

Nama : Adib Zilham Fadillah

Nim : 200401069

Matkul : Pemograman Framework ( Laravel )

1. Langkah install Laravel - Buka CMD ( Windows + R ) lalu masuk ke directori \xampp\htdocs -  
Lalu masukkan perintah : `composer create-project --prefer-dist laravel/laravel nama_project`  
( 200401069\_laravel )

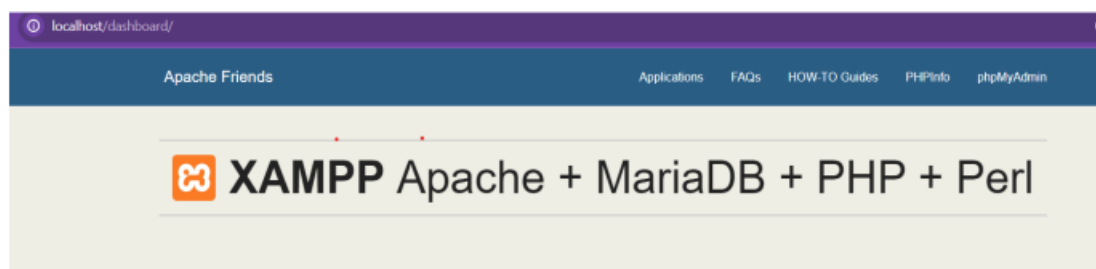
```
C:\xampp\htdocs>composer create-project --prefer-dist laravel/laravel 200401069_laravel
Creating a "laravel/laravel" project at "./200401069_laravel"
Installing laravel/laravel (v10.2.9)
- Installing laravel/laravel (v10.2.9): Extracting archive
Created project in C:\xampp\htdocs\200401069_laravel
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
Loading composer repositories with package information
Updating dependencies
Lock file operations: 110 installs, 0 updates, 0 removals
```

2. Langkah membuat database untuk aplikasi laravel

- Buka XAMPP, lalu start Apache dan MySQL



- Lalu buka chrome dan ketik localhost/



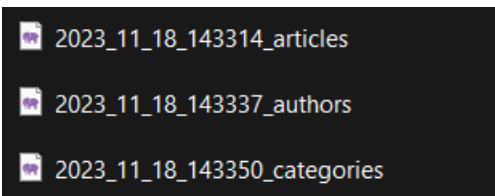
- Klik phpmyadmin, setelah itu buat database baru ( 200401069\_laravel )



3. Langkah membuat file migration untuk aplikasi laravel dengan 3 table ( articles, authors, dan categories )

- Buka CMD ( Windows + R ) lalu masuk ke directori \xampp\htdocs\200401069\_laravel
- Lalu ketik perintah :

```
C:\xampp\htdocs>cd \xampp\htdocs\200401069_laravel
C:\xampp\htdocs\200401069_laravel>php artisan make:migration articles
INFO Migration [C:\xampp\htdocs\200401069_laravel\database\migrations\2023_11_18_143314_articles.php] created successfully.
C:\xampp\htdocs\200401069_laravel>php artisan make:migration authors
INFO Migration [C:\xampp\htdocs\200401069_laravel\database\migrations\2023_11_18_143337_authors.php] created successfully.
C:\xampp\htdocs\200401069_laravel>php artisan make:migration categories
INFO Migration [C:\xampp\htdocs\200401069_laravel\database\migrations\2023_11_18_143350_categories.php] created successfully.
C:\xampp\htdocs\200401069_laravel>
```



#### 4. Membuat fungsi CRUD dengan konsep MVC pada laravel

- Pertama hubungkan database dengan project, dengan cara buka file .env pada project lalu pada bagian DB\_DATABASE ubah sesuai nama database yang akan dihubungkan

```
> bootstrap 10
> config 11 DB_CONNECTION=mysql
> database 12 DB_HOST=127.0.0.1
> public 13 DB_PORT=3306
> resources 14 DB_DATABASE=200401069_laravel
> routes 15 DB_USERNAME=root
> storage 16 DB_PASSWORD=
> tests 17
> vendor 18 BROADCAST_DRIVER=log
> .editorconfig 19 CACHE_DRIVER=file
> .env 20 FILESYSTEM_DISK=local
> .env.example 21 QUEUE_CONNECTION=sync
> .gitattributes 22 SESSION_DRIVER=file
23 SESSION_LIFETIME=120
24
```

- Buat model article dan categories ( sesuai ketentuan untuk fungsi CRUD ) dengan perintah :

```
PS C:\xampp\htdocs\200401069_laravel> php artisan make:model articles -m
INFO Model [C:\xampp\htdocs\200401069_laravel\app\Models\articles.php] created successfully.
INFO Migration [C:\xampp\htdocs\200401069_laravel\database\migrations\2023_11_18_144719_create_articles_table.php] created successfully.

PS C:\xampp\htdocs\200401069_laravel> php artisan make:model categories -m
INFO Model [C:\xampp\htdocs\200401069_laravel\app\Models\categories.php] created successfully.
INFO Migration [C:\xampp\htdocs\200401069_laravel\database\migrations\2023_11_18_144733_create_categories_table.php] created successfully.
```

- Buat Controller dengan perintah :

```
PS C:\xampp\htdocs\200401069_laravel> php artisan make:controller articleController -m
INFO Controller [C:\xampp\htdocs\200401069_laravel\app\Http\Controllers\articleController.php] created successfully.

PS C:\xampp\htdocs\200401069_laravel> php artisan make:controller CategoriesController -m
INFO Controller [C:\xampp\htdocs\200401069_laravel\app\Http\Controllers\CategoriesController.php] created successfully.
```

- Definisikan rute untuk articles dan categories dalam file `route/web.php`

```
200401069_laravel > routes > web.php
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4
5 use App\Http\Controllers\articleController;
6 use App\Http\Controllers\categoriesController;
7
8 Route::resource('Article', articleController::class);
9 Route::resource('Categories', categoriesController::class);
10
11
12 Route::get('/', function () {
13     return view('welcome');
14 });
15
```

- Implementasi Metode CRUD di controller pada file articleController.php dan categoriesController.php

```
200401069_laravel > app > Http > Controllers > articleController.php
3  namespace App\Http\Controllers;
4
5  use App\Models\article;
6  use Illuminate\Http\Request;
7
8  class articleController extends Controller
9  {
10     // articleController.php
11     public function index()
12     {
13         $articles = Article::all();
14         return view('articles.index', compact('articles'));
15     }
16
17     public function create()
18     {
19         return view('articles.create');
20     }
21
22     public function store(Request $request)
23     {
24         // Validasi data
25         // Simpan data ke database
26     }
27
28     public function edit($id)
29     {
30         $article = Article::find($id);
31         return view('articles.edit', compact('article'));
32     }
33
34     public function update(Request $request, $id)
35     {
36         // Validasi dan perbarui data di database
37     }
38 }
```

200401069\_laravel > app > Http > Controllers > 🐞 CategoriesController.php

```
3 namespace App\Http\Controllers;
4
5 use App\Models\categories;
6 use Illuminate\Http\Request;
7
8 class categoriesController extends Controller
9 {
10 // categoriesController.php
11 public function index()
12 {
13     $categories = Category::all();
14     return view('categories.index', compact('categories'));
15 }
16
17 public function create()
18 {
19     return view('categories.create');
20 }
21
22 public function store(Request $request)
23 {
24     // Validasi data
25     // Simpan data ke database
26 }
27
28 public function edit($id)
29 {
30     $category = Category::find($id);
31     return view('categories.edit', compact('category'));
32 }
33
34 public function update(Request $request, $id)
35 {
36     // Validasi dan perbarui data di database
37 }
38
39 public function destroy($id)
```