

A. Design Pattern

Design pattern adalah solusi umum untuk masalah umum dalam desain perangkat lunak. Mereka berfungsi sebagai pola dasar atau template untuk menyelesaikan.

Jenis-Jenis Design Pattern

Design pattern dapat dikategorikan menjadi 3 jenis utama:

a. Creational Pattern

Digunakan untuk membuat objek. Contoh: Singleton, Factory, Builder.

b. Structural Pattern

Digunakan untuk mengatur hubungan antar objek. Contoh: Adapter, Decorator, Composite.

c. Behavioral Pattern

Digunakan untuk mengatur perilaku objek. Contoh: Observer, Strategy, Command.

Keuntungan Menggunakan Design Pattern

Beberapa keuntungan menggunakan design pattern meliputi:

- * Aplikasi lebih mudah dipelihara
- * Aplikasi lebih mudah dikembangkan
- * Aplikasi lebih fleksibel
- * Aplikasi lebih terstruktur
- * Aplikasi lebih mudah diuji

B. Konsep MVC (Model-View-Controller) Design Pattern

MVC memisahkan logika aplikasi menjadi tiga komponen utama:

Model, View, dan Controller. Konsep ini bertujuan meningkatkan modularitas, fleksibilitas, dan pemeliharaan kode.

a. Model

Bertanggung jawab untuk menyimpan dan mengelola data. Model tidak boleh bergantung pada tampilan atau kontrol. Model dapat digunakan untuk melakukan berbagai operasi, seperti menambah, mengubah, menghapus, mencari, dan menampilkan data.

b. View

Bertanggung jawab untuk menampilkan data pengguna. View tidak boleh bergantung pada model atau kontrol. View biasanya berupa file HTML yang berisi kode HTML, CSS, dan JavaScript.

c. Controller

Bertanggung jawab untuk menerima input dari pengguna dan memrosesnya. Controller mengirimkan data dari model ke View.