

LAPORAN PRAKTIKUM
PROJECT AKHIR MEMBUAT KASIR



Disusun Oleh :

- **Dinda Parwati**
- **Rizky Firdaus**
- **Arga Surya S**

REKAYASA PERANGKAT LUNAK
SMKS BINA PUTRA

2025

1. Film.java

```
package kasirbioskop.model;

public class Film {
    private int id;
    private String judul;
    private String genre;
    private int durasi;

    public Film(int id, String judul, String genre, int durasi) {
        this.id = id;
        this.judul = judul;
        this.genre = genre;
        this.durasi = durasi;
    }

    public int getId() {
        return id;
    }

    public String getJudul() {
        return judul;
    }

    public String getGenre() {
        return genre;
    }

    public int getDurasi() {
        return durasi;
    }
}
```

2. Itempenjualan.java

```
package kasirbioskop.model;

public abstract class ItemPenjualan {
    protected String nama;
    protected double harga;

    public ItemPenjualan(String nama, double harga) {
        this.nama = nama;
        this.harga = harga;
    }

    public String getNama() {
        return nama;
    }

    public double getHarga() {
        return harga;
    }

    public abstract double hitungTotal();
}
```

3. Jadwal.java

```
package kasirbioskop.model;

import java.time.LocalDateTime;

public class Jadwal {
    private int id;
    private Film film;
    private String studio;
    private LocalDateTime waktuTayang;
    private double harga;

    public Jadwal(int id, Film film, String studio, LocalDateTime
waktuTayang, double harga) {
        this.id = id;
        this.film = film;
        this.studio = studio;
        this.waktuTayang = waktuTayang;
        this.harga = harga;
    }
}
```

```

    public int getId() {
        return id;
    }

    public Film getFilm() {
        return film;
    }

    public String getStudio() {
        return studio;
    }

    public LocalDateTime getWaktuTayang() {
        return waktuTayang;
    }

    public double getHarga() {
        return harga;
    }
}

```

4. **Makanminu.java**

```

package kasirbioskop.model;

public class MakananMinuman extends ItemPenjualan {
    private int jumlah;

    public MakananMinuman(String nama, double harga, int jumlah) {
        super(nama, harga);
        this.jumlah = jumlah;
    }

    @Override
    public double hitungTotal() {
        return harga * jumlah;
    }

    public int getJumlah() {
        return jumlah;
    }
}

```

5. Tiket.java

```
package kasirbioskop.model;

public class Tiket extends ItemPenjualan {
    protected String nomorKursi;

    public Tiket(String nama, double harga, String nomorKursi) {
        super(nama, harga);
        this.nomorKursi = nomorKursi;
    }

    @Override
    public double hitungTotal() {
        return harga;
    }

    public String getNomorKursi() {
        return nomorKursi;
    }
}
```

6. Tiketregular.java

```
package kasirbioskop.model;

public class TiketRegular extends Tiket {
    public TiketRegular(String nama, double harga, String nomorKursi) {
        super(nama, harga, nomorKursi);
    }

    @Override
    public double hitungTotal() {
        return harga;
    }
}
```

7. tiketVIP.java

```
package kasirbioskop.model;

public class TiketVIP extends Tiket {
    private double tambahanHarga = 30000;

    public TiketVIP(String nama, double harga, String nomorKursi) {
        super(nama, harga, nomorKursi);
    }
}
```

```

    }

    @Override
    public double hitungTotal() {
        return harga + tambahanHarga;
    }
}

```

8. Main.java

```

package com.cinemapos;

import java.time.LocalDateTime;
import java.util.*;

public class Main {
    private static final Scanner in = new Scanner(System.in);

    private static final List<Film> daftarFilm = new ArrayList<>();
    private static final List<Jadwal> daftarJadwal = new ArrayList<>();
    private static final List<MakananMinuman> daftarMenu = new
ArrayList<>();
    private static final List<Transaksi> daftarTransaksi = new
ArrayList<>();

    private static int tiketId = 1;
    private static int transaksiId = 1;

    public static void main(String[] args) {
        seedData();

        System.out.println("===== SISTEM KASIR BIOSKOP =====");
        System.out.print("Masukkan nama Anda: ");
        String nama = in.nextLine();

        System.out.print("Login sebagai (1=Admin / 2=Pelanggan): ");
        int role = Integer.parseInt(in.nextLine().trim());

        if (role == 1) {
            menuAdmin(nama);
        } else {
            menuPelanggan(nama);
        }
    }
}

```

```

private static void menuAdmin(String nama) {
    boolean jalan = true;
    while (jalan) {
        System.out.println("\n=== MENU ADMIN ===");
        System.out.println("1. Tambah Film");
        System.out.println("2. Tambah Jadwal Tayang");
        System.out.println("3. Tambah Menu Makanan/Minuman");
        System.out.println("4. Lihat Semua Data");
        System.out.println("5. Keluar");
        System.out.print("Pilih: ");
        int pilih = Integer.parseInt(in.nextLine().trim());

        switch (pilih) {
            case 1 -> tambahFilm();
            case 2 -> tambahJadwal();
            case 3 -> tambahMenu();
            case 4 -> tampilkanSemuaData();
            case 5 -> jalan = false;
            default -> System.out.println("Pilihan salah!");
        }
    }
}

```

```

private static void menuPelanggan(String nama) {
    boolean jalan = true;
    while (jalan) {
        System.out.println("\n=== MENU PELANGGAN ===");
        System.out.println("1. Lihat Film & Jadwal");
        System.out.println("2. Beli Tiket");
        System.out.println("3. Beli Makanan/Minuman");
        System.out.println("4. Lihat Transaksi Saya");
        System.out.println("5. Keluar");
        System.out.print("Pilih: ");
        int pilih = Integer.parseInt(in.nextLine().trim());

        switch (pilih) {
            case 1 -> tampilkanSemuaData();
            case 2 -> pembelianTiket(nama);
            case 3 -> pembelianMenu(nama);
            case 4 -> tampilkanTransaksi(nama);
            case 5 -> jalan = false;
            default -> System.out.println("Pilihan salah!");
        }
    }
}

```

```
    }  
  }  
}
```

```
private static void tambahFilm() {  
    System.out.print("Judul Film: ");  
    String judul = in.nextLine();  
    System.out.print("Genre: ");  
    String genre = in.nextLine();  
    System.out.print("Durasi (menit): ");  
    int durasi = Integer.parseInt(in.nextLine().trim());  
  
    Film f = new Film(daftarFilm.size() + 1, judul, genre, durasi);  
    daftarFilm.add(f);  
    System.out.println("Film berhasil ditambahkan!");  
}
```

```
private static void tambahJadwal() {  
    tampilkanFilm();  
    System.out.print("Pilih ID Film: ");  
    int filmId = Integer.parseInt(in.nextLine().trim());
```

```
    Optional<Film> film = daftarFilm.stream().filter(f -> f.getId() ==  
filmId).findFirst();  
    if (film.isEmpty()) {  
        System.out.println("Film tidak ditemukan!");  
        return;  
    }
```

```
    System.out.print("Studio: ");  
    String studio = in.nextLine();  
    System.out.print("Harga Tiket: ");  
    long harga = Long.parseLong(in.nextLine().trim());
```

```
    Jadwal j = new Jadwal(daftarJadwal.size() + 1, film.get(), studio,  
LocalDateTime.now().plusHours(2), harga);  
    daftarJadwal.add(j);  
    System.out.println("Jadwal berhasil ditambahkan!");  
}
```

```
private static void tambahMenu() {  
    System.out.print("Nama Menu: ");  
    String nama = in.nextLine();
```



```

        System.out.print("Harga: ");
        long harga = Long.parseLong(in.nextLine().trim());

        daftarMenu.add(new MakananMinuman(daftarMenu.size() + 1,
nama, harga));
        System.out.println("Menu berhasil ditambahkan!");
    }

    private static void pembelianTiket(String pelanggan) {
        tampilkanJadwal();
        System.out.print("Masukkan ID Jadwal: ");
        int id = Integer.parseInt(in.nextLine().trim());

        Optional<Jadwal> jadwalOpt = daftarJadwal.stream().filter(j ->
j.getId() == id).findFirst();
        if (jadwalOpt.isEmpty()) {
            System.out.println("Jadwal tidak ditemukan.");
            return;
        }

        Jadwal j = jadwalOpt.get();
        j.tampilkanKursi();

        System.out.print("Pilih nomor kursi: ");
        int kursi = Integer.parseInt(in.nextLine().trim());
        if (!j.isSeatAvailable(kursi)) {
            System.out.println("Kursi tidak tersedia.");
            return;
        }

        System.out.print("Tipe tiket (1=Regular / 2=VIP): ");
        int tipe = Integer.parseInt(in.nextLine().trim());

        Tiket tiket;
        if (tipe == 1)
            tiket = new TiketRegular(tiketId++, j.getId(), kursi, "TERJUAL",
j.getHarga());
        else
            tiket = new TiketVIP(tiketId++, j.getId(), kursi, "TERJUAL",
j.getHarga());

        j.bookSeat(kursi);
    }

```

```

        Transaksi trx = new Transaksi(transaksiId++, pelanggan, 0,
LocalDateTime.now());
        trx.tambahTiket(tiket);
        daftarTransaksi.add(trx);

        System.out.println("Tiket berhasil dibeli!");
        tiket.cetakInfo();
    }

    private static void pembelianMenu(String pelanggan) {
        tampilkanMenu();
        System.out.print("Masukkan ID Menu (0 = batal): ");
        int id = Integer.parseInt(in.nextLine().trim());
        if (id == 0) return;

        Optional<MakananMinuman> m = daftarMenu.stream().filter(x ->
x.getId() == id).findFirst();
        if (m.isEmpty()) {
            System.out.println("Menu tidak ditemukan!");
            return;
        }

        Transaksi trx = new Transaksi(transaksiId++, pelanggan, 0,
LocalDateTime.now());
        trx.tambahItem(m.get());
        daftarTransaksi.add(trx);
        System.out.println("Pembelian " + m.get().getNama() + " berhasil!");
    }

    private static void tampilkanTransaksi(String pelanggan) {
        boolean ada = false;
        for (Transaksi t : daftarTransaksi) {
            if (t.toString().contains(pelanggan)) {
                System.out.println(t);
                ada = true;
            }
        }
        if (!ada) System.out.println("Belum ada transaksi untuk " +
pelanggan);
    }

    private static void tampilkanSemuaData() {
        tampilkanFilm();
    }

```

```

        tampilkanJadwal();
        tampilkanMenu();
    }

    private static void tampilkanFilm() {
        System.out.println("\nDaftar Film:");
        for (Film f : daftarFilm) {
            System.out.printf("%d. %s (%s, %d menit)%n", f.getId(),
f.getJudul(), f.getGenre(), f.getDurasi());
        }
    }

    private static void tampilkanJadwal() {
        System.out.println("\nJadwal Tayang:");
        for (Jadwal j : daftarJadwal) {
            System.out.printf("ID %d | Film: %s | Studio: %s | Harga: Rp
%d%n",
                j.getId(), j.getFilm().getJudul(), j.getStudio(), j.getHarga());
        }
    }

    private static void tampilkanMenu() {
        System.out.println("\nMenu Makanan & Minuman:");
        for (MakananMinuman m : daftarMenu) {
            System.out.printf("%d. %s - Rp %,d%n", m.getId(), m.getNama(),
m.getHarga());
        }
    }

    private static void seedData() {
        daftarFilm.add(new Film(1, "Petualangan Waktu", "Sci-Fi", 120));
        daftarFilm.add(new Film(2, "Hati yang Hilang", "Drama", 100));

        daftarJadwal.add(new Jadwal(1, daftarFilm.get(0), "Studio 1",
LocalDateTime.now().plusHours(3), 50000));
        daftarJadwal.add(new Jadwal(2, daftarFilm.get(1), "Studio 2",
LocalDateTime.now().plusHours(4), 45000));

        daftarMenu.add(new MakananMinuman(1, "Popcorn", 20000));
        daftarMenu.add(new MakananMinuman(2, "Cola", 15000));
    }
}

```

HASIL PROGRAM

==== MENU ADMIN ====

1. Tambah Film

```
=== MENU ADMIN ===
1. Tambah Film
2. Tambah Jadwal Tayang
3. Tambah Menu Makanan/Minuman
4. Lihat Semua Data
5. Keluar
Pilih: 1
Judul Film: asep
Genre: romance
Durasi (menit): 120
Film berhasil ditambahkan!
```

2. Tambah Jadwal Tayang

```
=== MENU ADMIN ===
1. Tambah Film
2. Tambah Jadwal Tayang
3. Tambah Menu Makanan/Minuman
4. Lihat Semua Data
5. Keluar
Pilih: 2

Daftar Film:
1. Petualangan Waktu (Sci-Fi, 120 menit)
2. Hati yang Hilang (Drama, 100 menit)
3. asep (romance, 120 menit)
Pilih ID Film: 1
Studio: AI
Harga Tiket: 2000
Jadwal berhasil ditambahkan!
```

3. Tambah Menu Makanan/Minuman

```
=== MENU ADMIN ===
1. Tambah Film
2. Tambah Jadwal Tayang
3. Tambah Menu Makanan/Minuman
4. Lihat Semua Data
5. Keluar
Pilih: 3
Nama Menu: esjeruk
Harga: 10000
Menu berhasil ditambahkan!
```

4. Lihat Semua Data

=== MENU ADMIN ===

1. Tambah Film
2. Tambah Jadwal Tayang
3. Tambah Menu Makanan/Minuman
4. Lihat Semua Data
5. Keluar

Pilih: 4

Daftar Film:

1. Petualangan Waktu (Sci-Fi, 120 menit)
2. Hati yang Hilang (Drama, 100 menit)
3. asep (romance, 120 menit)

Jadwal Tayang:

ID 1 | Film: Petualangan Waktu | Studio: Studio 1 | Harga: Rp 50.000
ID 2 | Film: Hati yang Hilang | Studio: Studio 2 | Harga: Rp 45.000
ID 3 | Film: Petualangan Waktu | Studio: AI | Harga: Rp 2.000

Menu Makanan & Minuman:

1. Popcorn - Rp 20.000
2. Cola - Rp 15.000
3. esjeruk - Rp 10.000

=== MENU PELANGGAN ===

1. Lihat Film & Jadwal

=== MENU PELANGGAN ===

1. Lihat Film & Jadwal
2. Beli Tiket
3. Beli Makanan/Minuman
4. Lihat Transaksi Saya
5. Keluar

Pilih: 1

Daftar Film:

1. Petualangan Waktu (Sci-Fi, 120 menit)
2. Hati yang Hilang (Drama, 100 menit)

Jadwal Tayang:

ID 1 | Film: Petualangan Waktu | Studio: Studio 1 | Harga: Rp 50.000
ID 2 | Film: Hati yang Hilang | Studio: Studio 2 | Harga: Rp 45.000

Menu Makanan & Minuman:

1. Popcorn - Rp 20.000
2. Cola - Rp 15.000

2. Beli Tiket

```
=== MENU PELANGGAN ===
1. Lihat Film & Jadwal
2. Beli Tiket
3. Beli Makanan/Minuman
4. Lihat Transaksi Saya
5. Keluar
Pilih: 2

Jadwal Tayang:
ID 1 | Film: Petualangan Waktu | Studio: Studio 1 | Harga: Rp 50.000
ID 2 | Film: Hati yang Hilang | Studio: Studio 2 | Harga: Rp 45.000
Masukkan ID Jadwal: 2
Peta kursi (A = tersedia, X = terisi):
 1[A]  2[A]  3[A]  4[A]  5[A]  6[A]
 7[A]  8[A]  9[A] 10[A] 11[A] 12[A]
13[A] 14[A] 15[A] 16[A] 17[A] 18[A]
19[A] 20[A]
Pilih nomor kursi: 12
Tipe tiket (1=Regular / 2=VIP): 2
Tiket berhasil dibeli!
[VIP] Tiket #1 | Kursi 12 | Harga Rp 70.000
```

3. Beli Makanan/Minuman

```
=== MENU PELANGGAN ===
1. Lihat Film & Jadwal
2. Beli Tiket
3. Beli Makanan/Minuman
4. Lihat Transaksi Saya
5. Keluar
Pilih: 3

Menu Makanan & Minuman:
1. Popcorn - Rp 20.000
2. Cola - Rp 15.000
Masukkan ID Menu (0 = batal): 1
Pembelian Popcorn berhasil!
```

4. Lihat Transaksi Saya

```
=== MENU PELANGGAN ===
1. Lihat Film & Jadwal
2. Beli Tiket
3. Beli Makanan/Minuman
4. Lihat Transaksi Saya
5. Keluar
Pilih: 4

Transaksi #1 | andy | Total Rp 70000 | 2025-10-25T17:51:23.709244200
Transaksi #2 | andy | Total Rp 20000 | 2025-10-25T17:52:17.049753600
```