

Bikin Framework PHP Sendiri dengan Teknik OOP dan MVC

DAVID NAISTA



CV. LOKOMEDIA

Bikin Framework PHP Sendiri dengan OOP & MVC

Perpustakaan Nasional : Katalog Dalam Terbitan (KDT)

Penulis : David Naista

Bikin Framework PHP Sendiri dengan Teknik OOP dan MVC

- Cet. I. - Yogyakarta : Penerbit Lokomedia, 2016

183 halaman; 14 x 21 cm

ISBN : 978-602-62310-0-0

Penerbit Lokomedia,

Cetakan Pertama : April 2016

Editor : Lukmanul Hakim

Cover : Subkhan Anshori

Layout : Lukmanul Hakim

Diterbitkan pertama kali oleh :

CV. LOKOMEDIA

Jl. Jambon, Perum. Pesona Alam Hijau 2 Kav. B-4, Kricak
Yogyakarta 55242.

email : redaksi@bukulokomedia.com

website : www.bukulokomedia.com

Copyright © Lokomedia, 2016

Hak Cipta dilindungi oleh Undang-Undang

Dilarang memperbanyak, mencetak ataupun menerbitkan sebagian maupun seluruh isi buku ini tanpa izin tertulis dari penerbit.

KATA PENGANTAR

Puji dan syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa, atas berkat dan rahmat-Nya, penulis dapat menyelesaikan buku dengan judul **“Bikin Framework PHP Sendiri dengan Teknik OOP dan MVC”**.

Buku ini tidak akan selesai kepada pembaca tanpa bantuan dari berbagai pihak. Oleh karena itu, penulis menyampaikan terima kasih kepada:

1. Allah SWT, tanpa semua nikmat dan rahmatnya penulis tidak bisa menyelesaikan penulisan buku ini.
2. Kedua Orang Tua (Naimin Suhandi & Sri Utami), Adik (Edwin Naista) dan saudara-saudara penulis yang selalu memberikan support.
3. Bapak Ferdian Rahabista dan Ibu Dwi Retno Wahyuningsih.
4. Kepala Sekolah Ibu Hj Suindriyati, S.Pd, M.Pd dan semua Guru SMK Negeri 3 Metro.
5. Sekar Pratiwi, Mas Virgiawan Listanto, Mas Oky Trilupito, Vive Vio Permana, Ariyanto, Risky Pangestu, Nur Kholis, Bayu Hadi Dirgantara, Adam Wirayuda, Iqbal May, Muhammad Try Wibowo, Daim Kurniawan, Koh Hantze Sudarma, Koh Dennise Adrianto, Kak Iqbal Maulana, Aji Gumelar, Eric Yoman, Mahesa Indrajaya, Nicho Anggara, Adam Ridwany, Rian Nur Adli serta semua teman-teman dari Id-Webmaster, MetroCrew, TC ASC & WSC, Silpi Lovers dan teman kuliah BINUS yang tidak bisa disebutkan satu per satu.
6. Semua pihak yang turut membantu penulisan buku ini yang juga tidak bisa penyusun sebutkan satu persatu.
7. Serta para pembaca buku ini.

Tak ada gading yang tak retak. Demikian pula, tak ada karya yang sempurna. Oleh karena itu, penulis mengharapkan kritik dan saran dari buku ini.

Diharapkan buku ini dapat memberikan pengetahuan bagi teman-teman pembaca tentang apa itu OOP (*Object Oriented Programming*) PHP dan bagaimana membuat framework sendiri dengan konsep MVC (*Model View Controller*).

Akhir kata, semoga semua materi didalam buku ini dimengerti oleh pembaca dan dapat berguna dalam dunia kerja.

Jakarta, Februari 2016

David Naista

davidnaista83@gmail.com

DAFTAR ISI

BAB 1. Pengertian dan Konsep	1
1.1. Bahasa Pemrograman dan Bahasa Markup	2
1.2. Apa itu Framework?.....	5
1.3. Apa itu OOP?	7
1.3.1. Pengertian OOP.....	7
1.3.2. Kenapa Harus OOP?.....	7
1.3.3. Prosedural Vs OOP	8
1.4. Apa itu MVC?	10
BAB 2. Object Oriented Programming PHP	13
2.1. Pengenalan OOP PHP.....	14
2.2. Sintak-Sintak OOP	14
BAB 3. Bikin Framework PHP Sendiri	25
3.1. Pendahuluan	26
3.2. Membuat Susunan Folder.....	26
3.3. Membuat File Utama dan File Konfigurasi	27
3.3.1. Membuat File Utama (index.php).....	27
3.3.2. Membuat File Konfigurasi (config.php)	29
3.4. Membuat Library	31
3.4.1. Library Database	31
3.4.2. Library Resultset	36
3.4.3. Library Controller	37

3.4.4. Library Model	39
3.4.5. Library View	42
3.5. Membuat Halaman Default	44
3.5.1. Membuat Controller (HomeController.php)	44
3.5.2. Membuat View (welcome.view.php)	46

BAB 4. Proyek Membuat Website Sekolah dengan Framework Sendiri 49

4.1. Mempersiapkan Database dan Tabel-Tabelnya	50
4.2. Membuat Halaman Pengunjung	54
4.2.1. Setting File Konfigurasi	54
4.2.2. Membuat Template Halaman Pengunjung (Front End)	56
4.2.3. Halaman Home	66
4.2.4. Halaman Buku Tamu	69
4.2.5. Halaman Siswa	74
4.2.6. Halaman Detail Siswa	77
4.2.7. Halaman Alumni	81
4.2.8. Halaman Detail Alumni	84
4.2.9. Halaman Guru	85
4.2.10. Halaman Detail Guru	88
4.2.11. Halaman Tentang Sekolah	91
4.2.12. Halaman Kontak Kami	93
4.2.13. Halaman Artikel (Berita)	96
4.2.14. Halaman Detail Artikel	98
4.2.15. Halaman Kategori Artikel	101
4.2.16. Halaman Detail Kategori	102

BAB 5. Membuat Halaman Administrator dgn Framework Sendiri	105
5.1. Setting File Konfigurasi untuk Administrator.....	108
5.2. Membuat Template Halaman Administrator.....	108
5.3. Dashboard (Halaman Utama Administrator).....	114
5.4. Halaman Login.....	120
5.5. Modul Buku Tamu.....	125
5.6. Modul Kategori Artikel.....	128
5.7. Modul Artikel.....	136
5.8. Modul Jurusan.....	147
5.9. Modul Siswa.....	148
5.10. Modul Alumni.....	163
5.11. Modul Guru.....	165
5.12. Modul Tentang Sekolah.....	166
5.13. Modul Kontak Kami.....	169
5.14. Modul User.....	170
5.15. Penutup.....	183

Halaman ini Sengaja Dikosongkan

www.bukulokomedia.com

A decorative floral pattern in shades of gray, featuring intricate scrollwork and floral motifs, positioned to the right of the 'BAB I' text.

BAB I

PENGERTIAN DAN KONSEP

BAB 1

Pengertian dan Konsep

Hai teman-teman. Salam olahraga!. Akhirnya kita sampai di Bab 1, silahkan pakai sabuk pengaman, karena perjalanan kita akan segera dimulai hehe. Jadi, di bab ini untuk awal-awal kita belajar mengenai pengertian dan konsep terlebih dahulu, kalau misalnya kita langsung koding aja tanpa tahu konsep, kan nanti bisa jadi suram/buta konsep. Oke, langsung saja kita mulai:

1.1 Bahasa Markup dan Bahasa Pemrograman

Menurut Wikipedia, bahasa pemrograman atau sering di istilahkan juga dengan bahasa komputer adalah **instruksi standar untuk memerintah komputer**. Bahasa pemrograman ini merupakan **suatu himpunan dari aturan sintaks dan semantik yang dipakai untuk mendefinisikan program komputer**. Bahasa ini memungkinkan seorang programmer dapat menentukan secara persis data mana yang akan diolah oleh komputer, bagaimana data akan disimpan, dan jenis langkah apa secara persis yang akan diambil dalam berbagai situasi.

Dari pengertian tersebut, dapat disimpulkan bahasa pemrograman adalah **bahasa yang menghubungkan antara kita dengan komputer**. Berikut adalah contoh-contoh bahasa pemrograman yang kita gunakan dalam membuat sebuah website:

1. HTML

HyperText Markup Language atau lebih singkatnya HTML adalah **bahasa markup yang digunakan untuk membuat sebuah halaman web**, menampilkan berbagai informasi di dalam sebuah penjelajah web Internet dan pemformatan hiperteks sederhana yang ditulis dalam berkas format ASCII agar dapat menghasilkan tampilan wujud yang ter-integrasi.

Dengan kata lain, berkas yang dibuat dalam perangkat lunak pengolah kata dan disimpan dalam format ASCII normal, sehingga menjadi halaman web dengan perintah-perintah HTML. Jadi, disini belum ada yang namanya for, if, while, do while dan lain-lain, karena di HTML ini hanya menjadi tampilan dasar atau tampilan depan dari sebuah website. Dengan HTML, kita bisa **membuat sebuah website statis**.

2. CSS

Cascading Style Sheet atau lebih singkatnya CSS merupakan aturan untuk mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam.

Jadi, dengan CSS ini kita bisa **membuat halaman website yang dari html tadi menjadi lebih indah**, dengan adanya CSS maka kita bisa menambahkan style pada HTML, sehingga tampilan bisa menjadi lebih enak dilihat. Makanya banyak orang yang bilang “you are CSS in my HTML” (jiah malah gombal).

3. ASP

Active Server Pages atau lebih singkatnya ASP adalah **salah satu bahasa pemrograman web untuk menciptakan halaman web yang dinamis**. ASP merupakan salah satu produk teknologi yang disediakan oleh **Microsoft**.

ASP di proses melalui web server dan hasil proses ini menghasilkan HTML yang akan dikirimkan melalui penjelajah web (browser). ASP bekerja pada web server dan merupakan server side scripting. Dengan adanya ASP, maka kita bisa mengubah website yang kita buat statis dari HTML tadi menjadi website dinamis.

4. XML

Extensible Markup Language atau lebih singkatnya XML adalah bahasa markup untuk keperluan umum yang disarankan oleh W3C **untuk membuat dokumen markup keperluan pertukaran data antar sistem yang beraneka ragam**.

XML merupakan kelanjutan dari HTML (HyperText Markup Language) yang merupakan bahasa standar untuk melacak Internet.

5. WSDL

Web Services Description Language atau lebih singkatnya WSDL adalah format XML yang diterbitkan untuk menerangkan web service. WSDL mendefinisikan:

- Pesan-pesan (baik yang abstrak dan kongkrit) yang dikirim ke dan menuju web service
- Koleksi-koleksi digital dari pesan-pesan (port type, antarmuka)
- Bagaimana port type yang ditentukan dijadikan wire protokol di mana servis ditempatkan

6. Perl

Perl adalah **bahasa pemrograman untuk segala keperluan**. Perl dikembangkan pertama kali oleh Larry Wall di mesin Unix. Perl dirilis pertama kali pada tanggal 18 Desember 1987 ditandai dengan keluarnya Perl 1. Pada versi-versi selanjutnya, Perl tersedia pula untuk berbagai sistem operasi varian Unix (SunOS, Linux, BSD, HP-UX), juga tersedia untuk sistem operasi seperti DOS, Windows, PowerPC, BeOS, VMS, EBCDIC, dan PocketPC.

Dukungan terhadap pemrograman berbasis obyek (OOP) ditambahkan pada Perl 5, yang pertama kali dirilis pada tanggal 31 Juli 1993. Proyek pengembangan Perl 6 dimulai pada tahun 2000, dan masih berlangsung hingga kini tanpa tanggal yang jelas kapan mau di rilis. Ini dikatakan sendiri oleh Larry Wall dalam satu pidatonya yang dikenal dengan seri The State of the Onion.

7. JavaScript

JavaScript adalah **bahasa skrip client side yang populer di Internet** dan dapat bekerja di sebagian besar web browser populer seperti Internet Explorer (IE), Mozilla Firefox, Netscape dan Opera.

Kode JavaScript dapat disisipkan dalam halaman web menggunakan tag `<script>`. Dengan adanya JavaScript ini, kita bisa membuat website yang kita buat menjadi lebih menarik dengan animasi-animasi yang bisa di buat menjadi interaktif, contohnya ketika kita mau login, kemudian muncul alert ketika kita kosongkan fieldnya, dan lain-lain.

8. SQL

Structured Query Language atau lebih singkatnya SQL adalah sebuah bahasa yang digunakan **untuk mengakses data dalam basis data relasional**. Bahasa merupakan bahasa standar yang digunakan dalam manajemen basis data relasional. Saat ini hampir semua server basis data yang ada mendukung bahasa ini untuk melakukan manajemen datanya.

9. PHP

Hypertext Preprocessor atau lebih singkatnya PHP adalah **bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML**. Dengan PHP, kita dapat membuat website menjadi dinamis. PHP banyak dipakai oleh para programmer website. PHP dapat digunakan untuk membangun sebuah CMS (Content Management System).

1.2 Apa itu Framework?

Nah, salah satu materi pelajaran kita adalah membuat website dinamis menggunakan framework sendiri. Untuk itu, kita harus tahu dulu apa pengertian dari framework itu sendiri. Menurut wikipedia ”Framework adalah suatu struktur konseptual dasar yang digunakan untuk memecahkan atau menangani suatu masalah kompleks.”.

Jadi, singkatnya framework adalah **wadah/kerangka kerja dari sebuah website yang akan kita bangun**. Dengan adanya kerangka kerja tersebut, maka kita bisa lebih mempercepat waktu dalam pembuatan website sekaligus memudahkan melakukan perbaikan dari sebuah website.

Berikut beberapa contoh framework yang sering digunakan oleh banyak orang:

1. CodeIgniter

CodeIgniter merupakan aplikasi open source yang berupa **framework PHP dengan model MVC (Model, View, Controller) untuk membangun website dinamis dengan menggunakan PHP**.

CodeIgniter memudahkan developer untuk membuat aplikasi web dengan cepat mudah dibandingkan dengan membuatnya dari awal. CodeIgniter dirilis pertama kali pada 28 Februari 2006. Versi stabil terakhir adalah versi 3.0.3.

2. CakePHP

CakePHP merupakan sebuah rapid development framework yang gratis dan open source untuk PHP. CakePHP adalah sebuah **framework atau kerangka kerja untuk membuat aplikasi CRUD (Create, Read, Update, Delete) berbasis bahasa pemrograman PHP**.

CakePHP juga menjadi salah satu framework pilihan yang memungkinkan seorang dpengembang web untuk membuat sebuah aplikasi dengan karakter pengembangan RAD (Rapid Application Development), yang memungkinkan untuk digunakan dan dikembangkan menjadi aplikasi lain yang lebih kompleks.

3. Laravel

Laravel merupakan web application framework berbasis PHP yang open source, menggunakan konsep Model–View–Controller (MVC). Laravel berada dibawah lisensi MIT License, dengan menggunakan GitHub sebagai tempat berbagi kode.



Desember 2013, **Laravel menempati PHP framework terpopuler** dan berada diatas PHP framework lain seperti Phalcon, Symfony2, CodeIgniter dan lainnya.

4. **Phalcon.**

Phalcon adalah salah satu framework PHP yang populer. Perbedaan Phalcon dari framework PHP yang lainnya adalah bahwa **Phalcon di implementasikan dalam bentuk PHP extension.**

Phalcon menyediakan extension, sehingga tidak perlu diinstal secara manual oleh pengguna.

5. **Yii Framework**

Yii adalah **framework PHP berbasis-komponen dan berkinerja tinggi untuk pengembangan aplikasi web berskala besar.**

Yii menyediakan reusability maksimum dalam pemrograman Web dan mampu meningkatkan kecepatan pengembangan secara signifikan. Nama Yii (dieja sebagai /i:/) singkatan dari “Yes it is!”.

6. **Zend**

Zend Framework (ZF) merupakan web application framework berbasis PHP open source, berorientasi objek kerangka aplikasi web yang di implementasikan di PHP 5 dan berlisensi di bawah Lisensi New BSD.

7. **PHPixie**

PHPixie dimulai sebagai **kerangka mikro dan secara bertahap berkembang menjadi salah satu kerangka kerja PHP FullStack paling populer, namun tetap mempertahankan kinerja tinggi.** Hal ini dikarenakan arsitektur ketat yang menghindari perangkat umum seperti ketergantungan pada metode statis, lingkup global, lajang dan antipatterns lainnya, dengan demikian juga memastikan bahwa kode mudah dibaca, debug, memperluas dan tes.

Bahkan semua komponen PHPixie mengandalkan cakupan unit test penuh. Ini adalah kerangka PHP yang tidak akan pernah berdiri di jalan Anda dan memberi Anda kontrol penuh atas aliran eksekusi, mudah untuk belajar dan mudah untuk menguasai.

8. **Symfony**

Symfony adalah sebuah framework lengkap yang **didesain untuk mengoptimalkan pengembangan aplikasi berbasis web dengan**

menyuguhkan beberapa fitur-fitur andalan.

Untuk pemula, Symfony mengelompokkan aturan-aturan bisnis aplikasi (business rules), logika server, dan tampilan presentasi. Symfony menyediakan bermacam-macam alat dan class-class yang ditujukan untuk memperpendek waktu pengembangan sebuah aplikasi web yang kompleks.

Sebagai tambahan, Symfony mengotomatisasi tugas-tugas umum sehingga pengembang dapat memfokuskan diri pada spesifikasi aplikasi secara keseluruhan. Sebagai hasil akhir, dengan kelebihan-kelebihan ini berarti bahwa tidak perlu lagi menemukan (reinvent the wheel) ketika sebuah aplikasi web baru akan dibangun.

1.3 Apa itu OOP?

1.3.1 Pengertian OOP

OOP (Object Oriented Programming) adalah **suatu metode pemrograman yang berorientasi kepada objek**. Tujuan dari OOP diciptakan adalah **untuk mempermudah pengembangan program dengan cara mengikuti model yang telah ada di kehidupan sehari-hari**.

Jadi, setiap bagian dari suatu permasalahan adalah objek. Nah, objek itu sendiri merupakan gabungan dari beberapa objek yang lebih kecil lagi. Saya ambil contoh mobil. Mobil adalah sebuah objek. Mobil itu sendiri terbentuk dari beberapa objek yang lebih kecil lagi seperti mesin, roda, spion, kursi, dan sebagainya. Mobil sebagai objek yang terbentuk dari objek-objek yang lebih kecil saling berhubungan, berinteraksi, berkomunikasi dan saling mengirim pesan kepada objek-objek yang lainnya.

Begitu juga dengan program, sebuah objek yang besar dibentuk dari beberapa objek yang lebih kecil, objek-objek itu saling berkomunikasi, dan saling berkirim pesan kepada objek yang lain.

1.3.2 Kenapa Harus OOP?

Kenapa harus OOP? Itu adalah pertanyaan yang sering ada, karena dengan OOP biasanya orang berpikir “kenapa menggunakan OOP, cuman bikin ribet aja”. Iya memang dengan menggunakan OOP pembuatan program akan menjadi lebih lama, tapi **menjadi lebih memiliki struktur**. Dengan menggunakan OOP, kita

jadi bisa **lebih mudah mengerjakan sebuah proyek website dengan tim kita**, karena OOP dari manapun konsepnya akan sama, sehingga Programmer yang mengetahui OOP akan dengan mudah bisa membaca kode kita dibandingkan dengan kode yang dijalankan secara Prosedural.

Selain itu, dengan OOP apabila terjadi error, kita hanya mengubah 1 class yang menyebabkan error tersebut dan tidak mempengaruhi class lain. Jadi **lebih mudah untuk proses perbaikan atau perubahan dalam aplikasi**.

1.3.3 Prosedural Vs OOP

Apabila dibandingkan antara prosedural dengan OOP, masing-masing memiliki kelebihan dan kekurangan, yaitu:

Kelebihan Prosedural:

- ✓ Efektif digunakan untuk menyelesaikan masalah kecil.
- ✓ Memperhatikan urutan langkah-langkah perintah secara sistematis, logis , dan tersusun berdasarkan algoritma yang sederhana

Kekurangan Prosedural:

- ✓ Tidak cocok untuk menyelesaikan masalah yang rumit, karena nantinya akan kesulitan menemukan solusi permasalahan ketika terjadi error.

Kelebihan OOP:

- ✓ Data dan fungsi dibungkus dalam class-class atau objek-objek.
- ✓ Efektif digunakan untuk menyelesaikan masalah besar, karena OOP terdiri dari class-class yang memisahkan setiap kode program menjadi kelompok-kelompok kecil, sesuai dengan fungsinya.
- ✓ Fungsi-fungsi algoritma yang terbagi menjadi beberapa class dapat memudahkan kita dalam memahami program, begitu juga ketika ada bug pada program, kita bisa lebih mudah menemukan penyebab errornya dibanding ketika menggunakan prosedural programming.
- ✓ Object dan class dapat digunakan berkali-kali, sehingga dapat menghemat space memori.

Kekurangan OOP:

- ✓ Programmer yang telah terbiasa membuat program dengan konsep Prosedural sering merasa kesulitan ketika dihadapkan dengan OOP. Biasanya “mereka”

bingung dengan banyaknya istilah-istilah dalam OOP.

Apabila perbandingan Prosedural dan OOP dalam bentuk tabel bisa dilihat sebagai berikut:

Prosedural	OOP
Fokus utama pada fungsi dan prosedur yang beroperasi pada data.	Menekankan pada data yang sedang beroperasi dan tidak fungsi atau prosedur.
Program besar terbagi dalam program unit kecil disebut fungsi.	Program dibagi ke dalam apa yang disebut objek.
Data dan fungsi diperlakukan sebagai entitas terpisah.	Data dan fungsi bersama diperlakukan sebagai entitas terpisah.
Data bebas bergerak di sekitar sistem dari satu fungsi lain.	Data tersembunyi dan tidak dapat diakses oleh fungsi eksternal.
Data bersifat pasif.	Objek-objek dalam OOP aktif.
Program desain dengan pendekatan " <i>Top Down</i> " yaitu tugas-tugas kompleks dipecah menjadi bagian yang lebih kecil, sampai sub-tugas tersebut mudah diimplementasikan.	Program desain pendekatan " <i>Bottom Up</i> " yaitu membuat prosedur2 untuk menyelesaikan tugas-tugas yang sederhana, lalu menggabungkan prosedur2 tsb dlm prosedur yg lebih kompleks, sampai fungsionalitas tercapai.

1.4 Apa itu MVC?

Model-View-Controller atau disingkat MVC adalah **sebuah metode untuk membuat sebuah aplikasi dengan memisahkan antara data (Model) dari tampilan (View) dan cara bagaimana memprosesnya (Controller).**

Dalam implementasinya, kebanyakan framework dalam aplikasi website berbasis arsitektur MVC. MVC merupakan suatu konsep yang cukup populer dalam pembangunan aplikasi web. MVC memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi, seperti manipulasi data, user interface, dan bagian yang menjadi pengontrol aplikasi.

Terdapat tiga jenis komponen yang membangun suatu pola MVC dalam suatu aplikasi yaitu:

1. View

View merupakan bagian yang menangani presentation logic. Pada suatu aplikasi web, bagian ini biasanya **berupa file template HTML** yang diatur oleh Controller. View berfungsi **untuk menerima dan merepresentasikan data kepada user.** Bagian ini tidak memiliki akses langsung terhadap bagian Model.

2. Model

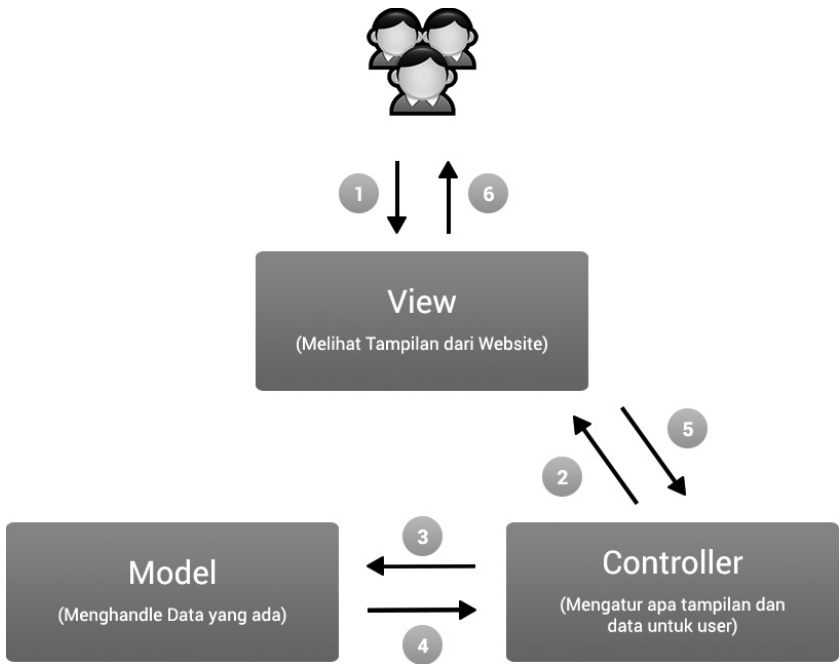
Model biasanya **berhubungan langsung dengan database untuk memanipulasi data** (Insert, Update, Delete, Search), menangani validasi dari bagian Controller, namun tidak dapat berhubungan langsung dengan bagian View.

3. Controller

Controller Merupakan bagian yang mengatur hubungan antara bagian Model dan bagian View. Controller berfungsi **untuk menerima request dan data dari user, kemudian menentukan apa yang akan diproses oleh aplikasi.**

Sebenarnya sistem dari MVC cukup sederhana, membedakan setiap bagian sesuai dengan fungsinya. Jadi, **halaman yang di request oleh user akan di kontrol oleh Controller, lalu Controller akan memberikan tampilan kepada user yaitu View, dan untuk bagian data akan di atur oleh Model.**

Kesimpulannya, Controller berurusan dengan kontrol, Model berurusan dengan data dan database, View adalah tampilan yang kita lihat. Untuk lebih jelasnya, alur MVC bisa dilihat pada gambar 1.1.



Gambar 1.1 Alur MVC (Model View Controller)