

Nama: Ferreyly Selara Ibn

NIM: 121140307

Kelas: RA

(*) Design pattern

↳ Solusi umum untuk masalah desain perangkat lunak yang sering muncul. Design Pattern merupakan Pedoman atau panduan yang digunakan untuk merancang struktur dan interaksi antar komponen dalam suatu sistem. Design pattern membantu mengatasi masalah umum yang mungkin muncul selama proses pengembangan perangkat lunak dan mempromosikan praktik desain yang baik.

(*) Konsep-konsep dalam design pattern

1) Reusable solutions

↳ Design pattern memberikan solusi yang telah divalidasi dan terbukti efektif untuk masalah tertentu.

2) Abstraction

↳ Design pattern menggunakan abstraksi untuk menyembunyikan detail kompleksitas dan fokus pada fitur penting dari suatu desain.

3) Common Vocabulary

↳ Design pattern menyediakan vokabulari umum yang dapat digunakan oleh tim pengembang untuk berkomunikasi.

4) Fleksibilitas

↳ Design pattern memberikan fleksibilitas dan stabilitas pada suatu sistem.

(*) Jenis-jenis design pattern

↳ (-) Creational Pattern

↳ Fokus pada pembuatan objek. Pola ini membantu untuk mengabstraksi proses pembuatan objek, sehingga kode program tidak bergantung jika dibuat objek yang spesifik.

(-) Structural

↳ Pola desain yang berkaitan dengan cara menyusun objek atau kelas. desain ini membantu untuk menyederhanakan struktur kode program, sehingga kode program lebih mudah dimodifikasi, ditambah atau diintegrasikan.

(-) Behavior

↳ Pola desain ini berkaitan dengan cara berinteraksi antara objek atau kelas. Pola desain ini membantu untuk meningkatkan komunikasi, koordinasi, dan kolaborasi antar objek atau kelas.

(*) MVC

↳ Singkatan dari Model View Controller, adalah suatu pola desain arsitektur perangkat lunak yang digunakan untuk mengatur struktur kode dalam pengembangan perangkat lunak. Pola ini memisahkan aplikasi menjadi 3 komponen utama yaitu Model, View, dan Controller. Tujuannya adalah memberikan tugas yang berbeda dalam pengembangan perangkat lunak agar kode lebih mudah dipahami dan dimodifikasi.

(*) Komponen

(-) Model

↳ bertanggung jawab untuk menyimpan dan mengelola data. Model tidak boleh bergantung pada View, atau Kontrol. Model dapat berfungsi sebagai alat untuk mengakses dan mengelola informasi yang tersimpan dalam database.

(-) View

↳ Tugas View adalah menampilkan informasi pada pengguna dan tidak memiliki korespondensi pada model atau control. View merupakan bagian dalam aplikasi web yang bertanggung jawab atas menampilkan data pada pengguna.

(-) Controller

↳ Berperan untuk menerima input dari pengguna serta memprosesnya dan mengirimkan data dari model ke View