

TUGAS WEEK 4
MANAJEMEN KONFIGURASI
DAN EVOLUSI PERANGKAT LUNAK



Disusun oleh :
Althafia Defiyandrea Laskanadya Wibowo
2211104011

S1 Rekayasa Perangkat Lunak
Fakultas Informatika
Telkom University Purwokerto
2025

1. Identifikasi Bad Smell

- **Large Class**
Class Song memiliki terlalu banyak tanggung jawab (data dan metode terkait lagu, album, dan artis).
- **Primitive Obsession**
Genre direpresentasikan sebagai integer tanpa enum, membuatnya sulit dibaca dan rentan terhadap kesalahan.
- **Long Method (printInfo)**
Method printInfo memiliki banyak kondisi if-else, yang bisa lebih bersih jika menggunakan pendekatan lain.
- **Feature Envy**
Informasi album dan artis disimpan dalam Song, padahal sebaiknya dibuat sebagai class terpisah.

2. Refactoring

- **Primitive Obsession**
Ubah genre dari integer ke enum agar lebih mudah dibaca dan lebih aman.
- **Long Method (printInfo)**
Sederhanakan dengan membuat helper method untuk mencetak informasi lagu, artis, dan album.
- **Encapsulation**
Tambahkan getter dan setter agar lebih terstruktur.
- **Magic Numbers**
Gantilah angka dalam printInfo dengan konstanta agar lebih mudah dipahami.

https://github.com/laskanaa/MKEPL_Tugas_2211104011.git

```
1 package Assignment;
2
3 public class Song {
4
5     private String id;
6     private String title;
7     private String releaseYear;
8     private String musicFileURL;
9     private Genre genre = Genre.UNDEFINED; // Default value
10
11     private String albumName;
12     private String albumCoverURL;
13
14     private String artistName;
15     private String artistAlias;
16     private String artistImageURL;
17
18     // Konstanta untuk level detail
19     private static final int DETAIL_SONG = 0;
20     private static final int DETAIL_ARTIST = 1;
21     private static final int DETAIL_ALBUM = 2;
22     private static final int DETAIL_FULL = 3;
23
24     public enum Genre {
25         UNDEFINED, POP, ROCK, HIP_HOP, RNB, JAZZ, INSTRUMENTALS, CLOWNCORE
26     }
27
28     public Song(String id, String title, String releaseYear, String musicFileURL) {
29         this.id = id;
30         this.title = title;
31         this.releaseYear = releaseYear;
32         this.musicFileURL = musicFileURL;
33     }
34
35     public void setAlbum(String albumName, String albumCoverURL) {
36         this.albumName = albumName;
37         this.albumCoverURL = albumCoverURL;
38     }
39 }
```

```
39
40     public void setArtist(String artistName, String artistAlias, String artistImageUrl) {
41         this.artistName = artistName;
42         this.artistAlias = artistAlias;
43         this.artistImageUrl = artistImageUrl;
44     }
45
46     public void setGenre(Genre genre) {
47         this.genre = genre;
48     }
49
50     public void printInfo(int detailLevel) {
51         System.out.println(getSongInfo());
52
53         if (detailLevel >= DETAIL_ARTIST) {
54             System.out.println(getArtistInfo());
55         }
56         if (detailLevel >= DETAIL_ALBUM) {
57             System.out.println(getAlbumInfo());
58         }
59     }
60
61     private String getSongInfo() {
62         return "Song Title: " + title + "\nRelease Year: " + releaseYear +
63             (genre != Genre.UNDEFINED ? "\nGenre: " + genre : "");
64     }
65
66     private String getArtistInfo() {
67         if (artistName == null || artistName.isEmpty()) return "";
68         return "Artist Name: " + artistName +
69             (!artistAlias.isEmpty() ? "\nAlso known as: " + artistAlias : "");
70     }
71
72     private String getAlbumInfo() {
73         if (albumName == null || albumName.isEmpty()) return "";
74         return "Album Title: " + albumName;
75     }
76 }
77
```