

LAPORAN TUGAS BESAR

PEMROGRAMAN BERORIENTASI OBJEK

PEMESANAN TIKET KERETA API

PEMINATAN : SOFTWARE ENGINEERING

Dosen: Adi Wahyu Pribadi, S.Si.,M.Kom



Disusun Oleh:

4522210004_TITIN SUGIHARTINI

PROGRAM STUDI

TEKNIK INFORMATIKA

UNIVERSITAS PANCASILA

JAKARTA SELATAN

2022/2023

KATA PENGANTAR

Pada era modern ini, teknologi informasi telah menjadi bagian yang tidak terpisahkan dari kehidupan sehari-hari. Hal ini juga berlaku untuk sistem pemesanan tiket kereta api. Sistem yang efisien dan terintegrasi sangat diperlukan untuk mendukung mobilitas masyarakat.

Proyek ini, yang berjudul "Sistem Pemesanan Tiket Kereta Api," merupakan upaya untuk menerapkan konsep-konsep Pengembangan Berorientasi Objek (PBO) dalam menghadirkan solusi untuk memenuhi kebutuhan perjalanan kereta api. Dalam proyek ini, penulis mendesain dan mengimplementasikan kelas-kelas yang merepresentasikan entitas dan proses utama dalam sistem pemesanan tiket kereta api. Mulai dari pengguna atau pelanggan, tiket, stasiun, hingga layanan pemesanan, kami memanfaatkan konsep Pengembangan Berorientasi Objek (PBO) seperti relasi, constructor, abstrak, interface untuk menciptakan sistem yang fleksibel, mudah dipelihara, dan dapat diandalkan.

Analisis mendalam terhadap kebutuhan pelanggan, perancangan struktur kelas, hingga implementasi dengan bahasa pemrograman Java, menjadi langkah-langkah penting dalam menghasilkan sistem yang responsif dan efektif. Penulis percaya bahwa proyek ini tidak hanya menjadi peluang untuk memahami lebih dalam konsep Pengembangan Berorientasi Objek (PBO), tetapi juga menyajikan solusi praktis untuk mempermudah proses pemesanan tiket kereta api.

Penulis mengucapkan terima kasih kepada semua pihak yang telah memberikan dukungan, saran, dan inspirasi selama proses pengembangan proyek ini. Semoga kontribusi kecil ini dapat memberikan nilai tambah dalam meningkatkan pengalaman perjalanan dan efisiensi layanan pemesanan tiket kereta api.

Jakarta, 20 Desember 2023

Titin Sugihartini

DAFTAR ISI

KATA PENGANTAR	i
DAFTAR ISI	ii
BAB I.....	1
PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	1
1.3 Tujuan	1
BAB II.....	2
2.1 Pengertian Pemrograman Berorientasi Objek.....	2
2.2 Library Java.....	2
3.3 Bahasa Java	3
BAB III.....	4
PEMBASAHAN.....	4
3.1 Penjelasan Singkat tentang Pembelian Tiket Kereta Api	4
3.2 Rumusan Masalah.....	4
3.3 Fitur – Fitur Yang Tersedia :	4
3.4 Java Library yang digunakan	4
3.5 Siapa yang terlibat	4
3.6 Use case dari masing masing actor	4
3.7 Data apa saja yang disimpan.....	5
3.8 Class Diagram	5
3.9 Buat Turunan / relasi / abstraksi/ interface.....	5
3.10 Skenario	6
3.11 Proses Pembuatan Java (Pembelian Tiket Kereta Api).....	6
a. Class Kereta.....	6
b. Class Gerbong	7
c. Class Customer.....	7
d. Class Tiket.....	8
e. Interface InfoTransportasi	9

f. Class Main	9
BAB IV	13
PENUTUP	13
4.1 Kesimpulan.....	13
4.2 Kritik	13
4.3 Saran	13
DAFTAR PUSTAKA	14

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pemesanan tiket kereta api secara online telah menjadi hal yang umum di era modern. Layanan ini memberikan kemudahan bagi pelanggan untuk memesan tiket tanpa harus mengunjungi stasiun secara langsung. Implementasi sistem pemesanan tiket kereta api menggunakan bahasa pemrograman Java merupakan langkah yang tepat untuk mendukung kemudahan akses pelanggan terhadap layanan ini. Aplikasi ini diharapkan dapat memberikan pengalaman pemesanan tiket yang lebih cepat, efisien, dan user-friendly.

1.2 Rumusan Masalah

- 1) Bagaimana mengimplementasikan struktur data untuk menyimpan informasi tiket, gerbong, dan kereta api?
- 2) Bagaimana mengelola proses pemesanan tiket, termasuk pemilihan tujuan, jam keberangkatan, dan gerbong?

1.3 Tujuan

- 1) Menyediakan sistem pemesanan tiket kereta api yang efisien dan mudah digunakan.
- 2) Memahami dan mengimplementasikan konsep pemrograman berorientasi objek dalam pembuatan aplikasi.
- 3) Memberikan pengalaman pemesanan tiket yang nyaman bagi pengguna.
- 4) Meningkatkan efisiensi dan akurasi dalam manajemen data tiket, gerbong, dan kereta api.

BAB II

LANDASAN TEORI

2.1 Pengertian Pemrograman Berorientasi Objek

Pemrograman berorientasi objek (PBO) adalah paradigma pemrograman yang memodelkan dunia nyata sebagai objek. Objek adalah unit dasar dari PBO yang terdiri dari data dan perilaku. Data adalah informasi yang disimpan oleh objek, sedangkan perilaku adalah tindakan yang dapat dilakukan oleh objek.

Pemrograman berorientasi objek (PBO) menawarkan sejumlah keuntungan yang dapat memberikan kontribusi penting dalam pengembangan perangkat lunak. Pertama, PBO memungkinkan penciptaan kode yang lebih efisien dengan menggabungkan data dan perilaku ke dalam unit tunggal yang disebut objek. Hal ini membuat pengelolaan data dan logika program menjadi lebih terorganisir dan mudah dipahami. Kedua, kekuatan PBO terletak pada kemampuannya untuk membuat kode yang lebih kuat dan aman karena data dan perilaku objek dapat dilindungi dari akses yang tidak sah atau tidak diinginkan. Ketiga, PBO memberikan fleksibilitas yang tinggi karena modifikasi data dan perilaku objek dapat dilakukan tanpa mempengaruhi bagian lain dari program. Dengan kata lain, PBO menyediakan dasar yang kuat untuk pengembangan perangkat lunak yang efisien, aman, dan fleksibel.

2.2 Library Java

- 1) Java Standard Library (Java API)
Java API adalah kumpulan alat dan fungsi yang dapat digunakan untuk mengembangkan aplikasi Java. Alat-alat ini mencakup kelas-kelas untuk mengelola data, mengakses file, dan menangani teks.
- 2) Apache Commons Libraries
Apache Commons Libraries adalah kumpulan alat dan fungsi yang dapat digunakan untuk membuat aplikasi Java lebih mudah dikembangkan. Alat-alat ini mencakup kelas-kelas untuk operasi string, operasi I/O, dan manipulasi koleksi.
- 3) Guava (Google Core Libraries for Java)
Guava adalah kumpulan alat dan fungsi yang dapat digunakan untuk membuat aplikasi Java lebih kuat dan efisien. Alat-alat ini mencakup kelas-kelas untuk koleksi, konfigurasi, dan utilitas lainnya.
- 4) Log4j atau SLF4J

Log4j dan SLF4J adalah alat yang dapat digunakan untuk mencatat aktivitas aplikasi. Alat-alat ini memungkinkan Anda untuk menentukan tingkat log, output, dan format pesan.

3.3 Bahasa Java

Bahasa Java adalah bahasa pemrograman tingkat tinggi yang bersifat objek, modular, dan platform-independen. Dikembangkan oleh Sun Microsystems pada tahun 1995, Java sejak itu telah menjadi salah satu bahasa pemrograman yang paling populer dan luas digunakan di dunia. Java menjadi pilihan yang populer untuk berbagai jenis pengembangan perangkat lunak, termasuk aplikasi desktop, perangkat seluler, dan aplikasi web.

- 1) Objek dan Modular: Program Java terdiri dari objek-objek yang saling berinteraksi, sehingga memudahkan pemeliharaan dan pengembangan kode.
- 2) Platform-Independen: Program Java dapat dijalankan di berbagai platform tanpa perlu dimodifikasi ulang, berkat penggunaan mesin virtual Java (JVM).
- 3) Sintaksis yang Bersih dan Mudah Dipahami: Bahasa Java didesain dengan sintaksis yang bersih dan mudah dibaca, sehingga mengurangi kompleksitas dan memfasilitasi pengembangan kode.
- 4) Keamanan: Java memiliki fitur-fitur keamanan bawaan yang melindungi sistem dari bahaya keamanan seperti buffer overflow dan pointer manipulation.
- 5) Ekosistem dan Keterbukaan: Java memiliki ekosistem yang kuat dengan ribuan pustaka dan framework yang mendukung berbagai kebutuhan pengembangan aplikasi.
- 6) Kemampuan Multithreading: Java dapat menangani multithreading, sehingga ideal untuk pengembangan aplikasi yang membutuhkan kinerja tinggi dan pengelolaan banyak tugas secara bersamaan.
- 7) Pembaruan Berkala: Java secara teratur mengalami pembaruan dan peningkatan fitur, sehingga bahasa ini tetap relevan dan mendukung tren terkini dalam pengembangan perangkat lunak.

BAB III

PEMBASAHAN

3.1 Penjelasan Singkat tentang Pembelian Tiket Kereta Api

Pembelian tiket kereta api adalah proses di mana seorang pelanggan dapat memesan dan membeli tiket perjalanan kereta api untuk tujuan tertentu. Proses ini melibatkan pemilihan tujuan, jam keberangkatan, serta pemilihan jenis kelas atau gerbong yang diinginkan. Setelah itu, pelanggan akan menerima informasi tentang kode tujuan, jam keberangkatan, dan harga tiket. Pada sistem ini, kita akan menggunakan pendekatan berorientasi objek (PBO) dalam mengimplementasikan proses pemesanan tiket kereta api.

3.2 Rumusan Masalah

- a. Pelanggan yang sering kehabisan tiket kereta saat sudah di Loker pemesanan.
- b. Petugas Loker yang sering kali membeludak karena pemesanan tiket secara offline.
- c. Melayani pembayaran dengan berbagai metode.

3.3 Fitur – Fitur Yang Tersedia :

- a. Menampilkan Jenis Kereta
- b. Menampilkan Kereta Tujuan
- c. Menampilkan Kereta Keberangkatan
- d. Menampilkan Jam Keberangkatan
- e. Menampilkan Harga Tiket

3.4 Java Library yang digunakan

- a. `Java.util.Scanner` (untuk menerima input dari pengguna)
- b. `Java.util.ArrayList` (untuk menyimpan daftar Kereta, Gerbong, dan tiket)

3.5 Siapa yang terlibat

- a. Pelanggan
- b. Petugas Tiket

3.6 Use case dari masing masing actor

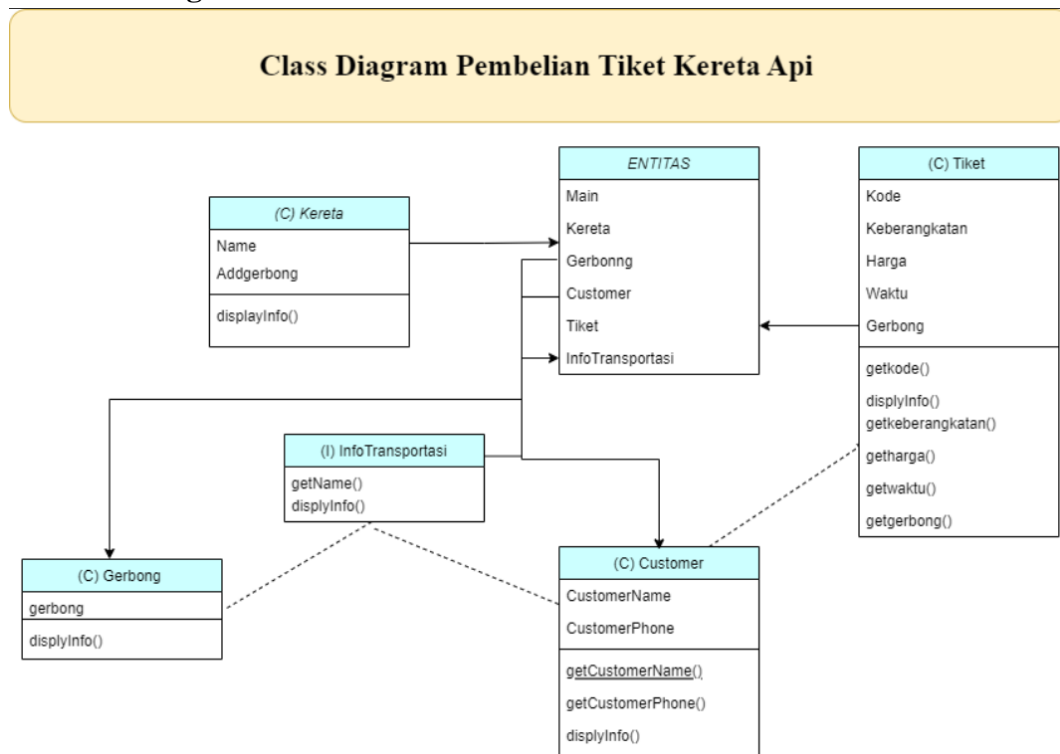
- a. Pelanggan :
 - Melihat daftar kereta
 - Memilih Gerbong
 - Memilih Kereta Tujuan

- Memilih Kereta Keberangkatan
 - Memilih Jam Keberangkatan
 - Melakukan Pemesanan Tiket
- b. Petugas Tiket :
- Menerima pesanan tiket
 - Memproses pesanan tiket
 - Mencetak pesanan tiket.

3.7 Data apa saja yang disimpan

- a. Informasi pengguna (nama, nomor telepon).
- b. Daftar kereta
- c. Keberangkatan kereta
- d. Tujuan kereta
- e. Jam keberangkatan kereta
- f. Informasi pemesanan tiket (kode kereta, Keberangkatan, Jam keberangkatan, Tujuan, Jumlah Pemesanan kereta).

3.8 Class Diagram



3.9 Buat Turunan / relasi / abstraksi/ interface

- a. Turunan

Gerbong dan Customer adalah turunan dari InfoTransportasi karena keduanya mengimplementasikan interface tersebut.

Tiket adalah turunan dari Customer karena Tiket memperluas (extends) kelas Customer.

Kereta adalah turunan dari InfoTransportasi karena Kereta mengimplementasikan interface tersebut.

b. Relasi

Main memiliki relasi dengan semua kelas karena menjadi tempat eksekusi program dan menggunakan objek-objek dari kelas lain.

Gerbong memiliki relasi dengan Kereta karena objek Kereta menggunakan objek Gerbong dalam strukturnya.

c. Interface

InfoTransportasi adalah sebuah interface yang diterapkan oleh Gerbong dan Kereta. Ini menetapkan kontrak bahwa kelas-kelas yang mengimplementasikan interface tersebut harus menyediakan implementasi untuk metode-metode yang didefinisikan dalam interface.

3.10 Skenario

- Pengguna menginstal aplikasi pemesanan Tiket Kereta API
- Pengguna memasukkan data diri seperti nama dan nomor telepon
- Pengguna memilih jenis gerbong kereta yang akan di gunakan
- Pengguna memilih lokasi keberangkatan
- Pengguna memilih lokasi Tujuan
- Pengguna memilih Jam keberangkatan
- System akan menyimpan informasi dan menampilkan invoice pemesanan tiket
- Pengguna dapat melihat tiket yang sudah di pesan.

3.11 Proses Pembuatan Java (Pembelian Tiket Kereta Api)

a. Class Kereta

```
package TiketKeretaAPI;

import java.util.ArrayList;
import java.util.List;

public class Kereta {
    private String nama;
    private List<Gerbong> gerbongs;

    public Kereta(String nama) {
```

```

        this.nama = nama;
        this.gerbongs = new ArrayList<>();
    }

    public void addGerbong(Gerbong gerbong) {
        gerbongs.add(gerbong);
    }

    public void displayInfo() {
        System.out.println("Info Kereta: " + nama);
        for (Gerbong gerbong : gerbongs) {
            gerbong.displayInfo();
        }
    }

    public String displayInfoGerbong(int pilihanGerbong) {
        if (pilihanGerbong >= 1 && pilihanGerbong <= gerbongs.size()) {
            return gerbongs.get(pilihanGerbong - 1).displayInfo();
        } else {
            return "Pilihan gerbong tidak valid.";
        }
    }
}

```

b. Class Gerbong

```

package TiketKeretaAPI;

class Gerbong {
    private String nama;

    public Gerbong(String nama) {
        this.nama = nama;
    }

    public String getNama() {
        return nama;
    }

    public String displayInfo() {
        return nama;
    }
}

```

c. Class Customer

```

package TiketKeretaAPI;

public class Customer {
    protected String CustomerName;
    protected String CustomerPhone;

    public String getName(){

```

```

        return this.CustomerName;
    }
    public void setName (String n){
        this.CustomerName = n;
    }
    public String getTelepon(){
        return this.CustomerPhone;
    }
    public void setTelepon (String t) {
        this.CustomerPhone=t;
    }
    public void displayInfo() {
    }
}

```

d. Class Tiket

```

package TiketKeretaAPI;

public class Tiket extends Customer {
    private String kode;
    private String keberangkatan;
    private int hargaTiket;
    private String jam;
    private String gerbong;

    public Tiket(String kode,String keberangkatan, int harga, String waktu, String
gerbong) {
        super();
        this.kode = kode;
        this.keberangkatan = keberangkatan;
        this.hargaTiket = harga;
        this.jam = waktu;
        this.gerbong = gerbong;
    }

    public Tiket() {
    }

    public String getKode() {
        return kode;
    }
    public void setKode(String kd) {
        this.kode = kd;
    }

    public String getKeberangkatan(){
        return keberangkatan;
    }
    public void setKeberangkatan(String keberangkatan){
        this.keberangkatan = keberangkatan;
    }
}

```

```

public String getGerbong() {
    return gerbong;
}

public void setGerbong(String gerbong) {
    this.gerbong = gerbong;
}

public int getHarga() {
    return hargaTiket;
}

public void setHarga(int hrg) {
    this.hargaTiket = hrg;
}

public String getWaktu() {
    return jam;
}

public void setWaktu(String waktu) {
    this.jam = waktu;
}
}

```

e. Interface InfoTransportasi

```

package TiketKeretaAPI;

public interface InfoTransportasi {
    String getNama();
    String displayInfo();
}

```

f. Class Main

```

package TiketKeretaAPI;

import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        String kode;
        String keberangkatan;
        String jam;
        int harga = 120000;
        Scanner input = new Scanner(System.in);
        Tiket user = new Tiket();
        System.out.println("Pemesanan Tiket Kereta Api");
        System.out.println("Masukkan Nama      :");
        String nama = input.next();
        System.out.println("Masukkan Nomor Telepon :");
        String nomor = input.next();
    }
}

```

```

user.setName(nama);
user.setTelepon(nomor);

System.out.println("Pilihan Keberangkatan :");
System.out.println("1. Manggarai ");
System.out.println("2. Pasar Senen");
System.out.println("3. Gambir ");
System.out.print("Masukkan pilihan :");
int a = input.nextInt();
if (a == 1) {
    keberangkatan = "M4994r41";
} else if (a == 2) {
    keberangkatan = "P454r 53n3n";
} else {
    keberangkatan = "64m31r";
}

System.out.println("Pilihan Tujuan :");
System.out.println("1. Jakarta ");
System.out.println("2. Yogya");
System.out.println("3. Bandung ");
System.out.print("Masukkan pilihan :");
int c = input.nextInt();
if (c == 1) {
    kode = "74k4rt4";
} else if (c == 2) {
    kode = "Y09y4";
} else {
    kode = "34nd03n9";
}

Kereta kereta = new Kereta("Argo Dwipangga");
Gerbong gerbong1 = new Gerbong("Gerbong Eksekutif");
Gerbong gerbong2 = new Gerbong("Gerbong Ekonomi");
System.out.println("Pilih Gerbong:");
System.out.println("1. " + gerbong1.getNama());
System.out.println("2. " + gerbong2.getNama());
System.out.print("Masukkan pilihan gerbong: ");
int pilihanGerbong = input.nextInt();

if (pilihanGerbong == 1) {
    kereta.addGerbong(gerbong1);
} else if (pilihanGerbong == 2) {
    kereta.addGerbong(gerbong2);
} else {
    System.out.println("Pilihan gerbong tidak valid.");
    return;
}

System.out.println("Pilihan Jam Keberangkatan");
System.out.println("1. 15:00");
System.out.println("2. 18:00");
System.out.println("3. 21:00");
System.out.print("Masukkan Pilihan :");
int b = input.nextInt();

```


Pilihan Tujuan :

1. Jakarta
2. Yogya
3. Bandung

Masukkan pilihan :3

Pilih Gerbong:

1. Gerbong Eksekutif
2. Gerbong Ekonomi

Masukkan pilihan gerbong: 1

Pilihan Jam Keberangkatan

1. 15:00
2. 18:00
3. 21:00

Masukkan Pilihan :3

=====PEMESANAN=====

=====

Costumer Nama : TITIN

Costumer Phone : 4522210004

=====

Kode Keberangkatan : M4994r41

Kode Tujuan : 34nd03n9

Info Gerbong : Gerbong Eksekutif

Jam Berangkat : 20:00

Harga Tiket : 120000Rupiah

=====

BAB IV

PENUTUP

4.1 Kesimpulan

Pemrograman Berorientasi Objek (PBO) memberikan keuntungan dalam efisiensi, kekuatan, dan fleksibilitas dalam pengembangan perangkat lunak. Penerapan konsep class, inheritance, encapsulation, dan polymorphism memungkinkan pembuatan kode yang lebih terstruktur, mudah dimengerti, dan dapat diubah dengan lebih mudah. Pemrograman Berorientasi Objek (PBO) memberikan keuntungan dalam efisiensi, kekuatan, dan fleksibilitas dalam pengembangan perangkat lunak. Penerapan konsep class, inheritance, encapsulation, dan polymorphism memungkinkan pembuatan kode yang lebih terstruktur, mudah dimengerti, dan dapat diubah dengan lebih mudah.

4.2 Kritik

Sistem pemesanan tiket kereta api ini telah memberikan gambaran umum tentang pemesanan tiket kereta api. Namun, masih terdapat beberapa aspek yang perlu diperhatikan lebih lanjut, yaitu:

Keamanan data pelanggan dan pembayaran belum diimplementasikan secara rinci. Manajemen error perlu ditingkatkan untuk mengatasi situasi-situasi yang tidak terduga dan memberikan feedback yang lebih informatif kepada pengguna.

4.3 Saran

Penulis menyadari bahwa laporan yang telah dibuat masih memiliki beberapa kekurangan dalam pembuatan aplikasi pembelian tiket kereta api. Meskipun aplikasi ini sudah berjalan, tetapi masih terdapat ruang untuk pengembangan lebih lanjut. Oleh karena itu, penulis mengajak pembaca untuk lebih mendalami dan memahami materi, terutama dalam mata kuliah Pemrograman Berorientasi Objek. Dengan begitu, diharapkan pembaca dapat lebih praktis dalam mengaplikasikan pemrograman dengan Bahasa Java dalam kehidupan sehari-hari.

DAFTAR PUSTAKA

[1] Adi Wahyu Pribadi, S.Si.,M.Kom, 2023 “ MODUL Pemrograman Berorientasi Object” semester III

[2]Huda Muhammad, 2019 “IMPLEMENTASI E-TICKETING KERETA API BERBASIS JAVA NETBEANS (STUDI KASUS : TANIA TRAVEL)” diakses 21 Desember 2023 : Seminar Nasional Riset dan Inovasi Teknologi (SEMNAS RISTEK) 2022

[3]Prama Yoga,2023 “Contoh Program Java Pemesanan Tiket Kereta Api “ diakses pada 20 Desember 2023: <https://yowatech.id/contoh-program-java-pemesanan-tiket-kereta-api/>

[4]Rival, 2016 “Pemrograman Java” diakses pada 21 Desember 2023 : <https://www.rival-face.com/2018/06/tugas-kuliah-pemrograman-java-source.html?m=1>