

# *Laravel Dasar*

Nur Muhammad Abdul Falaq

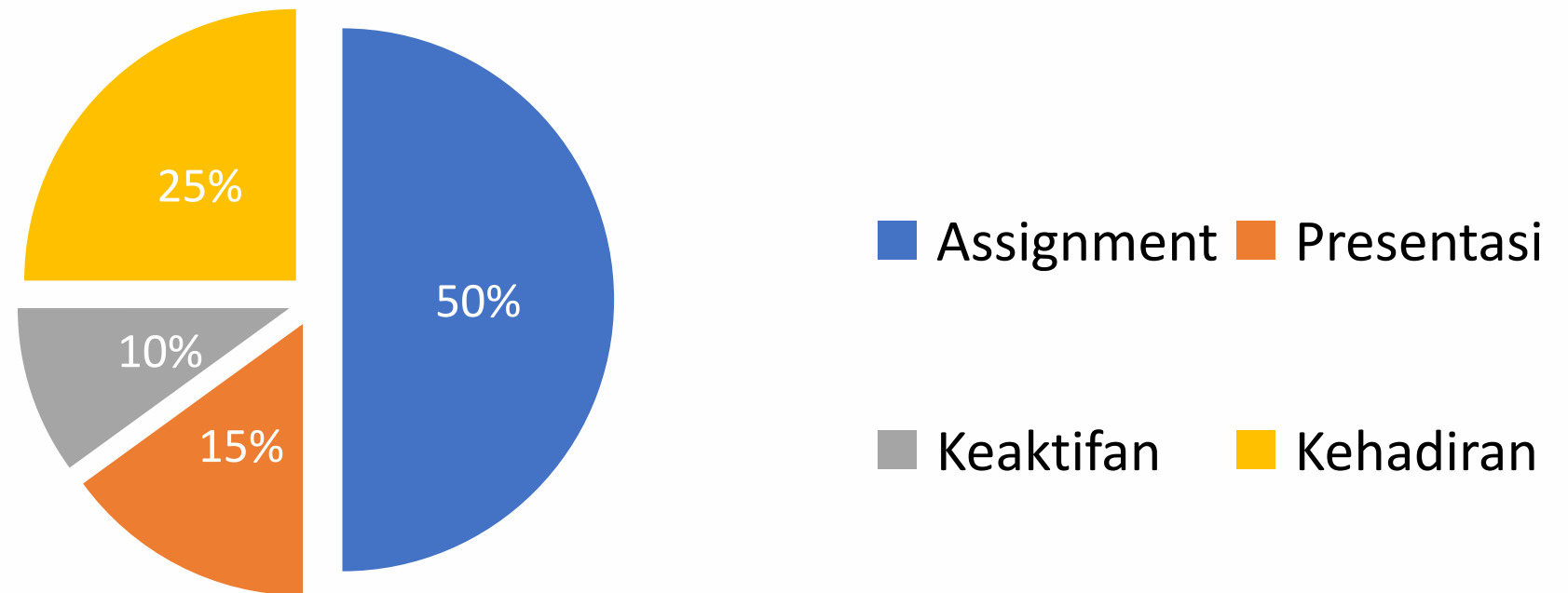


**PINTAR**



# Penilaian

**Komponen Penilaian**



# Scope

- Farmework
- Tentang Laravel
- Struktur Laravel
- Konfigurasi Dasar Laravel
- Composer
- Cara Install Composer
- Cara Install Laravel
- <https://laravel.com/docs/10.x>
- [https://www.youtube.com/watch?v=ClMX6TXvh\\_o&t=17s&pp=ygUNbGFyYXZlbCBkYXNhcg%3D%3D](https://www.youtube.com/watch?v=ClMX6TXvh_o&t=17s&pp=ygUNbGFyYXZlbCBkYXNhcg%3D%3D)
- [https://www.youtube.com/watch?v=HqAMb6kqILs&list=PLFIM0718LjIWiihbBIq-SWPU6b6x21Q\\_2](https://www.youtube.com/watch?v=HqAMb6kqILs&list=PLFIM0718LjIWiihbBIq-SWPU6b6x21Q_2)

# Framework

- framework adalah kerangka kerja untuk mengembangkan aplikasi berbasis website maupun desktop. Kerangka kerja disini sangat membantu developer dalam menuliskan sebuah kode dengan lebih terstruktur dan tersusun rapi.
- Kerangka kerja diciptakan untuk mempermudah kinerja dari programmer. Sehingga, seorang programmer tidak perlu untuk menuliskan kode secara berulang – ulang. Karena di dalamnya sendiri anda hanya perlu menyusun komponen – komponen pemrograman saja.

# Fungsi Framework

- Kode program lebih terstruktur
- Membantu kinerja dari developer
- Meningkatkan keamanan perangkat lunak / website
- Pemeliharaan dan dokumentasi dari website lebih mudah
- Mempercepat dalam proses pembuatan website

# Kode program lebih terstruktur

- Fungsi framework yang utama adalah membuat source code menjadi lebih terstruktur. Terstruktur disini, berarti program yang dibuat akan dimasukkan ke dalam setiap komponen sesuai dengan fungsi nya masing – masing.
- Salah satu contoh dari kode program terstruktur dapat dilihat dari framework PHP, yaitu Laravel. Yang menggunakan konsep paradigma MVC (Model, View, Controller). Terdapat tiga komponen utama untuk mengembangkan website menggunakan model framework tersebut.
- Model berfungsi untuk tempat atau wadah menampung kode program berupa algoritma pemrograman dan penghubung database aplikasi. View berfungsi sebagai wadah menampung kode program untuk membuat tampilan yang nantinya ditampilkan kepada customer / client. Dan controller berfungsi untuk menghubungkan model dan view agar menjadi sebuah website secara keseluruhan.
- Dapat dikatakan juga, model disini untuk menangani tugas back end. Kemudian, view untuk menangani tugas front end. Dengan adanya konsep MVC tersebut, kode program akan tersusun rapi serta mempersingkat kerja dari developer.

# Membantu kinerja dari developer

- Fungsi yang kedua adalah membantu kinerja dari developer sendiri. Dari sini, anda pasti sudah berpikir bahwa sebenarnya dalam membuat sebuah aplikasi dapat dilakukan tanpa menggunakan bantuan framework. Tentu saja, bisa dilakukan jika proyek yang ditangani dalam lingkup kecil.
- Apabila anda membuat aplikasi atau tampilan website untuk sebuah perusahaan atau organisasi besar tentu saja hal tersebut tidak disarankan bahkan mindset tersebut harus segera diubah. Framework diciptakan untuk memudahkan kinerja dari developer dalam segi efisiensi waktu serta resource yang dibutuhkan.
- Selain itu, apabila dikerjakan oleh beberapa tim maka framework akan sangat membantu efektivitas kerja dan sinkronisasi. Setiap dapat langsung memahami dan mengerjakan tugas masing – masing tanpa perlu mengecek satu persatu komponen baik dari segi back end maupun front end.

## Meningkatkan keamanan perangkat lunak / website

- Selain membantu kinerja dari developer, dari segi keamanan website maupun software juga akan meningkat. Karena dalam framework telah diidentifikasi oleh beberapa developer ahli dari berbagai negara.
- Kemudian, sebuah kerangka kerja juga memiliki beberapa versi yang setiap saat akan selalu ada pembaruan dari segi fitur serta untuk mengurangi adanya bugs. Keamanan akan selalu diperbarui dan dimonitoring agar tidak terjadi permasalahan yang lebih serius pada website misalnya terjadinya hacking maupun peretasan data.



# Pemeliharaan dan dokumentasi dari website lebih mudah

- Apabila anda ingin menambahkan atau mengurangi beberapa fitur dari website yang anda buat, maka dengan menggunakan framework dapat memudahkan anda untuk melakukan hal tersebut. Mengapa? Kami beri contoh framework Laravel sendiri memiliki fitur untuk melakukan maintenance pada program.
- Dengan melakukan maintenance, anda dapat mengubah versi website tersebut dan menambahkan beberapa fitur dengan lebih mudah dan aman. Sehingga, ketika ada perbaikan pada website maka terdapat notifikasi atau pesan bagi pengguna bahwa website masih dalam perbaikan.
- Selanjutnya, dari segi dokumentasi juga lebih terstruktur. Anda akan sangat kerepotan apabila dalam proses dokumentasi tidak menggunakan bantuan framework. Dalam kerangka kerja, setiap dokumen aplikasi yang dibangun dapat diidentifikasi dengan mudah dan cepat.

# Mempercepat dalam proses pembuatan website

- Fungsi terakhir adalah untuk mempercepat dalam proses pembuatan website. Dalam hal ini, bukan berarti pembuatan website yang baik dikerjakan dengan cepat saja. Tetapi, proses pembuatan dapat dilakukan dengan lebih cepat dan menghasilkan produk yang berkualitas dan sesuai dengan kebutuhan customer.
- Developer dapat mengembangkan aplikasi dengan menggunakan komponen – komponen yang telah tersedia dalam framework. Sehingga, tidak perlu untuk menyusun ulang dari awal kode program.

# Tentang 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.

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.

# Kelebihan Laravel

1. Template ringan
2. Library yang lengkap
3. Menggunakan model MVC
4. Tool artisan
5. Modul bersifat individu dan independen



Laravel

# Fitur-fitur Laravel

1. 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.
2. Testing and Debugging : Dalam Laravel sudah disediakan fitur untuk testing dan debugging sebuah website. Kamu tidak perlu khawatir lagi untuk melakukan testing dan debugging.
3. 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 rute. Kamu bisa memberikan nama routing, mengelompokkan, memfilter routing yang ada sesuai kebutuhan kamu.
4. Blade Template Engine. Framework ini dilengkapi dengan fitur blade yang bisa digunakan untuk mendesain layout yang konsisten.
5. Composer : Fitur ini akan membantu dalam menyempurnakan dan meningkatkan proses pengembangan website, serta mempermudah proses update-nya.

# Struktur Laravel

Struktur Laravel merupakan bagian-bagian folder dan file mana yang perlu penekanan selama pengembangan. Saya tidak akan menjelaskan secara keseluruhan, hanya yang menurut saya penting saja untuk pembuatan fitur minimal, seperti Create-Read-Update-Delete (CRUD)

```
└─ book-collector
   ├── app
   ├── bootstrap
   ├── config
   ├── database
   ├── public
   ├── resources
   ├── routes
   ├── storage
   ├── tests
   ├── vendor
   ├── .editorconfig
   ├── .env
   ├── .env.example
   ├── .gitattributes
   ├── .gitignore
   ├── artisan
   ├── composer.json
   ├── composer.lock
   ├── package.json
   ├── phpunit.xml
   ├── readme.md
   ├── server.php
   └── webpack.mix.js
```

# Folder app

Pada folder app , terdapat dua bagian besar yang perlu diperhatikan, yaitu folder Http\Controller dan folder app sendiri. Untuk setiap controller yang akan dibutuhkan pada aplikasi akan diletakkan pada folder Http\Controller dan untuk model akan diletakkan pada folder app secara langsung. File User.php diatas adalah salah satu model yang sudah ter-generate secara otomatis ketika mulai membuat project Laravel baru.

```
.
├── Console
├── Exceptions
├── Http
│   ├── Controllers
│   └── Middleware
├── Providers
└── User.php
```

```
.
├── Controllers
│   ├── Auth
│   │   ├── ForgotPasswordController.php
│   │   ├── LoginController.php
│   │   ├── RegisterController.php
│   │   ├── ResetPasswordController.php
│   │   └── VerificationController.php
│   └── Controller.php
├── Kernel.php
└── Middleware
```

# Folder resources

Pada folder resources , hal yang perlu diperhatikan adalah folder resources\view karena kita akan menyimpan semua view kita pada folder ini. File welcome.blade.php adalah salah satu contoh view yang sudah ter-generate ketika pembuatan project Laravel baru.

```
.  
├── js  
├── lang  
├── sass  
└── views  
    └── welcome.blade.php
```



# Folder routers

Pada folder routers, hal yang perlu diperhatikan adalah file web.php. Nantinya semua daftar router yang ada pada aplikasi akan kita tempatkan pada file web.php.

```
.  
├── api.php  
├── channels.php  
├── console.php  
└── web.php
```

# Folder database

Pada folder database, hal yang perlu diperhatikan adalah folder migration. folder ini berfungsi untuk pembuatan table di database dengan memanfaatkan fitur migrasi. jadi hanya melakukan satu perintah maka semua tabel di database otomatis terbuat

```
.
├── factories
│   └── UserFactory.php
├── migrations
│   ├── 2014_10_12_000000_create_users_table.php
│   └── 2014_10_12_100000_create_password_resets_table.php
└── seeds
    └── DatabaseSeeder.php
```

# Folder public

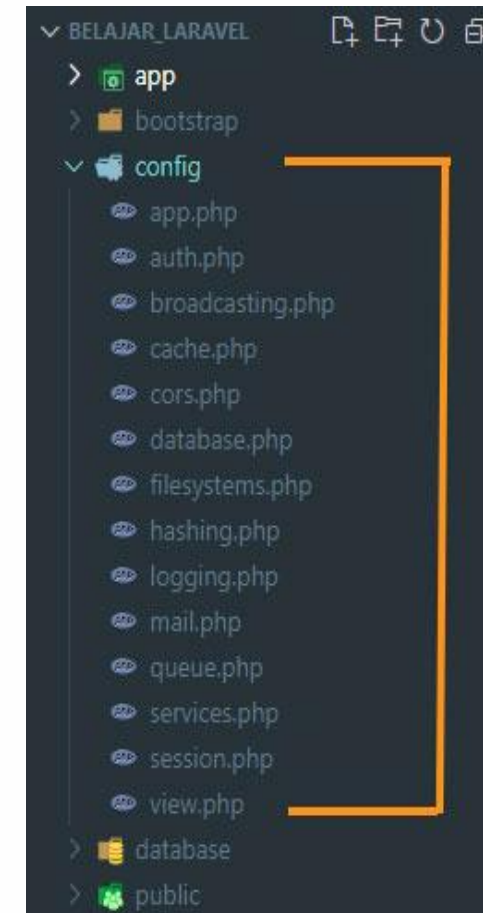
Bisa kita perhatikan file-file yang ada pada folder public terdapat file css dan js yang kita perlukan untuk styling aplikasi kita. Jika nanti kita menggunakan assets seperti css , js , dan image dari pengembang lain, maka semua assets tersebut akan ditempatkan di folder public.

```
.  
├── css  
│   └── app.css  
├── favicon.ico  
├── index.php  
├── js  
│   └── app.js  
├── robots.txt  
├── svg  
│   ├── 403.svg  
│   ├── 404.svg  
│   ├── 500.svg  
│   └── 503.svg  
└── web.config
```

# Konfigurasi Dasar Laravel

Semua konfigurasi Laravel ada dalam direktori config. Di dalam direktori tersebut, kamu bisa mengatur koneksi ke database, mail server, timezone dan masih banyak lagi.

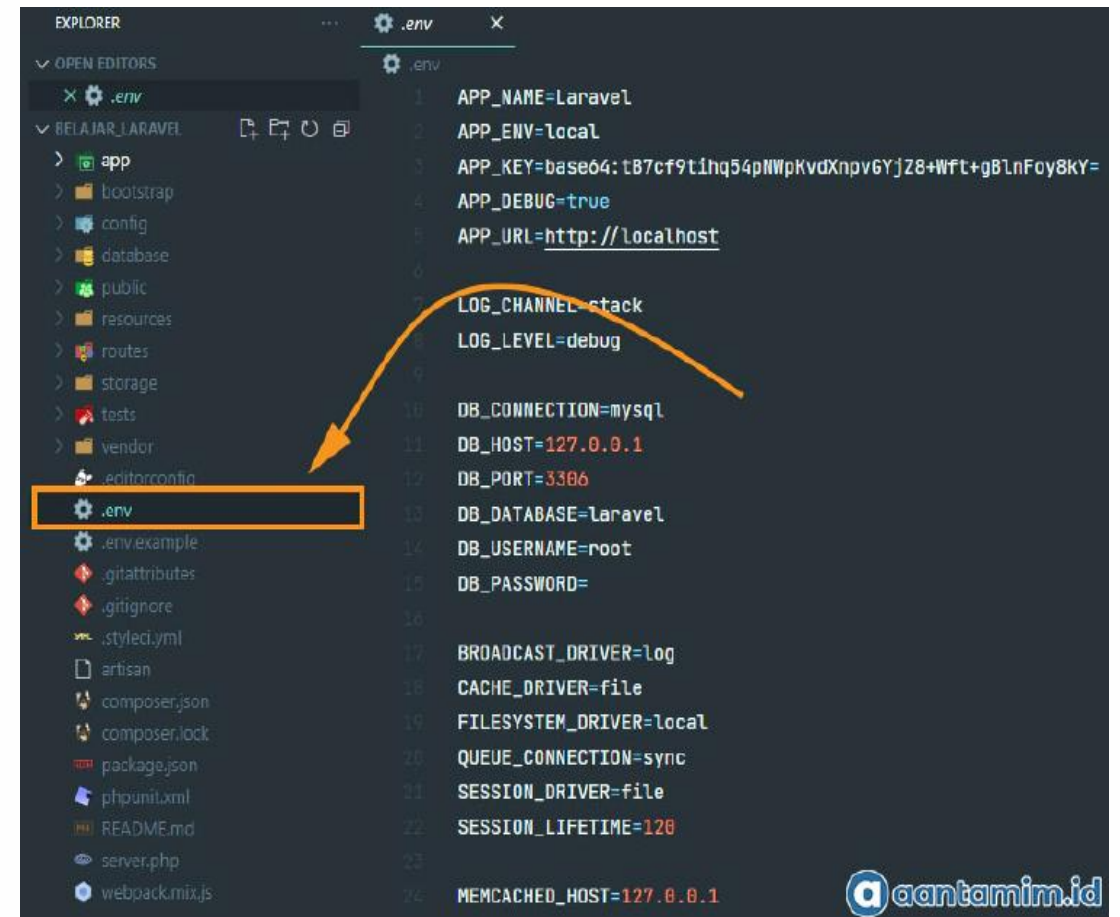
Selain itu, terdapat konfigurasi environment di Laravel. Pada dokumentasi Laravel, Laravel menggunakan library PHP DotEnv. Lalu apa bedanya dengan konfigurasi yang ada di dalam direktori config? Di dalam direktori config, konfigurasi terpisah-pisah menjadi beberapa buah file. Misalnya untuk konfigurasi database ada dalam file database.php, konfigurasi mailing ada dalam file mail.php, konfigurasi aplikasi ada dalam file app.php dan pengaturan-pengaturan lainnya.



# File .env

Adanya file .env, semua pengaturan yang ada dalam direktori config di atas dapat kita setting di satu file yang bernama .env tersebut. Jadi, kita tidak perlu berpindah-pindah file jika ingin mengubah konfigurasi aplikasi yang kita buat.

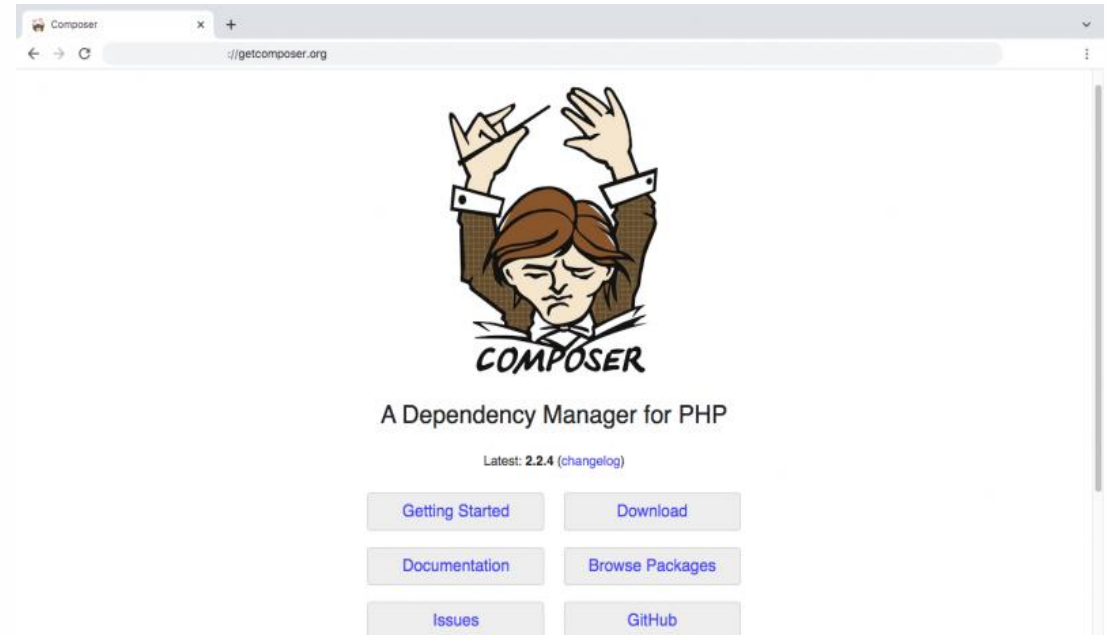
Semua konfigurasi yang ada dalam file .env lebih diutamakan daripada pengaturan yang ada dalam direktori config. Misalnya jika kita mengatur database dengan nama “penjualan” di dalam file config/database.php. Kemudian kita mengatur database dengan nama “perpustakaan” di file .env, maka database perpustakaan lah yang akan digunakan. Kecuali jika file .env kita hapus, maka pengaturan yang dijalankan adalah file-file pengaturan di dalam direktori config.



# Composer

Composer merupakan project open source yang pertama kali dikembangkan oleh Nils Adermann dan Jordi Boggiano. Sejak awal rilisnya pada 1 Maret 2012, project Composer di-host di Github dan masih terus dikembangkan hingga sekarang.

Composer adalah alat dependency manager untuk bahasa pemrograman PHP. Dengan kata lain, Composer adalah aplikasi yang diinstal ke perangkat untuk memfasilitasi developer menggunakan library open source milik orang lain ke dalam project yang sedang dibangun.






# Kenapa Developer Perlu Dependency Manager?

Programmer atau developer dianjurkan menggunakan dependency manager untuk memudahkan pengerjaan project. Selain membantu dalam penggunaan library external, berikut ini beberapa alasan mengapa developer butuh dependensi manajer:

- Memudahkan dalam mengupdate library eksternal yang digunakan
- Proses coding jadi lebih terstruktur dengan konsep MVC
- Package yang dibutuhkan akan langsung terinstal dan diupdate secara otomatis
- Terdapat autoload yang akan melakukan include terhadap semua file PHP atau class PHP yang dibutuhkan dalam projectmu
- Jika menggunakan dependency manager yang terhubung dengan packagist, kamu bisa menggunakan ribuan package yang tersedia.

# Cara Kerja Composer

1. Composer membantu menemukan paket yang tersedia pada Packagist
2. Sebelum mengunduh library-library yang diperlukan, composer akan memeriksa beberapa hal:
3. Apakah library tersebut membutuhkan library lainnya sebagai dependensi
4. Versi PHP
5. module yang diperlukan untuk library tersebut
6. Setelah proses pengecekan selesai, Composer akan mendownload library-library yang diperlukan ke dalam project pada direktori "vendor."

Name	Date modified
 vendor	23/5/2019 1:23 AM
 composer.json	23/5/2019 1:23 AM
 composer.lock	23/5/2019 1:23 AM



# Kenapa Menggunakan Composer?

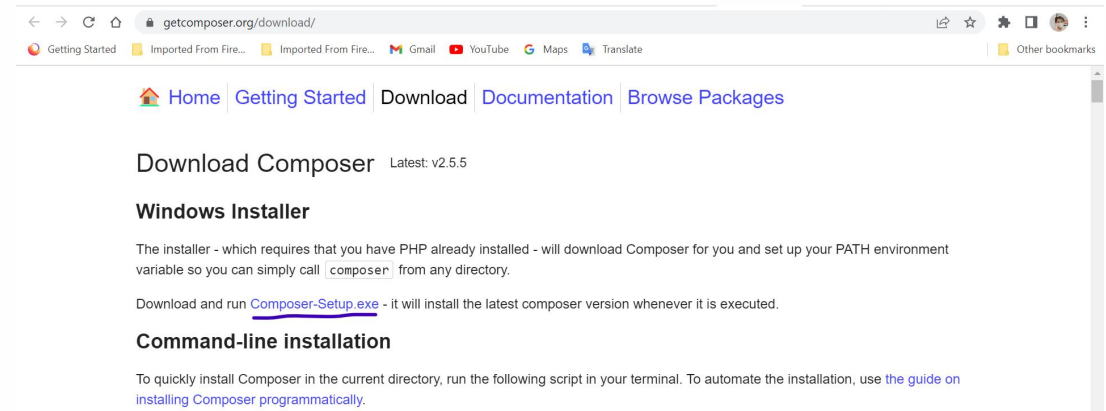
- Kehadiran Composer sebagai dependency manager membuat pengembangan project jadi lebih mudah. Mengapa?
- Fitur-fitur yang dibutuhkan untuk sebuah proyek akan memakan waktu lama bila semua dibuat sendiri dari nol. Nah solusinya, gunakan library yang sudah dibuat orang lain.
- Contohnya, kamu sedang membangun project e-commerce yang memerlukan fitur Shopping Cart, Invoice, dan Metode Pembayaran. Maka kamu tentu harus membuat fitur-fitur itu agar situs e-commerce tersebut bisa berjalan.
- Ternyata, ketiga fitur tersebut sudah pernah dibuat oleh orang lain pada situs e-commerce mereka. Tentunya akan lebih mudah bila kamu hanya tinggal mengambil script library buatan mereka dan menggunakannya di projectmu 'kan?
- Composer akan membantu mendapatkan library-library PHP milik orang lain untuk digunakan dalam project yang sedang dibuat. Asalkan dieksekusi dengan tepat, hal ini juga bisa mengurangi terjadinya error dalam proses penulisan kode.

# Cara Install Composer

1. Install PHP ke perangkatmu. Kamu bisa menggunakan XAMPP, MAMP, atau aplikasi sejenis lainnya.
2. Selanjutnya, download file Composer. <https://getcomposer.org/download/>
3. Setelah selesai mengunduh, buka dan jalankan file Composer dengan mengklik Run pada installation wizard. Kemudian lanjutkan proses instalasi dengan klik Next.
4. Selanjutnya, tentukan lokasi file instalasi. Akan muncul jendela yang meminta kamu mencari baris perintah PHP. Pastikan pilih lokasi instalasi Composer di C:/xampp/php/php.exe. Setelah itu, klik Next jika sudah benar.
5. Kemudian akan muncul permintaan Proxy Settings. Jika kamu ingin menggunakan proxy, beri centang dan masukkan URL proxy kamu. Lewati bagian ini dan klik Next jika kamu tidak menggunakan proxy. Jika sudah, klik Install.
6. Akan ada tampilan pemberitahuan perubahan Windows environment. Hal ini dimaksudkan agar Composer bisa dijalankan di command prompt.
7. Instalasi selesai! Kini Composer berhasil terinstall di perangkat Windows kamu. Klik Finish untuk menutup jendela instalasi.

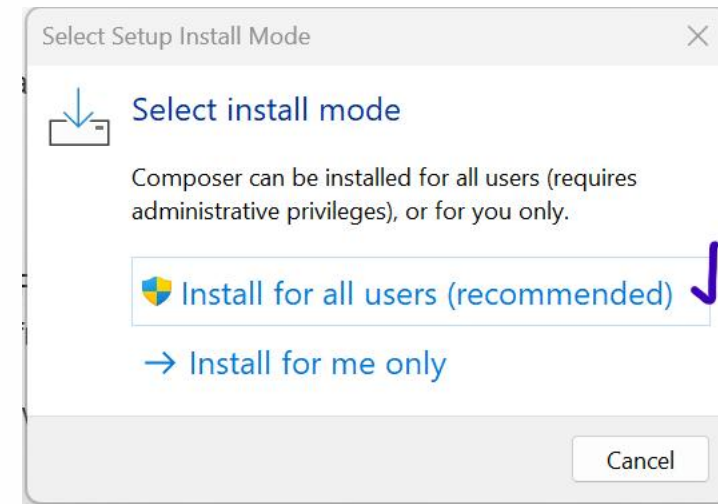
# Download Composer

1. download file Composer.  
<https://getcomposer.org/download/>
2. silahkan klik tulisan composer-setup.exe



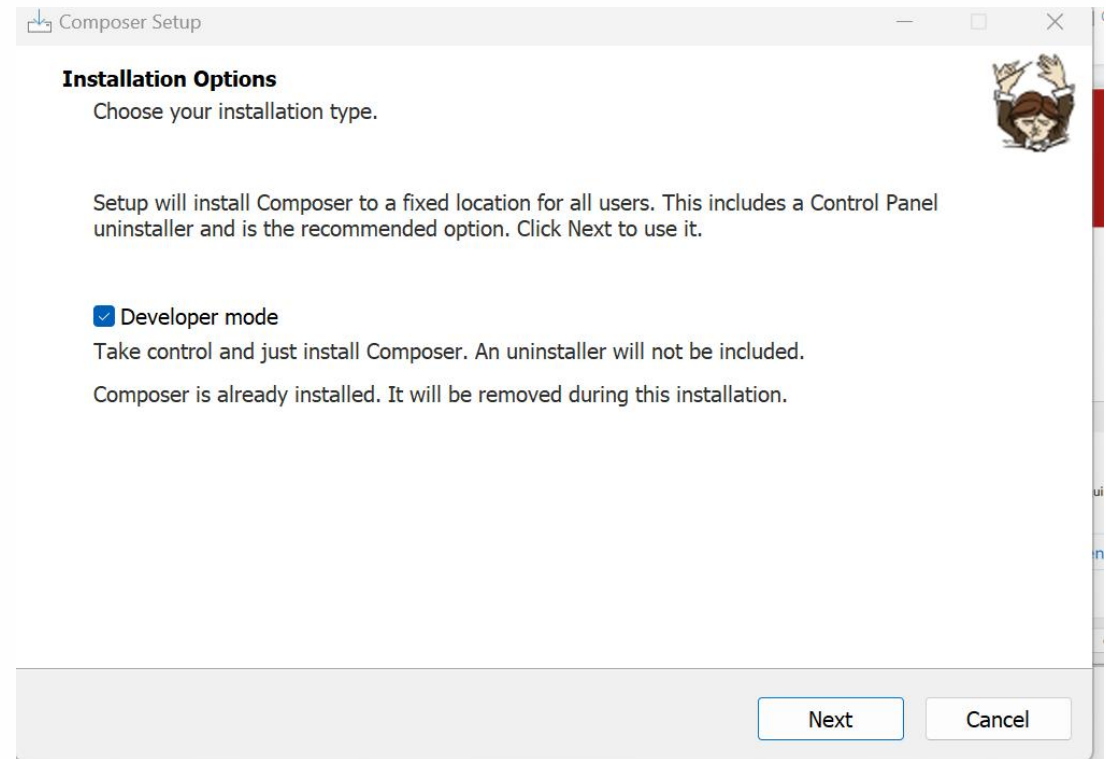
# Install Composer (1)

Jika muncul tampilan seperti ini  
ketika membuka file composer,  
maka pilih install for all users



# Install Composer (2)

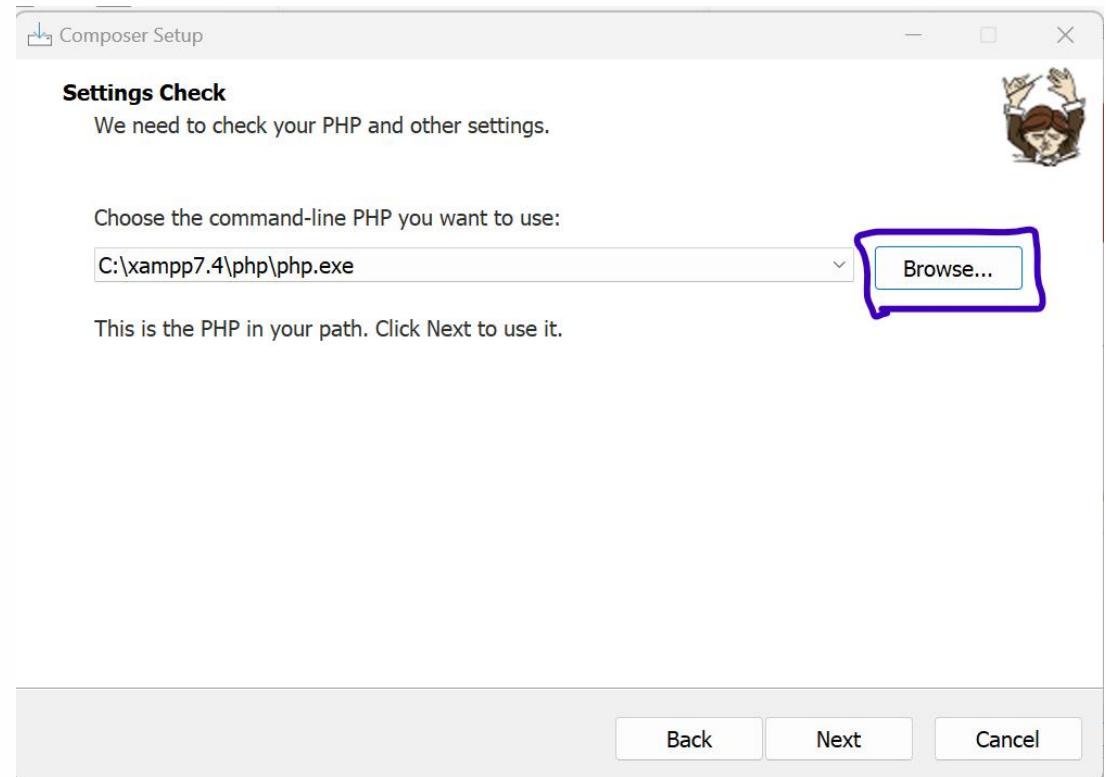
Jika muncul Tampilan seperti ini  
silahkan ceklis Developer mode  
dan klik tombol next



# Install Composer (3)

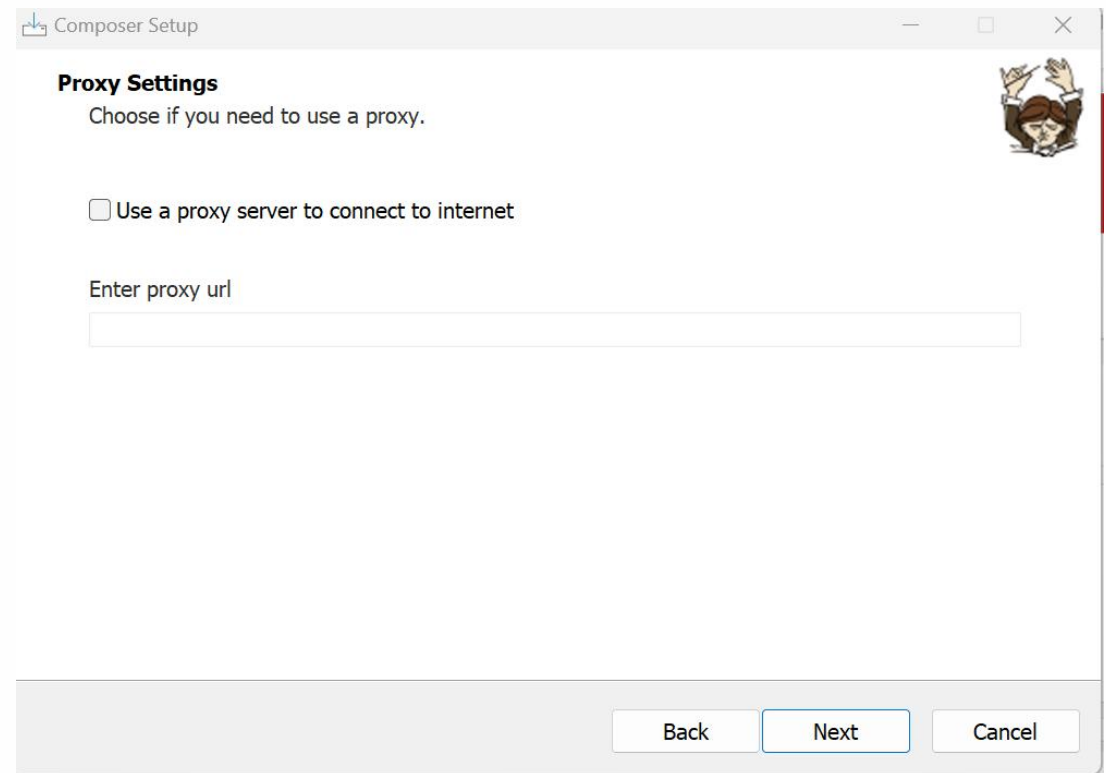
Jika Muncul Tampilan seperti ini ada dua opsi:

1. Jika hanya terdapat satu aplikasi xampp di komputer/laptop maka langsung saja klik tombol Next
2. jika terdapat lebih dari satu aplikasi xampp maka silahkan pilih xampp mana yang mau diambil file php.exe



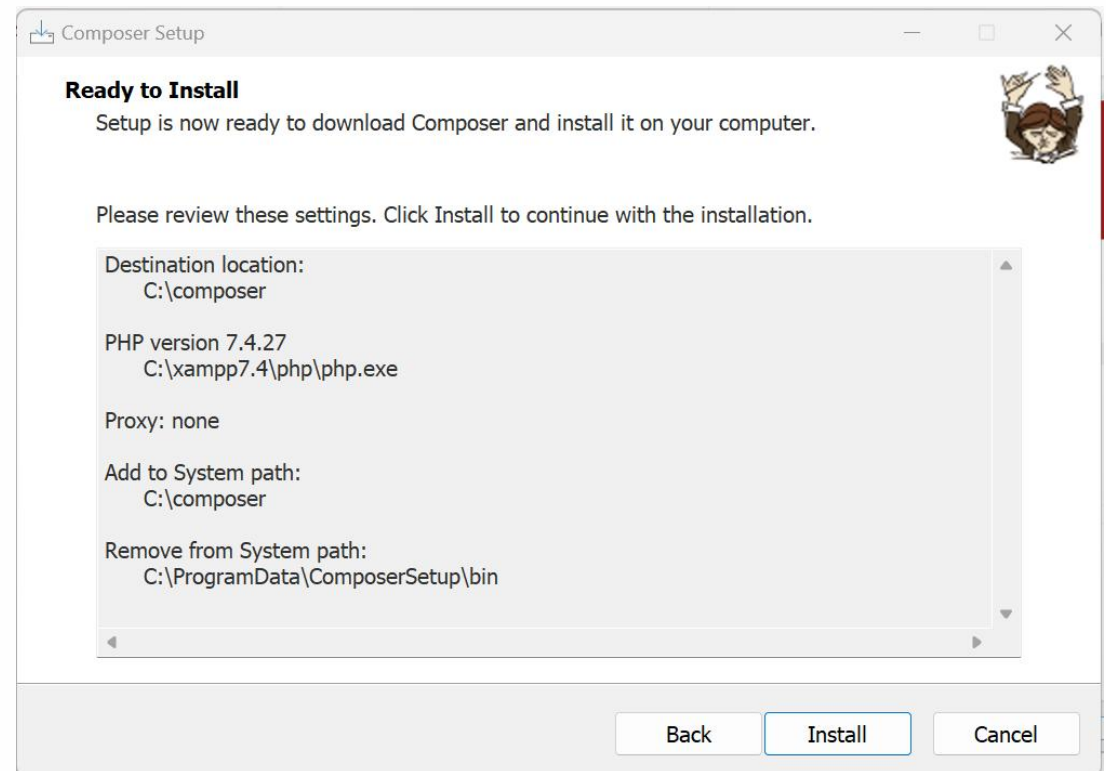
# Install Composer (4)

Jika muncul tampilan seperti ini langsung saja klik tombol next



# Install Composer (5)

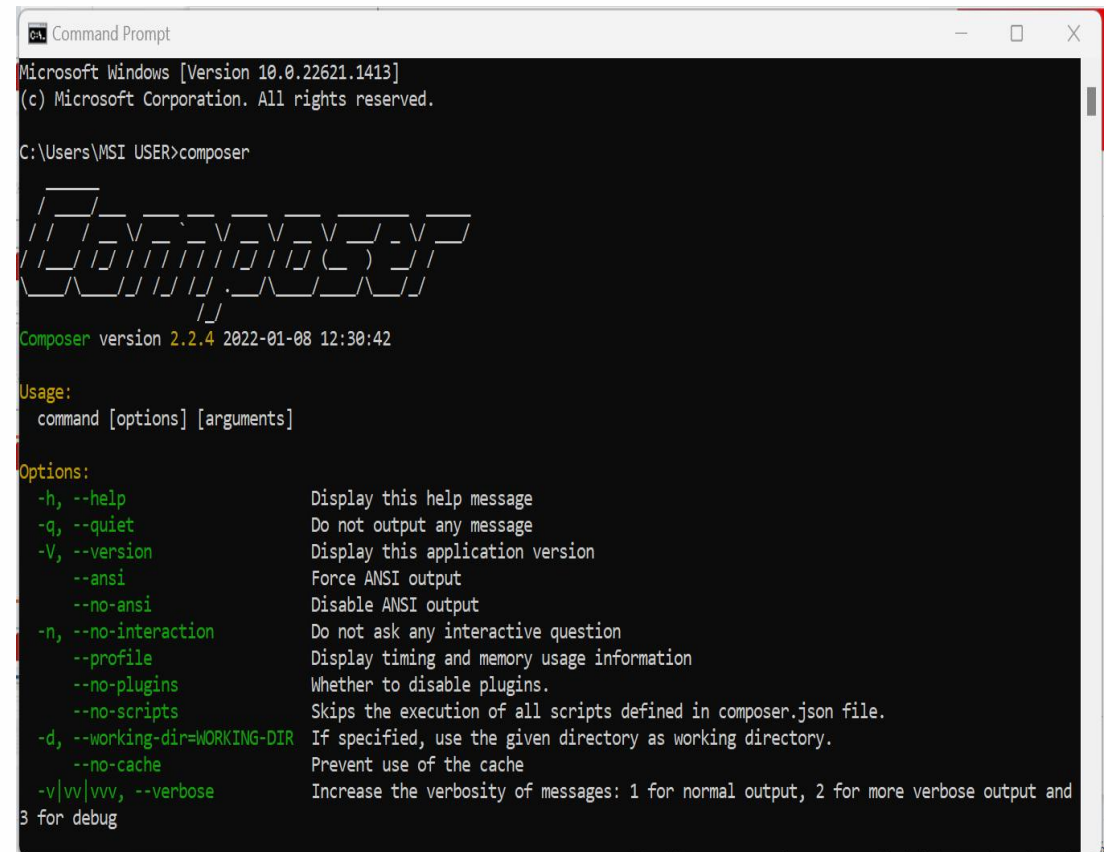
Jika muncul tampilan seperti ini  
silahkan klik tombol install





# Cek Composer

Untuk Mengetahui apakah composer sudah terinstall di kompoter atau laptop, silahkan buka CMD, dengan cara klik windows --> ketik cmd, setelah muncul tampilan cmd maka silahkan ketiki composer lalu lakukan enter



```
Command Prompt
Microsoft Windows [Version 10.0.22621.1413]
(c) Microsoft Corporation. All rights reserved.

C:\Users\MSI USER>composer

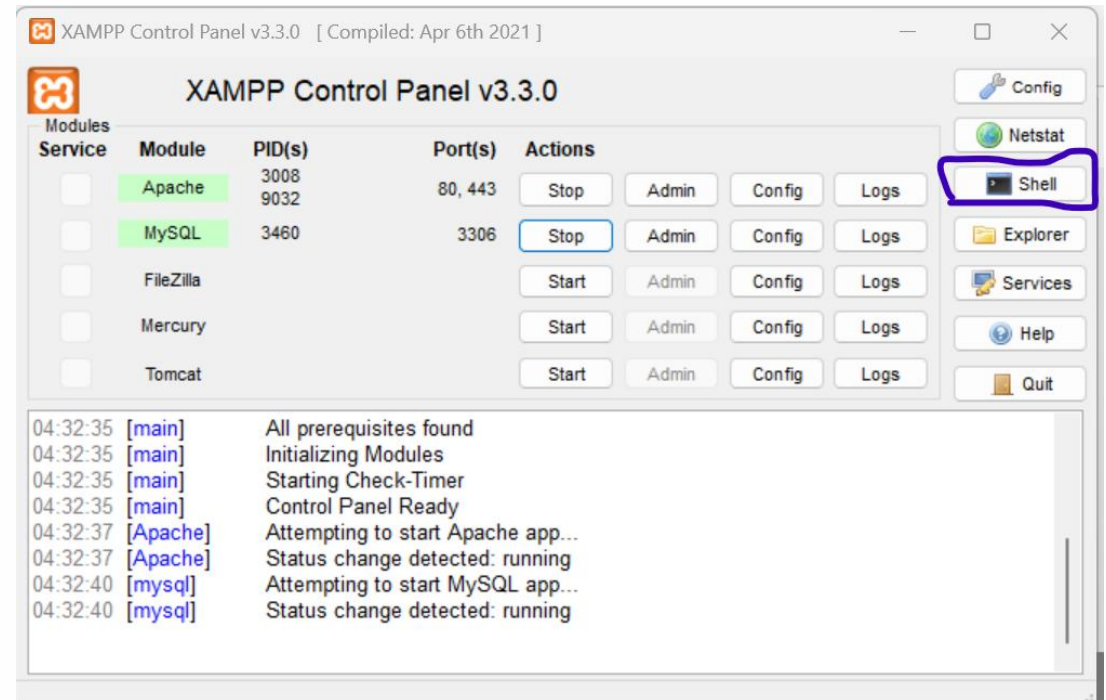
Composer version 2.2.4 2022-01-08 12:30:42

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
  --no-plugins               Whether to disable plugins.
  --no-scripts               Skips the execution of all scripts defined in composer.json file.
  -d, --working-dir=WORKING-DIR If specified, use the given directory as working directory.
  --no-cache                 Prevent use of the cache
  -v|vv|vvv, --verbose       Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and
                              3 for debug
```

# Install Laravel (1)

1. Silahkan aktifkan xampp terlebih dahulu
2. Silahkan klik tombol Shell seperti gambar di samping

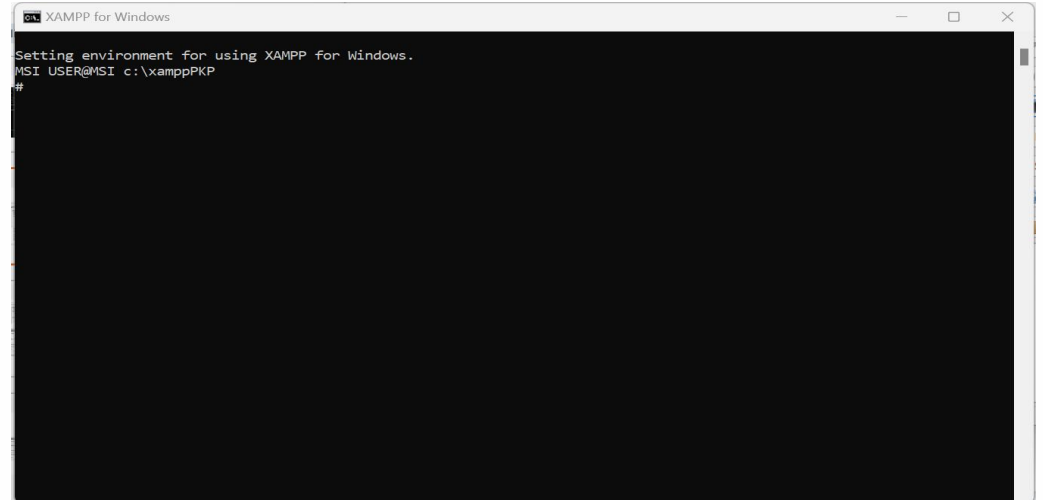


# Install Laravel (2)

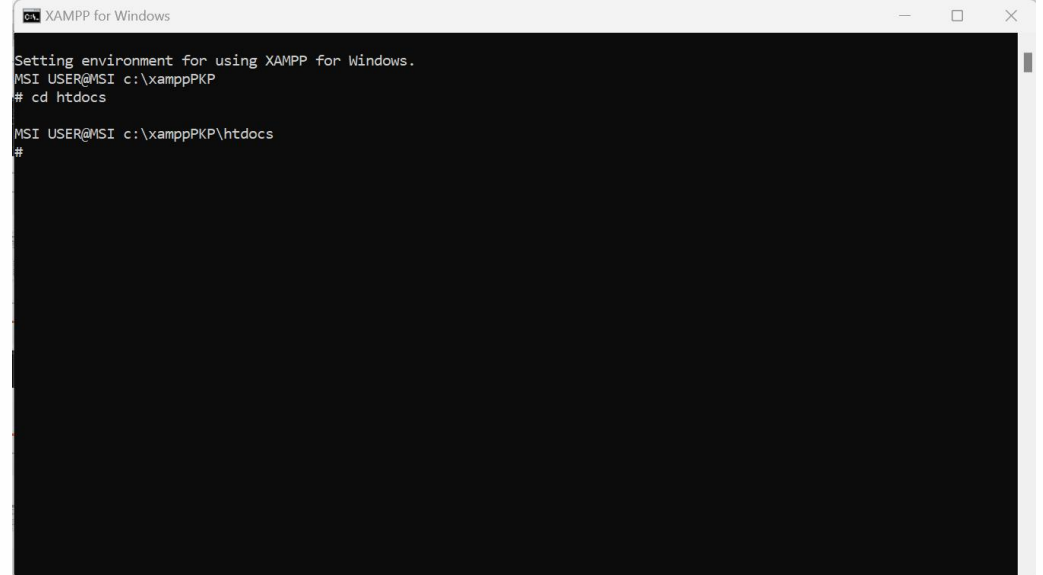
1. Maka akan muncul tampilan seperti ini

2. Silahkan ketik :

**cd htdocs**



```
XAMPP for Windows
Setting environment for using XAMPP for Windows.
MSI USER@MSI c:\xamppPKP
#
```



```
XAMPP for Windows
Setting environment for using XAMPP for Windows.
MSI USER@MSI c:\xamppPKP
# cd htdocs

MSI USER@MSI c:\xamppPKP\htdocs
#
```

# Install Laravel (3)

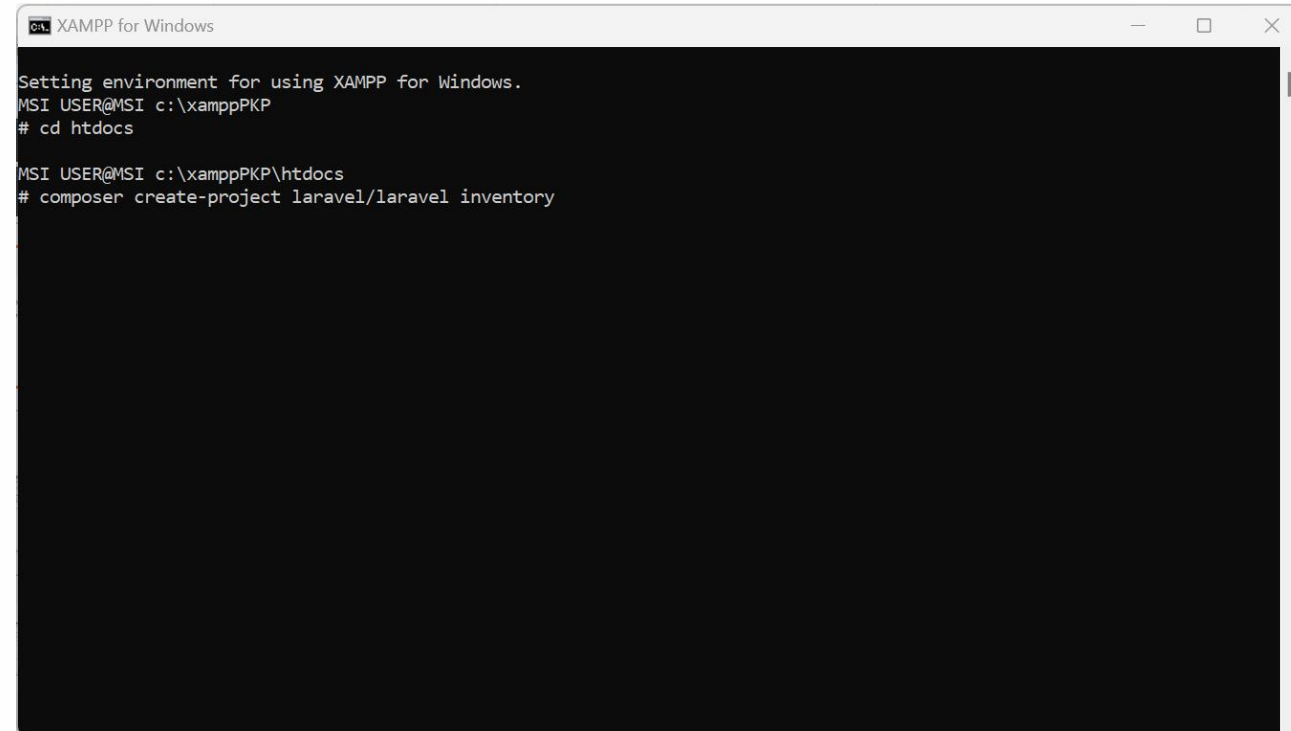
Silahkan ketik :

composer create-project

laravel/laravel nama\_sistem

contoh : composer create-project

laravel/laravel inventory



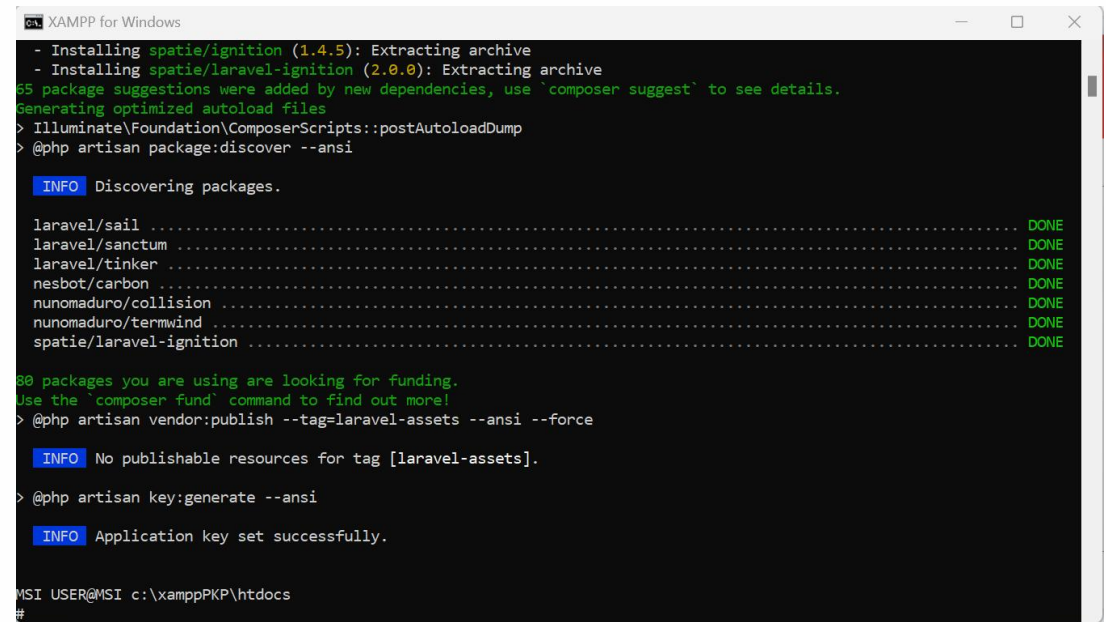
```
XAMPP for Windows

Setting environment for using XAMPP for Windows.
MSI USER@MSI c:\xamppPKP
# cd htdocs

MSI USER@MSI c:\xamppPKP\htdocs
# composer create-project laravel/laravel inventory
```

# Install Laravel (4)

Tunggu sampai selesai proses  
install laravelnya, seperti gambar  
di samping



```
XAMPP for Windows
- Installing spatie/ignition (1.4.5): Extracting archive
- Installing spatie/laravel-ignition (2.0.0): Extracting archive
65 package suggestions were added by new dependencies, use `composer suggest` to see details.
Generating optimized autoload files
> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi

 INFO  Discovering packages.

laravel/sail ..... DONE
laravel/sanctum ..... DONE
laravel/tinker ..... DONE
nesbot/carbon ..... DONE
nunomaduro/collision ..... DONE
nunomaduro/termwind ..... DONE
spatie/laravel-ignition ..... DONE

80 packages you are using are looking for funding.
Use the `composer fund` command to find out more!
> @php artisan vendor:publish --tag=laravel-assets --ansi --force

 INFO  No publishable resources for tag [laravel-assets].

> @php artisan key:generate --ansi

 INFO  Application key set successfully.

MSI USER@MSI c:\xamppPKP\htdocs
#
```

# Install Composer Pada Laravel

Setelah melakukan install laravel, silahkan lakukan install composer pada sistem laravel tersebut dengan cara:

1. Silahkan masuk ke sistem laravel, dengan ketik :  
cd inventory
2. Silahkan ketik composer install
3. Tunggu sampai selesai seperti gambar di samping

```
MSI USER@MSI c:\xamppPKP\htdocs
# cd inventory

MSI USER@MSI c:\xamppPKP\htdocs\inventory
#
```

```
MSI USER@MSI c:\xamppPKP\htdocs\inventory
# composer install
Installing dependencies from lock file (including require-dev)
Verifying lock file contents can be installed on current platform.
Nothing to install, update or remove
Generating optimized autoload files
> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi

 INFO  Discovering packages.

laravel/sail ..... DONE
laravel/sanctum ..... DONE
laravel/tinker ..... DONE
nesbot/carbon ..... DONE
nunomaduro/collision ..... DONE
nunomaduro/termwind ..... DONE
spatie/laravel-ignition ..... DONE

80 packages you are using are looking for funding.
Use the `composer fund` command to find out more!

MSI USER@MSI c:\xamppPKP\htdocs\inventory
#
```

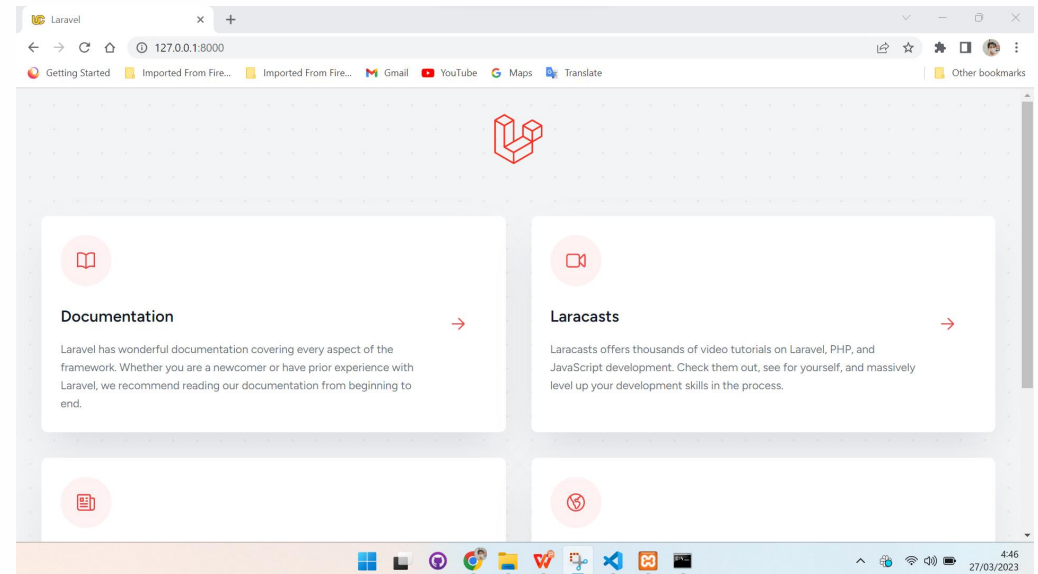
# Menjalankan Laravel

1. Setelah semua terinstall, silahkan jalankan laravel dengan ketik :  
  
php artisan serve
2. Silahkan ke browser dan akses url <http://127.0.0.1:8000>

```
MSI USER@MSI c:\xamppPKP\htdocs\inventory
# php artisan serve

INFO Server running on [http://127.0.0.1:8000].

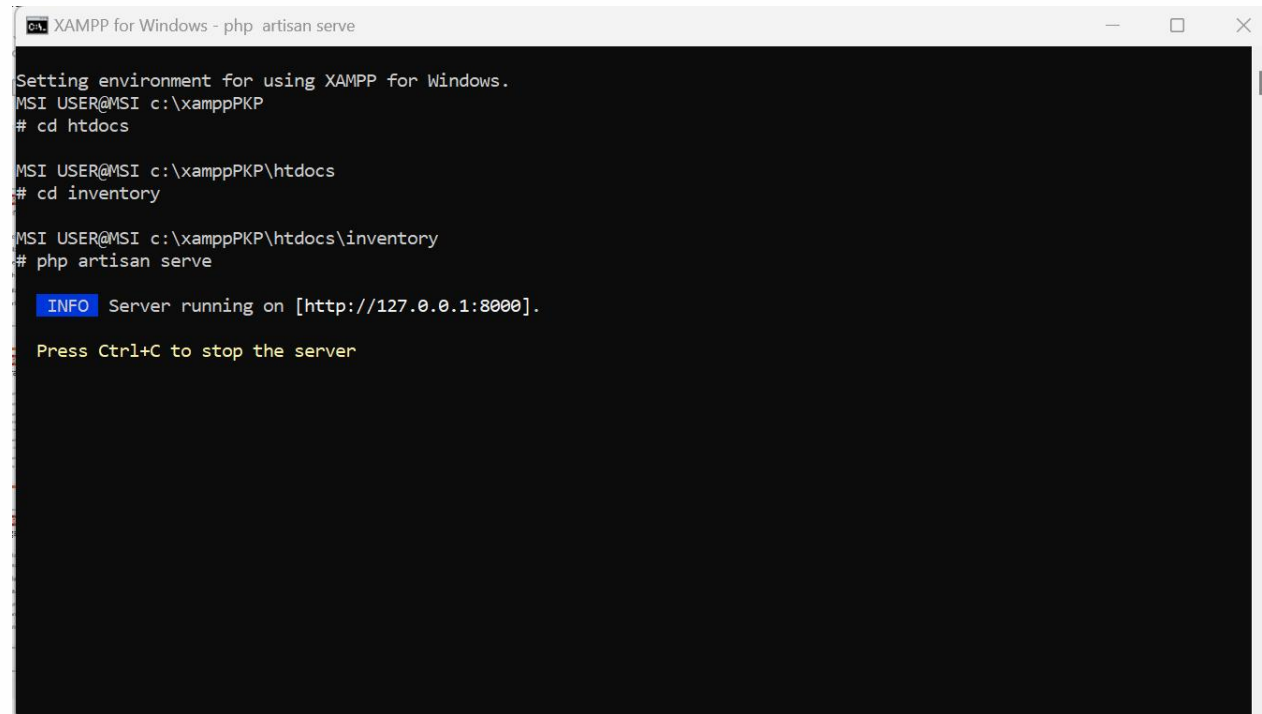
Press Ctrl+C to stop the server
```





# Cara Menjalankan Dari Awal

1. Silahkan Aktifkan Xampp
2. Silahkan klik tombol Shell pada Xampp
3. Silahkan masuk ke folder htdocs, dengan cara ketik :  
cd htdocs
4. Silahkan masuk ke folder sistem dengan cara ketik :  
cd inventory
5. Silahkan run sistem tersebut dengan cara ketik :  
php artisan serve
6. Silahkan akses url <http://127.0.0.1:8000> di browser



```
XAMPP for Windows - php artisan serve
Setting environment for using XAMPP for Windows.
MSI USER@MSI c:\xamppPKP
# cd htdocs

MSI USER@MSI c:\xamppPKP\htdocs
# cd inventory

MSI USER@MSI c:\xamppPKP\htdocs\inventory
# php artisan serve

INFO Server running on [http://127.0.0.1:8000].

Press Ctrl+C to stop the server
```



# Tugas

Silahkan masukan sistem laravel yang sudah dibuat ke dalam github dengan cara

1. buat branch baru pada github
2. format nama branch yaitu pertemuan-17-nama-lenhkap contoh pertemuan-17-abdul-falaq
3. Silahkan masukan sistem laravel
4. Silahkan url github kirim ke portal

