

Login y autenticación en Laravel



SERVICIO NACIONAL DE APRENDIZAJE - SENA

Centro de Servicios Financieros

Análisis y desarrollo de software

PHP-Laravel

Presentado por:

Juan Manuel Galindo Rojas

Bogotá, Diciembre de 2023

Contenido

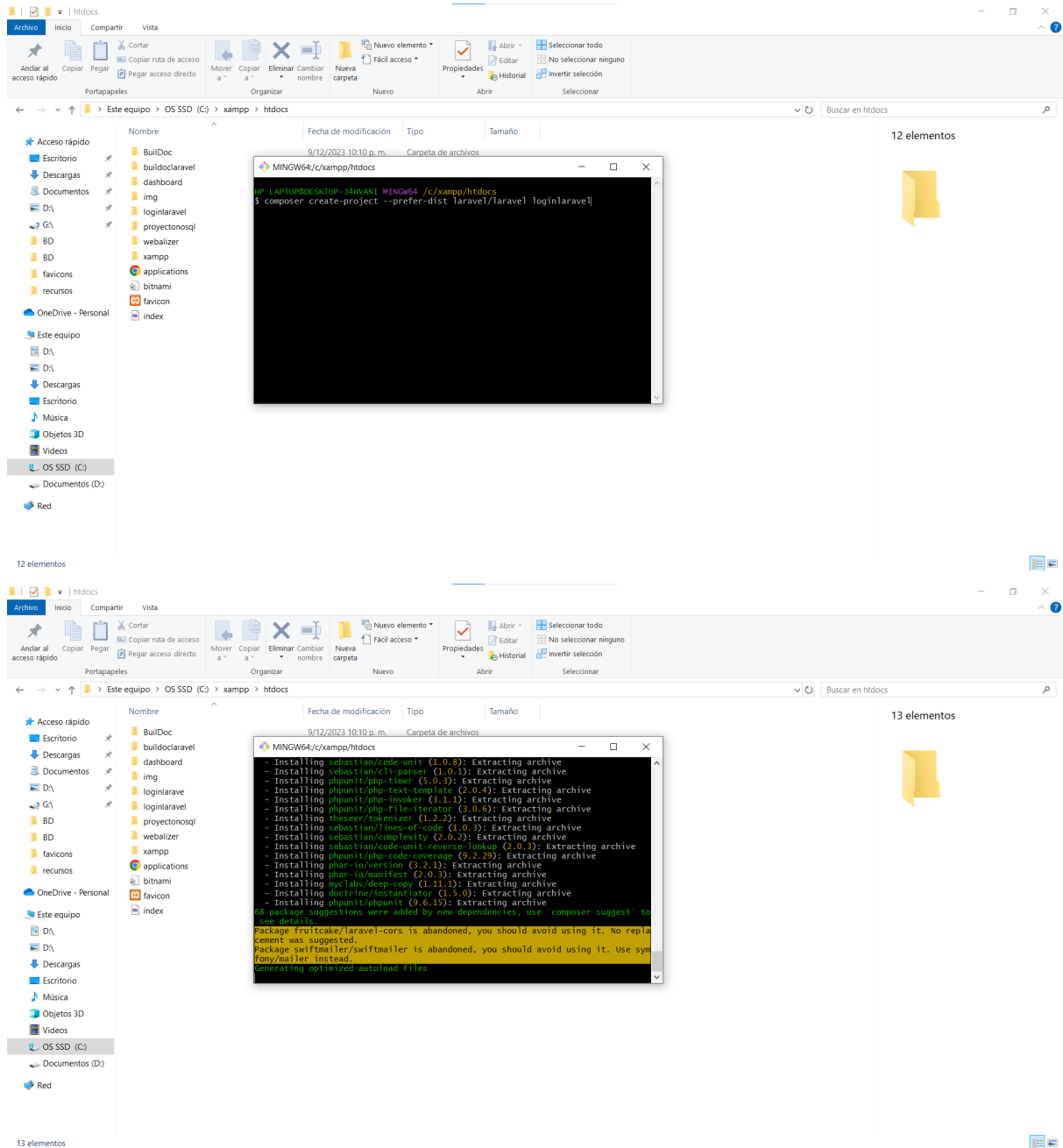
Contenido.....	2
1. Creación del Proyecto Laravel:.....	3
2. Instalación de Laravel/UI:.....	4
3. Creación de las Vistas de Autenticación con Bootstrap:.....	4
4. Instalación de Dependencias y Compilación de Recursos:.....	5
5. Creación y configuración de la Base de Datos:.....	6
6. Migración de la Base de Datos:.....	7
7. Personalización del Sistema de Autenticación:.....	8
8. Pruebas y Verificación del Sistema:.....	8

1. Creación del Proyecto Laravel:

En la fase inicial del proyecto, se inició la creación de un nuevo proyecto Laravel ejecutando el siguiente comando en la terminal:

composer create-project --prefer-dist laravel/laravel nombre-del-proyecto

Este comando descargó e instaló la versión más reciente de Laravel, junto con sus dependencias, y configuró la estructura básica del proyecto

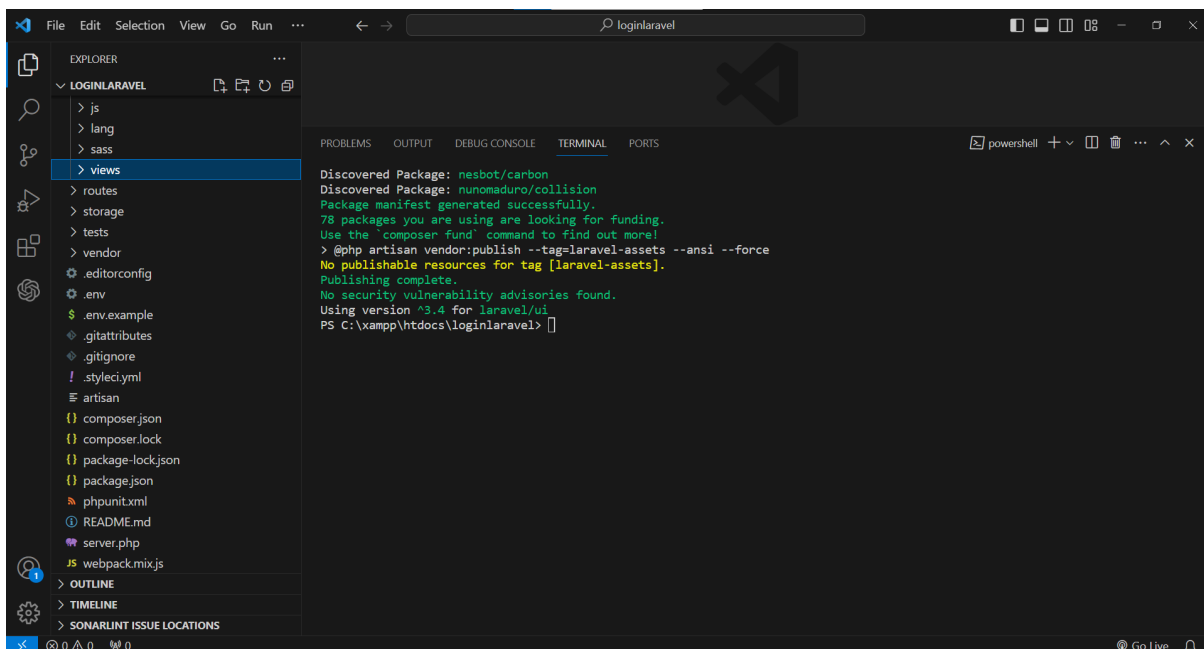


2. Instalación de Laravel/UI:

Se optó por integrar Laravel/UI para simplificar la generación de vistas relacionadas con la autenticación. Para llevar a cabo esta tarea, el equipo ejecutó el siguiente comando en la terminal:

composer require laravel/ui

Este comando permitió la instalación de Laravel/UI y la inclusión de las herramientas necesarias para la generación de las vistas de autenticación.

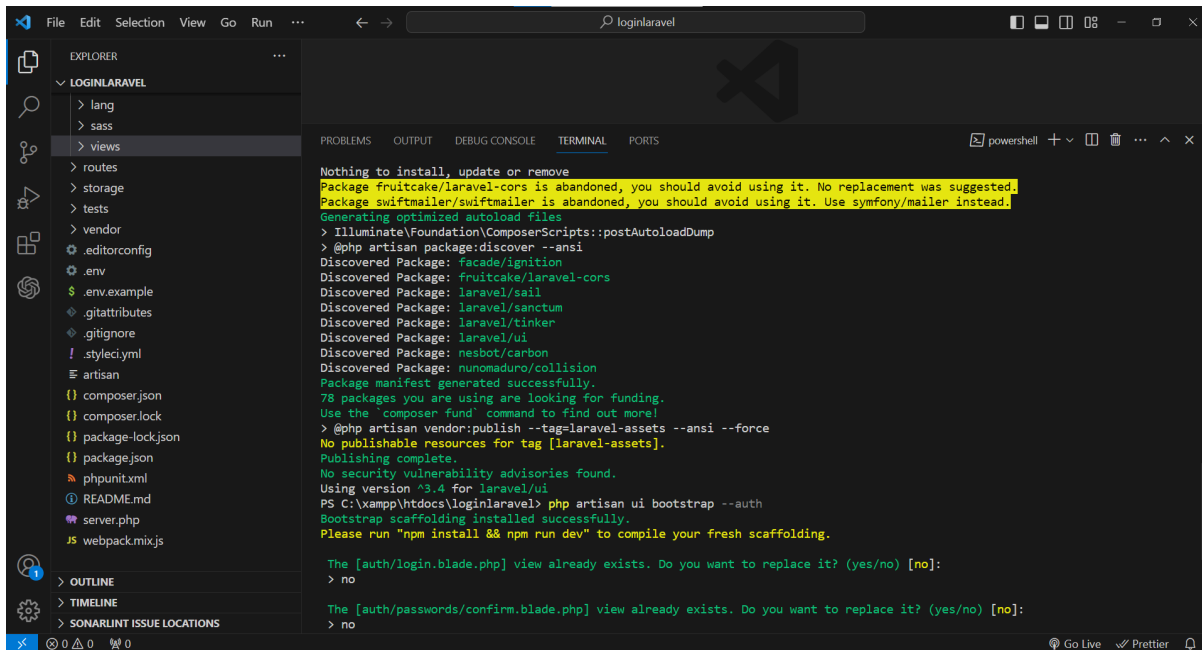


3. Creación de las Vistas de Autenticación con Bootstrap:

Una vez instalado Laravel/UI, se procedió a la generación de las vistas utilizando el comando:

php artisan ui bootstrap --auth

La opción `--auth` instruyó a Laravel a generar automáticamente las vistas básicas necesarias para un sistema de autenticación, aprovechando el framework Bootstrap para el diseño visual.



The screenshot shows the Visual Studio Code interface with the Explorer panel on the left displaying the file structure of a project named 'LOGINLARAVEL'. The file structure includes folders like 'lang', 'sass', 'views', 'routes', 'storage', 'tests', 'vendor', and files like '.editorconfig', '.env', '.env.example', '.gitattributes', '.gitignore', 'styleci.yml', 'artisan', 'composer.json', 'composer.lock', 'package-lock.json', 'package.json', 'phpunit.xml', 'README.md', 'server.php', and 'webpack.mix.js'. The Terminal panel on the right shows the output of several commands executed in a PowerShell window. The commands and their outputs are as follows:

```
Nothing to install, update or remove
Package fruitcake/laravel-cors is abandoned, you should avoid using it. No replacement was suggested.
Package swiftmailer/swiftmailer is abandoned, you should avoid using it. Use symfony/mailer instead.
Generating optimized autoload files
> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi
Discovered Package: facade/ignition
Discovered Package: fruitcake/laravel-cors
Discovered Package: laravel/sail
Discovered Package: laravel/sanctum
Discovered Package: laravel/tinker
Discovered Package: laravel/ui
Discovered Package: nesbot/carbon
Discovered Package: nunomaduro/collision
Package manifest generated successfully.
78 packages you are using are looking for funding.
Use the 'composer fund' command to find out more!
> @php artisan vendor:publish --tag=laravel-assets --ansi --force
No publishable resources for tag [laravel-assets].
Publishing complete.
No security vulnerability advisories found.
Using version ^3.4 for laravel/ui
PS C:\xampp\htdocs\loginlaravel> php artisan ui bootstrap --auth
Bootstrap scaffolding installed successfully.
Please run "npm install && npm run dev" to compile your fresh scaffolding.

The [auth/login.blade.php] view already exists. Do you want to replace it? (yes/no) [no]:
> no

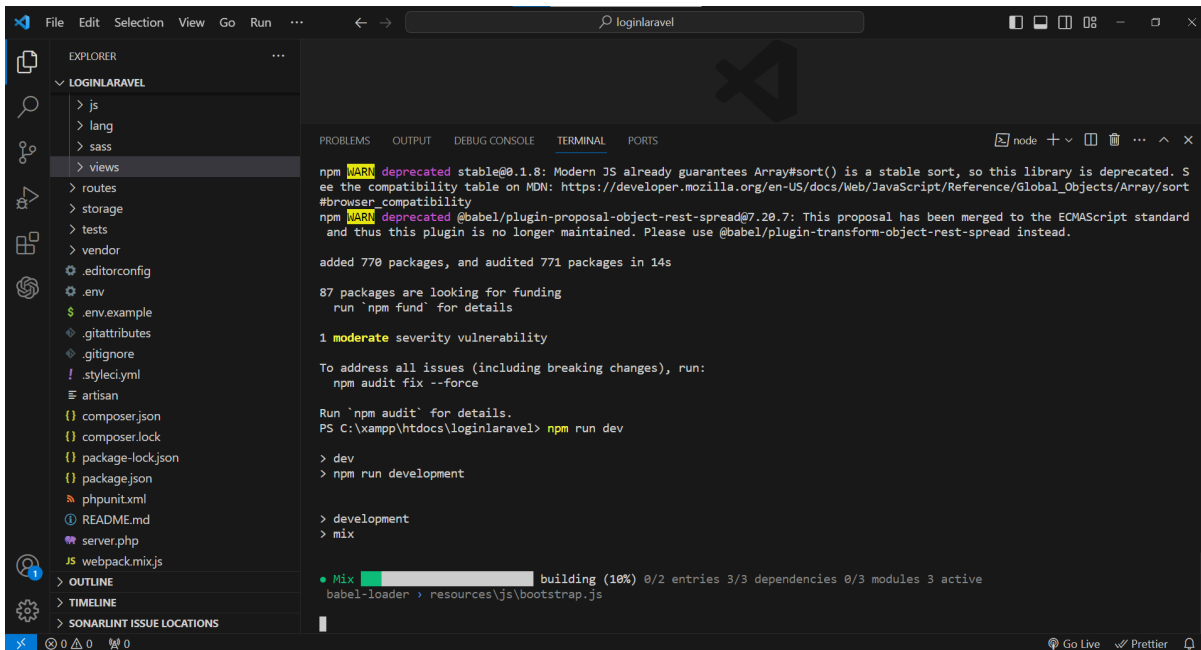
The [auth/passwords/confirm.blade.php] view already exists. Do you want to replace it? (yes/no) [no]:
> no
```

4. Instalación de Dependencias y Compilación de Recursos:

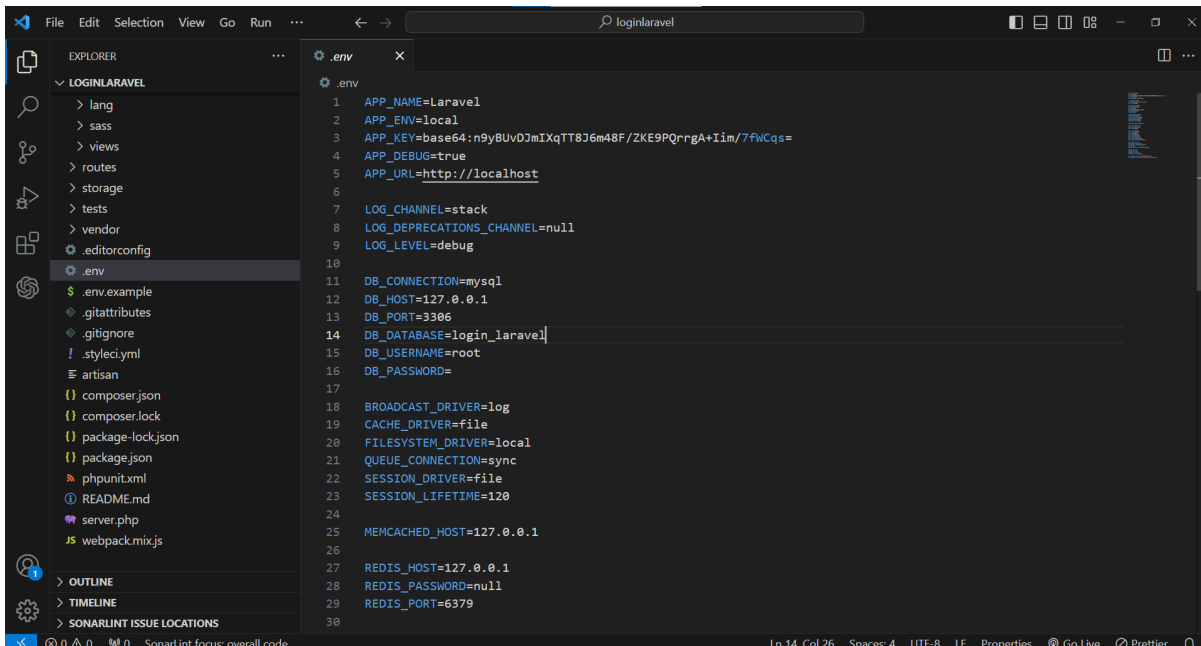
Con las vistas generadas, se prosiguió a la instalación de las dependencias JavaScript y CSS, así como a la compilación de los recursos necesarios. Esto se logró mediante los siguientes comandos:

npm install
npm run dev

La instalación de dependencias se encargó de asegurar la presencia de las bibliotecas requeridas, mientras que la compilación de recursos garantizó la correcta integración de Bootstrap con el sistema.



Para la persistencia de los datos relacionados con la autenticación, se modificaron las configuraciones de la base de datos en el archivo .env. En este paso, se ingresaron los detalles de la base de datos utilizada y las credenciales correspondientes.



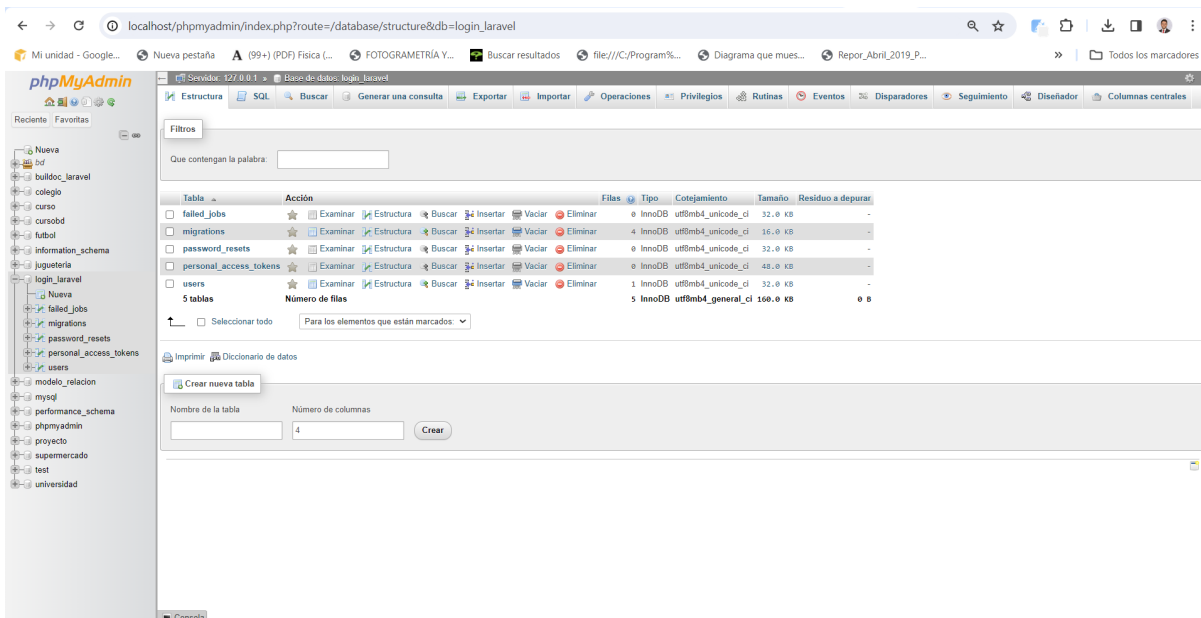
```
.env
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=base64:n9y8UvD3mIXqTT8J6m48F/ZKE9PQrrgA+Iim/7fWQqs=
4 APP_DEBUG=true
5 APP_URL=http://localhost
6
7 LOG_CHANNEL=stack
8 LOG_DEPRECATIONS_CHANNEL=null
9 LOG_LEVEL=debug
10
11 DB_CONNECTION=mysql
12 DB_HOST=127.0.0.1
13 DB_PORT=3306
14 DB_DATABASE=login_laravel
15 DB_USERNAME=root
16 DB_PASSWORD=
17
18 BROADCAST_DRIVER=log
19 CACHE_DRIVER=file
20 FILESYSTEM_DRIVER=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
```

6. Migración de la Base de Datos:

La migración de la base de datos se realizó mediante el comando:

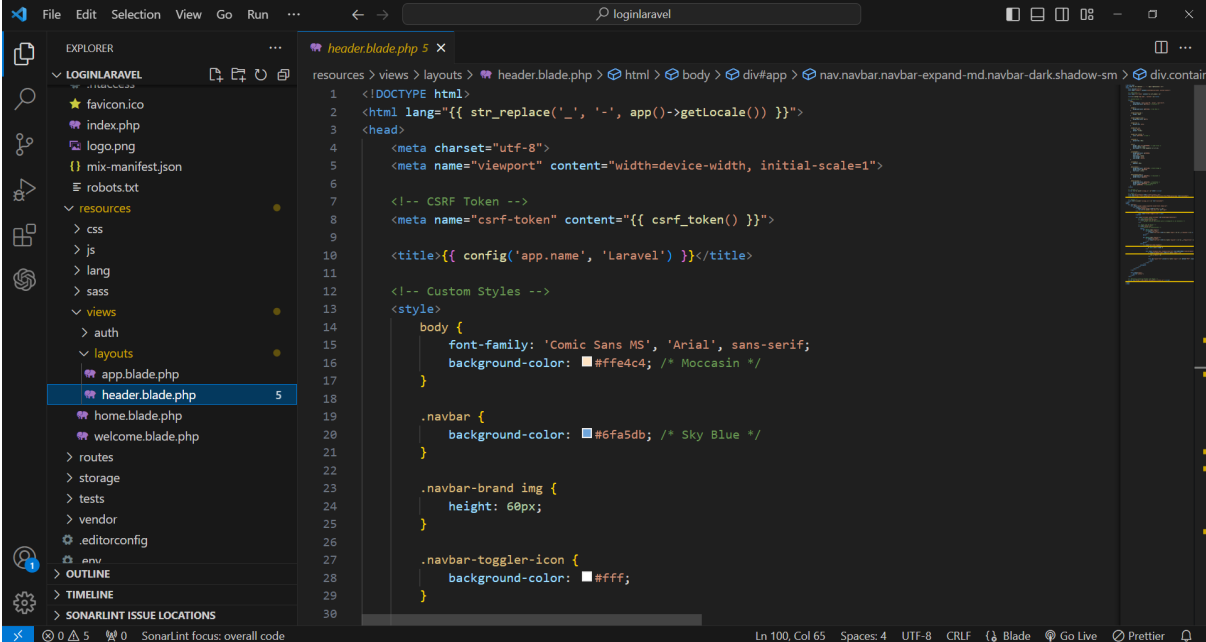
php artisan migrate

Este comando ejecutó las migraciones pendientes, creando así las tablas necesarias para gestionar la autenticación de usuarios en la base de datos.



7. Personalización del Sistema de Autenticación:

Fue necesario realizar ajustes adicionales para adaptar el sistema de autenticación a las necesidades específicas del proyecto. Esto incluyó la modificación de vistas generadas por Laravel/UI.



The screenshot shows a code editor with the file explorer on the left displaying the project structure. The file `header.blade.php` is selected under the `resources > views > layouts` directory. The main editor area shows the content of `header.blade.php`, which includes HTML boilerplate, a CSRF token, and custom CSS styles for the navigation bar.

```
1 <!DOCTYPE html>
2 <html lang="{{ str_replace('_', '-', app()->getLocale()) }}">
3 <head>
4     <meta charset="utf-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1">
6
7     <!-- CSRF Token -->
8     <meta name="csrf-token" content="{{ csrf_token() }}">
9
10    <title>{{ config('app.name', 'Laravel') }}</title>
11
12    <!-- Custom Styles -->
13    <style>
14        body {
15            font-family: 'Comic Sans MS', 'Arial', sans-serif;
16            background-color: #ffe4c4; /* Moccasin */
17        }
18
19        .navbar {
20            background-color: #6fa5db; /* Sky Blue */
21        }
22
23        .navbar-brand img {
24            height: 60px;
25        }
26
27        .navbar-toggler-icon {
28            background-color: #fff;
29        }
30    </style>
```

8. Pruebas y Verificación del Sistema:

Se llevó a cabo un proceso exhaustivo de pruebas para garantizar la funcionalidad correcta del sistema de autenticación. Se verificaron casos de éxito, así como situaciones de manejo de errores, con el objetivo de asegurar la robustez y seguridad del sistema implementado.

[Ingresar](#)[Registrarse](#)

Iniciar sesión

Dirección de correo electrónico

Contraseña

☐ Recuérdame[Iniciar sesión](#)[¿Olvidaste tu contraseña?](#)

localhost/loginlaravel/public/register

Mi unidad - Google... Nueva pestaña A (99+) (PDF) Física (... FOTOGRAFÍA Y... Buscar resultados file:///C:/Program%... Diagrama que mues... Repor_Abril_2019_P... Todos los marcadores

[Ingresar](#)[Registrarse](#)

Registro

Nombre

JUAN GALINDO

Dirección de correo electrónico

jmgalindor@unaLedu.co

Contraseña

.....

Confirmar contraseña

.....

[Registrar](#)