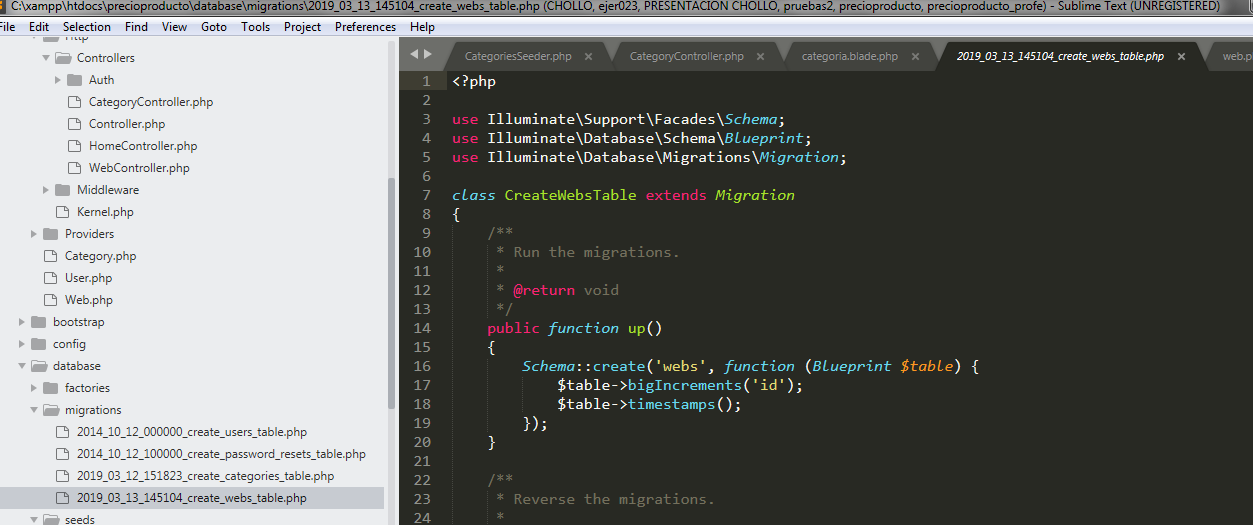
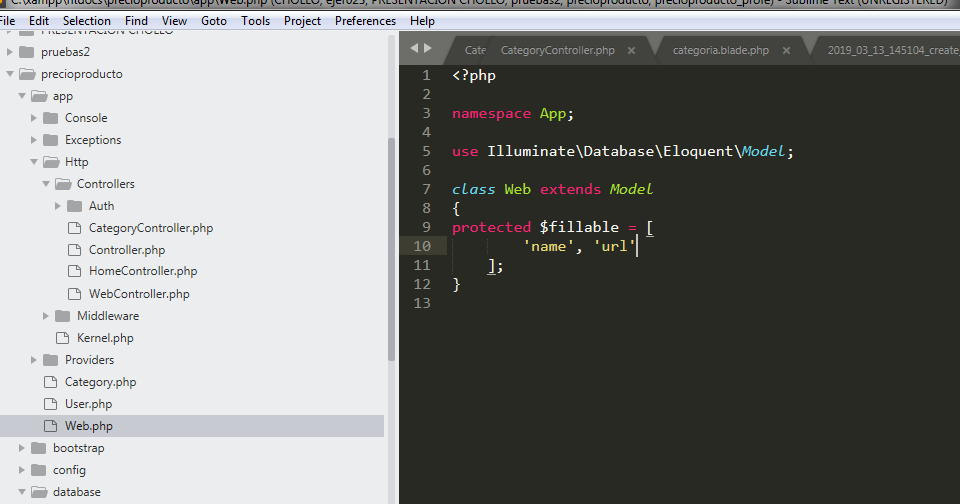
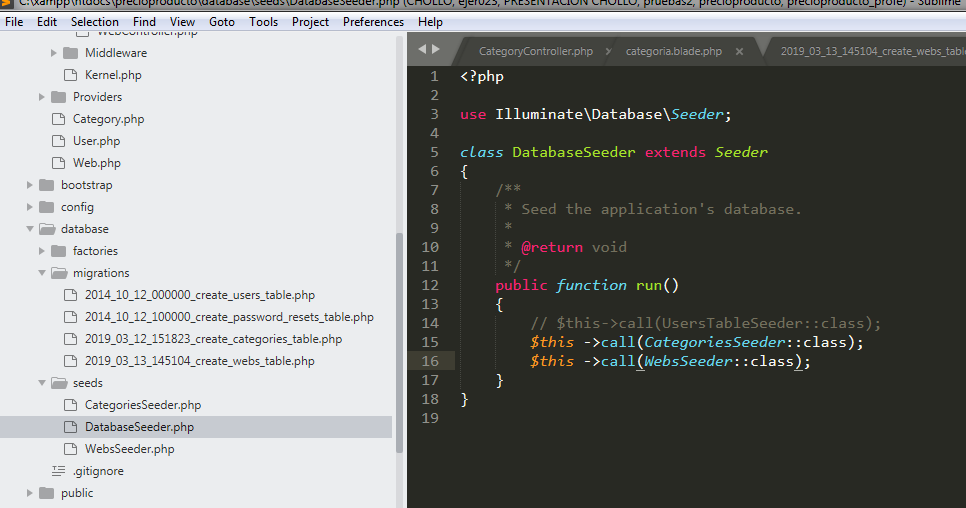
1. Crear modelo Web + Controller + migration
2. Migration: definir campos
3. Crear un seed para dar de alta
   1. Ebay
   2. Wallapop
   3. segundamano
4. Vista web -> listado
5. Ruta Controller@listar
6. Función listar

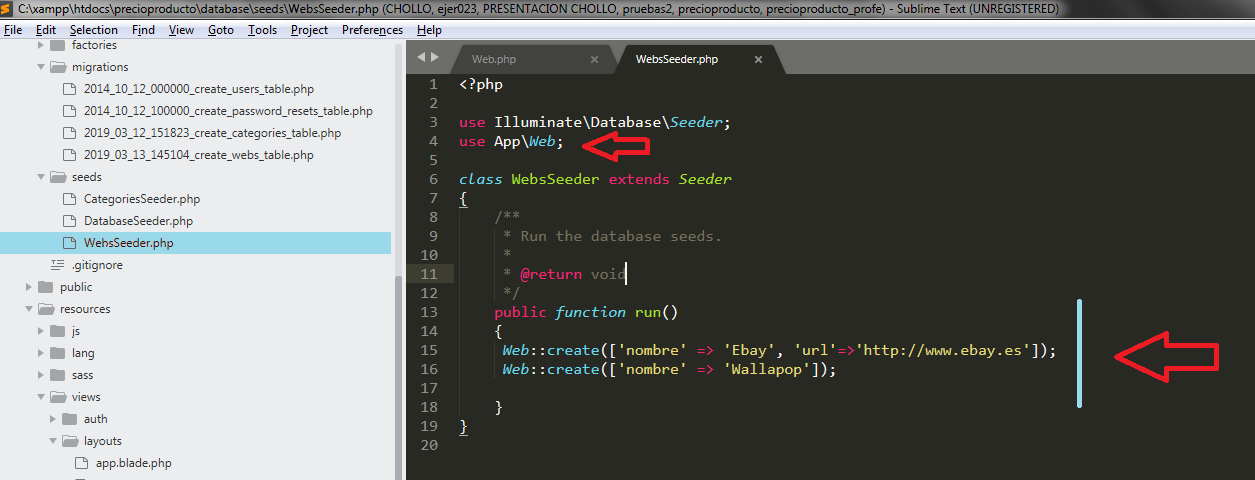


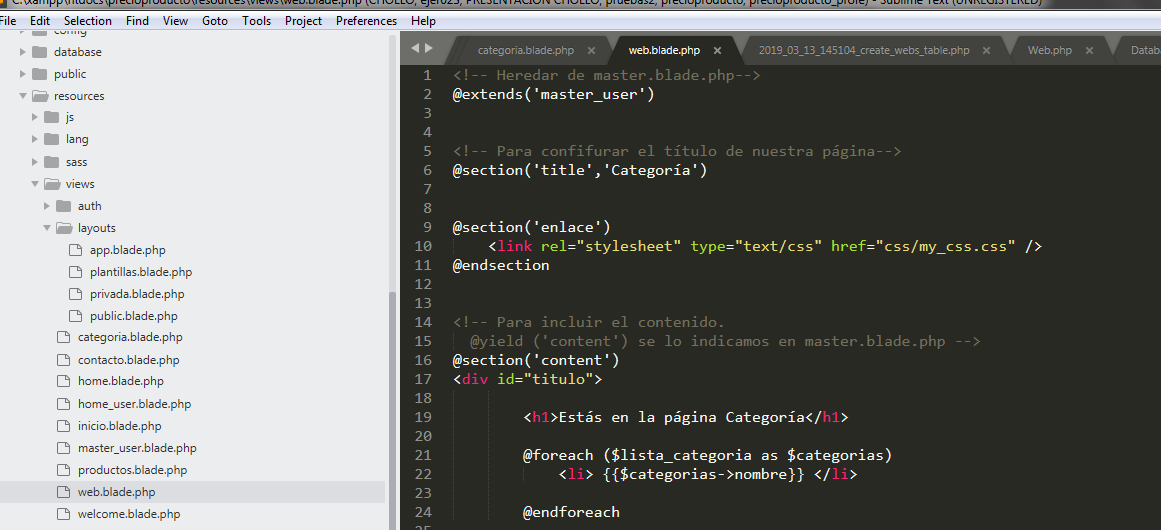


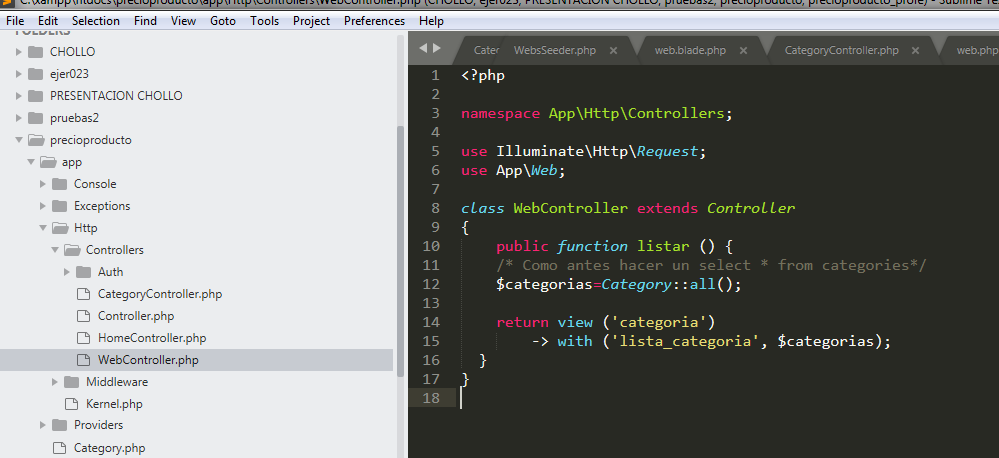


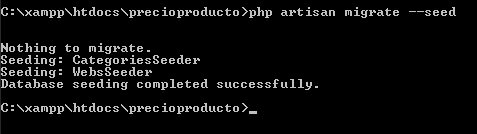










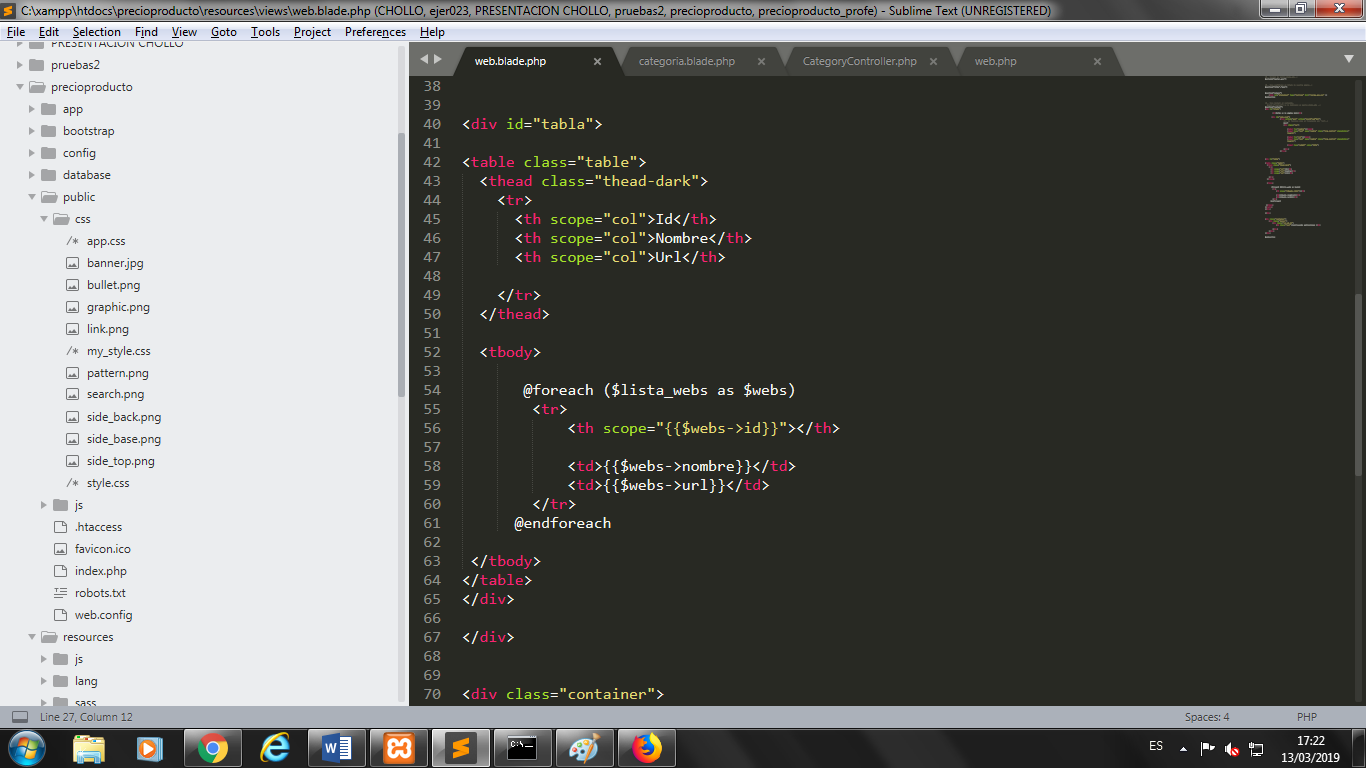


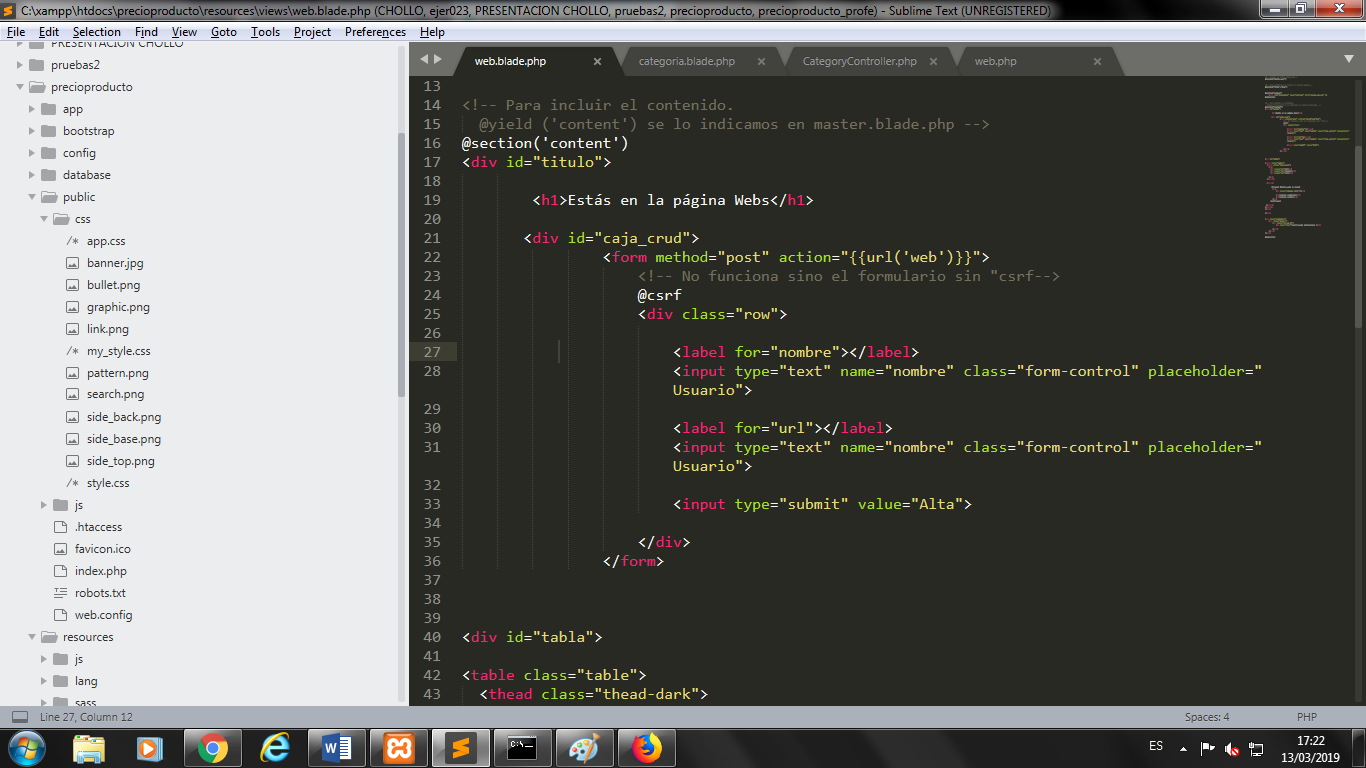
1. Modificar la View categorías –>Crear un form
2. Ruta categoría -> tipo post
   1. categoryController@insertar -> crear la función
3. ir a CategoryController a la función insertar y que la función
   1. recoja los datos de form
   2. insertar en BBDD
   3. VIEW (enseñar)

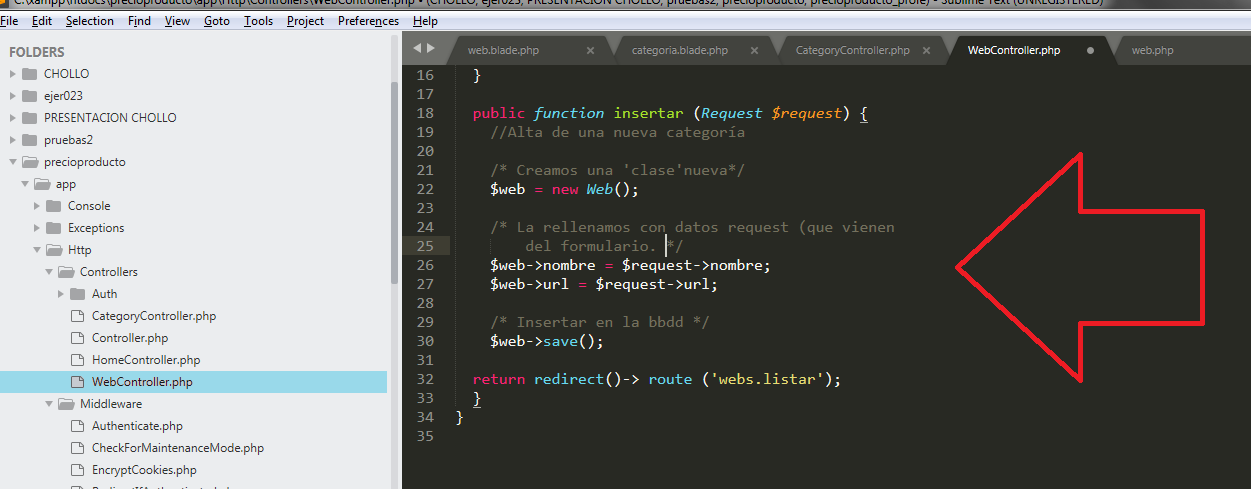
/\* Borra todo, la bd la reconstruye e inserta los datos. UN TODO EN 1\*/

Php artisan migrate:refresh –seed

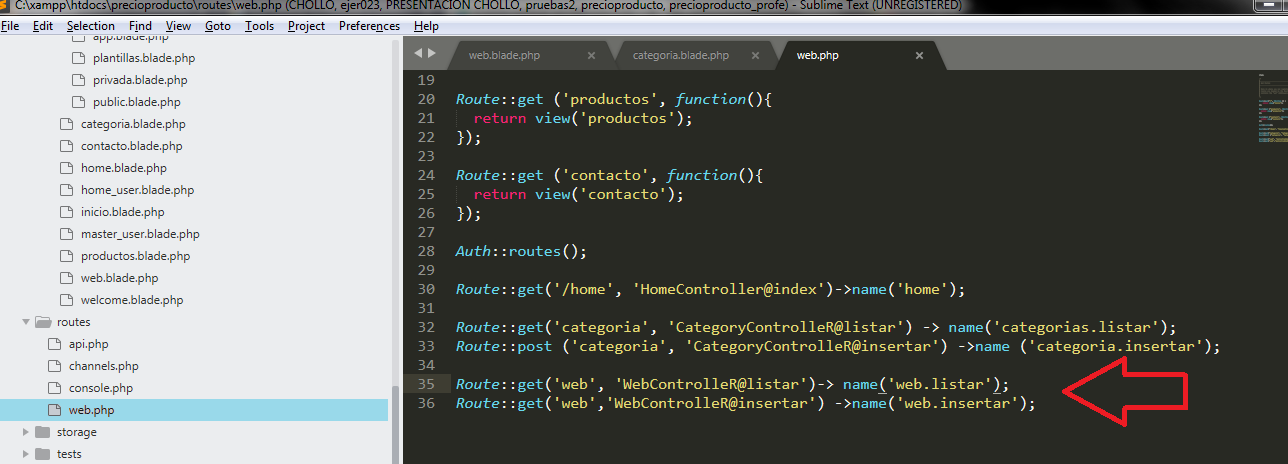
Tabla listar







Damos alias a las routas para poder redirigir en caso de necesitarlo como en la función añadir



***Falta por hacer: rutas resources***

**php artisan make:model Web –mcr**

crea:

modelo

m->nigration

c->controlller

r-> RESOURCE (para hacer crud)

php artisan route:list

aparte de decir las routas creadas, también indica quçe necesita cada llamada a esa ruta.

*Anuncios*

* Tendrá relación con 3 tablas

1. Crear la clase (Ad->advertismente :anuncio)

php artisan make :model Ad -mcr

modelo

m->nigration

c->controlller

r-> RESOURCE (para hacer crud)

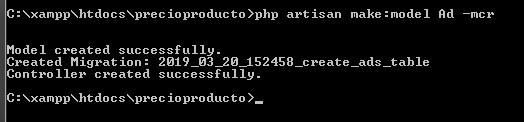
creará:

clase Ad.php

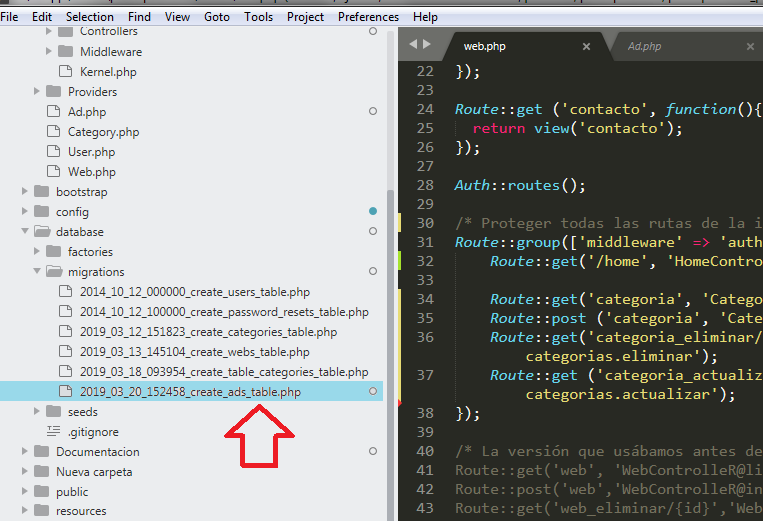
tabla: ads -> faltarán los campos

AdController

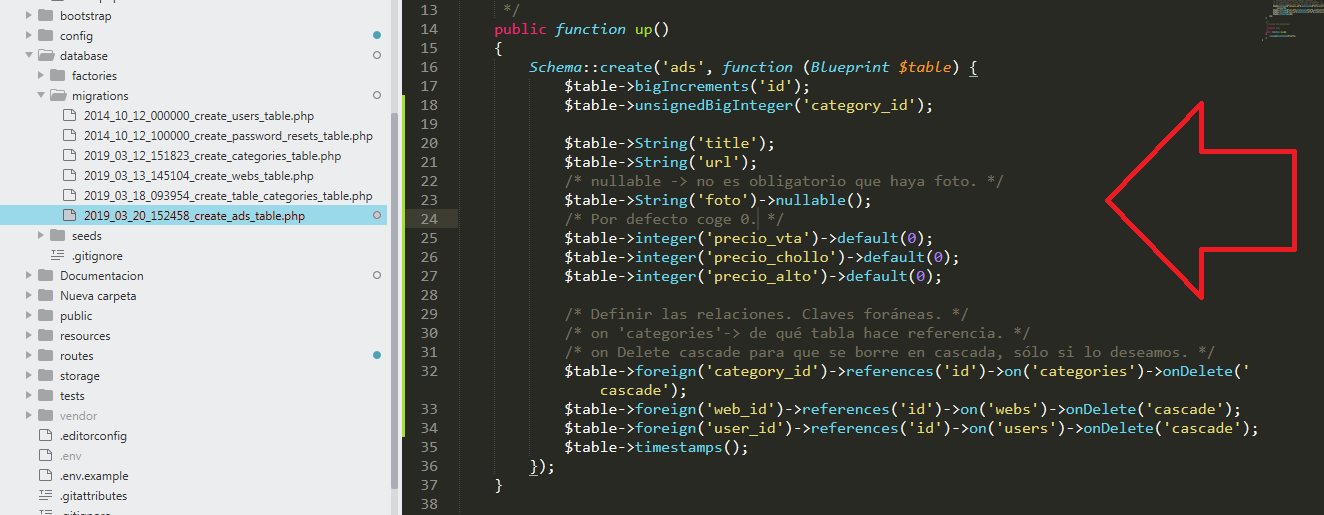
Route Ad(resource) --🡪 *Route::resource ('ads','AdController');*



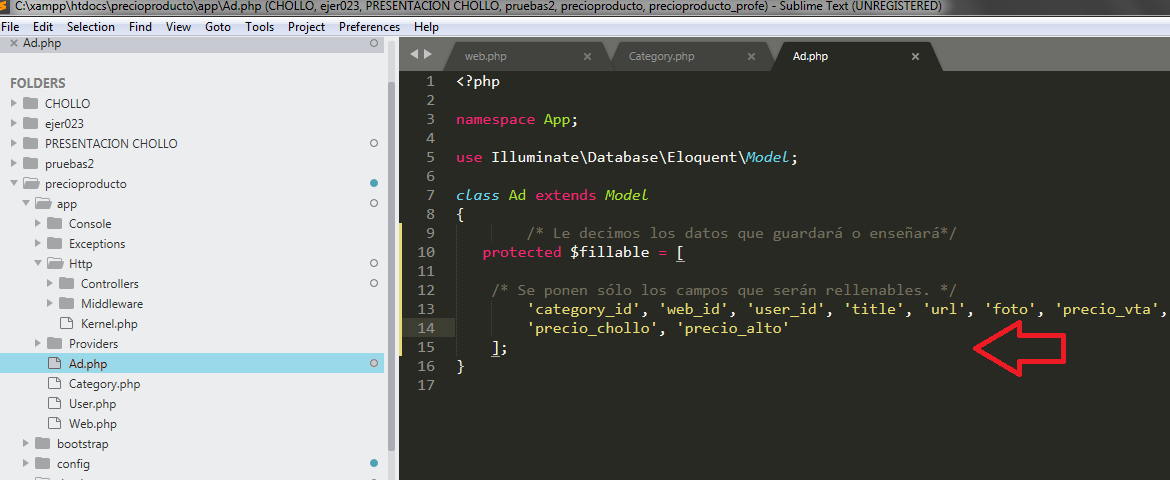
Vamos a la migración para añadir los campos a las tablas, el tipo, los tamaños, condiciones….



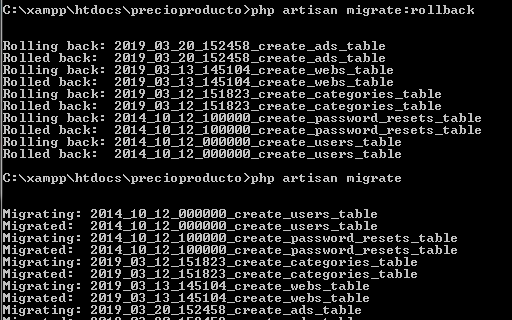
Si hay relaciones entre tablas, las creamos (las claves foráneas)



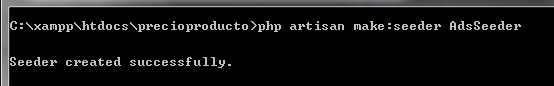
Ir a la clase Ad y ‘declarar’ los campos



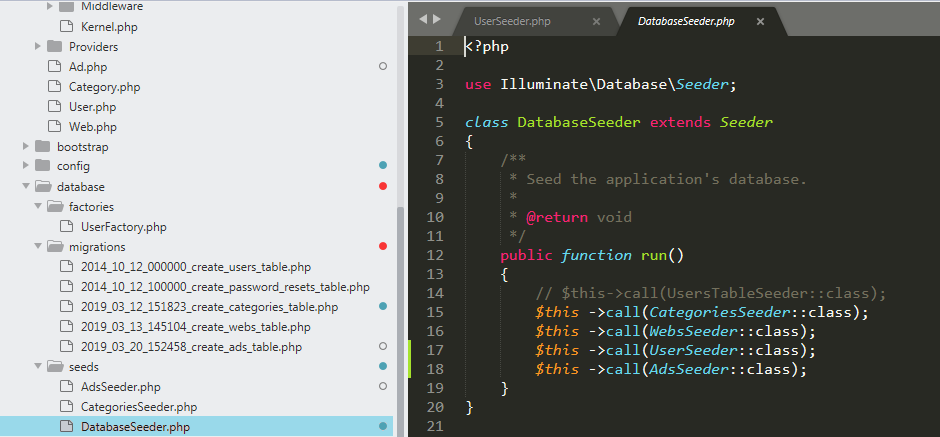
Rollback para hacer un migrate limpio y mnigrate de todo



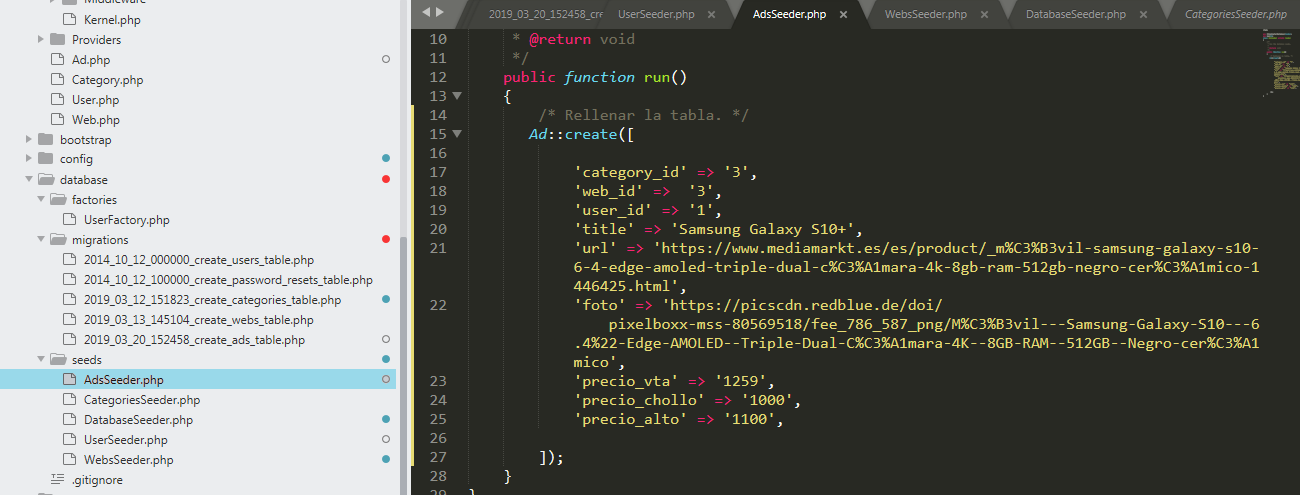
Crear el Seeder para rellenar datos



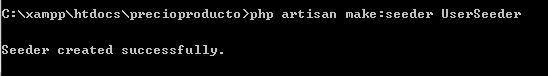
Añadimos la línea para que luego rellene en DataSeeder



En AdSeeder añadimos los campos con los datos que queremos rellenar la tabla

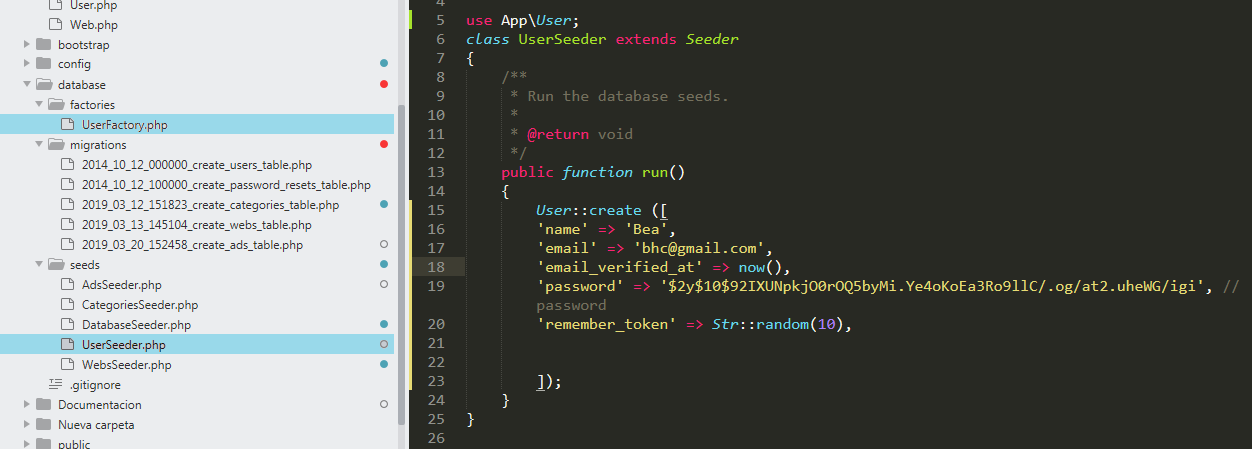


Faltaba el seeder de User

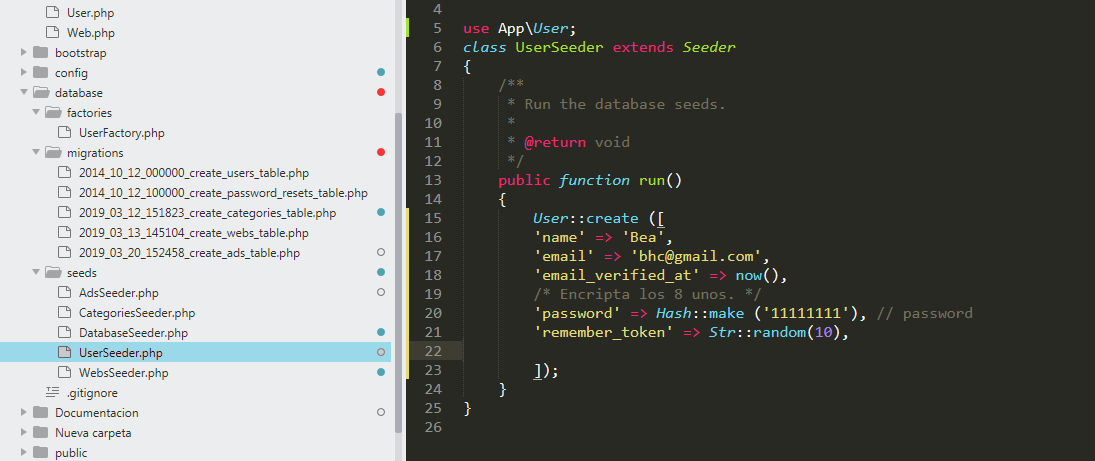


Como la clase user la crea el sistema… m,ejor seguir sus campos de la clase

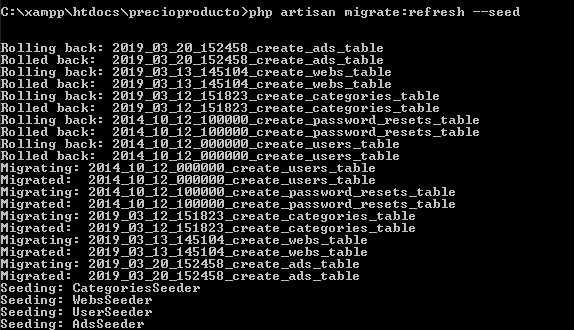
CreaMOS seeder para usuarios… copiamos los campos de pasword… de userfactory para que nos ponga por defecto el pasword ese con su token….



Lugo, modificamos el pasword para poner el pasword que nosotros queremos con la función hash, que encripta el pasword que le indicamos



Php artisan migrate:rollback



Rellena todas las tablas

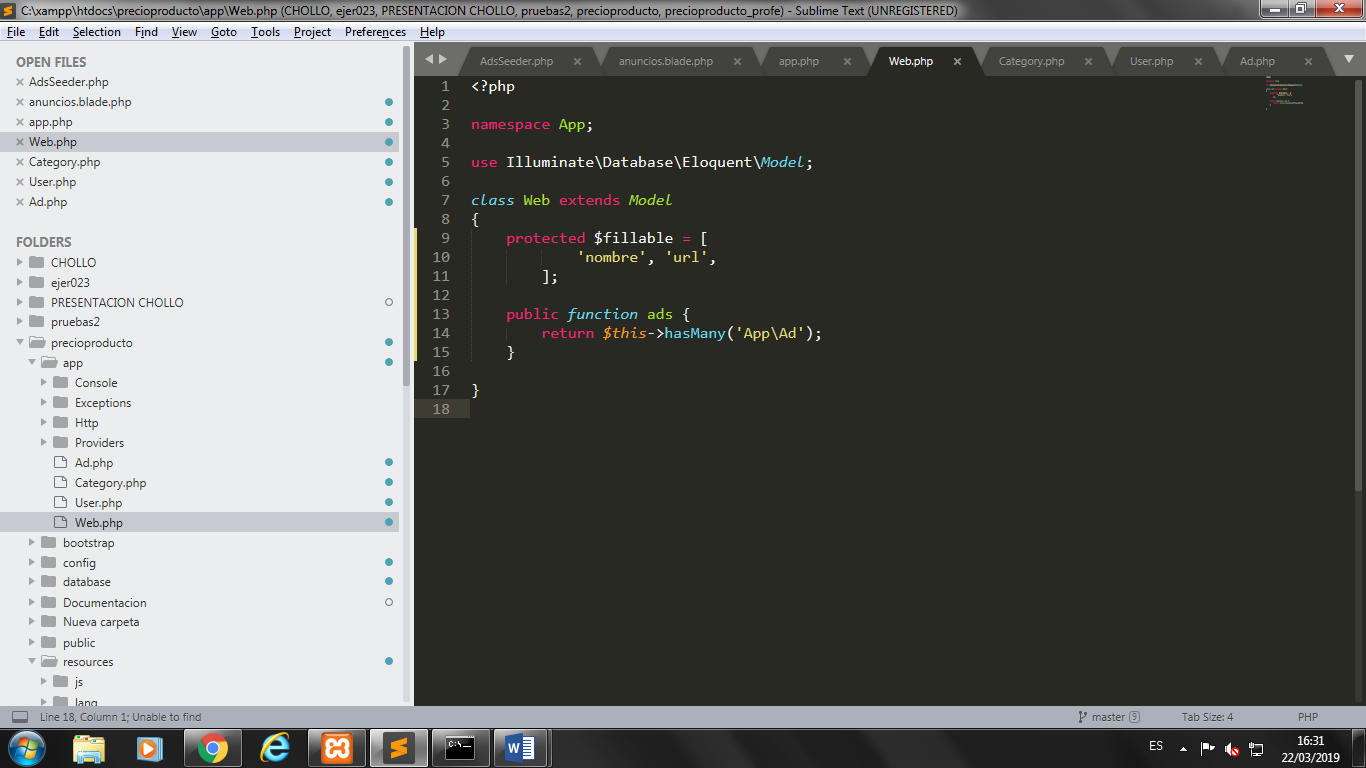
Mostrar en public los dos anuncios: url y foto

**Relaciones**

De 1 a muchos ---- 1-n

Se pone en el nombre de la función el nombre de la clase (de relación n –a muichos) su nombre en plurarl… en este caso ads ) -> anuncios

Y se declara *return $this->hasMany('App\Ad'); -> Donde Ad es la clase anuncios*



De 1 a 1

