

:

REST API (Konsep & Implementasi)

Bahasan: Routing API & JSON Response

Apa itu API?

API = Antarmuka yang memungkinkan sistem saling berkomunikasi.

Terdiri dari: endpoint, request, response.

API adalah jembatan komunikasi antara client (UI) dan server (backend).

Apa itu REST API?

REST = Representational State Transfer

Ciri: stateless, resource-based, menggunakan HTTP methods.

REST adalah arsitektur standar untuk membuat API modern yang mudah diintegrasikan.

KONSEP DASAR REST API

Resource dalam REST API

Semua data dianggap sebagai *resource*, contoh:

- /users
- /products
- /orders

REST API mengakses data sebagai objek tunggal atau koleksi.

HTTP Methods

- **GET** → membaca data
- **POST** → menambah data
- **PUT/PATCH** → memperbarui data
- **DELETE** → menghapus data

Method ini mewakili operasi CRUD dalam API.

Struktur Request

Isi Slide:

Request terdiri dari:

- URL endpoint
- HTTP method
- Header
- Body (untuk POST/PUT)

Client mengirimkan berbagai informasi agar server memproses permintaan.

Struktur Response

Isi Slide:

- Status code
- Header
- Body (biasanya JSON)

Response mengembalikan hasil pemrosesan server.

ROUTING DALAM API

Apa itu Routing API?

Routing = proses memetakan URL ke fungsi/aksi tertentu.

Routing menentukan endpoint mana yang dipanggil dan apa yang dikerjakan server.

Contoh Routing API (Laravel)

```
Route::get('/users', [UserController::class, 'index']);
```

Route di atas menangani request GET ke /users.

Contoh Routing API (Express.js)

```
app.get('/users', (req, res) => {  
  res.send("All users");  
});
```

Express menggunakan pendekatan function-based routing yang sederhana.

Naming & Struktur Endpoint

Gunakan pola konsisten:

- /users
- /users/{id}
- /products/{id}/reviews

Standarisasi naming membuat API lebih mudah dipahami dan digunakan.

Status Code pada API

- 200 OK
- 201 Created
- 400 Bad Request
- 401 Unauthorized
- 404 Not Found
- 500 Server Error

Status code membantu client mengetahui hasil operasi.

JSON RESPONSE

Slide 14 — Mengapa JSON?

JSON adalah format data ringan, mudah diparse, dan kompatibel lintas bahasa.

Menjadi standar response API karena efisiensi dan fleksibilitasnya.

Contoh JSON Response

```
{  
  "id": 1,  
  "name": "John Doe"  
}
```

JSON berisi pasangan key–value yang mewakili data.

JSON Response di Framework (Laravel)

```
return response()->json([  
    'status' => 'success',  
    'data' => $user  
]);
```

Laravel menyediakan helper bawaan untuk format JSON.

JSON Response di Express.js

```
res.json({  
    status: "success",  
    data: users  
});
```

Express menyediakan fungsi `res.json()` sebagai pengirim data standar JSON.

PRAKTIK DAN STRUKTUR API

Best Practice dalam Routing API

- Gunakan versi API (v1, v2)
- Gunakan resource-based naming
- Jangan mencampur endpoint web & API
- Sertakan response standar (status, message, data)

Best practice menjaga API tetap rapi dan mudah dikelola.

Contoh Standarisasi Format JSON

```
{  
    "success": true,  
    "message": "Data retrieved",  
    "data": []
```

```
}
```

Format seragam memudahkan debugging & integrasi front-end.

Kesimpulan & Arah Materi Berikutnya

- REST API adalah standar komunikasi modern
- Routing mengatur endpoint API
- JSON adalah format utama pertukaran data
- Implementasi API akan dilanjutkan ke CRUD API