

Universidad Estatal de Bolívar (UEB)

Docente

Ing. Maricela Espín

Estudiante

Yahir Anthony Sacan Caiza

Materia

Aplicaciones Distribuidas

PLANIFICACIÓN PROYECTO LARAVEL - SISTEMA DE GESTIÓN DE AULAS

API RESTful para Gestión de Aulas Universitarias

Estudiante: Yahir Sacan

Framework: Laravel

Período: 3 de noviembre - 8 de diciembre de 2025

Proyecto: Sistema de Gestión de Aulas - Facultad Universitaria

DESCRIPCIÓN DEL PROYECTO

Sistema RESTful para la gestión integral de aulas universitarias que permite:

- **Decano de Facultad:** Control total sobre aulas, asignaciones, usuarios y reportes
- **Coordinadores de Carrera:** Gestión de aulas de su carrera, asignaciones y horarios
- **Profesores:** Consulta de aulas asignadas, reservas y registro de quejas/peticiones

CRONOGRAMA DETALLADO CON FECHAS

SEMANA	PERÍODO	FECHA PRESENTACIÓN	ACTIVIDADES	PRODUCTOS A PRESENTAR
SEMANA 11: PLANIFICACIÓN	3-7 Nov 2025	14 de noviembre de 2025	<ul style="list-style-type: none">• Presentación de la planificación completa del proyecto• Definición de la estructura de la API RESTful• Definición de tablas de base de datos• Diseño de roles y permisos• Definición de reglas de negocio	<ul style="list-style-type: none">• Documento: Sacan_Laravel-part1.doc con planificación completa de 4 semanas• Diagrama Entidad-Relación• Matriz de roles y permisos

SEMANA 12: INSTALACIÓN Y CONFIGURACIÓ N	10-14 Nov 2025	14 de noviembre de 2025	<ul style="list-style-type: none"> • Instalación de Laravel y configuración del entorno • Configuración de base de datos MySQL • Creación de migraciones para todas las tablas • Creación de modelos Eloquent con relaciones • Seeders para datos iniciales (roles, usuarios de prueba) • Implementación de primeros endpoints GET 	<ul style="list-style-type: none"> • php artisan --version ejecutado • Estructura del proyecto Laravel • Archivo .env configurado • Base de datos con todas las tablas creadas • Código de todas las migraciones • Código de todos los modelos • Seeders ejecutados • Postman probando GET /api/classrooms • Postman probando GET /api/users
SEMANA 13: IMPLEMENTACI ÓN CRUD RESTful	17-21 Nov 2025	21 de noviembre de 2025	<ul style="list-style-type: none"> • Creación de controladores API para todas las entidades • Configuración completa de rutas en routes/api.php • Implementación de métodos REST: index, store, show, update, destroy • Creación de Form Requests para validaciones complejas • Creación de API Resources para formatear respuestas JSON • Implementación de filtros y búsquedas • Manejo de errores y excepciones 	<ul style="list-style-type: none"> • Archivo routes/api.php completo • ClassroomController con todos los métodos CRUD • AssignmentController con métodos CRUD • ComplaintController con métodos CRUD • Form Requests con reglas de validación • API Resources para cada entidad • Postman probando POST, PUT, DELETE para aulas • Postman probando POST, PUT, DELETE para asignaciones • Respuestas JSON correctamente formateadas • Manejo de errores de validación (422)

				<ul style="list-style-type: none"> • Manejo de errores 404 y 500
SEMANA 14: AUTENTICACIÓN Y AUTORIZACIÓN	24-28 Dic 2025	28 de noviembre de 2025	<ul style="list-style-type: none"> • Instalación y configuración de Laravel Sanctum • Creación de sistema de autenticación completo • Implementación de roles (Decano, Coordinador, Profesor) • Protección de rutas con middleware auth:sanctum • Implementación de Policies para autorización • Middleware personalizado para roles • Implementación de relaciones entre modelos • Pruebas de endpoints con diferentes roles 	<ul style="list-style-type: none"> • AuthController con register, login, logout • Middleware de roles implementado • Policies para Classroom, Assignment, Complaint • Postman probando login y generación de token • Postman probando endpoint sin token (error 401) • Postman probando endpoint con token pero sin permisos (error 403) • Postman probando endpoint con rol autorizado (éxito) • Código de relaciones en modelos (belongsTo, hasMany, belongsToMany) • JSON con datos relacionados (eager loading) • Tabla de pruebas por rol documentada
SEMANA 15: FUNCIONALIDADES AVANZADAS Y DOCUMENTACIÓN	1-5 Dic 2025	5 de diciembre de 2025	<ul style="list-style-type: none"> • Implementación de búsqueda de aulas (por nombre, capacidad, edificio) • Implementación de filtros (por carrera, disponibilidad, estado) • Implementación de paginación de resultados • Dashboard de estadísticas para Decano 	<ul style="list-style-type: none"> • Búsqueda de aulas funcionando con múltiples criterios • Filtros por carrera, estado, disponibilidad • Paginación de resultados en todos los listados • Endpoint de estadísticas para Decano • Endpoint de reportes de ocupación • Validación de conflictos de horarios

			<ul style="list-style-type: none"> • Reportes de ocupación de aulas • Sistema de notificaciones para quejas/peticiones • Validación de conflictos de horarios • Creación de documentación completa • Exportación de colección Postman • Pruebas finales de todos los endpoints 	<ul style="list-style-type: none"> funcionando • README.md con documentación completa • Colección Postman exportada (.json) con todas las pruebas • Diagrama de base de datos actualizado • Código fuente comprimido (.zip) • Manual de usuario por rol • Todos los endpoints RESTful funcionando
--	--	--	--	--

RECURSOS DE ESTUDIO POR SEMANA

SEMANA 12: INSTALACIÓN Y CONFIGURACIÓN

Consultar:

- Laravel. (2024). *Laravel 11.x Documentation*. <https://laravel.com/docs/11.x>
- Laravel. (2024). *Database: Migrations*. <https://laravel.com/docs/11.x/migrations>
- Laravel. (2024). *Eloquent: Getting Started*. <https://laravel.com/docs/11.x/eloquent>
- Laracasts. (2024). *Laravel 11 From Scratch*. <https://laracasts.com/series/laravel-11-from-scratch>

SEMANA 13: IMPLEMENTACIÓN CRUD

Consultar:

- Laravel. (2024). *Controllers*. <https://laravel.com/docs/11.x/controllers>
- Laravel. (2024). *Validation*. <https://laravel.com/docs/11.x/validation>
- Laravel. (2024). *Eloquent: API Resources*. <https://laravel.com/docs/11.x/eloquent-resources>
- RESTful API. (2024). *REST API Tutorial*. <https://restfulapi.net>

SEMANA 14: AUTENTICACIÓN Y AUTORIZACIÓN

Consultar:

- Laravel. (2024). *Laravel Sanctum*. <https://laravel.com/docs/11.x/sanctum>
- Laravel. (2024). *Authorization*. <https://laravel.com/docs/11.x/authorization>
- OWASP. (2024). *Security Best Practices*. <https://owasp.org>
- JWT.io. (2024). *JSON Web Tokens Introduction*. <https://jwt.io/introduction>

SEMANA 15: FUNCIONALIDADES AVANZADAS

Consultar:

- Laravel. (2024). *Eloquent: Relationships*. <https://laravel.com/docs/11.x/eloquent-relationships>
- Laravel. (2024). *Optimization*. <https://laravel.com/docs/11.x/deployment#optimization>
- Postman. (2024). *API Documentation*. <https://learning.postman.com>
- GitHub. (2024). *GitHub Docs*. <https://docs.github.com>

Instalación correcta de Laravel y sus componentes.

INSTALACIÓN DE LARAVEL EN WINDOWS 11

Estudiante: Yahir Sacan

Sistema Operativo: Windows 11

Editor: Visual Studio Code

Fecha: 14 de noviembre de 2025

PASO 1: INSTALAR XAMPP

Descargar XAMPP

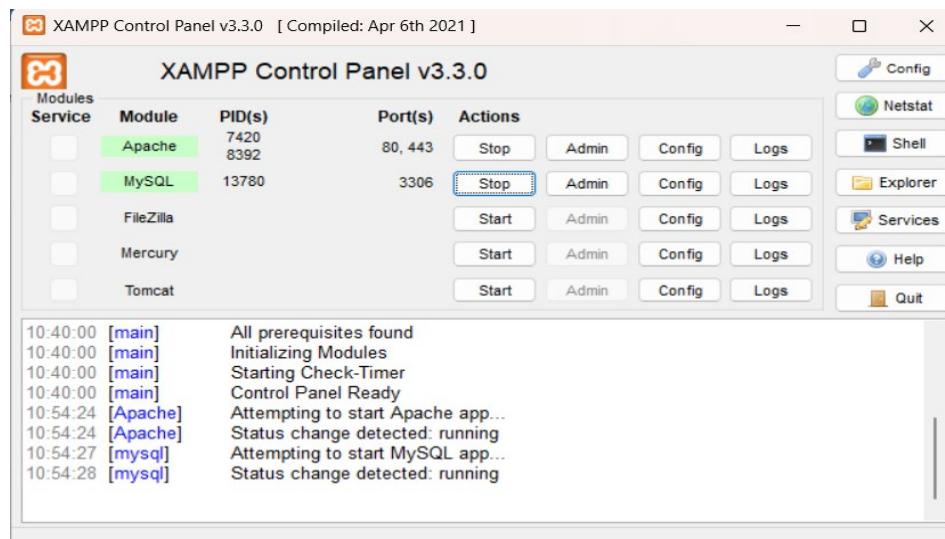
- URL: <https://www.apachefriends.org/download.html>
- Versión: PHP 8.2 o superior

Instalación

1. Ejecutar el instalador
2. Seleccionar componentes: Apache, MySQL, PHP, phpMyAdmin
3. Ruta de instalación: C:\xampp
4. Completar instalación

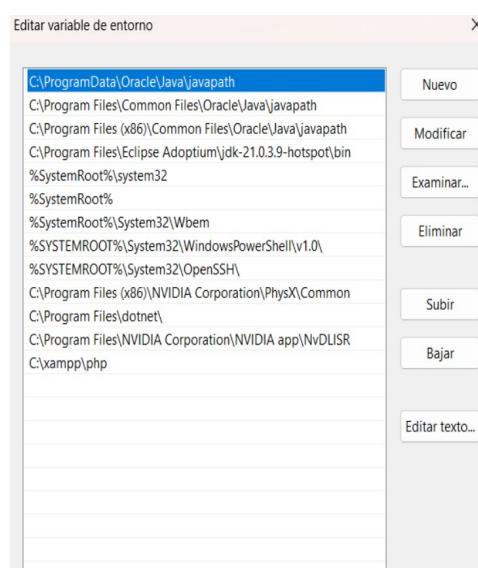
Iniciar servicios

1. Abrir XAMPP Control Panel
2. Iniciar Apache
3. Iniciar MySQL



Agregar PHP al PATH

1. Presionar Win + X > Sistema
2. Configuración avanzada del sistema
3. Variables de entorno
4. Editar Path en Variables del sistema
5. Agregar: C:\xampp\php



6. Aceptar todo
7. Cerrar y abrir nueva terminal

8. Verificar

PowerShell

```
php -v
```

```
PS C:\Users\sacan> php -v
PHP 8.2.12 (cli) (built: Oct 24 2023 21:15:15) (ZTS Visual C++ 2019 x64)
Copyright (c) The PHP Group
Zend Engine v4.2.12, Copyright (c) Zend Technologies
```

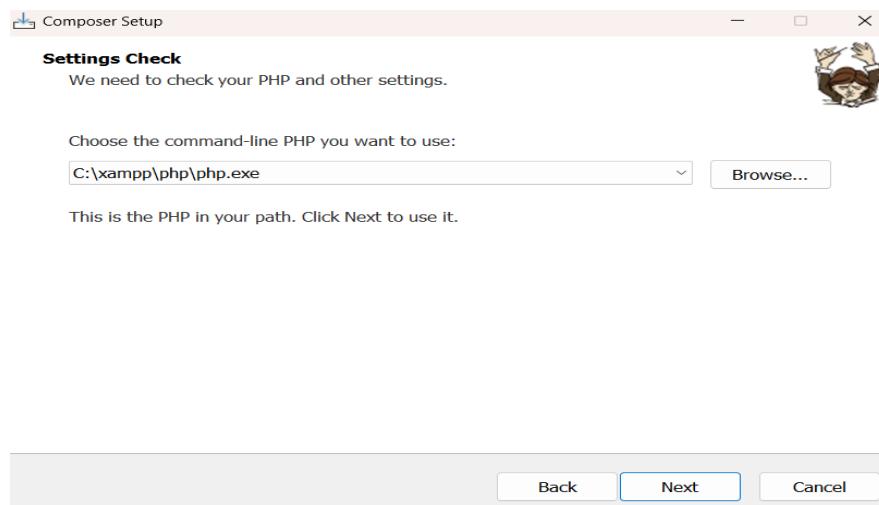
PASO 2: INSTALAR COMPOSER

Descargar

- URL: <https://getcomposer.org/Composer-Setup.exe>

Instalación

- Ejecutar Composer-Setup.exe
- Seleccionar PHP: C:\xampp\php\php.exe



- Completar instalación

Verificar

PowerShell

```
composer --version
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Instale la versión más reciente de PowerShell para obtener nuevas características y mejoras. https://aka.ms/PSWindows

PS C:\WINDOWS\system32> composer --version
Composer version 2.9.1 2025-11-13 16:10:38
PHP version 8.2.12 (C:\xampp\php\php.exe)
Run the "diagnose" command to get more detailed diagnostics output.
PS C:\WINDOWS\system32>
```

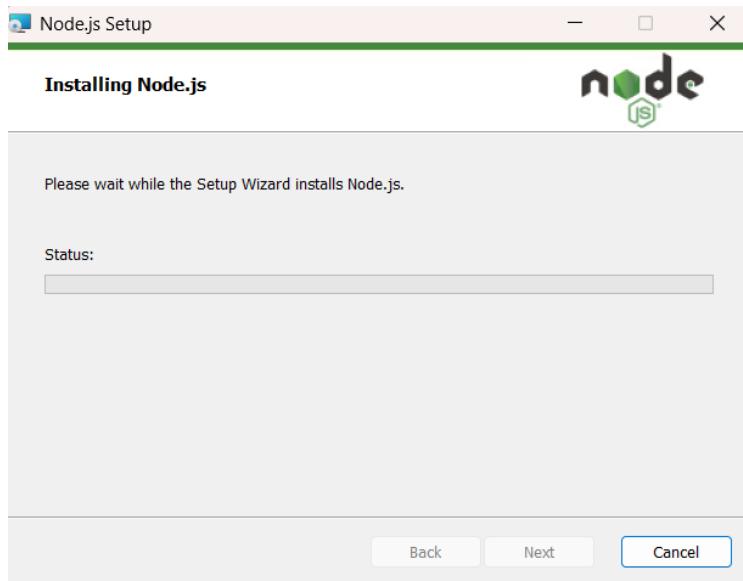
PASO 3: INSTALAR NODE.JS

Descargar

- URL: <https://nodejs.org/>
- Versión: LTS

Instalación

1. Ejecutar instalador
2. Opciones por defecto
3. Completar instalación



Verificar

PowerShell

node -v

npm -v

```
PS C:\Users\sacan> npm -v
11.6.2
PS C:\Users\sacan> node -v
v24.11.1
PS C:\Users\sacan>
```

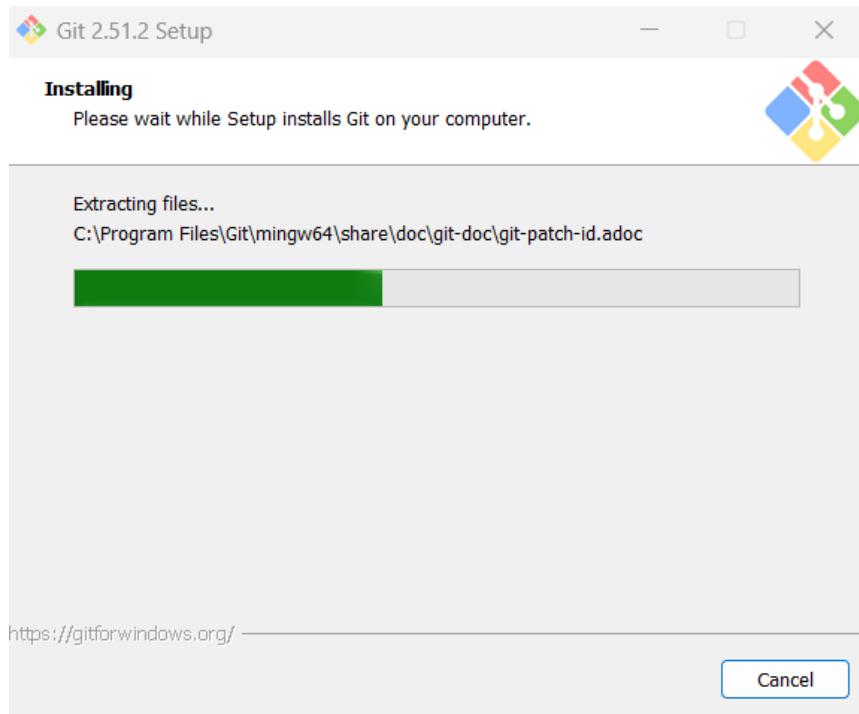
PASO 4: INSTALAR GIT

Descargar

- URL: <https://git-scm.com/download/win>

Instalación

1. Ejecutar instalador
2. Opciones por defecto
3. Editor: Visual Studio Code
4. Completar instalación



Verificar

PowerShell

git --version

```
PS C:\WINDOWS\system32> git --version
git version 2.51.2.windows.1
PS C:\WINDOWS\system32>
```

PASO 5: INSTALAR VISUAL STUDIO CODE

Descargar

- URL: <https://code.visualstudio.com/>

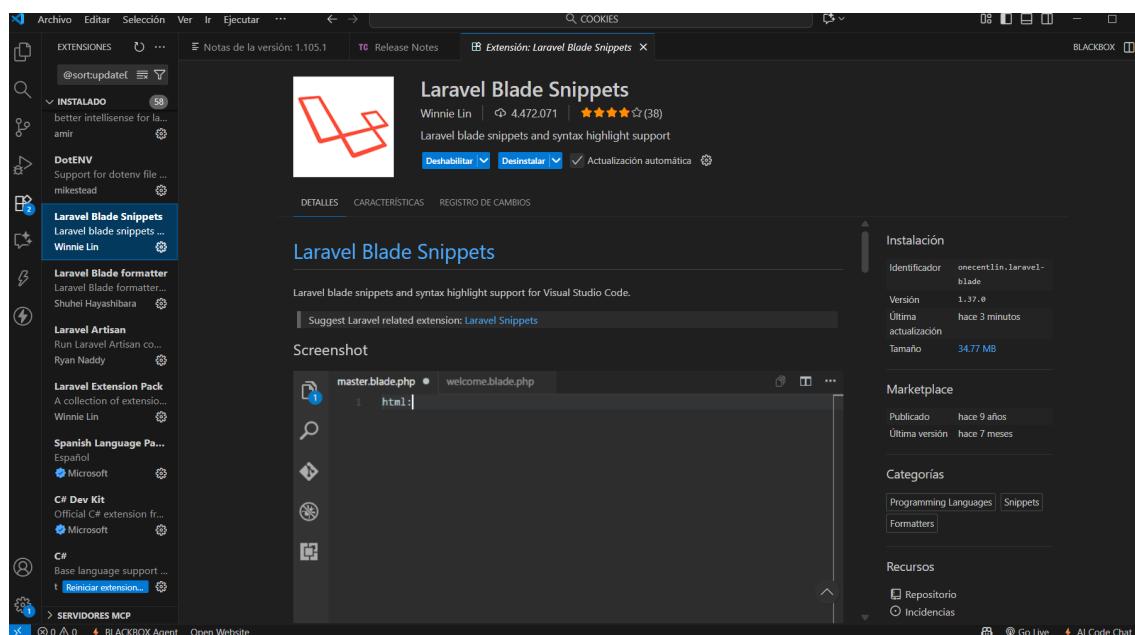
Instalación

1. Ejecutar instalador
2. Marcar:
 - Agregar "Abrir con Code" al menú contextual
 - Agregar a PATH
3. Completar instalación

Extensiones básicas

En VS Code (Ctrl + Shift + X), instalar:

- PHP Intelephense
- Laravel Extension Pack
- Thunder Client



PASO 6: CREAR PROYECTO LARAVEL

Abrir PowerShell

```
cd C:\xampp\htdocs
```

```
PS C:\WINDOWS\system32> cd C:\xampp\htdocs
```

Crear proyecto

```
composer create-project laravel/laravel gestion-aulas
```

```
PS C:\xampp\htdocs> composer create-project laravel/laravel gestion-aulas
Creating a "laravel/laravel" project at "./gestion-aulas"
Installing laravel/laravel (v12.10.1)
  Failed to download laravel/laravel from dist: The zip extension and unzip/7z commands are both missing, skipping.
The php.ini used by your command-line PHP is: C:\xampp\php\php.ini
  Now trying to download from source
    - Syncing laravel/laravel (v12.10.1) into cache
    - Installing laravel/laravel (v12.10.1): Cloning 1812490003 from cache
Created project in C:\xampp\htdocs\gestion-aulas
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
Loading composer repositories with package information
Updating dependencies
Lock file operations: 111 installs, 0 updates, 0 removals
  - Locking brick/math (0.14.0)
  - Locking carbonphp/carbon-doctrine-types (3.2.0)
  - Locking dflydev/dot-access-data (v3.0.3)
  - Locking doctrine/inflector (2.1.0)
  - Locking doctrine/lexer (3.0.1)
  - Locking dragonmantank/cron-expression (v3.6.0)
  - Locking egulias/email-validator (4.0.4)
  - Locking fakerphp/faker (v1.24.1)
  - Locking filp/whoops (2.18.4)
  - Locking fruitcake/php-cors (v1.3.0)
  - Locking graham-campbell/result-type (v1.1.3)
```

Ingresar al proyecto

```
cd gestion-aulas
```

Abrir en VS Code

```
code .
```

```
PS C:\xampp\htdocs> cd gestion-aulas
PS C:\xampp\htdocs\gestion-aulas> code .
PS C:\xampp\htdocs\gestion-aulas> php artisan serve
```

PASO 7: VERIFICAR INSTALACIÓN

Verificar versión de Laravel

PowerShell

```
php artisan --version
```

Iniciar servidor

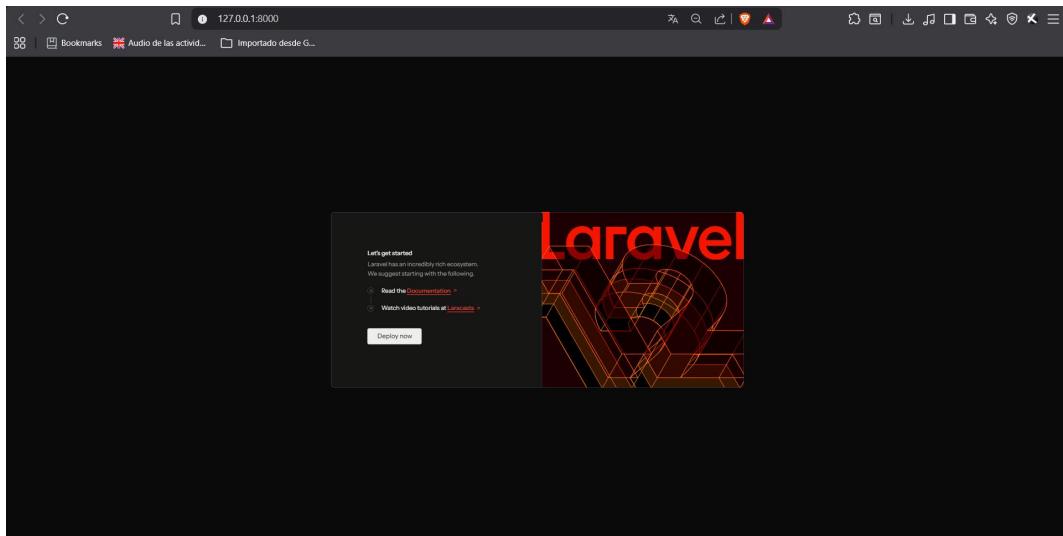
PowerShell

```
php artisan serve
```

```
PS C:\WINDOWS\system32> cd C:\xampp\htdocs\gestion-aulas
PS C:\xampp\htdocs\gestion-aulas> php artisan serve
  INFO  Server running on [http://127.0.0.1:8000].
  Press Ctrl+C to stop the server
```

Acceder en navegador

- URL: <http://127.0.0.1:8000>



RESUMEN DE COMANDOS DE VERIFICACIÓN

PowerShell

php -v

composer --version

node -v

npm -v

git --version

php artisan --version

INSTALACIÓN COMPLETADA

Laravel está instalado correctamente en:

- Ruta: C:\xampp\htdocs\gestion-aulas
- Versión: Laravel 11.x
- Servidor: <http://127.0.0.1:8000>