

Pertemuan 6

PEMODELAN SISTEM dengan UML (bagian 2)

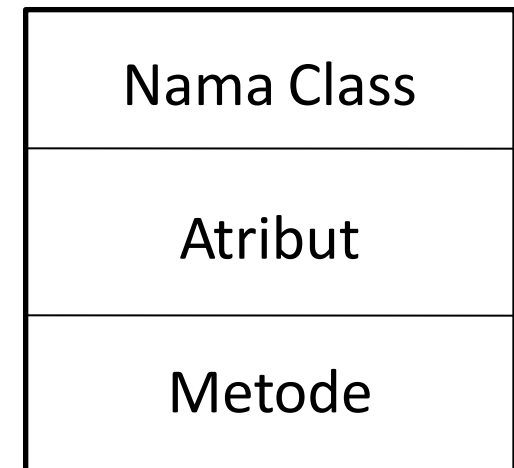
5. CLASS DIAGRAM

- Class Diagram adalah salah satu jenis diagram yang paling berguna dalam UML karena diagram tersebut secara jelas memetakan struktur sistem tertentu dengan memodelkan kelas, atribut, operasi, dan hubungan antar objek
- Mendeskripsikan jenis objek dalam sistem dan berbagai
- macam hubungan statis yang terdapat diantara objek.
- Menunjukkan properti dan operasi sebuah class dan batasan-batasan yang terdapat dalam hubungan objek tersebut.

CLASS DIAGRAM (Lanjutan)

A. Properti

- ✓ Merupakan sebuah konsep tunggal, tetapi tampak seperti dua notasi yang berbeda: atribut dan asosiasi.
- ✓ Mewakili fitur-fitur struktural dari sebuah class.
- Class berbentuk persegi panjang dengan tiga baris: nama class, atribut class, dan metode atau operasi yang mungkin digunakan class.
- Class dan subclass dikelompokkan bersama untuk menunjukkan hubungan statis antar objek



B. Atribut

Notasi atribut mendeskripsikan properti dengan sebaris teks di dalam kotak class

BU:

**visibility name: type multiplicity = default
(property-string)**

Ket:

- Visibility: menandakan atribut public (+), private (-), dan protected (#)
- Name: nama atribut
- Type: batasan tentang objek yang dapat diletakkan dalam atribut
- default value: nilai dari objek
- (property-string): properti tambahan

C. Asosiasi

- Merupakan garis solid antara dua class, ditarik dari class sumber ke class target

D. Multiplicity

- Merupakan indikasi tentang berapa banyak objek yang bisa berhubungan dengan objek lain.
- Bentuk multiplicity:
 - 1 → hanya satu
 - 0..1 → nol atau satu
 - * → lebih dari satu

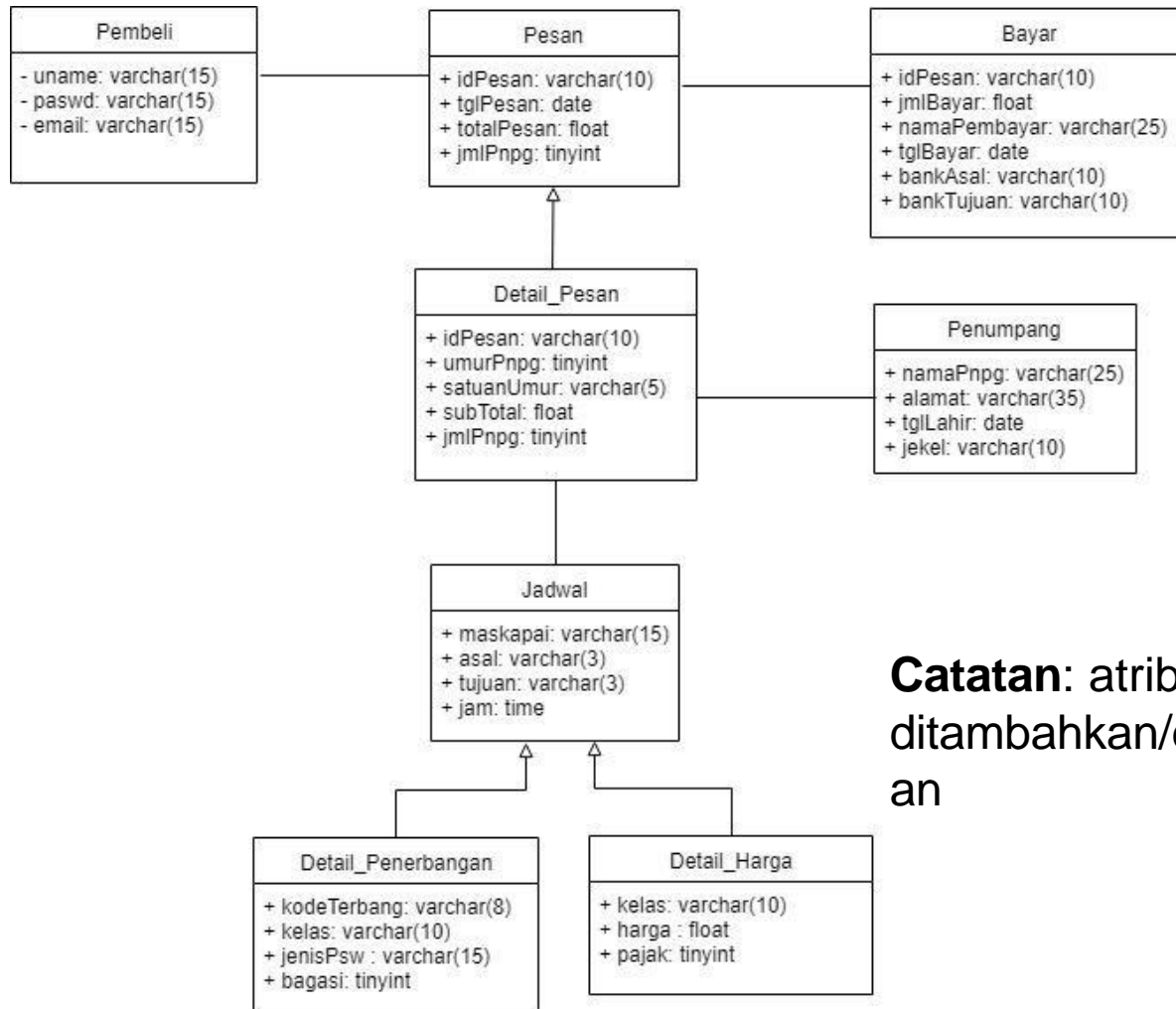
E. Operasi

- Suatu kegiatan yang akan dilakukan oleh sebuah class

Relasi pada Class Diagram

No	Simbol	Nama	Penjelasan
1	————	Asosiasi	Relasi antar class dengan makna umum, biasanya disertai dengan multiplicity
2	————→	Asosiasi berarah	Relasi antar class dengan makna class yang satu digunakan oleh class yang lain, disertai dengan multiplicity
3	————▷	Generalisasi	Relasi antar class dengan makna generalisasi-spesialisasi
4	----->	<i>Dependency</i>	Relasi antar class dengan makna kebergantungan antar class
5	————◇	Agregasi	Relasi antar class dengan makna semua-bagian (<i>whole-part</i>)

Contoh Class Diagram



Catatan: atribut dapat ditambahkan/disesuaikan

6. COMPONENT DIAGRAM

- *Component diagram* merepresentasikan dunia riil item yaitu ***component software***.
- Diakses melalui ***interfacenya***.
- Relasi antara *component* dan *interface* disebut *realization*.
- Mewakili potongan-potongan yang independen yang bisa dipesan dan diperbaharui sewaktu-waktu.
- Digunakan saat memecah sistem menjadi komponen-komponen dan menampilkan hubungan dengan antarmuka (memecah sistem menjadi lebih rendah).
- Pembagian sistem ke dalam *component* lebih didorong untuk kepentingan **marketing** daripada teknis.

Component Diagram (Lanjutan)

- Tujuan dari *component diagram* adalah untuk menunjukkan hubungan antara berbagai komponen dalam suatu sistem.
- Komponen dalam UML dapat mewakili
 - ✓ Komponen logis (misal: komponen bisnis, komponen proses)
 - ✓ Komponen fisik (misal: komponen CORBA, Java, .NET, komponen WSDL),
- Diagram komponen umumnya berisi:
 - ✓ Komponen
 - ✓ Antarmuka
 - ✓ Ketergantungan, generalisasi, asosiasi, dan realisasi

Komponen (*Component*)

- Komponen adalah kelas yang mewakili bagian modular dari suatu sistem dengan konten yang di-enkapsulasi.
- Sebuah komponen memiliki perilaku yang didefinisikan dalam hal antarmuka.
- Bagian komponen yang lebih besar dari fungsi sistem dapat digunakan kembali, dan menghubungkan antarmuka.
- Sebuah komponen dimodelkan sepanjang siklus hidup pengembangan dan disempurnakan secara berturut-turut ke dalam *deployment* dan *run-time*.
- Komponen dapat dimanifestasikan oleh satu atau lebih artefak.

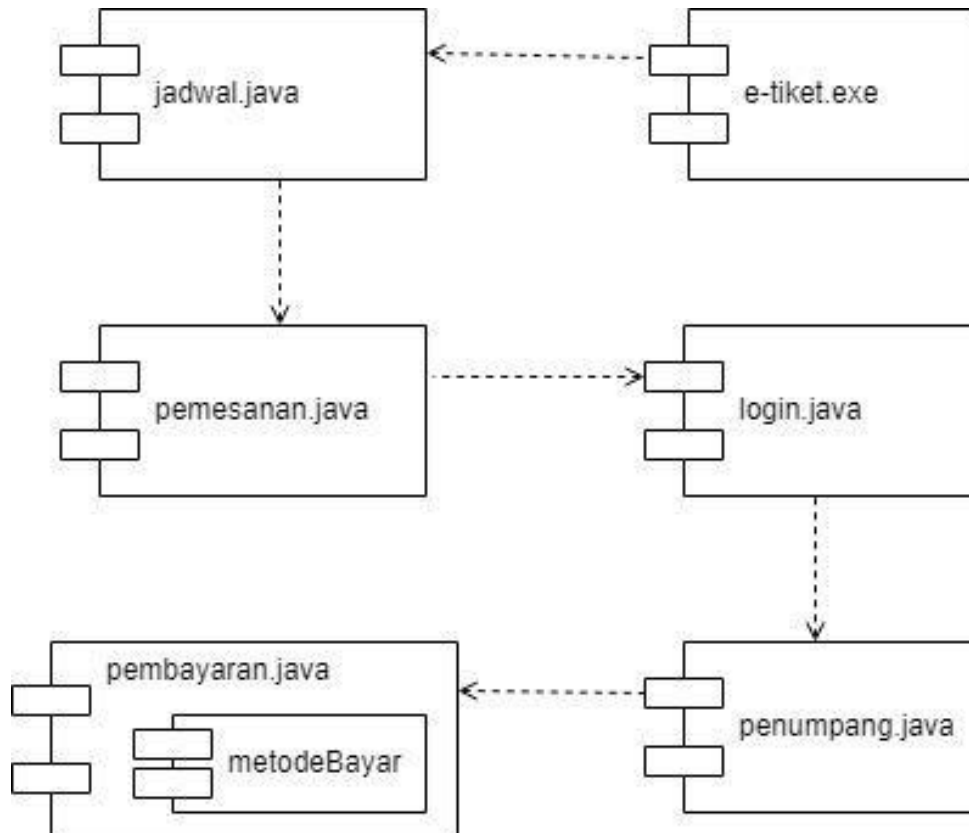
Diagram Komponen

Diagram Komponen dapat digambarkan dengan beberapa cara:

1. Memodelkan kode sumber (*source code*)
2. Memodelkan file eksekusi yang dirilis ke pengguna
3. Memodelkan basis data secara fisik
4. Memodelkan sistem yang dapat disesuaikan (adaptasi)

Contoh Component Diagram

Dengan model eksekusi



7. DEPLOYMENT DIAGRAM

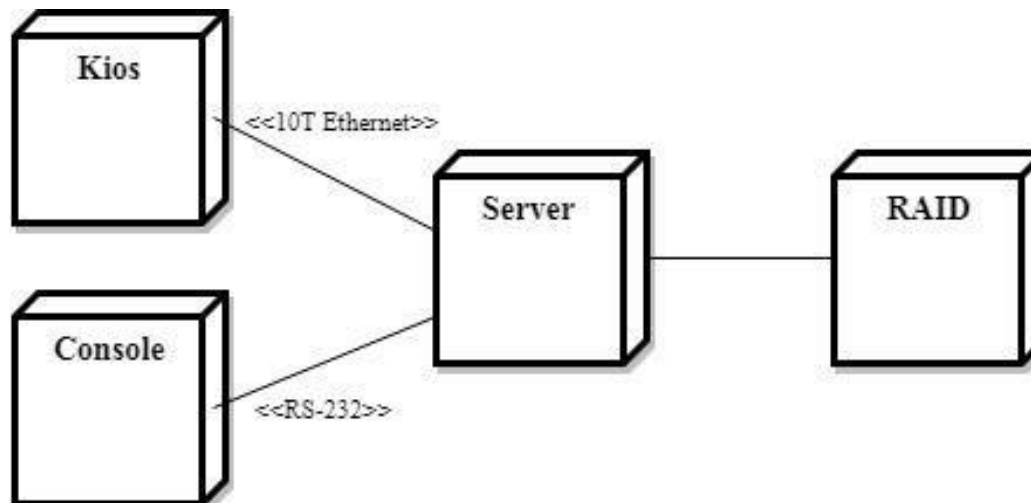
- *Deployment diagram* adalah diagram yang menunjukkan konfigurasi dari *node-node* pemrosesan waktu proses (*run time*) dan komponen-komponen yang ada di dalamnya.
- *Deployment diagram* menunjukkan tata letak sebuah sistem secara fisik, menampakkan bagian-bagian *software* yang berjalan pada bagian-bagian *hardware*.
- *Deployment diagram* umumnya mengandung:
 - ✓ *Node*: digambarkan dengan balok yang mewakili PL dasar atau elemen *hardware*, atau *node* dalam sistem.
 - ✓ Ketergantungan dan hubungan asosiasi yang digambarkan dengan garis solid dari *node* ke *node*.

Beberapa cara penamaan pada *node*:



Asosiasi

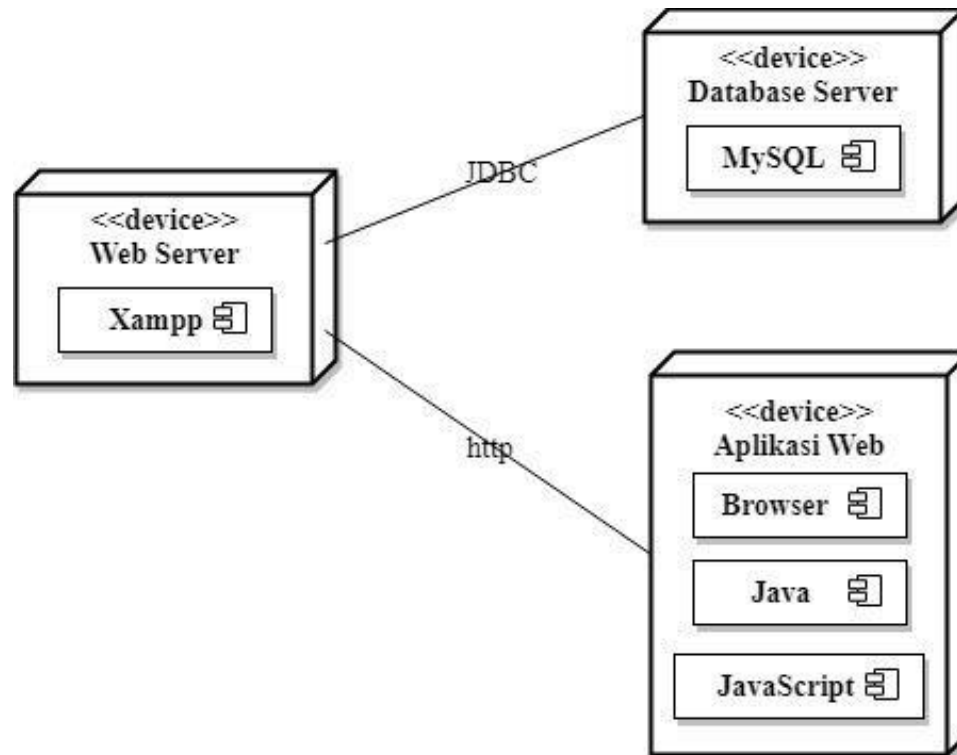
- Jenis hubungan yang paling umum adalah asosiasi yang merepresentasikan koneksi fisik antar *node* seperti koneksi *Ethernet*, saluran serial, atau *bus*.



Deployment Diagram (Lanjutan)

- *Deployment diagram* memiliki beberapa aplikasi yang dapat digunakan untuk:
 - ✓ Menampilkan elemen PL yang digunakan oleh elemen perangkat keras.
 - ✓ Mengilustrasikan pemrosesan *run time* untuk perangkat keras.
 - ✓ Memberikan pandangan tentang topologi sistem perangkat keras.
- *Deployment diagram* tidak hanya penting untuk memvisualisasikan, menentukan, dan mendokumentasikan *embedded, client/server*, dan sistem terdistribusi, tetapi juga untuk mengelola sistem yang dapat dieksekusi melalui teknik maju dan rekayasa ulang.

Contoh Deployment Diagram



LATIHAN

Dosen diminta untuk menggambarkan model-model diagram yang telah dibahas sesuai dengan kasus pada pertemuan 5.

TUGAS

- Dosen memberikan tugas kepada mahasiswa untuk membuat model diagram yang telah dibahas.
- Kasus dapat diberikan oleh dosen atau ditentukan sendiri oleh mahasiswa
- Tools untuk menggambarkan sistem adalah tools yang dipahami Mahasiswa
- Tugas dikumpulkan dalam bentuk makalah