

# Universidad Autónoma de San Luis Potosí

Facultad de Ingeniería

Área de Ciencias de la Computación


Aplicaciones Web Interactivas

Llamas De la Torre María Jocelyn  
Manual de Programador



# Descargas

Laragon: <https://laragon.org/download/index.html>

[Docs](#)[Why Laragon?](#)[Testimonials](#)[Download](#)[About](#)[Community](#)[English](#)

## Download

Laragon is a universal development environment. It has many features to make you more productive:

**Benefits of Laragon**

After downloading, You can add **git**, **phpmyadmin**, **Node.js/MongoDB**, **Python/Django/Flask/Postgres**, **Ruby**, **Java**, **Go** using "Tools > Quick add"

Note: **You can also download from GitHub**

### Edition

**Download Laragon - Full (147 MB)**

- Laragon Full (64-bit):** Apache 2.4, Nginx, MySQL 5.7, PHP 7.4, Redis, Memcached, Node.js 14, npm, git, bitmana...

**Download Laragon - Portable (38 MB)**

- Laragon Portable:** PHP 5.4, MySQL 5.1, bitmana - Good for getting started with PHP, then you can add newer versions of PHP/MySQL easily later using "Tools > Quick add"

### Other versions

Laragon 5.0 (32-bit)

### Migrate to Laragon

Migrate from WAMP  
Migrate from XAMPP  
Migrate from others

### Tutorials

How to install Rails  
Build a Blog with Rails  
How to install Django  
How to install Flask


### Misc.

How to install

### Contents

[Edition](#)  
[Why Laragon?](#)  
[Tutorials](#)  
[Back to Top](#)

Laravel: <https://laravel.com/docs/9.x>

**Laravel**

[Prologue](#)[Getting Started](#)

- Installation**
- [Configuration](#)
- [Directory Structure](#)
- [Starter Kits](#)
- [Deployment](#)

[Architecture Concepts](#)[The Basics](#)[Digging Deeper](#)[Security](#)[Database](#)[Eloquent ORM](#)[Testing](#)

Q Search

VERSION  
9.x

## Installation

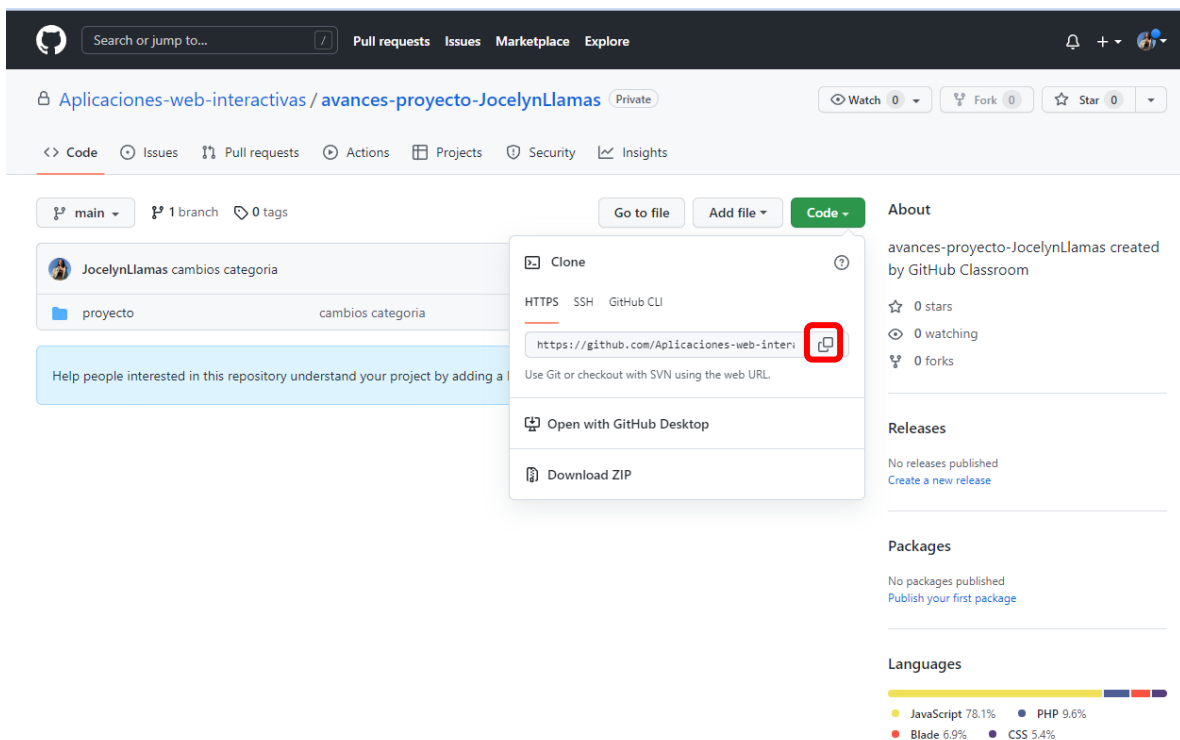
- # Meet Laravel**
  - [# Why Laravel?](#)
- # Your First Laravel Project**
  - [# Getting Started On macOS](#)
  - [# Getting Started On Windows](#)
  - [# Getting Started On Linux](#)
  - [# Choosing Your Sail Services](#)
  - [# Installation Via Composer](#)
- # Initial Configuration**
  - [# Environment Based Configuration](#)
  - [# Directory Configuration](#)
- # Next Steps**
  - [# Laravel The Full Stack Framework](#)
  - [# Laravel The API Backend](#)

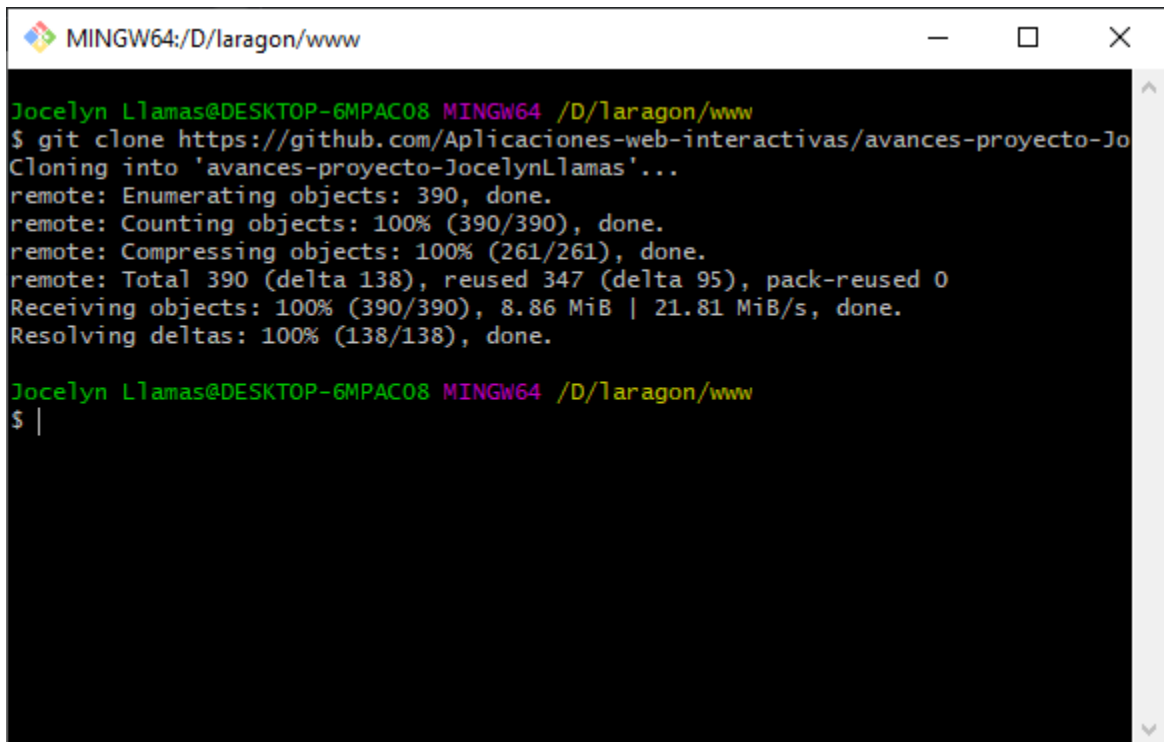
# Montaje del Sistema

Una vez instalado Laragon y Laravel, procedemos a clonar el repositorio en el que se encuentra el proyecto:

<https://github.com/Aplicaciones-web-interactivas/avances-proyecto-JocelynLlamas>

Copiamos el link del código y nos dirigimos a una terminal, ya sea CMD, PowerShell o GitBash, para clonar el repositorio a nuestra carpeta.



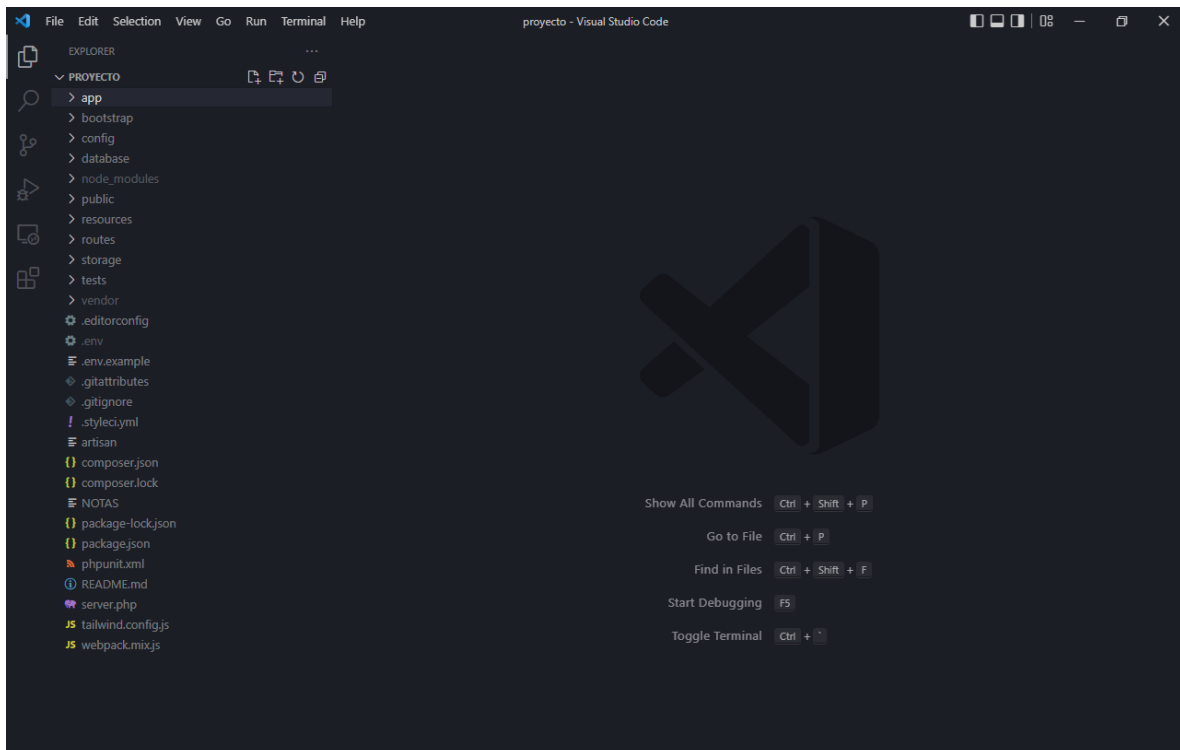


```
MINGW64:/D:/laragon/www

Jocelyn Llamas@DESKTOP-6MPAC08 MINGW64 /D:/laragon/www
$ git clone https://github.com/Aplicaciones-web-interactivas/avances-proyecto-Jo
Cloning into 'avances-proyecto-JocelynLlamas'...
remote: Enumerating objects: 390, done.
remote: Counting objects: 100% (390/390), done.
remote: Compressing objects: 100% (261/261), done.
remote: Total 390 (delta 138), reused 347 (delta 95), pack-reused 0
Receiving objects: 100% (390/390), 8.86 MiB | 21.81 MiB/s, done.
Resolving deltas: 100% (138/138), done.

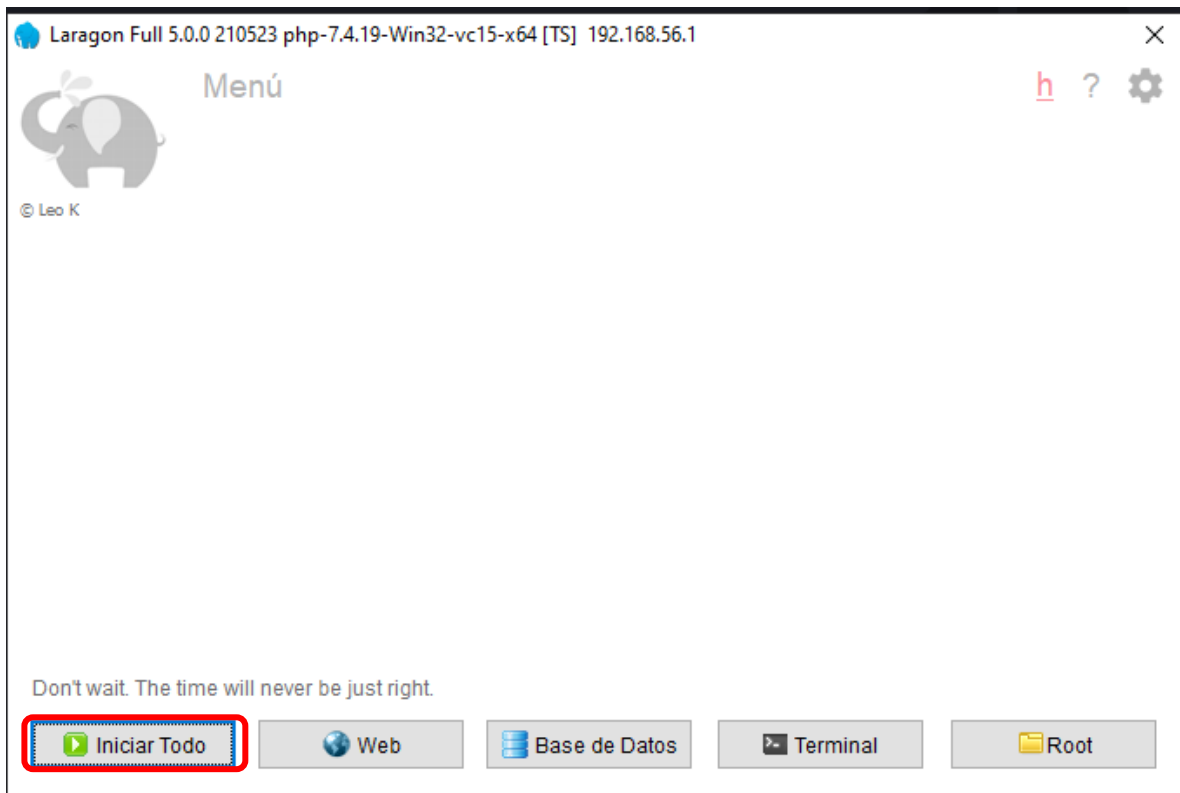
Jocelyn Llamas@DESKTOP-6MPAC08 MINGW64 /D:/laragon/www
$ |
```

Una vez clonado el repositorio, nos dirigimos a Visual Studio Code y abrimos la carpeta de "proyecto".

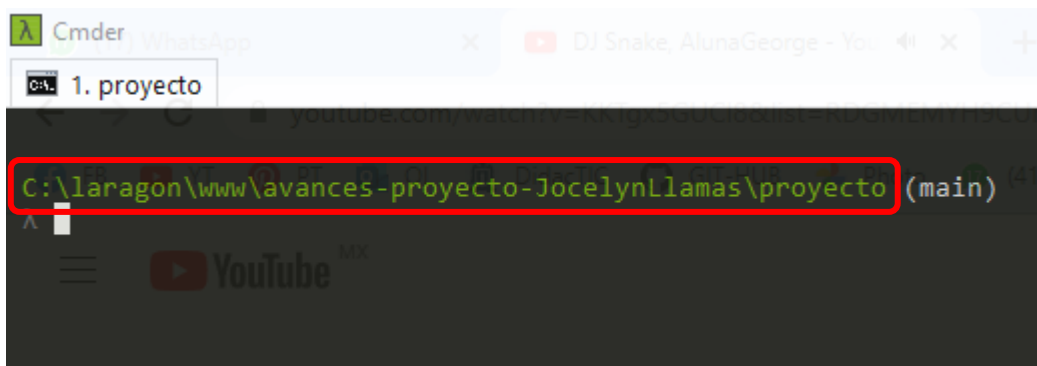
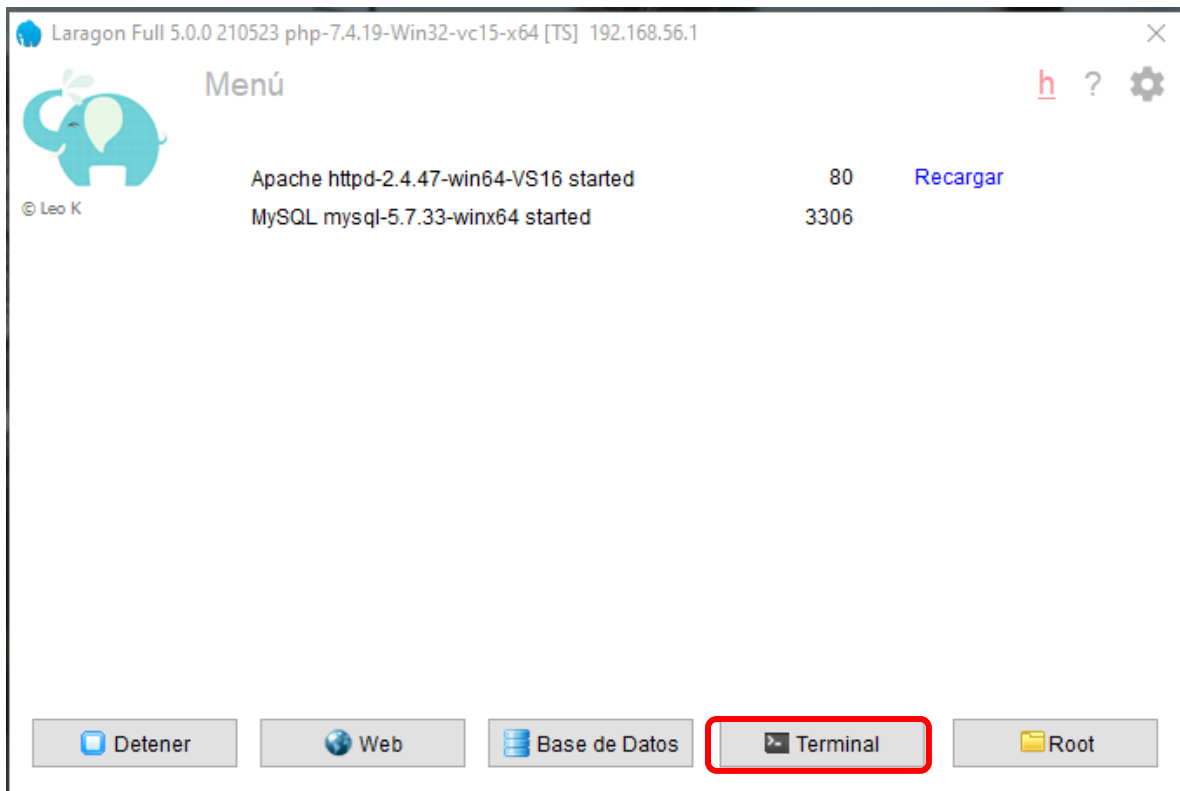


## Ejecución

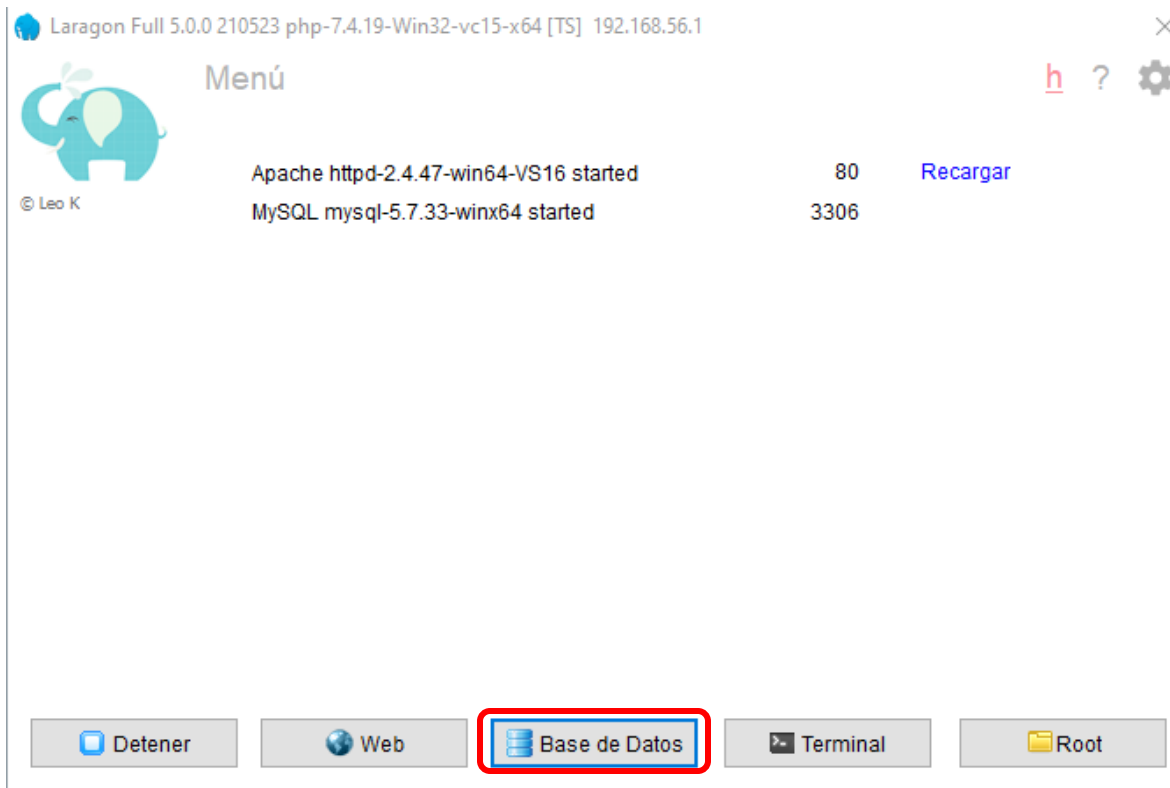
Para visualizar el Proyecto abriremos Laragon y daremos clic en Iniciar Todo.



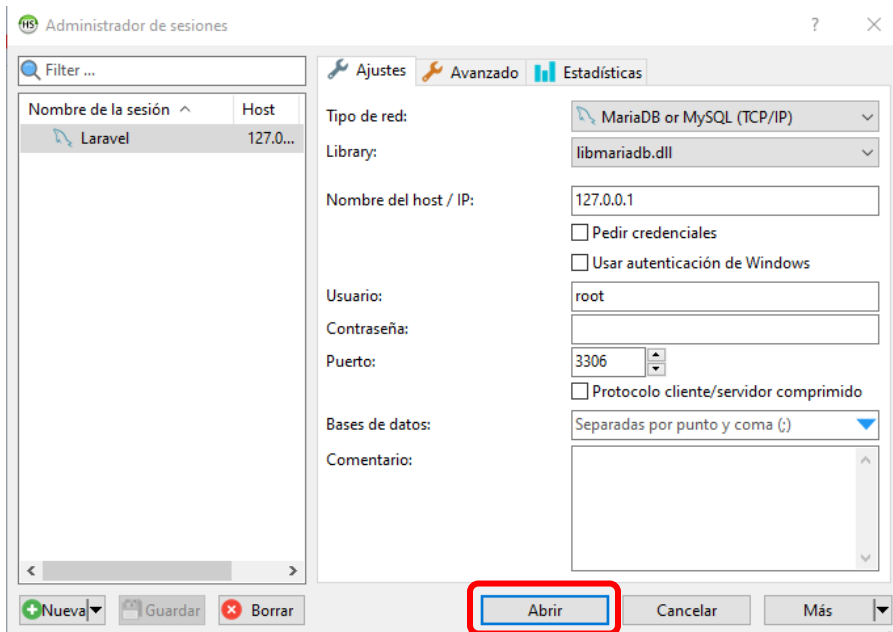
Después daremos clic en **Terminal** y nos posicionamos en la ruta de la carpeta Proyecto.



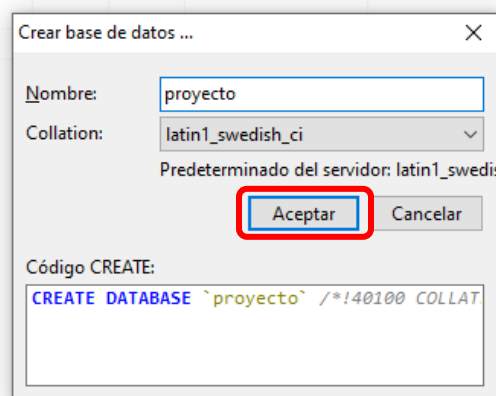
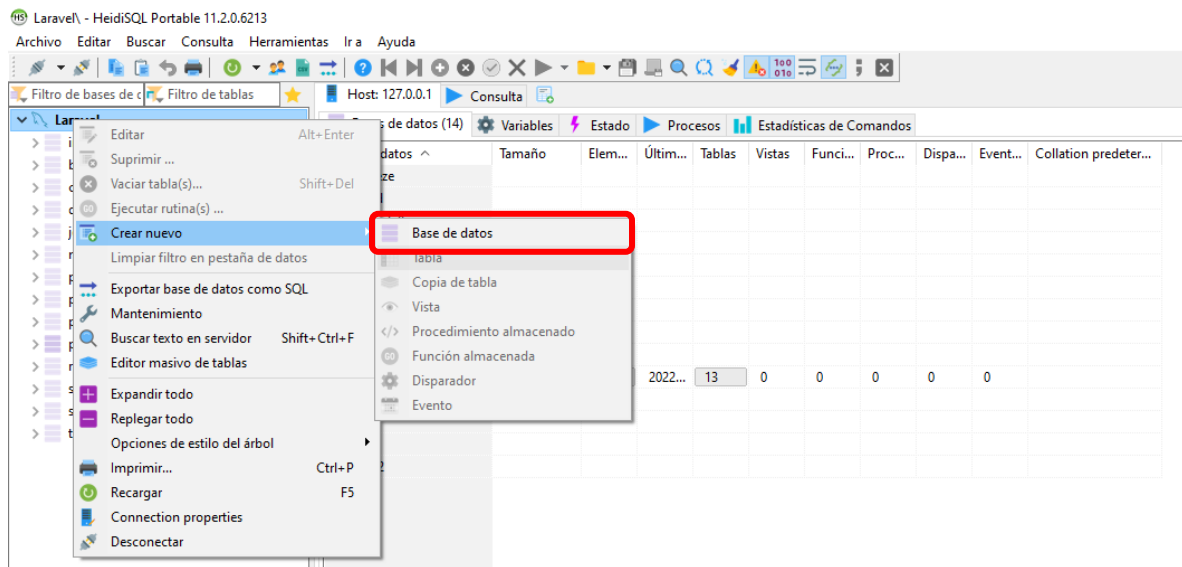
A continuación, nos dirigimos a la ventana de Laragon y hacemos clic en Base de Datos.



Abrirá la siguiente ventana y daremos clic en Abrir.



Después hacemos clic derecho en Laravel y creamos una base de datos llamada proyecto.



Después nos dirigimos a la terminal e ingresamos los comandos composer install, npm install y por último php artisan migrate.

```
C:\laragon\www\avances-proyecto-JocelynLlamas\proyecto (main)
λ composer install|
```

```
C:\laragon\www\avances-proyecto-JocelynLlamas\proyecto (main)
λ npm install|
```

```
C:\laragon\www\avances-proyecto-JocelynLlamas\proyecto (main)
λ php artisan key:generate|
```

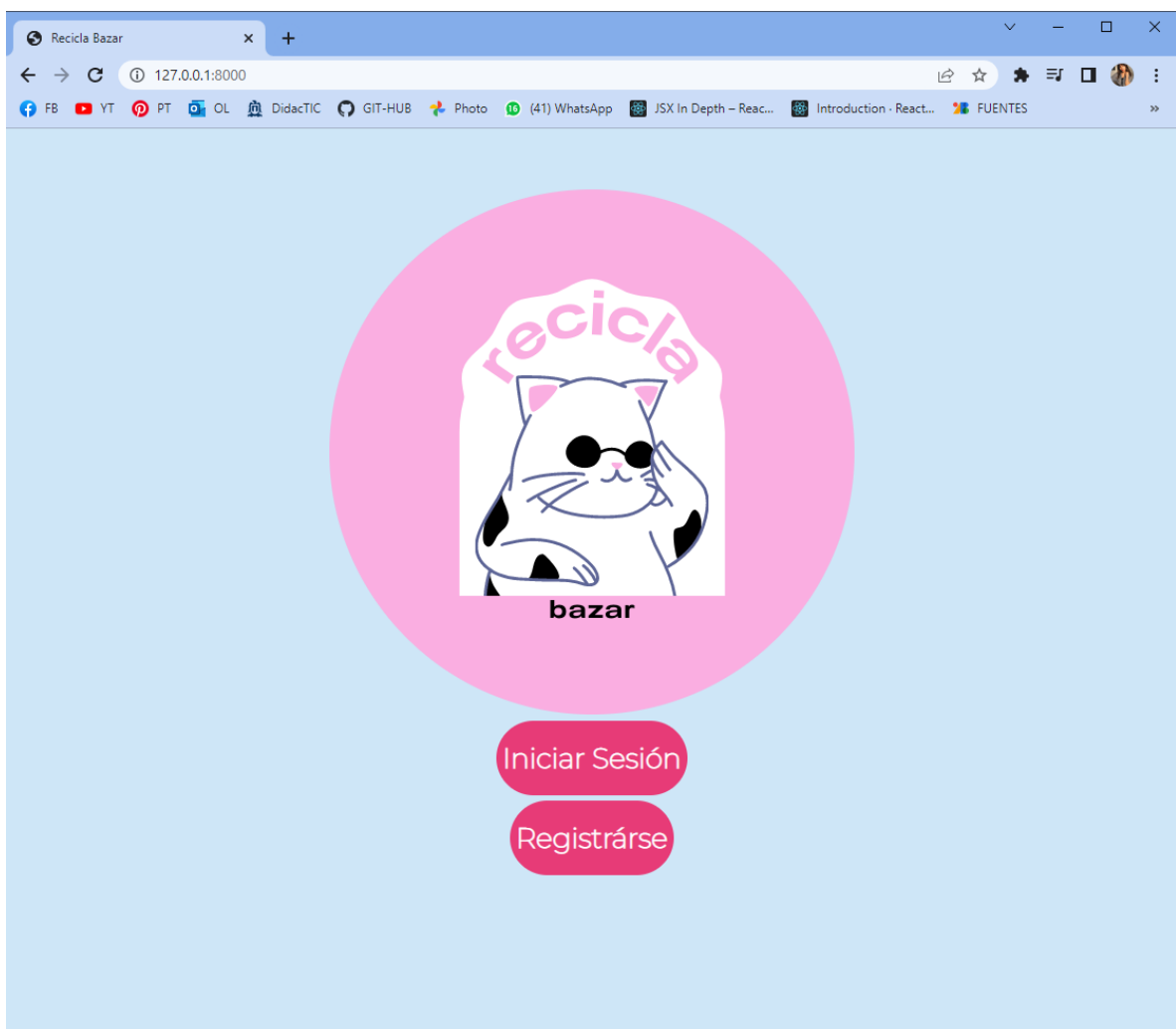
```
C:\laragon\www\avances-proyecto-JocelynLlamas\proyecto (main)
λ php artisan migrate|
```



A continuación, ingresamos el comando `php artisan serve`, copiamos la ruta que nos da y la pegamos en el navegador o simplemente damos `ctrl+clic`.

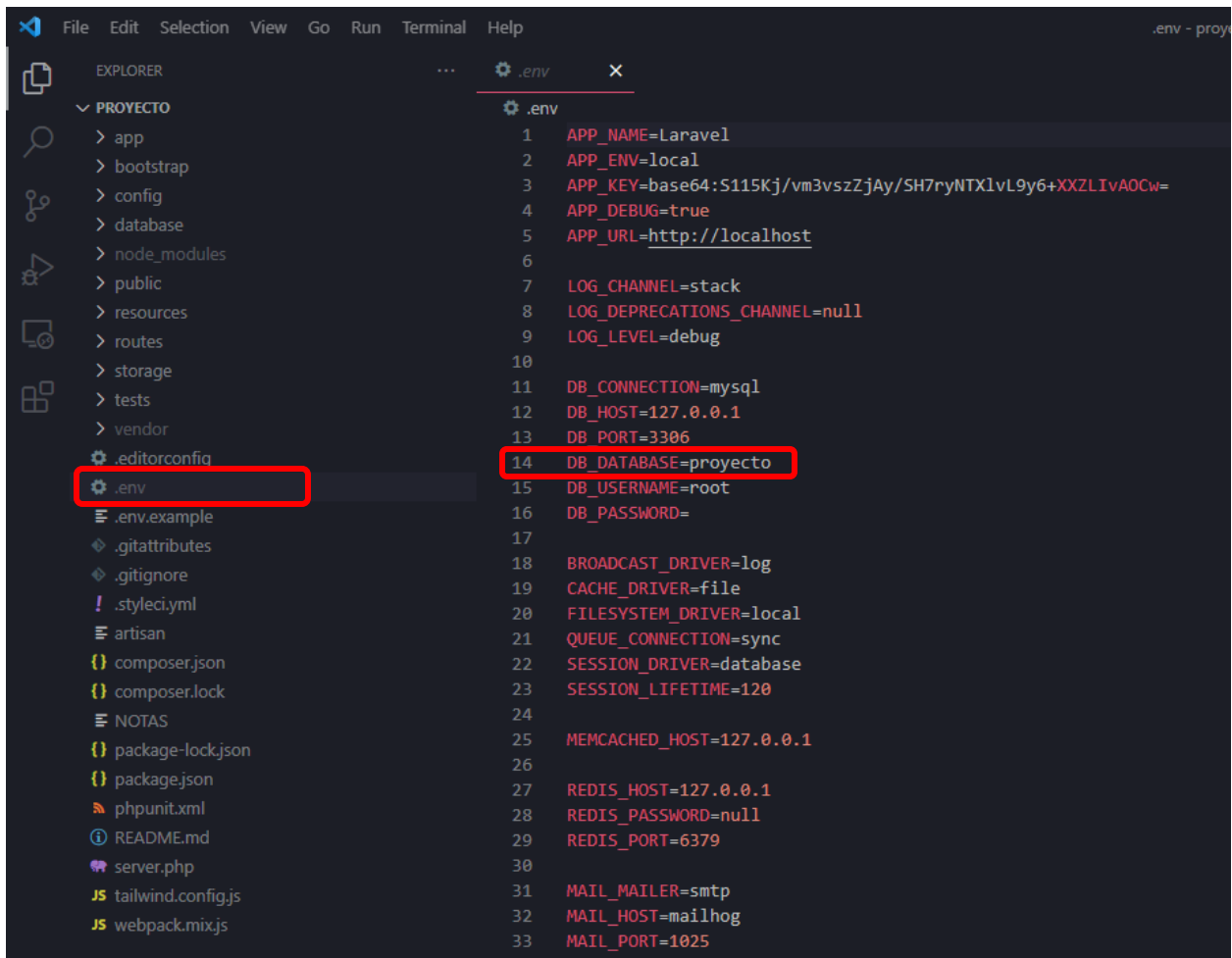
```
C:\laragon\www\avances-proyecto-JocelynLlamas\proyecto (main) (41) WhatsApp JSX In Depth - Re
λ php artisan serve
Starting Laravel development server: http://127.0.0.1:8000
[Sun May 29 20:21:50 2022] PHP 7.4.19 Development Server (http://127.0.0.1:8000) started
```

Una vez hecho esto ya se estará ejecutando el proyecto y lo podremos visualizar en el navegador.



## Desarrollador

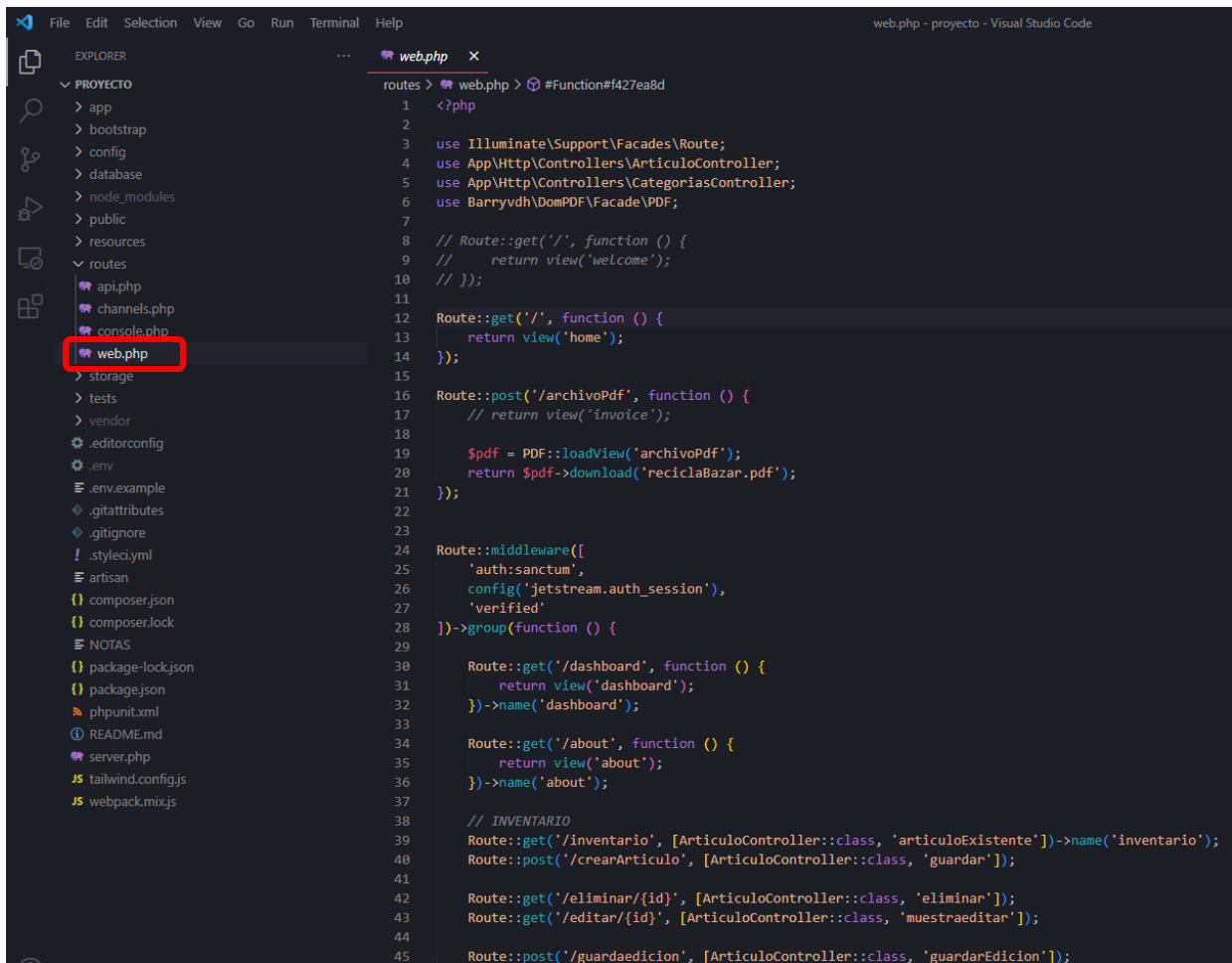
Para poder visualizar o cambiar el nombre de la base de datos dirígete al archivo `.env` y lo podrás visualizar o modificar en la línea 14.



The image shows a screenshot of the Visual Studio Code editor. On the left, the Explorer sidebar shows a project structure with folders like 'app', 'bootstrap', 'config', 'database', 'node\_modules', 'public', 'resources', 'routes', 'storage', 'tests', and 'vendor'. Below these are several files, including '.env', which is highlighted with a red rectangle. The main editor area displays the contents of the '.env' file. The file contains various environment variables for a Laravel application. Line 14, which sets 'DB\_DATABASE=proyecto', is highlighted with a red rectangle. Other visible lines include 'APP\_NAME=Laravel', 'APP\_ENV=local', 'APP\_KEY=base64:S115Kj/vm3vszZjAy/SH7ryNTX1vL9y6+XXZLIvAOCw=', 'APP\_DEBUG=true', 'APP\_URL=http://localhost', 'LOG\_CHANNEL=stack', 'LOG\_DEPRECATIONS\_CHANNEL=null', 'LOG\_LEVEL=debug', 'DB\_CONNECTION=mysql', 'DB\_HOST=127.0.0.1', 'DB\_PORT=3306', 'DB\_USERNAME=root', 'DB\_PASSWORD=', 'BROADCAST\_DRIVER=log', 'CACHE\_DRIVER=file', 'FILESYSTEM\_DRIVER=local', 'QUEUE\_CONNECTION=sync', 'SESSION\_DRIVER=database', 'SESSION\_LIFETIME=120', 'MEMCACHED\_HOST=127.0.0.1', 'REDIS\_HOST=127.0.0.1', 'REDIS\_PASSWORD=null', 'REDIS\_PORT=6379', 'MAIL\_MAILER=smtp', 'MAIL\_HOST=mailhog', and 'MAIL\_PORT=1025'.

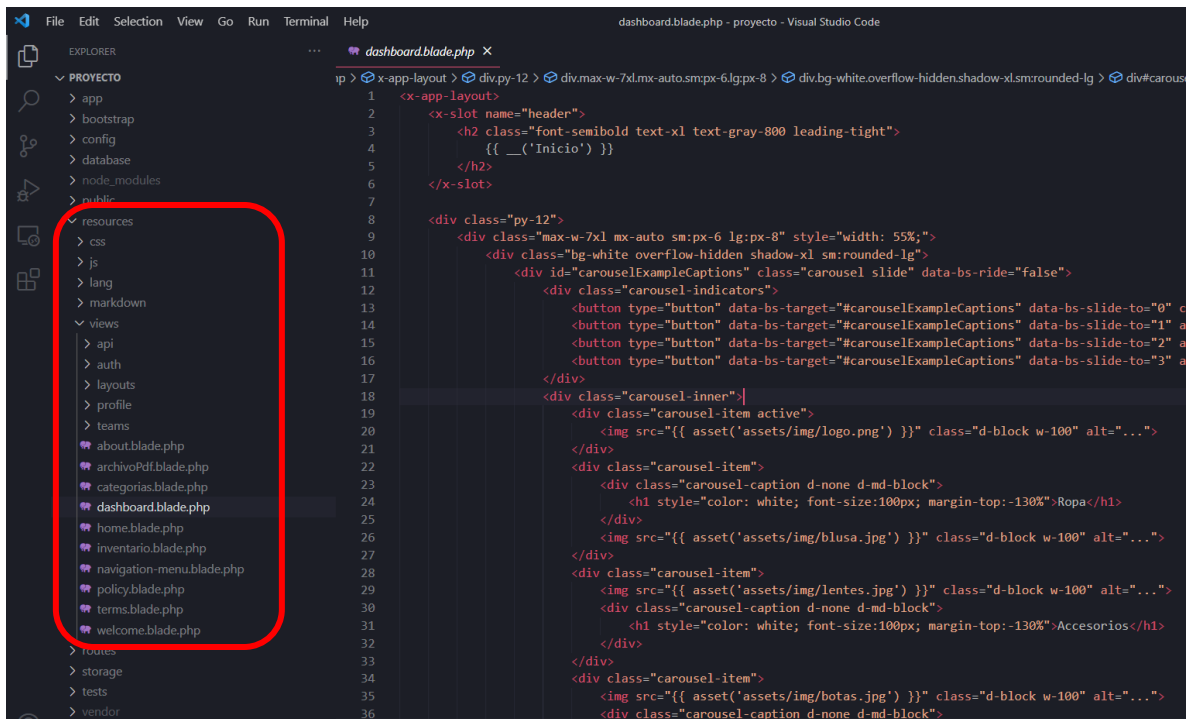
```
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=base64:S115Kj/vm3vszZjAy/SH7ryNTX1vL9y6+XXZLIvAOCw=
4 APP_DEBUG=true
5 APP_URL=http://localhost
6
7 LOG_CHANNEL=stack
8 LOG_DEPRECATIONS_CHANNEL=null
9 LOG_LEVEL=debug
10
11 DB_CONNECTION=mysql
12 DB_HOST=127.0.0.1
13 DB_PORT=3306
14 DB_DATABASE=proyecto
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=database
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=mailhog
33 MAIL_PORT=1025
```

Podrás visualizar o agregar nuevas rutas en el archivo de `web.php` que se encuentra en la carpeta `/routes`.



```
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4 use App\Http\Controllers\ArticuloController;
5 use App\Http\Controllers\CategoriasController;
6 use Barryvdh\DomPDF\Facade\Pdf;
7
8 // Route::get('/', function () {
9 //     return view('welcome');
10 // });
11
12 Route::get('/', function () {
13     return view('home');
14 });
15
16 Route::post('/archivoPdf', function () {
17     // return view('invoice');
18
19     $pdf = Pdf::loadView('archivoPdf');
20     return $pdf->download('reciclaBazar.pdf');
21 });
22
23
24 Route::middleware([
25     'auth:sanctum',
26     config('jetstream.auth_session'),
27     'verified'
28 ])->group(function () {
29
30     Route::get('/dashboard', function () {
31         return view('dashboard');
32     }->name('dashboard'));
33
34     Route::get('/about', function () {
35         return view('about');
36     }->name('about'));
37
38     // INVENTARIO
39     Route::get('/inventario', [ArticuloController::class, 'articuloExistente']->name('inventario'));
40     Route::post('/crearArticulo', [ArticuloController::class, 'guardar']);
41
42     Route::get('/eliminar/{id}', [ArticuloController::class, 'eliminar']);
43     Route::get('/editar/{id}', [ArticuloController::class, 'mostrarEditar']);
44
45     Route::post('/guardaEdicion', [ArticuloController::class, 'guardarEdicion']);
```

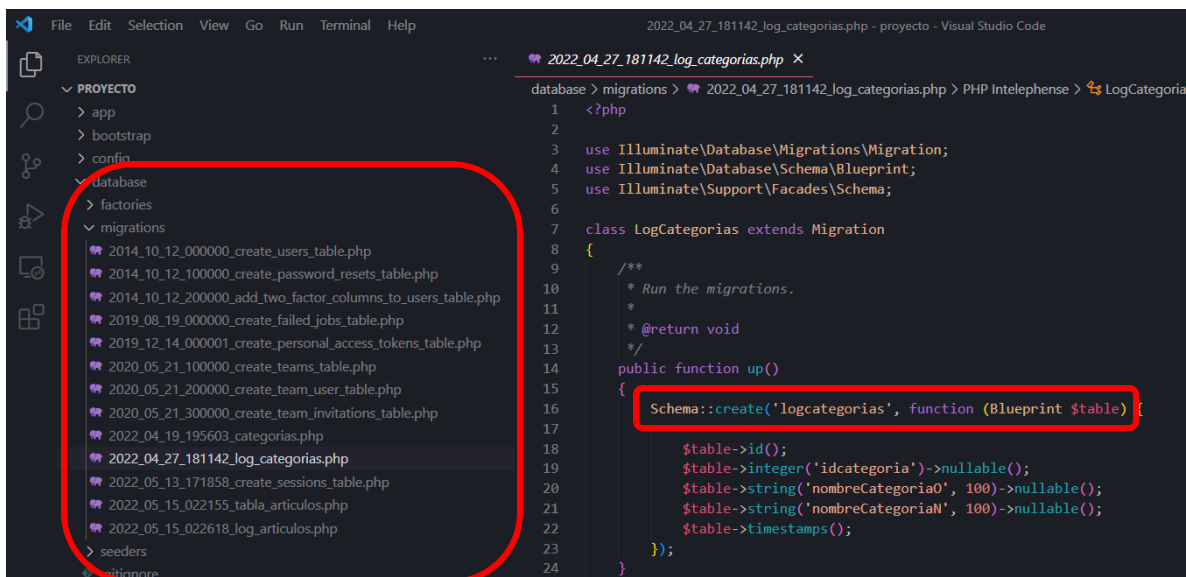
Podrás visualizar o agregar nuevas vistas en la ruta `/resources/views`. Estas ayudaran a ver la estructura de html de las páginas del proyecto, como inicio de sesión, edición de perfil, inventario, inicio, etc.



Para crear una nueva migración, dirígete a la terminal e ingresa el comando `php artisan make:migration 'nombre de la migración'`.

```
C:\laragon\www\avances-proyecto-JocelynLlamas\proyecto (main)
λ php artisan make:migration nuevaMigracion
```

Las migraciones se encuentran en la ruta `/database/migrations`, aquí puedes crear nuevas tablas para la base de datos.



Cuando hayas terminado de hacer alguna tabla en una migración, ve a la terminal e ingresa el comando `php artisan migrate`.

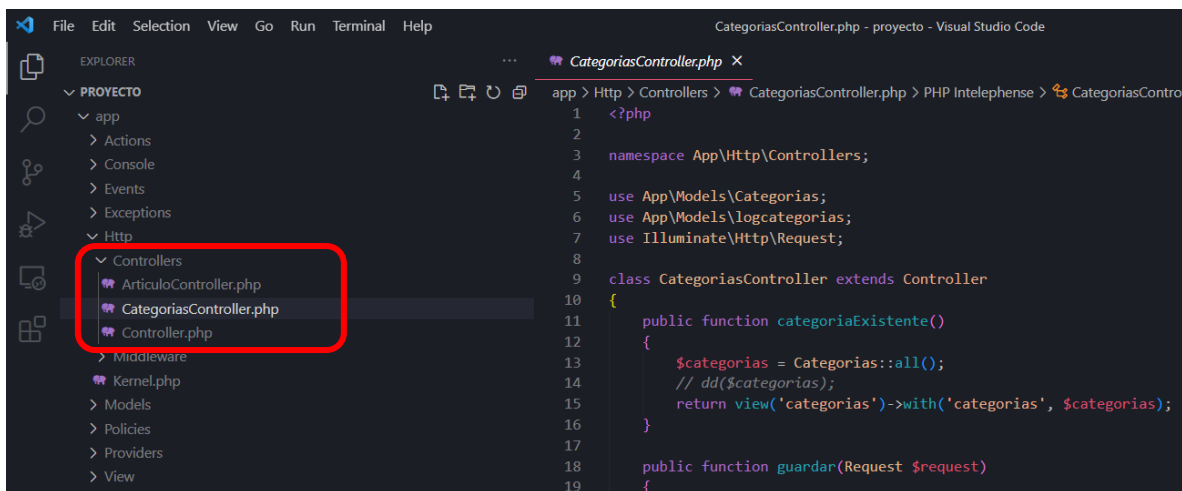
```
C:\laragon\www\avances-proyecto-JocelynLlamas\proyecto (main)
λ php artisan migrate
```

Con los controladores podrás ingresar las funciones necesarias para los cruds como crear información, guardarla, eliminarla, editarla, etc.

Para crear un controlador ingresa el comando `php artisan make:Controller 'nombre del controlador'`.

```
C:\laragon\www\avances-proyecto-JocelynLlamas\proyecto (main)
λ php artisan make:Controller nuevoControlador
```

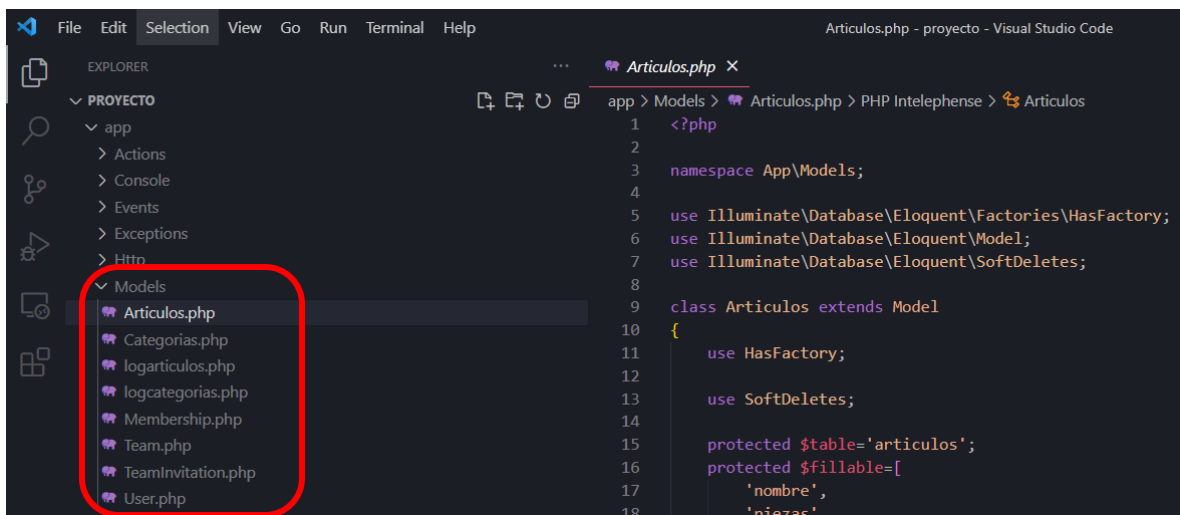
Los controladores se encuentran en la ruta `/app/Http/Controllers`, aquí puedes comenzar a añadir o modificar las funcionalidades del proyecto.



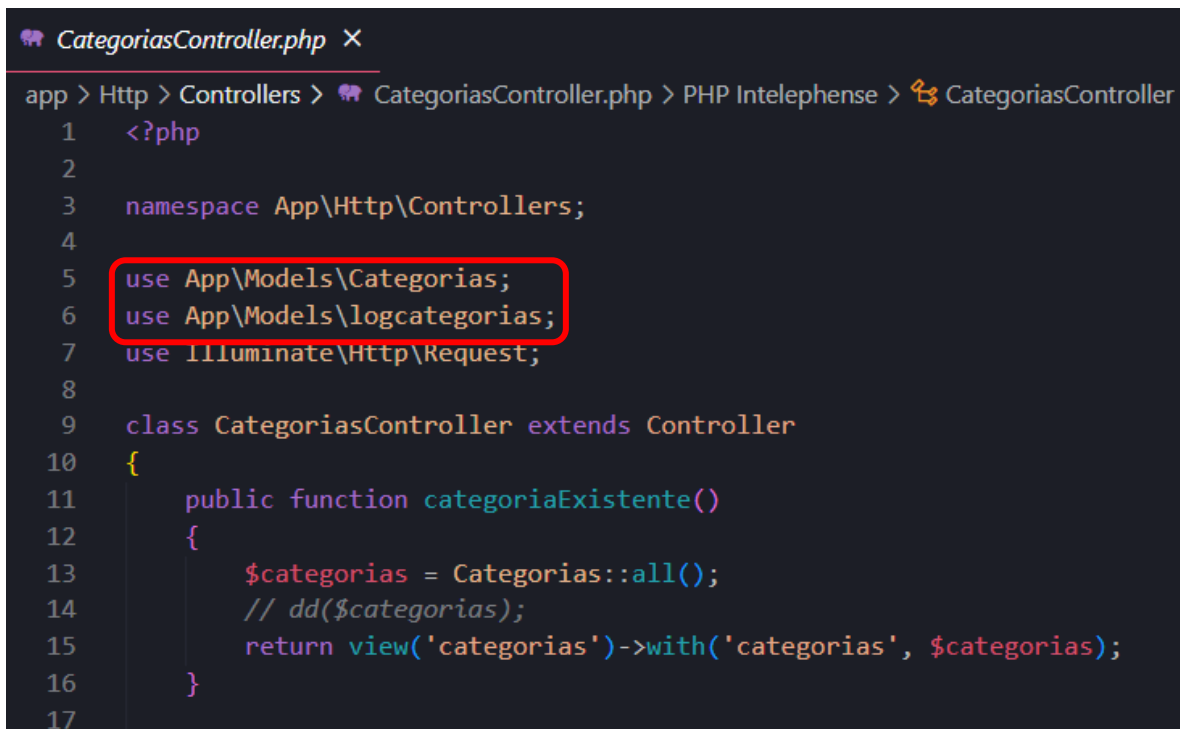
Con los modelos puedes ingresar los datos que tienes en las tablas que creaste con las migraciones. Para crear un Modelo ingresa el comando `php artisan make:Model 'nombre del modelo'`.

```
C:\laragon\www\avances-proyecto-JocelynLlamas\proyecto (main)
λ php artisan make:Model nuevoModelo
```

Los modelos se encuentran en la ruta `/app/Models`, aquí puedes comenzar a añadir o modificar los datos de las tablas.



Cabe recalcar que para hacer uso de los datos de los modelos debes importarlos en los controladores en los que los vayas a usar.



Para ir observando los cambios que se van haciendo ingresa al navegador en la ruta del proyecto y puedes acceder a las rutas que se mencionaron en el archivo de `web.php`.

```
12 Route::get('/', function () {
13     return view('home');
14 });
15
16 Route::post('/archivoPdf', function () {
17     // return view('invoice');
18
19     $pdf = PDF::loadView('archivoPdf');
20     return $pdf->download('reciclaBazar.pdf');
21 });
22
23
24 Route::middleware([
25     'auth:sanctum',
26     config('jetstream.auth_session'),
27     'verified'
28 ]->group(function () {
29
30     Route::get('/dashboard', function () {
31         return view('dashboard');
32     }->name('dashboard');
33
34     Route::get('/about', function () {
35         return view('about');
36     }->name('about');
37
38     // INVENTARIO
39     Route::get('/inventario', [ArticuloController::class, 'articuloExistente']->name('inventario'));
40     Route::post('/crearArticulo', [ArticuloController::class, 'guardar']);
41
42     Route::get('/eliminar/{id}', [ArticuloController::class, 'eliminar']);
43     Route::get('/editar/{id}', [ArticuloController::class, 'muestraeditar']);
44
45     Route::post('/guardaediccion', [ArticuloController::class, 'guardarEdicion']);
46
47 });
```

