

Introducción práctica a Laravel Eloquent ORM

[Eloquent](#) es un [mapeador relacional de objetos \(ORM\)](#) que se incluye por defecto dentro del *framework* Laravel. Un ORM es un software que facilita el manejo de los registros de la base de datos representando los datos como objetos, trabajando como una capa de abstracción sobre el motor de la base de datos utilizado para almacenar los datos de una aplicación.

- This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](#).
- **English language:** This article is a translation from English into Spanish, published under license «[Attribution-NonCommercial-ShareAlike 4.0 International \(CC BY-NC-SA 4.0\)](#)», written by [ErikaHeidi](#), published on line by the company for leasing virtual machines [DigitalOcean](#). The tittle is «***A Practical Introduction to Laravel Eloquent ORM***» and [we created a copy at Wayback Machine for prevent in future a broken link](#). This work is licensed under the mentioned license but, of course, in Castilian language (AKA *Spanish*): «[Atribución-NoComercial-CompartirIgual 4.0 Internacional \(CC BY-NC-SA 4.0\)](#)».
-

- Esta obra está bajo una [Licencia Creative Commons Atribución-NoComercial-CompartirIgual 4.0 Internacional](#).
- **En castellano:** Este artículo es una traducción del inglés al castellano, publicado bajo licencia (en idioma inglés) «[Attribution-NonCommercial-ShareAlike 4.0 International \(CC BY-NC-SA 4.0\)](#)» escrito por [Erika Heidi](#), publicado en línea por la empresa de alojamiento de máquinas virtuales [DigitalOcean](#). El título original en idioma inglés es «[A Practical Introduction to Laravel Eloquent ORM](#)» y [hemos creado una copia en Wayback Machine](#) para prevenir un posible enlace roto a futuro.

Eloquent facilita la tarea de interactuar con las tablas de la base de datos, proporcionando un enfoque orientado a objetos para insertar, actualizar y eliminar registros de la base de datos, a la vez que ofrece una interfaz optimizada para ejecutar consultas SQL complejas.

En esta serie basada en proyectos, aprenderá a realizar consultas a bases de datos y a trabajar con relaciones en **Laravel Eloquent**. Para practicar los ejemplos explicados a lo largo de la serie, descargará una [aplicación Laravel de demostración](#) y utilizará la [configuración incluida](#) de **Docker Compose** para ejecutar un entorno de desarrollo PHP en contenedores. La demo consiste en una aplicación de una sola página que muestra una lista de enlaces o marcadores. Mejorará la aplicación añadiendo nuevos modelos y definiendo nuevas relaciones entre ellos que ampliarán la estructura actual de la base de datos.

Si desea una introducción al *framework* Laravel, puede seguir la guía sobre "[Cómo construir una página de aterrizaje de enlaces en Laravel](#)". Ese tutorial explica cómo puede construir desde cero la aplicación de demostración que esta serie utiliza como base.

Prerrequisitos

Aunque el código compartido a lo largo de esta serie debería funcionar sin problemas en múltiples entornos y sistemas, las instrucciones explicadas aquí se probaron en un sistema local Ubuntu 20.04 que ejecuta **Docker** y **Docker Compose**. Independientemente del sistema operativo base, esto es lo que necesitará para empezar:

- **Docker** instalado en su máquina local o servidor de desarrollo remoto. Si está ejecutando Ubuntu 20.04, puede seguir los pasos 1 y 2 de "[Cómo instalar y utilizar Docker en Ubuntu 20.04](#)" para configurarlo. Los usuarios de Windows y MacOS deben instalar [Docker Desktop](#) en su lugar.
- **Docker Compose** instalado en su máquina local o servidor de desarrollo. Docker

Compose viene incluido por defecto con Docker Desktop para los sistemas Windows y MacOS, pero los usuarios de GNU/Linux tienen que instalar el ejecutable Compose, siguiendo el paso 1 de "[Cómo instalar y utilizar Docker Compose en Ubuntu 20.04](#)".

- Un editor de código para PHP (opcional). Un editor de código ayuda a que el código sea más fácil de leer y formatear, y puede mejorar su productividad al señalar los problemas antes de ejecutar su código. Puedes seguir nuestra guía sobre "[Cómo configurar Visual Studio Code para proyectos PHP](#)". Visual Studio Code (**VSCo**de) es un editor de código gratuito, dentro de su entorno de desarrollo local.

Configuración del proyecto de demostración

Para empezar, necesitará configurar el proyecto de demostración de [Landing Laravel](#). Hay dos maneras de obtener el código de la aplicación de demostración listo para usar con esta serie:

- La primera forma es seguir la guía sobre "[Cómo construir una página de aterrizaje de enlaces en Laravel](#)". Esa serie explica cómo construir desde cero la aplicación de demostración que usará como base para la serie actual. Si elige esta opción, puede pasar al primer tutorial de esta serie "[Cómo crear una relación uno-a-muchos en Laravel Eloquent](#)".
- La segunda opción es descargar el código completo de la aplicación de demostración y utilizarlo como base sobre la que se construirá en esta serie.

Para descargar el código desde GitHub, asegúrese primero de que tiene instaladas las herramientas **curl** y **unzip** en su sistema:

A continuación, cambie a su directorio principal utilizando el comando **cd**, y luego descargue la versión 0.1.1 del proyecto utilizando **curl**:

El código de demostración se guardará en el archivo **landing-laravel.zip**. Descomprima el contenido del archivo zip, cambie el nombre del directorio a **landing-laravel**, y cambie de directorio con **cd** en la carpeta de la aplicación utilizando los siguientes comandos:

La imagen de **MySQL** que utilizará con **Docker Compose** utiliza variables de entorno para configurar el usuario y la contraseña de la base de datos. Antes de poner en marcha el entorno por

KS7000+WP

KS7000 migra a GNU/Linux y escoge a WordPress para registrar el camino.
<https://www.ks7000.net.ve>

primera vez, tendrá que configurar un archivo `.env` que contenga las credenciales de la base de datos. La aplicación de demostración incluye un archivo `.env.example` de ejemplo que bien puede reutilizar. Cópelo a su lugar ejecutando el siguiente comando:

Ahora puede poner en funcionamiento todo su entorno con:

Instale las dependencias de la aplicación Laravel con:

Esto instalará las dependencias de PHP que Laravel necesita a través de Docker Composer. A continuación, cree usted una clave de aplicación única usando el comando **artisan**:

Debe obtener una salida como la siguiente, indicando que artisan agregó un clave única a de aplicación a su fichero `.env`:

```
Application key set successfully.
```

Por último, ejecute las migraciones de la base de datos y los *sembradores* para configurar la base de datos:

Recibirá una salida como la siguiente, que indica que las tablas de la base de datos y los datos de las *semillas* están configurados correctamente:

```
Migration table created successfully.  
.  
.  
.  
Database seeding completed successfully.
```

La aplicación está ahora configurada y funcionando en su entorno de desarrollo. Dirija su navegador web a:

KS7000+WP

KS7000 migra a GNU/Linux y escoge a WordPress para registrar el camino.

<https://www.ks7000.net.ve>

`http://localhost:8000`

y recibirá una página similar a esta:

Con esta estructura de base, puede continuar con el resto de los tutoriales de esta serie.

Cómo crear una relación uno-a-muchos en Laravel Eloquent

- This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](#).
- **English language:** This article is a translation from English into Spanish, published under license «[Attribution-NonCommercial-ShareAlike 4.0 International \(CC BY-NC-SA 4.0\)](#) », written by [ErikaHeidi](#), published on line by the company for leasing virtual machines [DigitalOcean](#). The tittle is «***A Practical Introduction to Laravel Eloquent ORM***» and [we created a copy at Wayback Machine for prevent in future a broken link](#). This work is licensed under the mentioned license but, of course, in Castilian language (AKA *Spanish*): «[Atribución-NoComercial-CompartirIguual 4.0 Internacional \(CC BY-NC-SA 4.0\)](#) ».

-

- Esta obra está bajo una [Licencia Creative Commons Atribución-NoComercial-CompartirIguual 4.0 Internacional](#).
- **En castellano:** Este artículo es una traducción del inglés al castellano, publicado bajo licencia (en idioma inglés) «[Attribution-NonCommercial-ShareAlike 4.0 International \(CC BY-NC-SA 4.0\)](#) » escrito por [Erika Heidi](#), publicado en línea por la empresa de alojamiento de máquinas virtuales [DigitalOcean](#). El título original en idioma inglés es «***A Practical Introduction to Laravel Eloquent ORM***» y [hemos creado una copia en Wayback Machine](#) para prevenir un posible enlace roto a futuro.