

# Contoh Analisis MVC

## Studi Kasus: Pengelolaan data Buku

Bagian 1: Struktur Tabel	
Database	: Ci
Tabel	: Buku
Struktur	: kode_buku char(10) PRIMARY KEY judul_buku char(30) penerbit char(30) isbn char(30) stok_buku int(11)

Bagian 2: Analisis Model (M)	
Nama file model	: Buku.php
Nama Class model	: Buku
Nama Fungsi	Tujuan
tampil_data_buku(\$kode)	untuk proses tampil data
simpan_data_buku(\$data)	untuk proses simpan data baik data baru ataupun data hasil koreksi.
hapus_data_buku(\$kode)	untuk proses hapus buku

Bagian 3: Analisis View (V)	
Nama File View	Tujuan
Perpustakaan.php	sebagai halaman awal aplikasi perpustakaan
tampilbuku.php	sebagai halaman untuk menampilkan daftar buku
tambahbuku.php	sebagai halaman form untuk menambah buku
koreksibuku.php	sebagai halaman form untuk koreksi buku
konfirmasihapus.php	sebagai halaman konfirmasi untuk hapus buku

Bagian 4: Analisis Controller (C)	
Nama file controller	: perpustakaan.php
Nama Class controller	: Perpustakaan
Nama Fungsi	Tujuan
__construct()	untuk memanggil constructor pada class parent. Fungsi constructor

	untuk melakukan set nilai default atau menjalankan proses default ketika class di load.
Index()	untuk memanggil halaman utama: <ul style="list-style-type: none"> <li>• load file view “perpustakaan.php”</li> </ul>
tampil_buku()	untuk melayani proses tampil buku: <ul style="list-style-type: none"> <li>• memanggil fungsi “tampil_data_buku(\$kode) pada model “Buku”, data yang ditampilkan adalah semua data buku.</li> <li>• Mengirim hasil pemanggilan fungsi ke view “tampilbuku.php”</li> </ul>
tambah_buku()	untuk melayani proses tambah buku: <ul style="list-style-type: none"> <li>• load file view “tambahbuku.php”</li> </ul>
simpan_buku()	untuk melayani proses simpan buku: <ul style="list-style-type: none"> <li>• menangkap data yang dikirim user melalui form, baik tambah buku (file view “tambahbuku.php”) atau koreksi buku (file view “koreksibuku.php”) melalui variabel POST</li> <li>• memanggil fungsi simpan_data_buku(\$data) pada model “Buku”.</li> </ul>
koreksi_buku(\$kd)	untuk melayani proses koreksi buku: <ul style="list-style-type: none"> <li>• memanggil fungsi tampil_data_buku(\$kode) dengan kode sesuai dari kode buku yang dipilih untuk dikoreksi</li> <li>• Mengirim hasil pemanggilan fungsi diatas ke file view “koreksibuku.php” untuk ditampilkan dan dikoreksi</li> </ul>
konfirmasi_hapus_buku(\$kd)	untuk melayani proses konfirmasi hapus buku: <ul style="list-style-type: none"> <li>• memanggil fungsi “tampil_data_buku(\$kode) pada model “Buku”, dengan kode sesuai dari kode buku yang dipilih untuk dihapus</li> <li>• Mengirim hasil pemanggilan fungsi diatas ke view “konfirmasihapus.php” untuk ditampilkan dan dikonfirmasi user untuk dihapus</li> </ul>
hapus_buku(\$kd)	untuk melayani proses hapus buku: <ul style="list-style-type: none"> <li>• memanggil fungsi “hapus_data_buku(\$data) pada model “Buku”, dengan kode/data sesuai dari kode buku yang dipilih untuk dihapus.</li> <li>• Mengembalikan nilai return status simpan buku berhasil atau tidak.</li> </ul>