“Autentikasi dan Otorisasi”
Mata Kuliah: Pemrograman Web 2

Pengertian Autentikasi

Autentikasi = proses memverifikasi identitas pengguna.
Contoh: login, token, session.

Penjelasan:
Autentikasi memastikan “siapa pengguna”.

Pengertian Otorisasi

Otorisasi = menentukan apa saja yang boleh diakses pengguna.
Contoh: admin hanya mengakses halaman tertentu.

Penjelasan:
Otorisasi memastikan “apa yang boleh dilakukan pengguna”.

Alur Autentikasi dalam Aplikasi Web

1. User mengirim kredensial
2. Server mengecek data
3. Sistem membuat session/token
4. User mendapat akses ke halaman tertentu

Penjelasan:
Menjelaskan bagaimana proses login bekerja di sisi backend.

Autentikasi Berbasis Session vs Token

- **Session (PHP/Laravel/CI)**
  Data login disimpan di server
- **Token (JWT – Express.js)**
  Data login disimpan sebagai token di client

Penjelasan:
Framework berbeda, mekanismenya mirip namun implementasi berbeda.

MPLEMENTASI AUTENTIKASI

Form Login

Isi Slide (contoh generik):

```
<form method="POST">
  <input type="email" placeholder="Email">
  <input type="password" placeholder="Password">
  <button>Login</button>
</form>
```

Penjelasan:
Form login merupakan gerbang awal autentikasi.

Validasi Login

- Email wajib format valid
- Password wajib
- Cek kecocokan dengan database

Penjelasan:
Autentikasi harus aman sejak input.

Proses Login (Backend)

1. Ambil email & password
2. Cari user di database
3. Cocokkan password (hash)
4. Simpan session/token
5. Redirect ke halaman dashboard

Penjelasan:
Alur standar untuk autentikasi aman.

Logout

- Hapus session / token
- Redirect ke halaman login

Penjelasan:
Mencegah akses tanpa autentikasi setelah logout.

IMPLEMENTASI OTORISASI (ROLE & PERMISSION)

Pengertian Role

Role memberi level akses:

- Admin
- User
- Editor

Penjelasan:
Role memudahkan pembagian hak akses.

Role-Based Access Control (RBAC)

Konsep RBAC:

- Pengguna → Role → Hak Akses

Penjelasan:
Konsep ini paling sering dipakai dalam aplikasi web.

Middleware / Guard

Middleware digunakan untuk:

- Mengunci halaman tertentu
- Cek user sudah login
- Cek role user

Penjelasan:
Digunakan di Laravel, CodeIgniter, Express.js.

Contoh Middleware Konseptual

Jika user **tidak login** → redirect ke login
Jika user **bukan admin** → blok akses

Penjelasan:
Intinya: middleware sebagai gerbang pengaman.

STUDI KASUS & BEST PRACTICES

Studi Kasus Aplikasi

Aplikasi dengan role:

- **Admin**: kelola user & data
- **User**: akses fitur standar

Penjelasan:
Contoh nyata yang mudah diterapkan mahasiswa.

Contoh Flow Autentikasi & Otorisasi

Login → Session → Dashboard → Cek Role → Akses Halaman

Penjelasan:
Menunjukkan hubungan autentikasi + otorisasi dalam satu alur.

Konsep Keamanan Penting

- Hashing password (bcrypt)
- CSRF protection
- Hindari menyimpan password teks asli
- Batasi percobaan login (brute force protection)

Penjelasan:
Mahasiswa perlu memahami aspek keamanan dasar.

PENUTUP

Ringkasan Materi

- Autentikasi = verifikasi identitas
- Otorisasi = pembatasan akses
- Digunakan bersama untuk keamanan aplikasi
- Implementasi umum: session, token, middleware, RBAC

Penjelasan:
Merangkum poin utama sebelum diskusi atau latihan.

Tugas Praktik

Buat sistem login + role sederhana:

- Halaman login
- Dashboard user
- Dashboard admin (hanya admin)
- Middleware / guard untuk proteksi halaman

Penjelasan:
Tugas ini melatih kemampuan mahasiswa menerapkan autentikasi dan otorisasi langsung pada proyek.