

# KONSEP MVC



# MVC?

- MVC adalah singkatan dari Model View Controller. MVC adalah teknik pemrograman yang memisahkan bisnis logic (alur pikir), data logic (penyimpanan data) dan presentation logic (antarmuka aplikasi) atau secara sederhana adalah memisahkan antara desain, data dan proses.

# Komponen MVC

## □ Model

Berhubungan dengan data dan interaksi ke text base, database atau webservice. Biasanya di ddlama model berisi akan berisi class dan fungsi untuk mengambil, melakukan update dan menghapus data web.

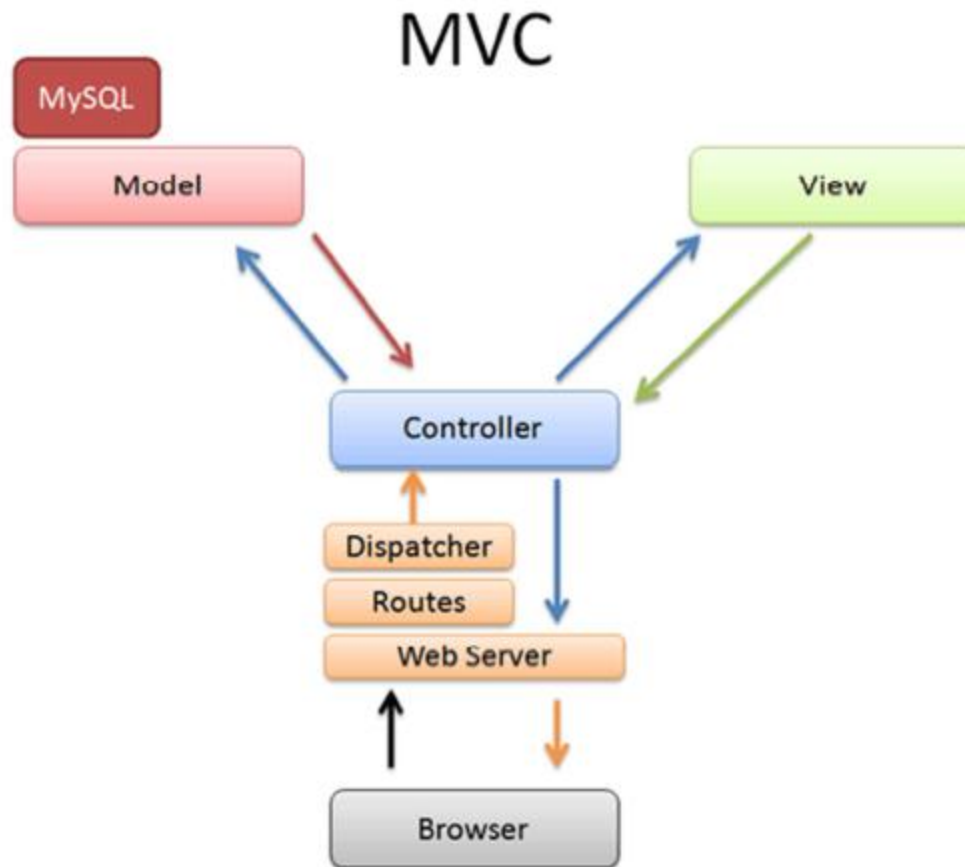
# View

- Berhubungan dengan segala sesuatu yang ditampilkan ke end-user. Bisa berupa halaman web, rss, javascript dan lain-lain. Kita harus menghindari adanya logika atau pemrosesan data di view. View bisa dikatakan sebagai halaman yang dibuat dengan HTML dan bantuan CSS atau Javascript.

# Controller

- Controller bertindak sebagai penghubung data dan view. Di Dalam controller inilah terdapat class-class dan fungsi-fungsi yang dapat memproses permintaan dari view ke dalam struktur data di dalam model. Controller juga tidak boleh berisi kode untuk mengakses basis data. Tugasnya adalah menyediakan berbagai variabel yang akan ditampilkan di view, memanggil model untuk melakukan akses ke basis data, menyediakan penanganan kesalahan, mengerjakan proses logika dari aplikasi dan melakukan validasi atau cek terhadap input.

# Proses mvc



# Proses kerja

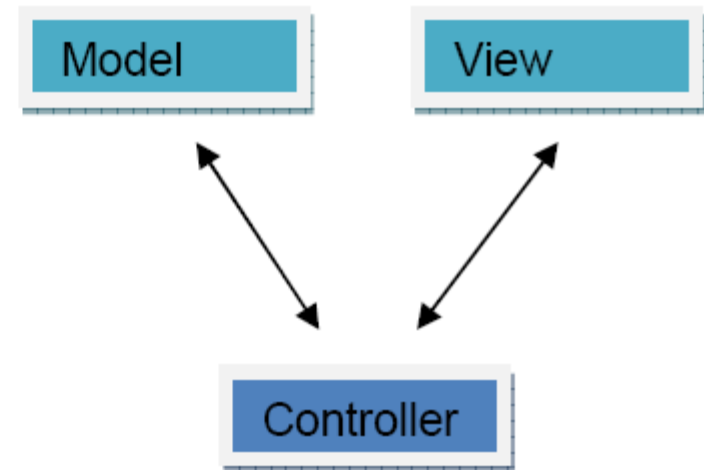
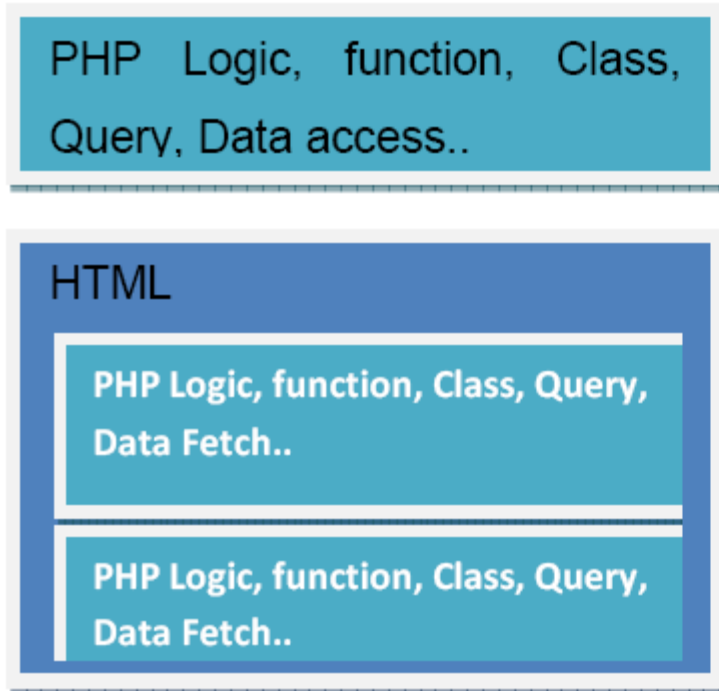
- Secara gampangnya sebenarnya controller adalah “otak” sedang view adalah “muka” dan model adalah “data”. Ketika browser memberi request maka akan dipilih untuk controller mana yang tepat untuk menangani request tersebut dan meminta data dari model dan meminta view yang dibutuhkan. Kemudian di keluarkan hasilnya lagi lewat controler ke browsernya.

# Perbedaan php biasa dan mvc

- Php biasa
- Browser akan me-load semua library dan fungsi yang dibutuhkan kemudian di proses di PHP dan selanjutnya digabung di HTML.
- MVC
- Browser berinteraksi dengan controller, controller yang menerima dan membalas request dari browser. Jika membutuhkan data maka akan diambil dari model dan UI akan meminta ke view



# Perbandingan php biasa dan mvc



# Kenapa MVC?

- Untuk kasus program sederhana tidak terlalu masalah tetapi untuk program besar yang memanggil suatu fungsi yang berulang-ulang akan menambah beban dan tidak efektif.

# Contoh MVC

## File book.php

1	<?
2	class book {
3	
4	public \$judul;
5	public \$pengarang;
6	public \$penerbit;
7	public \$tahun;
8	
9	public function __construct(\$title,\$author,\$publisher,\$year)
10	{
11	\$this->judul = \$title;
12	\$this->pengarang = \$author;
13	\$this->penerbit = \$publisher;
14	\$this->tahun = \$year;
15	}
16	
17	}
18	
19	?>

# model

## File model.php

```
1  <?
2  include_once 'book.php';
3
4  class model {
5
6      public function getData()
7      {
8          return array(
9              new book('Pemrograman PHP & MySql', 'Bayu
10 Priyambadha', 'UB Press', '2011'),
11              new book('Pemrograman Framework MVC dengan
12 PHP', 'Widhy', 'UB Press', '2011'),
13              new book('Membangun Aplikasi Web dengan PHP dan
14 AJAX', 'Achmad Arwan', 'UB Press', '2012'),
15              new book('Kolaborasi PHP, MVC dan AJAX', 'Bayu
16 Priyambadha', 'UB Press', '2012')
17          );
18      }
19  }
20
21  ?>
```

# view

## File view.php

```
1 <html>
2     <head>
3         <title>MVC dengan PHP</title>
4     </head>
5     <body>
6         <center>
7
8             <?=$data?>
9
10        </center>
11    </body>
12</html>
```

# controller

## File controller.php

```
1 <?
2 include_once 'model.php';
3
4 class controller {
5
6     function invoke()
7     {
8         $model_data = new model();
9         $row_data = $model_data->getData();
10
11         $data = "<table border=1>
12
13         <tr><th>JUDUL</th><th>PENGARANG</th><th>PENERBIT</th><th>TAHUN</th>
14 ></tr>";
15         foreach ($row_data as $key => $value) {
16             $data .= "<tr><td>". $value->judul. "</td><td>". $value->
17 >pengarang. "</td><td>". $value->penerbit. "</td><td>". $value->
18 >tahun. "</td></tr>";
19         }
20         $data .= "</table>";
21         include('view.php');
22     }
23 }
24
25 ?>
```