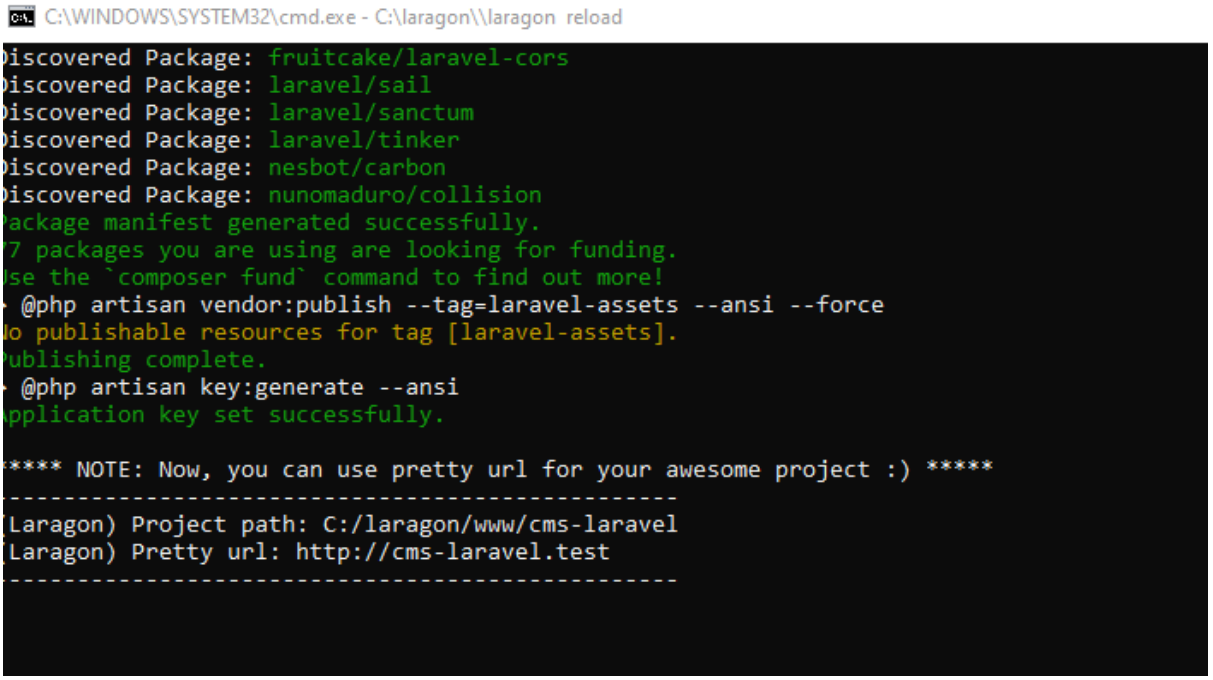
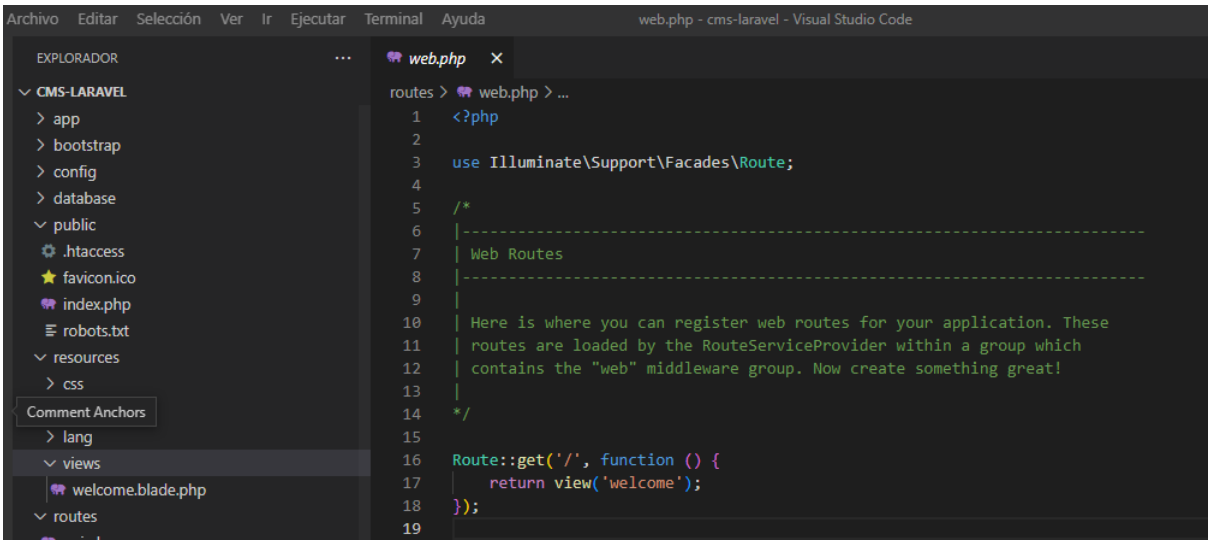
UF 2218 práctica

Para la realización de esta práctica seguimos el tutorial de laravel para la creación de un CMS: https://www.youtube.com/watch?v=nFKwq2PHJ64

Creamos un sitio nuevo de laravel a través de laragon





En este caso el archivo *web.php* nos devuelve la vista de la ruta "*welcome.blade.php*"

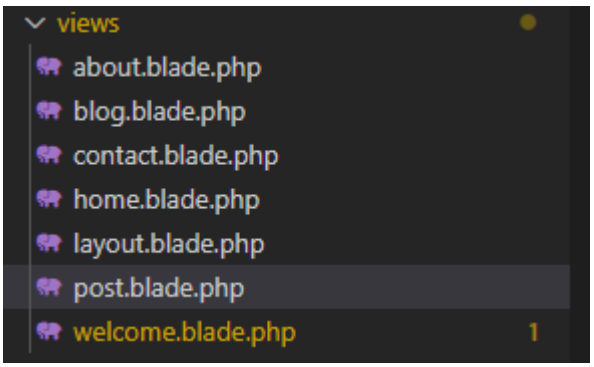
web.php

routes > web.php

```php
<?php

use Illuminate\Support\Facades\Route;

/*
|--------------------------------------------------------------------------
| 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::view('/', 'home')->name('home');
Route::view('acerca-de', 'about')->name('about');

Route::get('blog', 'BlogController@index')->name('blog.index');
Route::get('blog/{post:slug}', 'BlogController@show')->name('blog.show');

Route::view('contactos', 'contact')->name('contact');

```

De esta forma como se ve en la imagen creamos nuevas rutas. En el primer *route get blog* llamamos a la bbdd

A continuación creamos las nuevas vistas añadiendo los archivos que se ven en la siguiente imagen.



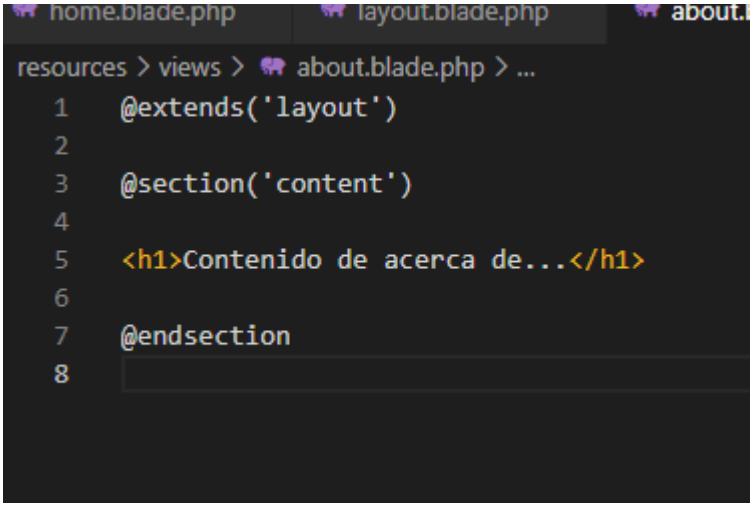
De entre ellos destacamos el archivo *layout.blade.php* que sería como nuestra plantilla.

home.blade.php | layout.blade.php | about.blade.php | blog.blade.php | contact.blade.p

esources > views > layout.blade.php > ...

```
1   <!DOCTYPE html>
2   <html lang="en">
3   <head>
4       <meta charset="UTF-8">
5       <meta http-equiv="X-UA-Compatible" content="IE=edge">
6       <meta name="viewport" content="width=device-width, initial-scale=1.0">
7       <title>cms laravel</title>
8   </head>
9   <body>
10      Menu
11
12      <div>
13          @yield('content')
14      </div>
15  </body>
16  </html>
17
```

Damos forma a las demás vistas declarando con el siguiente código lo que se vería en los archivos restantes de la carpetas views como se ve en las siguientes imágenes.



home.blade.php | layout.blade.php | about.blade.php | blog.blade.php

resources > views > blog.blade.php > h1

```blade
@extends('layout')

@section('content')

<h1>Contenido de blog</h1>

@endsection

```

erminal Ayuda contact.blade.php - cms-laravel - Visual Studio Code

home.blade.php | layout.blade.php | about.blade.php | blog.blade.php | contact.blade.php

resources > views > contact.blade.php > ...

```blade
@extends('layout')

@section('content')

<h1>Contenido de contactos</h1>

@endsection

```

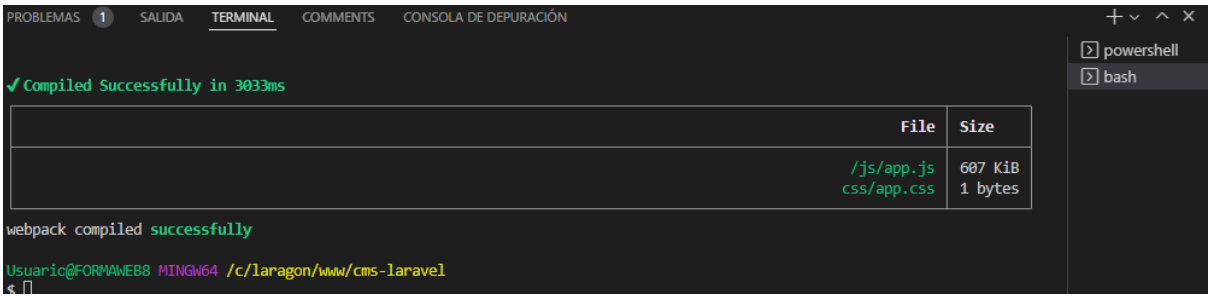
A continuación en el archivo *layout* añadimos el siguiente código, de esta forma generamos el menú.

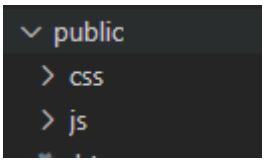
web.php | home.blade.php | layout.blade.php | about.blade.php | blog.blade.php

sources > views > layout.blade.php > html > body > ul

```blade
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>cms laravel</title>
</head>
<body>
  <ul>
      <li><a href="{{ route('home') }}">Home</a></li>
      <li><a href="{{ route('about') }}">Acerca de</a></li>
      <li><a href="{{ route('blog.index') }}">Blog</a></li>
      <li><a href="{{ route('contact') }}">Contactos</a></li>
  </ul>

    <div>
        @yield('content')
    </div>
</body>
</html>

```

Una vez hecho esto ejecutamos el comando *npm install* y *npm run dev*. De esta forma se crean nuevos archivos en la carpeta *public*, con especial relevancia el archivo *app.css* dentro de *css*





A continuación si visualizamos nuestro proyecto en el navegador este es el resultado:

Después en el archivo *.env* configuramos el nombre de la base de datos y en el archivo *layout.blade.php* enlazamos la hoja de estilos *app.css*

Una vez hecho esto ejecutamos el comando *php artisan make:model Post-msfc*

```
C:\laragon\www\cms-laravel
λ php artisan make:model Post -msfc
Model created successfully.
Factory created successfully.
Created Migration: 2022_05_23_110556_create_posts_table
```

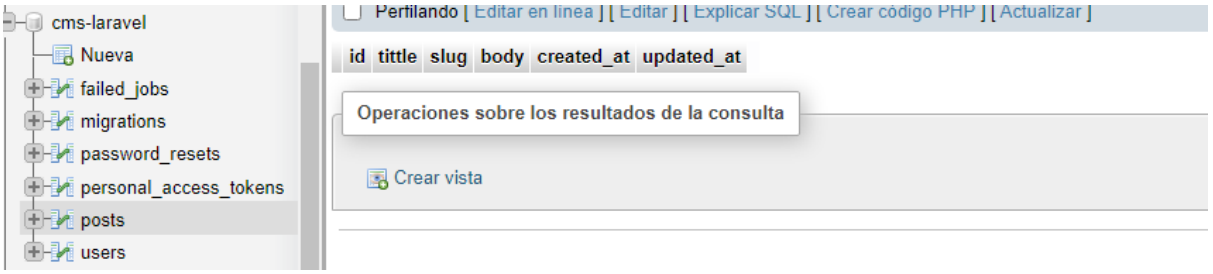
De esta forma en nuestra base de datos se han creado nuevas tablas, tal y como se recoge en el siguiente código.

web.php | .env | 2022_05_23_110556_create_posts_table.php | layout.blade.php

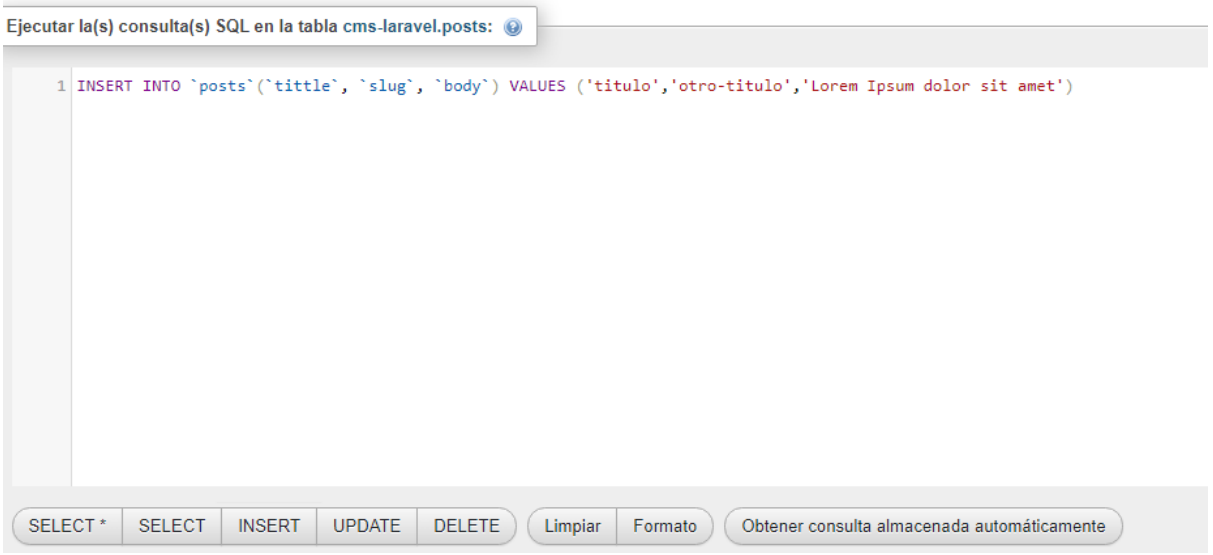
database > migrations > 2022_05_23_110556_create_posts_table.php > CreatePostsTable > up > #Function#bf6b1d3c

```php
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreatePostsTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('posts', function (Blueprint $table) {
            $table->id();

            $table->string('tittle');
            $table->string('slug');
            $table->text('body');

            $table->timestamps();
        });
    }
```

Podemos ver como se encuentra la tabla:



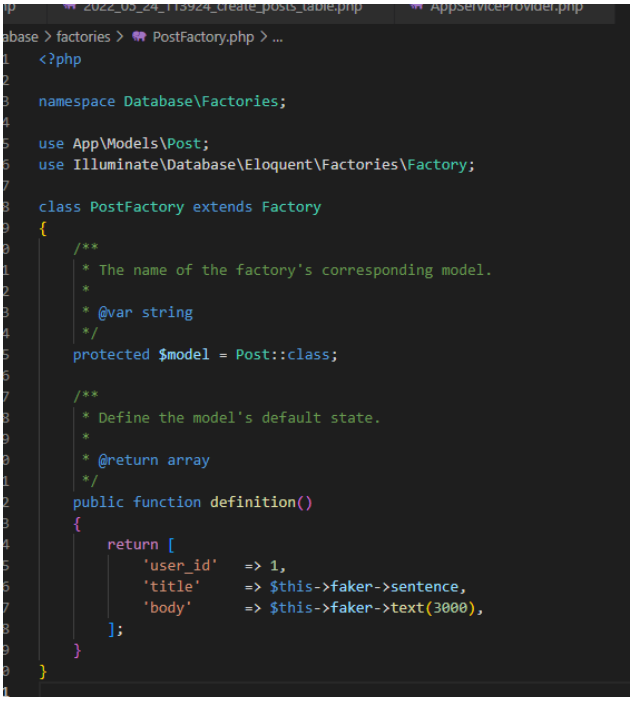
Podemos crear nuevos post a través de *phpmyAdmin*



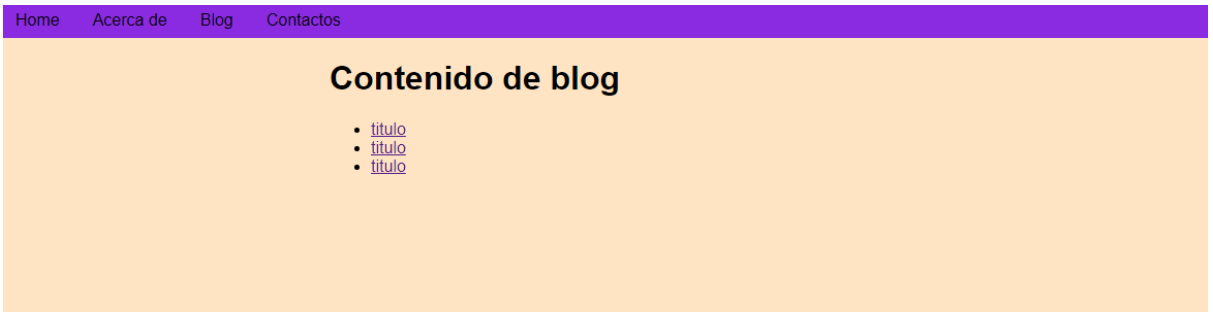
Cambiamos el archivo *PostController.php* y lo llamamos *BlogController.php*

Ejecutamos *php artisan cache:clear*

A continuación cambiamos el código en el archivo *PostFactory.phpi*



Añadimos estilos en ambos *app.css*



Finalmente, pinchando en cada post podemos ver su contenido:

Para terminar instalamos un framework de estilo en este caso tailwind. El resultado final de la práctica se visualizará de la siguiente forma:

Home Acerca de Blog Contactos

Fresas

La dulce fruta roja

Recetas →

☐ Lista de la compra

Arándanos

Un pequeño toque dulce

Recetas →

☐ Lista de la compra

Limones

Dale un toque ácido a tu día

Recetas →

☐ Lista de la compra

Home Acerca de Blog Contactos

Formulario de contactos

Nombre:

Apellido:

Receta deseada:

Tiempo de realización:

Regista tu email:

Enviar