



PROGRAM STUDI
TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS DIAN NUSWANTORO

Mata Kuliah
Analisis dan Perancangan
Berorientasi Obyek



Pengenalan Analisis dan Perancangan Berorientasi Obyek

Tim Pengampu Mata Kuliah

Analisis dan Perancangan Berorientasi Obyek

Capaian Pembelajaran

- Menjelaskan analisis dan perancangan perangkat lunak yang berorientasi obyek

Definisi

- Object-Oriented Analysis and Design adalah metode analisis yang memeriksa requirement dari sudut pandang kelas – kelas dan objek yang ditemui dalam ruang lingkup permasalahan
- OOAD merupakan cara baru dalam memikirkan suatu masalah dengan menggunakan model yang dibuat menurut konsep dunia nyata (real world)
- Pada dasarnya terdiri dari 2:
 - OOA (Object-Oriented Analysis)
 - OOD (Object-Oriented Design)

OOAD

- OOA mempelajari domain permasalahan bisnis dengan memberikan rekomendasi perbaikan system berdasarkan kebutuhan dalam menyelesaikan masalah
- OOD menentukan solusi teknis atau rancangan/**computer-based** berdasarkan identifikasi pada proses analisis

Mengapa OOAD

- Memudahkan pemanfaatan ulang code dan arsitektur
- Lebih mencerminkan dunia nyata
 - Lebih tepat dalam menggambarkan entitas, dekomposisi berdasarkan pembagian yang natural, lebih mudah untuk dipahami dan dirawat
- Kestabilan
 - Perubahan kecil dalam requirement tidak berarti perubahan yang signifikan dalam system yang dikembangkan
- Lebih mudah disesuaikan dengan perubahan (adaptif)

Konsep Kunci Perancangan Berorientasi Object

1. Menggunakan object sebagai peran sentral bukan proses
2. Menggunakan gagasan kelas
3. Satu Bahasa untuk mendefinisikan system (UML)
4. Kemampuan beradaptasi dan perluasan (extend)

Peran Sentral dari Object

- Object sebagai inti dari desain perangkat lunak bukan proses
 - Proses rentan terhadap perubahan dan Sebagian system lama tidak dapat digunakan Kembali (reusable)
- Object berpusat pada struktur data dan method yang dapat dimodifikasi atau disesuaikan dengan kebutuhan

Gagasan Kelas

- Kelas – kelas memungkinkan perancang perangkat lunak untuk melihat object sebagai jenis entitas yang berbeda
- Melihat sebagai object memungkinkan untuk menggunakan mekanisme klasifikasi untuk mengkategorikan jenis, mendefinisikan hirarki, dan terlibat pada ide – ide spesialisasi dan generalisasi

Suatu Bahasa untuk Mendefinisikan Sistem

- UML atau Unified Modeling Language telah terpilih sebagai alat standar untuk menggambarkan produk akhir dari kegiatan desain
- Dokumen – dokumen yang dihasilkan dalam Bahasa ini dapat dipahami secara universal
- UML dapat digunakan sebagai blueprint oleh engineer lainnya

Perluasan dan Kemampuan Beradaptasi

- Perangkat lunak memiliki fleksibilitas yang tidak biasanya ditemukan dalam perangkat keras dan ini memungkinkan kita untuk memodifikasi entitas yang ada
- Inheritance, memungkinkan menciptakan kelas baru dari keturunan kelas yang ada (Superclass)

Cohesion & Coupling

- Salah satu konsep penting dalam OOP adalah **Low Coupling High Cohesion**
- **Cohesion** berhubungan dengan responsibility sebuah class
- **Coupling** berhubungan dengan seberapa besar ketergantungan class dengan class lain

Cohesion & Coupling

- Semakin spesifik sebuah responsibility class, maka akan semakin rendah Tingkat ketergantungannya, begitu juga sebaliknya
- Tujuan utama dari konsep ini adalah fleksibilitas sebuah class, artinya class yang di desain dengan low coupling high cohesion akan mudah dimodifikasi



TERIMA KASIH

ANY QUESTIONS?