LARAVEL 5

Chandra Ramdhan P

2018

INFO

Chandra Ramdhan Purnama

CONTACT

@Chandra_ramdhan

160613024@gmail.com

0838-2189-6126

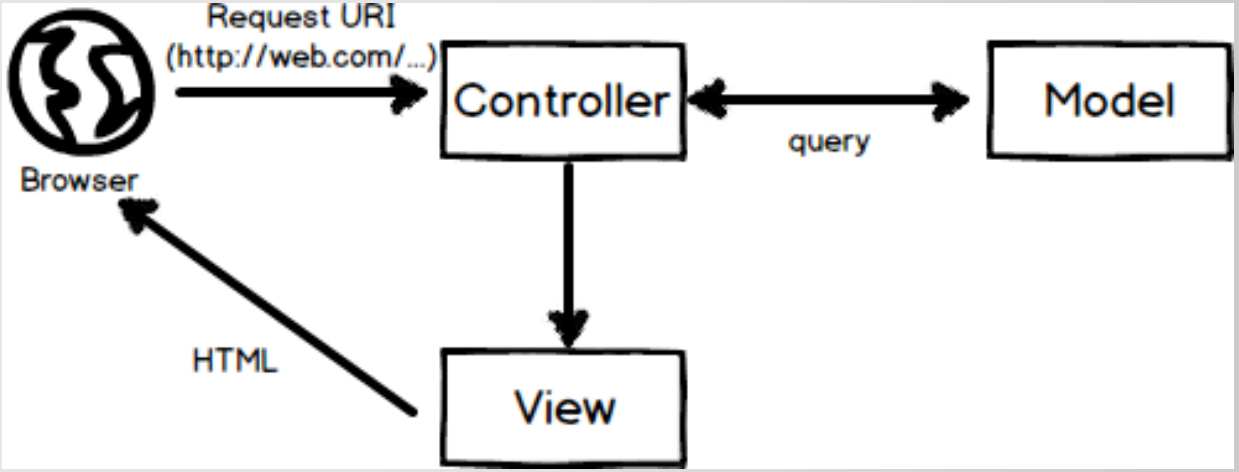
LARAVEL

- Sebuah Open Source Framework
- Laravel terkenal sederhana dan elegan karena pembuatannya memang ditujukan untuk *end-user (buat yang pernah ngoding)*
- *"The PHP Framework For Web Artisans"*
  - Laravel dibuat untuk orang – orang yang menyukai seni.

NIAGAHOSTER
Keuntungan menggunakan Laravel
1
Feature yang sudah terintegrasi
2
Memiliki popularitas
3
Perpindahan Database
4
Micro Framework Support
5
Versi terbaru PHP

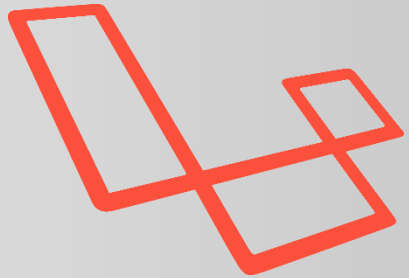
KEUNTUNGAN

MODEL VIEW CONTROLLER



- Alur kerja aplikasi web kita ketika user mengunjungi salah satu halaman yaitu:
  - Browser berhubungan dengan *server* untuk akses halaman.
  - *Request* (permintaan) browser ditangani oleh bagian Controller dari kode kita.
  - Controller akan melakukan pemanggilan ke Model untuk mendapatkan data yang relevan, dan kemudian mempersiapkan data tersebut untuk ditampilkan.
  - Controller memberikan data yang diperlukan kepada view.
  - View menampilkan data dan berbagai elemen antarmuka tambahan yang diperlukan.

KEPERLUAN



- Web Server
- PHP >= 7.1.3
- Database Server
- Tokenizer PHP Extension
- OpenSSL PHP Extension
- PDO PHP Extension
- Ctype PHP Extension
- JSON PHP Extension
- Composer
- INTERNET + KUOTA

LARAVEL 5

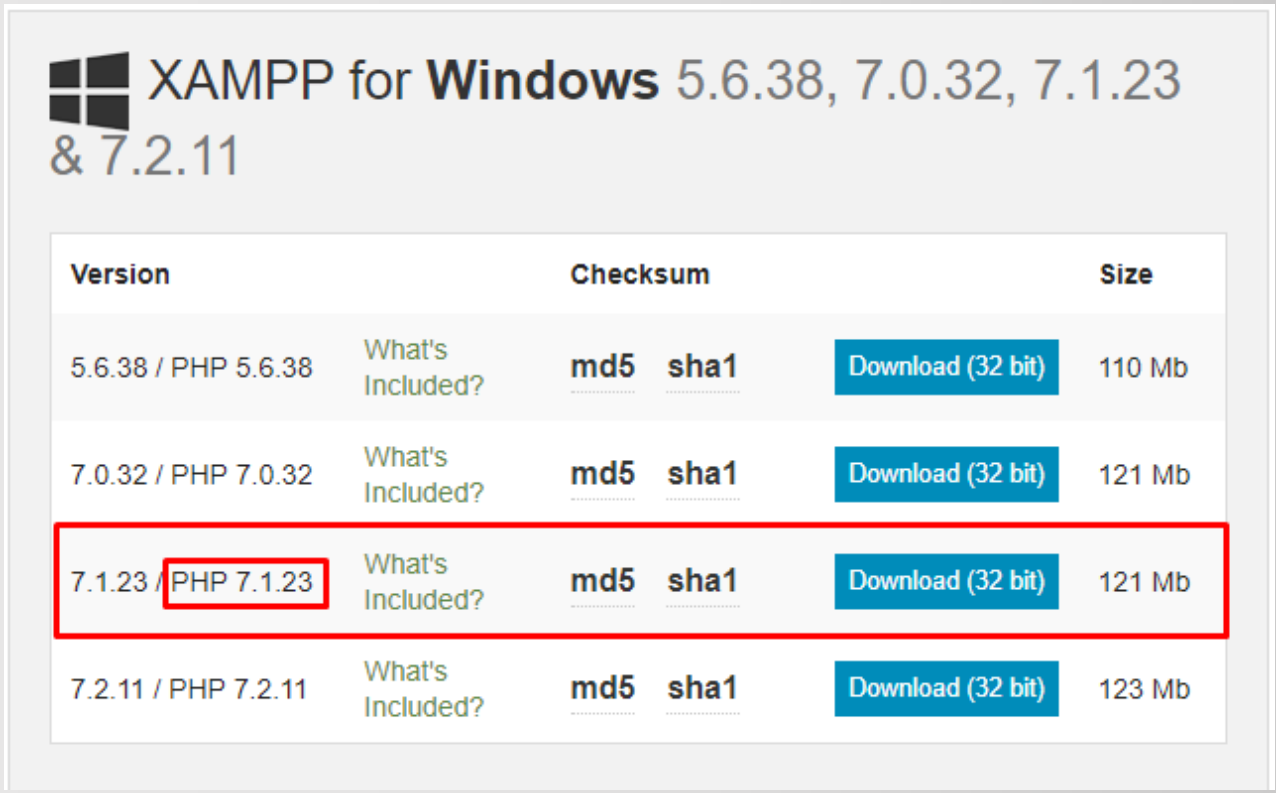
INSTALASI

OUTLINE

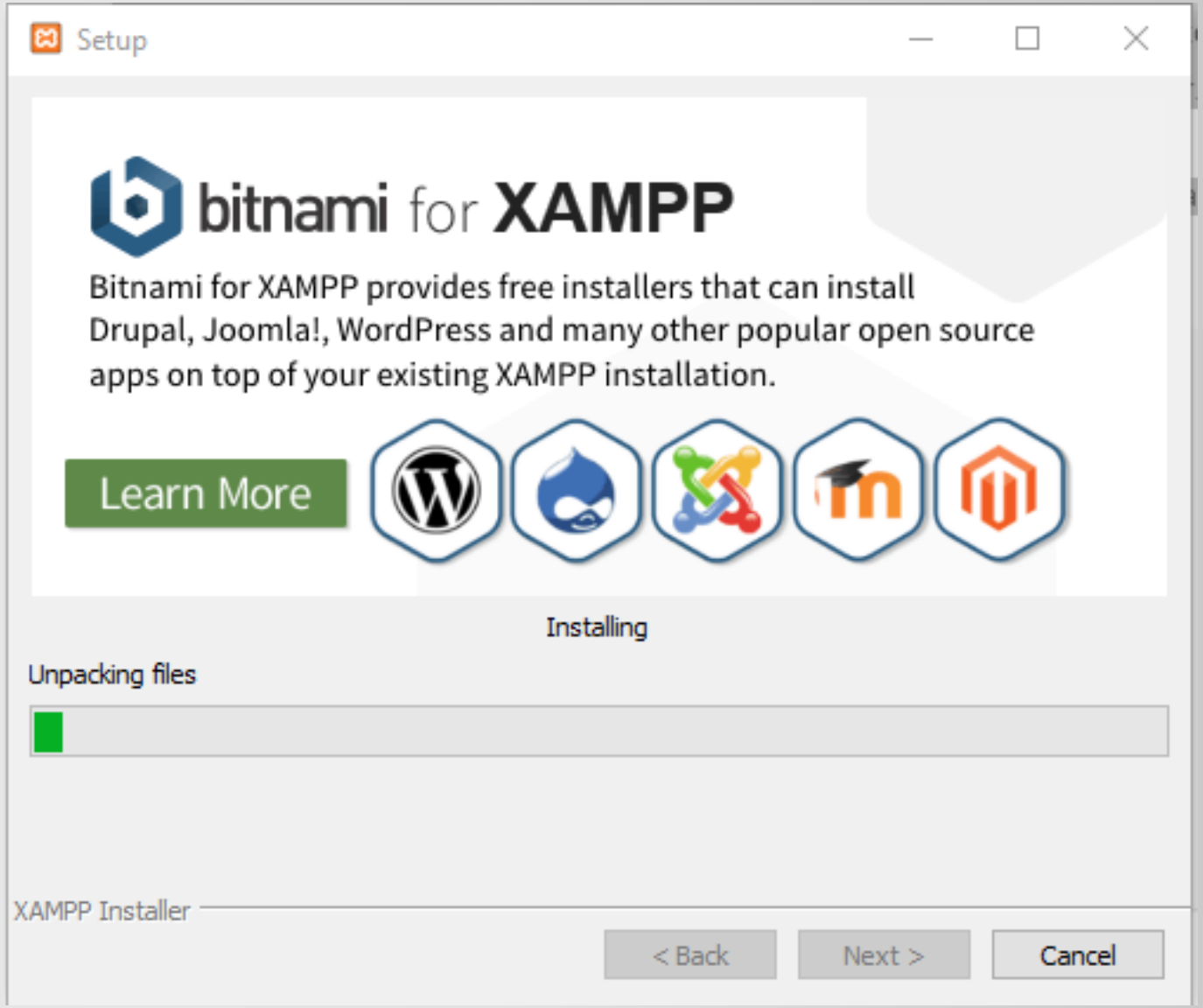
1. INSTALL XAMPP
2. INSTALL COMPOSER
3. INSTALL LARAVEL
4. MEMBUAT DATABASE
5. KONEKSI DATABASE
6. MEMBUAT VIEW
7. KONFIGURASI ROUTE
8. MEMBUAT FORM
9. CONTROLLER
10. MODEL
11. YIELD

INSTALL XAMPP

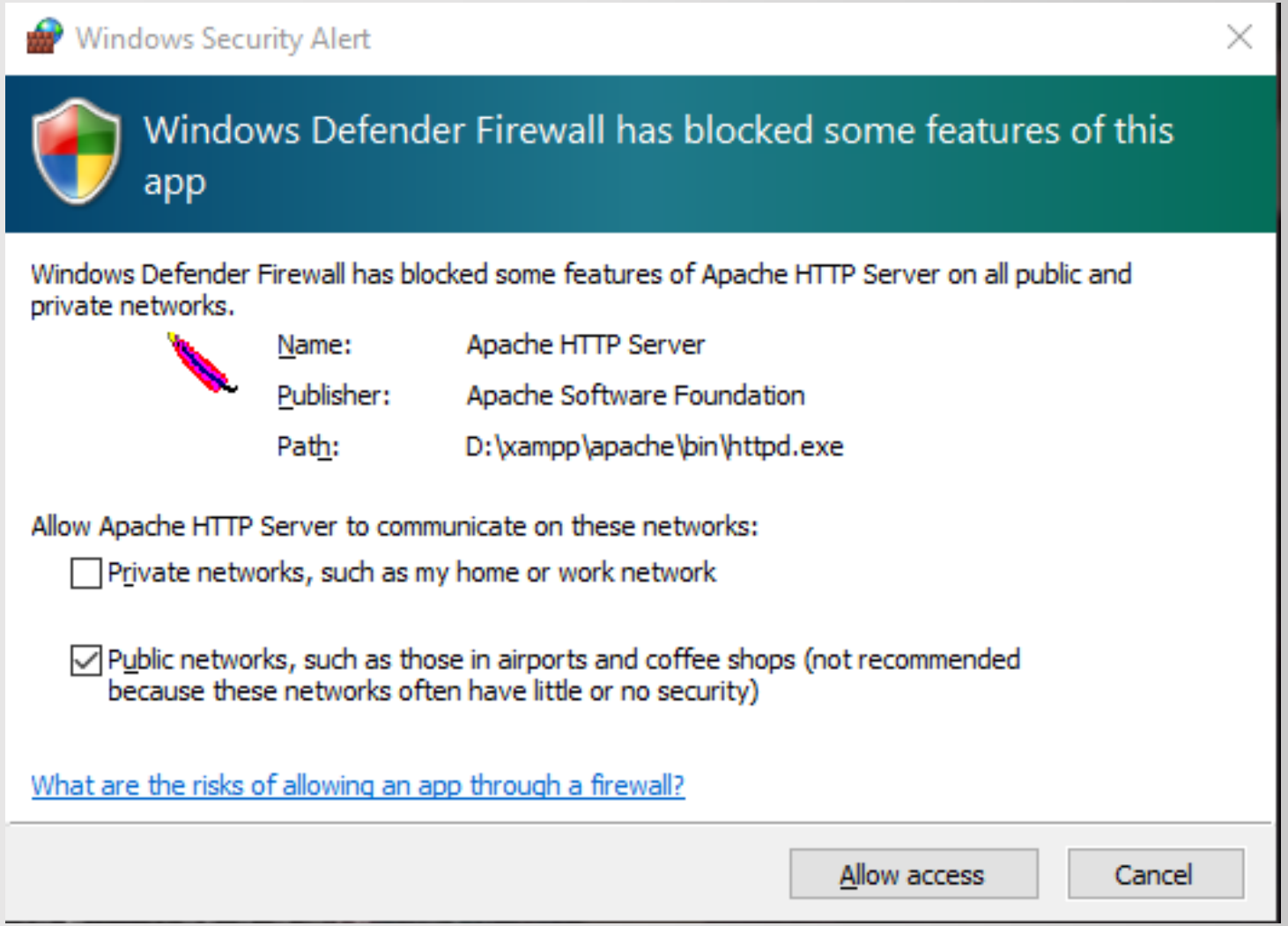
Download xampp di https://www.apachefriends.org/download.html



INSTALL XAMPP

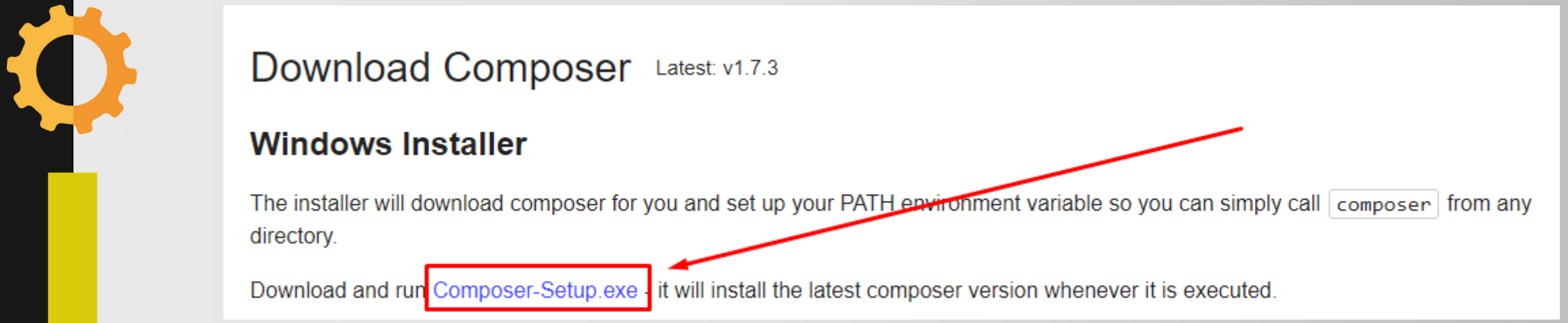


INSTALL XAMPP

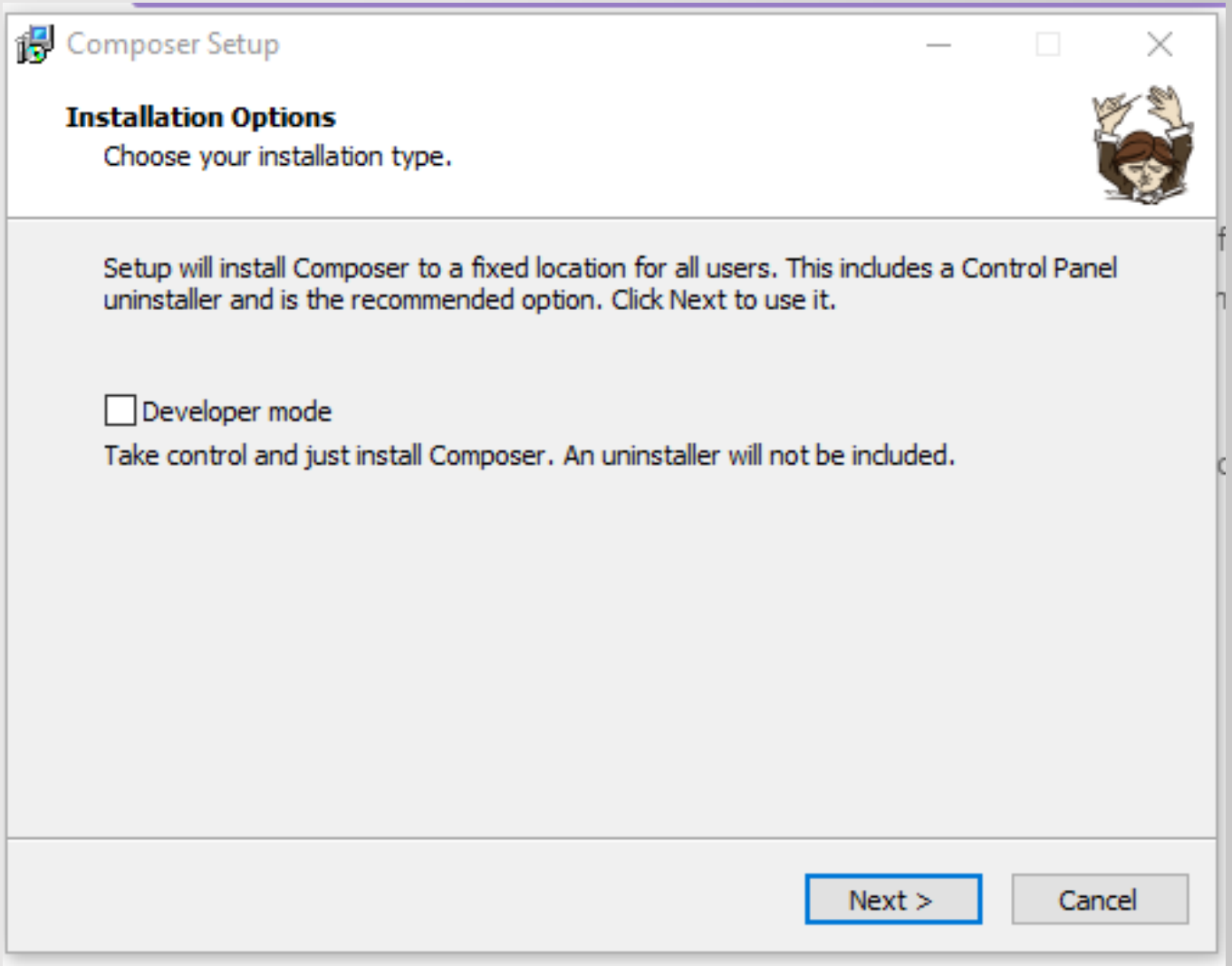


INSTALL COMPOSER

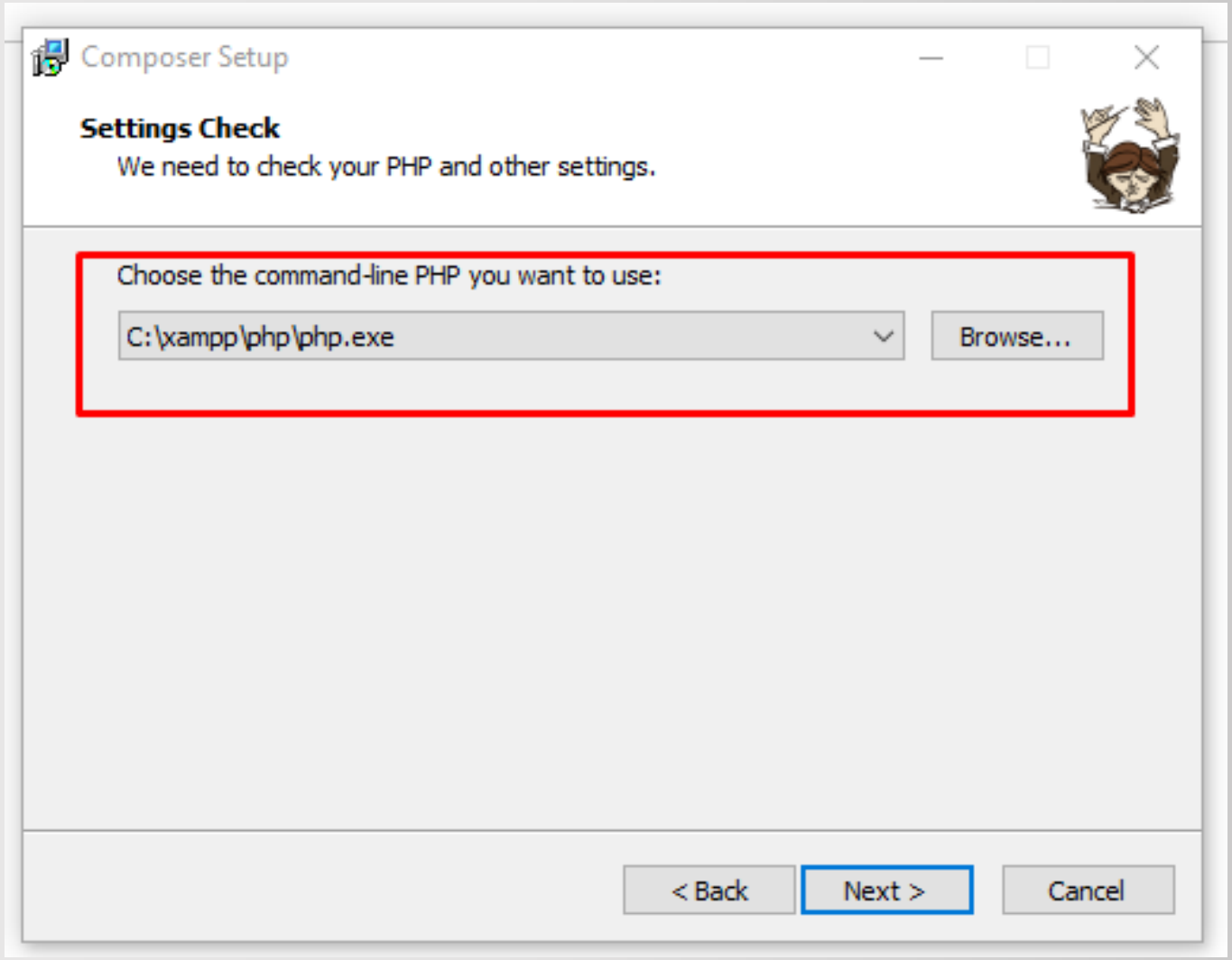
Download Composer di
https://getcomposer.org/download/



INSTALL COMPOSER



INSTALL COMPOSER



INSTALL LARAVEL

Buka laravel.com
Copy syntax

Installing Laravel

Laravel utilizes Composer to manage its dependencies. So, before using Laravel, make sure you have Composer installed on your machine.

Via Laravel Installer

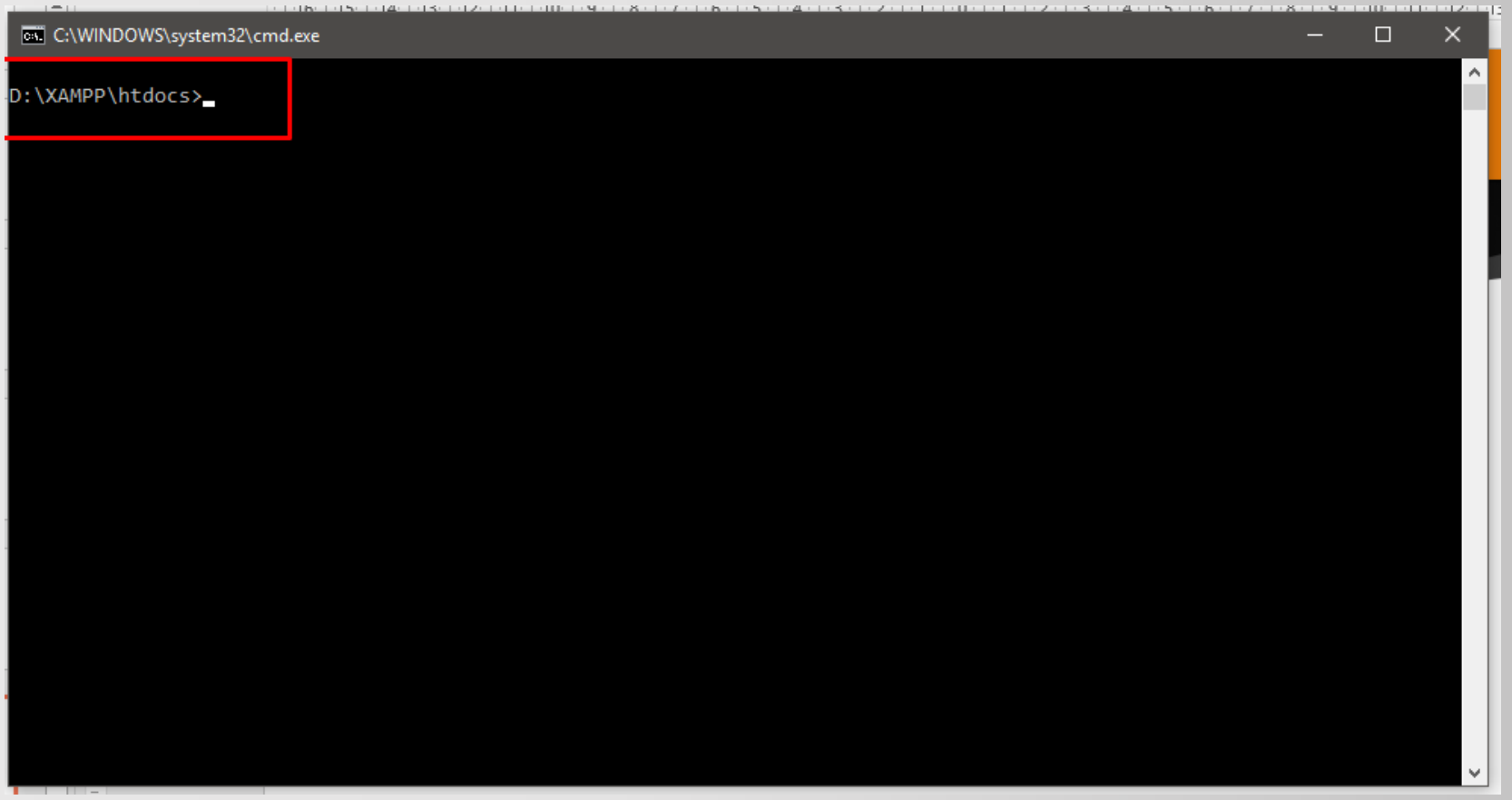
First, download the Laravel installer using Composer:

```
composer global require laravel/installer
```

Make sure to place composer's system-wide vendor bin directory in your `$PATH` so the laravel executable can be located by your system. This directory exists in different locations based on your operating system; however, some common locations include:

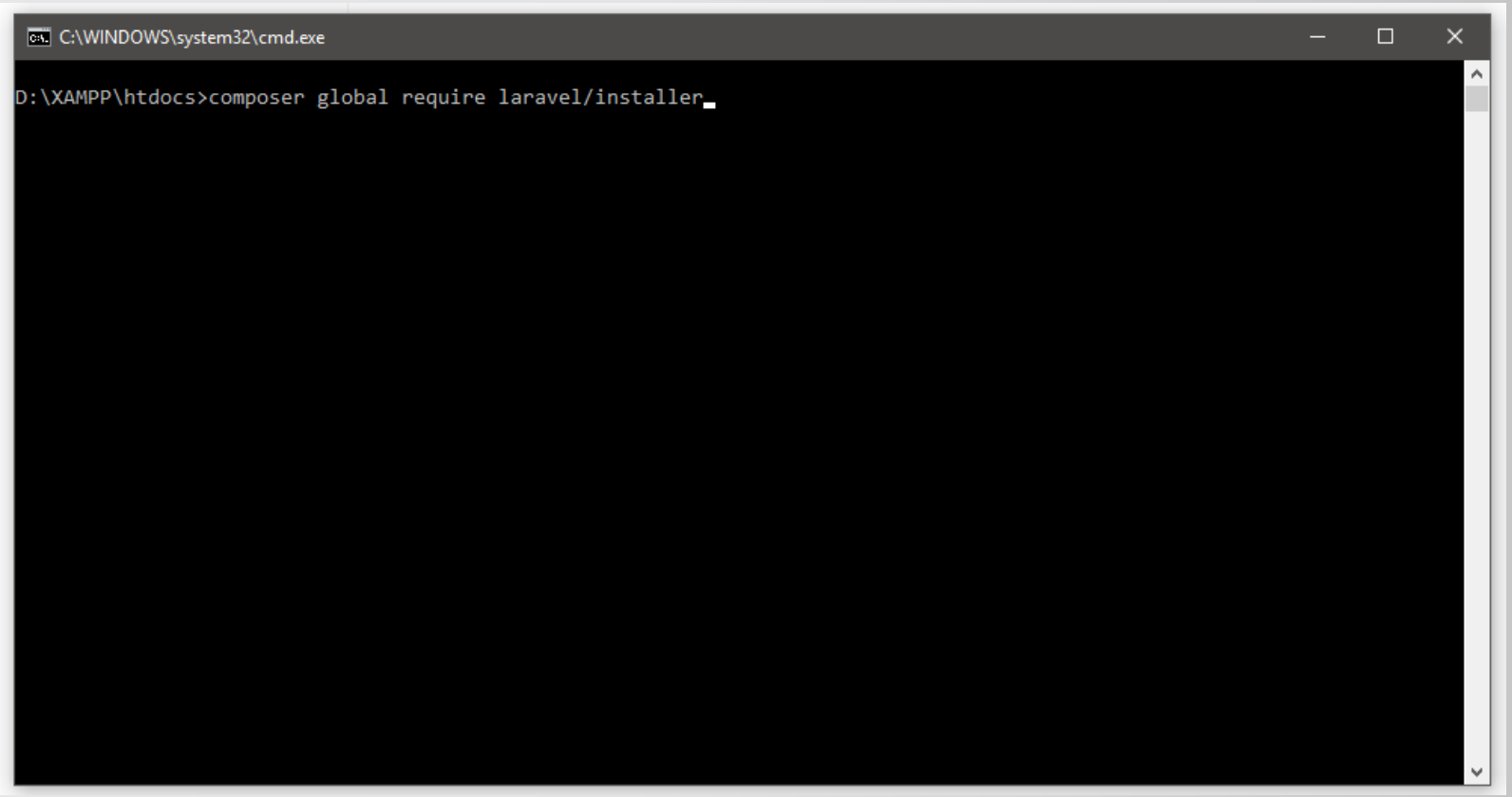
INSTALL LARAVEL

- Buka CMD pada folder HTDOCS xampp



INSTALL LARAVEL

- PASTE pada cmd dan enter...



INSTALL LARAVEL

- Masih di CMD copy syntax ini di cmd

- macOS: `$HOME/.composer/vendor/bin`
- GNU / Linux Distributions: `$HOME/.config/composer/vendor/bin`

Once installed, the `laravel new` command will create a fresh Laravel installation in the directory you specify. For instance, `laravel new blog` will create a directory named `blog` containing a fresh Laravel installation with all of Laravel's dependencies already installed:

```
laravel new blog
```

INSTALL LARAVEL

- Masukkan laravel new CRUD

C:\WINDOWS\system32\cmd.exe - laravel new CRUD

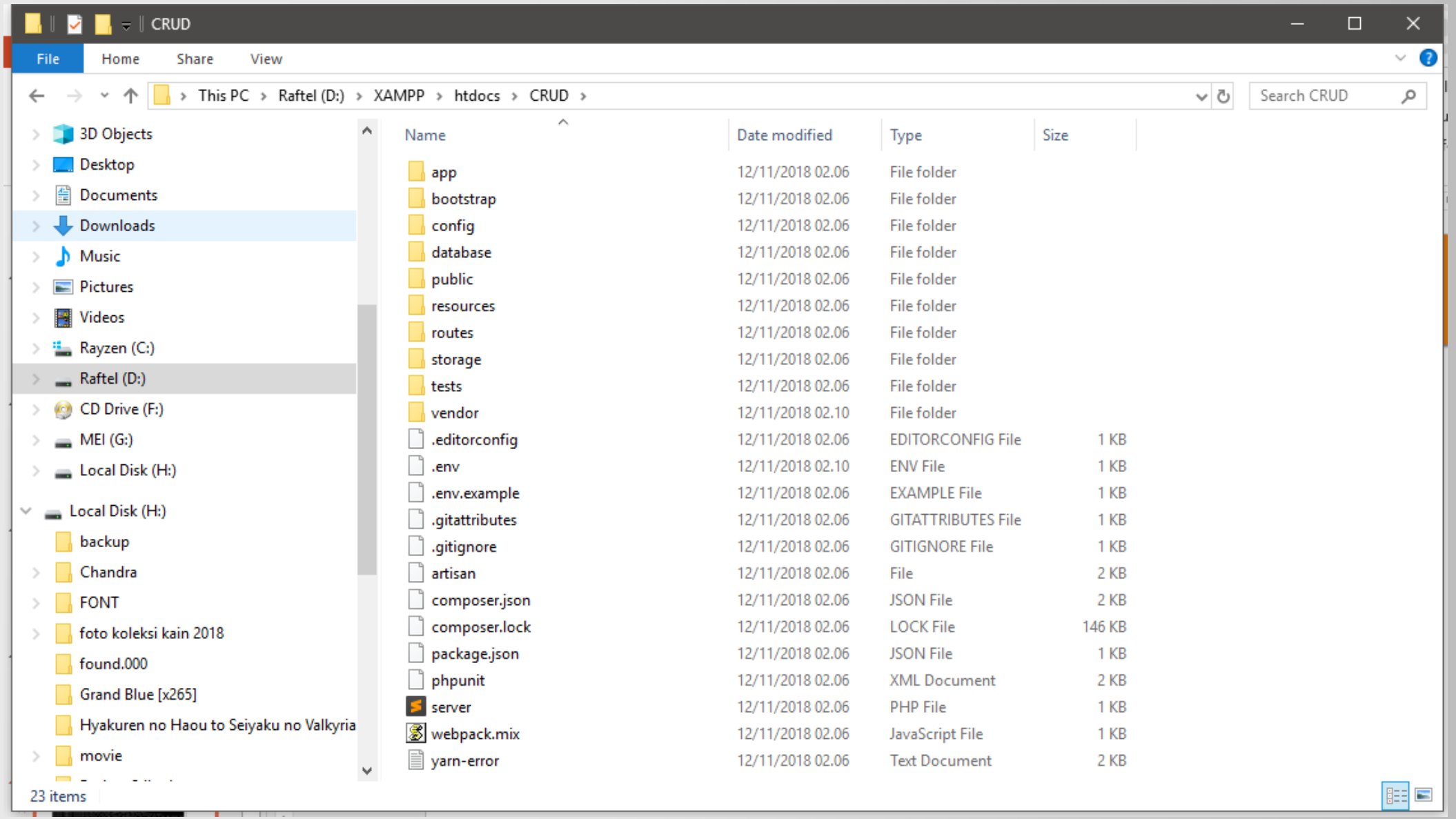
```
D:\XAMPP\htdocs>composer global require laravel/installer
Changed current directory to C:/Users/Chunz-PC/AppData/Roaming/Composer
Using version ^2.0 for laravel/installer
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
Nothing to install or update
Generating autoload files

D:\XAMPP\htdocs>laravel new CRUD
Crafting application...
Loading composer repositories with package information
Installing dependencies (including require-dev) from lock file
Package operations: 72 installs, 0 updates, 0 removals
  - Installing doctrine/inflector (v1.3.0): Loading from cache
  - Installing doctrine/lexer (v1.0.1): Loading from cache
  - Installing dragonmantank/cron-expression (v2.2.0): Loading from cache
  - Installing erusev/parsedown (1.7.1): Loading from cache
  - Installing vlucas/phpdotenv (v2.5.1): Loading from cache
  - Installing symfony/css-selector (v4.1.7): Downloading (100%)
  - Installing tijsverkoyen/css-to-inline-styles (2.2.1): Loading from cache
  - Installing symfony/polyfill-php72 (v1.10.0): Downloading (100%)
```

INSTALL LARAVEL

C:\WINDOWS\system32\cmd.exe

```
laravel/framework suggests installing pusher/pusher-php-server (Required to use the Pusher broadcast driver (^3.0).)
laravel/framework suggests installing symfony/dom-crawler (Required to use most of the crawler integration testing tools
 (^4.1).)
laravel/framework suggests installing symfony/psr-http-message-bridge (Required to psr7 bridging features (^1.0).)
psy/psysh suggests installing ext-pcntl (Enabling the PCNTL extension makes PsySH a lot happier :))
psy/psysh suggests installing ext-pdo-sqlite (The doc command requires SQLite to work.)
psy/psysh suggests installing ext-posix (If you have PCNTL, you'll want the POSIX extension as well.)
psy/psysh suggests installing hoa/console (A pure PHP readline implementation. You'll want this if your PHP install does
n't already support readline or libedit.)
filp/whoops suggests installing whoops/soap (Formats errors as SOAP responses)
sebastian/global-state suggests installing ext-uopz (*)
phpunit/php-code-coverage suggests installing ext-xdebug (^2.6.0)
phpunit/phpunit suggests installing ext-soap (*)
phpunit/phpunit suggests installing ext-xdebug (*)
phpunit/phpunit suggests installing phpunit/php-invoker (^2.0)
Generating optimized autoload files
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
> @php artisan key:generate --ansi
Application key set successfully.
> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi
Discovered Package: beyondcode/laravel-dump-server
Discovered Package: fideloper/proxy
Discovered Package: laravel/tinker
Discovered Package: nesbot/carbon
Discovered Package: nunomaduro/collision
Package manifest generated successfully.
Application ready! Build something amazing.

D:\XAMPP\htdocs>
```

INSTALL LARAVEL



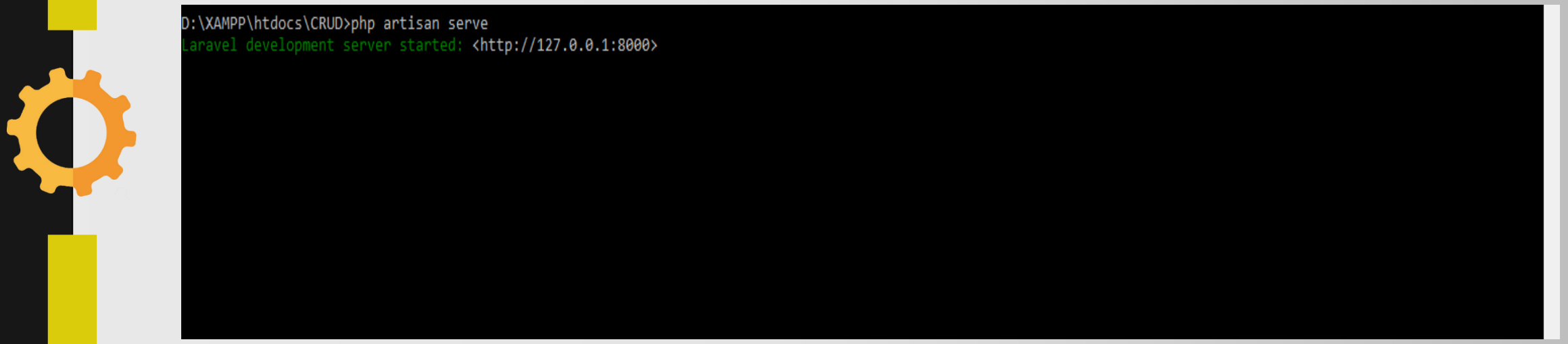
ARTISAN

C:\WINDOWS\system32\cmd.exe

```
D:\XAMPP\htdocs\CRUD>php artisan
Laravel Framework 5.7.13

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
      --env[=ENV]       The environment the command should run under
  -v|vv|vvv, --verbose  Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and 3 for debug

Available commands:
  clear-compiled       Remove the compiled class file
  down                 Put the application into maintenance mode
  dump-server          Start the dump server to collect dump information.
  env                  Display the current framework environment
  help                 Displays help for a command
  inspire              Display an inspiring quote
  list                 Lists commands
  migrate              Run the database migrations
  optimize             Cache the framework bootstrap files
  preset               Swap the front-end scaffolding for the application
  serve                Serve the application on the PHP development server
  tinker               Interact with your application
  up                   Bring the application out of maintenance mode
 app
  app:name             Set the application namespace
 auth
  auth:clear-resets    Flush expired password reset tokens
 cache
  cache:clear          Flush the application cache
  cache:forget         Remove an item from the cache
  cache:table          Create a migration for the cache database table
 config
  config:cache         Create a cache file for faster configuration loading
  config:clear         Remove the configuration cache file
 db
  db:seed              Seed the database with records
```

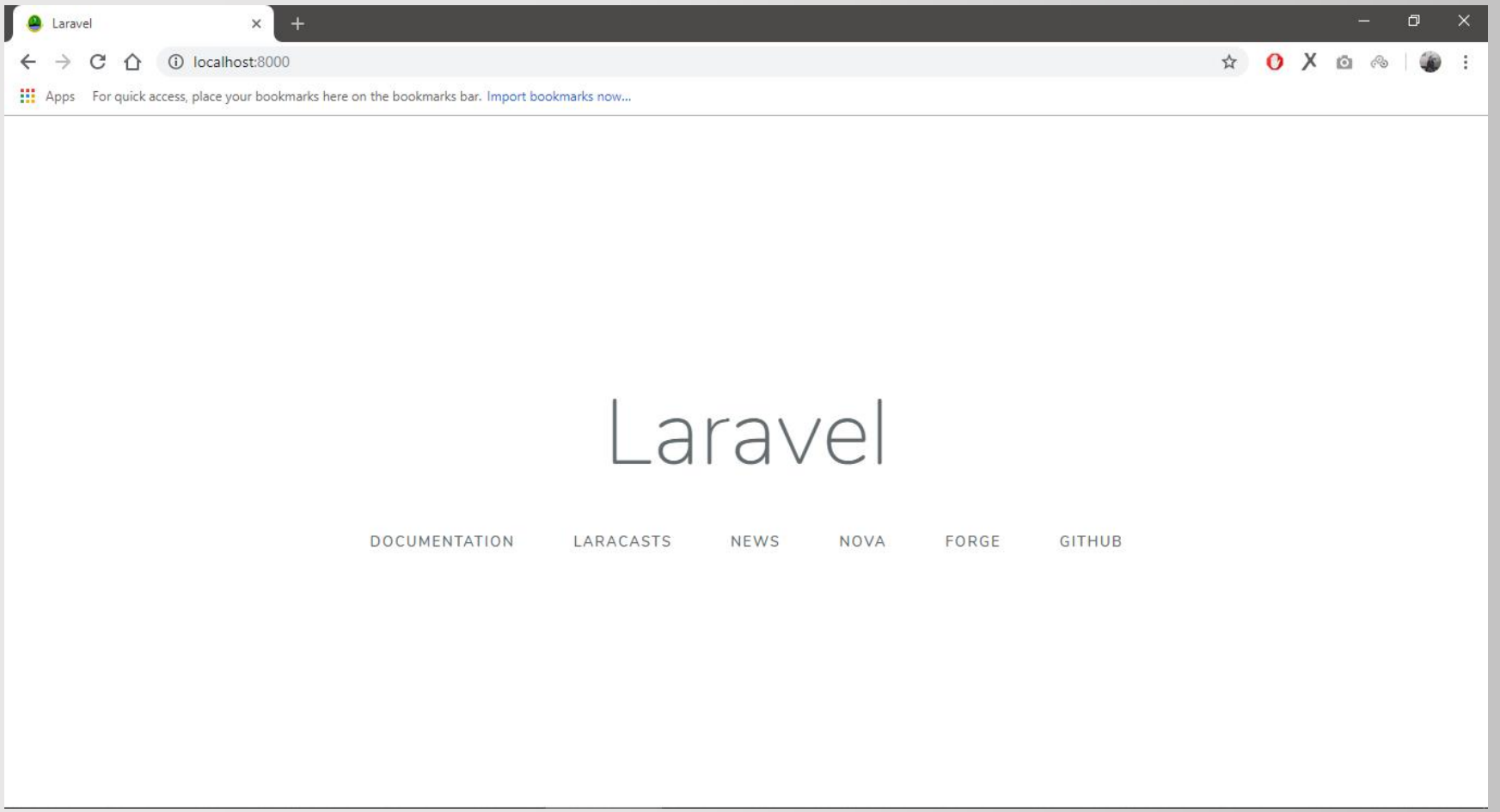
ARTISAN SERVE

- PHP ARTISAN SERVE

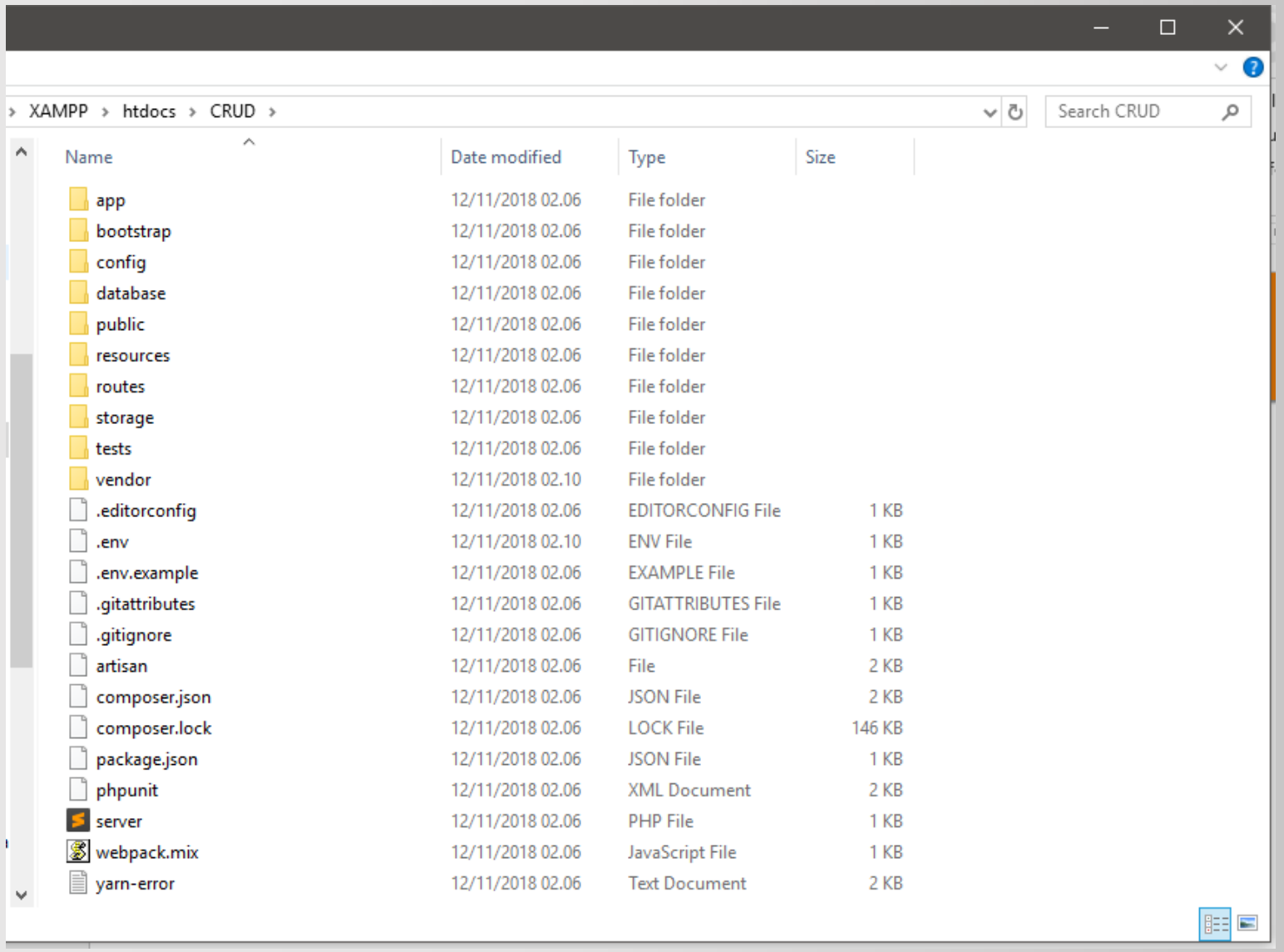


SELAMAT LARAVEL ANDA SUDAH SIAP

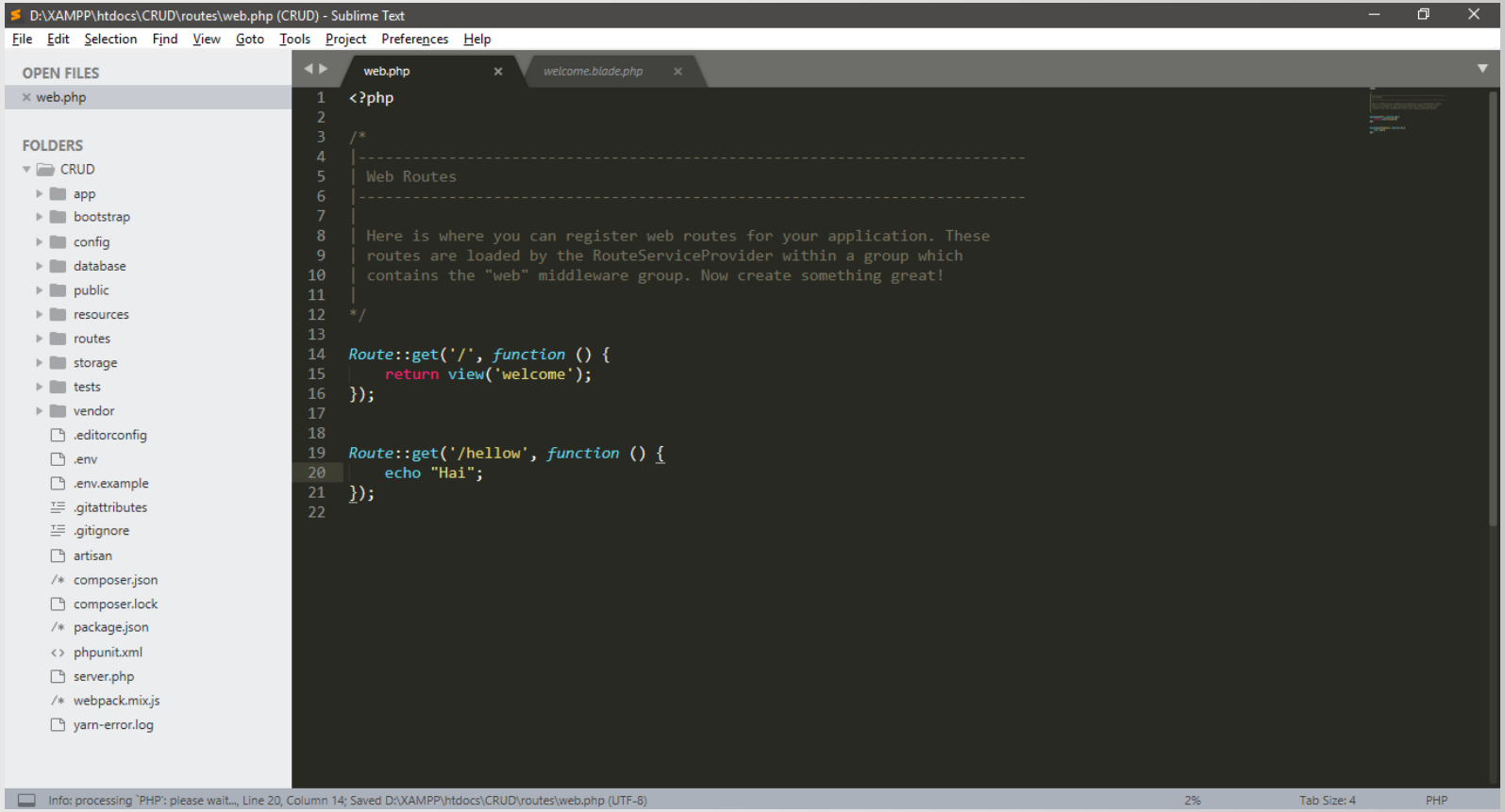
- Localhost:8000



DIRECTORY LARAVEL

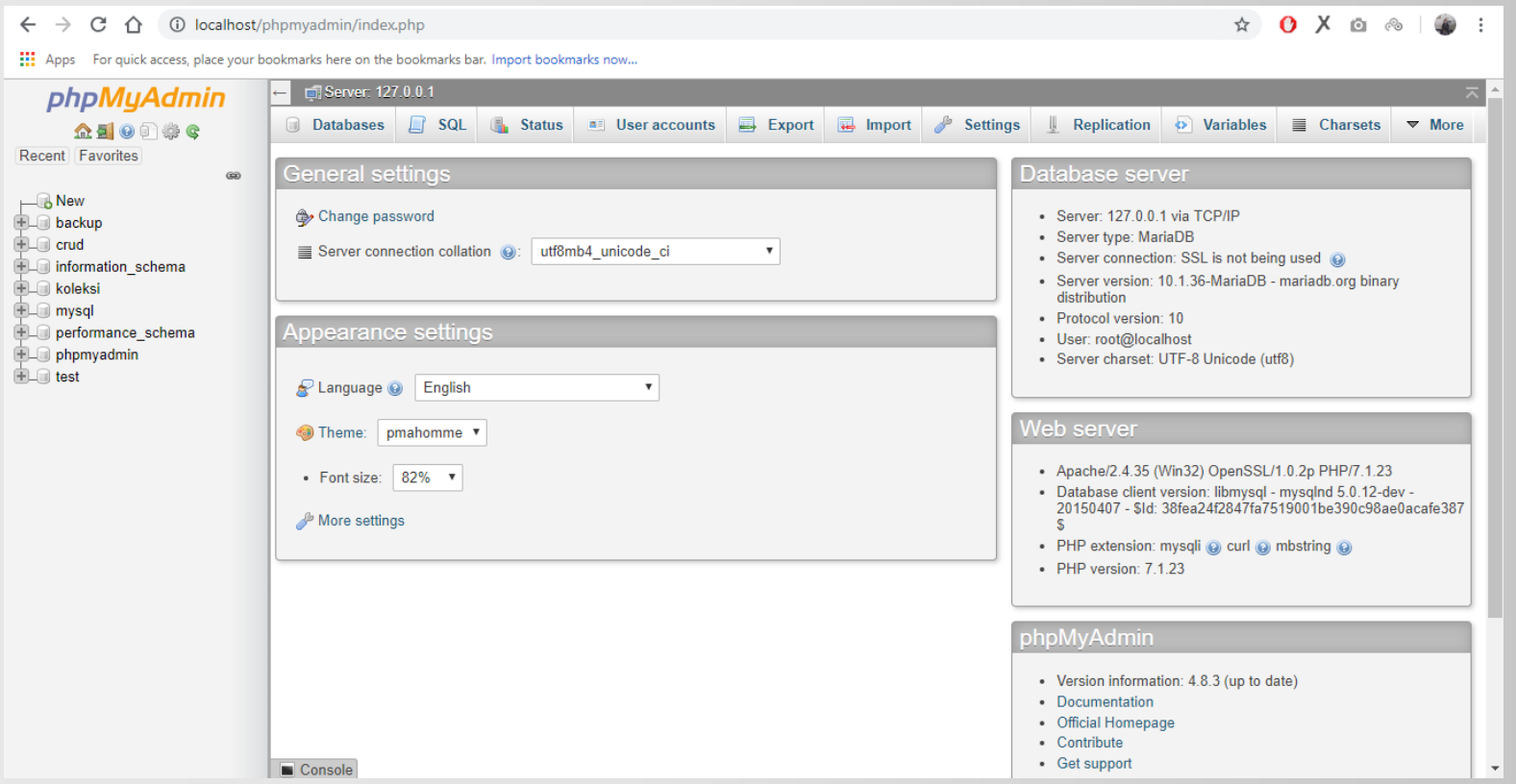


ROUTE


D:\XAMPP\htdocs\CRUD\routes\web.php (CRUD) - Sublime Text
File Edit Selection Find View Goto Tools Project Preferences Help
OPEN FILES
web.php
FOLDERS
CRUD
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
server.php
webpack.mix.js
yarn-error.log
web.php
welcome.blade.php
<?php
/*
| Web Routes
Here is where you can register web routes for your application. These
routes are loaded by the RouteServiceProvider within a group which
contains the "web" middleware group. Now create something great!
*/
Route::get('/', function () {
return view('welcome');
});
Route::get('/hellow', function () {
echo "Hai";
});
Info: processing `PHP`: please wait..., Line 20, Column 14; Saved D:\XAMPP\htdocs\CRUD\routes\web.php (UTF-8)
2%
Tab Size: 4
PHP

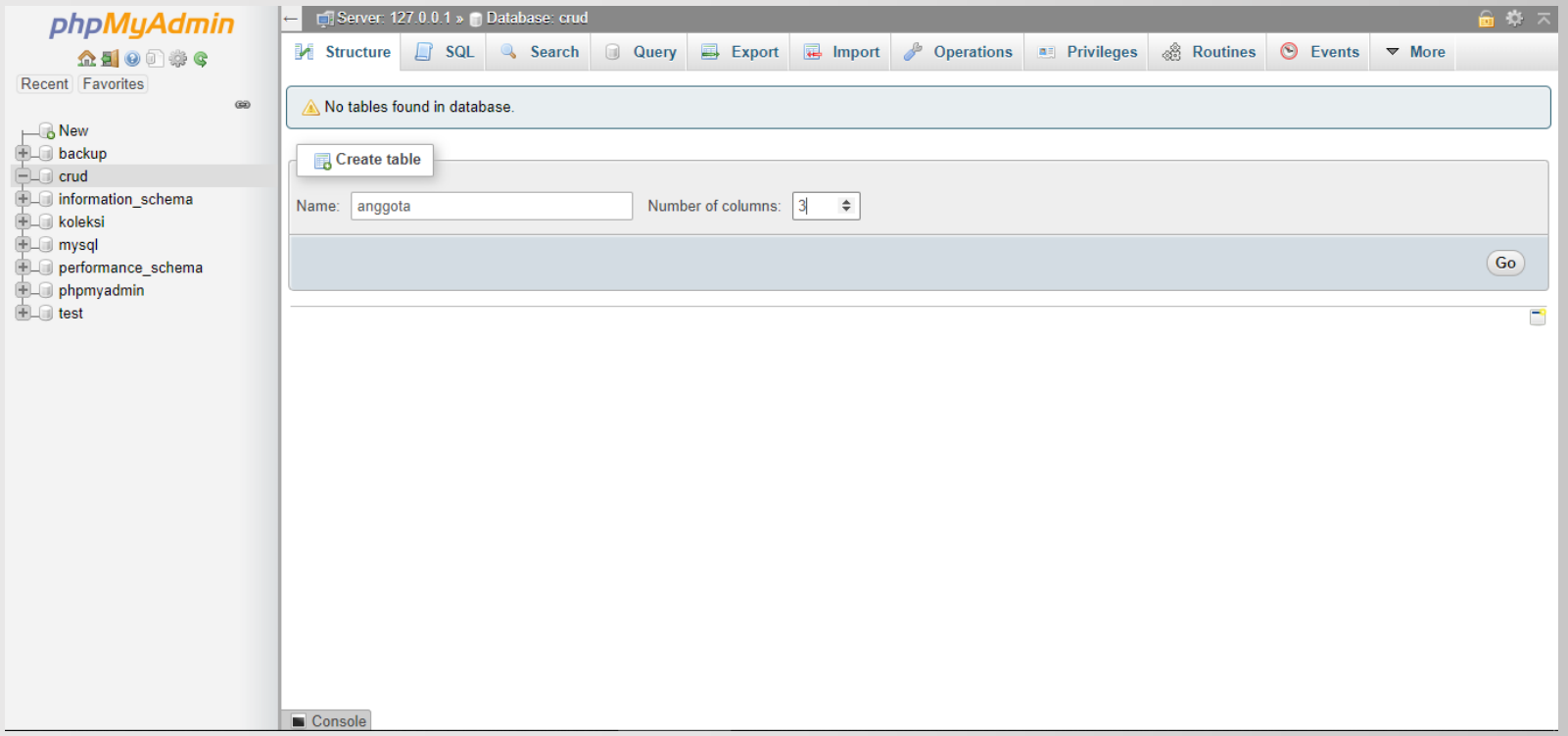
MEMBUAT DATABASE

Masuk ke localhost/phpmyadmin dan set up database

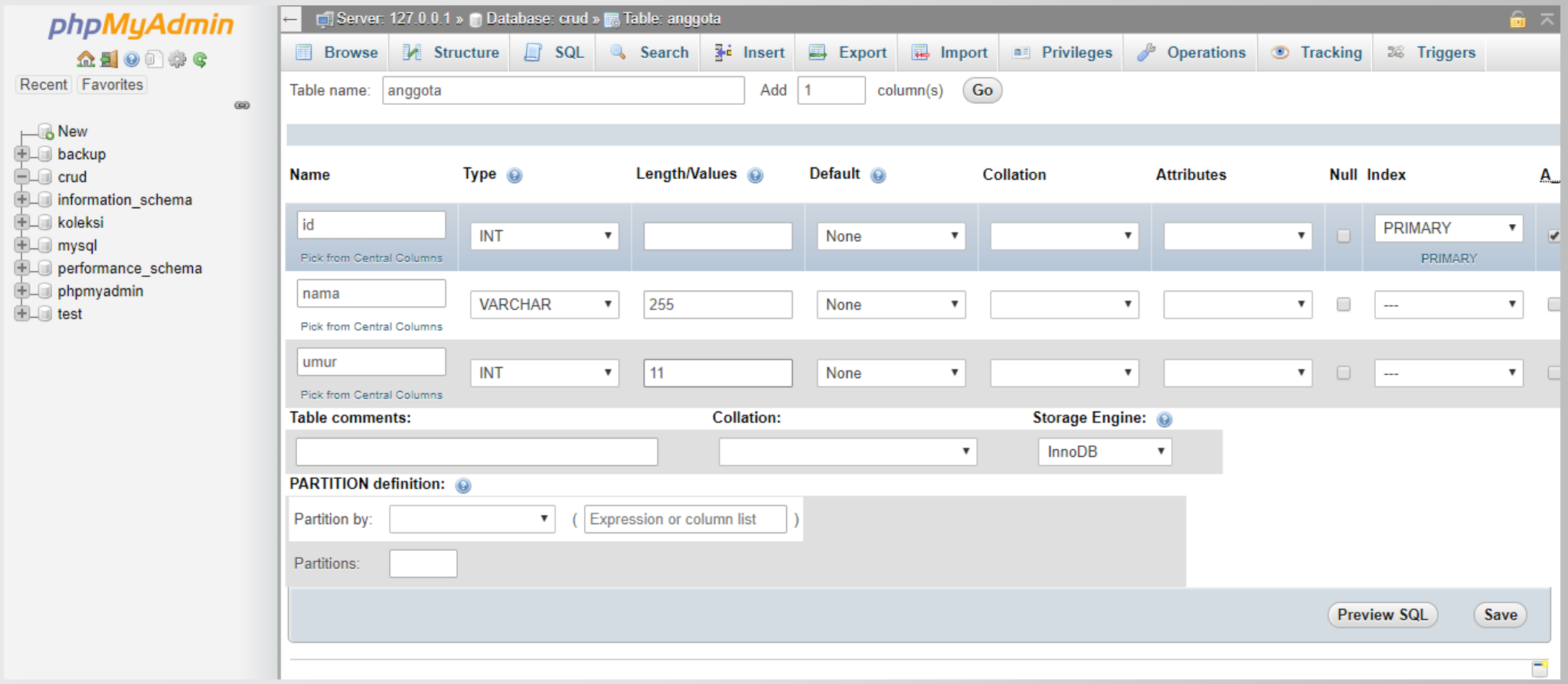


MEMBUAT DATABASE

- Buat database crud
- Buat Tabel Anggota dengan ketentuan
  - Id = int, autoincrement, primarykey
  - Nama = varchar(255)
  - Umur = int(5)

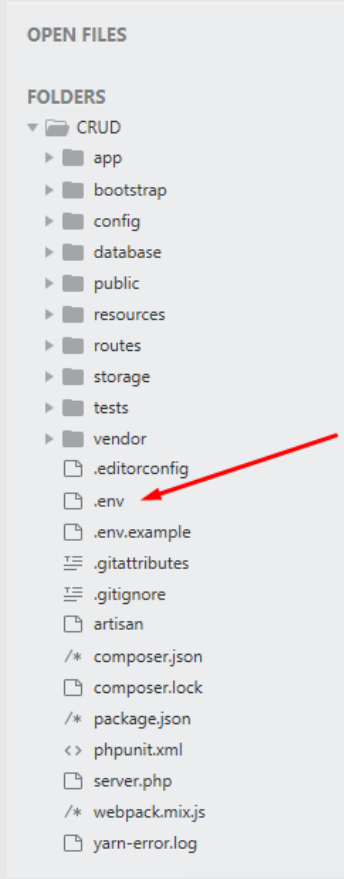


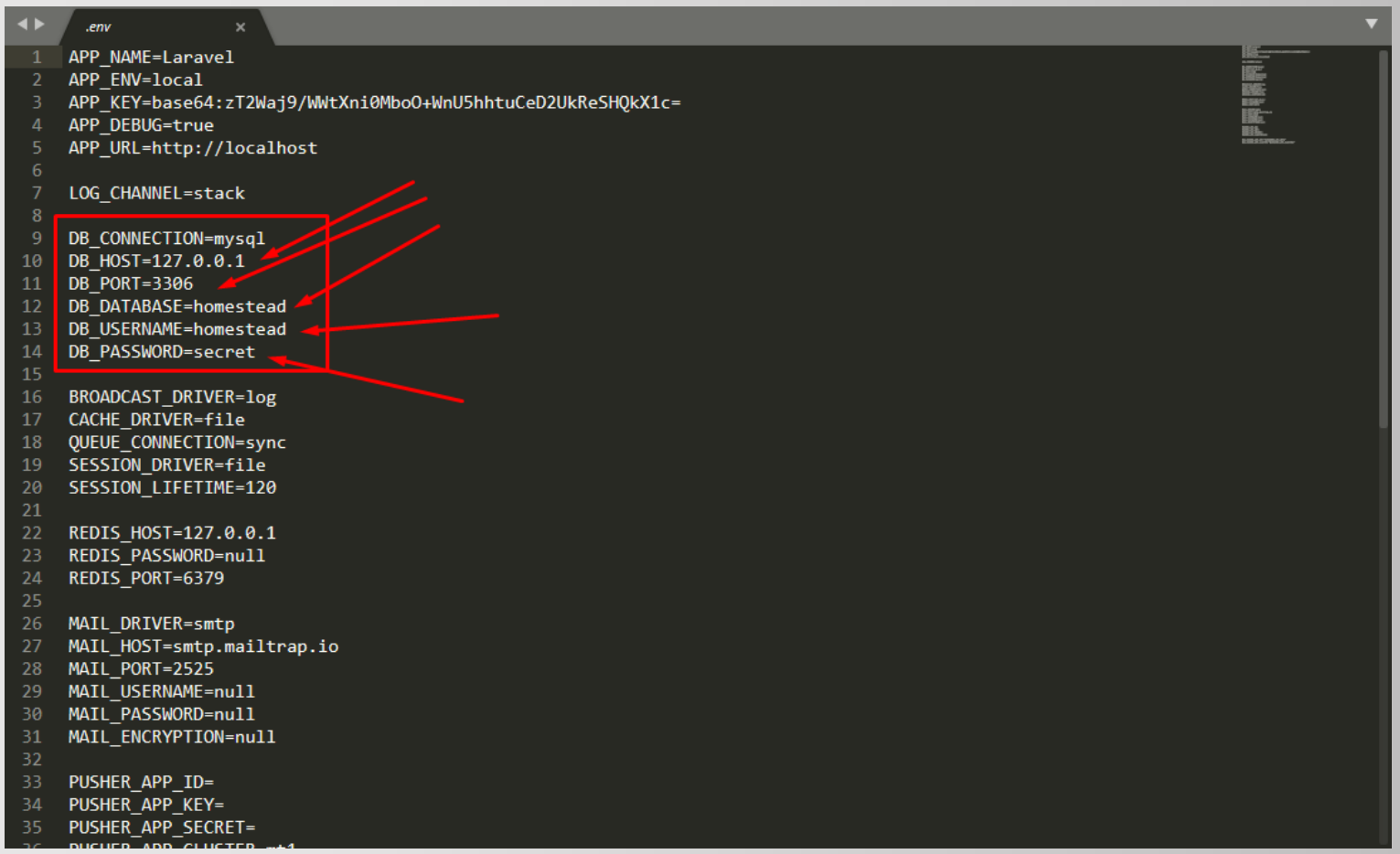
MEMBUAT DATABASE



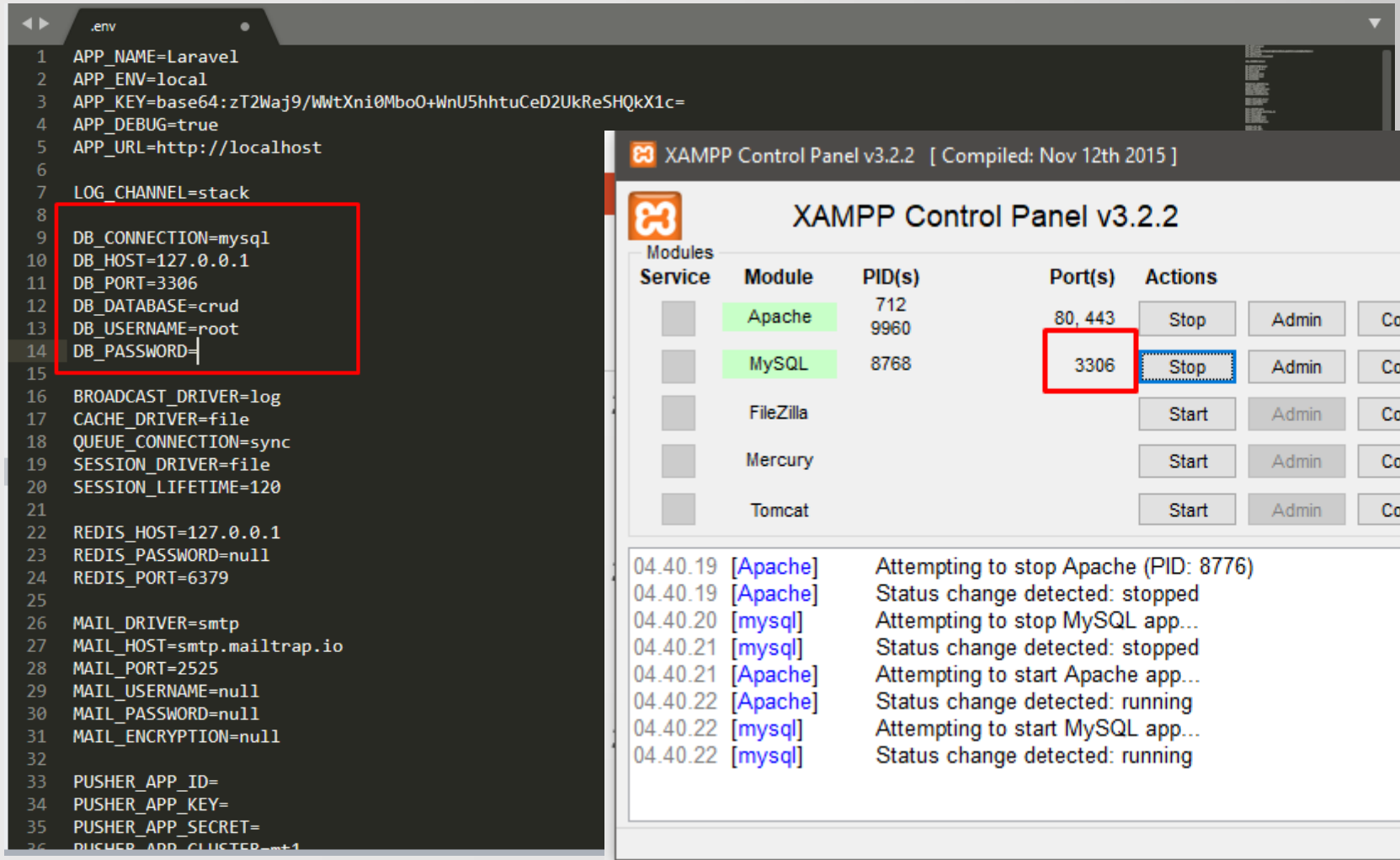
KONEKSI DATABASE

Buka file .env



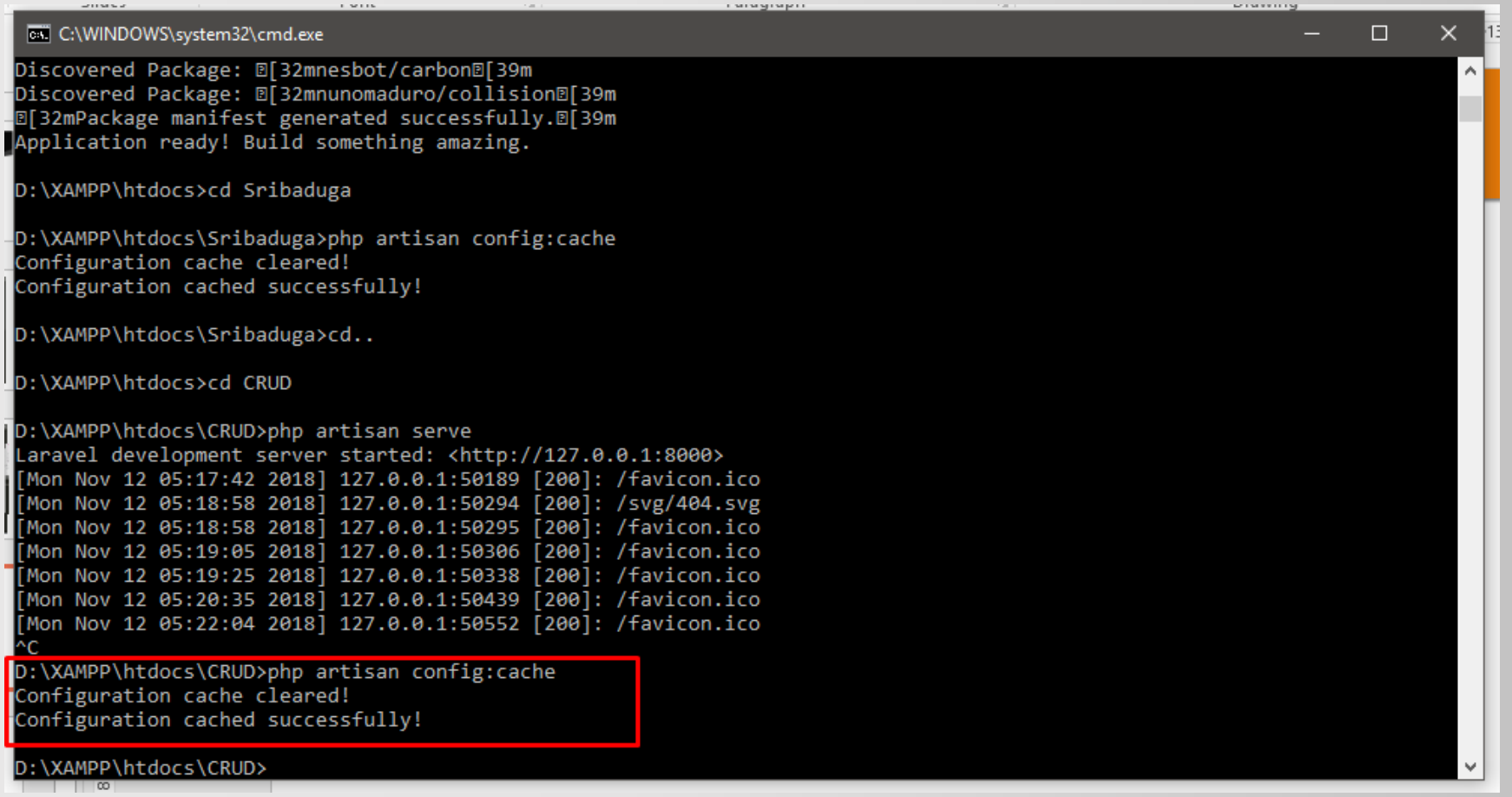


KONEKSI DATABASE



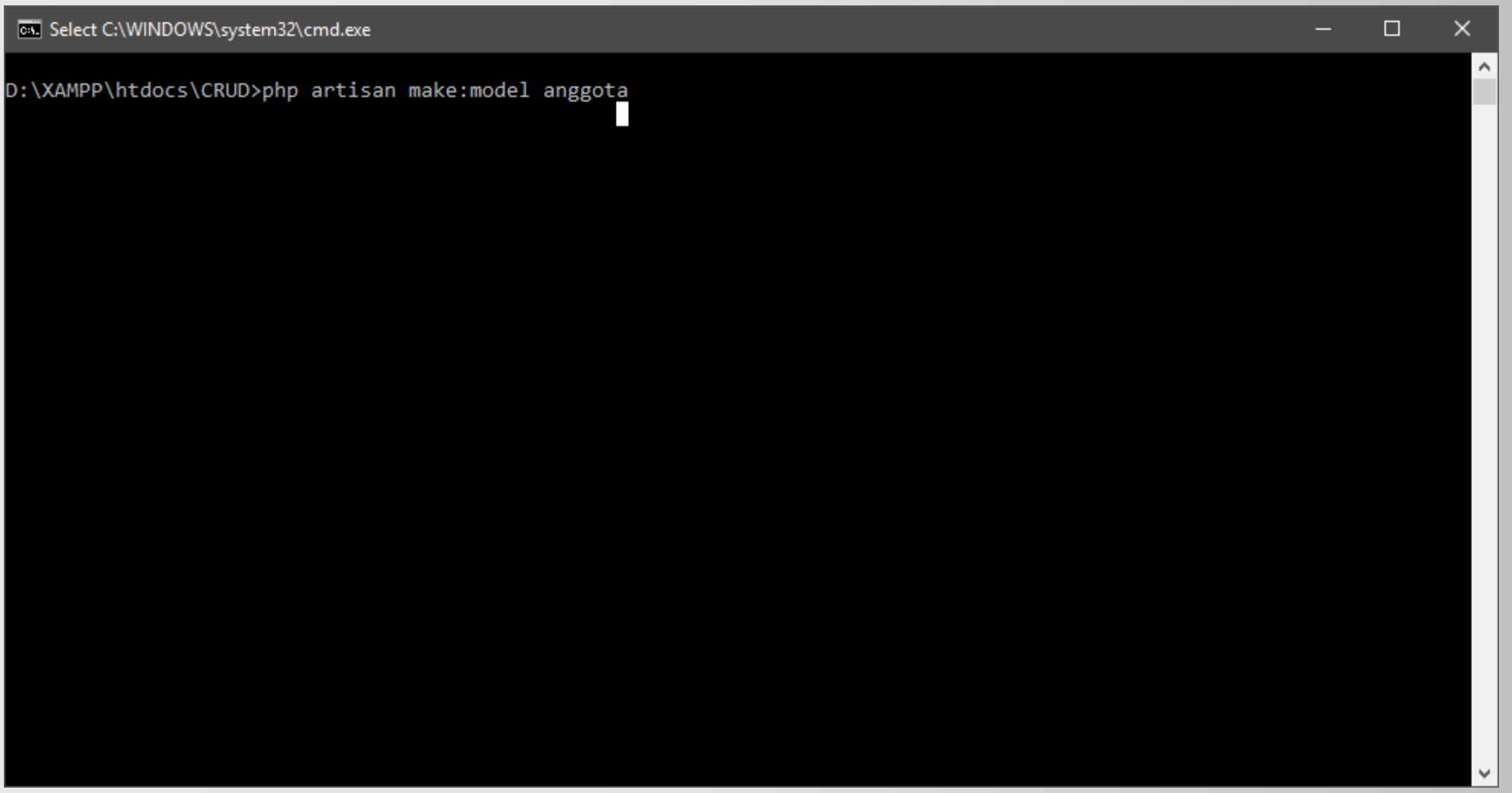
KONEKSI DATABASE

- Stop PHP ARTISAN SERVE
  - Ketik php artisan config:cache

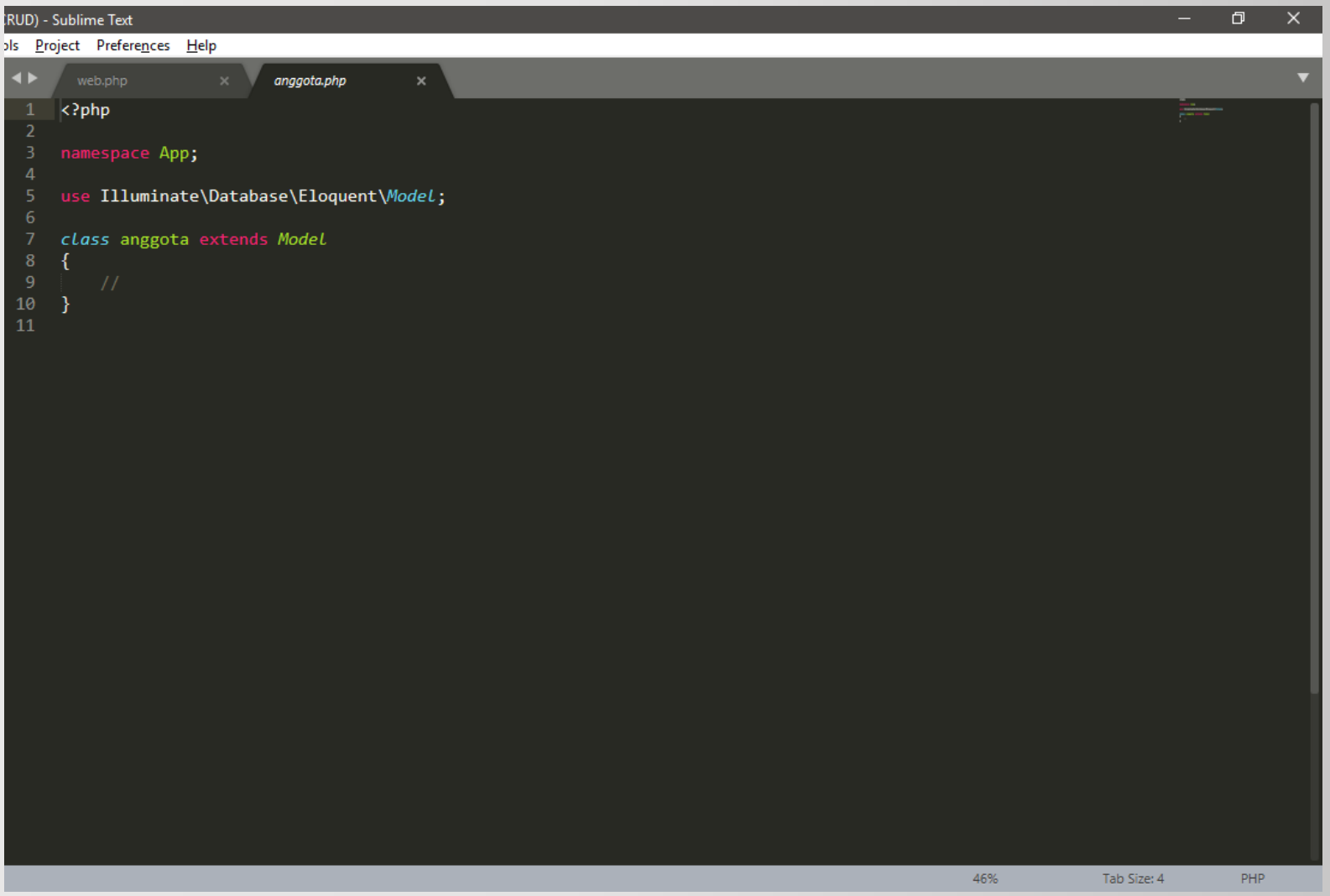


MEMBUAT MODEL

php artisan make:model anggota



MEMBUAT MODEL

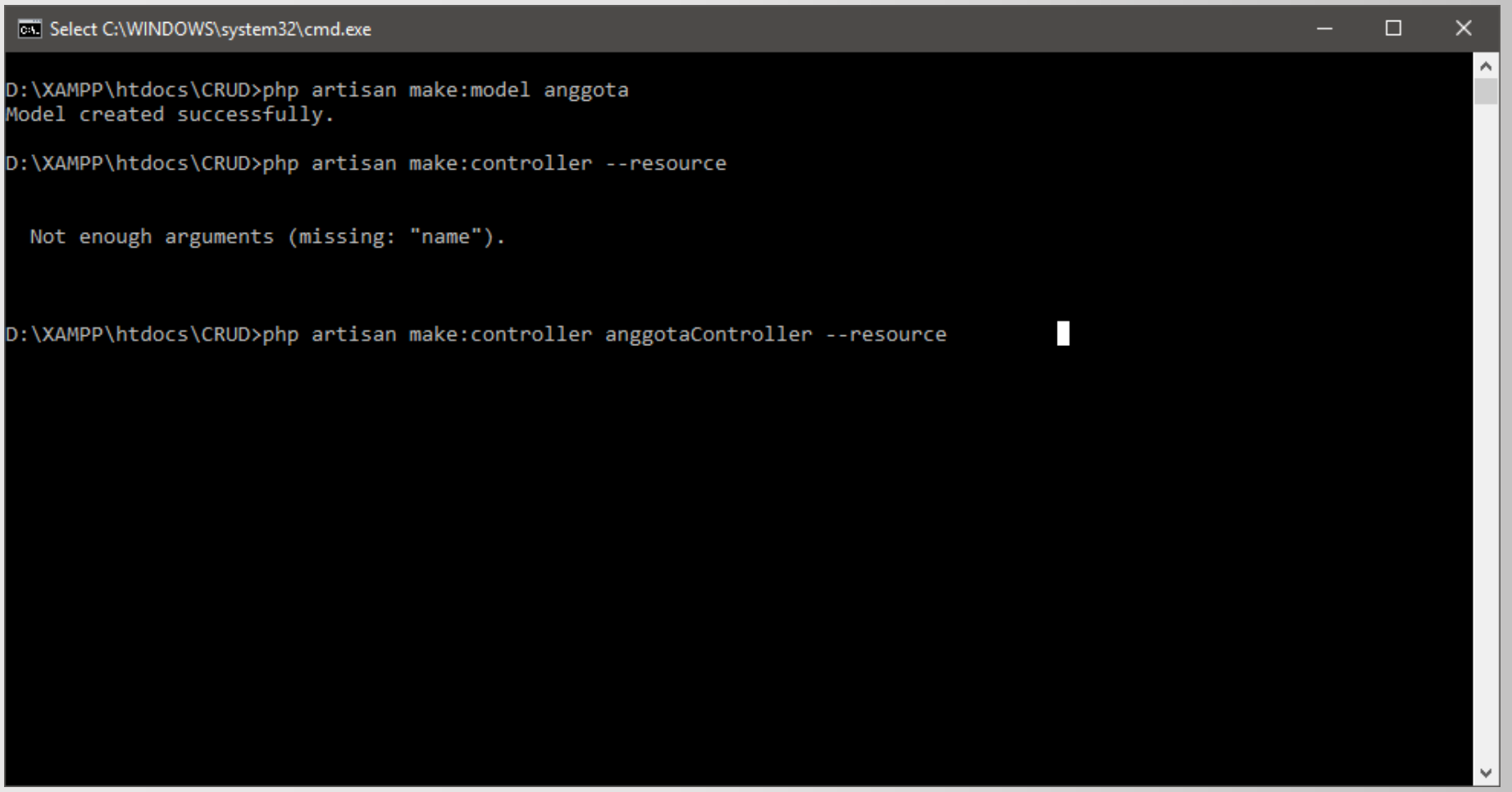


MEMBUAT MODEL

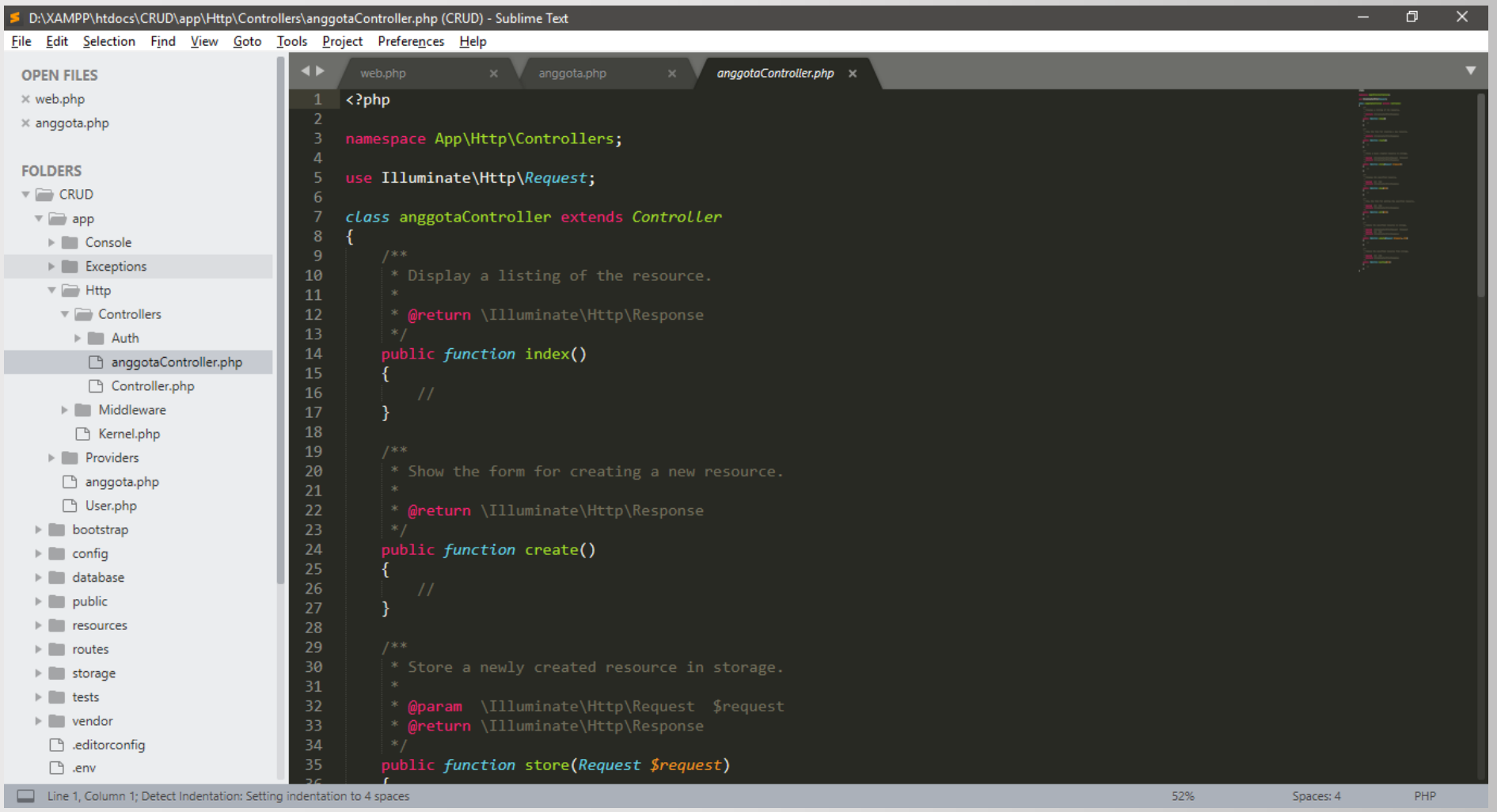
```
protected $table = ‘anggota’;
protected $primaryKey = ‘id’;
protected $fillable = [‘nama’,‘umur’];
public $timestamps = false;
```

MEMBUAT CONTROLLER

php artisan make:controller anggotaController --resource



MEMBUAT CONTROLLER



KONFIGURASI ROUTE

web.php | anggota.php | anggotaController.php

```php
<?php

/*
|--------------------------------------------------------------------------
| Web Routes
|--------------------------------------------------------------------------
|
| Here is where you can register web routes for your application. These
| routes are loaded by the RouteServiceProvider within a group which
| contains the "web" middleware group. Now create something great!
|
*/

Route::get('/', function () {
    return view('welcome');
});


Route::get('/hellow', function () {
    echo "Hai";
});

Route::resource('anggota', 'anggotaController');

```

MEMBUAT VIEW

1.Master.blade.php

2.create.blade.php

3.edit.blade.php

4.index.blade.php

MEMBUAT VIEW

Master.blade.php

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4    <title>Laravel 5.7 CRUD </title>
5    <link href="{{ asset('css/app.css') }}" rel="stylesheet" type="text/css" />
6  </head>
7  <body>
8    <div class="container">
9      @yield('content')
10   </div>
11   <script src="{{ asset('js/app.js') }}" type="text/js"></script>
12 </body>
13 </html>
```

MEMBUAT VIEW

create.blade.php

```
@extends('layouts.master')

@section('content')
<style>
  .uper {
    margin-top: 40px;
  }
</style>
<div class="card uper">
  <div class="card-header">
    Tambah Anggota
  </div>
  <div class="card-body">
    @if ($errors->any())
      <div class="alert alert-danger">
        <ul>
            @foreach ($errors->all() as $error)
              <li>{{ $error }}</li>
            @endforeach
        </ul>
      </div><br />
    @endif
      <form method="post" action="{{ route('anggota.store') }}">
          <div class="form-group">
              @csrf
              <label>Nama:</label>
              <input type="text" class="form-control" name="nama"/>
          </div>
          <div class="form-group">
              <label>Umur :</label>
              <input type="number" class="form-control" name="umur"/>
          </div>
          <button type="submit" class="btn btn-primary">Add</button>
      </form>
  </div>
</div>
@endsection
```

MEMBUAT VIEW

edit.blade.php

```
@extends('layouts.master')

@section('content')
<style>
  .uper {
    margin-top: 40px;
  }
</style>
<div class="card uper">
  <div class="card-header">
    Edit Anggota
  </div>
  <div class="card-body">
    @if ($errors->any())
      <div class="alert alert-danger">
        <ul>
            @foreach ($errors->all() as $error)
              <li>{{ $error }}</li>
            @endforeach
        </ul>
      </div><br />
    @endif
      <form method="post" action="{{ route('anggota.update', $anggota->id) }}">
        @method('PATCH')
        @csrf
        <div class="form-group">
          <label >Nama:</label>
          <input type="text" class="form-control" name="nama" value={{ $anggota->nama }} />
        </div>
        <div class="form-group">
          <label >Umur :</label>
          <input type="number" class="form-control" name="umur" value={{ $anggota->umur }} />
        </div>
        <button type="submit" class="btn btn-primary">Update</button>
      </form>
  </div>
</div>
@endsection
```

MEMBUAT VIEW

index.blade.php

```
@extends('layouts.master')

@section('content')
<style>
  .uper {
    margin-top: 40px;
  }
</style>
<div class="uper">
  @if(session()->get('success'))
    <div class="alert alert-success">
      {{ session()->get('success') }}
    </div><br />
  @endif
  <a href="{{route('anggota.create')}}" class="btn btn-primary">Tambah</a>
  <table class="table table-striped">
    <thead>
        <tr>
          <td>ID</td>
          <td>Nama</td>
          <td>Umur</td>
          <td colspan="2">Action</td>
        </tr>
    </thead>
    <tbody>
        @foreach($anggota as $aggo)
        <tr>
            <td>{{$aggo->id}}</td>
            <td>{{$aggo->nama}}</td>
            <td>{{$aggo->umur}}</td>
            <td><a href="{{ route('anggota.edit',$aggo->id)}}" class="btn btn-primary">Edit</a></td>
            <td>
                <form action="{{ route('anggota.destroy', $aggo->id)}}" method="post">
                  @csrf
                  @method('DELETE')
                  <button class="btn btn-danger" type="submit">Delete</button>
                </form>
            </td>
        </tr>
        @endforeach
    </tbody>
  </table>
<div>
@endsection
```

KONFIGURASI KEMBALI CONTROLLER

anggotaController

```php
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\anggota;


class anggotaController extends Controller
{
    /**
```

index

```php
    public function index()
    {
        $data = anggota::all();
        return view('index',['anggota'=> $data]);
    }

    /**
```

KONFIGURASI KEMBALI CONTROLLER

create

```php
public function create()
{
    return view('create');
}
```

store

```php
public function store(Request $request)
{
    $request->validate([
        'nama'=>'required',
        'umur'=> 'required|integer',
    ]);
    $anggota = new anggota([
        'nama' => $request->get('nama'),
        'umur'=> $request->get('umur'),
    ]);
    $anggota->save();
    return redirect('/anggota')->with('success', 'Anggota berhasil di tambahkan');
}
```

KONFIGURASI KEMBALI CONTROLLER

edit

```php
public function edit($id)
{
    $anggota = anggota::find($id);
    return view('edit', compact('anggota'));
}
```

update

```php
public function update(Request $request, $id)
{
   $request->validate([
    'nama'=>'required',
    'umur'=> 'required|integer',
]);

   $anggota = anggota::find($id);
   $anggota->nama = $request->get('nama');
   $anggota->umur = $request->get('umur');
   $anggota->save();

   return redirect('/anggota')->with('success', 'Anggota sudah berhasil di update');
}
```

KONFIGURASI KEMBALI CONTROLLER

destroy

```php
public function destroy($id)
{
    $anggota = anggota::find($id);
    $anggota->delete();

    return redirect('/anggota')->with('success', 'Anggota telah berhasil di hapus');
}
```

https://github.com/euclife/FILE-soft-Crud

Any Question ? . . .

Gracias . . .

CONTACT US AT:

@Chandra_ramdhan

160613024chandra@gmail.com

0838-2189-6126