Instalar laragon y crear proyecto en laravel

**Requisitos para la versión 6.x**

### [Requerimientos del servidor](https://laravel.com/docs/6.x#server-requirements)

**El marco de trabajo Laravel tiene algunos requisitos del sistema. La máquina virtual** [**Laravel Homestead**](https://laravel.com/docs/6.x/homestead) **cumple todos estos requisitos , por lo que se recomienda que utilice Homestead como su entorno de desarrollo local de Laravel.**

Sin embargo, si no está utilizando Homestead, deberá asegurarse de que su **servidor cumpla con los siguientes requisitos:**

* PHP> = 7.2.0
* Extensión PHP BCMath
* Extensión PHP Ctype
* Extensión PHP JSON
* Extensión PHP Mbstring
* Extensión PHP OpenSSL
* PDO PHP Extension
* Tokenizer PHP Extension
* Extensión XML PHP

Canal de ayuda <https://discordapp.com/channels/297040613688475649/486650445297877003>

tutorial youtube

<https://www.youtube.com/watch?v=2pQSt9ST22A>

luego de instalar laragon lo abrimos y abrimos la terminal, seleccionamos la carpeta root donde instalamos laragon y dentro desde CMD o el terminal integrado ejecutamos el siguiente comando para crear un proyecto en laravel

**composer create-project laravel/laravel nombre\_proyecto --prefer-dist**

Este comando instala laravel solo en la carpeta del proyecto creado

para instalar laravel globalmente, este comando también se debe ejecutar cuando se quiere actualizar un proyecto

**composer global require laravel/installer**

Ahora se puede crear un proyecto con el comando

**laravel new nombre\_proyecto**

Luego vamos a la carpeta donde se creó el proyecto y para levantar el proyecto ejecutamos

**php artisan serve**

Nos dará la url para ver el proyecto

<http://127.0.0.1:8000/>

Editar el index

resources/views/welcome.blade.php

**Para configurar host virtuales ir a las carpetas de apache o usar con click derecho sobre el icono de laravel y ir a apache>sites-enables>nombre\_archivo.conf**

**Debería tener una configuración similar a esta, en este ejemplo el proyecto lo llamamos 24\_7**

<VirtualHost \*:80>

DocumentRoot "C:/laragon/www/24\_7/public/"

ServerName 24\_7.test

ServerAlias \*.24\_7.test

<Directory "C:/laragon/www/24\_7/public/">

AllowOverride All

Require all granted

</Directory>

</VirtualHost>

# If you want to use SSL, enable it by going to Menu > Apache > SSL > Enabled

**Guia para configurar proyecto y tener acceso desde internet**

[**https://desarrolloweb.com/articulos/laragon.html**](https://desarrolloweb.com/articulos/laragon.html)

**Configurar el archivo .env para la conexión**

**DB\_CONNECTION=mysql**

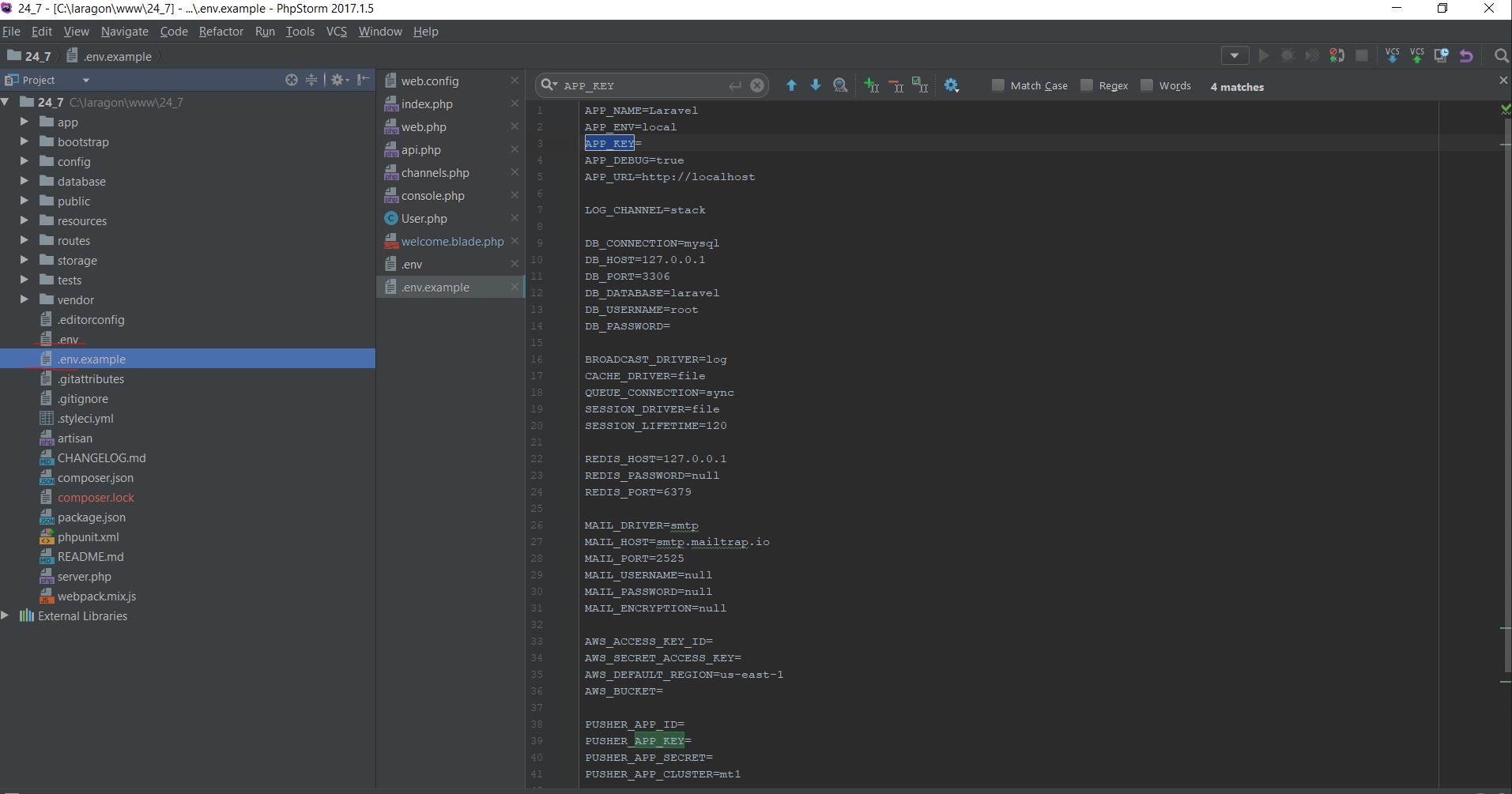
**DB\_HOST=127.0.0.1**

**DB\_PORT=3306**

**DB\_DATABASE=api\_notas**

**DB\_USERNAME=root**

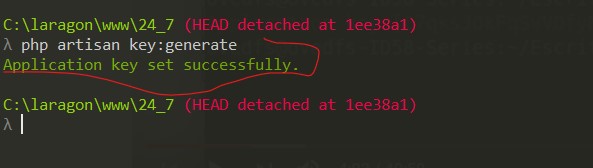
**DB\_PASSWORD=**

****

**crear la clave de encriptación en laravel(la versión 6.x trae esta clave)**

escribimos en la consola **php artisan key:generate**

para que se genere dentro de la variable **APP\_KEY=** la key dentro del archivo **.env**

****

**actualizar el composer**

composer update

**Instalar y actualizar npm (esta carpeta la podemos excluir si es subida a github ya que pesa mucho, y cuando se descargue el proyecto lo instalamos y actualizamos con los siguientes comandos)**

npm install

npm update

**apagar server y levantarlo**

php artisan serve

ORM

para relacionar ids y tablas

Eloquent: relationships

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* RUTAS \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

en terminal para ver rutas

**php artisan route:list**

las rutas están en carpeta routes en la raiz del proyecto

principalmente dentro de web.php

los controladores, están en la carpeta app/http/controllers

los modelos están en la raíz de app

las vistas están en resources/views

para llamar a los métodos serían luego del aroba "@"

crear model por consola

php artisan make:model mantenedor\_bodega

RESOLUCIÓN DE PROBLEMAS

**Cuando se instala la base de datos y el proyecto de laravel en un entorno nuevo si no actualiza ejecutar el comando**

verificar el .gitignore

composer update

npm install

npm update

El problema con la autenticación podría ser cambios en el archivo dentro de la carpeta **vendor** ya que este es ignorado

vendor/laravel/framework/src/Illuminate/Auth/Authenticatable.php

se cambio la variable del método

**public function** getAuthPassword()

{

**return** $this->password;

}

por $this->clave;

Para que funcione la clave y usuario, hacer un **copi as insert statement**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\* BASE DE DATOS \*\*\*\*\*\*\*\*\*\*\*\*\***

en el archivo .env se configura la conexión

para crear una nueva **base de datos** abrimos la sesión click derecho sobre el icono de laragon y abrimos database click en **nueva,** damos el nombre

para crear una tabla, tumbamos el servidor y volvemos a levantarlo para que se conecte y crear la **tabla** con composer en esta caso se llamara notas

**php artisan make:migration create\_notas\_table**

**php artisan migrate**

Estos comandos generan la tabla **notas** dentro de la base de datos y generará otras tablas por defecto, como **users**, etc...

Puedes ubicar la migración en database/migrations/. Agreguemos el código necesario en los metodos up (migrate) y down (rollback). Recuerda leer sobre migraciones en laravel si no estas claro en este tema <https://laravel.com/docs/5.2/migrations>

para agregar un nuevo campo a la tabla

en datase/migration/nombretabla.php

agregamos en nuevo campo por ejemplo

$table->string(**'nota'**);

luego refrescamos las tablas con el comando

**php artisan migrate:refresh**

**Ahora para llenar la tabla ponemos la clase en**

C:\laragon\www\api\_react\database\seeds\DatabaseSeeder.php

Ahora llenamos la tabla con datos usando los seeder de laravel, ejecutamos el comando

**php artisan make:seeder UsersTableSeeder**

crear model por consola

**php artisan make:model Notas**

esto genera el modelo la ruta puede varias segun S.O o nombre del proyecto

**C:\laragon\www\api\_react\app\Notas.php**

Ahora, configuremos nuestra ruta en **routes/web.php**

creamos el controlador para esta ruta

**php artisan make:controller nombre-controlador**

en este caso crearemos el **ApiController**

importamos en el controlador **use App\Notas;** para la conexión

en este punto levantamos el servidor nuevamente

con php artisan serve

y pasamos la ruta

<http://127.0.0.1:8000/notas>

Nos traerá un objeto en formato json con las notas

{

* **notas**:
* [
  + {
    - **id**: 1,
    - **nota**: "1",
    - **created\_at**: null,
    - **updated\_at**: null
  + },
  + {
    - **id**: 2,
    - **nota**: "7",
    - **created\_at**: null,
    - **updated\_at**: null
  + }
* ]

}

**\*\*\*\*\*\*\*\*\*\*\*\* GUIA PROYECTO API LARAVEL CON REACT \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

[**https://medium.com/@alejandroroa/consumir-una-api-con-react-js-d22864b1a578**](https://medium.com/@alejandroroa/consumir-una-api-con-react-js-d22864b1a578)

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* PASO PARA CREAR APP REACT \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

uso de react, podemos crear la carpeta fuera del proyecto para simular que estamos en otro lugar o servidor remoto o dentro del proyecto laravel y crear una carpeta donde estara todo el proyecto en react consumiendo la api en laravel

guia react

<https://medium.com/@ernestognw/construye-tu-primera-aplicaci%C3%B3n-web-con-reactjs-parte-1-4c8bcca3b9db>