

# **Konsep dasar pemodelan dengan UML**

**Kamal prihandani**

# Pemilik UML dan Sekilas tentang UML

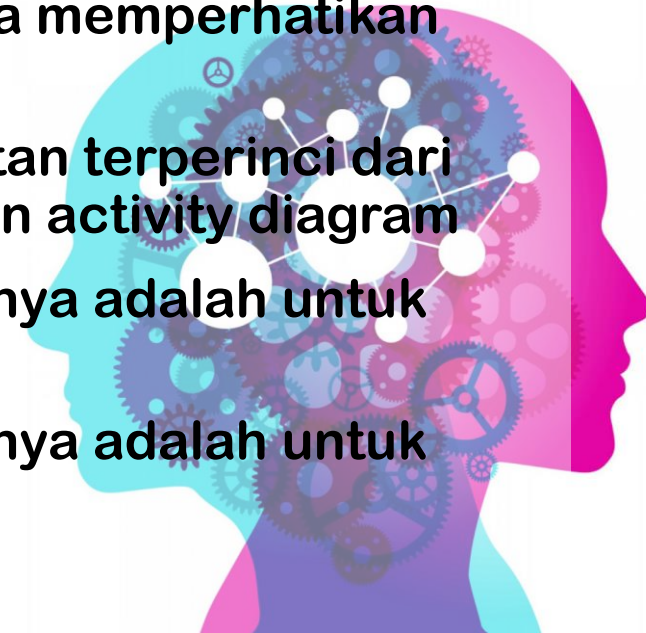
- Unified Modeling Language (UML) adalah standar bahasa pemodelan objek generasi ketiga, yang dimiliki oleh Object Management Group (OMG).
- notasi yang digunakan oleh UML mudah dikuasai dan sebagian besar mudah dipahami.
- sistem yang kompleks dapat dengan mudah dikembangkan dengan 4 diagram inti : usecase, class, activity dan state.
- UML mendukung *semua* hal yang diperlukan untuk memodelkan ketepatan waktu dan manajemen sumber daya yang mencirikan sistem ter-tenam dan real time



Object  
Management  
Group®

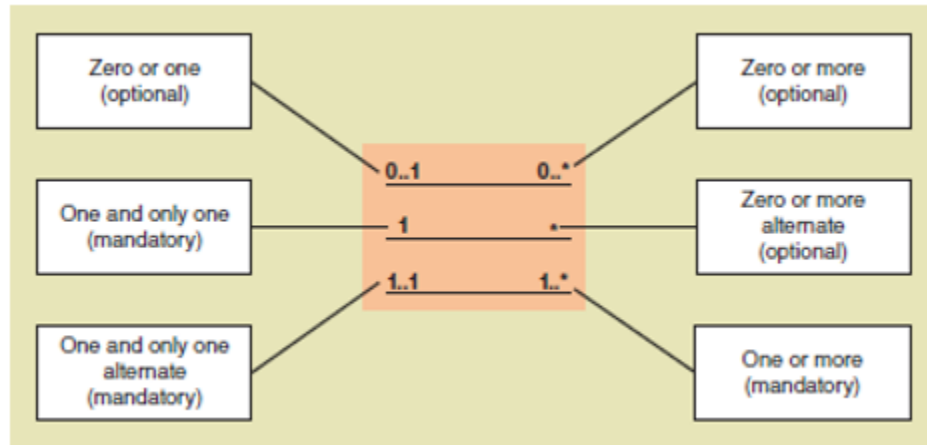
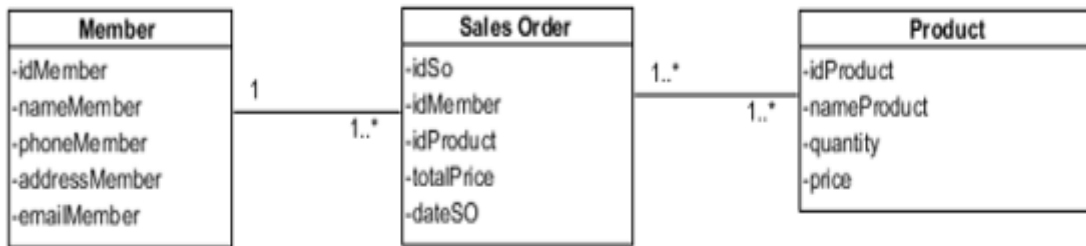
# Pemodelan

- Model adalah seperangkat abstraksi yang terintegrasi dan koheren yang mewakili sistem yang akan dirancang
- Bagian penting dari model pengguna adalah definisi semantik dari sistem yang sedang dikembangkan
- Semantik ini memiliki tiga aspek utama: **struktural**, **perilaku**, dan **fungsional**. Aspek struktural model mengidentifikasi hal-hal yang membentuk sistem. Misalnya, sekumpulan objek dan relasinya merepresentasikan keadaan atau kondisi sistem pada titik waktu tertentu
- aspek fungsional sistem mengacu pada perilaku yang diperlukan tanpa memperhatikan implementasi perilaku
- Dalam UML, aspek fungsional dimodelkan sebagai **use case**; persyaratan terperinci dari kasus penggunaan sistem dimodelkan menggunakan state diagram dan activity diagram
- Dengan penggunaan UML, pengguna membuat model aplikasi. Tujuannya adalah untuk membuat model aplikasi yang lengkap, konsisten, dan akurat
- Dengan penggunaan UML, pengguna membuat model aplikasi. Tujuannya adalah untuk membuat model aplikasi yang lengkap, konsisten, dan akurat



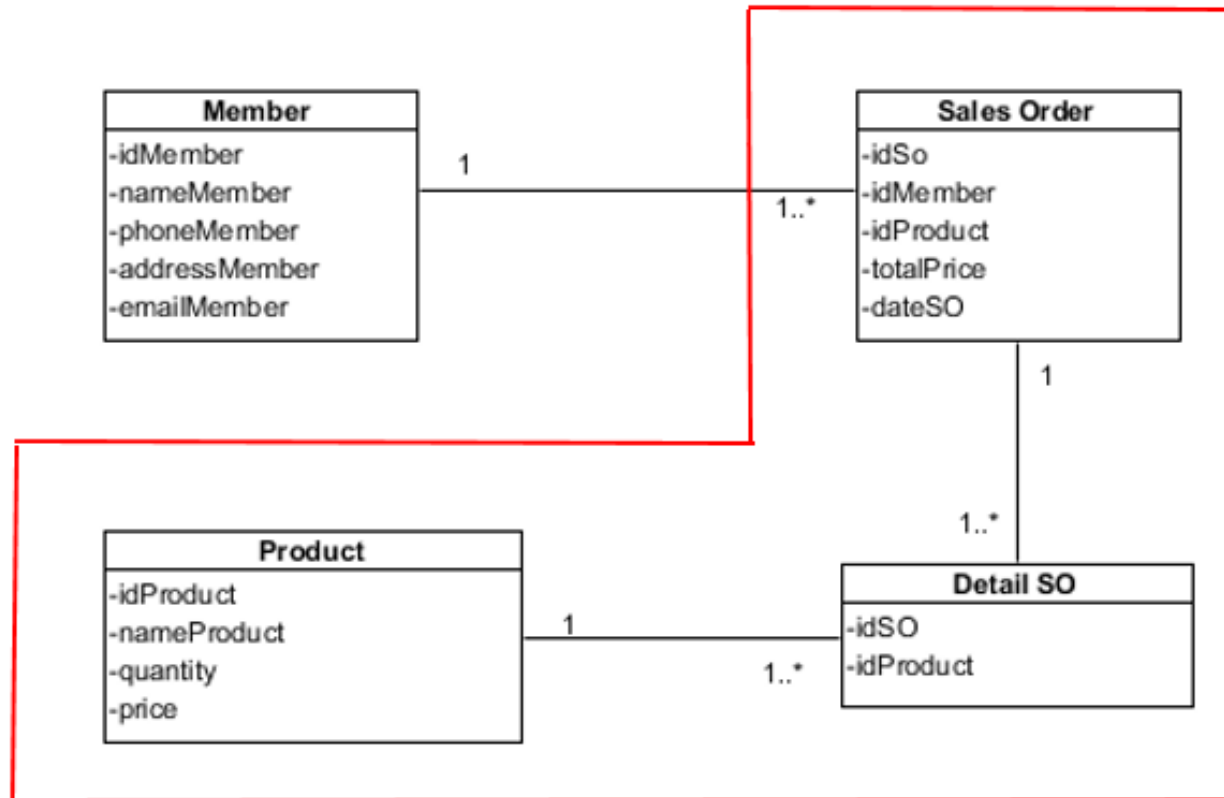
# Class diagram lanjutan

- Class diagram merupakan salah satu diagram utama dari UML untuk menggambarkan class atau blueprint object pada sebuah sistem.
- **Domain Modelling** merupakan teknik pengidentifikasian object-object pada kata benda yang terdapat pada daftar requirement yang diklasifikasikan pada area (domain) permasalahan yang sama untuk dijadikan candidate class pada class diagram.
- **Asosiasi**, yaitu hubungan statis antar class. Umumnya menggambarkan class yang memiliki atribut berupa class lain, atau class yang harus mengetahui eksistensi class lain.



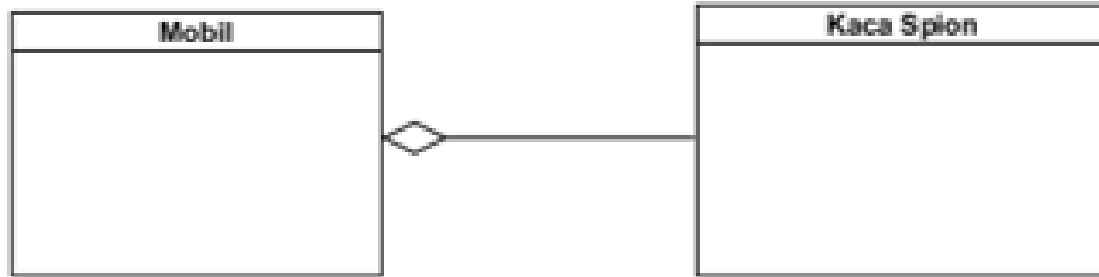
# Class diagram lanjutan

- Jika kita menemukan hubungan antar class seperti Sales Order dengan Product, yang disebut sebagai **many-to-many asosiasi**. Maka kita harus membuat **Detail Class**

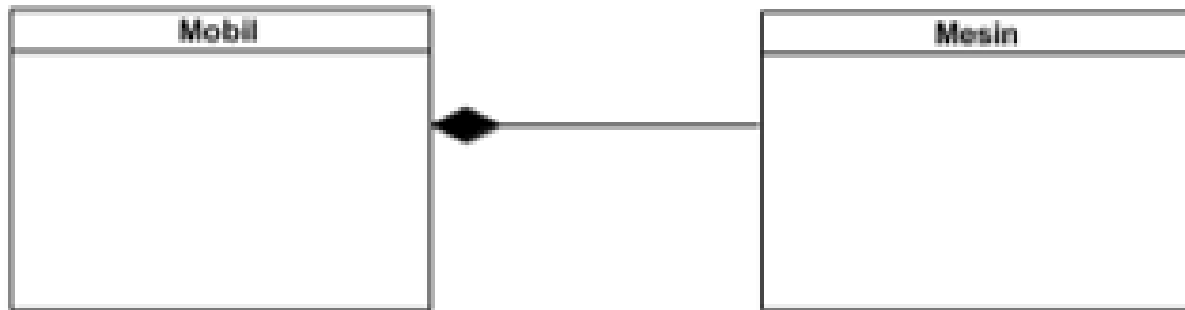


# Agregasi & Komposisi Class diagram

- Agregasi, yaitu hubungan antar class yang menyatakan hubungan **“has-a.”**



- Komposisi, yaitu hubungan antar class yang menyatakan hubungan **“part-of.”**



# Inheritance dan Generalization

- Inheritance adalah hubungan hirarkis antar class. Class dapat diturunkan dari class lain dan mewarisi semua atribut dan metoda class asalnya dan menambahkan fungsionalitas baru, sehingga ia disebut anak dari class yang diwarisinya. Kebalikan dari pewarisan adalah generalisasi.

# Tantangan saat ini ...

- Silahkan anda buat desain class diagram dengan komponen-komponen lengkap yang dibahas di slide ini, dengan *real case*.