

Práctica 1: Instalación de Laravel y Creación de una Tabla de Libros

Clara de la Cruz Rico

1. Instalacion de Laravel.

Para instalar laravel vamos a ir a la web <https://getcomposer.org/download/> y nos descargamos el ejecutable.

Realizamos todos los pasos siguientes y con **composer -v** podemos ver la version y que se haya instalado correctamente

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6

PS C:\Users\Propietario> composer -v

Composer version 2.7.1 2024-02-09 15:26:28

Usage:
  command [options] [arguments]

Options:
  -h, --help                Display help for the given command. When no command is given display help for the list
command
  -q, --quiet              Do not output any message
  -V, --version            Display this application version
                        --ansi|--no-ansi Force (or disable --no-ansi) 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
  -vv|vv|vvv, --verbose   Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and
3 for debug

Available commands:
  about                Shows a short information about Composer
  archive              Creates an archive of this composer package
  audit                Checks for security vulnerability advisories for installed packages
  browse              [home] Opens the package's repository URL or homepage in your browser
  bump                Increases the lower limit of your composer.json requirements to the currently installed versions
  check-platform-reqs Check that platform requirements are satisfied
  clear-cache          [clearcache|cc] Clears composer's internal package cache
  completion           Dump the shell completion script
  config              Sets config options
  create-project       Creates new project from a package into given directory
  depends              [why] Shows which packages cause the given package to be installed
  diagnose             Diagnoses the system to identify common errors
  dump-autoload        [dumpautoload] Dumps the autoloader
  exec                 Executes a vendored binary/script
  fund                Discover how to help fund the maintenance of your dependencies
  global              Allows running commands in the global composer dir ($COMPOSER_HOME)
  help                Display help for a command
```

En mi caso tuve que ejecutar el siguiente comando para poder realizar la creacion del proyecto con laravel **composer update --prefer-source**

Y luego vamos a realizar la creacion del proyecto con el siguiente comando **composer create-project laravel/laravel libreria** este ultimo nos instalará laravel con todo lo necesario

2. Configuración de la base de datos

En nuestro archivo .env vamos a modificar los campos señalados en rosa

```
.env
1 APP_NAME=libreria
2 APP_ENV=local
3 APP_KEY=base64:r0boA5EYwUrvenuP4mFWcQf0TaXwnSo6Ec6eJCM0jPU=
4 APP_DEBUG=true
5 APP_URL=http://localhost
6
7 LOG_CHANNEL=stack
8 LOG_DEPRECATIONS_CHANNEL=null
9 LOG_LEVEL=debug
10
11 DB_CONNECTION=mysql
12 DB_HOST=127.0.0.1
13 DB_PORT=3306
14 DB_DATABASE=libreria
15 DB_USERNAME=root
16 DB_PASSWORD=
17
18 BROADCAST_DRIVER=log
19 CACHE_DRIVER=file
20 FILESYSTEM_DISK=local
21 QUEUE_CONNECTION=sync
22 SESSION_DRIVER=file
23 SESSION_LIFETIME=120
24
25 MEMCACHED_HOST=127.0.0.1
26
27 REDIS_HOST=127.0.0.1
28 REDIS_PASSWORD=null
29 REDIS_PORT=6379
30
31 MAIL_MAILER=smtp
32 MAIL_HOST=mailpit
33 MAIL_PORT=1025
34 MAIL_USERNAME=null
35 MAIL_PASSWORD=null
36 MAIL_ENCRYPTION=null
37 MAIL_FROM_ADDRESS="hello@example.com"
38 MAIL_FROM_NAME="${APP_NAME}"
39
40 AWS_ACCESS_KEY_ID=
41 AWS_SECRET_ACCESS_KEY=
42 AWS_DEFAULT_REGION=us-east-1
43 AWS_BUCKET=
44 AWS_USE_PATH_STYLE_ENDPOINT=false
45
46 PUSHER_APP_ID=
47 PUSHER_APP_KEY=
48 PUSHER_APP_SECRET=
49 PUSHER_HOST=
50 PUSHER_PORT=443
51 PUSHER_SCHEME=https
52 PUSHER_APP_CLUSTER=mt1
53
54 VITE_APP_NAME="${APP_NAME}"
55 VITE_PUSHER_APP_KEY="${PUSHER_APP_KEY}"
56 VITE_PUSHER_HOST="${PUSHER_HOST}"
57 VITE_PUSHER_PORT="${PUSHER_PORT}"
58 VITE_PUSHER_SCHEME="${PUSHER_SCHEME}"
59 VITE_PUSHER_APP_CLUSTER="${PUSHER_APP_CLUSTER}"
60
```

Los amarillos son optativos, para esta tarea no se va a modificar pero si alguien quisiera si sería posible.

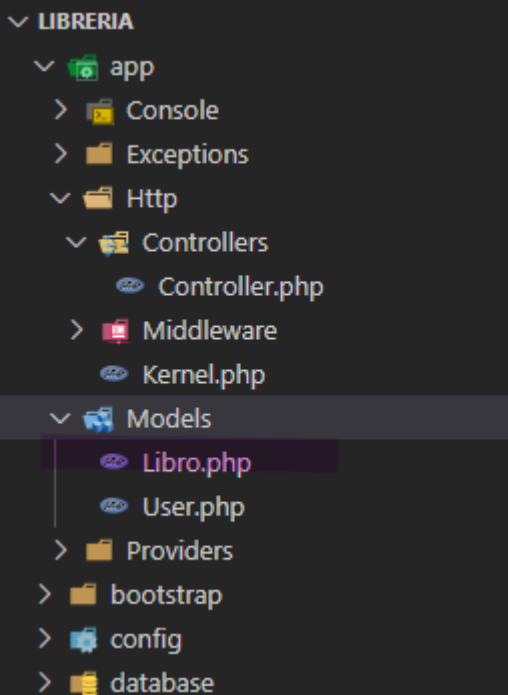
3. Creacion de la migracion de libros.

Para ello comprobamos que tenemos php y ejecutamos el comando para la creacion del modelo y la migracion de Libro

```
2024-03-10 14:50:41 ..... ~ 25
PS E:\REPOSITORIO\laravel\libreria> php -v
● PHP 8.2.12 (cli) (built: Oct 24 2023 21:15:15) (ZTS Visual C++ 2019 x64)
Copyright (c) The PHP Group
Zend Engine v4.2.12, Copyright (c) Zend Technologies
PS E:\REPOSITORIO\laravel\libreria> php artisan make:model Libro -m
●
[INFO] Model [E:\REPOSITORIO\laravel\libreria\app\Models\Libro.php] created successfully.

[INFO] Migration [E:\REPOSITORIO\laravel\libreria\database\migrations/2024_03_10_141922_create_libros_table.php] created successfully.

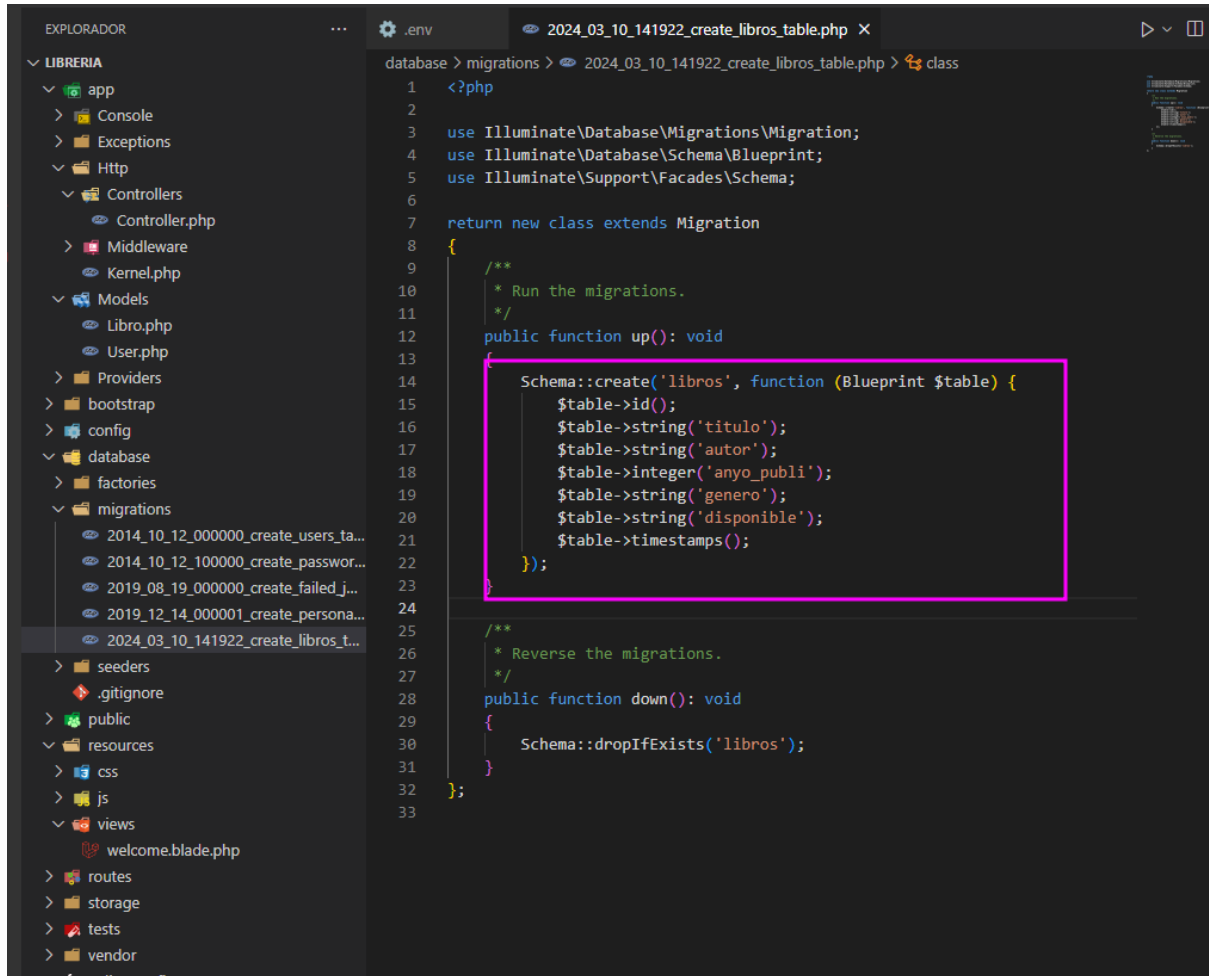
PS E:\REPOSITORIO\laravel\libreria> 
```



- LIBRERIA
 - app
 - Console
 - Exceptions
 - Http
 - Controllers
 - Controller.php
 - Middleware
 - Kernel.php
 - Models
 - Libro.php
 - User.php
 - Providers
 - bootstrap
 - config
 - database

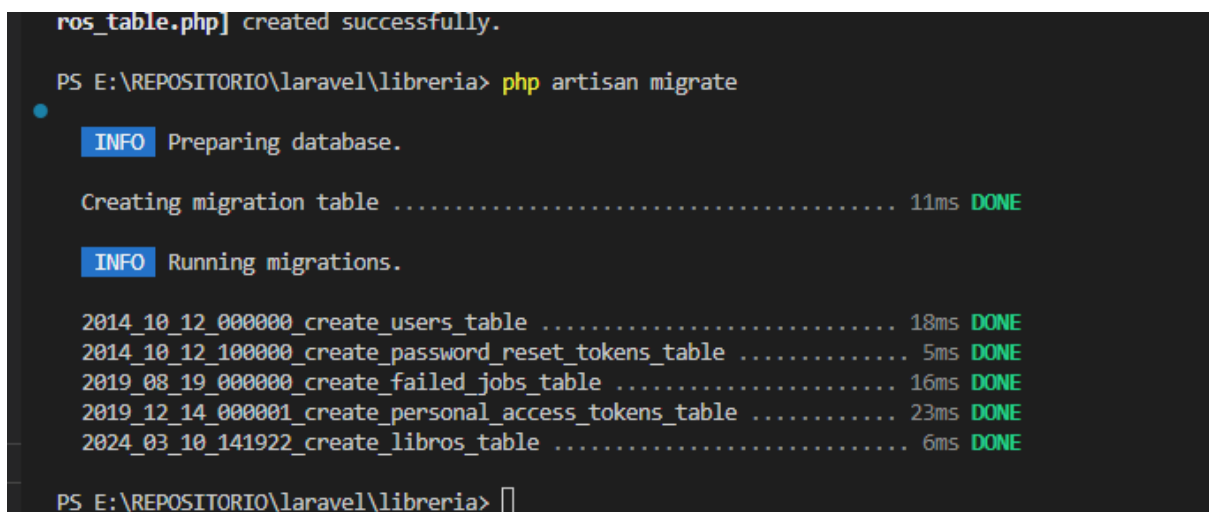
4. Edita la migración

Añadimos las columnas a la tabla



```
1 <?php
2
3 use Illuminate\Database\Migrations\Migration;
4 use Illuminate\Database\Schema\Blueprint;
5 use Illuminate\Support\Facades\Schema;
6
7 return new class extends Migration
8 {
9     /**
10      * Run the migrations.
11      */
12     public function up(): void
13     {
14         Schema::create('libros', function (Blueprint $table) {
15             $table->id();
16             $table->string('titulo');
17             $table->string('autor');
18             $table->integer('anyo_publicacion');
19             $table->string('genero');
20             $table->boolean('disponible');
21             $table->timestamps();
22         });
23     }
24
25     /**
26      * Reverse the migrations.
27      */
28     public function down(): void
29     {
30         Schema::dropIfExists('libros');
31     }
32 };
33
```

Y lanzamos el siguiente comando para ejecutarlo



```
ros_table.php] created successfully.

PS E:\REPOSITORIO\laravel\libreria> php artisan migrate

  INFO  Preparing database.

Creating migration table ..... 11ms DONE

  INFO  Running migrations.

2014_10_12_000000_create_users_table ..... 18ms DONE
2014_10_12_100000_create_password_reset_tokens_table ..... 5ms DONE
2019_08_19_000000_create_failed_jobs_table ..... 16ms DONE
2019_12_14_000001_create_personal_access_tokens_table ..... 23ms DONE
2024_03_10_141922_create_libros_table ..... 6ms DONE

PS E:\REPOSITORIO\laravel\libreria>
```

Si entramos en phpmyadmin podemos ver que efectivamente ha salido todo correctamente

phpMyAdmin

Reciente

Favoritas

Nueva

information_schema

libreria

Nueva

failed_jobs

libros

migrations

password_reset_tokens

personal_access_tokens

users

mysql

performance_schema

phpmyadmin

test

Servidor: 127.0.0.1 » Base de datos: libreria » Tabla: libros

Examinar

Estructura

SQL

Buscar

Insertar

Exportar

Importar

Privilegios

Operaciones

Seguimiento

Disparadores

MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

SELECT * FROM `libros`

Perfilando

Editar en línea

Editar

Explicar SQL

Crear código PHP

Actualizar

id titulo autor anyo_publico genero disponible created_at updated_at

Operaciones sobre los resultados de la consulta

Crear vista

Guardar esta consulta en favoritos

Etiqueta: ☐ Permitir que todo usuario pueda acceder a este favorito

Guardar esta consulta en favoritos

5. Creacion migracion Prestamos

Como hicimos anteriormente vamos a crear el modelo y la migracion de Prestamos

```
2024-03-10 14:19:22 Create libros_table ..... 6ms DONE

PS E:\REPOSITORIO\laravel\libreria> php artisan make:model Prestamos -m

INFO Model [E:\REPOSITORIO\laravel\libreria\app\Models\Prestamos.php] created successfully.

INFO Migration [E:\REPOSITORIO\laravel\libreria\database\Migrations\2024_03_10_144309_create_pre
stamos_table.php] created successfully.

PS E:\REPOSITORIO\laravel\libreria>
```

Realizamos la creacion

```
2024-03-10 17:10:26 ..... ~ 15
2024-03-10 17:10:26 /favicon.ico ..... ~ 15
PS E:\REPOSITORIO\laravel\libreria> php artisan migrate

INFO Running migrations.

2024_03_10_144309_create_prestamos_table ..... 74ms DONE

PS E:\REPOSITORIO\laravel\libreria>
```

Y comprobamos que esté correctamente:

