

Menguasai Codeigniter 4
Kasus Membuat Aplikasi PMB Kampus

Randi Adrika Putra



CV. LOKOMEDIA

Menguasai Codeigniter 4: Kasus Membuat Aplikasi PMB Kampus

Perpustakaan Nasional : Katalog Dalam Terbitan (KDT)

Penulis : Randi Adrika Putra

Menguasai Codeigniter 4: Kasus Membuat Aplikasi PMB Kampus

- Cet. I. - Yogyakarta : Penerbit Lokomedia, 2021

160 halaman; 14 x 21 cm

ISBN : 978-602-6231-27-7

Penerbit Lokomedia,

Cetakan Pertama : Februari 2021

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, 2021

Hak Cipta dilindungi oleh Undang-Undang

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

KATA PENGANTAR

Assalamu'alaikum warahmatullahi wabarakatuh

Puji syukur penulis haturkan kehadiran Allah SWT karena atas limpahan rahmat dan karunianya penulis dapat menyelesaikan buku sederhana ini sesuai dengan target waktu dan tanpa halangan yang berarti.

Melalui buku ini, kita akan belajar merancang dan membangun sebuah Aplikasi PMB (Pendaftaran Mahasiswa Baru) Kampus pada Bang Ambo University menggunakan CodeIgniter 4 dan Ajax. Aplikasi ini penulis buat menggunakan framework CodeIgniter 4.0.4 merupakan versi CodeIgniter 4 terbaru pada saat buku ini ditulis. Sistem operasi yang penulis gunakan pada saat membuat aplikasi ini adalah windows 10 64-bit.

Sebelum membuat aplikasi ini, kepada teman-teman pembaca diharapkan sudah mengerti dan memahami perintah-perintah dasar: HTML, CSS, Javascript, PHP (baik procedural maupun OOP), SQL (membuat dan memanipulasi data yang ada didalam database), dan CodeIgniter 4 (membuat CRUD dan sistem login sederhana).

Materi yang terdapat didalam buku ini dimulai dengan berkenalan dengan CodeIgniter 4, persiapan tools yang digunakan pada aplikasi, rancangan aplikasi, instalasi dan konfigurasi pada CodeIgniter 4, pengkodean aplikasi dari awal sampai aplikasi siap untuk digunakan, serta terakhir dilengkapi dengan hasil dan pembahasan dari aplikasi yang telah kita buat. Semua materi yang terdapat didalam buku ini, dibahas secara detail dengan menggunakan bahasa yang sederhana agar mudah dipahami oleh teman-teman pembaca.

Bagi pelajar/mahasiswa dan dosen buku ini sangat cocok dijadikan sebagai referensi penyusunan tugas akhir atau skripsi. Dan untuk pemula, buku ini juga sangat cocok dijadikan sebagai referensi belajar framework CodeIgniter 4 ketahap selanjutnya untuk merancang dan membangun sebuah project.

Kepada teman-teman pembaca, penulis mengucapkan banyak terimakasih telah meluangkan waktu dan pikirannya untuk membaca dan belajar bersama membuat Aplikasi Pendaftaran Mahasiswa Baru (PMB) Bang Ambo University dengan menggunakan framework CodeIgniter 4 dan Ajax melalui buku ini. Dan kepada teman-teman pembaca, penulis juga mempersilahkan untuk mengembangkan

aplikasi ini menjadi lebih efektif dan lebih baik lagi, dan jangan lupa untuk mencantumkan penulis sebagai sumber referensinya.

Buku ini penulis persembahkan kepada alm. Ayahanda dan Ibunda penulis. Semoga Allah swt mengampuni segala dosa ayahanda penulis (alm. Adrizal), merahmatinya, memaafkan kesalahan - kesalahannya, memuliakan kedudukannya, melapangkan kuburnya dan memasukkannya bersama orang - orang shaleh di surga-Nya. Aamiin.

Terimakasih kepada Ibunda penulis Enidarwati, atas cinta dan kasih sayang serta doa tulus yang tak pernah putus. Dan juga kepada ke tiga adik penulis yang menjadi penyemangat, Rizki Agustian, Rahman Syahri, dan Rubhil Adfal.

Dan juga tentunya buku ini penulis persembahkan tidak lain tidak bukan untuk teman-teman pembaca, semoga ilmu yang sedikit ini dapat bermanfaat bagi teman-teman pembaca dan untuk kemajuan ilmu pengetahuan dan teknologi di Indonesia. Aamiin.

Akhir kata, jika teman-teman pembaca memiliki masukan dan saran tentang buku ini, atau ingin menanyakan pertanyaan yang berkaitan dengan buku ini dapat menghubungi penulis melalui email penulis ke alamat: randiadrika@gmail.com.

Wassalamu'alaikum Warahmatullahi Wabarakatuh.

Dharmasraya, Oktober 2020

Randi Adrika Putra

DAFTAR ISI

BAB 1. Pengenalan Codeigniter 4	1
1.1. Berkenalan dengan Codeigniter	2
1.2. Untuk Siapa Codeigniter?.....	3
1.3. Server Requirements.....	3
1.4. Struktur Folder Codeigniter.....	4
1.5. Konsep Model View Controller.....	5
BAB 2. Tools Yang Digunakan	7
2.1. Mozilla Firefox.....	8
2.2. Sublime Text 3	8
2.3. XAMPP	8
2.4. Composer	10
2.5. Git	10
2.6. Codeigniter 4.....	12
2.7. AdminLTE 3.....	13
2.8. BizLand	13
2.9. Sweet Alert 2.....	13
2.10. Dompdf.....	14
2.11. Data Tables Server Side Processing	15
2.12. AJAX.....	15
BAB 3. Rancangan Aplikasi PMB	17
3.1. Penelitian Terkait.....	18
3.2. Use Case Diagram.....	19

3.3. Mekanisme Pendaftaran Mahasiswa Baru.....	20
3.4. Perancangan Database	21
BAB 4. Instalasi dan Konfigurasi.....	25
4.1. Instalasi Codeigniter 4 Melalui Composer.....	26
4.2. Konfigurasi pada File env.....	29
4.3. Membuat Template Layout Admin (Panitia)	29
4.4. Membuat Template Layout Member (Peserta)	32
4.5. Membuat Helper Sendiri.....	35
4.6. Validation.....	36
4.7. Encryption	42
4.8. Konfigurasi pada BaseController	42
4.9. Instalasi Dompdf.....	42
4.10. Menjalankan Local Development Server.....	44
BAB 5. Membuat Halaman Login	47
5.1. Controller Login.....	48
5.2. Model Login.....	51
5.3. View Login.....	51
BAB 6. Membuat Halaman Admin	57
6.1. Halaman Dashboard Admin	58
6.2. Halaman Kelola Informasi Pendaftaran.....	60
6.3. Halaman Kelola Data Fakultas.....	71
6.4. Halaman Kelola Data Prodi.....	83
6.5. Halaman Kelola Data Pendaftaran	94

BAB 7. Membuat Halaman Peserta	105
7.1. Halaman Pendaftaran	106
7.2. Halaman Dashboard Peserta	125
BAB 8. Halaman Utama Aplikasi PMB	133
8.1. Controller Home	134
8.2. View Home	135
BAB 9. Panduan Penggunaan Aplikasi PMB Kampus	139
9.1. Menjalankan Aplikasi	140
9.2. Halaman Utama Aplikasi	142
9.3. Halaman Login	143
9.4. Halaman Admin	143
9.4.1. Halaman Dashboard Admin	143
9.4.2. Halaman Kelola Informasi Pendaftaran	144
9.4.3. Halaman Kelola Data Fakultas	145
9.4.4. Halaman Kelola Data Prodi	145
9.4.5. Halaman Kelola Data Pendaftaran	147
9.5. Halaman Peserta	149
9.5.1. Halaman Pendaftaran Pembuatan Akun Peserta	149
9.5.2. Halaman Pendaftaran Tahap 1	151
9.5.3. Halaman Pendaftaran Tahap 2	151
9.5.4. Halaman Pendaftaran Tahap 3	152
9.5.5. Halaman Pendaftaran Tahap 4	152
9.5.6. Halaman Dashboard Peserta	155
9.5.7. Halaman Cetak Bukti Pendaftaran Online Peserta	156

Daftar Pustaka	158
Penutup	159
Tentang Penulis	160

BAB I



PENGENALAN CODEIGNITER

BAB 1

Pengenalan Codeigniter 4



CodeIgniter

1.1 Berkenalan dengan Codeigniter

CodeIgniter merupakan suatu kerangka kerja (*framework*) pengembangan aplikasi yang digunakan untuk membangun aplikasi berbasis *web* menggunakan bahasa pemrograman PHP (*Perl Hypertext Preprocessor*).

Kerangka kerja (*framework*) secara sederhana dapat diartikan kumpulan dari fungsi, prosedur dan *class* untuk tujuan tertentu yang sudah siap digunakan sehingga mempermudah dan mempercepat pekerjaan seorang *programmer*, tanpa harus membuat fungsi atau *class* dari awal.

Sasaran CodeIgniter adalah memudahkan kita sebagai seorang pengembang *website* untuk membuat aplikasi *website* dengan cepat mudah dibandingkan dengan membuatnya dari awal.

CodeIgniter juga menyediakan kumpulan pustaka (*libraries*) untuk tugas-tugas yang umumnya kita butuhkan, serta antarmuka yang sederhana (*simple interface*) dan struktur logis untuk mengakses pustaka (*libraries*). Codeigniter mempunyai *core* yang mudah diperpanjang bahkan diubah untuk bisa bekerja seperti yang kita inginkan.

Pada tahun 2014, CodeIgniter diakuisisi oleh *British Columbia Institute of Technology*, kemudian secara resmi diumumkan sebagai proyek yang dikelola oleh komunitas. Kemudian pada tahun 2019, *CodeIgniter Foundation* dibentuk untuk membantu memastikan masa depan dari *framework* CodeIgniter ini.

Codeigniter 4 rilis bertepatan dengan tanggal ulang tahun **James Parry** (tanggal

24 Februari) sebagai salah satu instruktur di *BCIT School of Computing and Academic Studies* yang akan memimpin proyek pengembangan CodeIgniter yang telah meninggal karena kanker paru-paru.

1.2 Untuk Siapa CodeIgniter?

CodeIgniter sangat tepat untuk kita jika:

1. Kita ingin sebuah *framework* dengan ukuran yang kecil.
2. Kita membutuhkan sebuah *framework* dengan kinerja yang luar biasa.
3. Kita ingin sebuah *framework* yang hampir tidak membutuhkan konfigurasi.
4. Kita ingin sebuah *framework* yang tidak mengharuskan kita untuk menggunakan *command line (CLI)*.
5. Kita ingin sebuah *framework* yang tidak mengharuskan kita untuk mematuhi aturan ketat dalam koding.
6. Kita tidak tertarik pada *monolithic libraries* skala besar seperti PEAR.
7. Kita tidak ingin dipaksa untuk belajar bahasa *templating*.
8. Kita menghindari kompleksitas, mendukung solusi yang sederhana.
9. Kita perlu dokumentasi yang jelas dan menyeluruh.

1.3 Server Requirements

Ada beberapa syarat yang harus kita penuhi untuk menggunakan CodeIgniter 4. Berikut ini adalah syarat- syarat yang harus ada di *server* kita:

1. PHP minimal versi **7.2** atau yang lebih baru, dengan **intl extension** dan **mbstring extension** terinstall.
2. Ekstensi **php-json**, **php-mysqlnd** dan **php-xml** harus aktif atau terinstall.
3. Untuk menggunakan **CURLRequest** harus menginstall **libcurl**.
4. *Database* yang didukung saat ini adalah:
 - MySQL versi 5.1 atau lebih baru.
 - PostgreSQL.
 - SQLite3.

1.4 Struktur Folder Codeigniter

Berbeda dengan struktur *folder* pada CodeIgniter 3. Secara *default*, direktori atau struktur *folder* pada CodeIgniter 4 adalah sebagai berikut:

```
/app
/system
/public
/writable
/tests
/docs
```

Folder /app

Folder /app merupakan tempat kode aplikasi yang dapat kita modifikasi sesuai dengan kebutuhan. Berikut ini adalah struktur *folder /app* beserta kegunaannya:

No	Folder	Keterangan
1	/config	Folder ini berfungsi sebagai tempat penyimpanan seluruh <i>file</i> konfigurasi
2	/controllers	Folder ini berisi <i>file controller</i> dari aplikasi yang akan kita bangun
3	/database	Folder ini berhubungan dengan <i>database migration</i> dan <i>seeds files</i>
4	/filters	Folder ini menyimpan <i>class</i> yang akan jalankan sebelum atau sesudah <i>controller</i>
5	/helpers	Folder ini menyimpan <i>file helpers</i> yang berisi fungsi-fungsi yang berdiri sendiri
6	/language	Folder ini berhubungan dengan dukungan beberapa bahasa terhadap aplikasi yang kita bangun
7	/libraries	Folder ini berhubungan dengan <i>class</i> yang berguna untuk pengembangan aplikasi kita
8	/models	Folder ini berhubungan dengan hubungan antara aplikasi ke <i>database</i> , <i>model</i> berfungsi sebagai representasi dari <i>business entities</i>
9	/thirdparty	Folder ini berisi <i>library</i> pihak ketiga yang ingin kita gunakan
10	/views	Folder ini berisi <i>file html</i> atau tampilan yang akan dilihat oleh <i>user /</i> yang akan tampil ke layar

Folder /system

Folder */system* merupakan *folder* yang berisi *file-file* yang membangun *framework* CodeIgniter 4. Folder */system* ini tidak boleh dimodifikasi.

Folder /public

Folder */public* menyimpan *file-file* yang bisa diakses oleh *browser* terhadap aplikasi *website* dan mencegah akses langsung ke *source code*. Folder ini berisi *file* *.htaccess*, *index.php* dan *assets* aplikasi seperti *CSS*, *javascript*, dan gambar.

Folder /writable

Folder */writable* berisi *file-file* yang harus disimpan oleh aplikasi, misalnya *cache*, *file* yang di *upload* oleh pengguna (*user*).

Folder /tests

Folder */tests* berfungsi untuk menyimpan semua *file test* kita. Terdapat *folder* */_support* yang berisi *class* dan beberapa utilitas lain yang akan memudahkan kita dalam membuat *file test*.

Folder /docs

Folder */docs* berisi *user guide* CodeIgniter 4 agar bisa diakses secara *offline*.

1.5 Konsep Model View Controller

Untuk membuat aplikasi, kita bisa menggunakan cara agar kode yang kita buat mudah ditemukan pada *file* yang tepat dan membuatnya mudah untuk dipelihara (*maintain*).

Seperti kebanyakan *framework website* lainnya, *CodeIgniter* menggunakan pola *Model, View, Controller (MVC)* untuk pengaturan *file*. Ini akan membuat data, tampilan, dan alur dari aplikasi terpisah menjadi beberapa bagian.

- **Models** berfungsi untuk manajemen data dari aplikasi dan membantu dalam melaksanakan *business rules* khusus yang aplikasi butuhkan.
- **Views** merupakan sebuah *file* yang terdapat sedikit atau bahkan tidak ada logika di dalamnya dan berfungsi untuk menampilkan informasi ke *user*.
- **Controllers** bertindak layaknya sebagai penghubung arus bolak balik antara tampilan dan penyimpanan data.

1.5.1 Views

Views merupakan *file* yang paling sederhana dari sebuah aplikasi dan biasanya berupa HTML dengan sedikit kode PHP. Jika terdapat kode PHP, itu hanya berfungsi untuk menampilkan konten dari sebuah variabel atau mengulangi beberapa item yang akan ditampilkan pada sebuah tabel.

Views memperoleh data dari *controllers* yang mengirimkan variabel ke *views* sebagai variabel yang dapat ditampilkan dengan perintah `echo`. Kita juga dapat menampilkan tampilan lain dalam satu tampilan halaman, sehingga untuk menampilkan *header* atau *footer* setiap tampilan halaman menjadi sangat mudah.

Views umumnya disimpan di folder `/app/Views`, kita juga dapat mengelompokkan tampilan menjadi beberapa folder. Misalnya, untuk tampilan halaman utama aplikasi dengan nama `index.php` dapat kita simpan pada folder `/app/Views/Home/index.php`.

Pengelompokan tampilan (*views*) tsb tidak menjadi masalah selama CodeIgniter dapat menemukan file tampilan (*views*) yang disimpan pada folder `/app/Views`.

1.5.2 Models

Models berperan untuk *maintain* sebuah *type* data untuk aplikasi, seperti data *users*, data postingan *blog*, data transaksi, dan lain-lain. Pada kasus ini, *models* memiliki dua tugas yaitu melaksanakan *business rules* pada data yang diambil dari atau dikirimkan ke *database*, dan mengurus penyimpanan data atau pengambilan data dari *database*.

Untuk *file models* dapat kita simpan pada folder `/app/Models`.

1.5.3 Controllers

Controllers biasanya bertugas menerima *input* dari *user* dan menentukan apa yang akan dilakukan oleh aplikasi terhadap *input* tersebut, seperti: mengirim data dari *input* ke *models* untuk disimpan atau mengambil data dari *models* untuk di tampilkan ke *views*. Jadi, *controllers* merupakan tempat dimana *developer* bisa mengatur siapa saja yang berhak mengakses dan mendapatkan data sesuai dengan format yang digunakan.

Untuk *file controllers* dapat kita simpan pada folder `app/Controllers/`.