

Merancang Database Terintegrasi dengan Web

Bagian 1 – Pengantar Database dalam Web Development

Slide 1 – Judul

Isi Slide:

Pemrograman Web 2

Sub-CPMK: Merancang Database Terintegrasi dengan Web

Pokok Bahasan: ERD, Relasi Tabel, Integrasi dengan ORM

Penjelasan:

Perkenalkan konteks: bagaimana database menjadi bagian penting dalam aplikasi web.

Visual: Ilustrasi client–server dengan simbol database di sisi backend.

Slide 2 – Tujuan Pembelajaran

Isi Slide:

- Memahami konsep perancangan basis data web
- Menyusun ERD untuk sistem web
- Membuat relasi antar tabel
- Mengintegrasikan database dengan framework melalui ORM

Penjelasan:

Sampaikan kompetensi yang akan dicapai setelah sesi pembelajaran ini.

Visual: checklist + ikon database.

Slide 3 – Peran Database dalam Web

Isi Slide:

- Menyimpan data dinamis (user, produk, transaksi)
- Mendukung autentikasi dan manajemen data
- Berinteraksi melalui *server-side scripting*

Penjelasan:

Jelaskan alur umum: *client* → *server* → *database*.

Visual: diagram 3 lapis (frontend–backend–database).

■ Bagian 2 – Entity Relationship Diagram (ERD)

Slide 4 – Konsep Dasar ERD

Isi Slide:

- Entity: objek nyata (User, Produk, Transaksi)
- Attribute: properti tiap entitas
- Relationship: hubungan antar entitas

Penjelasan:

Bahas konsep model data konseptual yang mendasari struktur tabel database.

Visual: diagram sederhana entitas “User–Order–Product”.

Slide 5 – Simbol ERD

Isi Slide:

Simbol	Arti
Persegi	Entity
Oval	Attribute

Belah ketupat Relationship

Penjelasan:

Kenalkan simbol standar dalam ERD (Chen Notation / Crow’s Foot).

Visual: contoh mini diagram ERD dengan simbol-simbol.

Slide 6 – Contoh ERD Sistem Web

Isi Slide:

- Entity: User, Produk, Pesanan
- Relationship:
 - User “melakukan” Pesanan
 - Pesanan “memiliki” Produk

Penjelasan:

Tampilkan contoh ERD sederhana aplikasi e-commerce.

Visual: diagram “User –< Pesanan >– Produk”.

Slide 7 – Kardinalitas & Relasi

Isi Slide:

- **One-to-One (1:1)** – User ↔ Profile
- **One-to-Many (1:N)** – User ↔ Pesanan
- **Many-to-Many (M:N)** – Produk ↔ Pesanan

Penjelasan:

Berikan ilustrasi tiap jenis relasi.

Visual: tiga diagram kecil menunjukkan perbedaan 1:1, 1:N, M:N.

Bagian 3 – Relasi Tabel dalam Database

Slide 8 – Konversi ERD ke Tabel

Isi Slide:

Langkah konversi:

1. Setiap entitas → tabel
2. Setiap atribut → kolom
3. Relasi → foreign key

Penjelasan:

Tunjukkan bagaimana desain konseptual (ERD) diterjemahkan menjadi tabel fisik di database.

Visual: tabel SQL hasil konversi ERD.

Slide 9 – Contoh Relasi 1:N

Isi Slide:

Tabel **Users** dan **Orders**

```
CREATE TABLE users (
```

```
  id INT PRIMARY KEY,
```

```
  name VARCHAR(50)
```

```
);
```

```
CREATE TABLE orders (
```

```
id INT PRIMARY KEY,  
user_id INT,  
FOREIGN KEY (user_id) REFERENCES users(id)  
);
```

Penjelasan:

Tunjukkan hubungan antara user dan order secara langsung menggunakan foreign key.

Visual: diagram dua tabel terhubung garis.

Slide 10 – Relasi M:N (Many-to-Many)

Isi Slide:

Gunakan *junction table*:

```
CREATE TABLE order_products (  
order_id INT,  
product_id INT,  
PRIMARY KEY (order_id, product_id)  
);
```

Penjelasan:

Jelaskan bahwa hubungan M:N butuh tabel penghubung (association table).

Visual: tiga tabel (orders–order_products–products).

Slide 11 – Normalisasi Data

Isi Slide:

- Hindari data ganda
- Setiap kolom menyimpan satu nilai
- Gunakan kunci primer dan kunci asing

Penjelasan:

Bahas prinsip normalisasi sampai bentuk ketiga (3NF) secara singkat.

Visual: diagram tabel “denormalized” → “normalized”.

Slide 12 – Apa itu ORM (Object Relational Mapping)?

Isi Slide:

- Abstraksi database dalam bentuk objek
- Menghindari query SQL manual
- Memudahkan CRUD data

Penjelasan:

Jelaskan fungsi ORM dalam framework seperti Laravel (Eloquent) atau Express.js (Sequelize).

Visual: diagram “Object ⇌ Table”.

Slide 13 – ORM di Laravel: Eloquent

Isi Slide:

- Model mewakili tabel database
- Relasi ditulis dengan fungsi:

```
public function orders() {  
    return $this->hasMany(Order::class);  
}
```

Penjelasan:

Tunjukkan bagaimana relasi *one-to-many* diimplementasikan dalam model Laravel.

Visual: kode model Laravel + panah ke tabel database.

Slide 14 – ORM di Express.js: Sequelize

Isi Slide:

```
User.hasMany(Order);  
  
Order.belongsTo(User);
```

Penjelasan:

Tunjukkan cara mendefinisikan relasi antar model di Sequelize. ORM memungkinkan JavaScript mengelola database secara deklaratif.

Visual: ikon Node.js + diagram relasi model.

Slide 15 – Operasi CRUD dengan ORM

Isi Slide:

Contoh (Laravel):

```
User::create(['name'=>'Ali']);
```

```
$users = User::all();
```

Penjelasan:

ORM mengubah query SQL menjadi kode yang mudah dibaca. Cocok untuk aplikasi skala besar dengan banyak entitas.

Visual: flowchart CRUD (Create, Read, Update, Delete).

Slide 16 – Keuntungan ORM**Isi Slide:**

- Produktivitas tinggi
- Kode lebih bersih
- Mengurangi kesalahan SQL
- Mendukung migrasi database otomatis

Penjelasan:

Tegaskan manfaat ORM dalam proyek web modern.

Visual: diagram perbandingan SQL manual vs ORM.

 Bagian 5 – Integrasi & Uji Coba**Slide 17 – Alur Integrasi Database dengan Web****Isi Slide:**

1. Desain ERD
2. Buat tabel & relasi
3. Buat model ORM
4. Hubungkan controller ke model
5. Render data di view

Penjelasan:

Paparkan langkah-langkah sistematis integrasi database → framework → web app.

Visual: alur panah vertikal dari ERD → ORM → web.

Slide 18 – Contoh Integrasi Sederhana

Isi Slide:

(Laravel)

```
Route::get('/users', function() {  
    return User::all();  
});
```

Penjelasan:

Endpoint ini langsung menampilkan data user dari database melalui model ORM.

Visual: diagram request–controller–model–database–response.

Slide 19 – Tantangan & Best Practice

Isi Slide:

- Desain ERD dengan cermat sebelum coding
- Gunakan migration & seeder
- Pastikan indexing untuk performa
- Validasi data di model

Penjelasan:

Bahas pentingnya praktik terbaik agar integrasi database stabil dan efisien.

Visual: ikon peringatan + tips singkat.

Bagian 6 – Penutup & Evaluasi

Slide 20 – Ringkasan & Diskusi

Isi Slide:

- ERD menggambarkan relasi antar entitas
- Relasi tabel diwujudkan lewat foreign key
- ORM menyederhanakan pengelolaan database
- Database terintegrasi adalah fondasi web dinamis

Diskusi:

“Apa keuntungan ORM dibanding query SQL manual di proyek besar?”

Penjelasan:

Akhiri dengan refleksi dan pertanyaan pemicu diskusi kelas.

Visual: infografik ringkasan (ERD → Relasi → ORM → Web).