt:
                        print("\nJangan menekan CTRL dan C secara bersamaan!")
            else :
                print("Nama pengguna tidak tersedia.")

```

Gambar 3.10 Implementasi: If Elif Else pada proses login admin

```

#login akun pengguna
def login_pengguna():
    data_pengguna = memuat_data(json_path)
    while True :
        print(f"\n{"="*5} LOGIN PENGGUNA {"="*5}")
        nama = input("Username = ")
        pw = input("Password = ")
        if nama in data_pengguna and data_pengguna[nama]["password"]==pw:
            menu_pengguna(nama)
        elif nama in data_pengguna and data_pengguna[nama]["password"]!=pw:
            print("Password Salah!")
            kembalilanjut()
        else :
            print("Nama pengguna tidak tersedia")
            kembalilanjut()

```

Gambar 3.11 Implementasi: If Elif Else pada proses login pengguna

```
#login admin
def login_admin():
    while True :
        try:
            print(f"\n{"="*5} LOGIN ADMIN {"="*5}")
            nama = input("Username = ")
            pw = pwinput.pwinput("Password = ")
```

Gambar 3.12 Implementasi: Pwinput pada proses login admin

```
#login akun pengguna
def login_pengguna():
    data_pengguna = memuat_data(json_path)
    while True :
        print(f"\n{"="*5} LOGIN PENGGUNA {"="*5}")
        nama = input("Username = ")
        pw = input("Password = ")
```

Gambar 3.13 Implementasi: Pwinput pada proses login pengguna

```
{
    "konser": {
        "DRS002": {
            "gs": "Sal Priadi",
            "tanggal": "23",
            "bulan": "April",
            "tahun": "2025",
            "harga": 160000
        },
        "DRS003": {
            "gs": "Reality Club",
            "tanggal": "23",
            "bulan": "Desember",
            "tahun": "2025",
            "harga": 800000
        }
    }
}
```

Gambar 3.14 Implementasi: Dictionary data konser

```
#login admin
def login_admin():
    while True :
        try:
            print(f"\n{"="*5} LOGIN ADMIN {"="*5}")
            nama = input("Username = ")
            pw = pwinput.pwinput("Password = ")
```

Gambar 3.15 Implementasi: Def login sebagai admin

```
#login akun pengguna
def login_pengguna():
    data_pengguna = memuat_data(json_path)
    while True :
        try:
            print(f"\n{"="*5} LOGIN PENGGUNA {"="*5}")
```

Gambar 3.16 Implementasi: Def login sebagai pengguna

```
while True :
    try:
        gs = input("Masukkan Guest Star = ").strip().lower().capitalize()
        if not gs:
            raise ValueError("Nama Guest Star tidak boleh kosong")
        else :
            break
    except ValueError as e:
        print(e)
```

Gambar 3.17 Implementasi: Try – except

```
#memuat dan menyimpan akun pengguna
json_path = r"D:\PROJECT_AKHIR_DDP\db_pengguna.json"
def memuat_data(json_path):
    try:
        with open(json_path, "r") as databasep:
            data_pengguna=json.load(databasep)
    except FileNotFoundError:
        data_pengguna={}
    return data_pengguna
def simpan_data(json_path, data_pengguna):
    with open(json_path, "w") as databasep :
        json.dump(data_pengguna,databasep, indent =4)
```

Gambar 3.18 Implementasi: Json memuat data

```
while True :
    k1 = input("apakah anda ingin kembali atau lanjut?(Kembali/lanjut)
    if k1 == "Kembali" or k1 == "kembali" or k1 == "KEMBALI":
        pengguna()
    elif k1 == "Lanjut" or k1 == "lanjut" or k1 == "LANJUT":
        break
```

Gambar 3.19 Implementasi: break memberhentikan program

```
def perbarui_konser():
    data_konser = memuat_data(json_path3)
    print(f"\n{"="*21} MEMPERBARUI KONSER {"="*21}")
    print(tampilkan_tabel(data_konser))
```

Gambar 3.20 Implementasi: print menampilkan tabel yang diperbarui

```
#untuk shorting data konsernya
def update_table(data_konser, kriteria):
    tabel = PrettyTable()
    tabel.field_names = ["ID", "Guest Star", "Jadwal", "Harga"]
    if kriteria == "harga1":
        sorted_konser = sorted(data_konser["konser"].items(), key=lambda x: int(x[1]["harga"]))
    elif kriteria == "harga2": sorted_konser = sorted(data_konser["konser"].items(), key=lambda x: int(x[1]["harga"]), reverse=True)
    else:
        raise ValueError("Kriteria tidak dikenal")
    for konser_id, data_konser_item in sorted_konser:
        jadwal = f"{data_konser_item['tanggal']} {data_konser_item['bulan']} {data_konser_item['tahun']}"
        tabel.add_row([konser_id, data_konser_item["gs"], jadwal, f"Rp.{data_konser_item['harga']}"])
    print(tabel)
```

Gambar 3.21 Implementasi: shorting mengurutkan data konser

3.2 Alur Program

Berikut ini adalah alur program bagian input dengan deskripsi sebagai berikut:

1. Menu Utama

```
#pilih role
def pilih_role():
    while True:
        try:
            print(f"\n{"="*5} SILAHKAN PILIH ROLE {"="*5}")
            print("[1]. Admin")
            print("[2]. Pengguna")
            print("[3]. Pengunjung ")
            print("[4]. Keluar")
            role = input("Masukkan pilihan = ")
            if role == "1" :
                login_admin()
            elif role == "2":
                pengguna()
            elif role == "3":
                pengunjung()
            elif role == "4":
                print("\n=== ***** SELAMAT TINGGAL DAN TERIMA KASIH ***** ===")
                exit()
            else :
                print("Pilihan tidak tersedia.")
        except KeyboardInterrupt:
            print("\nJangan menekan CTRL dan C Secara bersamaan!")
```

Gambar 3.22 Alur Program: Menu Utama 1

Pada role menu utama ini terdapat 4 role pilihan yaitu:

- Login sebagai admin
- Login sebagai Pengguna
- Login sebagai Pengunjung
- Keluar

```

#login admin
def login_admin():
    while True :
        try:
            print(f"\n{"="*5} LOGIN ADMIN {"="*5}")
            nama = input("Username = ")
            pw = pwinput.pwinput("Password = ")
            if nama in data_admin and data_admin[nama]==pw:
                menu_admin()
            elif nama in data_admin and data_admin[nama]!=pw:
                print("Password salah!")
                while True :
                    try:
                        kl = input("apakah anda ingin kembali atau lanjut?(kembali/lanjut) = ").lower()

```

Gambar 3.23 Alur Program: Menu Utama 2

Pada saat memilih pilihan yang pertama maka akan langsung masuk ke portal menu login sebagai admin, maka user akan diminta untuk menginputkan username dan password dengan benar yang sudah di daftarkan sebelumnya untuk dapat lanjut mengakses ke menu admin selanjutnya yaitu terdapat pilihan Lihat Event Konser, Tambah Event Konser, Perbarui Event Konser, Hapus Event konser, dan Logout.

```

#login akun pengguna
def login_pengguna():
    data_pengguna = memuat_data(json_path)
    while True :
        try:
            print(f"\n{"="*5} LOGIN PENGGUNA {"="*5}")
            nama = input("Username = ")
            pw = pwinput.pwinput("Password = ")
            if nama in data_pengguna and data_pengguna[nama][["password"]==pw:
                menu_pengguna(nama)
            elif nama in data_pengguna and data_pengguna[nama][["password"]]!=pw:
                print("Password Salah!")
                kembalilanjut()
            else :
                print("Nama pengguna tidak tersedia.")
                kembalilanjut()
        except KeyboardInterrupt:
            print("\nJangan menekan CTRL dan C secara bersamaan!")

```

Gambar 3.24 Alur Program: Menu Utama Login pengguna 3

Pada saat memilih login sebagai pengguna di pilihan pertama, maka akan langsung masuk ke portal menu pengguna dan otomatis langsung diberi tiga pilihan untuk melakukan “Login” jika sudah memiliki username dan password sebelumnya, “Daftar” terlebih dahulu untuk yang belum memiliki username dan password dan jika berhasil untuk daftar maka akan di konfirmasi apakah akun sudah tersedia atau tidak di menu login, dan terakhir menu untuk “ kembali”.

```

#pengunjung
def pengunjung():
    data_konser = memuat_data(json_path3)
    print(f"\n{' '*21}  MODE PENGUNJUNG {' '*21}")
    print(f"\n{' '*21}  LIHAT KONSER {' '*21}")
    print(tampilkan_tabel(data_konser))
    while True:
        try:
            kl = input("apakah anda ingin kembali atau login sebagai pengguna?(Kembali/login) = ").lower()
            if kl == "kembali" :
                pilih_role()
            elif kl == "login":
                pengguna()
            else :
                print("Input tidak sesuai!")
        except KeyboardInterrupt:
            print("\nJangan menekan CTRL dan C secara bersamaan!")

```

Gambar 3.25 Alur Program: Menu Utama Pengunjung 4

Pada saat memilih sebagai pengunjung di pilihan pertama, maka akan langsung masuk ke portal menu pengunjung yang menampilkan List konser yang berisi Guest Star, Jadwal hingga Harga dan setelah melihat data tersebut pengunjung ditanya ingin kembali atau login sebagai pengguna.

2. Menu Admin

```

#menu admin
def menu_admin():
    while True :
        try:
            print(f"\n{' '*5} MENU ADMIN {' '*5}")
            print("[1]. Lihat Event Konser")
            print("[2]. Tambah Event Konser")
            print("[3]. Perbarui Event Konser")
            print("[4]. Hapus Event Konser")
            print("[5]. Logout")
            pilih = input("Masukkan pilihan = ")
            if pilih == "1" :
                lihat_konser()
            elif pilih == "2":
                tambah_konser()
            elif pilih == "3":
                perbarui_konser()
            elif pilih == "4":
                hapus_konser()
            elif pilih == "5":
                pilih_role()
            else :
                print("Pilihan tidak tersedia.")

```

Gambar 3.22 Alur Program: Menu Utama Admin

Di menu utama admin terdapat lima pilihan untuk admin yaitu: ketika admin ingin memilih pilihan pertama maka akan masuk ke menu lihat konser, Jika admin memilih pilihan kedua maka akan masuk ke menu tambah event konser, Jika admin ingin memperbarui event konser maka harus memilih pilihan ketiga, selanjutnya ketika ingin menghapus event

konser maka admin harus memilih pilihan ke-empat dan terakhir admin bisa memilih untuk logout di pilihan kelima maka otomatis program akan kembali ke menu admin.

3. Menu Pengguna

```
#menu pengguna
def menu_pengguna(nama):
    data_pengguna = memuat_data(json_path)
    while True :
        try:
            print(f"\n{"="*5} MENU PENGGUNA {"="*5}")
            print("[1]. Lihat Event Konser")
            print("[2]. Beli Tiket Event Konser")
            print("[3]. Lihat Saldo E-Money")
            print("[4]. Lihat Tiket yang dibeli")
            print("[5]. Logout")
            pilih = input("Masukkan pilihan = ")
            if pilih == "1" :
                lihat_konserp(nama)
            elif pilih == "2":
                if data_pengguna[nama]["tiket"] != "kosong":
                    print("\nMaaf anda sudah memiliki tiket")
                    print("Harap hapus tiket anda terlebih dahulu")
                    menu_pengguna(nama)
                else:
                    beli_tiket(nama)
            elif pilih == "3":
                cek_saldo(nama)
            elif pilih == "4":
                invoice(nama)
            elif pilih == "5":
                pengguna()
            else :
                print("Pilihan Tidak Tersedia.")
        except KeyboardInterrupt:
            print("\nJangan menekan CTRL dan C secara bersamaan!")
```

Gambar 3.23 Tampilan Program: Tampilan Menu Utama Pengguna

Di menu utama pengguna terdapat lima pilihan untuk pengguna yaitu: ketika pengguna ingin memilih pilihan pertama maka akan masuk ke portal menu lihat event konser disitu pengguna akan ditampilkan list dari data konser, jika pengguna memilih pilihan kedua maka akan masuk ke menu beli tiket konser disana pengguna akan disuruh untuk menginput nomor konser yang dipilih, jika pengguna ingin cek saldo maka harus memilih pilihan ketiga, selanjutnya ketika ingin melihat tiket yang sudah dibeli harus memilih pilihan ke-empat dan terakhir jika sudah pengguna bisa memilih untuk logout di pilihan kelima maka otomatis program akan kembali ke menu admin .

3.3 Source Code

```
from prettytable import PrettyTable
import json
import pwinput

#memuat dan menyimpan akun pengguna
json_path = r"D:\PROJECT AKHIR DDP\db pengguna.json"
```

```

def simpan_data(json_path3, data_konser):
    with open(json_path3, "w") as databasek:
        json.dump(data_konser, databasek, indent=4)

#prettytable
def tampilkan_tabel(data_konser):
    tabel = PrettyTable()
    tabel.field_names = ["ID", "Guest Star", "Jadwal", "Harga"]
    for konser_id, data_konser_item in data_konser["konser"].items():
        jadwal = f"{data_konser_item['tanggal']} {data_konser_item['bulan']} {data_konser_item['tahun']}"
        tabel.add_row([konser_id, data_konser_item["gs"], jadwal, f"Rp.{data_konser_item['harga']}"])
    return tabel

#untuk shorting data konsernya
def update_table(data_konser, kriteria):
    tabel = PrettyTable()
    tabel.field_names = ["ID", "Guest Star", "Jadwal", "Harga"]
    if kriteria == "harga1":
        sorted_konser = sorted(data_konser["konser"].items(), key=lambda x: int(x[1]["harga"]))
    elif kriteria == "harga2": sorted_konser = sorted(data_konser["konser"].items(), key=lambda x: int(x[1]["harga"]), reverse=True)
    else:
        raise ValueError("Kriteria tidak dikenal")
    for konser_id, data_konser_item in sorted_konser:
        jadwal = f"{data_konser_item['tanggal']} {data_konser_item['bulan']} {data_konser_item['tahun']}"
        tabel.add_row([konser_id, data_konser_item["gs"], jadwal, f"Rp.{data_konser_item['harga']}"])
    print(tabel)

```



```

#pilih role
def pilih_role():
    while True:
        try:
            print(f"\n{ "="*5 } SILAHKAN PILIH ROLE { "="*5 }")
            print("[1]. Admin")
            print("[2]. Pengguna")
            print("[3]. Pengunjung ")
            print("[4]. Keluar")
            role = input("Masukkan pilihan = ")
            if role == "1" :
                login_admin()
            elif role == "2":
                pengguna()
            elif role == "3":
                pengunjung()
            elif role == "4":
                print("\n===== ***** SELAMAT TINGGAL DAN TERIMA KASIH *****
=====")
                exit()
            else :
                print("Pilihan tidak tersedia.")
        except KeyboardInterrupt:
            print("\nJangan menekan CTRL dan C Secara bersamaan!")

#login admin
def login_admin():
    while True :
        try:
            print(f"\n{ "="*5 } LOGIN ADMIN { "="*5 }")
            nama = input("Username = ")
            pw = pwinput.pwinput("Password = ")

```

```

if nama in data_admin and data_admin[nama]==pw:
    menu_admin()
elif nama in data_admin and data_admin[nama]!=pw:
    print("Password salah!")
    while True :
        try:
            kl = input("apakah anda ingin kembali atau lanjut?(kembali/lanjut =
").lower()
            if kl == "kembali":
                pilih_role()
            elif kl == "lanjut":
                break
            else :
                print("Input tidak sesuai!")
        except KeyboardInterrupt:
            print("\nJangan menekan CTRL dan C secara bersamaan!")
    else :
        print("Nama pengguna tidak tersedia.")
        while True :
            try:
                kl = input("apakah anda ingin kembali atau lanjut?(kembali/lanjut =
").lower()
                if kl == "kembali":
                    pilih_role()
                elif kl == "lanjut":
                    break
                else :
                    print("Input tidak sesuai!")
            except KeyboardInterrupt:
                print("\nJangan menekan CTRL dan C secara bersamaan!")
        except KeyboardInterrupt:

```

```

except KeyboardInterrupt:
    print("\nJangan menekan CTRL dan C secara bersamaan!")
else :
    print("Nama pengguna tidak tersedia.")
    while True :
        try:
            kl = input("apakah anda ingin kembali atau lanjut?(kembali/lanjut = ").lower()
            if kl == "kembali":
                pilih_role()
            elif kl == "lanjut":
                break
            else :
                print("Input tidak sesuai!")
        except KeyboardInterrupt:
            print("\nJangan menekan CTRL dan C secara bersamaan!")
except KeyboardInterrupt:
    print("\nJangan menekan CTRL dan C secara bersamaan!")

#menu admin
def menu_admin():
    while True :
        try:
            print(f"\n{ "="*5} MENU ADMIN { "="*5} ")
            print("[1]. Lihat Event Konser")
            print("[2]. Tambah Event Konser")
            print("[3]. Perbarui Event Konser")
            print("[4]. Hapus Event Konser")
            print("[5]. Logout")
            pilih = input("Masukkan pilihan = ")
            if pilih == "1" :
                lihat_konser()

```

```

elif pilih == "2":
    tambah_konser()
elif pilih == "3":
    perbarui_konser()
elif pilih == "4":
    hapus_konser()
elif pilih == "5":
    pilih_role()
else :
    print("Pilihan tidak tersedia.")
except KeyboardInterrupt:
    print("\nJangan menekan CTRL dan C secara bersamaan!")

```

#lihat konser admin

```

def lihat_konser():
    print(f"\n{'='*21} LIST KONSER {'='*21}")
    data_konser = memuat_data(json_path3)
    print(tampilkan_tabel(data_konser))
    while True:
        try:
            print(f"\n{'='*5} PILIH KRITERIA SHORTING {'='*5}")
            print("[1]. Harga terendah ke tertinggi")
            print("[2]. Harga tetinggi ke terendah")
            print("[3]. Kembali")
            pilihan = input("Masukkan nomor pilihan Anda = ").strip()
            if pilihan == "1":
                kriteria = "harga1"
                update_table(data_konser, kriteria)
            elif pilihan == "2":
                kriteria = "harga2"
                update_table(data_konser, kriteria)
            elif pilihan == "3":
                menu_admin()

```

```

else:
    print("Pilihan tidak tersedia.")
except KeyboardInterrupt:
    print("\nJangan menekan CTRL dan C Secara bersamaan!")

#Menambah konser
def tambah_konser():
    data_konser = memuat_data(json_path3)
    print(f"\n{ "*"21 } MENAMBAH KONSER { "*"21 }")
    print(tampilkan_tabel(data_konser))
    while True:
        try:
            no = (input("masukan angka untuk ID (Contoh:002) = "))
            if no.isdigit() and 3<=len(no)<4 :
                no = no.zfill(3)
                id_konser = f"DRS{no}"
                if id_konser in data_konser["konser"]:
                    print("ID sudah digunakan.")
                else :
                    break
            else:
                print("Harus memasukan 3 angka")
        except ValueError:
            print("Input harus angka.")
        except KeyboardInterrupt:
            print("\nJangan menekan CTRL dan C secara bersamaan!")
    while True :
        try:
            gs = input("Masukkan Guest Star = ").strip().lower().capitalize()
            if 4 <= len(gs) <= 15:
                if not gs:

```

```

        raise ValueError("Nama Guest Star tidak boleh kosong.")
    else :
        break
else:
    print("Nama harus antara 4 hingga 15 karakter.")
except ValueError as e:
    print(e)
except KeyboardInterrupt:
    print("\nJangan menekan CTRL dan C secara bersamaan!")
while True :
    try:
        tanggal = int(input("Masukkan tanggal (1-31) = "))
        if 0<tanggal<32:
            tanggal = str(tanggal)
            break
        else :
            print("Tanggal tidak sesuai.")
    except ValueError:
        print("Input harus angka.")
    except KeyboardInterrupt:
        print("\nJangan menekan CTRL dan C secara bersamaan!")
while True :
    bulan = input("Masukkan bulan (Januari-Desember) = ").lower().capitalize()
    if bulan not in data_konser["bulan"]:
        print("Bulan tidak tersedia.")
    else :
        break
while True :
    try:
        tahun = int(input("Masukkan tahun (2024-2030) = "))
        if 2024<=tahun<2031:
            tahun = str(tahun)

```

```

        break

    else :

        print("Tahun tidak sesuai.")
except ValueError:
    print("Input harus angka.")
except KeyboardInterrupt:
    print("\nJangan menekan CTRL dan C secara bersamaan!")
while True :
    try:
        harga = int(input("Masukkan harga (minimal 50rb dan maksimal 1jt) = Rp. "))
        if harga<50000:
            print("Harga terlalu rendah.")
        elif harga>1000000:
            print("Harga terlalu tinggi.")
        else :
            data_konser["konser"][id_konser] = {"gs": gs, "tanggal": tanggal, "bulan": bulan,
"tahun": tahun, "harga":harga}

            simpan_data(json_path3, data_konser)

            print(f'Konser baru berhasil ditambahkan dengan ID {no}.')

            print(f"\n{ "="*21 } LIST KONSER { "="*21 }")

            print(tampilkan_tabel(data_konser))

            break

    except ValueError:
        print("Harap masukkan angka.")
    except KeyboardInterrupt:
        print("\nJangan menekan CTRL dan C secara bersamaan!")
while True :
    try:
        yt = input("Ingin menambah konser lagi(Ya/Tidak)? = ").lower()

        if yt == "ya":
            tambah_konser()

```

```

elif yt == "tidak" :

    menu_admin()

except KeyboardInterrupt:

    print("\nJangan menekan CTRL dan C secara bersamaan!")

#Perbarui Konser

def perbarui_konser():

    data_konser = memuat_data(json_path3)

    print(f"\n{ "="*21 } MEMPERBARUI KONSER { "="*21 }")

    print(tampilkan_tabel(data_konser))

    while True :

        try :

            id_konser = input("Masukkan ID konser yang ingin di ubah = ")

            if id_konser in data_konser["konser"]:

                while True:

                    try:

                        gs = input("Masukkan Guest Star = ").strip().lower().capitalize()

                        if 4 <= len(gs) <= 15:

                            if not gs:

                                raise ValueError("Nama Guest Star tidak boleh kosong.")

                            else :

                                break

                        else:

                            print("Nama harus antara 4 hingga 15 karakter.")

                    except KeyboardInterrupt:

                        print("\nJangan menekan CTRL dan C secara bersamaan!")

                while True :

                    try:

                        tanggal = (int(input("Masukkan tanggal(1-31) = ")))

                        if 0<tanggal<32:

                            tanggal = str(tanggal)

                            break

```



```

        else :
            print("tanggal tidak sesuai.")
    except ValueError:
        print("Harap masukkan angka.")
    except KeyboardInterrupt:
        print("\nJangan menekan CTRL dan C secara bersamaan!")
while True :
    bulan = input("Masukkan bulan (Januari-Desember) = ").lower().capitalize()
    if bulan not in data_konser["bulan"]:
        print("Bulan tidak tersedia.")
    else :
        break
while True :
    try:
        tahun = int(input ("Masukkan tahun(2024-2030) = "))
        if 2024<=tahun<2031:
            tahun = str(tahun)
            break
        else :
            print("Tahun tidak sesuai.")
    except ValueError:
        print("Harap masukkan angka.")
    except KeyboardInterrupt:
        print("\nJangan menekan CTRL dan C secara bersamaan!")
while True :
    try:
        harga = int(input("Masukkan Harga (minimal 50rb dan maksimal 1jt) = "))
        if harga<50000:
            print("Harga terlalu rendah.")
        elif harga>1000000:
            print("Harga terlalu tinggi.")

```

```

        else :
            data_konser["konser"][id_konser] = {"gs": gs, "tanggal": tanggal, "bulan": bulan,
            "tahun": tahun, "harga": harga}
            simpan_data(json_path3, data_konser)
            print(f"Konser dengan ID {id_konser} berhasil diperbarui.")
            print(f"\n{ "="*21 } LIST KONSER { "="*21 }")
            print(tampilkan_tabel(data_konser))
            while True :
                yt = input("ingin memperbarui konser lagi(Ya/Tidak)? = ").lower()
                if yt == "ya":
                    perbarui_konser()
                elif yt == "tidak":
                    menu_admin()
            except ValueError:
                print("Harap masukkan angka.")
            except KeyboardInterrupt:
                print("\nJangan menekan CTRL dan C secara bersamaan!")
    else :
        print("Nomor konser tidak ditemukan.")
    except KeyboardInterrupt:
        print("\nJangan menekan CTRL dan C secara bersamaan!")

#hapus konser
def hapus_konser():
    data_konser = memuat_data(json_path3)
    print(f"\n{ "="*21 } HAPUS KONSER { "="*21 }")
    print(tampilkan_tabel(data_konser))
    while True :
        try:
            id_konser = input("Masukkan nomor konser yang ingin dihapus = ")
            if id_konser in data_konser["konser"]:
                del data_konser["konser"][id_konser]

```

```

simpan_data(json_path3, data_konser)

print(f"Konser dengan ID {id_konser} berhasil dihapus.")

print(f"\n{"="*21} LIST KONSER {"="*21}")

print(tampilkan_tabel(data_konser))

while True :

    yt = input("Ingin menghapus konser lagi(Ya/Tidak)? = ").lower()

    if yt == "ya":

        hapus_konser()

    elif yt == "tidak":

        menu_admin()

    else:

        print("Nomor konser tidak ditemukan.")

except KeyboardInterrupt:

    print("\nJangan menekan CTRL dan C secara bersamaan!")

```

#pengguna

```

def pengguna():

    while True:

        try:

            print(f"\n{"="*5} PENGGUNA {"="*5}")

            print("[1]. Login")

            print("[2]. Daftar")

            print("[3]. Kembali")

            pilih = input("Masukkan pilihan = ")

            if pilih == "1" :

                login_pengguna()

            elif pilih == "2":

                daftar_akun()

            elif pilih == "3":

                pilih_role()

            else :

```

```

        print("Pilihan tidak tersedia.")

except KeyboardInterrupt:

    print("\nJangan menekan CTRL dan C secara bersamaan!")


#login akun pengguna
def login_pengguna():
    data_pengguna = memuat_data(json_path)
    while True :
        try:
            print(f"\n{ "="*5} LOGIN PENGGUNA { "="*5}")
            nama = input("Username = ")
            pw = pwinput.pwinput("Password = ")
            if nama in data_pengguna and data_pengguna[nama]["password"]==pw:
                menu_pengguna(nama)
            elif nama in data_pengguna and data_pengguna[nama]["password"]!=pw:
                print("Password Salah!")
                kembalilanjut()
            else :
                print("Nama pengguna tidak tersedia.")
                kembalilanjut()
        except KeyboardInterrupt:
            print("\nJangan menekan CTRL dan C secara bersamaan!")


#daftar akun pengguna
def daftar_akun():
    data_pengguna = memuat_data(json_path)
    while True:
        print(f"\n{ "="*5} DAFTAR AKUN { "="*5}")
        try:
            print("\nMasukan Username:")
            nama = input("Username (4-10 karakter) = ").strip()
            if 4 <= len(nama) <= 15:

```

```

if not nama:
    raise ValueError("Nama tidak boleh kosong.")
elif nama in data_pengguna:
    print("Username sudah digunakan.")
else :
    while True:
        try:
            print("\nMasukan Password anda:")
            pw = input("Password =").strip()
            if 4 <= len(pw):
                if not pw:
                    raise ValueError("Password tidak boleh kosong.")
                else :
                    data_pengguna[nama]={ "password":pw, "saldo":0,"tiket":"kosong"}
                    simpan_data(json_path, data_pengguna)
                    print("Akun berhasil dibuat.")
                    pengguna()
            else:
                print("Password minimal 4 karakter.")
        except ValueError as e:
            print(e)
        except KeyboardInterrupt:
            print("\nJangan menekan CTRL dan C secara bersamaan!")
    else:
        print("Nama harus antara 4 hingga 15 karakter.")
except ValueError as e:
    print(e)
except KeyboardInterrupt:
    print("\nJangan menekan CTRL dan C secara bersamaan!")

#menu pengguna
def menu_pengguna(nama):
    data_pengguna = memuat_data(json_path)
    while True :

```

```

try:

    print(f"\n{ '='*5 } MENU PENGGUNA { '='*5 }")

    print("[1]. Lihat Event Konser")

    print("[2]. Beli Tiket Event Konser")

    print("[3]. Lihat Saldo E-Money")

    print("[4]. Lihat Tiket yang dibeli")

    print("[5]. Logout")

    pilih = input("Masukkan pilihan = ")

    if pilih == "1" :

        lihat_konserp(nama)

    elif pilih == "2":

        if data_pengguna[nama]["tiket"] != "kosong":

            print("\nMaaf anda sudah memiliki tiket")

            print("Harap hapus tiket anda terlebih dahulu")

            menu_pengguna(nama)

        else:

            beli_tiket(nama)

    elif pilih == "3":

        cek_saldo(nama)

    elif pilih == "4":

        invoice(nama)

    elif pilih == "5":

        pengguna()

    else :

        print("Pilihan Tidak Tersedia.")

except KeyboardInterrupt:

    print("\nJangan menekan CTRL dan C secara bersamaan!")

```

```

def lihat_konserp(nama):

    data_konser = memuat_data(json_path3)

    print(f"\n{ '='*21 } LIST KONSER { '='*21 }")

    print(tampilkan_tabel(data_konser))

```

```

while True:
    try:
        print(f"\n{ "="*5 } PILIH KRITERIA SHORTING { "="*5 }")
        print("[1]. Harga terendah ke tertinggi")
        print("[2]. Harga tetinggi ke terendah")
        print("[3]. Kembali")
        pilihan = input("Masukkan nomor pilihan Anda = ").strip()
        if pilihan == "1":
            kriteria = "harga1"
            update_table(data_konser, kriteria)
        elif pilihan == "2":
            kriteria = "harga2"
            update_table(data_konser, kriteria)
        elif pilihan == "3":
            menu_pengguna(nama)
        else:
            print("Pilihan tidak tersedia.")
    except KeyboardInterrupt:
        print("\nJangan menekan CTRL dan C Secara bersamaan!")

#menu tiket
def beli_tiket(nama):
    data_konser = memuat_data(json_path3)
    data_pengguna = memuat_data(json_path)
    while True:
        try:
            print(f"\n{ "="*5 } BELI TIKET { "="*5 }")
            id_konser = input("Masukkan nomor konser yang ingin dipilih = ")
            if id_konser in data_konser["konser"]:
                while True :
                    try:

```

```

        banyak = int(input("Jumlah tiket yang ingin dibeli = "))
        harga_tiket = data_konser["konser"][id_konser]["harga"]*banyak
        print("\nKeterangan :")
        print(f"Guest Star = {data_konser["konser"][id_konser]["gs"]}")
        print(f"Rp.{data_konser["konser"][id_konser]["harga"]} X {banyak} =
Rp.{harga_tiket}")
        break
    except ValueError:
        print("Harap masukkan angka.")
    except KeyboardInterrupt:
        print("\nJangan menekan CTRL dan C secara bersamaan!")
while True :
    try:
        bayar = input("ingin lanjutkan pembayaran atau kembali ke menu(ya/kembali)? =
").strip().lower()
        if bayar == "ya":
            if data_pengguna[nama]["saldo"] >= harga_tiket:
                data_pengguna[nama]["saldo"] -= harga_tiket
                data_pengguna[nama]["tiket"] =
{"gs":data_konser["konser"][id_konser]["gs"],
"harga":data_konser["konser"][id_konser]["harga"], "jumlah":banyak, "total":harga_tiket}
                simpan_data(json_path, data_pengguna)
                print(f"\n{"="*5} PEMBELIAN BERHASIL {"="*5}")
                print(f>Nama : {nama} ")
                print(f"Guest Star : {data_konser["konser"][id_konser]["gs"]}")
                print(f"Harga :Rp{data_konser["konser"][id_konser]["harga"]} x {banyak}")
                print(f>Total :Rp.{harga_tiket}")
                print("="*30)
                menu_pengguna(nama)
            else:
                print("\nsaldo tidak cukup, silahkan melakukan pengisian.")
                print("Proses pembayaran dibatalkan.")
                menu_pengguna(nama)
    
```



```

        elif bayar == "kembali":
            print("Proses pembayaran dibatalkan.")
            menu_pengguna(nama)
        else :
            print("Input tidak sesuai.")
    except KeyboardInterrupt:
        print("\nJangan menekan CTRL dan C secara bersamaan!")
    else: print("Nomor konser tidak valid.")
except KeyboardInterrupt:
    print("\nJangan menekan CTRL dan C secara bersamaan!")

#Cek saldo
def cek_saldo(nama):
    data_pengguna = memuat_data(json_path)
    while True:
        try:
            print(f"\n{ "="*5 } E-MONEY { "="*5 }")
            print(f"Saldo Anda = Rp.{data_pengguna[nama]["saldo"]}")
            print("[1]. Top Up")
            print("[2]. Kembali")
            pilih = input("Masukkan pilihan = ")
            if pilih == "1" :
                topup(nama)
            elif pilih == "2":
                menu_pengguna(nama)
            else :
                print("Pilihan tidak tersedia!")
        except KeyboardInterrupt:
            print("\nJangan menekan CTRL dan C secara bersamaan!")

#top up E-Money
def topup(nama):

```



```

        elif tambah > 2000000:
            print("nominal terlalu besar.")
        else :
            data_pengguna[nama]["saldo"] = data_pengguna[nama]["saldo"] +
int(tambah)
            simpan_data(json_path, data_pengguna)
            print(f"\n{"="*5} TOP UP BERHASIL {"="*5}")
            print(f>Nama : {nama} ")
            print(f>Nominal :Rp.{tambah}")
            print("="*27)
            cek_saldo(nama)
        except ValueError:
            print("nominal tidak valid.")
        except KeyboardInterrupt:
            print("\nJangan menekan CTRL dan C secara bersamaan!")
    elif pilih == "4":
        cek_saldo(nama)
    else :
        print("Pilihan Tidak Tersedia.")
except KeyboardInterrupt:
    print("\nJangan menekan CTRL dan C secara bersamaan!")

def invoice(nama):
    data_pengguna = memuat_data(json_path)
    if data_pengguna[nama]["tiket"] == "kosong":
        print("\nAnda belum mempunyai tiket.")
        menu_pengguna(nama)
    else :
        data_pengguna = memuat_data(json_path)
        print(f"\n{"="*10} INVOICE {"="*10}")
        print(f>Nama : {nama} ")
        print(f"Guest Star : {data_pengguna[nama]["tiket"]["gs"]}")

```

```

print(f"Harga :Rp{data_pengguna[nama][\"tiket\"][\"harga\"]} x
{data_pengguna[nama][\"tiket\"][\"jumlah\"]}")

print(f"Total :Rp.{data_pengguna[nama][\"tiket\"][\"total\"]}")

print("="*29)

while True:

    try:

        hapus = input("apakah anda ingin menghapus tiket anda (ya/tidak) = ").lower()

        if hapus == "ya":

            data_pengguna[nama][\"tiket\"] = "kosong"

            simpan_data(json_path, data_pengguna)

            menu_pengguna(nama)

        elif hapus == "tidak":

            menu_pengguna(nama)

        else :

            print("Input tidak sesuai.")

    except KeyboardInterrupt:

        print("\nJangan menekan CTRL dan C secara bersamaan!")

```

#pengunjung

```

def pengunjung():

    data_konser = memuat_data(json_path3)

    print(f"\n{' '*21} MODE PENGUNJUNG {' '*21}")

    print(f"\n{' '*21} LIHAT KONSER {' '*21}")

    print(tampilkan_tabel(data_konser))

    while True:

        try:

            kl = input("apakah anda ingin kembali atau login sebagai pengguna?(Kembali/login) = ").lower()

            if kl == "kembali" :

                pilih_role()

            elif kl == "login":

                pengguna()

            else :

```

```

        print("Input tidak sesuai!")

except KeyboardInterrupt:

    print("\nJangan menekan CTRL dan C secara bersamaan!")

#Kembali atau lanjut
def kembalilanjut():
    while True :
        try:
            kl = input("apakah anda ingin kembali atau lanjut?(Kembali/lanjut) = ").lower()
            if kl == "kembali":
                pengguna()
            elif kl == "lanjut":
                break
            else :
                print("Input tidak sesuai!")
        except KeyboardInterrupt:
            print("\nJangan menekan CTRL dan C secara bersamaan!")

#tampilan awal program
def main():
    print("\n===== ***** SELAMAT DATANG DI CTICK.COM ***** =====")
    pilih_role()

#memulai program
main()

```

D. Tampilan Program

```
{ } db_admin.json > ...  
1 {  
2   "Rina": "6003",  
3   "Dwi": "6012",  
4   "Sadikin": "6031"  
5 }
```

Gambar 3.24 Tampilan Program: Database admin

```
} db_pengguna.json > ...  
1 {  
2   "Nabil": {  
3     "password": "aslabA1",  
4     "saldo": 10000,  
5     "tiket": "kosong"  
6   },  
7   "Rizky": {  
8     "password": "aslabA2",  
9     "saldo": 20000,  
10    "tiket": "kosong"  
11  },  
12  "Dwi pebri": {  
13    "password": "1234",  
14    "saldo": 0,  
15    "tiket": "kosong"  
16  }  
17 }
```

Gambar 3.25 Tampilan Program: Database Pengguna

```

db_konser.json > {} konser > {} DRS004
1  {
2    "konser": {
3      "DRS002": {
4        "gs": "Sal Priadi",
5        "tanggal": "23",
6        "bulan": "April",
7        "tahun": "2025",
8        "harga": 160000
9      },
10     "DRS003": {
11       "gs": "Reality Club",
12       "tanggal": "23",
13       "bulan": "Desember",
14       "tahun": "2025",
15       "harga": 800000
16     },
17     "DRS004": {
18       "gs": "Bernadya",
19       "tanggal": "5",
20       "bulan": "Maret",
21       "tahun": "2025",
22       "harga": 300000
23     },
24     "DRS006": {
25       "gs": "Hindia",
26       "tanggal": "21",
27       "bulan": "Februari",
28       "tahun": "2026",
29       "harga": 300000
30     },

```

Gambar 3.26 Tampilan Program: Database konser 1

```

30   },
31   "DRS007": {
32     "gs": "Feast.",
33     "tanggal": "31",
34     "bulan": "Oktober",
35     "tahun": "2025",
36     "harga": 400000
37   },
38   "DRS008": {
39     "gs": "Ardhito Pramono",
40     "tanggal": "21",
41     "bulan": "Februari",
42     "tahun": "2027",
43     "harga": 600000
44   },
45   "DRS009": {
46     "gs": "Juicy Lucy",
47     "tanggal": "7",
48     "bulan": "November",
49     "tahun": "2024",
50     "harga": 300000
51   },

```

Gambar 3.27 Tampilan Program: Database konser 2

```

52     "DRS010": {
53         "gs": "Boy pablo",
54         "tanggal": "30",
55         "bulan": "Januari",
56         "tahun": "2025",
57         "harga": 700000
58     },
59     "DRS012": {
60         "gs": "Riski febian",
61         "tanggal": "23",
62         "bulan": "Agustus",
63         "tahun": "2026",
64         "harga": 600000
65     },
66     "DRS001": {
67         "gs": "Isrina",
68         "tanggal": "6",
69         "bulan": "April",
70         "tahun": "2025",
71         "harga": 600000
72     }

```

Gambar 3.28 Tampilan Program: Database konser 3

```

74     "bulan": [
75         "Januari",
76         "Februari",
77         "Maret",
78         "April",
79         "Mei",
80         "Juni",
81         "Juli",
82         "Agustus",
83         "September",
84         "Oktober",
85         "November",
86         "Desember"
87     ]
88 }

```

Gambar 3.29 Tampilan Program: Database bulan konser


```
===== ***** SELAMAT DATANG DI CTICK.COM ***** =====  
  
===== SILAHKAN PILIH ROLE =====  
[1]. Admin  
[2]. Pengguna  
[3]. Pengunjung  
[4]. Keluar  
Masukkan pilihan =
```

Gambar 3.30 Tampilan Program: Tampilan Menu Awal

Pada tampilan awal disini terdapat 4 opsi seperti yang ada pada gambar diatas. Opsi pertama ditujukan untuk login admin, opsi kedua ditujukan kepada pengguna yang telah memiliki akun dan belum memiliki akun, opsi ketiga ditujukan kepada pengunjung dan opsi terakhir apabila user ingin keluar dari program.

```
===== SILAHKAN PILIH ROLE =====  
[1]. Admin  
[2]. Pengguna  
[3]. Pengunjung  
[4]. Keluar  
Masukkan pilihan = 1  
  
===== LOGIN ADMIN =====  
Username = Rina  
Password = *****
```

Gambar 3.31 Tampilan Program: Tampilan Login Admin

Untuk login sebagai admin, diminta untuk memasukkan username dan juga password yang sesuai dengan yang telah ditentukan sebelumnya.

```
===== MENU ADMIN =====  
[1]. Lihat Event Konser  
[2]. Tambah Event Konser  
[3]. Perbarui Event Konser  
[4]. Hapus Event Konser  
[5]. Logout  
Masukkan pilihan = █
```

Gambar 3.32 Tampilan Program: Tampilan Menu Admin

Setelah berhasil menginput data sebagai admin sesuai, maka akan langsung masuk ke dalam Menu Admin yang di dalamnya terdapat 5 menu dan user diminta untuk memasukkan pilihan sesuai yang diinginkan.

Masukkan pilihan = 1

```
===== LIST KONSER =====
```

ID	Guest Star	Jadwal	Harga
DRS002	Sal Priadi	23 April 2025	Rp.160000
DRS003	Reality Club	23 Desember 2025	Rp.800000
DRS004	Bernadya	5 Maret 2025	Rp.300000
DRS006	Hindia	21 Februari 2026	Rp.300000
DRS007	Feast.	31 Oktober 2025	Rp.400000
DRS008	Ardhito Pramono	21 Februari 2027	Rp.600000
DRS009	Juicy Lucy	7 November 2024	Rp.300000
DRS010	Boy pablo	30 Januari 2025	Rp.700000
DRS012	Riski febian	23 Agustus 2026	Rp.600000
DRS001	Isrina	6 April 2025	Rp.600000

Gambar 3.33 Tampilan Program: Tampilan Menu Admin 1

Jika user menginputkan pilihan 1, maka akan menampilkan List konser yang tertera mulai dari Guest Star, Jadwal konser hingga Harga tiket konser.

```
===== LIST KONSER =====
```

ID	Guest Star	Jadwal	Harga
DRS002	Sal Priadi	23 April 2025	Rp.160000
DRS003	Reality Club	23 Desember 2025	Rp.800000
DRS004	Bernadya	5 Maret 2025	Rp.300000
DRS006	Hindia	21 Februari 2026	Rp.300000
DRS007	Feast.	31 Oktober 2025	Rp.400000
DRS008	Ardhito Pramono	21 Februari 2027	Rp.600000
DRS009	Juicy Lucy	7 November 2024	Rp.300000
DRS010	Boy pablo	30 Januari 2025	Rp.700000
DRS012	Riski febian	23 Agustus 2026	Rp.600000
DRS001	Isrina	6 April 2025	Rp.600000

```
===== PILIH KRITERIA SHORTING =====
```

[1]. Harga terendah ke tertinggi
 [2]. Harga tetinggi ke terendah
 [3]. Kembali
 Masukkan nomor pilihan Anda =

Gambar 3.34 Tampilan Program: Tampilan Memilih Shorting

Ketika sudah melihat list konser, tetapi ingin mencari dari harga yang termurah maka bisa melakukan shorting untuk memudahkan. Sorting akan mengurutkan range harga dari murah ke mahal dan sebaliknya.

```

===== PILIH KRITERIA SHORTING =====
[1]. Harga terendah ke tertinggi
[2]. Harga tetinggi ke terendah
[3]. Kembali
Masukkan nomor pilihan Anda = 1
+-----+-----+-----+-----+
| ID | Guest Star | Jadwal | Harga |
+-----+-----+-----+-----+
| DRS002 | Sal Priadi | 23 April 2025 | Rp.160000 |
| DRS004 | Bernadya | 5 Maret 2025 | Rp.300000 |
| DRS006 | Hindia | 21 Februari 2026 | Rp.300000 |
| DRS009 | Juicy Lucy | 7 November 2024 | Rp.300000 |
| DRS007 | Feast. | 31 Oktober 2025 | Rp.400000 |
| DRS008 | Ardhito Pramono | 21 Februari 2027 | Rp.600000 |
| DRS012 | Riski febian | 23 Agustus 2026 | Rp.600000 |
| DRS001 | Isrina | 6 April 2025 | Rp.600000 |
| DRS010 | Boy pablo | 30 Januari 2025 | Rp.700000 |
| DRS003 | Reality Club | 23 Desember 2025 | Rp.800000 |
+-----+-----+-----+-----+

```

Gambar 3.35 Tampilan Program: Tampilan Shorting 1

Disini user memasukkan pilihan 1 untuk melakukan shorting dari harga paling murah ke mahal.

```

===== PILIH KRITERIA SHORTING =====
[1]. Harga terendah ke tertinggi
[2]. Harga tetinggi ke terendah
[3]. Kembali
Masukkan nomor pilihan Anda = 2
+-----+-----+-----+-----+
| ID | Guest Star | Jadwal | Harga |
+-----+-----+-----+-----+
| DRS003 | Reality Club | 23 Desember 2025 | Rp.800000 |
| DRS010 | Boy pablo | 30 Januari 2025 | Rp.700000 |
| DRS008 | Ardhito Pramono | 21 Februari 2027 | Rp.600000 |
| DRS012 | Riski febian | 23 Agustus 2026 | Rp.600000 |
| DRS001 | Isrina | 6 April 2025 | Rp.600000 |
| DRS007 | Feast. | 31 Oktober 2025 | Rp.400000 |
| DRS004 | Bernadya | 5 Maret 2025 | Rp.300000 |
| DRS006 | Hindia | 21 Februari 2026 | Rp.300000 |
| DRS009 | Juicy Lucy | 7 November 2024 | Rp.300000 |
| DRS002 | Sal Priadi | 23 April 2025 | Rp.160000 |
+-----+-----+-----+-----+

```

Gambar 3.36 Tampilan Program: Tampilan Shorting 2

Disini user memasukkan pilihan 2 untuk melakukan shorting dari harga paling mahal ke paling murah. Jika user memilih pilihan 3 maka akan Kembali ke menu admin.

```

===== MENAMBAH KONSER =====
+-----+-----+-----+-----+
| ID | Guest Star | Jadwal | Harga |
+-----+-----+-----+-----+
| DRS002 | Sal Priadi | 23 April 2025 | Rp.160000 |
| DRS003 | Reality Club | 23 Desember 2025 | Rp.800000 |
| DRS004 | Bernadya | 5 Maret 2025 | Rp.300000 |
| DRS006 | Hindia | 21 Februari 2026 | Rp.300000 |
| DRS007 | Feast. | 31 Oktober 2025 | Rp.400000 |
| DRS008 | Ardhito Pramono | 21 Februari 2027 | Rp.600000 |
| DRS009 | Juicy Lucy | 7 November 2024 | Rp.300000 |
| DRS010 | Boy pablo | 30 Januari 2025 | Rp.700000 |
| DRS012 | Riski febian | 23 Agustus 2026 | Rp.600000 |
| DRS001 | Isrina | 6 April 2025 | Rp.600000 |
+-----+-----+-----+-----+
masukan angka untuk ID (Contoh:002) = 015
Masukkan Guest Star = Mahalini
Masukkan tanggal (1-31) = 22
Masukkan bulan (Januari-Desember) = Maret
Masukkan tahun (2024-2030) = 2028
Masukkan harga (minimal 50rb dan maksimal 1jt) = Rp.500000
Konsert baru berhasil ditambahkan dengan ID 015.

```

Gambar 3.37 Tampilan Program: Tampilan Memasukkan Konser Baru

Disini user memilih pilihan 2 maka akan ditampilkan list konser dan diminta untuk memasukkan Id untuk menambah konser. Setelah memasukkan Id selanjutnya user menginputkan data penambahan konser sesuai keinginan. Dengan menginputkan Guest star, tanggal, bulan, tahun, hingga harga sesuai yang diinginkan.

```

===== LIST KONSER =====
+-----+-----+-----+-----+
| ID | Guest Star | Jadwal | Harga |
+-----+-----+-----+-----+
| DRS002 | Sal Priadi | 23 April 2025 | Rp.160000 |
| DRS003 | Reality Club | 23 Desember 2025 | Rp.800000 |
| DRS004 | Bernadya | 5 Maret 2025 | Rp.300000 |
| DRS006 | Hindia | 21 Februari 2026 | Rp.300000 |
| DRS007 | Feast. | 31 Oktober 2025 | Rp.400000 |
| DRS008 | Ardhito Pramono | 21 Februari 2027 | Rp.600000 |
| DRS009 | Juicy Lucy | 7 November 2024 | Rp.300000 |
| DRS010 | Boy pablo | 30 Januari 2025 | Rp.700000 |
| DRS012 | Riski febian | 23 Agustus 2026 | Rp.600000 |
| DRS001 | Isrina | 6 April 2025 | Rp.600000 |
| DRS015 | Mahalini | 22 Maret 2028 | Rp.500000 |
+-----+-----+-----+-----+
Ingin menambah konser lagi(Ya/Tidak)? = 

```

Gambar 3.38 Tampilan Program: Tampilan Menambahkan konser lagi

Disini penambahan data konser berhasil ditambahkan, setelahnya user diberi pilihan ingin tetap di program atau ingin keluar dari program. Jika memilih tidak dan ingin keluar program akan kembali ke menu admin.

```
===== MENU ADMIN =====
[1]. Lihat Event Konser
[2]. Tambah Event Konser
[3]. Perbarui Event Konser
[4]. Hapus Event Konser
[5]. Logout
Masukkan pilihan = 3

===== MEMPERBARUI KONSER =====
+-----+-----+-----+-----+
| ID | Guest Star | Jadwal | Harga |
+-----+-----+-----+-----+
| DRS002 | Sal Priadi | 23 April 2025 | Rp.160000 |
| DRS003 | Reality Club | 23 Desember 2025 | Rp.800000 |
| DRS004 | Bernadya | 5 Maret 2025 | Rp.300000 |
| DRS006 | Hindia | 21 Februari 2026 | Rp.300000 |
| DRS007 | Feast. | 31 Oktober 2025 | Rp.400000 |
| DRS008 | Ardhito Pramono | 21 Februari 2027 | Rp.600000 |
| DRS009 | Juicy Lucy | 7 November 2024 | Rp.300000 |
| DRS010 | Boy pablo | 30 Januari 2025 | Rp.700000 |
| DRS012 | Riski febian | 23 Agustus 2026 | Rp.600000 |
| DRS015 | Mahalini | 22 Maret 2028 | Rp.500000 |
+-----+-----+-----+-----+

Masukkan ID konser yang ingin di ubah = DRS012
Masukkan Guest Star = Dwi Peterpan
Masukkan tanggal(1-31) = 15
Masukkan bulan (Januari-Desember) = Juli
Masukkan tahun(2024-2030) = 2016
Tahun tidak sesuai.
Masukkan tahun(2024-2030) = 2025
Masukkan Harga (minimal 50rb dan maksimal 1jt) = 300000
Konser dengan ID DRS012 berhasil diperbarui.
```

Gambar 3.39 Tampilan Program: Tampilan Memperbarui konser

Disini user memilih pilihan 3 akan maka akan ditampilkan list konser dan diminta untuk memasukkan Id konser, Guest star, tanggal, bulan, tahun, hingga harga yang ingin diperbarui sesuai yang diinginkan.

```
===== LIST KONSER =====
+-----+-----+-----+-----+
| ID | Guest Star | Jadwal | Harga |
+-----+-----+-----+-----+
| DRS002 | Sal Priadi | 23 April 2025 | Rp.160000 |
| DRS003 | Reality Club | 23 Desember 2025 | Rp.800000 |
| DRS004 | Bernadya | 5 Maret 2025 | Rp.300000 |
| DRS006 | Hindia | 21 Februari 2026 | Rp.300000 |
| DRS007 | Feast. | 31 Oktober 2025 | Rp.400000 |
| DRS008 | Ardhito Pramono | 21 Februari 2027 | Rp.600000 |
| DRS009 | Juicy Lucy | 7 November 2024 | Rp.300000 |
| DRS010 | Boy pablo | 30 Januari 2025 | Rp.700000 |
| DRS012 | Dwi peterpan | 15 Juli 2025 | Rp.300000 |
| DRS015 | Mahalini | 22 Maret 2028 | Rp.500000 |
+-----+-----+-----+-----+

ingin memperbarui konser lagi(Ya/Tidak)? =
```

Gambar 3.40 Tampilan Program: Tampilan Berhasil Memperbarui konser

Setelah berhasil memperbarui konser, diberi pilihan ingin memperbarui lagi atau tidak. Jika ya maka akan melakukan input ulang dan jika tidak maka akan keluar dari program perbarui event konser.

```
===== MENU ADMIN =====
[1]. Lihat Event Konser
[2]. Tambah Event Konser
[3]. Perbarui Event Konser
[4]. Hapus Event Konser
[5]. Logout
Masukkan pilihan = 4

===== HAPUS KONSER =====
+-----+-----+-----+-----+
| ID | Guest Star | Jadwal | Harga |
+-----+-----+-----+-----+
| DRS002 | Sal Priadi | 23 April 2025 | Rp.160000 |
| DRS003 | Reality Club | 23 Desember 2025 | Rp.800000 |
| DRS004 | Bernadya | 5 Maret 2025 | Rp.300000 |
| DRS006 | Hindia | 21 Februari 2026 | Rp.300000 |
| DRS007 | Feast. | 31 Oktober 2025 | Rp.400000 |
| DRS008 | Ardhito Pramono | 21 Februari 2027 | Rp.600000 |
| DRS009 | Juicy Lucy | 7 November 2024 | Rp.300000 |
| DRS010 | Boy pablo | 30 Januari 2025 | Rp.700000 |
| DRS012 | Riski febian | 23 Agustus 2026 | Rp.600000 |
| DRS001 | Isrina | 6 April 2025 | Rp.600000 |
| DRS015 | Mahalini | 22 Maret 2028 | Rp.500000 |
+-----+-----+-----+-----+
Masukkan nomor konser yang ingin dihapus = 001
Nomor konser tidak ditemukan.
Masukkan nomor konser yang ingin dihapus = DRS001
Konser dengan ID DRS001 berhasil dihapus.
```

Gambar 3.41 Tampilan Program: Tampilan Menu Admin 4

Pada tampilan 4 seperti yang ada pada gambar diatas, yaitu user menginputkan nomor konser yang ingin dihapus tetapi dengan syarat menggunakan (DRS) jika tidak maka akan gagal dan nomor konser tidak ditemukan.

```
===== LIST KONSER =====
```

ID	Guest Star	Jadwal	Harga
DRS002	Sal Priadi	23 April 2025	Rp.160000
DRS003	Reality Club	23 Desember 2025	Rp.800000
DRS004	Bernadya	5 Maret 2025	Rp.300000
DRS006	Hindia	21 Februari 2026	Rp.300000
DRS007	Feast.	31 Oktober 2025	Rp.400000
DRS008	Ardhito Pramono	21 Februari 2027	Rp.600000
DRS009	Juicy Lucy	7 November 2024	Rp.300000
DRS010	Boy pablo	30 Januari 2025	Rp.700000
DRS012	Riski febian	23 Agustus 2026	Rp.600000
DRS015	Mahalini	22 Maret 2028	Rp.500000

```
Ingin menghapus konser lagi(Ya/Tidak)? = Tidak
```

Gambar 3.42 Tampilan Program: Tampilan Konser Berhasil Dihapus

Setelah di cek penghapusan konser berhasil, maka user ditanya ingin menghapus konser lagi atau tidak. Jika tidak maka akan kembali ke menu admin.

```
===== MENU ADMIN =====
[1]. Lihat Event Konser
[2]. Tambah Event Konser
[3]. Perbarui Event Konser
[4]. Hapus Event Konser
[5]. Logout
Masukkan pilihan = 5

===== SILAHKAN PILIH ROLE =====
[1]. Admin
[2]. Pengguna
[3]. Pengunjung
[4]. Keluar
```

Gambar 3.43 Tampilan Program: Tampilan Akan Logout

Setelah selesai menggunakan semua program, user ingin kembali maka memilih pilihan 5, setelahnya akan masuk kembali ke portal menu awal.

```
===== SILAHKAN PILIH ROLE =====  
[1]. Admin  
[2]. Pengguna  
[3]. Pengunjung  
[4]. Keluar  
Masukkan pilihan = 2
```

Gambar 3.44 Tampilan Program: Tampilan Memilih Menu 2

Pada portal menu awal ini untuk login sebagai user memilih pilihan 2 dan setelahnya akan masuk ke dalam menu pengguna.

```
===== PENGGUNA =====  
[1]. Login  
[2]. Daftar  
[3]. Kembali  
Masukkan pilihan = █
```

Gambar 3.45 Tampilan Program: Tampilan Awal Menu Pengguna

Pada tampilan ini memiliki 3 opsi untuk pengguna ,diminta untuk menginput pilihan untuk login, daftar dan keluar.

```
===== LOGIN PENGGUNA =====  
Username = Dwi Pebri  
Password = ****
```

Gambar 3.46 Tampilan Program: Tampilan Login pengguna

Pada menu ke-1 ini setelah akun berhasil dibuat dan proses login sukses maka akan masuk ke menu untuk pengguna.Maka pengguna diberi pilihan untuk mengakses menu yang diinginkan.


```
===== PENGGUNA =====
[1]. Login
[2]. Daftar
[3]. Kembali
Masukkan pilihan = 2

===== DAFTAR AKUN =====

Masukan Username:
Username (4-10 karakter) = Dwi Pebri

Masukan Password anda:
Password =2345
Akun berhasil dibuat.
```

Gambar 3.47 Tampilan Program:Tampilan Daftar akun pengguna

Ketika memilih menu pilihan ke-2 pengguna diminta untuk daftar akun agar bisa login, maka pengguna harus memasukkan username yang terdiri dari 4-10 karakter ,serta memasukkan password dan selanjutnya jika sudah sesuai maka akun berhasil dibuat.

```
===== MENU PENGGUNA =====
[1]. Lihat Event Konser
[2]. Beli Tiket Event Konser
[3]. Lihat Saldo E-Money
[4]. Lihat Tiket yang dibeli
[5]. Logout
Masukkan pilihan = █
```

Gambar 3.48 Tampilan Program: Tampilan Menu Pengguna

Pada tampilan menu pengguna diatas memiliki 5 opsi , pengguna diminta untuk menginput pilihan yang diinginkan.

```
Masukkan pilihan = 1

===== LIST KONSER =====
+-----+-----+-----+-----+
| ID | Guest Star | Jadwal | Harga |
+-----+-----+-----+-----+
| DRS002 | Sal Priadi | 23 April 2025 | Rp.160000 |
| DRS003 | Reality Club | 23 Desember 2025 | Rp.800000 |
| DRS004 | Bernadya | 5 Maret 2025 | Rp.300000 |
| DRS006 | Hindia | 21 Februari 2026 | Rp.300000 |
| DRS007 | Feast. | 31 Oktober 2025 | Rp.400000 |
| DRS008 | Ardhito Pramono | 21 Februari 2027 | Rp.600000 |
| DRS009 | Juicy Lucy | 7 November 2024 | Rp.300000 |
| DRS010 | Boy pablo | 30 Januari 2025 | Rp.700000 |
| DRS012 | Riski febian | 23 Agustus 2026 | Rp.600000 |
| DRS015 | Mahalini | 22 Maret 2028 | Rp.500000 |
+-----+-----+-----+-----+

===== PILIH KRITERIA SHORTING =====
[1]. Harga terendah ke tertinggi
[2]. Harga tetinggi ke terendah
[3]. Kembali
```

Gambar 3.49 Tampilan Program: Tampilan Menu Pengguna 1

User memilih pilihan menu 1, maka akan menampilkan List konser dari mulai dari Guest star, jadwal konser dan harga konser. Pada menu ini disediakan Shorting untuk memudahkan pengguna dalam menemukan list dari harga yang paling murah hingga mahal. Jika memilih kriteria shorting (1) maka akan menampilkan list dari harga terendah ke tertinggi, jika memilih kriteria shorting (2) maka akan menampilkan list harga dari tertinggi ke terendah dan jika memilih (3) maka akan kembali ke menu pengguna.

```
===== MENU PENGGUNA =====
[1]. Lihat Event Konser
[2]. Beli Tiket Event Konser
[3]. Lihat Saldo E-Money
[4]. Lihat Tiket yang dibeli
[5]. Logout
Masukkan pilihan = 2

===== BELI TIKET =====
Masukkan nomor konser yang ingin dipilih = DRS004
Jumlah tiket yang ingin dibeli = 2

Keterangan :
Guest Star = Bernadya
Rp.300000 X 2 = Rp.600000
ingin lanjutkan pembayaran atau kembali ke menu(ya/kembali)? = ya

saldo tidak cukup, silahkan melakukan pengisian.
Proses pembayaran dibatalkan.
```

Gambar 3.50 Tampilan Program: Tampilan Menu Pengguna 2

Untuk menu pilihan ke-2 ini pengguna ingin membeli tiket konser dan pengguna diminta untuk menginputkan nomor konser serta jumlah tiket yang ingin dibeli. Setelahnya akan ditampilkan total harga tiket yang diinginkan dan diberi pilihan untuk melanjutkan ke proses pembayaran atau kembali ke menu. Jika pengguna memilih ya maka akan tetap di program dan masuk ke proses pembayarn tetapi jika saldo tidak cukup maka proses pembelian tiket dibatalkan dan diberi pemberitahuan untuk melakukan pengisian saldo terlebih dahulu, jika pengguna memilih tidak maka akan keluar dari menu beli tiket dan kembali ke portal menu pengguna.

```
===== MENU PENGGUNA =====  
[1]. Lihat Event Konser  
[2]. Beli Tiket Event Konser  
[3]. Lihat Saldo E-Money  
[4]. Lihat Tiket yang dibeli  
[5]. Logout  
Masukkan pilihan = 3
```

Gambar 3.51 Tampilan Program:Tampilan Memilih menu 3

Disini pengguna memilih pilihan 3 maka masuk ke menu lihat saldo yang pengguna miliki.

```
===== E-MONEY =====  
Saldo Anda = Rp.0  
[1]. Top Up  
[2]. Kembali  
Masukkan pilihan = 1  
  
===== TOP UP =====  
[1]. Rp.50.000  
[2]. Rp.100.000  
[3]. Nominal lain  
[4]. Batalkan  
Masukkan pilihan = 3  
minimal top up Rp.20000 dan maksimal Rp.2000000  
Nominal = Rp. 200000
```

Gambar 3.52 Tampilan Program: Tampilan Menu Pengguna 3

Saldo yang pengguna miliki ditampilkan, dan pengguna diberi pilihan ingin Top Up atau kembali , Jika ingin top up maka akan masuk lagi ke menu Top Up yang ada beberapa pilihan didalamnya yang disediakan, disini pengguna memilih menu top up ke-3 ingin melakukan top up

nominal lain ,seperti pada gambar pengguna ingin top up dengan nominal 2 jutamaka pengguna memasukkan nominal nya.

```
===== TOP UP BERHASIL =====  
Nama : Dwi Pebri  
Nominal :Rp.2000000  
=====
```

Gambar 3.53 Tampilan Program: Tampilan Berhasil Top Up

Setelah meunggu proses Top up, jika berhasil maka saldo milik pengguna akan bertambah. Selanjutnya Ketika top up sudah berhasil maka akan kembali ke menu pengguna lagi.

```
===== MENU PENGGUNA =====  
[1]. Lihat Event Konser  
[2]. Beli Tiket Event Konser  
[3]. Lihat Saldo E-Money  
[4]. Lihat Tiket yang dibeli  
[5]. Logout  
Masukkan pilihan = 2  
  
===== BELI TIKET =====  
Masukkan nomor konser yang ingin dipilih = DRS004  
Jumlah tiket yang ingin dibeli = 2  
  
Keterangan :  
Guest Star = Bernadya  
Rp.300000 X 2 = Rp.600000  
ingin lanjutkan pembayaran atau kembali ke menu(ya/kembali)?
```

Gambar 3.54 Tampilan Program: Tampilan Pembayaran Konser

Disini Pengguna ingin membeli tiket dengan memilih menu ke-2 lagi setelah berhasil untuk top up, maka akan diminta input Kembali nomor konser dan jumlah tiket serta keterangan berapa total yang harus dibayar seluruhnya. Setelahnya diberi pilihan kembali ingin lanjut pembayaran atau tidak, Jika memilih keluar maka akan keluar dari program beli tiket.

```
===== INVOICE =====  
Nama : Dwi Pebri  
Guest Star :Bernadya  
Harga :Rp300000 x 2  
Total :Rp.600000  
=====  
apakah anda ingin menghapus tiket anda (ya/tidak) = █
```

Gambar 3.55 Tampilan Program: Tampilan Invoice

Tampilan ketika pengguna ingin melihat tiket yang telah berhasil dibeli yang tertera Nama, Guest star, Harga tiket dan total keseluruhan dari jumlah tiket yang dibeli. Setelah itu pengguna diberi pilihan ingin menghapus tiket atau tidak, jika ya maka tiket akan terhapus.

```
===== MENU PENGGUNA =====  
[1]. Lihat Event Konser  
[2]. Beli Tiket Event Konser  
[3]. Lihat Saldo E-Money  
[4]. Lihat Tiket yang dibeli  
[5]. Logout  
Masukkan pilihan = 5
```

Gambar 3.56 Tampilan Program: Tampilan Logout

Setelah berhasil untuk mengakses semua menu, pengguna ingin keluar dari program dan memilih pilihan ke-5 setelahnya akan balik ke menu awal.

```
===== SILAHKAN PILIH ROLE =====  
[1]. Admin  
[2]. Pengguna  
[3]. Pengunjung  
[4]. Keluar  
Masukkan pilihan = 3
```

Gambar 3.57 Tampilan Program: Tampilan Memilih Pengunjung

Selanjutnya memilih user memilih role ke-3 sebagai pengunjung yang hanya dapat melihat jadwal konser.

```
===== MODE PENGUNJUNG =====
```

```
===== LIHAT KONSER =====
```

ID	Guest Star	Jadwal	Harga
DRS002	Sal Priadi	23 April 2025	Rp.160000
DRS003	Reality Club	23 Desember 2025	Rp.800000
DRS004	Bernadya	5 Maret 2025	Rp.300000
DRS006	Hindia	21 Februari 2026	Rp.300000
DRS007	Feast.	31 Oktober 2025	Rp.400000
DRS008	Ardhito Pramono	21 Februari 2027	Rp.600000
DRS009	Juicy Lucy	7 November 2024	Rp.300000
DRS010	Boy pablo	30 Januari 2025	Rp.700000
DRS012	Riski febian	23 Agustus 2026	Rp.600000
DRS015	Mahalini	22 Maret 2028	Rp.500000

```
apakah anda ingin kembali atau login sebagai pengguna?(Kembali/login) =
```

Gambar 3.58 Tampilan Program: Tampilan Pengunjung

Pada menu pengunjung diatas hanya menampilkan jadwal konser dan diberi pilihan ingin login sebagai pengguna atau keluar dari program pengunjung.

```
===== SILAHKAN PILIH ROLE =====
[1]. Admin
[2]. Pengguna
[3]. Pengunjung
[4]. Keluar
Masukkan pilihan = 4

=== ***** SELAMAT TINGGAL DAN TERIMA KASIH ***** =====
PS D:\PROJECT_AKHIR_DDP>
```

Gambar 3.59 Tampilan Program: Tampilan Program Selesai

Jika telah selesai mengakses semua role 1, 2 dan 3 di menu awal, ketika memilih role 4 maka otomatis akan keluar dari semua program dan akan berhenti.

BAB IV PENUTUP

4.1 Kesimpulan

Dalam laporan ini kami telah menguraikan pembahasan mengenai Program sistem pengelolaan event konser. Kesimpulan yang diambil terhadap laporan kali ini yaitu, dengan membuat sistem pengelolaan event konser ini kami dapat mengimplementasikan teori yang telah didapatkan selama praktikum dan menjadikan visualisasi pemahaman teori menjadi lebih nyata dengan membuat program yang dapat membantu admin dalam menggunakan mengelola dan menggunakan fitur dengan cara digital tanpa menggunakan cara manual dalam menginput dan mengelola data pengunjung konser yang dapat meminimalkan terjadinya kesalahan data dengan menggunakan program tersebut serta untuk memudahkan pengguna bertransaksi dalam membeli tiket konser.

4.2 Saran

Adapun saran yang kami berikan untuk laporan yang telah kami buat yaitu sebagai berikut:

1. Perlunya pengembangan dalam pembuatan program Aplikasi Untuk Konser Online yang diharapkan mampu membantu para pengguna serta pengunjung untuk lebih memberikan kemudahan dan kenyamanan.
2. Kritik dan saran dari pembaca sangat diharapkan demi kesempurnaan penulisan laporan di kemudian hari

DAFTAR PUSTAKA

LAMPIRAN

Tabel Kontribusi

Nama	Kontribusi	Bagian
Dwi Pebriyanto Pradana (2409116012)	Konsep, Coding, Flowchart, Dan Laporan	<ol style="list-style-type: none"> 1. Mengkonsep Program 2. Coding 3. Perbaikan Flowchart 4. Perbaikan Laporan
Isrina Luthfiah (2409116003)	Flowchart, Pengecekan ,Laporan PDF	<ol style="list-style-type: none"> 1. Membuat Flowchart pengguna 2. Mengecek Bug dan Error 3. Penyusunan Laporan PDF 4. Menyarankan Konsep
Muhammad Sadikin Samir (2409116031)	Flowchart, Pengecekan ,Laporan PDF	<ol style="list-style-type: none"> 1. Membuat Flowchart Admin dan lain-lain 2. Memperbaiki Flowchart 3. Mencari Bug dan Error 4. Membuat Laporan Github 5. Menyarankan Konsep

Tabel 1.1 Tabel Kontribusi