Curso Laravel

https://www.youtube.com/watch?v=MYyJ4PuL4pY&t=1572s

Cmd para crear un proyecto de laravel (ejm: speaksmarter): **composer create-project laravel/laravel speaksmarter**

En la carpeta **resources** están todos los elemento visuales y funcionales del proyecto.

En la carpeta **routes** hay varios elementos, entre ellos los documentos **web** que define las rutas URL que podemos seguir y sus respectivas respuestas según la petición HTTP que hagamos. Y el archivo **api** que, al igual que web, define las resuestas de las apis que definamos allí.

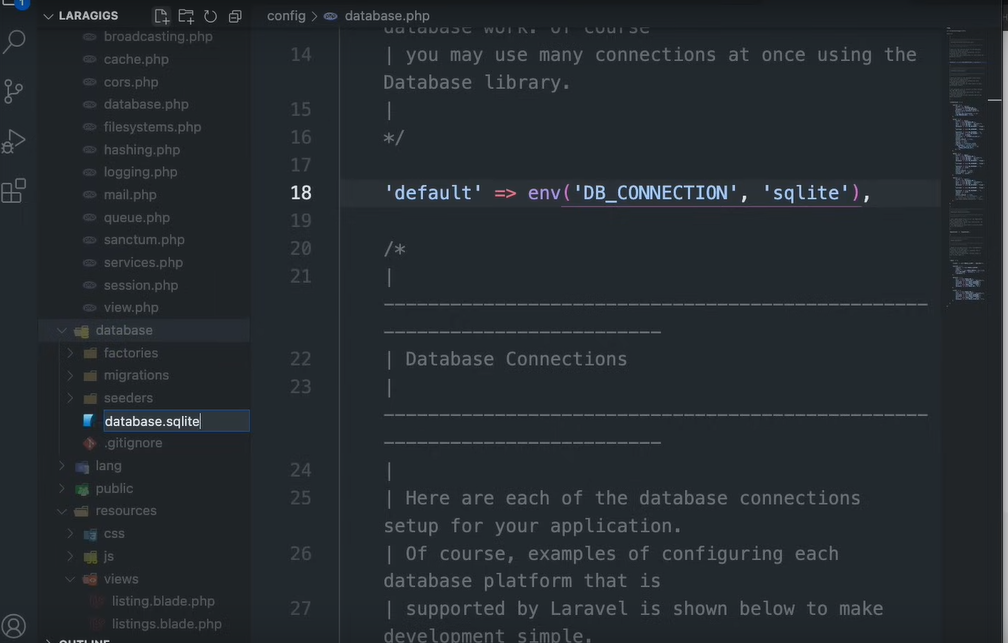
Basic model: para crear un modelo usamos el comando (por ejmplo, para el modelo Listing):

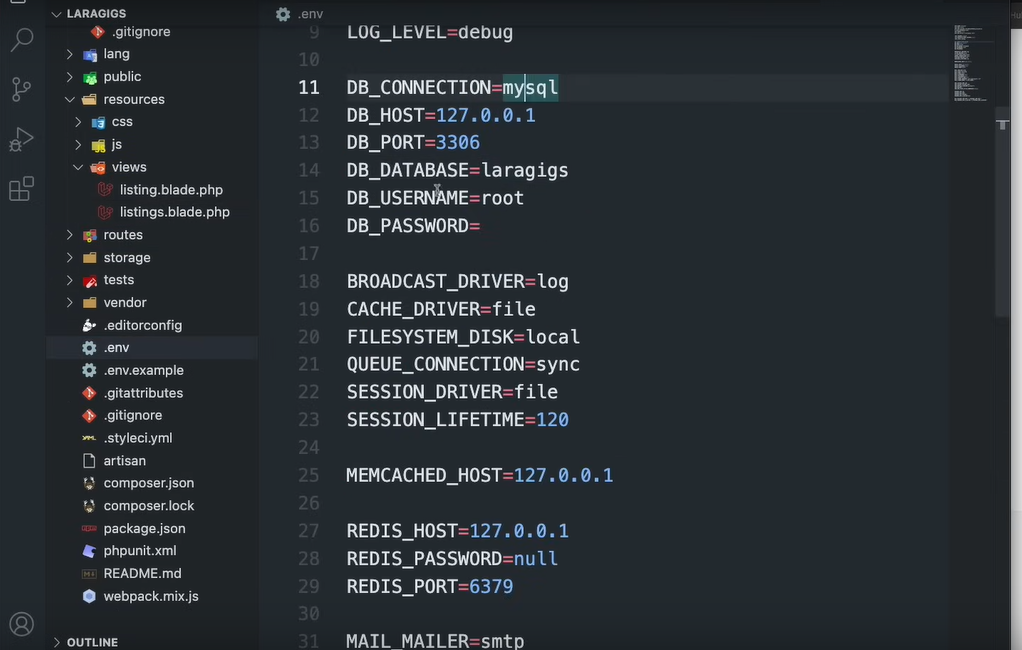
**php artisan make:model Listing**

Estos modelos se pueden crear también de forma manual, para ello nos dirigimos a la carpeta de **‘app/Models’**. Allí habrá un modelo de prueba llamado User.

En el archivo **‘/config/database.php’** podemos encontrar el archivo de configuraciones a BD del proyecto, allí podemos encontrar parámetros de configuración para distintos motores de BD.

En el caso que utilices sqlite NO necesitarías hacer algún tipo de configuración y para poder empezar a utilizarlo necesitarías crear un archivo en **/database**/ llamado **database.sqlite**



Posteriormente faltaría dirigirse al archivo de variables de entorno (**.env**) y llenar los datos de configuración del archivo, definiendo las constantes necesarios con los datos de conexión a la BD. 

En este curso creamos una bd llamada laravelproject, no creamos tablas, la idea es que Laravel migre todos nuestros modelos a la BD.

Si vas a la carpeta de /dabase/migrations/ allí podras encontrar algunas migraciones de ejemplo.

Creación de archivo de migración para listings: **php artisan make:migration create\_listings\_table**

Correr las migraciones: **php artisan migrate**

Comando para actualizar (refescar) las migraciones: **php artisan migrate:refresh**

Esto actualizará los cambios hechos en los archivos de migración, prácticamente elimina y vuelve a crear toda la estructura de BD, con ello elimina también los datos que tengamos en las tablas y estas quedarán como nuevas (de hecho porque son nuevas).

Para hacer un rollback de las migraciones podemos utilizar el siguiente código: **php artisan migrate:reset**

Cmd para crear un seeder (ejm. LevelSeeder): **php artisan make:seeder LevelSeeder**

Comando para correr los seeders en BD: **php artisan db:seed**

Estos seeders se usan para poblar las tablas de BD, la mayoría de veces con datos aleatorios.

Con el comando: **php artisan migrate:refresh --seed**

Hará lo mismo que antes pero volverá a poblar las tablas con los scitps seeders que tengamos habilitados.

Comando para crear modelos con elocuent (ejm. Modelo Listing): **php artisan make:model Listing**

Cmnd para crear un factory para un modelo(ejm. ListingFactory): **php artisan make:factory ListingFactory**

Cmd para crear un controller(ejm. ListingController): **php artisan make:controller ListingController**

Cmd para hacer consultas sql y trabajar con los modelos laravel desde consola: **php artisan tinker**

Uso de jetstream e inertia para la creación de autorizaciones de usuarios:

Cmd para instalar jetstream: **composer require laravel/jetstream**

Cmd para instalar inertia: **php artisan jetstream:install inertia**

Uso de spartie para la creación de usuarios y roles en la app:

Cmd para instalar spartie: **composer require spatie/laravel-permission**

Con spartie podemos crear toda ls infraestructura que necesita la app para el proceso de gestión de usuarios, roles y permisos.

Cmd para generar los archivos de configuración y migración de la infraestructura:

**php artisan vendor:publish --provider=”Spatie\Permission\PermissionServiceProvider”**

Para el Proyecto Laravel + Vue:

Instalar dependencias de Jetstream : **npm install**

Correr el servidor de Laravel y Vite: **npm run dev**

Correr el servidor de Laravel : **php artisan serve**

Hasta el momento se han utilizado las 3 librerías para el desarrollo del proyecto.

//Common Resource Routes:

// index - Show all listings

// show - Show single listening

// create - Show form to create new listing

// store - Store new listing

// edit - Show form to edit listing

// update - Update listing

// destroy - Delete listing