

Universitat
Oberta
de Catalunya

M4.255 - Desarrollo back-end con PHP

**PEC3. Backend de un portal de noticias y un API de
consulta básico con Laravel**

Aníbal Santos Gómez

ÍNDICE: DOCUMENTACIÓN.

1. **Instalación de Laravel en servidor local.**
2. **Creación de migraciones y modelos. Generación de Factory Fakes para la base de datos.**
3. **Creación de API de consulta pública y comprobación en navegador.**
4. **Migración de la web y publicación.**
5. **Diferencias entre CodeIgniter y Laravel.**

1. Instalación de Laravel en servidor local.

En primer lugar necesitaremos instalar XAMPP y Composer. En nuestro caso utilizaremos XAMPP y Composer bajo Ubuntu 20.04.

- Instalación de XAMPP: para instalar XAMPP en Ubuntu, ejecutaremos el siguiente comando en nuestra terminal:

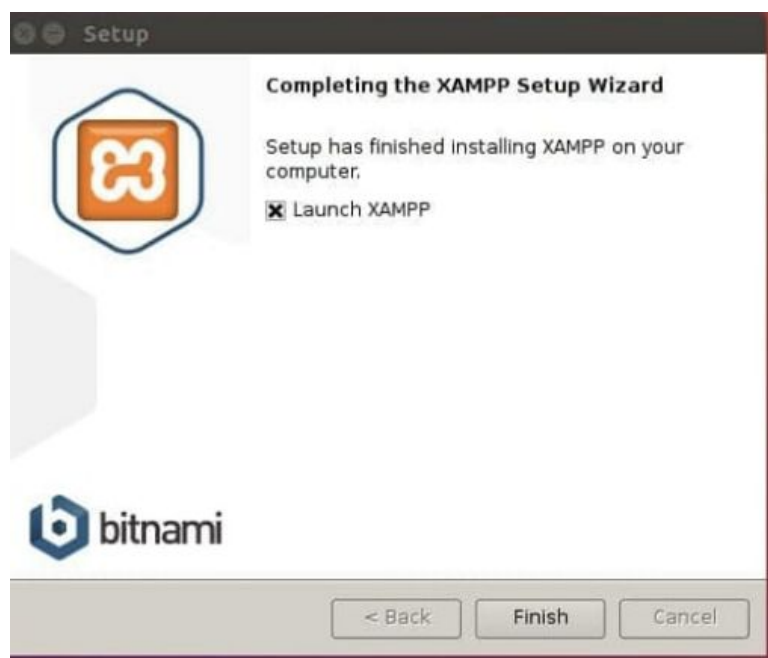
```
wget -r https://www.apachefriends.org/xampp-files/7.4.8/xampp-linux-x64-7.4.8-0-installer.run
```

Después tendremos que darle permisos e instalarlo:

```
chmod 755 xampp-linux-*-installer.run
```

```
sudo ./xampp-linux-*-installer.run
```

Nos aparecerá una interfaz de instalación donde ejecutaremos los pasos que nos indican.



Para arrancarlo simplemente ejecutaremos:

```
sudo /opt/lampp/lampp start
```

Y para detenerlo:

```
sudo /opt/lampp/lampp stop
```

- Instalación de Composer: a continuación instalaremos Composer de la siguiente manera:

Primero, actualizamos la caché del administrador de paquetes e instalamos los paquetes ejecutando lo siguiente:

```
sudo apt update  
sudo apt install php-cli unzip
```

Segundo, descargamos e instalamos Composer ejecutando lo siguiente:

```
cd ~  
curl -sS https://getcomposer.org/installer -o composer-setup.php
```

A continuación, verificaremos que el instalador descargado coincida con el hash SHA-384 para el instalador más reciente disponible en la página Composer Public Keys/Signatures:

```
HASH=`curl -sS https://composer.github.io/installer.sig`  
echo $HASH
```

Ahora, ejecutamos el siguiente código PHP, para verificar que la secuencia de comandos de instalación se pueda ejecutar de forma segura:

```
php -r "if (hash_file('SHA384', 'composer-setup.php') === '$HASH') { echo 'Installer verified'; } else { echo 'Installer corrupt'; unlink('composer-setup.php'); } echo PHP_EOL;"
```

Veremos el siguiente resultado: *"Installer verified"*

Para instalar Composer de manera global ejecutaremos lo siguiente:

```
sudo php composer-setup.php --install-dir=/usr/local/bin --filename=composer
```

Escribiendo composer en el terminal deberíamos ver las opciones que nos permite realizar en un explicativo:

```
Composer version 2.0.3 2020-10-28 15:50:55

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.
  -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
```

- Instalación de **Laravel**: para instalar Laravel mediante Composer, en primer lugar accederemos a la ruta de XAMPP:

Cd /opt/lampp/htdocs

Aquí crearemos un directorio con *mkdir nombre_carpeta*

Accederemos al directorio que acabamos de crear y posteriormente ejecutaremos el siguiente comando de Composer:

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

Este comando iniciará la instalación del Framework. Tardará unos cuantos minutos dependiendo de la conexión a Internet que tengamos. Si todo termina correctamente nos mostrará algo similar a lo siguiente:

```

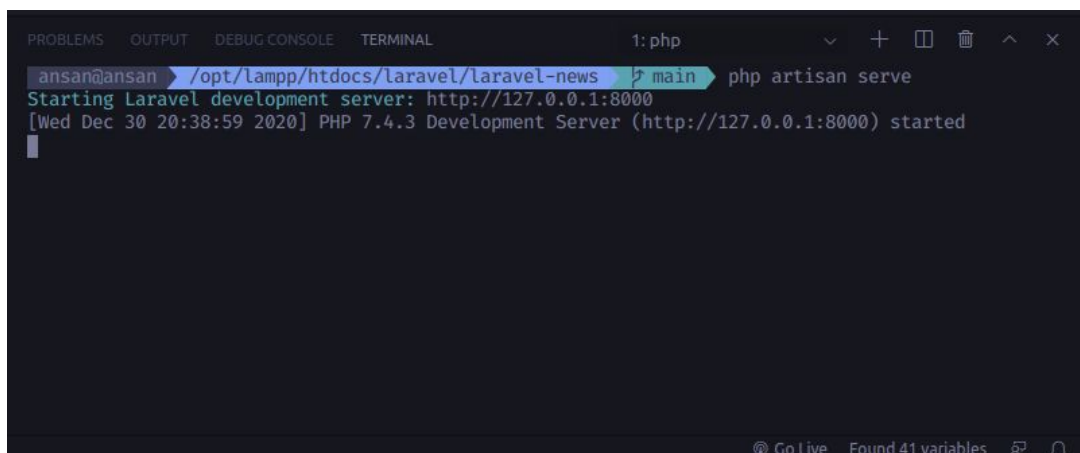
- Installing sebastian/comparator (4.0.6): Extracting archive
- Installing sebastian/code-unit (1.0.8): Extracting archive
- Installing sebastian/cli-parser (1.0.1): Extracting archive
- Installing phpunit/php-timer (5.0.3): Extracting archive
- Installing phpunit/php-text-template (2.0.4): Extracting archive
- Installing phpunit/php-invoker (3.1.1): Extracting archive
- Installing phpunit/php-file-iterator (3.0.5): Extracting archive
- Installing theseer/tokenizer (1.2.0): Extracting archive
- Installing sebastian/lines-of-code (1.0.3): Extracting archive
- Installing sebastian/complexity (2.0.2): Extracting archive
- Installing sebastian/code-unit-reverse-lookup (2.0.3): Extracting archive
- Installing phpunit/php-code-coverage (9.2.5): Extracting archive
- Installing doctrine/instantiator (1.4.0): Extracting archive
- Installing phpspec/prophecy (1.12.2): Extracting archive
- Installing phar-io/version (3.0.4): Extracting archive
- Installing phar-io/manifest (2.0.1): Extracting archive
- Installing myclabs/deep-copy (1.10.2): Extracting archive
- Installing phpunit/phpunit (9.5.0): Extracting archive
75 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
Discovered Package: facade/ignition
Discovered Package: fideloper/proxy
Discovered Package: fruitcake/laravel-cors
Discovered Package: laravel/sail
Discovered Package: laravel/tinker
Discovered Package: nesbot/carbon
Discovered Package: nunomaduro/collision
Package manifest generated successfully.
73 packages you are using are looking for funding.
Use the `composer fund` command to find out more!
> @php artisan key:generate --ansi
Application key set successfully.
ansan@ansan /opt/lampp/htdocs

```

Ahora, para iniciar el servicio de Laravel, desde el terminal ejecutaremos el siguiente comando:

php artisan serve

Y si el servicio inicia correctamente mostrará un mensaje como el de la captura siguiente con el servicio iniciado en el puerto por defecto 8000.

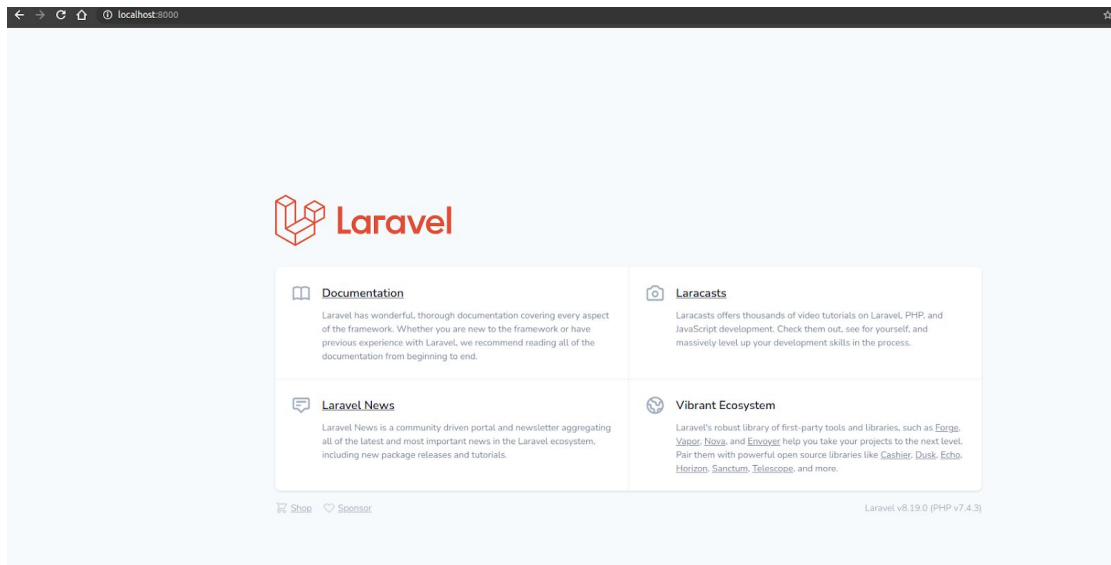


```

1: php
ansan@ansan /opt/lampp/htdocs/laravel/laravel-news main php artisan serve
Starting Laravel development server: http://127.0.0.1:8000
[Wed Dec 30 20:38:59 2020] PHP 7.4.3 Development Server (http://127.0.0.1:8000) started

```

Ahora solo deberemos ir a nuestro navegador para comprobar que carga correctamente.



El proceso de instalación de CodeIgniter y Laravel con Composer en XAMPP es bastante similar, simplemente tenemos que ejecutar el comando de instalación del gestor de paquetes y se descargará el framework elegido.

2. Creación de migraciones y modelos. Generación de Factory Fakes para la base de datos.

A continuación configuraremos la base de datos. Primero la crearemos en phpmyadmin y luego setearmos la configuración en el archivo .env de la siguiente manera:

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

Para crear las migraciones de la base de datos ejecutaremos los siguiente comandos:

```
php artisan make:migration create_authors_table
php artisan make:migration create_categories_table
php artisan make:migration create__table
```

Esto nos generará los archivos de migración para crear las tablas en la base de datos:

```
public function up()
{
    Schema::create('authors', function (Blueprint $table) {
        $table->id();
        $table->string('name');
        $table->timestamps();
    });
}

/**
 * Reverse the migrations.
 *
 * @return void
 */
public function down()
{
    Schema::dropIfExists('authors');
}
}
```



```

public function up()
{
    Schema::create('categories', function (Blueprint $table) {
        $table->id();
        $table->string('name');
        $table->timestamps();
    });
}

/**
 * Reverse the migrations.
 *
 * @return void
 */
public function down()
{
    Schema::dropIfExists('categories');
}
}

```

```

public function up()
{
    Schema::create('articles', function (Blueprint $table) {
        $table->id();
        $table->string('title');
        $table->unsignedBigInteger('author');
        $table->date('published');
        $table->longText('content');
        $table->string('image');
        $table->unsignedBigInteger('category');
        $table->timestamps();
        $table->foreign('author')->references('id')->on('authors')->onDelete('cascade');
        $table->foreign('category')->references('id')->on('categories')->onDelete('cascade');
    });
}

/**
 * Reverse the migrations.
 *
 * @return void
 */
public function down()
{
    Schema::dropIfExists('articles');
}
}

```

Ejecutaremos las migraciones con el siguiente comando:

`php artisan migrate`

Y esto nos generará las tres tablas con sus correspondientes campos:

Tabla de artículos:

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
<input type="checkbox"/> 1	id	bigint(20)		UNSIGNED	No	Ninguna		AUTO_INCREMENT	Cambiar Eliminar Más
<input type="checkbox"/> 2	title	varchar(255)	utf8mb4_unicode_ci		No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/> 3	author	bigint(20)		UNSIGNED	No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/> 4	published	date			No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/> 5	content	longtext	utf8mb4_unicode_ci		No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/> 6	image	varchar(255)	utf8mb4_unicode_ci		No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/> 7	category	bigint(20)		UNSIGNED	No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/> 8	created_at	timestamp			Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/> 9	updated_at	timestamp			Sí	NULL			Cambiar Eliminar Más

Tablas de categorías y autores:

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
<input type="checkbox"/> 1	id	bigint(20)		UNSIGNED	No	Ninguna		AUTO_INCREMENT	Cambiar Eliminar Más
<input type="checkbox"/> 2	name	varchar(255)	utf8mb4_unicode_ci		No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/> 3	created_at	timestamp			Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/> 4	updated_at	timestamp			Sí	NULL			Cambiar Eliminar Más

Para ejecutar crear contenido aleatorio en la base de datos utilizaremos los siguientes comandos:

```
php artisan make:seeder ArticleTableSeeder
```

```
php artisan make:seeder AuthorTableSeeder
```

```
php artisan make:seeder CategoryTableSeeder
```

Estos comandos nos crearan los archivos seeds que completaremos de la siguiente manera para posteriormente generar el contenido:

```

public function run()
{
    // Let's truncate our existing records to start from scratch.
    Article::truncate();

    $faker = \Faker\Factory::create();

    // And now, let's create a few articles in our database:
    for ($i = 0; $i < 200; $i++) {
        Article::create([
            'title' => $faker->sentence,
            'author' => $faker->numberBetween(1, 10),
            'published' => $faker->date('Y-m-d', 'now'),
            'content' => $faker->paragraph,
            'image' => $faker->word,
            'category' => $faker->numberBetween(1, 5),
        ]);
    }
}

```

```

public function run()
{
    $faker = \Faker\Factory::create();

    // And now, let's create a few articles in
    for ($i = 0; $i < 10; $i++) {
        Author::create([
            'name' => $faker->name,
        ]);
    }
}

```

```

public function run()
{
    $categories = array('National', 'International', 'Sports', 'Finance', 'Society',);

    foreach ($categories as $category) {
        Category::create([
            'name' => $category,
        ]);
    }
}

```

Por último sembraremos nuestra base de datos:

`php artisan db:seed`

Tabla articulos:

Opciones

Editar

Copiar

Borrar

1

Ut facere labore et ullam velit.

2

1987-08-24

Amet est saepe fugiat omnis fugiat enim. Ullam quo...

ut

1

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

2

Quia ab maxime laboriosam.

10

1985-05-19

Fuga tempora autem labore qui amet labore quia. De...

rerum

4

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

3

Sed et facilis perferendis.

10

2003-01-15

Unde velit itaque pariatur. Unde molestias eos lab...

et

5

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

4

Vitae architecto voluptatibus placeat accusamus.

1

2012-06-22

Occaecati qui ipsum ut est illum qui amet. Sit nih...

ratione

1

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

5

Voluptas quibusdam animi quasi voluptatibus.

1

2012-08-03

Perspiciatis sit tempore culpa praesentium. Id at ...

exercitationem

4

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

6

Deleniti quam minima possimus omnis est quidem sap...

10

1994-05-22

Tenetur molestiae ut repellendus et illum accusant...

corporis

1

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

7

Ut qui fugit temporibus autem.

6

2012-10-02

Fuga quis enim necessitatibus reprehenderit earam....

fuga

1

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

8

Ut ut ab molestias rem nisi magnam.

4

1994-12-28

Rerum quasi qui ipsum sed. Placeat dolores ea illo...

laborum

1

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

9

Consequatur pariatur omnis et provident.

5

1978-09-23

Aspernatur fugit aut dicta ratione omnis et. Illum...

rerum

4

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

10

Aut modi ullam molestiae molestiae suscipit iusto ...

9

2012-05-27

Quia architecto eos est debitis quos. Corrupti in ...

nulla

2

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

11

Animi a odio quos blanditiis.

6

1981-06-02

Eius rerum enim autem nostrum velit quo unde. Illu...

maiores

4

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

12

Officiis sit omnis voluptas voluptas aut voluptate...

7

2013-03-27

Quo non eius rerum ut illum cum consectetur. Est l...

ut

3

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

13

Quaerat sunt modi cum voluptatem tempore eaque.

2

1989-10-14

Voluptatum excepturi corrupti occaecati. Sint qui ...

sed

1

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

14

Sit eveniet porro aliquid nisi recusandae quidem d...

8

1977-05-25

Suscipit et nisi sequi nobis sed maxime voluptate ...

eius

5

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

15

Molestiae deserunt voluptas sint qui laboriosam si...

5

2012-11-27

Expedita eos voluptatem dolorum qui illo aperiam. ...

aperiam

3

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

16

Perferendis commodi voluptatem corrupti incidunt.

6

1998-03-18

Eligendi quis aliquid sed nemo voluptatem voluptat...

aut

1

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

17

Cumque eos provident cum voluptatem voluptas aut a...

3

1988-08-08

Architecto voluptas cupiditate quam iusto est face...

nobis

5

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

18

Est provident quis repellendus commodi enim.

10

2012-08-19

Eum sunt eum nemo eos fugiat rem beatae. Doloribus...

officiis

4

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

19

Cumque nam assumenda animi exercitationem id dolor...

6

2019-11-17

Sit sed sint neque nisi dicta et qui. Expedita err...

quibusdam

1

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

20

Itaque eos quam sit quisquam natus ea.

7

1988-09-26

Beatae nihil et tempore qui eum dolore ipsum. Et a...

unde

4

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

21

Ab fugiat dolor explicabo voluptatum voluptate dol...

4

2013-01-09

Et consequatur vitae quos commodi minus tenetur et...

praesentium

2

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

22

Et tempore molestias corporis quo.

6

1974-06-22

Velit possimus qui sed corrupti voluptate reprehen...

corrupti

1

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

23

Libero beatae dolorem molestias dignissimos debiti...

10

1991-08-26

Laudantium quidem eligendi minima beatae. Voluptas...

distinctio

5

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

24

Velit velit eius dolores iste.

10

1985-02-24

Est qui aut quibusdam aliquam totam. Voluptas dist...

et

3

2020-12-18 01:04:40

2020-12-18 01:04:40

Editar

Copiar

Borrar

25

Quod blanditiis labore recusandae quasi culpa dele...

3

1971-02-17

At natus ab ipsa voluptatem omnis molestias facere...

ipsum

1

2020-12-18 01:04:40

2020-12-18 01:04:40

Tabla Autores:

+ Opciones

<div>←T→</div>		id	name	created_at	updated_at
<div><div><div><div></div></div><div>Editar</div><div><div></div><div></div></div><div>Copiar</div><div><div></div><div></div></div><div>Borrar</div></div></div>		1	Alena Harvey	2020-12-18 01:04:31	2020-12-18 01:04:31
<div><div><div><div></div></div><div>Editar</div><div><div></div><div></div></div><div>Copiar</div><div><div></div><div></div></div><div>Borrar</div></div></div>		2	Cooper Cummings	2020-12-18 01:04:31	2020-12-18 01:04:31
<div><div><div><div></div></div><div>Editar</div><div><div></div><div></div></div><div>Copiar</div><div><div></div><div></div></div><div>Borrar</div></div></div>		3	Deonte Jenkins	2020-12-18 01:04:31	2020-12-18 01:04:31
<div><div><div><div></div></div><div>Editar</div><div><div></div><div></div></div><div>Copiar</div><div><div></div><div></div></div><div>Borrar</div></div></div>		4	Chelsey Quigley	2020-12-18 01:04:31	2020-12-18 01:04:31
<div><div><div><div></div></div><div>Editar</div><div><div></div><div></div></div><div>Copiar</div><div><div></div><div></div></div><div>Borrar</div></div></div>		5	Lesley Klein I	2020-12-18 01:04:31	2020-12-18 01:04:31
<div><div><div><div></div></div><div>Editar</div><div><div></div><div></div></div><div>Copiar</div><div><div></div><div></div></div><div>Borrar</div></div></div>		6	Beau Zulauf	2020-12-18 01:04:31	2020-12-18 01:04:31
<div><div><div><div></div></div><div>Editar</div><div><div></div><div></div></div><div>Copiar</div><div><div></div><div></div></div><div>Borrar</div></div></div>		7	Raphael Volkman	2020-12-18 01:04:31	2020-12-18 01:04:31
<div><div><div><div></div></div><div>Editar</div><div><div></div><div></div></div><div>Copiar</div><div><div></div><div></div></div><div>Borrar</div></div></div>		8	Cierra Kuhn III	2020-12-18 01:04:31	2020-12-18 01:04:31
<div><div><div><div></div></div><div>Editar</div><div><div></div><div></div></div><div>Copiar</div><div><div></div><div></div></div><div>Borrar</div></div></div>		9	Dr. Hildegard Bartoletti	2020-12-18 01:04:31	2020-12-18 01:04:31
<div><div><div><div></div></div><div>Editar</div><div><div></div><div></div></div><div>Copiar</div><div><div></div><div></div></div><div>Borrar</div></div></div>		10	Ms. Rosamond Nolan II	2020-12-18 01:04:31	2020-12-18 01:04:31

Tabla Categorías:

+ Opciones

<div><div>←T→</div></div>			id	name	created_at	updated_at
<div><div><div><div></div></div><div>Editar</div><div><div></div><div></div></div><div>Copiar</div><div><div></div><div></div></div><div>Borrar</div></div></div>		1	National	2020-12-18 01:04:26	2020-12-18 01:04:26	
<div><div><div><div></div></div><div>Editar</div><div><div></div><div></div></div><div>Copiar</div><div><div></div><div></div></div><div>Borrar</div></div></div>		2	International	2020-12-18 01:04:26	2020-12-18 01:04:26	
<div><div><div><div></div></div><div>Editar</div><div><div></div><div></div></div><div>Copiar</div><div><div></div><div></div></div><div>Borrar</div></div></div>		3	Sports	2020-12-18 01:04:26	2020-12-18 01:04:26	
<div><div><div><div></div></div><div>Editar</div><div><div></div><div></div></div><div>Copiar</div><div><div></div><div></div></div><div>Borrar</div></div></div>		4	Finance	2020-12-18 01:04:26	2020-12-18 01:04:26	
<div><div><div><div></div></div><div>Editar</div><div><div></div><div></div></div><div>Copiar</div><div><div></div><div></div></div><div>Borrar</div></div></div>		5	Society	2020-12-18 01:04:26	2020-12-18 01:04:26	

Para finalizar crearemos los siguientes Modelos:

Article.php:

```
class Article extends Model
{
    use HasFactory;

    protected $fillable = ['title', 'author', 'published', 'content', 'image', 'category'];

    public function author()
    {
        return $this->hasOne(Author::class, 'foreign_key');
    }

    public function category()
    {
        return $this->hasOne(Category::class, 'foreign_key');
    }
}
```

Author.php:

```
class Author extends Model
{
    use HasFactory;

    protected $fillable = ['name'];

    public function article()
    {
        return $this->belongsTo(Article::class, 'foreign_key');
    }
}
```

Category.php:

```
class Category extends Model
{
    use HasFactory;

    protected $fillable = ['name'];

    public function article()
    {
        return $this->belongsTo(Article::class, 'foreign_key');
    }
}
```

3. Creación de API de consulta pública y comprobación en navegador.

A continuación desarrollaremos todos los puntos de entrada a la API y comprobaremos su funcionamiento en un navegador web.

Para ello editaremos el archivo routes/api.php y crearemos los controladores para separar la lógica:

routes/api.php:

```
Route::get('articles/page/{page}', [ArticleController::class, 'index']);
Route::get('articles/{article}', [ArticleController::class, 'show']);
Route::get('categories/{category}/page/{page}', [CategoryController::class, 'show']);
Route::get('authors/', [AuthorController::class, 'index']);
Route::get('authors/{author}/page/{page}', [AuthorController::class, 'show']);
```

app/Http/Controllers/ArticleController.php:

```
<?php

namespace App\Http\Controllers;

use App\Models\Article;

class ArticleController extends Controller
{
    public function index($page)
    {
        // $article = Article::paginate(10, ['id', 'title', 'published'])->items();
        // return $article;
        return Article::paginate(10, ['id', 'title', 'published'], 'page', $page)->items();
    }

    public function show(Article $article)
    {
        return $article;
    }
}
```

app/Http/Controllers/AuthorController.php:

```
<?php

namespace App\Http\Controllers;

use App\Models\Author;
use App\Models\Article;
use Illuminate\Support\Facades\DB;

You, seconds ago | 1 author (You)
class AuthorController extends Controller
{
    public function show(Author $author, $page)
    {
        $articles = Article::where('author', '=', $author["id"])
        →paginate(10, ['id', 'title', 'author', 'published'], 'page,', $page)
        →items();      You, seconds ago • Uncommitted changes
        return $articles;
    }
}
```

app/Http/Controllers/CategoryController.php:

```
<?php

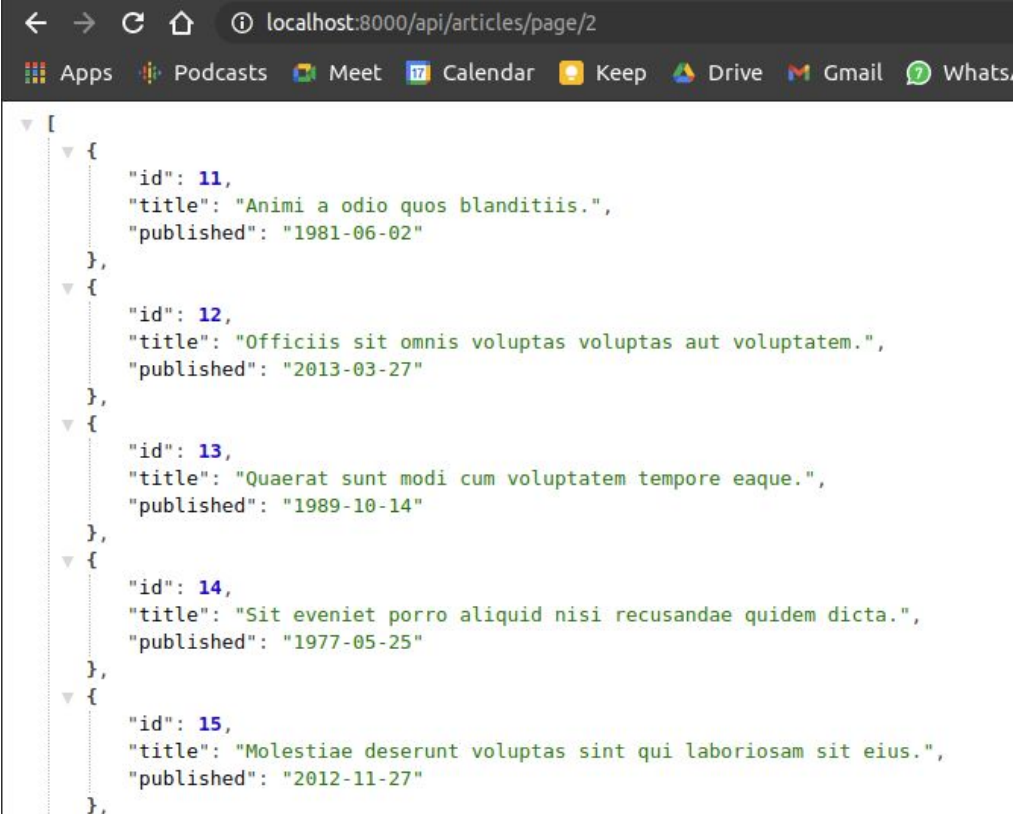
namespace App\Http\Controllers;

use App\Models\Category;
use App\Models\Article;
use Illuminate\Support\Facades\DB;

You, seconds ago | 1 author (You)
class CategoryController extends Controller
{
    public function show(Category $category, $page)
    {
        $articles = Article::where('category', '=', $category["id"])
        →paginate(10, ['id', 'title', 'category', 'published'], 'page,', $page)
        →items();      You, seconds ago • Uncommitted changes
        return $articles;
    }
}
```

Resultado en de las consultas en el navegador:

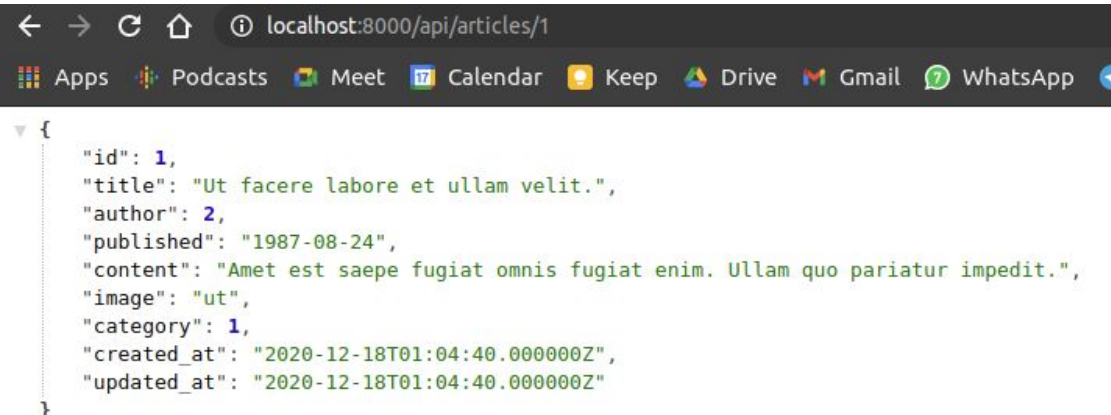
- api/noticias/<page> Con un parámetro <page> para seleccionar cada una de las páginas de resultado (con 10 noticias por página). Devuelve un listado con los títulos, ids y fechas de publicación de cada una de las noticias.



```
localhost:8000/api/articles/page/2

[
  {
    "id": 11,
    "title": "Animi a odio quos blanditiis.",
    "published": "1981-06-02"
  },
  {
    "id": 12,
    "title": "Officiis sit omnis voluptas voluptas aut voluptatem.",
    "published": "2013-03-27"
  },
  {
    "id": 13,
    "title": "Quaerat sunt modi cum voluptatem tempore eaque.",
    "published": "1989-10-14"
  },
  {
    "id": 14,
    "title": "Sit eveniet porro aliquid nisi recusandae quidem dicta.",
    "published": "1977-05-25"
  },
  {
    "id": 15,
    "title": "Molestiae deserunt voluptas sint qui laboriosam sit eius.",
    "published": "2012-11-27"
  }
]
```

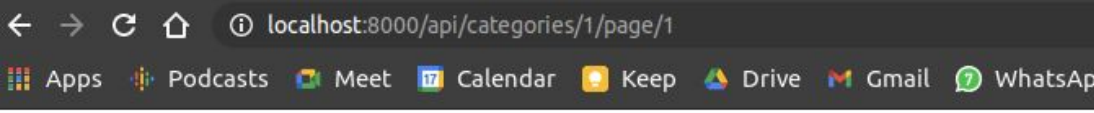
- api/noticia/<id> Donde <id> es el identificador de cada noticia. Devuelve todos los campos de la noticia.



```
localhost:8000/api/articles/1

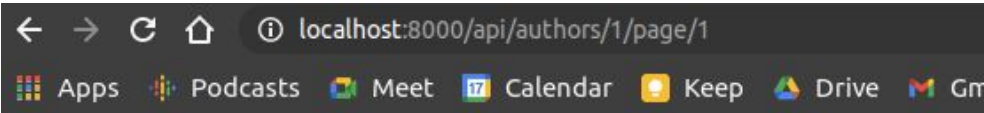
{
  "id": 1,
  "title": "Ut facere labore et ullam velit.",
  "author": 2,
  "published": "1987-08-24",
  "content": "Amet est saepe fugiat omnis fugiat enim. Ullam quo pariat impedit.",
  "image": "ut",
  "category": 1,
  "created_at": "2020-12-18T01:04:40.000000Z",
  "updated_at": "2020-12-18T01:04:40.000000Z"
}
```


- `api/categoria/<id>/<pagina>` Donde `<id>` es el identificador de la categoria. Devuelve un listado con los títulos, ids y fechas de publicación de cada una de las noticias. Y `<pagina>` la página de resultados deseada.



```
[
  {
    "id": 1,
    "title": "Ut facere labore et ullam velit.",
    "category": 1,
    "published": "1987-08-24"
  },
  {
    "id": 4,
    "title": "Vitae architecto voluptatibus placeat accusamus.",
    "category": 1,
    "published": "2012-06-22"
  },
  {
    "id": 6,
    "title": "Deleniti quam minima possimus omnis est quidem sapiente nemo.",
    "category": 1,
    "published": "1994-05-22"
  },
  {
    "id": 7,
    "title": "Ut qui fugit temporibus autem.",
    "category": 1,
    "published": "2012-10-02"
  }
],
```

- `api/autor/<id>/<pagina>` Donde `<id>` es el identificador del autor. Devuelve un listado con los títulos, ids y fechas de publicación de cada una de las noticias. Y `<pagina>` la página de resultados deseada.



```
[
  {
    "id": 4,
    "title": "Vitae architecto voluptatibus placeat accusamus.",
    "author": 1,
    "published": "2012-06-22"
  },
  {
    "id": 5,
    "title": "Voluptas quibusdam animi quasi voluptatibus.",
    "author": 1,
    "published": "2012-08-03"
  }
],
```

4. Migración de la web y publicación.

d) Finalmente migraremos el prototipo en la web de pruebas de la documentando brevemente los pasos seguidos y comprobando su funcionamiento.

Especificar la url de la aplicación en el servidor.

Por último para migrar al servidor deberemos:

En primer lugar, exportar la base de datos generada en local e importarla en nuestro servidor.

Filtros

Que contengan la palabra:

Tabla	Acción							Filas	Tipo	Cotejamiento	Tamaño	Residuo a depurar
<input type="checkbox"/> articles	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	200	InnoDB	utf8mb4_unicode_ci	128.0 KB	-
<input type="checkbox"/> authors	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	10	InnoDB	utf8mb4_unicode_ci	16.0 KB	-
<input type="checkbox"/> categories	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	5	InnoDB	utf8mb4_unicode_ci	16.0 KB	-
<input type="checkbox"/> migrations	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	3	InnoDB	utf8mb4_unicode_ci	16.0 KB	-
4 tablas	Número de filas							218	InnoDB	latin1_swedish_ci	176.0 KB	0 B

A continuación cambiaremos la configuración del archivo .env:

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=anibalsantosgo
DB_USERNAME=anibalsantosgo
DB_PASSWORD=MYPASS
```

Ya tendríamos seteada la configuración de nuestra base de datos.

Para finalizar subimos la carpeta de nuestro proyecto al servidor por FTP y daremos permisos 775 a nuestras dependencias para asegurarnos de que todo funciona correctamente.

La url de acceso sería:

<https://eimtcms.eimt.uoc.edu/~anibalsantosgo/laravel-news/public/>

Y las de consulta de la API:

<https://eimtcms.eimt.uoc.edu/~anibalsantosgo/laravel-news/public/api/articles/page/1>

<https://eimtcms.eimt.uoc.edu/~anibalsantosgo/laravel-news/public/api/articles/1>

<https://eimtcms.eimt.uoc.edu/~anibalsantosgo/laravel-news/public/api/categories/1/page/1>

<https://eimtcms.eimt.uoc.edu/~anibalsantosgo/laravel-news/public/api/authors/1/page/1>

5. Diferencias entre CodeIgniter y Laravel.

- CodeIgniter:
 - El núcleo de CodeIgniter es bastante ligero y las páginas se procesan muy rápido
 - Sencillo de instalar, basta con subir los archivos al ftp y tocar un archivo de configuración para definir el acceso a la base de datos.
 - La reutilización de código es fácil y rápida.
 - Crear nuevos módulos, páginas o funcionalidades es relativamente fácil.
 - Motor Vista Controlador: Separación de la lógica y arquitectura de la web.
 - La curva de aprendizaje es muy rápida, permitiendo ver resultados muy rápidamente.
 - Al no definir una manera estricta de trabajar puede ser difícil trabajar en equipo.
 - Al poder utilizar cualquier versión de PHP pueden darse fallos de seguridad en las versiones más antiguas.
 - El ORM de CodeIgniter es muy pobre. Lo que tiene parece más un constructor de consultas SQL con interfaz de OOP.

- Laravel:
 - Utiliza un sistema ORM muy potente: Eloquent, que permite incorporar fácilmente paginación automática, relaciones uno-a-uno, uno-a-muchos, muchos-a-muchos, polimórficas, validación, protección de campos, generación de arrays/json...
 - Artisan permite realizar muchas acciones directamente desde la consola y programar acciones propias.
 - Utiliza Composer.
 - Cuenta con un sistema de migraciones para gestionar la estructura de la base de datos.
 - La instalación se hace mediante Composer, no es tan fácil como CodeIgniter.
 - Está pensado para las últimas versiones de PHP y solo soporta a partir de PHP 5.4
 - La curva de aprendizaje es más lenta de CodeIgniter.