Ujian Tengah Semester (UTS) Semester Ganjil 20251

Nama Mahasiswa	: SITI MAGFIRATUN WARAHMAH
NIM	: 230104040059
Kelas	: TI23B
Hari/Tanggal	: Senin, 10 November 2025
Program Studi	: Teknologi Informasi
Mata Kuliah	: Web Service Engineering
Dosen Pengampu	: Muhayat, M.IT

A. Soal Ujian

"**Membangun RESTful API dengan CRUD Lengkap dan 7 RESTful Principles**"

B. Deskripsi Singkat

Membangun RESTful API berbasis Express.js untuk satu resource baru, dengan operasi CRUD lengkap, validasi input, serta penerapan 7 Prinsip RESTful API seperti yang sudah dipelajari pada Praktikum 5 & 6.

API harus menampilkan data dalam format JSON, menggunakan status code yang tepat, serta memiliki endpoint dokumentasi (/api/info).

C. Tujuan

1. Mendesain endpoint RESTful untuk resource baru.
2. Mengimplementasikan CRUD lengkap dengan Express.js.
3. Menggunakan metode HTTP dan status code sesuai standar REST.
4. Menerapkan validasi dan error handling yang tepat.
5. Menghasilkan representasi data JSON yang konsisten.
6. Menyusun struktur folder modular dan mudah dibaca.
7. Menerapkan 7 RESTful Principles secara eksplisit.

D. Ketentuan Umum

1. Gunakan Node.js + Express.js.
2. Tidak menggunakan database — cukup data dummy di file .js / .json.
3. Gunakan struktur folder standar:

```
src/
├── app.js
├── routes/
│   └── <resource>.routes.js
├── controllers/
│   └── <resource>.controller.js
├── data/
│   └── <resource>.data.js
└── README.md
```

4. Semua response berbentuk JSON.
5. Gunakan port default 3000.

6. Sertakan endpoint /api/info untuk identitas API.
7. Kode wajib dijalankan dengan npm run dev.

E. 7 Prinsip RESTful yang Wajib Diterapkan

No	Prinsip	Implementasi di Soal
1	Resource-Oriented URI	Gunakan kata benda jamak, contoh /api/books
2	Proper HTTP Methods	GET, POST, PUT, DELETE
3	Stateless Communication	Tidak ada session atau state di server
4	Consistent Status Codes	200, 201, 204, 400, 404, 500
5	JSON Representation	Response rapi dan seragam
6	Validation & Error Handling	Periksa field wajib, kirim error 400
7	Discoverability	Tambahkan /api/info sebagai metadata service

F. Penentuan Resource Berdasarkan Digit Akhir NIM

Digit Akhir NIM	Resource	Deskripsi	Field Utama (wajib ada)
9	movies	Data film	title, genre, year

G. Spesifikasi Endpoint yang Wajib Ada

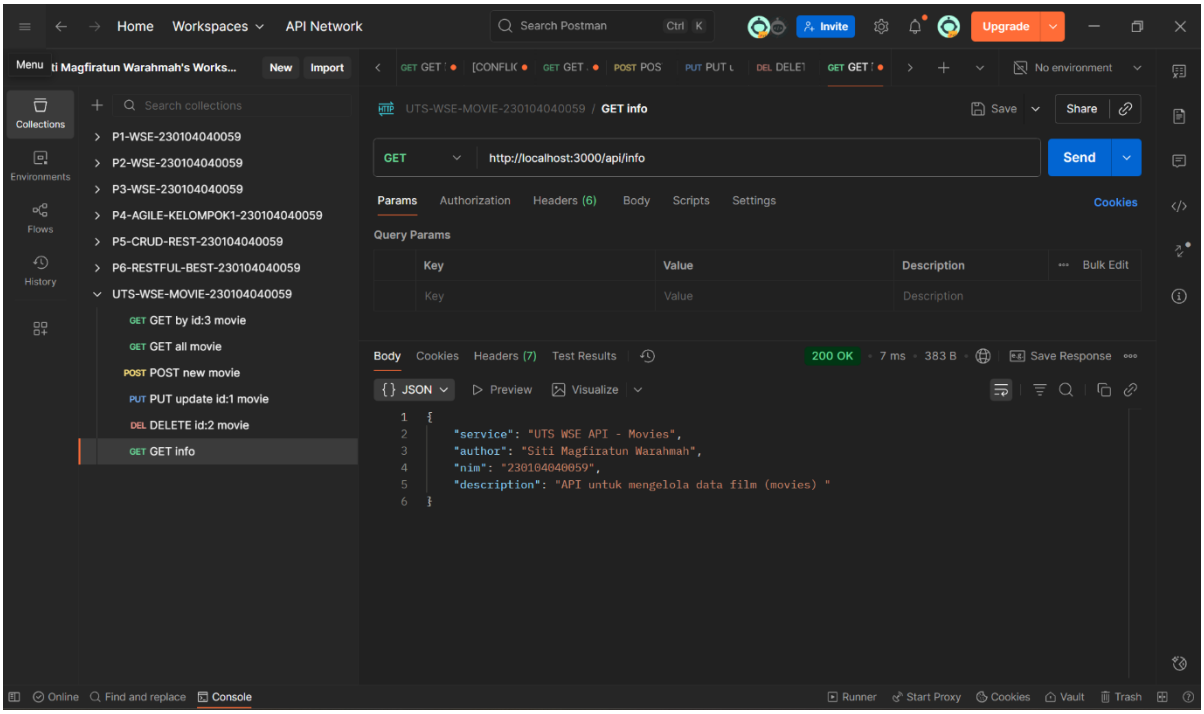
Method	**Endpoint**	**Deskripsi**	**Status Code**
GET	/api/<resource>	Ambil semua data	200
GET	/api/<resource>/:id	Ambil data berdasarkan id	200 / 404
POST	/api/<resource>	Tambah data baru	201 / 400
PUT	/api/<resource>/:id	Update data	200 / 400 / 404
DELETE	/api/<resource>/:id	Hapus data	204 / 404
GET	/api/info	Informasi service	200

H. Contoh Validasi (Wajib Ada)

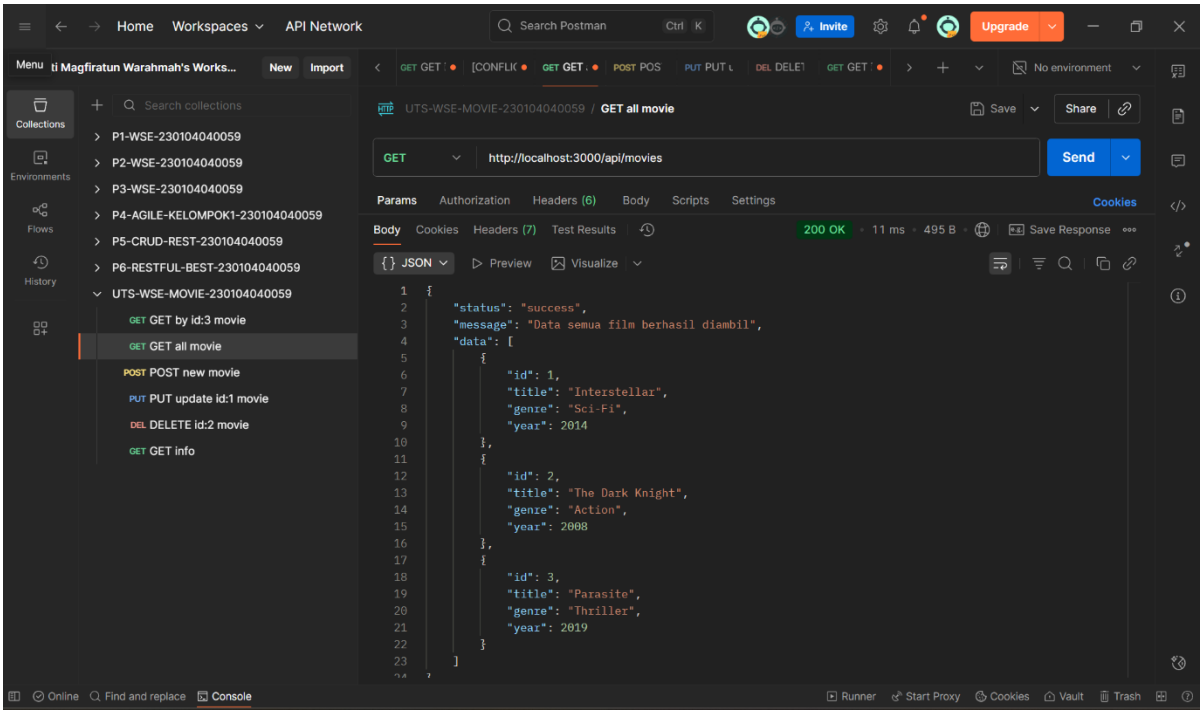
Field	**Syarat**	**Respon Error**
title	Tidak boleh kosong	{ "status": "fail", "message": "Field 'title' wajib diisi" }
author	Tidak boleh kosong	{ "status": "fail", "message": "Field 'author' wajib diisi" }

I. Screenshot Hasil Uji di Postman

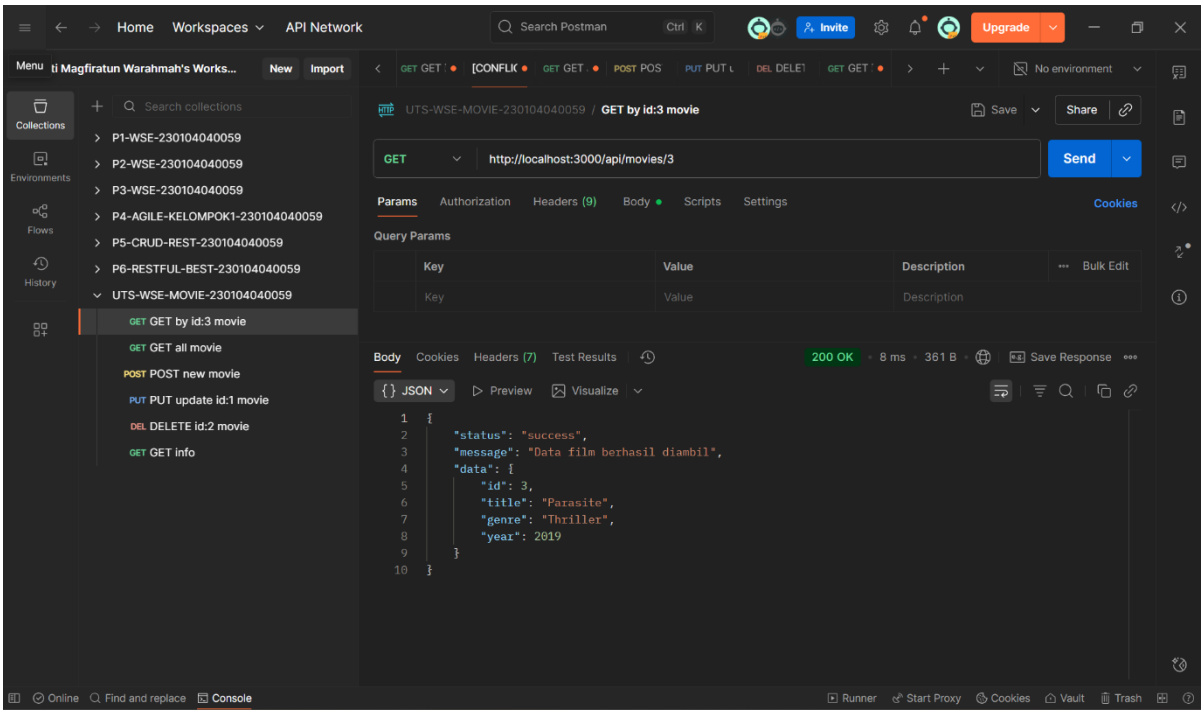
☑ GET /api/info (Berhasil ambil info API) → 200 OK



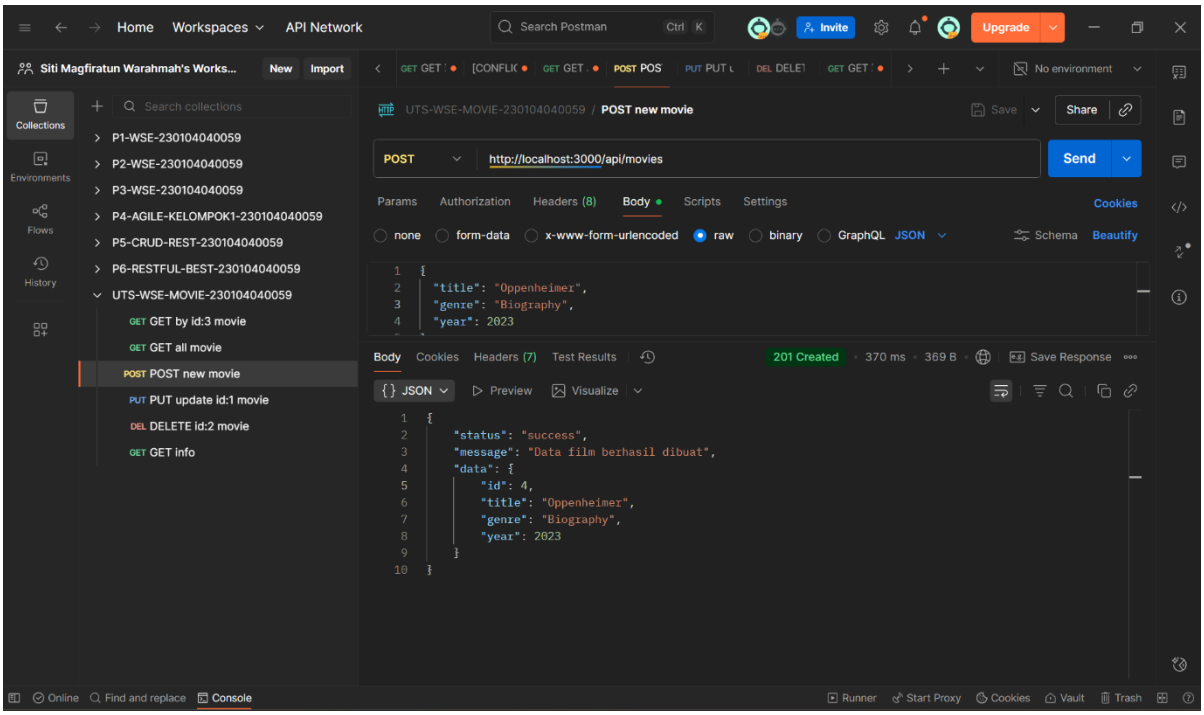
☑ GET /api/movies (Berhasil ambil semua data film) → 200 OK



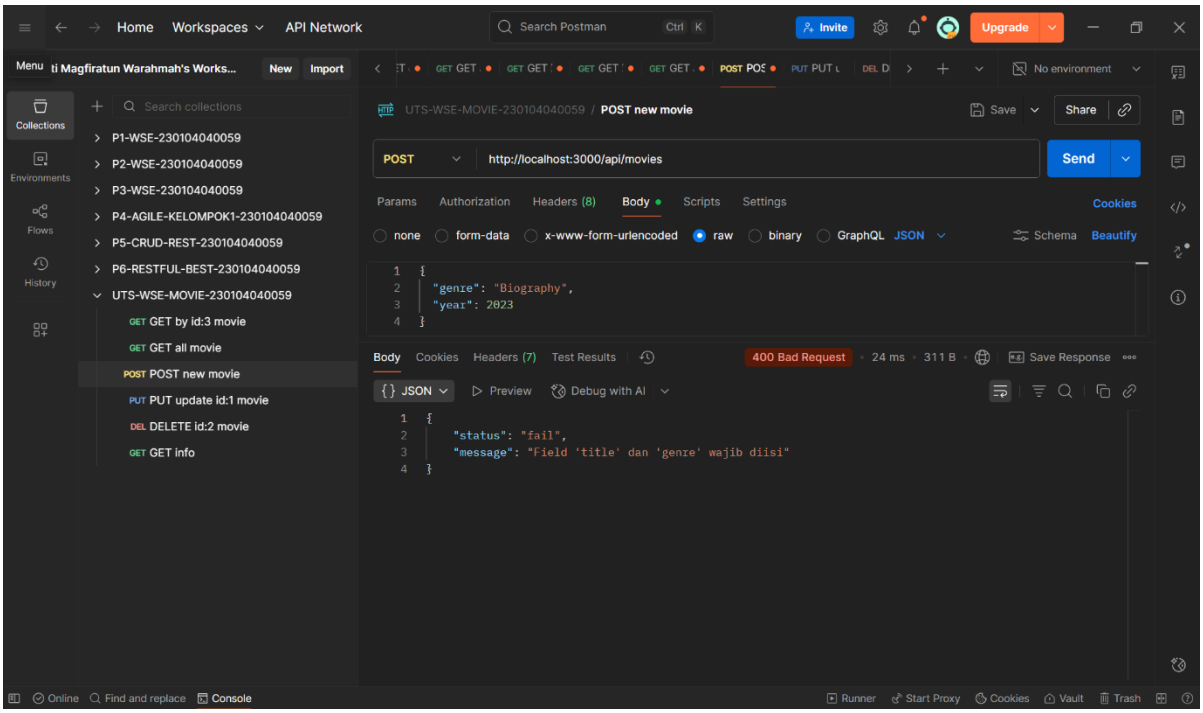
☑ GET /api/movies/3 (Berhasil ambil data film by ID) → 200 OK



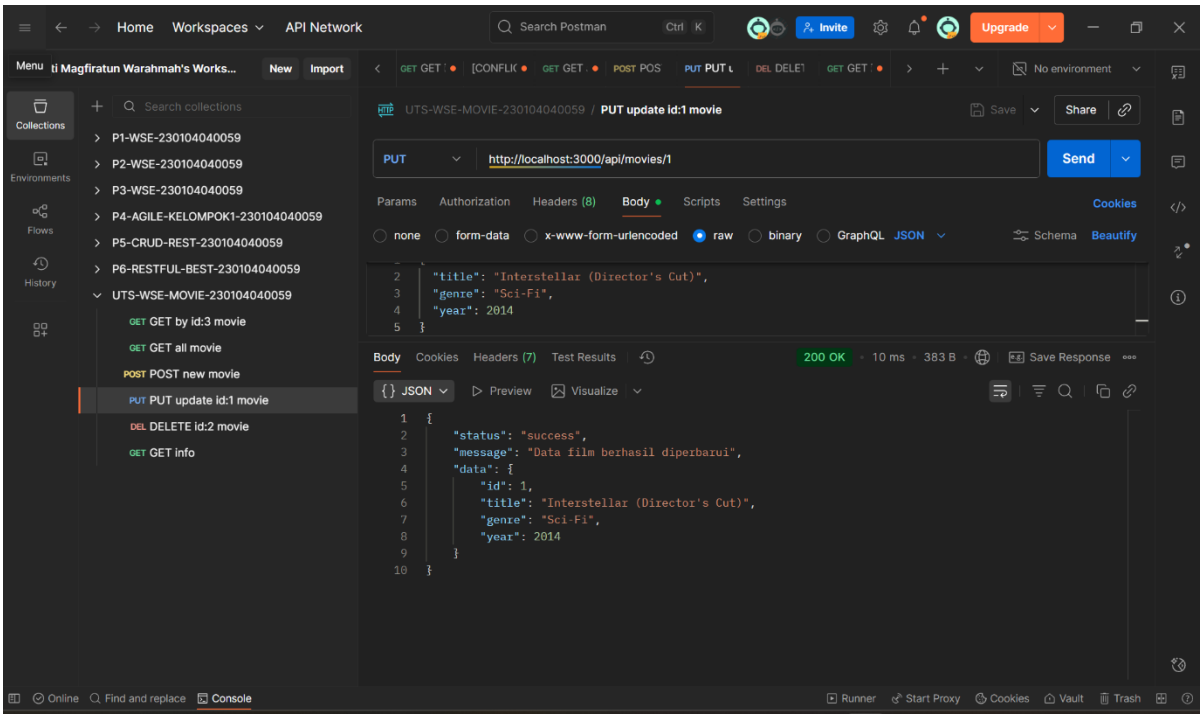
☑ POST /api/movies (Berhasil tambah data film) → 201 Created



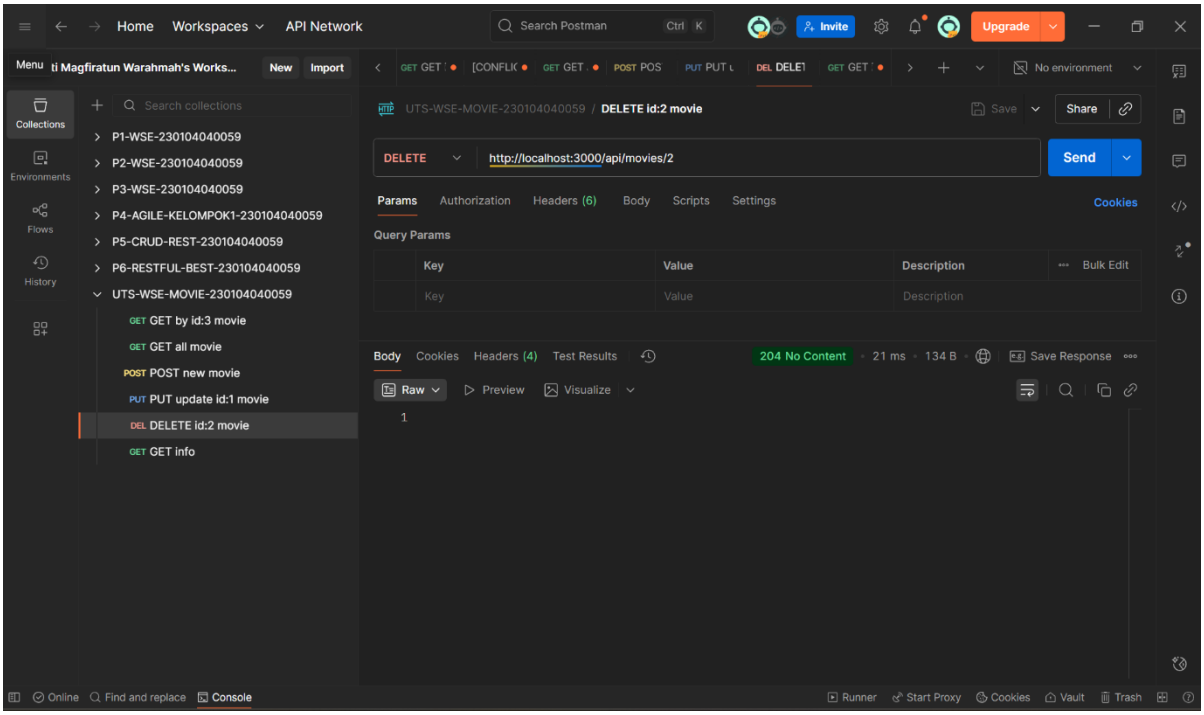
⚠ POST /api/movies (Gagal validasi data) → 400 Bad Request



☑ PUT /api/movies/1 (Berhasil update data film) → 200 OK



☑ DELETE /api/movies/2 (Berhasil hapus data film) → 204 No Content



☑ Checklist Mahasiswa

No	Komponen	Status
1	CRUD lengkap berjalan	☐
2	Validasi input & error handling	☐
3	Status code konsisten	☐
4	Response JSON rapi	☐
5	Endpoint /api/info aktif	☐
6	Struktur folder sesuai template	☐
7	Screenshot hasil uji CRUD	☐
8	Penerapan 7 RESTful Principles	☐