# Instalación de laravel

* **Instalación de Composer**:
  + <https://getcomposer.org/download/>

sudo mv composer.phar /usr/local/bin/composer

* + - (En caso de que se te haya olvidado copiarlo de la página)

sudo composer self-update

* **Creación del host virtual y configuración de Apache**:

sudo cp /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-available/milaravel.es.conf

sudo gedit /etc/apache2/sites-available/milaravel.es.conf

<VirtualHost \*:80>

ServerName milaravel.es

ServerAdmin webmaster@localhost

DocumentRoot /var/www/milaravel.es/public

<Directory /var/www/milaravel.es>

AllowOverride All

</Directory>

ErrorLog ${APACHE\_LOG\_DIR}/error.log

CustomLog ${APACHE\_LOG\_DIR}/access.log combined

</VirtualHost>

sudo a2ensite milaravel.es

sudo a2enmod rewrite

mkdir /var/www/milaravel.es

sudo service apache2 restart

sudo gedit /etc/hosts

127.0.0.1 milaravel.es

* **Creamos el proyecto de Laravel**:

cd /var/www

composer create-project laravel/laravel milaravel.es "9.5.1" --prefer-dist

sudo chgrp -R www-data /var/www/milaravel.es

sudo chmod -R 775 /var/www/milaravel.es/storage

* **Instalación de plugin para obtener documentación y autocompletado en cualquier IDE**:

cd milaravel.es

composer require barryvdh/laravel-ide-helper

cd config

sudo gedit app.php

* + - Añade:

Barryvdh\LaravelIdeHelper\IdeHelperServiceProvider::class

al array de providers.

php artisan ide-helper:generate

php artisan ide-helper:models

para generar los archivos de ayuda.

* **Verificar instalación de Laravel (ver la versión instalada)**:

php artisan -V

# Conexión con BB.DD

1. Editar el archivo .env y configurar los siguientes campos:

DB\_CONNECTION=mysql

DB\_HOST=127.0.0.1

DB\_PORT=3306

DB\_DATABASE=biblioteca

DB\_USERNAME=super

DB\_PASSWORD=123456

1. Crear/importar la BB.DD en phpMyAdmin.

# Migraciones.

1. Crear la migración:

php artisan make:migration create\_libros\_table

Se genera un fichero así:

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

1. Borrar el timestamps. Solo guarda en una columna la fecha y hora de cuando se creó el registro en la tabla.
2. Ejemplo:

/\*\*

\* Run the migrations.

\*

\* @return void

\*/

public function up()

{

Schema::create('equipos', function (Blueprint $table) {

$table->id("eqp\_id");

$table->string("eqp\_nombre");

$table->integer("eqp\_puntos", false)->default(0);

$table->integer("eqp\_golesfavor", false)->default(0);

$table->integer("eqp\_golescontra", false)->default(0);

// $table->timestamps();

});

}

/\*\*

\* Reverse the migrations.

\*

\* @return void

\*/

public function down()

{

Schema::dropIfExists('equipos');

}

1. Finalizar migración:

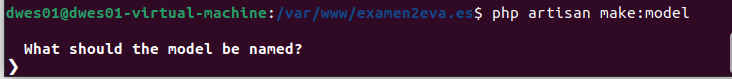
php artisan migrate

Para revertir migración:

php artisan migrate:rollback

# Creación del modelo y el controlador a la vez.

php artisan make:model



Marcamos la opción 6:

Patrón de fondo

Descripción generada automáticamente con confianza baja

Finalizado:

Texto

Descripción generada automáticamente