



E S P E

ESCUELA POLITÉCNICA DEL EJÉRCITO CAMINO A LA EXCELENCIA

Nombre: Edison Nicolasc Guamialama Haro

NRC: 29692

Instructor: Ing. Vilmer David Criollo

PRY_MICROSRV_AUTENTICACION

Resumen del Proyecto

Este repositorio aloja el **Microservicio de Autenticación**, el componente fundamental para la gestión de acceso en el sistema de publicaciones distribuido. Su diseño sigue los principios de **Arquitectura Orientada a Microservicios (MSA)**, garantizando el desacoplamiento total de la lógica de usuarios y posts.

La funcionalidad principal es proporcionar un **Servidor de Autorización** centralizado que utiliza tokens Sanctum y un control de acceso basado en perfiles (RBAC) a través de un *middleware* personalizado.

Arquitectura Técnica y Componentes Clave

Especificaciones de Implementación

Componente	Detalle Técnico	Propósito MSA
Framework	Laravel 12 (PHP v8.2+)	Base ágil para API RESTful.
Base de Datos	MySQL	Persistencia de datos de usuario.
Autenticación	Laravel Sanctum	Emisión y gestión de <i>Bearer Tokens</i> (Stateless).
Autorización	App\Http\Middleware\CheckUserRole	Valida el <i>perfil (role)</i> del usuario en cada solicitud protegida.

Componente	Detalle Técnico	Propósito MSA
Comunicación	Endpoint <code>/api/validate-token</code>	Permite que otros microservicios deleguen la verificación del token.
Perfil/Rol	Columna <code>perfil</code> en <code>users</code>	Soporta <code>administrador</code> , <code>editor</code> y <code>usuario común</code> .

Guía de Instalación y Despliegue

Prerrequisitos del Entorno

- **PHP:** Versión 8.2 o superior.
- **Composer:** Gestor de dependencias de PHP.
- **MySQL:** Servidor de base de datos.

Pasos Operacionales

Creacion del Proyecto "PRY_MICROSRV_AUTENTICACION"

```
Command Prompt
C:\workspace>new PRY_MICROSRV_AUTENTICACION
'new' is not recognized as an internal or external command,
operable program or batch file.

C:\workspace>laravel new PRY_MICROSRV_AUTENTICACION

Which starter kit would you like to install? [None]:
[none] None
[react] React
[vue] Vue
[livewire] Livewire
> none

Which testing framework do you prefer? [Pest]:
[0] Pest
[1] PHPUnit
>

Creating a "laravel/laravel" project at "./PRY_MICROSRV_AUTENTICACION"
Installing laravel/laravel (v12.10.1)
  - Installing laravel/laravel (v12.10.1): Extracting archive
```

Configuracion la base de datos a utilizar

```
> cp .env.example .env && cp .env.example .env &&
INFO Application key set successfully.

Which database will your application use? [SQLite]:
[sqlite] SQLite
[mysql] MySQL
[mariadb] MariaDB
[pgsql] PostgreSQL (Missing PDO extension)
[sqlsrv] SQL Server (Missing PDO extension)
> mysql

Default database updated. Would you like to run the default database migrations? (yes/no) [yes]:
> no

./composer.json has been updated
  Cannot use pestphp/pest's latest version v4.1.3 as it requires php ^8.3.0 which is not satisfied
  Using version ^3.8 for pestphp/pest
```

check instalacion exitosa

```
Would you like to run npm install and npm run build? (yes/no) [yes]:
> yes

added 83 packages, and audited 84 packages in 22s

21 packages are looking for funding
  run `npm fund` for details

Found 0 vulnerabilities

> build
> vite build

  vite v7.2.2 building client environment for production...
transforming...
  ✓ 53 modules transformed.
rendering chunks...
computing gzip size...
public/build/manifest.json      0.33 kB | gzip:  0.17 kB
public/build/assets/app-iJWsMRAy.css 33.28 kB | gzip:  8.43 kB
public/build/assets/app-CAiCLEjY.js  36.35 kB | gzip: 14.71 kB
/ built in 1.63s
INFO Application ready in [PRY_MICROSRV_AUTENTICACION]. You can start your local development using:
→ cd PRY_MICROSRV_AUTENTICACION
→ composer run dev
```

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=db_users
DB_USERNAME=db_users
DB_PASSWORD=db_users
```

configuracion de .env

Entorno de desarrollo IP maquina

```
Nicolas@Nico MINGW64 /c/workspace/PRY_MICROSRV_AUTENTICACION
$ ipconfig
  Connection-specific DNS Suffix . :

Wireless LAN adapter Wi-Fi:

  Connection-specific DNS Suffix . :
  Link-local IPv6 Address . . . . . : fe80::ad43:c280:1fab:9abd%27
  IPv4 Address. . . . . : 192.168.0.203
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . . : 192.168.0.1

Ethernet adapter VMware Network Adapter VMnet1:

  Connection-specific DNS Suffix . :
```

1. Instalar Dependencias:

```
composer install
```

2. Configurar Variables de Entorno:

```
cp .env.example .env
php artisan key:generate
# EDITE las variables de base de datos y la clave APP_ID para la conexión
con otros microservicios.
```

3. Migrar y Seeders:

```
php artisan migrate --seed
```

4. Iniciar Servidor:

```
php artisan serve
```

El servicio estará accesible en <http://127.0.0.1:8000>.

```
Nicolas@Nico MINGW64 /c/workspace/PRY_MICROSRV_AUTENTICACION
$ php artisan make:middleware CheckUserRole
[INFO] Middleware [C:\workspace\PRY_MICROSRV_AUTENTICACION\app\Http\Middleware\CheckUserRole.php] created successfully.

Nicolas@Nico MINGW64 /c/workspace/PRY_MICROSRV_AUTENTICACION
```

creacion de nuevo usuario en Xampp <http://localhost:8012/phpmyadmin/>

Ha agregado un nuevo usuario.

```
CREATE_USER 'db_users' '@localhost' IDENTIFIED VIA mysql_native_password USING '****';GRANT ALL PRIVILEGES ON *.* TO 'db_users'@'localhost' REQUIRE NONE WITH GRANT OPTION MAX_QUERIES_PER_HOUR 0 MAX_CONNECTIONS_PER_HOUR 0 MAX_UPDATES_PER_HOUR 0 MAX_USER_CONNECTIONS 0;CREATE DATABASE IF NOT EXISTS `db_users`;GRANT ALL PRIVILEGES ON `db_users`.* TO 'db_users'@'localhost';
```

[Editar en línea] [Editar] [Crear código PHP]

Global Base de datos Change password Información de la cuenta

Editar los privilegios: Cuenta de usuario 'db_users'@'localhost'

Privilegios globales Seleccionar todo

Nota: Los nombres de los privilegios de MySQL están expresados en inglés.

<input checked="" type="checkbox"/> Datos	<input checked="" type="checkbox"/> Estructura	<input checked="" type="checkbox"/> Administración	Limites de recursos
<input checked="" type="checkbox"/> SELECT <input checked="" type="checkbox"/> INSERT <input checked="" type="checkbox"/> UPDATE <input checked="" type="checkbox"/> DELETE <input checked="" type="checkbox"/> FILE	<input checked="" type="checkbox"/> CREATE <input checked="" type="checkbox"/> ALTER <input checked="" type="checkbox"/> INDEX <input checked="" type="checkbox"/> DROP <input checked="" type="checkbox"/> CREATE TEMPORARY TABLES <input checked="" type="checkbox"/> SHOW VIEW <input checked="" type="checkbox"/> CREATE ROUTINE <input checked="" type="checkbox"/> ALTER ROUTINE <input checked="" type="checkbox"/> EXECUTE <input checked="" type="checkbox"/> CREATE VIEW <input checked="" type="checkbox"/> EVENT <input checked="" type="checkbox"/> TRIGGER	<input checked="" type="checkbox"/> GRANT <input checked="" type="checkbox"/> SUPER <input checked="" type="checkbox"/> PROCESS <input checked="" type="checkbox"/> RELOAD <input checked="" type="checkbox"/> SHUTDOWN <input checked="" type="checkbox"/> SHOW DATABASES <input checked="" type="checkbox"/> LOCK TABLES <input checked="" type="checkbox"/> REFERENCES <input checked="" type="checkbox"/> REPLICATION CLIENT <input checked="" type="checkbox"/> REPLICATION SLAVE <input checked="" type="checkbox"/> CREATE USER	<small>Nota: si cambia los parámetros de estas opciones a 0 (cero), remueve el límite.</small> MAX QUERIES PER HOUR 0 MAX UPDATES PER HOUR 0 MAX CONNECTIONS PER HOUR 0 MAX USER CONNECTIONS 0

SSL

REQUIRE NONE
 REQUIRE SSL
 REQUIRE X509

phpMyAdmin

Servidor: localhost Base de datos: db_users

Estructura (actual) SQL Buscar Generar una consulta

⚠ No se han encontrado tablas en la base de datos.

Crear nueva tabla

Nombre de la tabla

Número de columnas

4

Crear

creacion de perfil y migracion

```

UserFactory.php
migrations C:\workspace\PRY_MICROSRV_AUTENTICACION\database\factories\UserFactory.php
  0001_01_01_000000_create_users_table.php
  0001_01_01_000001_create_cache_table.php
  0001_01_01_000002_create_jobs_table.php
  2025_11_18_151853_create_personal_access_tokens_table.php
  2025_11_18_174300_add_profile_to_users_table.php
seeders
  migrations

```

creacion de usuario

```

use Illuminate\Http\Request;
use Illuminate\Support\Facades\Route;

Route::post('/tokens/create', function (Request $request) {

    // search user by email enguamialama@espe.edu.ec

    $user = User::where('email', 'enguamialama@espe.edu.ec')->first();    Undefined type 'User'.

    $token = $request->user()->createToken($request->token_name);

    return ['token' => $token->plainTextToken];
})->middleware('auth:sanctum');

Route::get('/user', function (Request $request) {
    return $request->user();
})->middleware('auth:sanctum');

```

levantar el servicio en la direccion local

```

Nicolas@Nico MINGW64 /c/workspace/PRY_MICROSRV_AUTENTICACION
$ php artisan serve --host=192.168.0.203

INFO Server running on [http://192.168.0.203:8000].
Press Ctrl+C to stop the server

2025-11-18 09:40:37 /api/tokens/create ..... .
2025-11-18 09:53:09 /api/tokens/create ..... .

```

[] In 24 Col 6 Spacers: 4

🔑 5. Endpoints Principales

Base URL: <http://127.0.0.1:8000/api>

Método	Endpoint	Headers	Payload (JSON)	Respuesta
--------	----------	---------	----------------	-----------

Método	Endpoint	Headers	Payload (JSON)	Respuesta
POST	/register	Content-Type: application/json	{name, email, password, password_confirmation, rol}	201 + {token}
POST	/login	N/A	{email, password}	200 + {token}
POST	/logout	Authorization: Bearer	N/A	200 {message}
GET	/user	Authorization: Bearer	N/A	200 {user_data}
GET	/validate-token	Authorization: Bearer	N/A	200 OK o 401 Unauthorized

Evidencias de las practicas

Creacion de administrador

The screenshot shows a Postman request to `http://192.168.0.203:8000/api/register`. The request method is `POST`. The body is a JSON object:

```

1  {
2    "name": "Admin User",
3    "email": "admin@test.com",
4    "password": "password123",
5    "password_confirmation": "password123",
6    "perfil": "administrador"
7  }

```

The response status is `201 Created`, with a response time of `367 ms` and a response size of `425 B`. The response body is:

```

1  {
2    "message": "Registro exitoso. Token generado.",
3    "user": {
4      "id": 13,
5      "name": "Admin User",
6      "email": "admin@test.com",
7      "perfil": "administrador"
8    },
9    "token": "610pJEHVrqNfLfqpQN4CI4akgid5jgatkhwh1ze9g6434d949e"
10 }

```

token

POST http://192.168.0.203:8000/api/register

Body (raw) JSON

```

1 {
2   "name": "Admin User",
3   "email": "admin@test.com",
4   "password": "password123",
5   "password_confirmation": "password123",
6   "perfil": "administrador"
7 }

```

201 Created • 367 ms • 425 B

```

1 {
2   "message": "Registro exitoso. Token generado.",
3   "user": {
4     "id": 13,
5     "name": "Admin User",
6     "email": "admin@test.com",
7     "perfil": "administrador"
8   },
9   "token": "6|0pJEHVrqNfLfqpQN4CI4akgid5jgatkwh1ze9g6434d949e"
10 }

```

verificacion

POST http://192.168.0.203:8000/api/tokens/create

Body (x-www-form-urlencoded)

Key	Value	Description	Bulk Edit
<input checked="" type="checkbox"/> token_name	accesoalsistema		
Key	Value	Description	

200 OK • 265 ms • 287 B

```

1 {
2   "token": "3|61YPw0Ldj56pZ8HtQzfLGASdre0UC782srR6epnw3bf3a5e3"
3 }

```

verificacion 2

The screenshot shows a Postman interface for a GET request to `http://192.168.0.203:8000/api/user`. The 'Authorization' tab is selected, showing 'Bearer Token' as the type. A token placeholder is visible. The 'Body' tab shows a JSON response with the following data:

```
1 {  
2   "id": 11,  
3   "name": "NICOLAS GUAMIALAMA",  
4   "email": "enguamialama@espe.edu.ec",  
5   "email_verified_at": "2025-11-18T15:09:24.000000Z",  
6   "created_at": "2025-11-18T15:09:24.000000Z",  
7   "updated_at": "2025-11-18T15:09:24.000000Z"  
8 }
```

The status bar indicates a 200 OK response with 212 ms latency and 433 B size.

registro de segundo usuario

POST **Send**

Docs Params Authorization Headers (10) **Body** Scripts Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL

Key	Value	Description	...	Bulk Edit
<input checked="" type="checkbox"/> email	eguami@ejemplo.com			
<input checked="" type="checkbox"/> password	password12345			
<input checked="" type="checkbox"/>				
Key	Value	Description		

Body Cookies (2) Headers (7) Test Results

200 OK • 357 ms • 439 B •

{ } JSON Preview Visualize

```

1  {
2      "message": "Inicio de sesión exitoso. Token generado.",
3      "user": {
4          "id": 12,
5          "name": "Edison Guami",
6          "email": "eguami@ejemplo.com",
7          "perfil": "administrador"
8      },
9      "token": "5|0pKVsqChR1YZ0QaTvfRHNPtowspjmGdiIIzp01Mcb68c00b0"
10 }

```

HTTP

POST **Send**

Docs Params Authorization Headers (9) **Body** Scripts Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL

Key	Value	Description	...	Bulk Edit
<input checked="" type="checkbox"/> name	Edison Guami	nuevo usuario		
<input checked="" type="checkbox"/> email	eguami@ejemplo.com	email unico		
<input checked="" type="checkbox"/> password	password12345	fuerte		
<input checked="" type="checkbox"/> password_confirmation	password12345	confirmacion		
<input checked="" type="checkbox"/> perfil	administrador	se usa Administrador tambien puede ser editor ...		
Key	Value	Description		

Body Cookies Headers (7) Test Results

201 Created • 394 ms • 431 B •

{ } JSON Preview Visualize

```

1  {
2      "message": "Registro exitoso. Token generado.",
3      "user": {
4          "id": 12,
5          "name": "Edison Guami",
6          "email": "eguami@ejemplo.com",
7          "perfil": "administrador"
8      },
9      "token": "4|8ocpvRED4WxSEY38e65yaBzuPXi1HE0qmz7JIwIJ425e4f73"
10 }

```

tercer usuario comun

POST http://192.168.0.203:8000/api/register

Body

```

1 {
2   "name": "Common User",
3   "email": "user@test.com",
4   "password": "password123",
5   "password_confirmation": "password123"
6   // 'perfil' es opcional, por defecto es 'usuario común'
7 }

```

Response Headers:

- 201 Created
- 377 ms
- 430 B
- ...

Response Body:

```

1 {
2   "message": "Registro exitoso. Token generado.",
3   "user": {
4     "id": 14,
5     "name": "Common User",
6     "email": "user@test.com",
7     "perfil": "usuario común"
8   },
9   "token": "7|F5ZmW0BJ0kqd3usa6EPdGyCV34fkt08wtSedHhEd56455f96"
10 }

```

confirmacion en la base de datos (db_user) las creaciones de los usuarios

Opciones extra	id	name	email	email_verified_at	password	perfil	remen
Editar Copiar Borrar	1	Isabell Skiles	kihn.talon@example.net	2025-11-18 15:09:23	\$2y\$12\$X7ZNK.0YPQ6LREONFswq1OtJcB46AD5c/NbfGSzISjf...	usuario común	Kiz49y
Editar Copiar Borrar	2	Bailey Kris	ulindgren@example.net	2025-11-18 15:09:24	\$2y\$12\$X7ZNK.0YPQ6LREONFswq1OtJcB46AD5c/NbfGSzISjf...	usuario común	EVARh
Editar Copiar Borrar	3	Oren Lang DDS	mohr.katelyn@example.com	2025-11-18 15:09:24	\$2y\$12\$X7ZNK.0YPQ6LREONFswq1OtJcB46AD5c/NbfGSzISjf...	usuario común	OVUD
Editar Copiar Borrar	4	Dr. Zane Collins	tyrei84@example.org	2025-11-18 15:09:24	\$2y\$12\$X7ZNK.0YPQ6LREONFswq1OtJcB46AD5c/NbfGSzISjf...	usuario común	u9dtcj
Editar Copiar Borrar	5	Margarett Nienow	wswaniawski@example.org	2025-11-18 15:09:24	\$2y\$12\$X7ZNK.0YPQ6LREONFswq1OtJcB46AD5c/NbfGSzISjf...	usuario común	5kH1I
Editar Copiar Borrar	6	Ms. Verde Hyatt	mertie.ankunding@example.net	2025-11-18 15:09:24	\$2y\$12\$X7ZNK.0YPQ6LREONFswq1OtJcB46AD5c/NbfGSzISjf...	usuario común	RJCX
Editar Copiar Borrar	7	Lulu Steuber	rgoyette@example.net	2025-11-18 15:09:24	\$2y\$12\$X7ZNK.0YPQ6LREONFswq1OtJcB46AD5c/NbfGSzISjf...	usuario común	qqO1v
Editar Copiar Borrar	8	Wilford Rau	ratke.eloise@example.com	2025-11-18 15:09:24	\$2y\$12\$X7ZNK.0YPQ6LREONFswq1OtJcB46AD5c/NbfGSzISjf...	usuario común	vmWD
Editar Copiar Borrar	9	Mona Lueilwitz	bert44@example.org	2025-11-18 15:09:24	\$2y\$12\$X7ZNK.0YPQ6LREONFswq1OtJcB46AD5c/NbfGSzISjf...	usuario común	YlaU4
Editar Copiar Borrar	10	Shane Murazik	upton.jayde@example.net	2025-11-18 15:09:24	\$2y\$12\$X7ZNK.0YPQ6LREONFswq1OtJcB46AD5c/NbfGSzISjf...	usuario común	cAzQ
Editar Copiar Borrar	11	NICOLAS GUAMIALAMA	enguamialama@espe.edu.ec	2025-11-18 15:09:24	\$2y\$12\$X7ZNK.0YPQ6LREONFswq1OtJcB46AD5c/NbfGSzISjf...	usuario común	gW7W
Editar Copiar Borrar	12	Edison Guami	eguami@ejemplo.com	NULL	\$2y\$12\$EhNICCFwOVcG6bHG8ygL/eoNb6wlwAhGhwH/pYvhU... administrador	administrador	NULL

6. Seguridad y Pruebas

Características de Seguridad

- **Contraseñas hasheadas** con `bcrypt` (10 rounds).
- **Tokens de acceso** de 40 caracteres, almacenados hasheados (`sha256`).
- **Middleware auth:sanctum** protege rutas sensibles.
- **Middleware personalizado role** restringe acciones por perfil.
- Sin almacenamiento de sesiones en servidor (stateless).

```
Nicolas@Nico MINGW64 /c/workspace/PRY_MICROSRV_AUTENTICACION
● $ php artisan test

    PASS Tests\Unit\ExampleTest
    ✓ that true is true

    PASS Tests\Feature\ExampleTest
    ✓ the application returns a successful response

    PASS Tests\Feature\RoleSecurityTest
    ✓ solo administrador puede eliminar articulo
    ✓ solo admin y editor pueden crear articulo
    ✓ solo administrador puede acceder a ruta admin

Tests:    5 passed (19 assertions)
Duration: 0.66s
```

Ejecución de Pruebas

Para ejecutar las pruebas que validan el middleware de roles (`CheckUserRole`), use:

```
php artisan test
```

```
**Clonar y Acceder:**  
```bash  
git clone [https://github.com/ENiko-Gh/PRY_MICROSRV_AUTENTICACION.git
cd pry-microserv-autenticacion
```

## ## About Laravel

Laravel is a web application framework with expressive, elegant syntax. We believe development must be an enjoyable and creative experience to be truly fulfilling. Laravel takes the pain out of development by easing common tasks used in many web projects, such as:

- [Simple, fast routing engine](<https://laravel.com/docs/routing>).
- [Powerful dependency injection container](<https://laravel.com/docs/container>).
- Multiple back-ends for [session](<https://laravel.com/docs/session>) and [cache](<https://laravel.com/docs/cache>) storage.
- Expressive, intuitive [database ORM](<https://laravel.com/docs/eloquent>).
- Database agnostic [schema migrations](<https://laravel.com/docs/migrations>).
- [Robust background job processing](<https://laravel.com/docs/queues>).
- [Real-time event broadcasting](<https://laravel.com/docs/broadcasting>).

Laravel is accessible, powerful, and provides tools required for large, robust applications.

## ## Learning Laravel

Laravel has the most extensive and thorough [documentation](<https://laravel.com/docs>) and video tutorial library of all modern web application frameworks, making it a breeze to get started with the framework. You can also check out [Laravel Learn](<https://laravel.com/learn>), where you will be guided through building a modern Laravel application.

If you don't feel like reading, [Laracasts](<https://laracasts.com>) can help. Laracasts contains thousands of video tutorials on a range of topics including Laravel, modern PHP, unit testing, and JavaScript. Boost your skills by digging into our comprehensive video library.

## ## Laravel Sponsors

We would like to extend our thanks to the following sponsors for funding Laravel development. If you are interested in becoming a sponsor, please visit the [Laravel Partners program](<https://partners.laravel.com>).

### ### Premium Partners

- \*\*[Vehikl](<https://vehikl.com>)\*\*
- \*\*[Tighten Co.](<https://tighten.co>)\*\*
- \*\*[Kirschbaum Development Group](<https://kirschbaumdevelopment.com>)\*\*
- \*\*[64 Robots](<https://64robots.com>)\*\*
- \*\*[Curotec](<https://www.curotec.com/services/technologies/laravel>)\*\*
- \*\*[DevSquad](<https://devsquad.com/hire-laravel-developers>)\*\*
- \*\*[Redberry](<https://redberry.international/laravel-development>)\*\*
- \*\*[Active Logic](<https://activelogic.com>)\*\*

## ## Contributing

Thank you for considering contributing to the Laravel framework! The contribution guide can be found in the [Laravel documentation]

(<https://laravel.com/docs/contributions>).

## ## Code of Conduct

In order to ensure that the Laravel community is welcoming to all, please review and abide by the [Code of Conduct] (<https://laravel.com/docs/contributions#code-of-conduct>).

## ## Security Vulnerabilities

If you discover a security vulnerability within Laravel, please send an e-mail to Taylor Otwell via [[taylor@laravel.com](mailto:taylor@laravel.com)] (<mailto:taylor@laravel.com>). All security vulnerabilities will be promptly addressed.

## ## License

The Laravel framework is open-sourced software licensed under the [MIT license] (<https://opensource.org/licenses/MIT>).