

Laravel

Kompilasi Tentang apa itu Laravel

```
*  
* @var boolean  
*/  
define('PSI_INTERNAL_XML', false);  
  
if (version_compare("5.2", PHP_VERSION, ">")) {  
    die("PHP 5.2 or greater is required!!!");  
}  
if (!extension_loaded("pcre")) {  
    die("phpSysInfo requires the pcre extension to php in order to work  
    properly.");  
}  
  
require_once APP_ROOT.'/includes/autoloader.inc.php';  
  
// Load configuration  
require_once APP_ROOT.'/config.php';  
  
if (!defined('PSI_CONFIG_FILE') || !defined('PSI_DEBUG')) {  
    $tpl = new Template("/templates/html/error_config.html");  
    echo $tpl->fetch();  
    die();  
}
```

KATA PENGANTAR

Pertama-tama saya ucapkan puja dan puji syukur atas kehadiran Allah Subhanahuwata'ala yang telah memberikan kasih dan sayang-nya kepada kita, tidak lupa pula sholawat serta salam kita curahkan kepada Nabi kita Muhammad SAW beserta keluarga dan para sahabat Nabi. Sehingga penulis dapat menyelesaikan makalah yang berjudul “Website Menggunakan Framework Laravel” dengan baik. Penyusunan artikel Laravel ini dilaksanakan di Bandung

Penyusun

Bandung 12 November 2023

- **Kata Pengantar**
- **Daftar Isi**
- **Apa itu Laravel**
 - Pengertian Laravel
 - Sejarah Laravel
 - Kenapa Memilih Laravel
- **Instalasi**
 - Instalasi Text Editor (Visual Studio Code)
 - Instalasi XAMPP
 - Instalasi Composer
 - Instalasi Laravel 9
 - instalasi Snippets laravel (Visual Studio Code)
- **Intro Struktur Folder (Untuk Pembuatan Tabel Dinamis)**
 - Controllers
 - Public - Assets
 - Resources - View
 - Routes Web.php
- **Intro Blade Sebagai Templating Engine**
- **Praktik - Membuat Tabel Dinamis (Create, Read, Update, Delete)**
 1. Membuat Tabel (Html & Bootstrap)
 2. Membuat Database & Table (MySQL)
 3. Menampilkan Isi Database Ke Halaman Tabel Website
 4. Membuat Halaman Form Untuk Pengisian Data
 5. Membuat Fungsi Action Edit untuk mengupdate
 6. Membuat Fungsi Action Delete untuk menghapus
- **Penutup**

Latar Belakang Masalah

Laravel adalah framework bahasa pemrograman Hypertext Preprocessor (PHP) yang dibuat oleh Taylor Otwell dan pertama kali dirilis pada tanggal 9 Juni 2011. Laravel memiliki banyak fitur modern yang sangat membantu developer dalam membuat aplikasi. Beberapa fitur tersebut diantaranya adalah Bundles, Eloquent ORM (Object-Relational Mapping), Query Builder, Resource Controller, Blade, Migration, Middleware, dan Automatic Pagination. Laravel juga memiliki beberapa keunggulan yaitu menggunakan Command Line Interface (CLI) Artisan, menggunakan package manager PHP Composer, penulisan kode program lebih singkat, mudah dimengerti, dan ekspresif.

1.

Apa itu Laravel?

Pengertian, Fitur dan Kelebihannya

- by [M Ali Maksum](#) . June 10, 2022

Dalam mengembangkan sebuah website, salah satu hal penting yang harus diperhatikan adalah pemilihan framework yang tepat. Dengan begitu, website yang dihasilkan akan lebih maksimal karena fiturnya disesuaikan dengan kebutuhan. Salah satu framework yang direkomendasikan untuk pembuatan aplikasi ataupun website adalah framework Laravel.

Dengan menggunakan framework Laravel, biaya pembuatan website akan menjadi lebih hemat. Hal ini karena fitur-fitur yang dibuat merupakan fitur yang dibutuhkan saja. Tak hanya menghemat biaya, dari sisi waktu pun akan menjadi lebih hemat waktu.

Mau tau apa itu Laravel dan kelebihannya? Yuk simak sampai selesai.

Apa Itu Laravel?

Laravel adalah *framework* berbasis bahasa pemrograman [PHP](#) yang bisa digunakan untuk membantu proses pengembangan sebuah website agar lebih maksimal. Dengan menggunakan Laravel, website yang dihasilkan akan lebih dinamis.

Kehadiran *framework* Laravel menjadikan [bahasa pemrograman](#) PHP menjadi lebih powerful. Perlu kita ketahui bahwa kehadiran *framework* Laravel selalu menghadirkan fitur-fitur terbaru dibandingkan *framework* lainnya.

Framework Laravel menggunakan struktur MVC (*Model View Controller*). MVC merupakan model aplikasi yang memisahkan antara data dan tampilan berdasarkan komponen aplikasi. Dengan adanya model MVC, pengguna Laravel menjadi lebih mudah dalam mempelajari Laravel. Serta menjadikan proses pembuatan aplikasi berbasis website menjadi lebih cepat.



The PHP Framework for Web Artisans

Laravel is a web application framework with expressive, elegant syntax. We've already laid the foundation — freeing you to create without sweating the small things.

[GET STARTED](#)[WATCH LARACASTS](#)

St. Jude Children's
Research Hospital

ABOUT YOU



Disney



The New York Times



Bankrate

Laravel juga menyediakan fitur bawaan yang lengkap, salah satunya adalah fitur otentikasi. *Framework* yang satu ini cenderung berfokus pada level *end-user*. Di mana *framework* ini memiliki keunggulan pada kesederhanaannya, baik dalam sisi penulisan kode maupun tampilannya. Meskipun demikian, *framework* Laravel tetap dapat digunakan untuk membuat aplikasi berbasis website dengan fitur yang lengkap.

Karena keunggulan dan fleksibilitasnya dalam membantu proses pengembangan aplikasi berbasis website, banyak perusahaan dan pengembang yang memilih Laravel. Mulai dari perusahaan kecil, sampai perusahaan besar.

Fungsi Laravel

Sebagai *framework* yang memiliki fitur yang lengkap, Laravel berfungsi untuk membantu proses pembuatan website menjadi lebih maksimal. Selain itu, dengan menggunakan Laravel, perusahaan juga akan menghemat waktu dan biaya.

Hal ini karena nantinya hanya fitur yang sedang dibutuhkanlah yang akan dibuat. Sedangkan fitur yang belum dibutuhkan akan dibuat pada saat masa pengembangan website tersebut.

Kelebihan Laravel

Bagi kamu yang belum pernah mencoba Laravel, berikut ini adalah beberapa kelebihan dari Laravel.

Template ringan

Kelebihan yang pertama adalah template yang ringan. Dengan kelebihan ini, developer menjadi terbantu dalam membuat website yang lebih powerful. Template yang disediakan juga bisa dicustom dengan model tampilan yang kita inginkan. Sehingga pengembang akan terbantu dari sisi efisiensi waktu pembuatan.

Baca Juga: [15 Text Editor Terbaik untuk Coding dan Pemrograman](#)

Library yang lengkap

Laravel menyediakan banyak *library* yang bisa digunakan secara gratis. Hal ini tentu akan sangat membantu proses pembuatan website. Terlebih jika website yang dibuat adalah website dengan fitur yang kompleks.

Menggunakan model MVC

MVC merupakan satu fitur yang juga sangat membantu pada saat pengembangan website. Dengan begitu website yang dihasilkan memiliki file yang tersusun dengan rapi. Hal ini akan mempermudah proses pengembangan website tersebut.

Tool artisan

Tool *artisan* merupakan salah satu keunggulan Laravel yang disukai penggunanya. Tool ini berfungsi agar website yang sedang dikembangkan dapat berinteraksi dengan *framework* lainnya dengan bantuan *command line*. Adanya tool ini membuat para pengembang menjadi lebih mudah dalam melakukan kegiatan antar *framework* tanpa hambatan.

Modul bersifat individu dan independen

Terdapat banyak modul *library* yang dapat digunakan dan bersifat individu. Selain itu, ada juga yang bersifat independen. Adanya modul-modul ini kemudian membuat aplikasi yang dihasilkan lebih powerful.

Fitur-fitur Laravel

Selain memiliki banyak kelebihan, Laravel juga memiliki beberapa fitur-fitur yang harus kamu ketahui. Berikut ini beberapa fitur Laravel:

- **Authentication** : Fitur ini merupakan fitur yang dapat kamu gunakan untuk membuat website dengan otentikasi. Kamu tidak perlu menulis kode dari awal untuk menambahkan fitur otentikasi di website kamu.
- **Testing and Debugging** : Dalam Laravel sudah disediakan fitur untuk testing dan debugging sebuah website. Kamu tidak perlu khawatir lagi untuk melakukan testing dan debugging.
- **Routing** : Dengan fitur yang satu ini, kamu dapat membuat aplikasi yang dengan lebih mudah. Hal ini karena dalam *framework* ini, semua request akan dipetakan dengan menggunakan bantuan *route*. Kamu bisa memberikan nama *routing*, mengelompokkan, memfilter *routing* yang ada sesuai kebutuhan kamu.
- **Blade Template Engine**. *Framework* ini dilengkapi dengan fitur *blade* yang bisa digunakan untuk mendesain layout yang konsisten.
- **Composer** : Fitur ini akan membantu dalam menyempurnakan dan meningkatkan proses pengembangan website, serta mempermudah proses update-nya.

Sudah Tahu Apa itu Laravel?

Laravel adalah salah satu *framework* berbasis bahasa pemrograman PHP yang bisa digunakan untuk membantu proses pengembangan sebuah website. Sekian artikel mengenai Laravel, kamu bisa membaca dokumentasi Laravel yang lengkap di situs resminya. Buat para pengembang website yang membutuhkan hosting, silahkan gunakan cloud hosting terbaik dari Dewaweb. Sekian, salam sukses online!

Asset Blog Dewaweb

M Ali Maksum

<https://www.dewaweb.com/blog/apa-itu-laravel/>

2.

5 Kelebihan Framework Laravel untuk Pengembangan Web

BY [ANGGIE IRFANSYAH](#) 2023-01-06 2 MINS READ

Dalam mengembangkan sebuah website, ada banyak framework yang bisa digunakan. Framework yang paling banyak digunakan dalam pengembangan website adalah PHP.

Framework yang tepat dalam pengembangan website dapat membantu mempermudah pengembangan, sehingga tidak memakan waktu lebih lama. Selain itu, framework juga bisa menyederhanakan proses pembuatan dan menghasilkan performa yang lebih maksimal.

Salah satu framework yang sangat populer dalam pengembangan website adalah Laravel. Framework laravel sendiri dikenal dengan kesederhanaan, dan bisa menghasilkan aplikasi web yang powerful.



Ilustrasi framework Laravel

Apa itu Framework Laravel?

Untuk pengembang web, pasti sudah tidak asing lagi dengan framework PHP. Laravel adalah satu-satunya framework yang membantu memaksimalkan penggunaan PHP dalam pengembangan website.

PHP sendiri merupakan bahasa pemrograman yang sangat dinamis, namun sejak adanya Laravel, PHP menjadi lebih cepat, powerful, aman, dan simple.

Laravel pertama kali diluncurkan pada tahun 2011 dan mengalami pertumbuhan yang sangat eksponensial. Di tahun 2015, Laravel mendapatkan banyak bintang di GitHub. Kini framework Laravel menjadi salah satu yang paling populer di seluruh dunia.



ilustrasi microsoft excel

Kelebihan framework Laravel

1. Laravel bersifat Open Source

Laravel merupakan framework yang bersifat open source, bisa digunakan dengan gratis. Framework ini juga banyak digunakan karena developer dapat menggunakannya untuk dikembangkan lagi.

2. Laravel punya blade template

Selain open source, Laravel juga punya blade template. Blade template pada Laravel ini dapat membantu developer dalam membuat sebuah template menjadi lebih terstruktur serta dinamis. Dengan fitur ini, developer akan lebih hemat waktu dalam mengembangkan website.

3. Laravel punya dokumentasi yang lengkap.

Laravel banyak digunakan karena dokumentasinya yang lengkap dan selalu diperbaharui. Dokumentasi yang disediakan pada Laravel juga tergolong rapi dan jelas.

4. Laravel lebih aman

Laravel adalah framework PHP yang memberikan beberapa pilihan yang dapat digunakan agar aplikasi tetap aman. Salah satu sistem keamanan yang dimiliki Laravel adalah PDO yang bisa mencegah SQL Injection.

5. Laravel punya fitur migrasi database

Laravel juga punya fitur migrasi database. Ini merupakan salah satu fitur unggulan Laravel. Dengan migrasi database, developer tidak perlu membuat ulang struktur database yang baru karena sistem akan melakukan migrasi data secara langsung.

[Anggie Irfansyah](#)

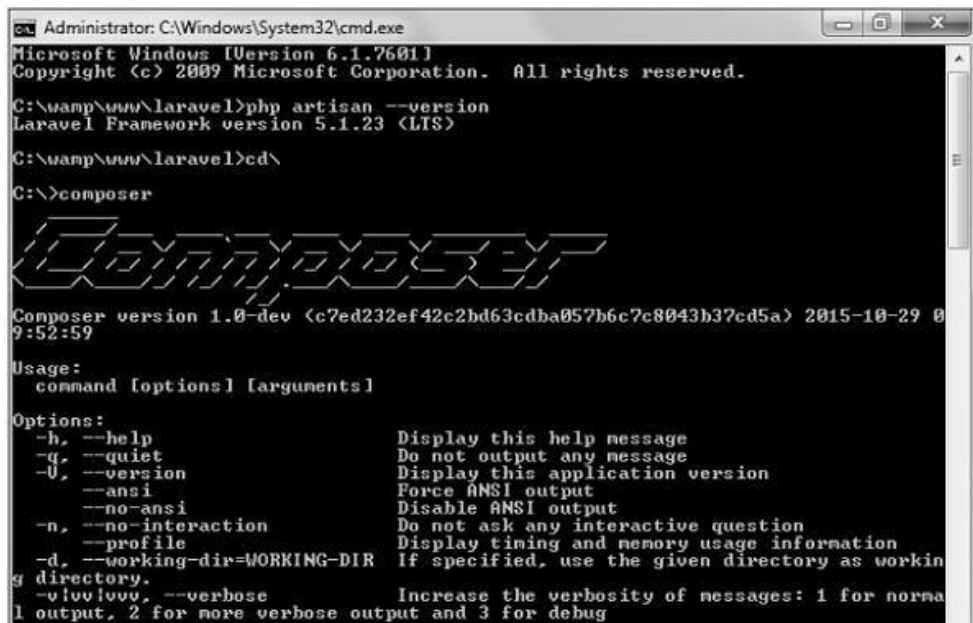
Eduparx adalah platform pembelajaran IT online nomor 1 di Indonesia yang menyediakan pelatihan berkualitas dan bersertifikat. Eduparx hadir sebagai solusi untuk meningkatkan kompetensi

masyarakat dalam mempelajari teknologi informasi dengan pelatihan dan produk online yang berkualitas dan dapat diakses dimana saja dan kapan saja.

<https://eduparx.id/blog/insight/development/5-kelebihan-framework-laravel/>

3.

Tutorial PHP Framework Laravel Part 1



```
Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\wamp\www\laravel>php artisan --version
Laravel Framework version 5.1.23 <LTS>

C:\wamp\www\laravel>cd\

C:\>composer

Composer version 1.0-dev (c7ed232ef42c2bd63cd8a057b6c7c8043b37cd5a) 2015-10-29 0
9:52:59

Usage:
  command [options] [arguments]

Options:
  -h, --help                Display this help message
  -q, --quiet               Do not output any message
  -U, --version             Display this application version
      --ansi               Force ANSI output
      --no-ansi            Disable ANSI output
  -n, --no-interaction      Do not ask any interactive question
      --profile            Display timing and memory usage information
  -d, --working-dir=WORKING-DIR If specified, use the given directory as workin
g directory.
      --verbose            Increase the verbosity of messages: 1 for norma
l output, 2 for more verbose output and 3 for debug
```

Laravel adalah sebuah PHP *Framework* yang dirancang untuk pengembang yang membutuhkan sebuah *toolkit* yang sederhana dan elegan untuk menciptakan aplikasi *web* berfitur lengkap. Laravel diciptakan oleh Taylor Otwell pada tahun 2011. Sebelum menggunakan Laravel ini, anda diharapkan telah menguasai beberapa bahasa pemrograman seperti HTML, *core* PHP, dan *advance* PHP. Hal tersebut agar dapat membantu anda dalam mengerjakan proyek dengan lebih mudah.

Pengenalan

Laravel

Laravel adalah MVC (*Model, View, Controller*) *framework* dengan *bundle*, migrasi dan Artisan CLI (*Command Line Interface*). Laravel merupakan *software open source* yang memiliki fitur-fitur untuk membantu pengembangan *web*. Jika anda akrab dengan *core* PHP dan *advance* PHP, Laravel akan membuat tugas anda jauh lebih mudah dan menghemat banyak waktu anda ketika anda berencana untuk mengembangkan sebuah *website* dari nol. Selain itu, *website* yang dibangun dengan menggunakan Laravel ini juga aman dari berbagai serangan yang terjadi pada situs *web*. Lalu, *script code* yang dibuat pada Laravel ini juga sangat rapih, karena setiap *script code* yang dibuat akan dipisah sesuai dengan fungsinya masing-masing, seperti *script code* untuk *front end* akan ditempatkan pada *blade engine template* dan untuk *back end* akan ditempatkan pada *controller*.

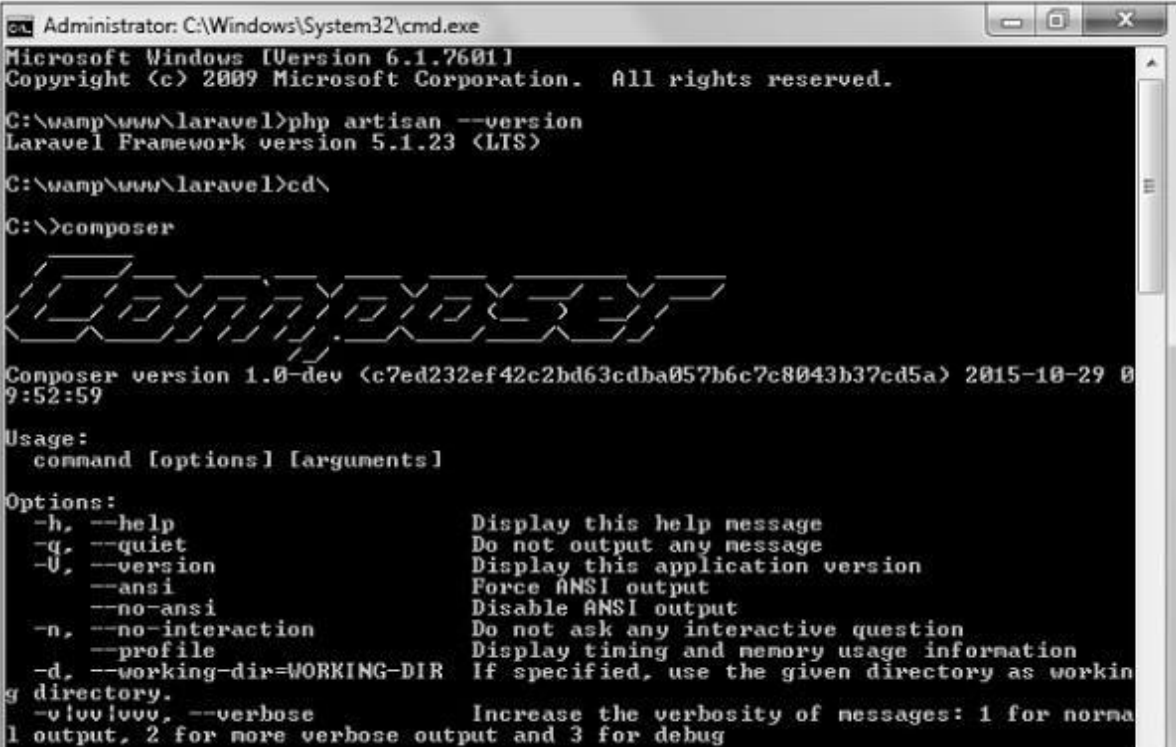
Instalasi

Laravel

Sebelum menggunakan Laravel, berikut langkah-langkah yang perlu anda lakukan:

1. Pertama-tama, anda harus meng-*install* aplikasi *composer*-nya terlebih dahulu dengan mengunjungi *link* berikut: <https://getcomposer.org/download/>
2. *Download* dan *install* aplikasi *composer* yang disebutkan pada langkah ke-1. Sebagai catatan, untuk tutorial ini menggunakan *Windows* sehingga, aplikasi *composer* yang di-*download* adalah yang versi *Windows*.
3. Setelah *composer* ter-*install*, periksa apakah instalasinya sudah benar dengan cara mengetikkan perintah “**composer**” di *command prompt* seperti yang ditunjukkan pada Gambar 1.

Gambar 1. Tampilan setelah perintah “composer” dieksekus



```
Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\wamp\www\laravel>php artisan --version
Laravel Framework version 5.1.23 <LIS>

C:\wamp\www\laravel>cd\

C:\>composer

Composer version 1.0-dev (c7ed232ef42c2bd63cdba057b6c7c8043b37cd5a) 2015-10-29 09:52:59

Usage:
  command [options] [arguments]

Options:
  -h, --help                Display this help message
  -q, --quiet               Do not output any message
  -V, --version             Display this application version
      --ansi               Force ANSI output
      --no-ansi            Disable ANSI output
  -n, --no-interaction      Do not ask any interactive question
      --profile            Display timing and memory usage information
  -d, --working-dir=WORKING-DIR If specified, use the given directory as working directory.
  -vvvv, --verbose          Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and 3 for debug
```

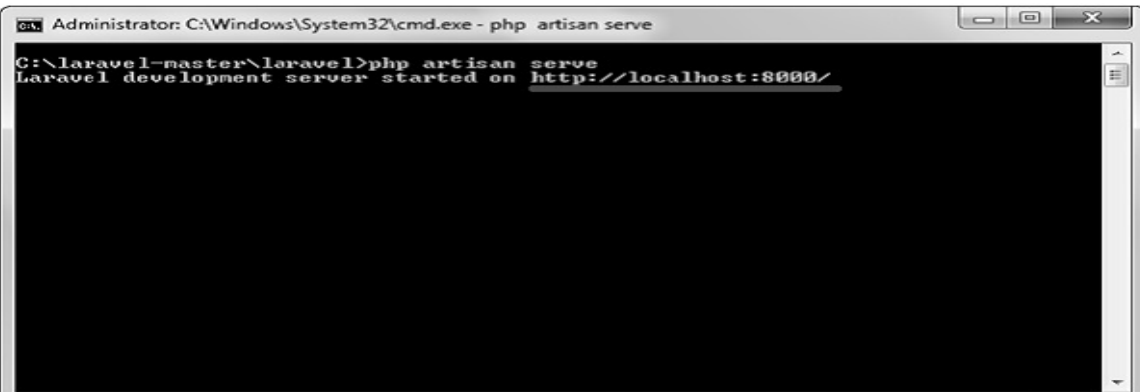
4. Buat sebuah direktori baru dimana saja pada sistem anda untuk proyek Laravel baru anda.
5. Setelah itu, masuk ke dalam *command prompt* dan pindahkan posisi inputan perintah ke direktori yang baru dibuat pada langkah ke-4 sebelumnya.
6. Ketik perintah berikut untuk meng-*install* Laravel pada sistem anda.

```
composer create-project laravel/laravel --prefer-dist
```

7. Setelah Laravel ter-*install* pada sistem, anda dapat memulainya dengan memasukkan perintah berikut pada *command prompt* di direktori Laravel-nya dan hasilnya dapat dilihat pada Gambar 2 di bawah ini.

```
php artisan serve
```

Gambar 2. Tampilan hasil perintah untuk memulai Laravel service



```
Administrator: C:\Windows\System32\cmd.exe - php artisan serve

C:\laravel-master\laravel>php artisan serve
Laravel development server started on http://localhost:8000/
```

8. *Copy* URL yang digaris-bawahi pada Gambar 2 di atas dan *paste* pada *browser* anda. Jika anda melihat layar seperti pada Gambar 3, maka anda telah berhasil meng-*install* Laravel.

Gambar 3. Tampilan jika Laravel berhasil ter-install



ABIAN MUTHRIF (1801423534)

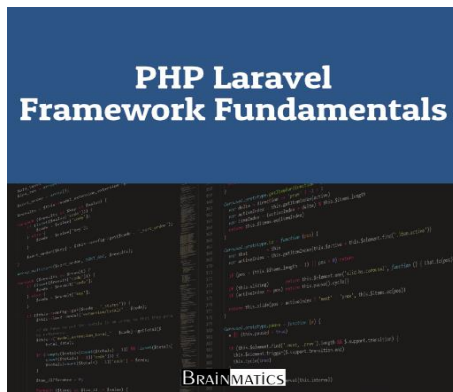
Student of Computer Science Program | School of Computer Science

<https://socs.binus.ac.id/2017/09/15/laravel51/>

4.

PHP Laravel Framework Fundamentals

User AvatarTeacher NURIS AKBARCategories FULL-STACK (WEB) DEVELOPER, POPULAR COURSE, SOFTWARE ENGINEERING



Berdasarkan W3Techs-Web Technology Surveys sejak Mei 2019, hampir 80% dari seluruh *website* di dunia menggunakan bahasa pemrograman PHP yang berjalan di sisi server (*server-side scripting*). Oleh karena itu, PHP, atau *hypertext preprocessor*, menjadi bahasa pemrograman *server-side* yang paling populer.

Seiring dengan tingginya permintaan pengembangan aplikasi web berbasis PHP menuntut developer untuk mempercepat proses pengembangannya. Para developer berlomba-lomba membuat *framework* berbasis PHP agar struktur dasar sistem yang dikembangkan dapat digunakan berulang kali. PHP Framework membuat proses pengembangan aplikasi web menjadi lebih cepat dan responsif. Beberapa contoh PHP Framework diantaranya adalah Laravel, Symfony, CodeIgniter, Yii 2, Phalcon, CakePHP, Zend, Slim, dsb. Adapun PHP Framework yang paling populer selama lima tahun terakhir ini dibandingkan dengan PHP Framework lainnya dengan komunitas pengembang yang besar dalam industri adalah Laravel.

Laravel merupakan framework yang tepat untuk pengembangan aplikasi dengan kebutuhan *backend* yang kompleks, baik skala kecil maupun besar. Instalasi Laravel juga semakin mudah dengan dikenalkannya Homestead (bundel all in one). Laravel merupakan sebuah PHP Framework yang penuh feature yang membantu melakukan *customize* aplikasi kompleks. Fitur-fitur tersebut diantaranya adalah migrasi data tanpa batas, mendukung arsitektur MVC (Model View Controller), keamanan, *routing*, melihat *template engine*, otentikasi dsb.

Training ini mempelajari tentang bagaimana struktur dan model MVC pada Laravel, instalasi dan pengaturan Laravel serta bagaimana membangun suatu aplikasi web. Setelah training selesai peserta diharapkan memiliki kemampuan dan kinerja dalam memperbaharui dan membangun aplikasi web maupun *website* yang lebih mudah, aman dan cepat menggunakan Laravel.

OBJECTIVES

1. Memahami konsep pembuatan aplikasi web dengan *framework* Laravel
2. Mampu mengikuti kaidah MVC pada *framework* Laravel
3. Mampu membuat aplikasi web dengan *framework* Laravel

AUDIENCE

Web Developer

PREREQUISITES

CONTENT

1. Introducing to Laravel
 - 1.1. Installing Laravel
 - 1.2. Creating the TODO Parrot Application
 - 1.3. Configuring Laravel Application
 - 1.4. Useful Development and Debugging Tools
 - 1.5. Testing Laravel Application with PHPUnit
2. Managing Project Controllers, Layout, Views, and Other Assets
 - 2.1. Creating First View
 - 2.2. Creating First Controller
 - 2.3. Managing Application Routes
 - 2.4. Introducing the Blade Template Engine
 - 2.5. Integrating Images, CSS and JavaScript
 - 2.6. Introducing Elixir
 - 2.7. Testing Views
3. Introducing Laravel Models
 - 3.1. Configuring Project Database
 - 3.2. Introducing the Eloquent ORM
 - 3.3. Creating First Model
 - 3.4. Introducing Migrations
 - 3.5. Defining Accessors, Mutators, and Methods
 - 3.6. Validating Models
 - 3.7. Creating a RESTful Controller
 - 3.8. Seeding the Database
 - 3.9. Finding Data
 - 3.10. Inserting New Records
 - 3.11. Updating Existing Records
 - 3.12. Deleting Records
 - 3.13. Introducing Query Builder
 - 3.14. Creating Sluggable URLs
 - 3.15. Testing Models
4. Model Relations, Scopes, and Other Advanced Features
 - 4.1. Introducing Relations
 - 4.2. Introducing One-to-One Relations
 - 4.3. Introducing the Belongs To Relation
 - 4.4. Introducing One-to-Many Relations
 - 4.5. Introducing Many-to-Many Relations
 - 4.6. Introducing Has Many Through Relations
 - 4.7. Introducing Polymorphic Relations
 - 4.8. Eager Loadin
 - 4.9. Introducing Scopes
5. Integrating Web Forms

- 5.1. Web Form Fundamentals
- 5.2. Creating a User Feedback Form
- 5.3. Creating New TODO Lists
- 5.4. Updating a TODO List
- 5.5. Deleting TODO Lists
- 5.6. Associating Tasks with Categories
- 5.7. Uploading Files
- 6. Introducing Middleware
 - 6.1. Introducing Laravel's Default Middleware
 - 6.2. Creating Middleware Solution
 - 6.3. Using Middleware Parameters
- 7. Authenticating and Managing Users
 - 7.1. Configuring Laravel Authentication
 - 7.2. Registering Users
 - 7.3. Retrieving the Authenticated User
 - 7.4. Restricting Access to Authenticated Users
 - 7.5. Restricting Forms to Authenticated Users
 - 7.6. Creating Route Aliases
 - 7.7. Adding Custom Fields to the Registration Form
 - 7.8. Restricting an Entire Application to Authenticated Users
- 8. Deploying, Optimizing and Maintaining Application
 - 8.1. Introducing the Laravel 5 Command Scheduler
 - 8.2. Optimizing Application
 - 8.3. Deploying Application
 - 8.4. Placing Application in Maintenance Mode
- 9. Creating a Restricted Administration Console
 - 9.1. Identifying Administrators
 - 9.2. Creating the Administration Controllers
 - 9.3. Restricting Access to the Administration Console
- 10. Introducing Lumen
 - 10.1. Creating First Lumen Application
 - 10.2. Creating a Status API
- 11. Introducing Events
 - 11.1. Creating an Event
 - 11.2. Broadcasting Events

<https://brainmatics.com/php-laravel-framework-fundamentals/>

Laravel merupakan framework website yang berbasis PHP yang open source, yang menggunakan model – view-control. Laravel adalah merupakan framework PHP yang di rilis dibawah lisensi MIT dengan menggunakan Git – Hub sebagai tempat berbagi kode.

Pada jaman sekarang, Menurut niagahoster dan teknologi.id menyatakan, framework Laravel menjadi salah satu framework PHP yang cukup populer digunakan. Tapi apakah kalian tahu apakah bagaimana sejarah perkembangan LARAVEL sebelum dapat berkembang hingga saat ini?

Framework Laravel dibuat oleh Taylor Otwell, proyek Laravel pertama kali dimulai pada April 2011, Framework Laravel pertama kali dijalankan dikarenakan Taylor Otwell kesulitan menemukan framework PHP yang up – to – date dengan versi PHP, lalu Taylor Otwell membuat framework Laravel dengan persyaratan harus menggunakan PHP dengan versi 5.3 keatas.

Pada Agustus 2009, PHP 5.3 resmi dirilis. Pada PHP versi 5.3 sudah support object oriented yang lebih baik dari versi sebelumnya, tetapi belum banyak framework yang support dengan versi tersebut, framework yang baru bisa support antara lain : Symfony, Zend, Kohana, Lithium dan CodeIgniter. Pada saat awal peluncuran PHP v.5.3, CodeIgniter yang lebih sering dikenal dengan CI adalah framework PHP yang mungkin paling terkenal pada masa itu, banyak developer framework PHP yang menyukai CI dikarenakan dokumentasinya terdapat dari banyak forum dan source code yang digunakan sederhana. Namun pada tahun 2011, Taylor Otwell mengungkapkan kekurangan yang dimiliki oleh CI antara lain ada beberapa fitur fungsional yang penting tidak support seperti autentikasi dan routing. Pada 9 Juni 2011, perilisan pertama Laravel versi beta 1, untuk mengatasi kekurangan yang ada CI tersebut.

Pada tahap Laravel versi 1 belum berbasis MVC seperti yang kita kenal sekarang, awalnya dibangun dengan fitur autentikasi dan Eloquent ORM (Object Relational Mapping) untuk operasi database, localization, model dan relationship, mekanisme routing yang sederhana, caching, session, views, module dan library, hingga HTML. Lalu Laravel pada bulan – bulan berikutnya terus dikembangkan dengan menambahkan method validasi, pagination, paket comand line installer, ekspansi Eloquent ORM (Object Relational Mapping), dan termasuk beberapa ratus unit testing untuk komponen framework. Lalu dalam kurun waktu kurang dari 6 bulan, Laravel berhasil memunculkan Laravel versi 2.

Pada tahap Laravel versi 2, dirilis ke developer pada 24 November 2011 dengan updatean pada beberapa fitur antara lain dukungan controller, engine template dan penggunaan invers. Dengan penambahan fitur controller ini, maka Laravel versi 2 ini sudah resmi menjadi framework yang berbasis MVC. Lalu perkembangan Laravel begitu cepat hingga kurang dari tiga bulan, Laravel berhasil meluncurkan Laravel versi 3.

Pada tahap Laravel versi 3, pertama kali dirilis pada 22 februari 2012, perkembangan pada versi 3 ini, lebih memfokuskan pada unit test integration, artisan command line interface, database migration, session driver dan database driver. Lalu setelah 5 bulan rilis, pembuat Laravel memutuskan untuk membuat ulang framework dari awal sebagai satu set paket yang didistribusikan melalui composer, karena keputusan tersebut perkembangan Laravel memakan waktu hingga satu tahun tiga bulan, hingga akhirnya Laravel berhasil memunculkan Laravel versi 4.

Pada tahap Laravel versi 4, terdapat perkembangan yang signifikan yang menampilkan arsitektur yang berbeda dari inti framework, tetapi pada perkembangan pada tahap ini ada beberapa developer yang menyebutkan “terlalu cepat” antara jadwal update dari versi satu ke versi yang lainnya, hal ini

mengakibatkan mereka harus bermigrasi ke versi baru dan kadang-kadang itu hanya tidak mungkin dengan aplikasi besar yang sudah dibangun pada arsitektur sebelumnya, karena tidak sedikit keluhan dari masyarakat, pada Laravel versi 4 berbeda dengan versi sebelumnya, Laravel 4 ini akan ada jadwal rilis secara teratur setiap 6 bulan untuk update (patch dan perbaikan bug). Dengan unit test yang meliputi 100% dari fungsi framework tersebut, Laravel 4 ini akan menjanjikan untuk menjadi stabil dan mudah di update secara online melalui composer.

Sources:

<https://sis.binus.ac.id/2019/04/05/laravel-framework/>)

<https://teknologi.id/insight/5-rekomendasi-framework-php-terbaik-dan-terpopuler/>)

<https://www.niagahoster.co.id/blog/framework-php-terbaik/>)

<https://jogjaweb.co.id/blog/sejarah-laravel>)

<https://sis.binus.ac.id/2020/03/23/sejarah-perkembangan-laravel/>

6.

Framework Laravel, sejarah dan cara kerjanya



[admin](#) 2 months ago [Belajar Laravel](#)

Laravel adalah sebuah kerangka kerja aplikasi web sumber terbuka yang ditulis dalam bahasa pemrograman PHP. Kerangka kerja ini dirancang untuk membantu pengembang dalam membangun aplikasi web dengan cepat dan efisien, menggunakan pola arsitektur MVC (Model-View-Controller). Laravel menyediakan banyak fitur seperti sistem routing, manajemen basis data, autentikasi, dan lainnya, yang membantu dalam proses pengembangan aplikasi.

Sejarah

Laravel pertama kali dirilis pada tahun 2011 oleh Taylor Otwell. Laravel dirancang untuk membantu pengembang dalam membangun aplikasi web dengan lebih efisien dan mudah. Itu mendapatkan popularitas karena fitur-fitur modern seperti sintaks ekspresif, ORM (Object-Relational Mapping) Eloquent, migrasi database, sistem rute yang kuat, dan banyak lagi. Sejak dirilis, Laravel telah mengalami beberapa pembaruan dan versi baru dengan peningkatan fitur dan kinerja.

Cara kerja

Laravel bekerja dengan menggunakan konsep Model-View-Controller (MVC), yang memisahkan berbagai komponen dalam pengembangan aplikasi web. Berikut adalah cara kerja umum Laravel:

1. **Model:** Model mewakili struktur data dan bisnis logika aplikasi. Ini berhubungan dengan basis data dan mengatur cara data disimpan, diambil, dan diperbarui. Laravel menggunakan Eloquent ORM untuk menyederhanakan interaksi dengan basis data, memungkinkan Anda untuk mengelola data dengan objek-objek PHP.
2. **View:** View menangani tampilan dan antarmuka pengguna. Dalam konteks Laravel, view adalah file-template yang menghasilkan HTML yang akhirnya akan ditampilkan kepada pengguna. Anda dapat menggunakan sintaks Blade untuk membuat tampilan yang dinamis dan ekspresif.
3. **Controller:** Controller mengatur aliran logika bisnis dan menghubungkan model dan view. Ketika pengguna melakukan permintaan HTTP, seperti membuka halaman web, controller menangkap permintaan tersebut dan mengambil tindakan yang sesuai. Ini bisa termasuk mengambil data dari model, mengolahnya, dan mengarahkan ke view yang tepat.
4. **Rute:** Rute dalam Laravel memetakan URL yang diterima ke tindakan-tindakan dalam controller. Rute menentukan bagaimana permintaan HTTP tertentu harus ditangani oleh aplikasi Anda.
5. **Middleware:** Middleware adalah serangkaian filter yang dapat diterapkan pada rute tertentu. Middleware memungkinkan Anda untuk melakukan tindakan sebelum atau setelah tindakan pada controller dijalankan, seperti otentikasi, validasi, dan lainnya.
6. **Artisan:** Artisan adalah alat baris perintah yang disediakan oleh Laravel untuk membantu dalam pengembangan. Anda dapat menggunakan Artisan untuk membuat migrasi basis data, menghasilkan kode, menjalankan tugas-tugas terjadwal, dan masih banyak lagi.

Secara umum, saat pengguna melakukan permintaan, rute mengarahkan permintaan tersebut ke controller yang sesuai. Controller berinteraksi dengan model untuk mengambil atau memanipulasi data, kemudian mengirimkan data tersebut ke view yang tepat. View menghasilkan tampilan yang akhirnya ditampilkan kepada pengguna melalui browser. Seluruh proses ini mengikuti prinsip-prinsip MVC untuk menjaga pemisahan dan keteraturan kode. Semoga bermanfaat.

<https://www.lembarilmu.com/post/detail/framework-laravel-sejarah-dan-cara-kerjanya>

7.

8 Alasan Mengapa Harus Memilih Laravel Menjadi Framework Untuk Website Kamu!



Aisyah Rosmalia January 15, 2020 No Comments

8 Alasan Mengapa Harus Memilih Laravel Menjadi Framework Untuk Website Kamu!

Membuat sebuah aplikasi teknologi, salah satunya website menjadi sebuah pekerjaan yang tidak bisa dibilang mudah. Namun, hal tersebut tak menghalangi para developer untuk terus mengembangkan teknologi. Dan di sisi lain, berhasil merebut perhatian masyarakat dunia dalam beberapa waktu terakhir ini. Melihat fenomena ini, akhirnya munculah tool-tool ataupun teknik baru yang diciptakan untuk memudahkan pekerjaan para developer untuk membangun website lebih mudah dan cepat. Tool itu adalah framework.

Kecermatan dalam memilih framework yang baik, menjadi hal yang cukup fatal. Framework yang baik adalah framework yang sesuai dengan kebutuhan aplikasi website yang akan kalian bangun. Tidak hanya itu, framework juga harus bisa menyederhanakan proses pembuatan dan menghasilkan performa aplikasi web yang kita bangun menjadi lebih maksimal.

Dan seperti yang sudah sebagian besar dari kalian ketahui, bahwa sudah cukup banyak framework PHP di dunia ini. Mulai dari **Codeigniter** yang ringan dan mudah dipelajari, **Zend Framework** yang terlalu kompleks dan enterprise, **Cake PHP** yang bisa dengan ajaib membuat prototype CRUD dalam hitungan detik, hingga yang lebih robust si **Yii Framework**. Dan pada dasarnya, semua framework yang ada sama saja, hanya saja dengan struktur yang berbeda.

Akhir-akhir ini, dari sekian banyak framework PHP yang muncul, ada satu framework PHP yang menjadi primadona bagi sebagian besar para developer nih, ya ya ya, ini dia **LARAVEL!**

Apa Sih Laravel Itu?

Laravel adalah salah satu framework PHP terbaik yang dikembangkan oleh Taylor Otwell. Sebagai sebuah framework PHP, Laravel hadir sebagai platform web development yang bersifat open source. Yang menarik dari laravel adalah sintaksnya yang ekspresif dan elegan yang dirancang khusus untuk memudahkan dan mempercepat proses web development.

Nah, sejak kemunculannya di tahun 2012, Laravel mengaku menjadi framework yang “clean and classy”, yang dimana pada tahun itu Laravel dianggap cukup aneh dan berbeda dimana framework ini muncul dengan membawa ideologi baru yang selama masa itu jarang diperhatikan, yaitu aspek “clean code” dan “expressiveness”. Dua aspek itu menjadikan laravel memiliki kode yang lebih singkat, mudah dimengerti, dan ekspresif. Jadi, hanya dengan membaca sekilas kode yang ditulis kalian sudah bisa menduga apa maksudnya tanpa perlu membaca dokumentasi.

Manfaat Laravel untuk Pengembangan Website

Dengan menyediakan beberapa keunggulan dibidang tool dan fitur unggulan lainnya, Laravel menawarkan beberapa keuntungan apabila kalian menggunakan laravel sebagai framework dasar dalam mengembangkan website kalian. Apa aja? Yuk kita lihat!

1. Website menjadi lebih mudah dikembangkan
2. Terdapat namespace dan tampilan yang membantu kalian dalam mengorganisir dan mengatur sumber daya website
3. Proses pengembangan menjadi lebih cepat sehingga menghemat waktu karena Laravel dapat dikombinasikan dengan beberapa komponen dari framework lain untuk semakin mengembangkan website yang kalian bangun

Nah, setelah tau gambaran umum mengenai Laravel. Yuk kita simak apa saja yang dimiliki Laravel hingga banyak developer yang menjadi jatuh hati dan memutuskan untuk memilih Laravel, dan mungkin kalian selanjutnya?

8 Alasan Mengapa Laravel menjadi Pilihan Terbaik!

1. Mudah dan Dokumentasi Lengkap

Sebuah framework yang baik adalah yang bisa dengan mudah digunakan dan tergolong awam untuk pengguna pemula. Di Laravel ini, meski kalian bukan seorang yang expert dalam bidang web development, akan tetapi sudah memiliki basic pemahaman PHP dengan baik, maka kamu tidak perlu ragu dan khawatir. Laravel akan dengan mudah bisa kalian pahami strukturnya, sebab dokumentasi yang mereka sediakan tergolong ke dalam dokumentasi yang sangat baik, rapi, mudah dan jelas. Sehingga akan sangat membantu kalian dalam memahami dan mempelajari Laravel.

2. Keamanan

Laravel akan memberikan kalian beberapa pilihan penting yang dapat digunakan untuk membuat aplikasi yang kalian bangun tetap aman. ORM Laravel menggunakan PDO yang dapat mencegah SQL Injection. Serta memiliki fitur proteksi csrf token, dan masih banyak hal lainnya lagi yang mampu menunjang keamanan aplikasi website kalian.

3. Syntax yang Cool dan Expressive

Laravel diciptakan dengan melakukan pendekatan yang berbeda dengan framework lainnya, misalnya dengan membuang parameter yang sifatnya flagging dan memilih untuk membuat fungsi yang berbeda. Pendekatan semacam ini lebih jelas, karena minimal satu parameter fungsi telah disederhanakan. Dan yang lebih ekstrimnya lagi, kalian bisa saja membuat aplikasi website tanpa Model, View, atau Controller.

4. Template Layout Ringan

Framework laravel menyediakan template layout yang ringan dan terdapat fitur untuk menambahkan template tambahan yang ringan. Sehingga para developer dapat membuat dan melakukan pengembangan menggunakan layout yang cukup powerful.

5. Mendukung Framework MVC (Model-View-Controller)

Laravel adalah salah satu dari sekian banyak framework PHP yang mendukung framework MVC. MVC ini menangani hampir keseluruhan bagian aplikasi dari Laravel. Dengan MVC, antara logika dan tampilan aplikasi dipisahkan, sehingga performa menjadi lebih optimal.

6. Mempunyai Banyak Library Object Oriented

Library Object Oriented disediakan dengan skala yang cukup banyak oleh laravel. Selain itu, laravel juga mempunyai berbagai macam library yang tidak akan kalian jumpai di dalam framework PHP yang tak kalah populer lainnya. Salah satu library yang paling penting adalah library otentikasi. Laravel memiliki library otentifikasi Bcrypt hashing dimana itu menjadi fitur yang tercanggih di bidang otentifikasi.

7. Tersedia Tool Artisan

Kalian para developer pasti terkadang membutuhkan interaksi antara laravel dengan framework lain menggunakan command line yang berguna untuk membuat dan menangani environment di aplikasi website kalian.

Nah, laravel mempunyai tool **Artisan** yang mana merupakan tool command line. Fungsi utama dari Artisan adalah melakukan tugas repetisi dan memproses permograman yang kompleks yang mana terkadang kalian sebagai developer terkadang mengabaikannya selama proyek aplikasi website kalian berjalan.

8. Memiliki Fitur Migration

Migration adalah salah satu fitur utama yang dimiliki oleh Laravel. Dengan *migration* memungkinkan kalian untuk mempertahankan struktur *database* yang kalian miliki tanpa harus membuatnya kembali. *Migration* memungkinkan kalian untuk menuliskan *code php* untuk mengatur *database*. Dengan *migration* pula kalian dapat mengembalikan beberapa perubahan terakhir yang kalian lakukan pada *database*.

Itu tadi informasi apa saja yang bisa kalian jadikan pertimbangan untuk memilih Laravel dan terangkum dalam 8 Alasan Mengapa kamu harus memilih laravel sebagai framework pembangun aplikasi website yang akan kalian bangun. Dan sebenarnya masih ada begitu banyak kelebihan, fitur, dan tool yang akan menunjang berbagai kemudahan untuk kalian. Kalian bisa langsung mencoba dan merasakan ke eleganan laravel. Selamat mencoba dan **Semangat Belajar!**

<https://argiaacademy.com/8-alasan-mengapa-harus-memilih-laravel-menjadi-framework-untuk-website-kamu/>

8.

Instalasi Laravel

[Proyek Laravel Pertama Anda](#)

Sebelum membuat proyek Laravel pertama Anda, Anda harus memastikan bahwa mesin lokal Anda telah menginstal PHP dan [Komposer](#). Jika Anda mengembangkannya di MacOS, PHP dan Komposer dapat diinstal dalam hitungan menit melalui [Laravel Herd](#). Selain itu, kami menyarankan [untuk menginstal Node dan NPM](#).

Setelah Anda menginstal PHP dan Composer, Anda dapat membuat proyek Laravel baru melalui create-project perintah Composer:

```
composer create-project laravel/laravel example-app
```

Atau, Anda dapat membuat proyek Laravel baru dengan menginstal installer Laravel secara global melalui Composer. Atau, jika Anda menginstal PHP dan Composer melalui [Laravel Herd](#), installer Laravel sudah tersedia untuk Anda:

```
composer global require laravel/installer
laravel new example-app
```

Setelah proyek dibuat, mulai server pengembangan lokal Laravel menggunakan server perintah Artisan CLI Laravel:

```
cd example-app
php artisan serve
```

Setelah Anda memulai server pengembangan Artisan, aplikasi Anda akan dapat diakses di browser web Anda di <http://localhost:8000>. Selanjutnya, Anda siap untuk [mulai mengambil langkah selanjutnya ke dalam ekosistem Laravel](#). Tentu saja, Anda mungkin juga ingin [mengkonfigurasi database](#).

Jika Anda ingin lebih awal dalam mengembangkan aplikasi Laravel Anda, pertimbangkan untuk menggunakan salah satu [starter kit](#) kami. Starter kit Laravel menyediakan perancangan orientasi backend dan frontend untuk aplikasi Laravel baru Anda.

[Laravel & Docker](#)

Kami ingin memulai Laravel semudah mungkin, apa pun sistem operasi pilihan Anda. Jadi, ada berbagai pilihan untuk mengembangkan dan menjalankan proyek Laravel di mesin lokal Anda. Meskipun Anda mungkin ingin menjelajahi opsi ini di lain waktu, Laravel menyediakan [Sail](#), solusi bawaan untuk menjalankan proyek Laravel Anda menggunakan [Docker](#).

Docker adalah alat untuk menjalankan aplikasi dan layanan dalam "kontainer" kecil dan ringan yang tidak mengganggu perangkat lunak atau konfigurasi yang diinstal pada mesin lokal Anda. Ini berarti Anda tidak perlu khawatir tentang konfigurasi atau pengaturan alat pengembangan yang rumit seperti server web dan database di mesin lokal Anda. Untuk memulai, Anda hanya perlu menginstal [Docker Desktop](#).

Laravel Sail adalah antarmuka baris perintah yang ringan untuk berinteraksi dengan konfigurasi Docker default Laravel. Sail memberikan titik awal yang bagus untuk membangun aplikasi Laravel menggunakan PHP, MySQL, dan Redis tanpa memerlukan pengalaman Docker sebelumnya.

Anda menjadi ahli Docker? Jangan khawatir! Semua sesuatu tentang Sail dapat diotomatiskan.

menggunakan `docker-compose.yml` file yang disertakan dengan Laravel

[Memulai MacOS](#)

Jika Anda mengembangkan di Mac dan [Docker Desktop](#) sudah terinstal, Anda dapat menggunakan perintah terminal sederhana untuk membuat proyek Laravel baru. Misalnya, untuk membuat aplikasi Laravel baru di direktori bernama "example-app", Anda dapat menjalankan perintah berikut di terminal Anda:

```
curl -s "https://laravel.build/example-app" | bash
```

Tentu saja, Anda dapat mengubah "aplikasi contoh" di URL ini sesuai keinginan Anda - pastikan saja nama aplikasi hanya berisi karakter alfanumerik, tanda hubung, dan garis bawah. Direktori aplikasi Laravel akan dibuat di dalam direktori tempat Anda menjalankan perintah.

Instalasi Sail mungkin memerlukan waktu beberapa menit sementara wadah aplikasi Sail dibuat di mesin lokal Anda.

Setelah proyek dibuat, Anda dapat menavigasi ke direktori aplikasi dan memulai Laravel Sail. Laravel Sail menyediakan antarmuka baris perintah sederhana untuk berinteraksi dengan konfigurasi Docker default Laravel:

```
cd example-app  
./vendor/bin/sail up
```

Setelah wadah Docker aplikasi dimulai, Anda dapat mengakses aplikasi di browser web Anda di: <http://localhost>.

Untuk terus mempelajari lebih lanjut tentang Laravel Sail, lihat [dokumentasi lengkapnya](#).

Memulai di Windows

Sebelum kita membuat aplikasi Laravel baru di mesin Windows Anda, pastikan untuk menginstal [Docker Desktop](#). Selanjutnya, Anda harus memastikan bahwa Subsistem Windows untuk Linux 2 (WSL2) diinstal dan diaktifkan. WSL memungkinkan Anda menjalankan executable biner Linux secara asli di Windows 10. Informasi tentang cara menginstal dan mengaktifkan WSL2 dapat ditemukan dalam [dokumentasi lingkungan pengembang Microsoft](#).

Setelah menginstal dan mengaktifkan WSL 2, Anda harus memastikan bahwa Docker Desktop [dikonfigurasi untuk menggunakan backend WSL2](#).

Selanjutnya, Anda siap membuat proyek Laravel pertama Anda. Luncurkan [Terminal Windows](#) dan mulai sesi terminal baru untuk sistem operasi WSL2 Linux Anda. Selanjutnya, Anda dapat menggunakan perintah terminal sederhana untuk membuat proyek Laravel baru. Misalnya, untuk membuat aplikasi Laravel baru di direktori bernama "example-app", Anda dapat menjalankan perintah berikut di terminal Anda:

```
curl -s https://laravel.build/example-app | bash
```

Tentu saja, Anda dapat mengubah "aplikasi contoh" di URL ini sesuai keinginan Anda - pastikan saja nama aplikasi hanya berisi karakter alfanumerik, tanda hubung, dan garis bawah. Direktori aplikasi Laravel akan dibuat di dalam direktori tempat Anda menjalankan perintah.

Instalasi Sail mungkin memerlukan waktu beberapa menit sementara wadah aplikasi Sail dibuat di mesin lokal Anda.

Setelah proyek dibuat, Anda dapat menavigasi ke direktori aplikasi dan memulai Laravel Sail. Laravel Sail menyediakan antarmuka baris perintah sederhana untuk berinteraksi dengan konfigurasi Docker default Laravel:

```
cd example-app  
./vendor/bin/sail up
```

Setelah wadah Docker aplikasi dimulai, Anda dapat mengakses aplikasi di browser web Anda di: <http://localhost>.

Untuk terus mempelajari lebih lanjut tentang Laravel Sail, tinjau [dokumentasi lengkapnya](#).

Berkembang Dalam WSL2

Tentu saja, Anda harus bisa memodifikasi file aplikasi Laravel yang dibuat dalam instalasi WSL2 Anda. Untuk mencapai hal ini, kami merekomendasikan penggunaan editor [Visual Studio Code](#) Microsoft dan ekstensi pihak pertama untuk [Pengembangan Jarak Jauh](#).

Setelah alat-alat ini terinstal, Anda dapat membuka proyek Laravel apa pun dengan menjalankan perintah `code` dari direktori root aplikasi Anda menggunakan Terminal Windows.

Memulai Di Linux

Jika Anda mengembangkan di Linux dan [Docker Compose](#) sudah terinstal, Anda dapat menggunakan perintah terminal sederhana untuk membuat proyek Laravel baru.

Pertama, jika Anda menggunakan Docker Desktop untuk Linux, Anda harus menjalankan perintah berikut. Jika Anda tidak menggunakan Docker Desktop untuk Linux, Anda dapat melewati langkah ini:

```
docker context use default
```

Kemudian, untuk membuat aplikasi Laravel baru di direktori bernama "example-app", Anda dapat menjalankan perintah berikut di terminal Anda:

```
curl -s https://laravel.build/example-app | bash
```

Tentu saja, Anda dapat mengubah "aplikasi contoh" di URL ini sesuai keinginan Anda - pastikan saja nama aplikasi hanya berisi karakter alfanumerik, tanda hubung, dan garis bawah. Direktori aplikasi Laravel akan dibuat di dalam direktori tempat Anda menjalankan perintah.

Instalasi Sail mungkin memerlukan waktu beberapa menit sementara wadah aplikasi Sail dibuat di mesin lokal Anda.

Setelah proyek dibuat, Anda dapat menavigasi ke direktori aplikasi dan memulai Laravel Sail. Laravel Sail menyediakan antarmuka baris perintah sederhana untuk berinteraksi dengan konfigurasi Docker default Laravel:

```
cd example-app  
./vendor/bin/sail up
```

Setelah wadah Docker aplikasi dimulai, Anda dapat mengakses aplikasi di browser web Anda di: <http://localhost>.

Untuk terus mempelajari lebih lanjut tentang Laravel Sail, lihat [dokumentasi lengkapnya](#).

[Memilih Layanan Layar Anda](#)

Saat membuat aplikasi Laravel baru melalui Sail, Anda dapat menggunakan variabel string kueri untuk memilih layanan mana yang harus dikonfigurasi dalam file aplikasi baru Anda `docker-compose.yml`. Layanan yang tersedia antara lain `mysql`, `pgsql`, `mariadb`, `redis`, `memcached`, `meilisearch`, `minio`, `selenium`, dan `mailpit`:

```
curl -s "https://laravel.build/example-app?with=mysql,redis" | bash
```

Jika Anda tidak menentukan layanan mana yang ingin Anda konfigurasikan, tumpukan default mysql, redis, meilisearch, mailpit, dan seleniومان dikonfigurasi.

Anda dapat menginstruksikan Sail untuk instal [Devcontainer](#) default dengan menambahkan devcontainerparameter ke URL:

```
curl -s "https://laravel.build/example-app?with=mysql,redis&devcontainer" | bash
```

[Konfigurasi Awal](#)

Semua file konfigurasi untuk framework Laravel disimpan di configdirektori. Setiap opsi didokumentasikan, jadi silakan melihat-lihat file dan memahami opsi yang tersedia untuk Anda.

Laravel hampir tidak memerlukan konfigurasi tambahan. Anda bebas untuk mulai berkembang! Namun, Anda mungkin ingin meninjau config/app.phpfile dan dokumentasinya. Ini berisi beberapa opsi seperti timezonedan localeyang mungkin ingin Anda ubah sesuai dengan aplikasi Anda.

[Konfigurasi Berbasis Lingkungan](#)

Karena banyak nilai opsi konfigurasi Laravel dapat bervariasi tergantung pada apakah aplikasi Anda berjalan di mesin lokal atau di server web produksi, banyak nilai konfigurasi penting yang ditentukan menggunakan file .envyang ada di root aplikasi Anda.

File Anda .envtidak boleh dikomit ke kontrol sumber aplikasi Anda, karena setiap pengembang/server yang menggunakan aplikasi Anda mungkin memerlukan konfigurasi lingkungan yang berbeda. Selain itu, hal ini akan menjadi risiko keamanan jika penyusup mendapatkan akses ke repositori kontrol sumber Anda, karena kredensial sensitif apa pun akan terekspos.

Untuk informasi lebih lanjut tentang .envkonfigurasi berbasis file dan lingkungan, lihat [dokumentasi konfigurasi lengkap](#).

Basis Data & Migrasi

Sekarang setelah Anda membuat aplikasi Laravel, Anda mungkin ingin menyimpan beberapa data dalam database. Secara default, file konfigurasi aplikasi Anda `.env` menentukan bahwa Laravel akan berinteraksi dengan database MySQL dan akan mengakses database di 127.0.0.1. Jika Anda mengembangkan di macOS dan perlu menginstal MySQL, Postgres, atau Redis secara lokal, Anda mungkin merasa nyaman menggunakan [DBngin](#).

Jika Anda tidak ingin menginstal MySQL atau Postgres di mesin lokal Anda, Anda selalu dapat menggunakan database [SQLite](#). SQLite adalah mesin database yang kecil, cepat, dan mandiri. Untuk memulai, perbarui file konfigurasi Anda untuk menggunakan driver database `.env` Laravel `sqlite`. Anda dapat menghapus opsi konfigurasi database lainnya:

```
DB_CONNECTION=sqlite
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=laravel
DB_USERNAME=root
DB_PASSWORD=
```

Setelah Anda mengonfigurasi database SQLite, Anda dapat menjalankan migrasi database aplikasi Anda, yang akan membuat tabel database aplikasi Anda:

```
php artisan migrate
```

Jika database SQLite tidak ada untuk aplikasi Anda, Laravel akan menanyakan apakah Anda ingin database tersebut dibuat. Biasanya, file database SQLite akan dibuat di `database/database.sqlite`.

Konfigurasi Direktori

Laravel harus selalu disajikan dari root "direktori web" yang dikonfigurasi untuk server web Anda. Anda tidak boleh mencoba menyajikan aplikasi Laravel dari subdirektori "direktori web". Mencoba melakukannya dapat mengekspos file sensitif yang ada dalam aplikasi Anda.

Dukungan IDE

Anda bebas menggunakan editor kode apa pun yang Anda inginkan saat mengembangkan aplikasi Laravel; namun, [PhpStorm](#) menawarkan dukungan ekstensif untuk Laravel dan ekosistemnya, termasuk [Laravel Pint](#).

Selain itu, plugin [Laravel Idea](#) PhpStorm yang dikelola komunitas menawarkan berbagai penambahan IDE yang bermanfaat, termasuk pembuatan kode, penyelesaian sintaksis Eloquent, penyelesaian aturan validasi, dan banyak lagi.

[Langkah selanjutnya](#)

Sekarang setelah Anda membuat proyek Laravel, Anda mungkin bertanya-tanya apa yang harus dipelajari selanjutnya. Pertama, kami sangat menyarankan agar Anda memahami cara kerja Laravel dengan membaca dokumentasi berikut:

- [Permintaan Siklus Hidup](#)
- [Konfigurasi](#)
- [Struktur Direktori](#)
- [Paling depan](#)
- [Wadah Layanan](#)
- [Fasad](#)

Bagaimana Anda ingin menggunakan Laravel juga akan menentukan langkah selanjutnya dalam perjalanan Anda. Ada berbagai cara untuk menggunakan Laravel, dan kita akan mengeksplorasi dua kasus penggunaan utama untuk framework tersebut di bawah.

Anda mengenal Laravel? Kunjungi [Laravel Bootcamp](#) untuk terinspirasi tentang kerangka kerja.

Sementara kami memandu Anda dalam membangun aplikasi Laravel pertama Anda.

[Laravel Kerangka Tumpukan Penuh](#)

Laravel dapat berfungsi sebagai kerangka tumpukan penuh. Yang kami maksud dengan kerangka kerja "tumpukan penuh" adalah Anda akan menggunakan Laravel untuk merutekan permintaan ke aplikasi Anda dan merender frontend Anda melalui [templat Blade](#) atau teknologi hibrid aplikasi satu halaman seperti [Inertia](#). Ini adalah cara paling umum untuk menggunakan framework Laravel, dan menurut kami, cara paling produktif untuk menggunakan Laravel.

Jika ini adalah cara Anda berencana menggunakan Laravel, Anda mungkin ingin melihat dokumentasi kami tentang [pengembangan frontend](#) , [perutean](#) , [tampilan](#) , atau [Eloquent ORM](#) . Selain itu, Anda mungkin tertarik mempelajari paket komunitas seperti [Livewire](#) dan [Inertia](#) . Paket-paket ini memungkinkan Anda untuk menggunakan Laravel sebagai kerangka full-stack sambil menikmati banyak manfaat UI yang disediakan oleh aplikasi JavaScript satu halaman.

Jika Anda menggunakan Laravel sebagai kerangka full stack, kami juga sangat menyarankan Anda mempelajari cara mengkompilasi CSS dan JavaScript aplikasi Anda menggunakan [Vite](#) .

Jika Anda ingin lebih cepat dalam membangun aplikasi Anda, lihat salah satu [starter kit aplikasi](#) resmi kami.

[Laravel Bagian Belakang API](#)

Laravel juga dapat berfungsi sebagai backend API untuk aplikasi satu halaman JavaScript atau aplikasi seluler. Misalnya, Anda mungkin menggunakan Laravel sebagai backend API untuk aplikasi [Next.js](#) Anda . Dalam konteks ini, Anda dapat menggunakan Laravel untuk menyediakan [otentikasi](#) dan penyimpanan/pengambilan data untuk aplikasi Anda, sekaligus memanfaatkan layanan canggih Laravel seperti antrian, email, notifikasi, dan banyak lagi.

Jika ini adalah cara Anda berencana menggunakan Laravel, Anda mungkin ingin melihat dokumentasi kami tentang [perutean](#) , [Laravel Sanctum](#) , dan [Eloquent ORM](#) .

Butuh langkah awal untuk melakukan scaffolding pada backend Laravel dan frontend Next.js?

Anda? Laravel Breeze menawarkan [tumpukan API](#) serta [implementasi frontend Next.js](#) sehingga

Anda dapat memulai dalam hitungan menit.

Butuh langkah awal untuk melakukan scaffolding pada backend Laravel dan frontend Next.js? Anda? Laravel Breeze menawarkan [tumpukan API](#) serta [implementasi frontend Next.js](#) sehingga Anda dapat memulai dalam hitungan menit.

<https://laravel.com/docs/10.x/installation>

Pendahuluan

Jika sebelumnya kalian sudah membuat view sebagai respon menampilkan data. Sekarang saatnya untuk mengenal Blade. Blade adalah *Template Engine* Laravel. Pada dasarnya, blade adalah view, namun blade dengan blade kita memiliki sintak tambahan yang dapat membantu kita menampilkan data. Dan yang paling menarik adalah fitur *Template Inheritance*.

Isi

Bagaimana sih menggunakan blade? cukup menambahkan ekstensi .blade.php pada file yang digunakan Pada folder siswa , rename file index.php menjadi index.blade.php. Sangat mudah kan menggunakan file blade, lalu lakukan hal yang sama dengan file homepage dan about menjadi about.blade.php dan homepage.blade.php

Sejauh ini, kita sudah memiliki 3 buat route, seperti terlihat berikut ini:

```
Route::get('/', function () {
    return view('pages/homepage');
});
```

```
Route::get('about', function(){
    return view('pages/about');
});
```

```
Route::get('/siswa', function(){
    $siswa = ['Arba Nugraha', 'Riki Amudra', 'Mahsa Vania Salsabila'];
    return view('siswa.index', compact('siswa'));
});
```

Untuk setiap route diatas, kita memberikan respon berupa view. Pada aplikasi sesungguhnya, biasanya ada sebuah file view yang berfungsi sebagai master template. Jadi, file view ini berfungsi untuk mengatur layout dari aplikasi yang kita buat. Di file template tersebut, kita bisa memanggil semua resource yang diperlukan , misalnya file style.css, skrip javascript, bahkan bisa memanggil framework UI seperti bootstrap.

Untuk membuat master template, buat file bernama template.blade.php kemudian simpan di laravelapp/resources/views . Ikuti code dibawah ini:

```
<!DOCTYPE html>

<html lang="en">

<head>
```

```

<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<meta http-equiv="X-UA-Compatible" content="ie=edge">
<title>LaravelApp</title>
</head>
<body>
    @yield('main')

    @yield('footer')
</body>
</html>

```

Keterangan Perintah @yield adalah perintah blade. Gunanya untuk menandai bahwa pada bagian itu akan ditempati potongan kode yang biasa disebut section. Dalam hal ini nama section tersebut adalah main dan footer. Kita akan membuat section main pada tutorial selanjutnya.

Untuk selanjutnya kita membuat Child Page. Child Page adalah Child Template atau template anak untuk masing - masing view. Gunanya agar master template yang sudah kita buat sebelumnya bisa berfungsi.

Ikuti langkah berikut untuk membuat child template untuk masing masing route yang kita buat. Pada file laravelapp/pages/homepage.blade.php ubah isinya menjadi:

```

@extends('template')

@section('main')
<div id="homepage">
<h2>Homepage</h2>
<p>Selamat belajar laravel!</p>
</div>

@endsection

@section('footer')
<div id='footer'>
    <p>&copy; 2020 laravelapp.dev</p>
</div>

@endsection

```

Keterangann

1. File homepage.blade.php berfungsi sebagai *child-page* yang mengextends file template.blade.php. Jadi nanti file ini memanfaatkan template.blade.php sebagai *parent page*. Ini seperti OOP dimana suatu class dapat mengextends class lainnya.
2. Kalian ingat pada file template.blade.php terdapat kode `@yield('main')`. Bagian itu akan diisi section main pembuatan section bernama main yang akan mengisi bagian tersebut. Begitu pula dengan section footer pada master template.

Pada file laravelapp/pages/about.blade.php ubah isinya menjadi :

```
@extends('template')
```

```
@section('main')
```

```
<div id="about">
```

```
<h2>About</h2>
```

```
<p>Aplikasi <strong>laravelapp</strong> dibuat sebagai latihan untuk mempelajari Laravel</p>
```

```
</div>
```

```
@endsection
```

Penjelasannya sama dengan sebelumnya. Hanya perlu diperhatikan pada about.blade.php kita tidak mendefinisikan section footer

Pada file laravelapp/siswa/index.blade.php ubah isinya menjadi :

```
@extends('template')
```

```
@section('main')
```

```
<div id="siswa">
```

```
<h2>Siswa</h2>
```

```
<?php if(!empty($siswa)): ?>
```

```
<ul>
```

```
<?php foreach($siswa as $anak): ?>
```

```
<li><?= $anak ?></li>
```

```
<?php endforeach ?>
```

```
</ul>
```

```
<?php else: ?>
```

```
<p>Tidak ada data siswa</p>
```

```
<?php endif ?>
```

```
</div>
```

```
@endsection
```

```
@section('footer')
```

```
<div id="footer">
```

```
<p>&copy; 2020 laravelapp.dev</p>
```

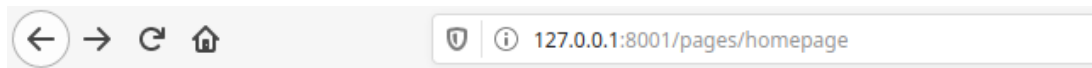
```
</div>
```

```
@endsection
```

Keterangan

1. File `homepage.blade.php` berfungsi sebagai *child page* yang mengextends file `template.blade.php`. Jadi nanti kita akan memanfaatkan `template.blade.php` sebagai *parent page*. Ini seperti OOP dimana suatu class dapat mengextend class lainnya.
2. Kalian ingat pada `template.blade.php` terdapat kode `@yield('main')`. Bagian itu akan diisi section bernama `main`

Pada browser kalian bisa akses halaman `pages/homepage` akan menampilkan



Homepage

Selamat belajar laravel!

© 2020 laravelapp.dev

Halaman Homepage

Jika kalian mengakses halaman siswa akan menampilkan



Siswa

- Arba Nugraha
- Riki Amudra
- Mahsa Vania Salsabila

© 2020 laravelapp.dev

Halaman Siswa

Dan untuk halaman about hampir sama dengan halaman homepage dan siswa untuk penjelasannya

Penutup

Mungkin itu sedikit pendahuluan menggunakan blade engine untuk selanjutnya kita akan bahas mempercantik halaman blade dengan bootstrap dll. See you ditutorial selanjutnya.

Related Posts

- [Send Notifikasi Slack With Webhook](#)
- [\[laravel\] Blade Dengan Bootstrap](#)
- [\[laravel\] Memahami View](#)
- [\[laravel\] Memahami View Data](#)
- [\[laravel\] Membuat Route Untuk Halaman](#)



Muhammad Bella Buay Nunyai

A man who tries to be human

[Tweet](#) [Share](#)

<https://bellabeen.github.io/laravel-blade/>

10.

Tabel Dinamis yang Dapat Digunakan Kembali dengan Laravel Livewire

Nama Apród Illés



Tabel selalu memainkan peran penting dalam sejarah perkembangan web. Beberapa dekade yang lalu, meja mengalami masa keemasannya; mereka digunakan tidak hanya untuk menyusun data tetapi juga untuk tata letak halaman web. Sangat mudah untuk mengatur segala sesuatunya dengan meja, di <tr>sini, di <td>sana, dan semuanya berada pada posisi yang tepat.

Dengan keinginan untuk mengembangkan situs web yang responsif dan mudah diakses serta karena rangkaian fitur CSS yang semakin berkembang, tabel akhirnya digunakan sesuai tujuan pembuatannya: menampilkan data terstruktur.

Tabel bisa jadi jauh lebih rumit dari yang kita kira. Untuk memecah kerumitannya, kita dapat membuat komponen yang dapat digunakan kembali di seluruh proyek dan juga membantu menciptakan tampilan dan pengalaman pengguna yang terpadu.

Ini adalah tujuan dari komponen tabel kami:

- kolom dapat disesuaikan untuk berbagai jenis data
- kolom dapat ditata secara khusus
- dapat diberi nomor halaman
- dapat diurutkan berdasarkan kolom

Berikut sekilas apa yang ingin kami bangun pada akhir ini:

NAME	EMAIL	STATUS	CREATED AT
Eva Jacobs	mueller.al@example.com	ACTIVE	14 years ago
Margarete Dietrich	dstroman@example.net	ACTIVE	35 years ago
Margie Nikolaus DDS	valerie.blick@example.org	ACTIVE	21 years ago
Leslie Leuschke Jr.	wbraun@example.com	ACTIVE	29 years ago
Mr. Jerald Schimmel I	clotilde25@example.org	ACTIVE	20 years ago
Alva Auer	eriberto.dicki@example.net	DELETED	42 years ago
Miss Jennyfer Graham PhD	pjast@example.org	DELETED	42 years ago
Dane Rau	pink.kulas@example.com	INACTIVE	18 years ago
Miss Vivianne Kunze	fblick@example.net	DELETED	38 years ago
Oran Bernhard	alexandrea.crist@example.net	DELETED	28 years ago

Showing 1 to 10 of 55 results

< 1 2 3 4 5 6 >

Komponen ini dimaksudkan untuk dapat digunakan kembali. Untuk mencapai hal ini, kita akan membuat komponen Tabel abstrak, dan kelas Kolom pembantu. Jika digabungkan, ini akan membantu kita dengan cepat menyusun tabel yang berguna di aplikasi kita.

Siapkan beberapa contoh data

Pertama, buat [proyek Laravel baru](#), [instal Livewire menggunakan composer](#), dan [instal Tailwind menggunakan npm](#).

Untuk contoh ini, kita akan membuat tabel pengguna. Untuk menyimpan data yang kita inginkan, mari edit `create_users_table` migrasi dan tambahkan baris berikut:

```
Schema::create('users', function (Blueprint $table) {
    ...
    $table->string('avatar');
    $table->string('status');
    $table->string('country');
    ...
});
```

Mari kita edit juga kelasnya `UserFactory` untuk menghasilkan beberapa data palsu yang bagus untuk kita.

```
public function definition()
{
    return [
        ...
        'avatar' => $this->faker->imageUrl(40,40),
        'status' => $this->faker->
            randomElement(['active', 'inactive', 'deleted']),
        'country' => $this->faker->country,
    ];
}
```

Selanjutnya, mari kita edit `DatabaseSeeder` kelas untuk membuat 55 pengguna.

```
public function run()
{
    \App\Models\User::factory(55)->create();
}
```

Saat kita menjalankan perintah `php artisan migrate:refresh --seed`, itu akan menyegarkan dan menyemai database kita.

Besar! Kami telah menyiapkan struktur basis data dan basis data pengembang kami diisi dengan data pengguna palsu yang tampak realistis. Saatnya memulai komponen!

Membuat komponen tabel

Pertama, mari kita buat sebuah `abstract` komponen, yang disebut `Tabel`. Itu akan menangani fungsionalitas dasar tabel. Kita dapat memperluas berbagai jenis tabel dari itu.

Mari kita mulai dengan menjalankan: `php artisan make:livewire table`

Kami mengubah `Table` kelas menjadi `abstract` dan menambahkan dua fungsi abstrak yang disebut `query()` dan `columns()`. Selain itu, kami akan menambahkan fungsi non-abstrak `data()` yang mengembalikan hasil kueri.

```
abstract class Table extends Component
{
    ...

    public abstract function query() : \Illuminate\Database\Eloquent\Builder;

    public abstract function columns() : array;

    public function data()
    {
        return $this
            ->query()
            ->get();
    }
}
```

Kelas kolom

Kelas kolom adalah inti dari komponen tabel kita. Kita dapat membuat `Table` folder di `app` direktori kita, dan membuat `Column` kelas di sini. (Ini bisa ditempatkan di mana saja dalam proyek Anda, itu tergantung pada bagaimana Anda menyusun kode Anda.)

Mari tambahkan `$component` properti ke kelas. Ini menjelaskan komponen yang harus dirender untuk menampilkan data kolom. Kita dapat menggantinya untuk menyesuaikannya.

Kami juga akan menambahkan `__construct($key, $label)` fungsi yang menetapkan dua properti. Dengan membuat `make()` fungsi statis, kita dapat menggunakan kelas kita sebagai [API lancar](#) yang memungkinkan rangkaian metode.

Setelah perubahan tersebut, inilah tampilan kelas kita:


```

class Column
{
    public string $component = 'columns.column';

    public string $key;

    public string $label;

    public function __construct($key, $label)
    {
        $this->key = $key;
        $this->label = $label;
    }

    public static function make($key, $label)
    {
        return new static($key, $label);
    }
}

```

Seperti yang Anda lihat, kami menyetel \$component ke 'columns.column'. Agar ini berfungsi, kita perlu membuat komponen ini sebagai resources/views/components/columns/column.blade.php

Komponen ini bertanggung jawab untuk menampilkan data yang dikirimkan sebagai prop bernama value. Kita dapat mewujudkannya di column.blade.php komponen kita seperti yang ditunjukkan di bawah ini.

```

@props([
    'value',
])

<div>
    {{ $value }}
</div>

```

Membuat tata letak meja

Mari fokus pada table.blade.php file sekarang.

Pertama, kita mulai dengan tata letak tabel sederhana; lalu, kita merender judulnya dengan mengulangi \$this->column() array.

Setelah itu, kami mengulangi \$this->data() array, yang saat ini berisi pengguna kami. Menggunakan <x-dynamic-component> helper secara dinamis merender kumpulan komponen di \$column->component properti.

```

<div>
  <div class="relative overflow-x-auto shadow-md rounded-lg">
    <table class="w-full text-sm text-left text-gray-500">
      <thead class="text-xs text-gray-700 uppercase bg-gray-50">
        <tr>
          @foreach($this->columns() as $column)
            <th>
              <div class="py-3 px-6 flex items-center">
                {{ $column->label }}
              </div>
            </th>
          @endforeach
        </tr>
      </thead>
      <tbody>
        @foreach($this->data() as $row)
          <tr class="bg-white border-b hover:bg-gray-50">
            @foreach($this->columns() as $column)
              <td>
                <div class="py-3 px-6 flex items-center cursor-pointer">
                  <x-dynamic-component
                    :component="$column->component"
                    :value="$row[$column->key]"
                  >
                </x-dynamic-component>
              </div>
            </td>
          @endforeach
        </tr>
      @endforeach
    </tbody>
  </table>
</div>
</div>

```

Menginisialisasi kamiUsersTable

Kami membuat semua persiapan yang diperlukan untuk merender tabel pertama kami. Mari kita wujudkan.

Mari buat UsersTable komponen Livewire menggunakan perintah ini php artisan make:livewire UsersTable.

Kemudian kita akan mengedit UsersTable kelas untuk diperluas dari Table kelas abstrak kita. Selain itu, mari kita hapus fungsinya render() karena kita tidak perlu menyimpannya.

```

class UsersTable extends Table
{
}

```

Kita membuat dua fungsi abstrak di Table kelas dasar kita, mari kita implementasinya di sini.

Dalam query()fungsi tersebut kita perlu mengembalikan sebuah Eloquent\Builderyang menggambarkan model data kita.

```
public function query() : Builder
{
    return User::query();
}
```

Dalam columns()fungsinya, kita perlu mengembalikan array Columninstance. Kami akan mengatur kunci untuk mencocokkan database kami dan label sesuai keinginan kami.

```
public function columns() : array
{
    return [
        Column::make('name', 'Name'),
        Column::make('email', 'Email'),
        Column::make('status', 'Status'),
        Column::make('created_at', 'Created At'),
    ];
}
```

Langkah terakhir adalah merender tabel kita. Untuk mempermudah, kami hanya akan memasukkannya ke dalam welcome.blade.php.

```
<html>
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1">

    <title>Livewire Table</title>

    <link href="{{ mix('/css/app.css') }}" rel="stylesheet">
    @livewireStyles
  </head>
  <body class="flex items-center justify-center min-h-screen">

    <div class="w-full max-w-6xl">
      <livewire:users-table></livewire:users-table>
    </div>

    @livewireScripts
  </body>
</html>
```

Ketika kita menjalankannya, kita akan melihat sesuatu seperti ini. Kami memiliki komponen tabel yang berfungsi dan dapat digunakan kembali!

NAME	EMAIL	STATUS	CREATED AT
Eva Jacobs	mueller.al@example.com	active	2008-02-05 13:05:28
Margarete Dietrich	dstroman@example.net	active	1987-02-15 12:29:02
Margie Nikolaus DDS	valerie.blick@example.org	active	2001-01-03 02:41:47
Leslie Leuschke Jr.	wbraun@example.com	active	1992-08-11 06:14:49
Mr. Jerald Schimmel I	clotilde25@example.org	active	2002-03-28 03:36:05
Alva Auer	eriberto.dicki@example.net	deleted	1980-02-12 05:26:32
Miss Jennyfer Graham PhD	pjast@example.org	deleted	1979-12-28 16:34:36
Dane Rau	pink.kulas@example.com	inactive	2003-08-10 16:33:19
Miss Vivianne Kunze	fblick@example.net	deleted	1984-03-20 08:45:39
Oran Bernhard	alexandrea.crist@example.net	deleted	1994-04-15 03:18:01

Menyesuaikan gaya kolom dan pemformatan data

Menerapkan gaya khusus ke kolom kita merupakan persyaratan umum. Bagaimana kita melakukan itu?

Kita bisa membuat `component($component)` fungsi di `Column` kelas kita, yang mengatur `$component` properti kita.

```
public function component($component)
{
    $this->component = $component;

    return $this;
}
```

Kembali dengan `$this` adalah pola umum ketika Anda ingin membuat API yang lancar. Dengan menggunakan pola ini, kita dapat memanggil beberapa fungsi secara berurutan pada instance yang sama.

Kami akan memasukkan komponen-komponen yang umum digunakan ke dalam folder `common` bernama `resources/views/components/columns`. Buat file di sana bernama `human-diff.blade.php` dengan konten berikut:

```
@props([
    'value'
])

<div>
    {{ \Carbon\Carbon::make($value)->diffForHumans() }}
</div>
```

Kolom yang sangat spesifik untuk tabel kita bisa masuk ke dalam `resources/views/components/columns/users` folder. Mari kita buat file kolom `status` `status.blade.php` yang memformat data status kita dengan lebih elegan.

```
@props([
  'value'
])

<div class="flex">
  <div @class([
    'text-white rounded-xl px-2 uppercase font-bold text-xs',
    'bg-red-500' => $value === 'deleted',
    'bg-green-500' => $value === 'active',
    'bg-gray-500' => $value === 'inactive',
  ])>
    {{ $value }}
  </div>
</div>
```

Sekarang, kita akan mengedit `columns()` fungsi di kelas kita `UsersTable` untuk memanggil `component($component)` fungsi yang dibuat sebelumnya.

```
public function columns() : array
{
  return [
    Column::make('name', 'Name'),
    Column::make('email', 'Email'),
    Column::make('status', 'Status')->component('columns.users.status'),
    Column::make('created_at', 'Created At')-
    >component('columns.common.human-diff'),
  ];
}
```

Kolom kita status sekarang memiliki gaya khusus, dan `created_at` kolom tersebut menampilkan data yang diformat.

NAME	EMAIL	STATUS	CREATED AT
Eva Jacobs	mueller.al@example.com	ACTIVE	14 years ago
Margarete Dietrich	dstroman@example.net	ACTIVE	35 years ago
Margie Nikolaus DDS	valerie.blick@example.org	ACTIVE	21 years ago
Leslie Leuschke Jr.	wbraun@example.com	ACTIVE	29 years ago
Mr. Jerald Schimmel I	clotilde25@example.org	ACTIVE	20 years ago
Alva Auer	eriberto.dicki@example.net	DELETED	42 years ago
Miss Jennyfer Graham PhD	pjast@example.org	DELETED	42 years ago
Dane Rau	pink.kulas@example.com	INACTIVE	18 years ago
Miss Vivianne Kunze	fblick@example.net	DELETED	38 years ago
Oran Bernhard	alexandrea.crist@example.net	DELETED	28 years ago

Livewire lebih baik di Fly

Aplikasi Livewire berjalan lebih lancar saat Anda memindahkannya lebih dekat ke pengguna Anda. Terapkan aplikasi Anda di Fly.io dalam hitungan menit!

[Terapkan aplikasi Laravel Anda!→](#)

Menambahkan paginasi

Livewire memiliki komponen penomoran halaman bawaan yang menggunakan Tailwind CSS secara default; ayo gunakan sekarang.

Kita perlu menggunakan sifat Livewire `WithPagination` dan menambahkan dua properti ke `Table` kelas.

```
abstract class Table extends Component
{
    use WithPagination;

    public $perPage = 10;

    public $page = 1;

    ...
}
```

Mari kita edit fungsinya `data()` untuk membuat halaman hasil.

```
public function data()
{
    return $this
        ->query()
        ->paginate($this->perPage);
}
```

Kita dapat memasukkan komponen pagination dalam `table.blade.php` file kita menggunakan `$this->data()->links()`. Bagi mereka yang penasaran tentang cara kerjanya, Anda dapat mempelajari lebih lanjut di [dokumen Livewire](#).

```
<div class="flex flex-col gap-5">
```

```
<div class="flex flex-col gap-5">
    <div class="relative overflow-x-auto shadow-md rounded-lg">
        <table class="w-full text-sm text-left text-gray-500">
            ...
        </table>
    </div>
    {{ $this->data()->links() }}
</div>
```

Kami memiliki pagination yang berfungsi!

NAME	EMAIL	STATUS	CREATED AT
Coy Christiansen	metz.consuelo@example.org	INACTIVE	30 years ago
Miss Cassidy Williamson	madonna.macejkovic@example.org	ACTIVE	20 years ago
Remington Collier PhD	gene35@example.net	DELETED	51 years ago
Sterling Leffler	gianni67@example.com	DELETED	36 years ago
Jessyca Hagenes I	xdooley@example.com	ACTIVE	7 years ago
Jaqueline Metz	wbatz@example.net	DELETED	42 years ago
Dejon DuBuque	prohaska.gwendolyn@example.com	INACTIVE	8 years ago
Dayana Nolan	jeffrey.sauer@example.com	ACTIVE	1 year ago
Sandra Conn Sr.	alden35@example.net	ACTIVE	29 years ago
Newton Murazik MD	carter.clara@example.net	INACTIVE	47 years ago

Showing 11 to 20 of 55 results

< 1 2 3 4 5 6 >

Urutkan kolomnya

Untuk mengurutkan kolom, kita perlu melakukan beberapa modifikasi.

Pertama, kita akan menambahkan properti a \$sortBy dan a \$sortDirection ke kelas kita Table.

```
public $sortBy = '';
```

```
public $sortDirection = 'asc';
```

Kami akan mengedit data() fungsi untuk menambahkan fungsionalitas penyortiran dengan \$query->orderBy().

```
public function data()
{
    return $this
        ->query()
        ->when($this->sortBy !== '', function ($query) {
            $query->orderBy($this->sortBy, $this->sortDirection);
        })
        ->paginate($this->perPage);
}
```

Kita memerlukan tindakan di Table kelas kita yang dipanggil ketika judul kolom diklik. Jika kunci yang dipilih cocok dengan kunci label yang diklik, maka arahnya akan terbalik. Jika tidak, ini akan menyetel \$sortBy ke tombol yang diklik, dan \$sortDirection ke 'asc'. Ini juga menangani pengaturan ulang halaman agar berfungsi dengan baik dengan penomoran halaman.

```
public function sort($key) {
    $this->resetPage();

    if ($this->sortBy === $key) {
        $direction = $this->sortDirection === 'asc' ? 'desc' : 'asc';
        $this->sortDirection = $direction;
    }

    return;
}

$this->sortBy = $key;
$this->sortDirection = 'asc';
}
```

Mari kita modifikasi `table.blade.php` untuk menambahkan pemroses klik ke label. Kami juga akan menambahkan ikon untuk mewakili status penyortiran saat ini. Kami akan menggunakan ikon dari [heroicons](#) untuk itu.

```
<th wire:click="sort('{{ $column->key }}')">
  <div class="py-3 px-6 flex items-center cursor-pointer">
    {{ $column->label }}
    @if($sortBy === $column->key)
      @if ($sortDirection === 'asc')
        <svg xmlns="http://www.w3.org/2000/svg" class="h-5 w-5" viewBox="0
0 20 20"
          fill="currentColor">
          <path fill-rule="evenodd"
            d="M14.707 10.293a1 1 0 010 1.414l-4 4a1 1 0 01-1.414 0l-
4-4a1 1 0 111.414-1.414L9 12.586V5a1 1 0 012 0v7.586l2.293-2.293a1 1 0
011.414 0z"
            clip-rule="evenodd"/>
          </svg>
        @else
          <svg xmlns="http://www.w3.org/2000/svg" class="h-5 w-5" viewBox="0
0 20 20"
            fill="currentColor">
            <path fill-rule="evenodd"
              d="M5.293 9.707a1 1 0 010 1.414l4 4a1 1 0 011.414 0l4
1 0 01-1.414 1.414L11 7.414V5a1 1 0 012 0v7.414l6.707 6.707a1 1 0 01-
1.414 0z"
              clip-rule="evenodd"/>
            </svg>
          @endif
        @endif
      </div>
    </th>
```

Mari kita sortir!

NAME	EMAIL	STATUS	CREATED AT ↑
Abraham Borer	rohan.vada@example.org	INACTIVE	1 year ago
Dayana Nolan	jeffrey.sauer@example.com	ACTIVE	1 year ago
Jordyn Wilkinson	shields.jennings@example.org	DELETED	2 years ago
Kathryn Okuneva DDS	von.beau@example.org	DELETED	5 years ago
Jessyca Hagenes I	xdooley@example.com	ACTIVE	7 years ago
Dejon DuBuque	prohaska.gwendolyn@example.com	INACTIVE	8 years ago
Elna Okuneva	solon.bosco@example.org	DELETED	9 years ago
Dr. Jonatan Jones I	pdavis@example.com	ACTIVE	12 years ago
Arthur Cremin	iconroy@example.net	DELETED	12 years ago
Ena Kreiger	dandre67@example.org	INACTIVE	12 years ago

Showing 1 to 10 of 55 results

< 1 2 3 4 5 6 >

Kesimpulan

Membuat tabel bisa menjadi tugas yang monoton; menggunakan perpustakaan seperti Livewire dan Tailwind dapat mempercepat proses dan membuat pengembangan menyenangkan. Membuat komponen yang dapat digunakan kembali membuat penambahan tabel tambahan menjadi sangat mudah.

Menambahkan perilaku dinamis tanpa menulis Javascript adalah keunggulan Livewire. Dengan bantuan bawaan seperti penomoran halaman, kita dapat mengimplementasikan solusi untuk masalah kompleks hanya dengan beberapa baris kode. Dengan semakin populernya Livewire, semakin banyak paket sumber terbuka yang tersedia setiap hari.

Dengan cara Livewire mengaktifkan perilaku dinamis tanpa memuat halaman penuh atau bahkan harus menulis API JSON, Anda dapat membuat aplikasi yang cepat dan responsif untuk pengguna Anda lebih cepat dari sebelumnya.

TERAKHIR DIPERBARUI

-

21 SEPTEMBER 2022

<https://fly.io/laravel-bytes/reusable-dynamic-tables-with-laravel-livewire/>

DAFTAR ISI

Kata pengantar.....	i
Gambar daftar isi.....	ii
Latar Belakang Masalah.....	iii
1. Apa itu Laravel? Pengertian, Fitur dan Kelebihannya.....	1
2. 5 Kelebihan Framework Laravel untuk Pengembangan Web.....	5
3. Tutorial PHP Framework Laravel Part 1.....	8
4. PHP Laravel Framework Fundamentals.....	11
5. Sejarah Perkembangan LARAVEL.....	14
6. Framework Laravel, sejarah dan cara kerjanya.....	16
7. 8 Alasan Mengapa Harus Memilih Laravel Menjadi Framework Untuk Website Kamu!.....	17
8. Instalasi Laravel.....	20
9. [laravel] Blade (Template Engine).....	29
10. Tabel Dinamis yang Dapat Digunakan Kembali dengan Laravel Livewire.....	34
Daftar ISI.....	46
Penutup.....	47
Daftar Fustaka.....	48

Penutup

Kesimpulan

framework laravel metode iterative yaitu analisis kebutuhan perangkat lunak, desain, pengkodean, pengujian. Sistem informasi berbasis web dengan framework laravel dikembangkan dengan menggunakan framework laravel dapat mengatasi masalah pengelolaan seperti perpustakaan yang masih belum terkomputerisasi. Sistem informasi tersebut memiliki fitur-fitur meliputi: 1) Kelola anggota; 2) Kelola buku; 3) Kelola Peminjaman; 4) Kelola pengembalian; 5) sehingga sistem ini memberikan keuntungan sebagai berikut : 1. Dengan terbentuknya program aplikasi ini memberikan kemudahan kepada Pustakawan sebagai admin perpustakaan pengguna sistem informasi dalam mengelola data anggota dan buku, serta memudahkan dalam melakukan transaksi peminjaman dan pengembalian buku karena sudah terkomputerisasi. 2. Sistem informasi perpustakaan dapat memudahkan anggota perpustakaan untuk mencari buku peminjaman tanpa harus mencari satu persatu ke dalam rak buku, karena sistem perpustakaan sudah terkomputerisasi. 3. Sistem informasi perpustakaan memudahkan untuk melihat berapa jumlah anggota, buku, peminjaman dan pengembalian, serta bisa Dan framework Laravel bisa digunakan untuk sistem berbasis web lainnya.

DAFTAR FUSTAKA

1. <https://www.dewaweb.com/blog/apa-itu-laravel/>
2. <https://eduparx.id/blog/insight/development/5-kelebihan-framework-laravel/>
3. <https://socs.binus.ac.id/2017/09/15/laravel51/>
4. <https://brainmatics.com/php-laravel-framework-fundamentals/>
5. <https://sis.binus.ac.id/2020/03/23/sejarah-perkembangan-laravel/>
6. <https://www.lembarilmu.com/post/detail/framework-laravel-sejarah-dan-cara-kerjanya>
7. <https://argiaacademy.com/8-alasan-mengapa-harus-memilih-laravel-menjadi-framework-untuk-website-kamu/>
8. <https://laravel.com/docs/10.x/installation>
9. <https://bellabeen.github.io/laravel-blade/>
10. <https://fly.io/laravel-bytes/reusable-dynamic-tables-with-laravel-livewire/>