



Yayasan Pendidikan Alfian Husin

ibi INFORMATICS
& BUSINESS
INSTITUTE
DARMAJAYA
BANDAR LAMPUNG - INDONESIA

Perancangan pada Peringkat Komponen

Chapter 23

1. Pendahuluan

Perancangan tingkat komponen terjadi setelah iterasi pertama pada perancangan arsitektural diselesaikan.

“Sebuah komponen adalah bangunan pembentuk PL komputer yang bersifat modular, dapat di *deploy* , dan dapat digantikan, yang membungkus implementasi dan memperlihatkan sejumlah antarmuka”.

1.1 Pandangan yang penting tentang suatu komponen

1. Pandangan berorientasi objek

Pada sudut pandang ini sebuah komponen sesungguhnya adalah sejumlah kelas yang saling berkolaborasi.

Tahapan yang dilakukan :

- Memulai model-model kebutuhan
- mengelaborasi kelas-kelas analisis, dan kelas-kelas infrastruktur.

2. Pandangan tradisional

Pada dasarnya dalam konteks ini adalah elemen fungsional program yang membentuk logika pemrosesan dan sebuah antarmuka yang memungkinkan komponen itu dipanggil dan data dilewatkan ke dalamnya.

Peran penting dalam dalam pelayanan pandangan ini adalah :

- Komponen kendali yang mengkoordinasi pemanggilan semua komponen ranah permasalahan yang lain
- Komponen ranah permasalahan yang mengimplementasikan secara lengkap atau secara fungsi yang diperlakukan oleh pelanggan

3. Komponen infrastruktur yang bertanggung jawab untuk fungsi-fungsi yang mendukung pemrosesan yang diperlukan dalam ranah permasalahan

2. Perancangan Komponen Berbasis Kelas

a. Prinsip-prinsip perancangan Dasar

Yang mendasari prinsip ini adalah membuat perancangan yang memungkinkan dilakukannya perubahan-perubahan dan untuk mengurangi penalaran imbas saat perubahan-perubahan itu terjadi.

Prinsip-prinsip dari perancangan dasar yaitu :

- OCP (Open Closed Principle)

Sebuah komponen seharusnya terbuka untuk perluasan tetapi tertutup bagi modifikasi

- LSP (Liskov Substitution Principle)

Subkelas seharusnya dapat disubsitusi untuk kelas-kelas dasar mereka.

- DIP (Dependency Inversion Principle)

Bergantung pada abstraksi-abstraksi, jangan bergantung pada kesatuan.

- ISP (Interface Segregation Principle)

Banyak antarmuka yang bersifat spesifik terhadap klien lebih baik daripada satu antarmuka besar yang memiliki kegunaan yang bersifat umum.

b. Panduan perancangan pada peringkat komponen

Panduan-panduannya terdiri atas :

- Komponen
- Perencanaan lantai dasar
- Antarmuka
- Kebergantungan dan pewarisan

c. Kohesi

Pada sistem berorientasi objek kohesi mengimplikasikan bahwa suatu komponen atau kelas hanya membungkus didalamnya atribut-atribut dan operasi-operasi yang berhubungan dekat satu sama lain dan hanya membungkus attribute dan operasi yang berhubungan dekat dengan kelas atau komponen itu sendiri.

Beberapa jenis kohesi yang berbeda, antara lain :

- Fungsional
- Lapisan
- komunikasi

d. Derajat keterhubungan antarkomponen

Derajat keterhubungan antarkomponen adalah pengukuran kualitatif tentang derajat bagaimana kelas-kelas saling terhubung satu dengan yang lainnya, saat komponen menjadi saling terhubung, derajat keterhubungan juga bertambah.

Kategori-kategorinya antara lain :

- Keterhubungan isi
- Keterhubungan umum
- Keterhubungan kendali
- Keterhubungan data
- Keterhubungan pemanggilan rutin
- Keterhubungan penggunaan tipe data
- Keterhubungan inklusi (impor)
- Keterhubungan eksternal.

3. Melakukan perancangan pada peringkat komponen

Langkah-langkah yang diterapkan untuk sistem yang berparadigma “pemrograman berorientasi objek” adalah :

- a. Mendefinisikan semua kelas-kelas perancangan yang berhubungan dengan ranah permasalahan.
- b. Mengelaborasi semua kelas-kelas perancangan yang berhubungan dengan infrastruktur

c. Mengelaborasi semua kelas-kelas yang tidak diperlukan sebagai kelas-kelas yang dapat digunakan ulang, tahapan yang dilakukan diantaranya :

- spesifikasi rincin pesan saat kelas-keas atau komponen-komponen berkolaborasi
- Identifikasi antarmuka-antarmuka yang sesuai untuk masing-masing komponen
- Elaborasi attribute-attribute dan definisikan tipe data dan struktur data yang diperlukan
- Deskripsikan aliran pemrosesan didalam masing-masing operasi secara rinci.

- d. Mendeskripsikan sumber-sumber data persiten (basis data dan berkas/file) dan mengidentifikasi kelas-kelas yang dibutuhkan untuk mengelolanya
- e. Mengembangkan dan mengelaborasi representasi-representasi perilaku untuk kelas atau komponen
- f. Elaborasi diagram penebaran komponen untuk menyediakan rincian implementasi tambahan
- g. Lakukan refaktorisasi sertiap representasi perancangan pada peringkat komponen

Questions & Discussion