MANAJEMEN KONFIGURASI DAN EVOLUSI PERANGKAT LUNAK
LAPORAN TUGAS W04

PRAKTIKUM REFACTORING

disusun Oleh:
Andera Singgih Pratama
2211104007
SE0601

S1 REKAYASA PERANGKAT LUNAK
FAKULTAS INFORMATIKA TELKOM UNIVERSITY
2025

1. Pendahuluan

Dalam pengembangan perangkat lunak, struktur kode yang baik sangat penting untuk meningkatkan keterbacaan, pemeliharaan, dan skalabilitas. Pada tugas ini, dilakukan refaktorisasi terhadap class Song.java agar lebih modular, mengikuti prinsip Single Responsibility Principle (SRP), dan menghilangkan bad smell dalam kode.

2. Identifikasi Masalah dalam Kode Asli

Beberapa masalah yang ditemukan pada kode asli Song.java:

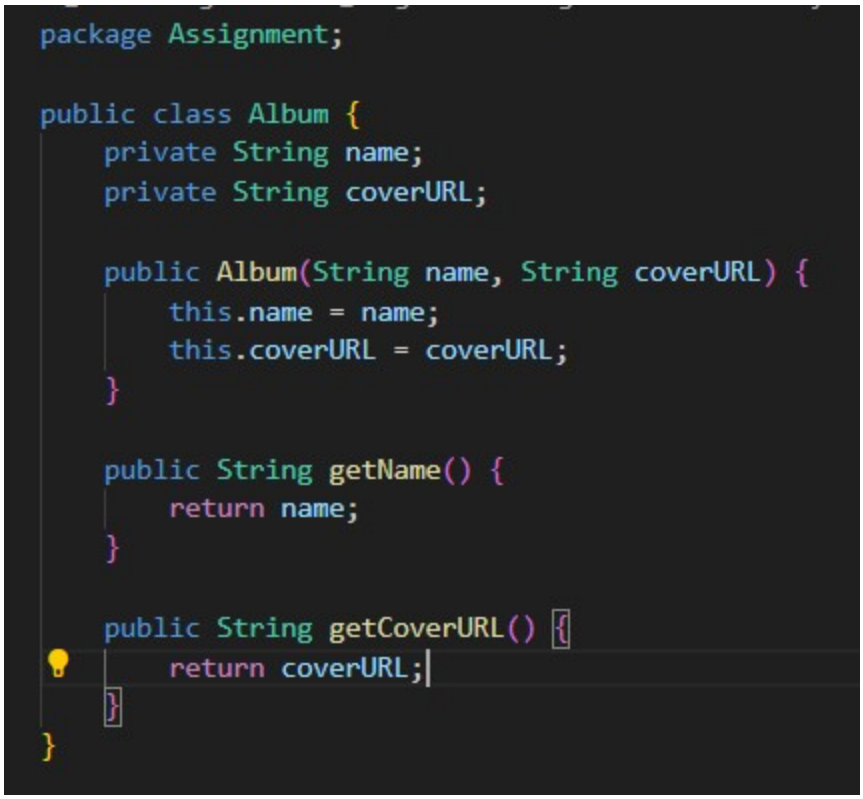
a. God Class: Class Song menangani terlalu banyak tanggung jawab, termasuk informasi album dan artis.
b. Magic Number: Genre direpresentasikan dengan angka (0-7), yang sulit dipahami.
c. Long Method: Fungsi printInfo(int detailLevel) terlalu kompleks dengan banyak if-else.

3. Langkah Refaktorisasi a. Pemisahan Class

- Membuat Class Album untuk menyimpan informasi album.
- Membuat Class Artist untuk menyimpan informasi artis.
- Menggunakan Enum Genre untuk menggantikan angka dalam representasi genre.

b. Implementasi Class Baru

- Class Album.java



```java
package Assignment;

public class Album {
    private String name;
    private String coverURL;

    public Album(String name, String coverURL) {
        this.name = name;
        this.coverURL = coverURL;
    }

    public String getName() {
        return name;
    }

    public String getCoverURL() {
        return coverURL;
    }
}
```

- Class Artist.java

```java
package Assignment;

public class Artist {
    private String name;
    private String alias;
    private String imageURL;

    public Artist(String name, String alias, String imageURL) {
        this.name = name;
        this.alias = alias;
        this.imageURL = imageURL;
    }

    public String getName() {
        return name;
    }

    public String getAlias() {
        return alias;
    }

    public String getImageURL() {
        return imageURL;
    }
}
```

- Enum Genre.java

```java
package Assignment;

public enum Genre {
    UNDEFINED, POP, ROCK, HIPHOP, RNB, JAZZ, INSTRUMENTALS, CLOWNCORE
}
```

c. Perubahan pada Class Song.java

```java
package Assignment;

public class Song {
    private String id;
    private String title;
    private String releaseYear;
    private String musicFileURL;
    private Genre genre;
    private Album album;
    private Artist artist;

    public Song(String id, String title, String releaseYear, String musicFileURL) {
        this.id = id;
        this.title = title;
        this.releaseYear = releaseYear;
        this.musicFileURL = musicFileURL;
        this.genre = Genre.UNDEFINED;
    }

    public void setAlbum(Album album) {
        this.album = album;
    }

    public void setArtist(Artist artist) {
        this.artist = artist;
    }

    public void setGenre(Genre genre) {
        this.genre = genre;
    }

    public void printInfo() {
        System.out.println("Song title: " + title);
        System.out.println("Release year: " + releaseYear);
        System.out.println("Genre: " + genre);
        if (artist != null) {
            System.out.println("Artist: " + artist.getName() + " (aka " + artist.getAlias() + ")");
        }
        if (album != null) {
            System.out.println("Album: " + album.getName());
        }
    }
}
```

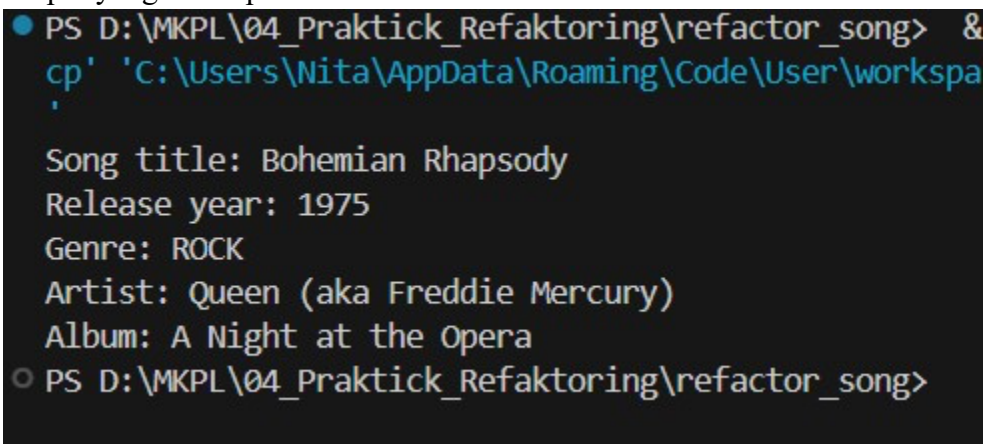
d. Pembuatan Class Main.java untuk Testing

```java
package Assignment;

public class Main {
    public static void main(String[] args) {
        Song song = new Song(id:"1", title:"Bohemian Rhapsody", releaseYear:"1975", musicFileURL:"bohemian.mp3");
        song.setGenre(Genre.ROCK);

        Album album = new Album(name:"A Night at the Opera", coverURL:"cover.jpg");
        song.setAlbum(album);

        Artist artist = new Artist(name:"Queen", alias:"Freddie Mercury", imageURL:"queen.jpg");
        song.setArtist(artist);

        song.printInfo();
    }
}
```

4. Pengujian Kode

Setelah refaktorisasi, kode diuji dengan langkah berikut:

a. Kompilasi semua file Java: javac -d bin -sourcepath src src/Assignment/*.java
b. Jalankan program: java -cp bin Assignment.Main
c. Output yang diharapkan:



d. Github Code:

https://github.com/anderasinggih/MKEPL_Andera-Singgih-Pratama_2211104007/tree/main/04_Prak_Refaktoring/refactor_song/src/Assignment

https://github.com/anderasinggih/MKEPL_Andera-Singgih-Pratama_2211104007.git

5. Kesimpulan

a. Refaktorisasi meningkatkan modularitas dan keterbacaan kode.
b. Menggunakan Enum untuk genre menggantikan magic number.
c. Memisahkan informasi album dan artis ke dalam class terpisah.
d. Metode printInfo() lebih sederhana dan mudah dipahami.

Dengan perubahan ini, kode lebih terstruktur, lebih mudah diuji, dan lebih siap untuk dikembangkan lebih lanjut.