

Laporan Praktikum
Mata Kuliah Pemrograman Web & Pemrograman Berorientasi Objek



Dosen Pengampu:

Willdan Aprizal Arifin, S.Pd., M.Kom.

Disusun Oleh :

Silvia Isti Lestary (2311883)

Novia Ramadhani (2305968)

PROGRAM STUDI SISTEM INFORMASI KELAUTAN
UNIVERSITAS PENDIDIKAN INDONESIA

2024

Link GitHub

https://github.com/silviaalestary/miniproject_1

PENDAHULUAN

API (Application Programming Interface) adalah sekumpulan aturan, protokol, dan alat yang memungkinkan satu aplikasi atau sistem untuk berkomunikasi dengan aplikasi atau sistem lainnya. API berperan sebagai jembatan yang memungkinkan pertukaran data dan instruksi antara berbagai komponen perangkat lunak, meskipun keduanya dikembangkan dengan teknologi yang berbeda. Dalam pengembangan perangkat lunak modern, API menjadi fondasi penting karena memungkinkan integrasi yang lebih cepat, fleksibel, dan efisien. Misalnya, API memungkinkan aplikasi e-commerce untuk memproses pembayaran menggunakan layanan pihak ketiga seperti PayPal atau sistem perbankan. Selain itu, API juga digunakan untuk mengakses layanan cloud, sistem otentikasi, atau integrasi dengan platform media sosial.

Penggunaan API tidak hanya meningkatkan efisiensi dalam pengembangan perangkat lunak tetapi juga memungkinkan perusahaan untuk membuka akses ke layanan mereka kepada pengembang pihak ketiga, yang dapat memperluas ekosistem aplikasi. Salah satu jenis API yang paling populer adalah RESTful API, yang menggunakan protokol HTTP untuk mengirim dan menerima data dalam format seperti JSON atau XML. Dengan API, pengembang tidak perlu membangun semuanya dari awal; mereka dapat memanfaatkan layanan yang sudah ada, meningkatkan produktivitas, dan memastikan komunikasi antar sistem berjalan dengan lancar. Oleh karena itu, API menjadi komponen krusial dalam dunia teknologi dan pengembangan aplikasi saat ini.

➤ API

```
POST http://localhost:3000/auth/register
Content-Type: application/json

{
  "username": "admin",
  "password": "admin123",
  "role": "admin"
}

###
POST http://localhost:3000/auth/login
Content-Type: application/json

{
  "username": "admin",
  "password": "admin123"
}

###
POST http://localhost:3000/kapal
Headers:
Authorization: Bearer
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6MSwidXNlcm5hbWUiOiJhZG1pbiIsInJvbmUiOiJhZG1pbiIsIm1hdCI6MTczNTMwODU5NiwiZXhwIjoxNzM1Mzk0OTk2fQ.V7_qjYH7vQEBYsyK26XTNgvb0msTv6ex-5-THjd5o88
Content-Type: application/json

{
  "nama_kapal": "Laut Biru",
  "jenis_kapal": "Penangkap Ikan",
  "kapasitas_muatan": 5000
}

###
GET http://localhost:3000/kapal
Headers:
Authorization: Bearer
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6MSwidXNlcm5hbWUiOiJhZG1pbiIsInJvbmUiOiJhZG1pbiIsIm1hdCI6MTczNTMwODU5NiwiZXhwIjoxNzM1Mzk0OTk2fQ.V7_qjYH7vQEBYsyK26XTNgvb0msTv6ex-5-THjd5o88

###
PUT http://localhost:3000/kapal/1
Headers:
Authorization: Bearer
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6MSwidXNlcm5hbWUiOiJhZG1pbiIsInJvbmUiOiJhZG1pbiIsIm1hdCI6MTczNTMwODU5NiwiZXhwIjoxNzM1Mzk0OTk2fQ.V7_qjYH7vQEBYsyK26XTNgvb0msTv6ex-5-THjd5o88
```

```

GUiOiJhZG1pbiIsIm1hdCI6MTczNTMwODU5NiwiZXhwIjoxNzM1Mzk0Tk2fQ.V7_qjYH7vQEBYsyK
26XTNgbv0msTv6ex-5-THjd5o88
Content-Type: application/json

{
  "nama_kapal": "Laut Biru Update",
  "jenis_kapal": "Penangkap Ikan",
  "kapasitas_muatan": 6000
}

###
DELETE http://localhost:3000/kapal/1
Headers:
Authorization: Bearer
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6MSwidXNlcm5hbWUiOiJhZG1pbiIsInJvbiI6MTczNTMwODU5NiwiZXhwIjoxNzM1Mzk0Tk2fQ.V7_qjYH7vQEBYsyK
26XTNgbv0msTv6ex-5-THjd5o88

###
POST http://localhost:3000/auth/register
Headers:
Authorization: Bearer
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6MSwidXNlcm5hbWUiOiJhZG1pbiIsInJvbiI6MTczNTMwODU5NiwiZXhwIjoxNzM1Mzk0Tk2fQ.V7_qjYH7vQEBYsyK
26XTNgbv0msTv6ex-5-THjd5o88
Content-Type: application/json

{
  "username": "user1",
  "password": "user123",
  "role": "user"
}

```

KESIMPULAN

Keseluruhan kode API yang disajikan menggambarkan sistem yang terstruktur untuk manajemen autentikasi pengguna dan operasi CRUD pada data kapal menggunakan protokol HTTP. API ini dirancang untuk memastikan keamanan dan akses kontrol yang ketat melalui penggunaan JWT (JSON Web Token) sebagai mekanisme otorisasi. Proses dimulai dengan endpoint `/auth/register` untuk pendaftaran pengguna baru dengan menentukan `'username'`, `'password'`, dan `'role'` (admin atau user), diikuti dengan endpoint `/auth/login` untuk autentikasi dan mendapatkan token JWT yang digunakan sebagai kunci akses untuk endpoint lainnya.

Manajemen data kapal difasilitasi oleh endpoint `/kapal` yang mendukung operasi CRUD: `'POST'` untuk menambahkan data kapal baru, `'GET'` untuk membaca data kapal, `'PUT'` untuk memperbarui data kapal berdasarkan ID, dan `'DELETE'` untuk menghapus data kapal berdasarkan ID tertentu. Setiap operasi ini memerlukan token otorisasi yang valid untuk

memastikan hanya pengguna dengan hak akses yang sesuai yang dapat melakukan manipulasi data. Selain itu, endpoint tambahan `/auth/register` dengan token otorisasi memungkinkan admin untuk menambahkan pengguna baru dari dalam sistem.