



Desain Perangkat Lunak

STRUCTURAL DESIGN PATTERN COMPOSITE

Kelompok 5





KELOMPOK 5

- 1 Muhammad Fathurraihan Saputra (2205076)**
- 2 Muhammad Rizki (2205078)**
- 3 Azizah Dwi Lestari (2205065)**
- 4 Adha Jihaddien HR (2205083)**


PENGERTIAN

Composite adalah salah satu pola desain (design pattern) yang termasuk dalam kategori Structural Patterns. Pola desain Composite digunakan untuk mengomposisi objek ke dalam struktur pohon untuk mewakili hierarki bagian-keseluruhan. Dengan menggunakan pola ini, Anda dapat memperlakukan objek individual dan komposisi objek secara seragam.






KONSEP POLA DESAIN COMPOSITE

- Mengelola objek yang merupakan bagian dari komposisi dengan cara yang sama seperti Anda mengelola objek tunggal.
 - Membuat struktur pohon yang fleksibel yang dapat berisi objek individual (daun) dan objek komposit (cabang) secara hierarkis.
- 

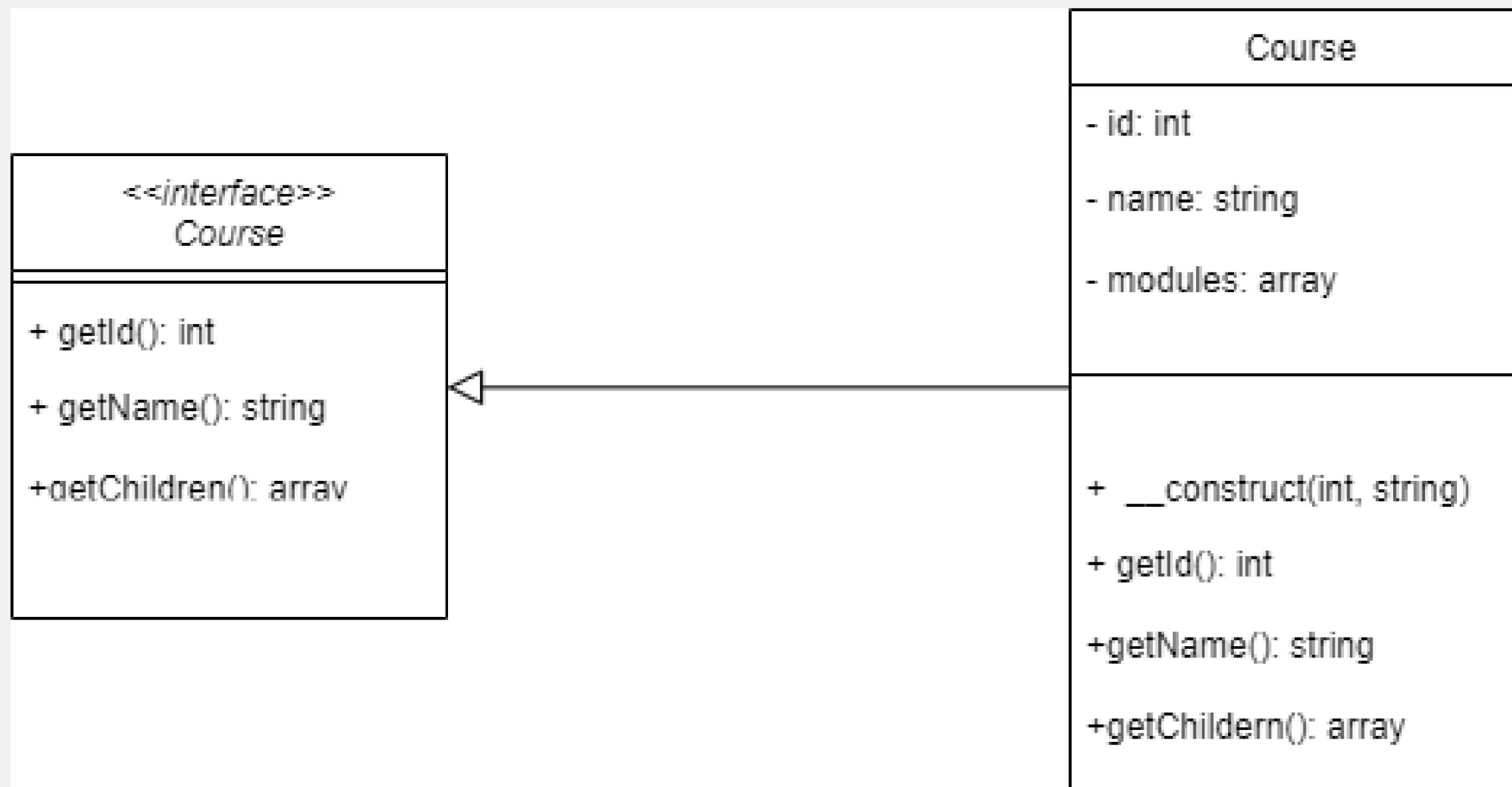


STUDI KASUS

penerapan pola desain Composite dalam konteks kursus dan modul. Terdapat antarmuka Course dengan tiga metode: `getId()`, `getName()`, dan `getChildren()`. Antarmuka ini diimplementasikan oleh kelas Course, yang memiliki atribut `id`, `name`, dan `modules` (array modul).




DIAGRAM






KELEBIHAN

- **Konsistensi dalam Penggunaan:** Memungkinkan perlakuan objek individu dan komposit secara seragam.
 - **Fleksibilitas dalam Penambahan Komponen:** Mudah diperluas dengan menambahkan jenis komponen baru.
 - **Penyederhanaan Pengelolaan Hierarki:** Menyederhanakan pengelolaan objek kompleks melalui struktur pohon.
 - **Reusabilitas dan Pemeliharaan Kode:** Meningkatkan modularitas dan kemudahan pemeliharaan kode.
- 



KEKURANGAN

- **Kompleksitas Implementasi:** Implementasi bisa menjadi rumit dengan banyak jenis komponen.
 - **Membutuhkan Lebih Banyak Memori:** Memori lebih besar diperlukan untuk menyimpan referensi anak-anak.
 - **Kesulitan dalam Penegakan Aturan Spesifik:** Sulit menerapkan aturan bisnis spesifik tanpa logika tambahan.
 - **Overhead dalam Pengelolaan Objek:** Overhead tambahan dalam operasi menambah, menghapus, dan mengelola objek.
- 



THANK YOU

