

**TUGAS JURNAL  
KONSTRUKSI PERANGKAT LUNAK**

**MODUL IX**

**API**



**Disusun Oleh :**

Muhammad Rizaldy Akbar / 2211104065

SE06-3

**PROGRAM STUDI S1 SOFTWARE ENGINEERING**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2024**

## TUGAS JURNAL FILM

### A. film.ipynb

```
!pip install fastapi uvicorn nest-asyncio pyngrok

from fastapi import FastAPI, HTTPException
from pydantic import BaseModel
from typing import List
import nest_asyncio
from pyngrok import ngrok, conf
import uvicorn

conf.get_default().auth_token = "2xc0lliv6l0wuwMCPdaV9KYuHJZ_BTjy5b8xShlw4MECV3jY"

app = FastAPI()

class Film(BaseModel):
    title: str
    director: str
    cast: List[str]
    synopsis: str

film_storage = [
    {
        "title": "Inception",
        "director": "Christopher Nolan",
        "cast": ["Leonardo DiCaprio", "Joseph Gordon-Levitt", "Ellen Page"],
        "synopsis": "Seorang pencuri yang memiliki kemampuan untuk masuk ke dalam mimpi orang lain ditugaskan untuk menanamkan ide ke dalam pikiran targetnya."
    },
    {
        "title": "Interstellar",
        "director": "Christopher Nolan",
        "cast": ["Matthew McConaughey", "Anne Hathaway", "Jessica Chastain"],
        "synopsis": "Sebuah tim penjelajah ruang angkasa melakukan perjalanan melalui wormhole untuk mencari planet baru yang layak huni bagi umat manusia."
    },
    {
        "title": "Parasite",
        "director": "Bong Joon-ho",
        "cast": ["Song Kang-ho", "Lee Sun-kyun", "Cho Yeo-jeong"],
        "synopsis": "Keluarga miskin di Korea menyusup ke kehidupan keluarga kaya, mengakibatkan serangkaian kejadian tak terduga yang penuh ketegangan dan satire sosial."
    }
]

@app.get("/api/Movies")
def tampilkan_semua_film():
    return film_storage

@app.get("/api/Movies/{film_id}")
def ambil_film_berdasarkan_id(film_id: int):
    if 0 < film_id < len(film_storage):
        return film_storage[film_id]
    return {"status": "gagal", "pesan": "Film tidak ditemukan"}

@app.post("/api/Movies")
def tambah_film(film: Film):
    film_storage.append(film.dict())
    return {"status": "berhasil", "pesan": "Film berhasil ditambahkan"}

@app.delete("/api/Movies/{film_id}")
def hapus_film(film_id: int):
    if 0 < film_id < len(film_storage):
        data_dihapus = film_storage.pop(film_id)
        return {"status": "berhasil", "pesan": "Film dihapus", "film": data_dihapus}
    return {"status": "gagal", "pesan": "Film tidak ditemukan"}

nest_asyncio.apply()

tunnel_url = ngrok.connect(8000)
print("@ Swagger UI:", tunnel_url.public_url + "/docs")
print("@ API Endpoint:", tunnel_url.public_url)
```

## B. Output

**FastAPI** 0.1.0 OAS 3.1  
/openapi.json

**default**

- GET** `/api/Movies` Tampilkan Semua Film
- POST** `/api/Movies` Tambah Film
- GET** `/api/Movies/{film_id}` Ambil Film Berdasarkan Id
- DELETE** `/api/Movies/{film_id}` Hapus Film

**Schemas**

- Film** > Expand all object
- HTTPValidationError** > Expand all object
- ValidationError** > Expand all object

## C. Get Film

**Request URL**  
`https://e80b-34-106-203-137.ngrok-free.app/api/Movies`

**Server response**

**Code** **Details**

200

**Response body**

```
[
  {
    "title": "Inception",
    "director": "Christopher Nolan",
    "cast": [
      "Leonardo DiCaprio",
      "Joseph Gordon-Levitt",
      "Ellen Page"
    ],
    "synopsis": "Seorang pencuri yang memiliki kemampuan untuk masuk ke dalam mimpi orang lain ditugaskan untuk menanamkan ide ke dalam pikiran targetnya."
  },
  {
    "title": "Interstellar",
    "director": "Christopher Nolan",
    "cast": [
      "Matthew McConaughey",
      "Anne Hathaway",
      "Jessica Chastain"
    ],
    "synopsis": "Sebuah tim penjelajah ruang angkasa melakukan perjalanan melalui wormhole untuk mencari planet baru yang layak huni bagi umat manusia."
  },
  {
    "title": "Parasite",
    "director": "Bong Joon-ho",
    "cast": [
      "Song Kang-ho",
      "Lee Sun-kyun",
      "Go Yoon-jung"
    ],
    "synopsis": "Sebuah keluarga miskin yang berakrabib dengan keluarga kaya akhirnya berakrabib dengan keluarga kaya tersebut."
  }
]
```

**Response headers**

```
content-length: 805
content-type: application/json
date: Fri, 30 May 2025 13:49:12 GMT
ngrok-agent-ips: 34.106.203.137
server: uvicorn
```

## D. Post Film

Curl

```
curl -X 'POST' \
  'https://e80b-34-106-203-137.ngrok-free.app/api/Movies' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "title": "string",
    "director": "string",
    "cast": [
      "string"
    ],
    "synopsis": "string"
  }'
```

Request URL

https://e80b-34-106-203-137.ngrok-free.app/api/Movies

Server response

Code	Details
200	<div><div>Response body</div><div><pre>{   "status": "berhasil",   "pesan": "Film berhasil ditambahkan" }</pre></div><div><div>Download</div></div></div> <div><div>Response headers</div><div><pre>content-length: 57 content-type: application/json date: Fri, 30 May 2025 13:52:50 GMT ngrok-agent-ips: 34.106.203.137 server: uvicorn</pre></div></div>

## E. Get Film by ID

film\_id \* required

integer (path)

2

Execute

Clear

Responses

Curl

```
curl -X 'GET' \
  'https://e80b-34-106-203-137.ngrok-free.app/api/Movies/2' \
  -H 'accept: application/json'
```

Request URL

https://e80b-34-106-203-137.ngrok-free.app/api/Movies/2

Server response

Code	Details
200	<div><div>Response body</div><div><pre>{   "title": "Parasite",   "director": "Bong Joon-ho",   "cast": [     "Song Kang-ho",     "Lee Sun-kyun",     "Cho Yeo-jeong"   ],   "synopsis": "Keluarga miskin di Korea menyusup ke kehidupan keluarga kaya, mengakibatkan serangkaian kejadian tak terduga yang penuh ketegangan dan satire sosial." }</pre></div><div><div>Download</div></div></div> <div><div>Response headers</div><div><pre>content-length: 264 content-type: application/json date: Fri, 30 May 2025 13:54:16 GMT ngrok-agent-ips: 34.106.203.137 server: uvicorn</pre></div></div>

# F. Delete Film

film\_id required

integer

(path)

2

Execute

Clear

Responses

Curl

```
curl -X 'DELETE' \
  'https://e80b-34-106-203-137.ngrok-free.app/api/Movies/2' \
  -H 'accept: application/json'
```

Request URL

https://e80b-34-106-203-137.ngrok-free.app/api/Movies/2

Server response

Code

Details

200

Response body

```
{
  "status": "berhasil",
  "pesan": "film dihapus",
  "film": {
    "title": "string",
    "director": "string",
    "cast": [
      "string"
    ],
    "synopsis": "string"
  }
}
```

Download

Response headers

```
content-length: 128
content-type: application/json
date: Fri, 30 May 2025 13:57:15 GMT
ngrok-agent-ips: 34.106.203.137
server: unicorn
```

# Tugas Jurnal Mahasiswa

## A. Mahasiswa.ipynb

```
!pip install fastapi uvicorn nest-asyncio pyngrok

from fastapi import FastAPI, HTTPException
from pydantic import BaseModel
from typing import List
import nest_asyncio
from pyngrok import ngrok, conf
import uvicorn

...

[ ] app = FastAPI()
conf.get_default().auth_token = "2xe1li1v8loewvHOM2wV9CvH1Z_GUTy2b8n5t3uA48EV7jY"

[ ] class Mahasiswa(BaseModel):
    name: str
    nim: str
    course: List[str]
    year: int

[ ] mahasiswa_list = [
    {
        "name": "Muhammad Hadriq Subono",
        "nim": "1101220003",
        "course": ["Pemrograman Web", "Kecerdasan Buatan"],
        "year": 2022
    },
    {
        "name": "Andi Setiawan",
        "nim": "1101220003",
        "course": ["Basis Data", "Jaringan Komputer"],
        "year": 2022
    },
    {
        "name": "Siti Murtaliza",
        "nim": "1101220003",
        "course": ["Struktur Data", "Etika Profesi"],
        "year": 2022
    }
]

[ ] @app.get("/api/Mahasiswa")
def get_mahasiswa():
    return mahasiswa_list

@app.get("/api/Mahasiswa/{id}")
def get_mahasiswa_by_id(id: int):
    if 0 < id < len(mahasiswa_list):
        return mahasiswa_list[id]
    return {"error": "Mahasiswa tidak ditemukan"}

@app.post("/api/Mahasiswa")
def add_mahasiswa(mha: Mahasiswa):
    mahasiswa_list.append(mha.dict())
    return {"pesan": "Mahasiswa berhasil ditambahkan"}

@app.delete("/api/Mahasiswa/{id}")
def delete_mahasiswa(id: int):
    if 0 < id < len(mahasiswa_list):
        deleted = mahasiswa_list.pop(id)
        return {"pesan": "Mahasiswa berhasil dihapus", "data": deleted}
    return {"error": "Mahasiswa tidak ditemukan"}

[ ] nest_asyncio.apply()
tunnel = ngrok.connect(8000)
print(f"Swagger Docs:", tunnel.public_url + "/docs")
print(f"API Root:", tunnel.public_url)
uvicorn.run(app, port=8000)
```

## B. Output

**FastAPI** 0.1.0 OAS 3.1  
/openapi.json

default ^

GET	/api/Mahasiswa	Get Mahasiswa	▼
POST	/api/Mahasiswa	Add Mahasiswa	▼
GET	/api/Mahasiswa/{id}	Get Mahasiswa By Id	▼
DELETE	/api/Mahasiswa/{id}	Delete Mahasiswa	▼

Schemas ^

HTTPValidationError > Expand all object

Mahasiswa > Expand all object

ValidationError > Expand all object

## C. Get mahasiswa

Curl

```
curl -X 'GET' \
  'https://dccb-35-233-195-161.ngrok-free.app/api/Mahasiswa' \
  -H 'accept: application/json'
```

Request URL

https://dccb-35-233-195-161.ngrok-free.app/api/Mahasiswa

Server response

Code Details

200

Response body

```
{
  "name": "Muhammad Hadziq Subono",
  "nim": "1501220001",
  "course": [
    "Pemrograman Web",
    "Kecerdasan Buatan"
  ],
  "year": 2022
},
{
  "name": "Andi Setiawan",
  "nim": "1501220002",
  "course": [
    "Basis Data",
    "Jaringan Komputer"
  ],
  "year": 2022
},
{
  "name": "Siti Nurhaliza",
  "nim": "1501220003",
  "course": [
    "Struktur Data",
    "Etika Profesi"
  ],
  "year": 2022
}
]
```

Download

## D. Post mahasiswa

Responses

Curl

```
curl -X 'POST' \
  'https://dccb-35-233-195-161.ngrok-free.app/api/Mahasiswa' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "name": "string",
    "nim": "string",
    "course": [
      "string"
    ],
    "year": 0
  }'
```

Request URL

https://dccb-35-233-195-161.ngrok-free.app/api/Mahasiswa

Server response

Code	Details
200	<p>Response body</p> <pre>{   "pesan": "Mahasiswa berhasil ditambahkan" }</pre> <p>Response headers</p> <pre>content-length: 42 content-type: application/json date: Fri, 30 May 2025 14:03:42 GMT ngrok-agent-ips: 35.233.195.161 server: uvicorn</pre>

## E. Get mahasiswa by ID

Name Description

id required

integer

(path)

2

Execute Clear

Responses

Curl

```
curl -X 'GET' \
  'https://dccb-35-233-195-161.ngrok-free.app/api/Mahasiswa/2' \
  -H 'accept: application/json'
```

Request URL

https://dccb-35-233-195-161.ngrok-free.app/api/Mahasiswa/2

Server response

Code	Details
200	<p>Response body</p> <pre>{   "name": "Siti Nurhaliza",   "nim": "1101220003",   "course": [     "Struktur Data",     "Etika Profesi"   ],   "year": 2022 }</pre>



## F. Delete mahasiswa

Name	Description
id <small>required</small>	<input type="text" value="2"/>
integer	
(path)	

Execute

Clear

Responses

Curl

```
curl -X 'DELETE' \
'https://dccb-35-233-195-161.ngrok-free.app/api/Mahasiswa/2' \
-H 'accept: application/json'
```

Request URL

```
https://dccb-35-233-195-161.ngrok-free.app/api/Mahasiswa/2
```

Server response

Code	Details
200	<div><div>Response body</div><div><pre>{   "pesan": "Mahasiswa berhasil dihapus",   "data": {     "name": "string",     "nim": "string",     "course": [       "string"     ],     "year": 0   } }</pre></div><div><div>Download</div></div></div>