

MINI PROJECT KELOMPOK

LAPORAN ANALISIS STUDIO GHIBLI API



Dosen pengampu :

Bapak Sepyan Purnama Kristanto, S.Kom., M.Kom

Disusun oleh : Kelompok 5

Anggota :

1. Citra Resa Permata (362458302019)
2. Nazwa Ridev Ramadhani (362458302068)
3. Rasya Aprilia (362458302115)
4. Maulina Dwi Wahyunengtias (362458302140)

PROGRAM STUDI

DIV TEKNOLOGI REKAYASA PERANGKAT LUNAK

POLITEKNIK NEGERI BANYUWANGI

2025

Langkah 1.1: Persiapan Kolaborasi dan Analisis

1. DESKRIPSI API

Studio Ghibli API adalah layanan API publik yang menyajikan informasi seputar film-film karya Studio Ghibli. API ini menyediakan data terstruktur dalam format JSON yang mencakup judul, sinopsis, sutradara, tahun rilis, skor penilaian kritikus, hingga tautan ke sumber terkait seperti karakter, lokasi, dan kendaraan. Dengan API ini, kita bisa membuat aplikasi web atau mobile buat nampilin daftar film Ghibli, detail tiap film, atau bahkan mengolah datanya buat analisis.

2. ANALISIS SUMBER DAYA FILMS

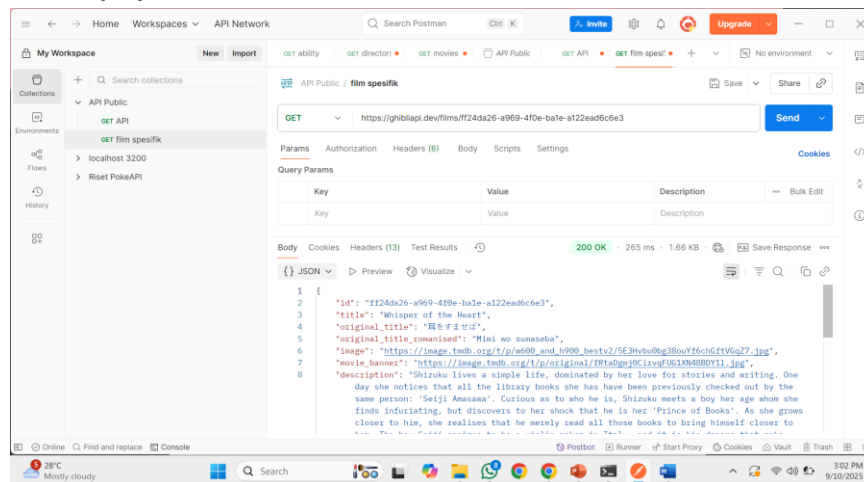
- Jelaskan struktur data JSON dari satu objek film. Sebutkan minimal 5 properti kunci dan tipe datanya (misal: title (String), release date (String)).

Jawab :

5 properti kunci dan tipe data

1. Id (String)
identitas unik film dalam bentuk UUID
Contoh: "ff24da26-a969-4f0e-ba1e-a122ead6c6e3"
2. Title (String)
judul film dalam bahasa utama atau dalam bahasa inggris
Contoh: "Whisper of the Heart"
3. Original_title (String)
Judul asli film dalam bahasa Jepang
Contoh: "耳をすませば"
4. description (String)
Ringkasan cerita atau sipnosis film
Contoh: "Shizuku lives a simple life, dominated by her love for stories and writing..."
5. director (String)
Nama sutradara film.
Contoh: "Yoshifumi Kondō"

- Tempelkan contoh respons JSON (potongan yang relevan saja) dari GET /films/{id}.



3. ENDPOINT YANG DIANILISIS :

- **GET /films: Jelaskan fungsinya.**

Jawab: Fungsi ini mengambil semua data film yang ada di Studio Ghibli yang berisi judul film, sinopsis, sutradara, tahun rilis, dan rating

- **GET /films/{id}: Jelaskan fungsinya**

Jawab : Fungsi Endpoint ini untuk mengambil data detail lengkap dari satu film di Studio Ghibli, berdasarkan ID film-nya.

Langkah 1.2: Konstruksi API Lokal Read-Only

1. Struktur Proyek dan Instalasi

Langkah pertama yang dilakukan menjalankan perintah berikut :

```
D:\miniproject\kel5-api-ulasan-film>npm init -y
Wrote to D:\miniproject\kel5-api-ulasan-film\package.json:

{
  "name": "kel5-api-ulasan-film",
  "version": "1.0.0",
  "description": "\"# Kel5 API Ulasan Film\"",
  "main": "index.js",
  "scripts": {
    "test": "echo \\Error: no test specified\\ && exit 1"
  },
  "repository": {
    "type": "git",
    "url": "git+https://github.com/maulinadwi/kel5-api-ulasan-film.git"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "type": "commonjs",
  "bugs": {
    "url": "https://github.com/maulinadwi/kel5-api-ulasan-film/issues"
  },
  "homepage": "https://github.com/maulinadwi/kel5-api-ulasan-film#readme"
}

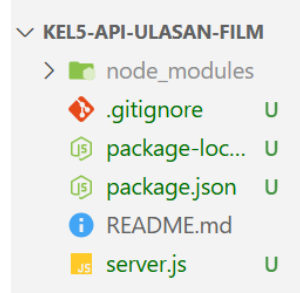
D:\miniproject\kel5-api-ulasan-film>npm install express cors

added 70 packages, and audited 71 packages in 7s

16 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

Lalu membuat file server.js dan .gitignore di visual code

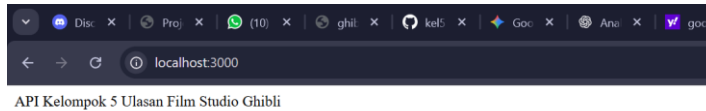


2. Perancangan Data Lokal

- Struktur data untuk ulasan (reviews) di server.js.

```
server.js > [0] port
1  const express = require('express');
2  const cors = require('cors');
3
4  const app = express();
5  const port = 3300;
6
7  // Basis data sementara untuk ulasan
8  let reviews = [
9    {
10     id: 1,
11     film_id: "2baf70d1-42bb-4437-b551-e5fed5a87abe", // Spirited Away
12     user: "Andi",
13     rating: 5,
14     comment: "Film animasi terbaik sepanjang masa!"
15   }
16 ];
```

- Implementasi Server dan Endpoint GET
 - Tulis kode server di server.js yang berjalan di port 3300.



- Implementasikan endpoint berikut:

- **GET /status**

```
app.get('/status', (req, res) => {
  res.send({ message: 'Server is running', status: 'ok' });
});
```

- **GET /reviews**

```
app.get('/reviews', (req, res) => {
  res.json(reviews);
});
```

- **GET /reviews/:id.**

```
app.get('/reviews/:id', (req, res) => {
  const id = parseInt(req.params.id);
  const review = reviews.find(r => r.id === id);
  if (review) {
    res.json(review);
  } else {
    res.status(404).json({ message: 'Review not found' });
  }
});
```

- Setelah setiap endpoint berhasil dibuat dan diuji secara lokal, lakukan commit dan push ke GitHub.

```
C:\Mini project\kel5-api-ulasan-film>git add .
C:\Mini project\kel5-api-ulasan-film>git commit -m "pesan commit"
[main a733f4d] pesan commit
 4 files changed, 938 insertions(+)
 create mode 100644 .gitignore
 create mode 100644 package-lock.json
 create mode 100644 package.json
 create mode 100644 server.js
C:\Mini project\kel5-api-ulasan-film>git push origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (6/6), 8.91 KiB | 2.23 MiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/maulinadwi/kel5-api-ulasan-film.git
 6a1fe1f..a733f4d  main -> main
```

