

Proyecto Integrado

CarMotor.es

Carlos José Martínez Sánchez
30 de mayo del 2022

Índice

Índice	1
Introducción	3
Motivación	3
Descripción funcional de la aplicación	3
Características y entorno de desarrollo	4
Elementos de investigación	4
Diseño de la base de datos	4
Documentación de la aplicación	5
Creación de recursos principales	5
Modelos, controladores, migraciones, Factories y Seeders.	5
Uso de recursos	9
Creación de tablas	9
Creación de datos ficticios	10
Configuración de controladores	11
Archivo de rutas	12
Manual de usuario	13
Página principal	13
Login y register	15
Diseño	19
Maquetación	19
Multimedia	25
Usabilidad	25
Visibilidad del estado del sistema	26
Relación con el mundo real	27
Control y libertad del usuario	27
Prevención de errores	28
Reconocimiento antes que recuerdo	29
Flexibilidad y eficiencia de uso	30
Estética y diseño minimalista	31
Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de errores	31
Ayuda y documentación	31

Despliegue en AWS	32
Conclusiones finales	33
Problemas encontrados y soluciones	33
Anexo	40

Introducción

El proyecto CarMotor.es consiste en una web de venta de productos relacionados con el mundo del motor, productos tales como frenos, transmisiones, o motores.

El usuario podrá buscar en un amplio catálogo de piezas con el fin de buscar la que más se adapte a sus necesidades, junto con un proceso de compra sencillo, amigable e intuitivo.

Motivación

La motivación para este proyecto viene dada principalmente por mi pasión por el mundo del motor. Desde el mero hecho de conducir hasta saber cómo funciona un vehículo por dentro, cómo es capaz un freno de detener el vehículo, como funciona una transmisión,...

Descripción funcional de la aplicación

Por parte de los usuarios, estos podrán buscar y comprar entre una gran variedad de piezas de vehículos, pudiendo añadir los que mas les interesen a una lista de deseos por si en algún momento estos bajan de precio y terminan adquiriendolos. También dispondrán de un apartado donde podrán editar su perfil de usuario para poder añadir una foto de perfil y cambiar campos tales como su nombre y apellidos, su dirección de correo electrónico, su contraseña e incluso su dirección, todo ello bien indicado para que el usuario no tenga ninguna duda. También dispondrán de un carrito de compra con el que poder realizar varias compras en una.

En cuanto a los administradores, estos tendrán todo lo mencionado anteriormente y además una sección donde podrán administrar todo lo relacionado con la web. Aspectos como la gestión de los usuarios, de los productos, de las marcas e incluso de las facturas podrán ser administrados desde la propia web.

Características y entorno de desarrollo

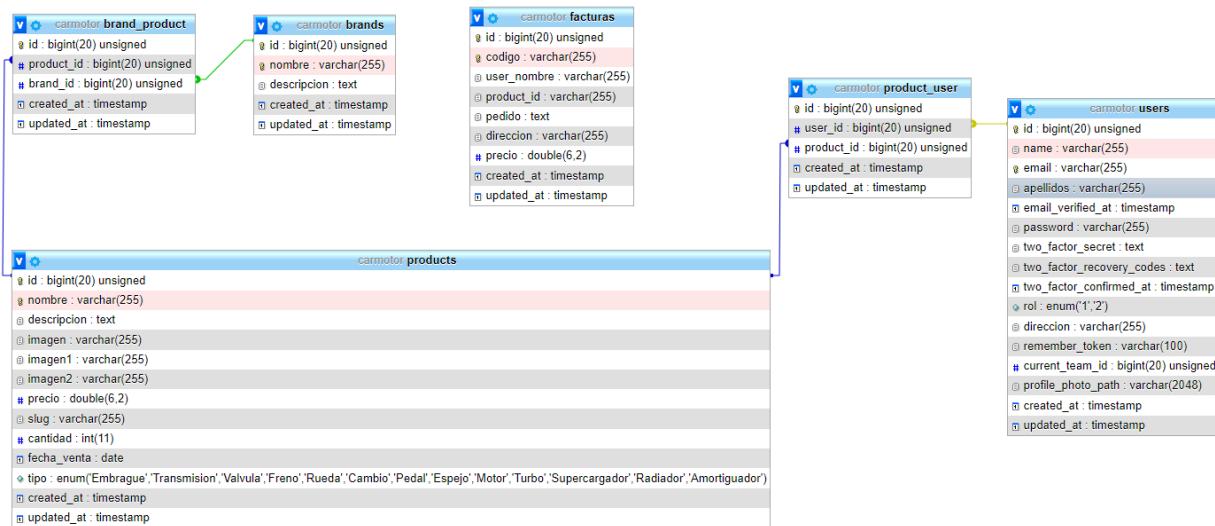
Como entorno de desarrollo he usado mi ordenador que dispone de Windows 10 pro junto con los programas Visual Studio Code y XAMPP

Elementos de investigación

Como parte de la investigación resaltar la creación de una página de error 404, la exportación de una tabla en formato csv, aprender en profundidad el framework de laravel y cómo se organiza un sistema CMS.

Diseño de la base de datos

El diseño de la base de datos es el siguiente



Documentación de la aplicación

Creación de recursos principales

Modelos, controladores, migraciones, Factories y Seeders.

Los modelos son los siguientes

User.php

Como este modelo ya lo realiza JetStream, solo mostraré la parte de las relaciones



```
app > Models > User.php > PHP Intelephense > User > $fillable
72
73     /**
74      * Get the URL to the user's profile photo.
75      *
76      * Función que ha sido sobreescrita que aunque hace exactamente lo mismo, esta puede ser llamada por la variable $user para poder mostrarse la imagen
77      * por defecto en caso de que dicho usuario no la tenga
78      *
79      * @return string
80      */
81     public function getProfilePhotoUrlAttribute()
82     {
83         return $this->profile_photo_path ? Storage::url($this->profile_photo_path) : $this->defaultProfilePhotoUrl();
84     }
85
86     /* Función que se relaciona con la tabla Products */
87     public function products()
88     {
89         return $this->belongsToMany(Product::class);
90     }
91
92     /* Relacion con la tabla UserProducts */
93     public function userProducts()
94     {
95         return $this->hasMany(UserProduct::class);
96     }
97 }
98 }
```

Product.php

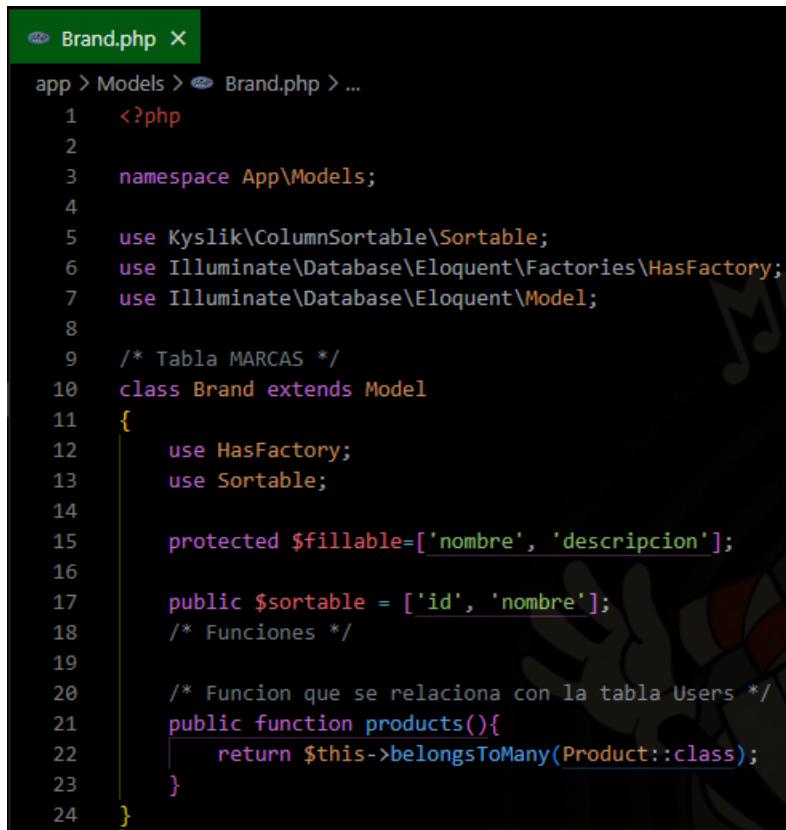
```
Product.php X
app > Models > Product.php > PHP Intelephense > Product > scopeNombre
3   namespace App\Models;
4
5   use Illuminate\Database\Eloquent\Factories\HasFactory;
6   use Illuminate\Database\Eloquent\Model;
7   use Kyslik\ColumnSortable\Sortable;
8
9   class Product extends Model
10  {
11      use HasFactory;
12      protected $fillable=['nombre', 'descripcion', 'imagen', 'imagen1', 'imagen2', 'precio', 'cantidad', 'fecha_venta', 'tipo', 'slug'];
13      use Sortable;
14
15      /* Funciones */
16
17      /**
18       * Get the route key for the model.
19       *
20       * @return string
21       */
22      public function getRouteKeyName()
23      {
24          return 'slug';
25      }
26
27      /* Funcion que se relaciona con la tabla Users */
28      public function users(){
29          return $this->belongsToMany(User::class);
30      }
31
32      /* Funcion que se relaciona con la tabla Brands */
33      public function brands(){
34          return $this->belongsToMany(Brand::class);
35      }
36
37      //Metodos scopes
38      public function scopeNombre($query, $v){
39          if(!isset($v)){
40              return $query->where('nombre', 'like', '%');
41          }
42          return $query->where('nombre', 'like', "%$v%");
43      }
44
45      public function scopeTipos($query, $v){
46          if($v=="todos" || !isset($v)){
47              return $query->where('tipo', 'like', "%");
48          }
49          return $query->where('tipo' , $v);
50      }
51 }
```

Factura.php



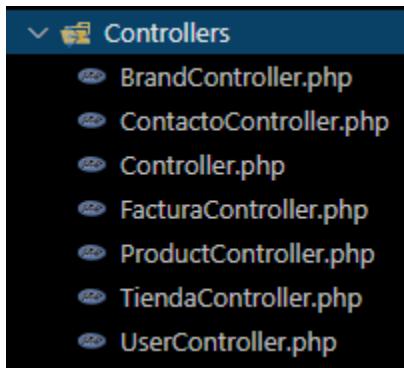
```
Factura.php X
app > Models > Factura.php > ...
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7 use Kyslik\ColumnSortable\Sortable;
8
9 class Factura extends Model
10 {
11     use HasFactory;
12
13     protected $fillable=['codigo', 'user_nombre', 'direccion', 'product_id', 'pedido', 'precio'];
14     use Sortable;
15
16     //Metodos scopes
17     public function scopeCodigo($query, $v){
18         if(!isset($v)){
19             return $query->where('codigo', 'like', '%');
20         }
21         return $query->where('codigo', 'like', "%$v%");
22     }
23 }
24 }
```

Brands.php (Marcas)

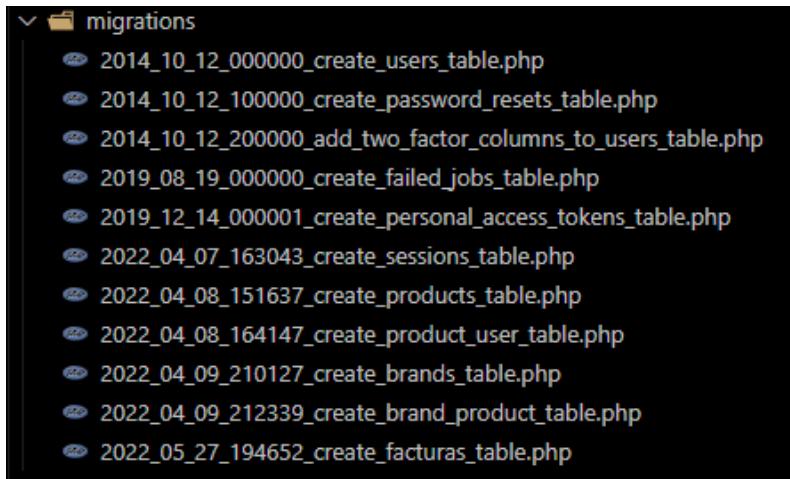


```
Brand.php X
app > Models > Brand.php > ...
1 <?php
2
3 namespace App\Models;
4
5 use Kyslik\ColumnSortable\Sortable;
6 use Illuminate\Database\Eloquent\Factories\HasFactory;
7 use Illuminate\Database\Eloquent\Model;
8
9 /* Tabla MARCAS */
10 class Brand extends Model
11 {
12     use HasFactory;
13     use Sortable;
14
15     protected $fillable=['nombre', 'descripcion'];
16
17     public $sortable = ['id', 'nombre'];
18     /* Funciones */
19
20     /* Funcion que se relaciona con la tabla Users */
21     public function products(){
22         return $this->belongsToMany(Product::class);
23     }
24 }
```

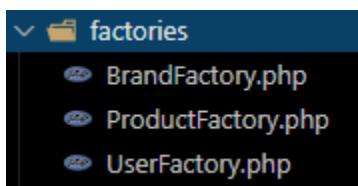
En cuanto a los controladores mostraré solo los que hay



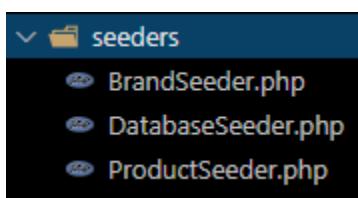
De igual forma haré con las migraciones



Igual con los factories



Pasamos a los Seeders



Uso de recursos

Creación de tablas

Para la creación de las tablas se ha hecho uso de las migraciones. Mostraré la migración de la tabla Products como ejemplo. Como bien se indica en un comentario, existe un campo de la tabla que solo se usará en la url para mejorar el posicionamiento en la red.



```
database > migrations > 2022_04_08_151637_create_products_table.php > ...
1  <?php
2
3  use Illuminate\Database\Migrations\Migration;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Support\Facades\Schema;
6
7  return new class extends Migration
8  {
9      /**
10      * Run the migrations.
11      *
12      * @return void
13      */
14      public function up()
15  {
16          Schema::create('products', function (Blueprint $table) {
17              $table->id();
18
19              /* Campos */
20              $table->string('nombre')->unique();
21              $table->text('descripcion');
22              /* Imagenes. 1 de ellas es obligatoria, las otras 2 opcionales */
23              $table->string('imagen');
24              $table->string('imagen1')->nullable();
25              $table->string('imagen2')->nullable();
26              $table->float('precio', 6, 2); /* 6 cifras de las cuales 2 son decimales */
27              /* */
28
29              /* Campo para el SEO (Este campo lo usaremos en la url para el posicionamiento SEO) */
30              $table->string('slug')->nullable();
31              /* */
32
33              $table->integer('cantidad'); /* Cantidad de stock de este producto */
34              $table->date('fecha_venta');
35              $table->enum('tipo', ['Embrague', 'Transmision', 'Valvula', 'Freno', 'Rueda', 'Cambio', 'Pedal', 'Espejo', 'Motor', 'Turbo', 'Supercargador', 'Radiador', 'Amortiguador']);
36              /* Cambio = Caja de cambios */
37
38              $table->timestamps();
39          });
40
41      /**
42      * Reverse the migrations.
43      *
44      * @return void
45      */
46      public function down()
47  {
48          Schema::dropIfExists('products');
49      }
49 }
```

Las demás migraciones siguen la misma estela, cada uno con sus respectivos campos.

Creación de datos ficticios

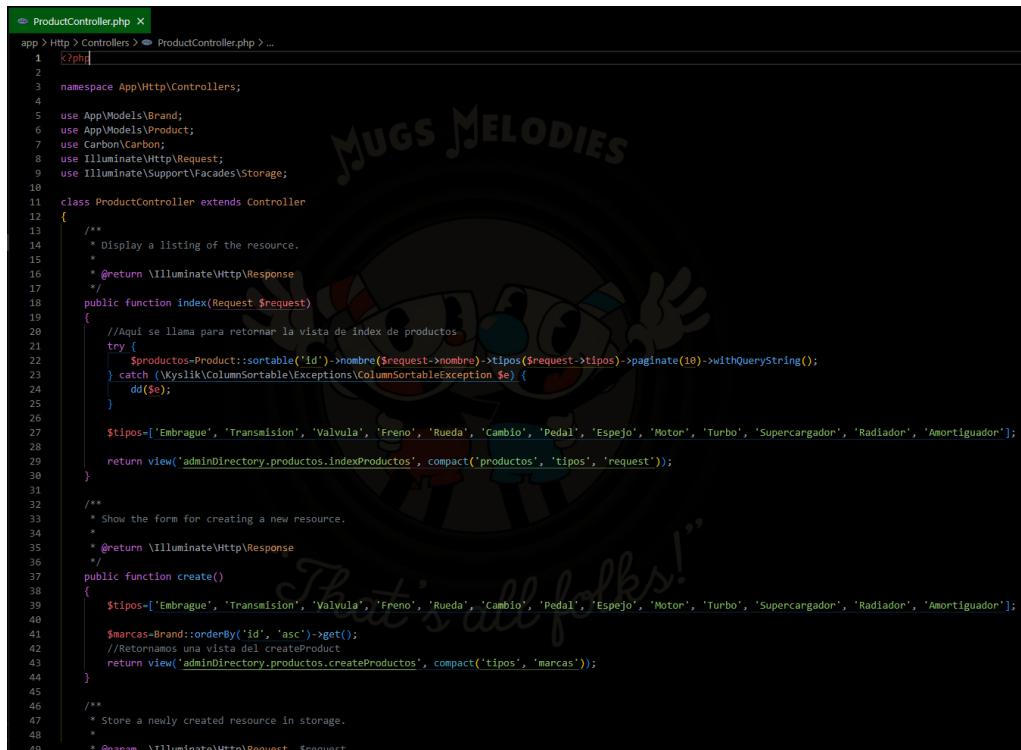
Para ello he usado Faker, que nos permite generar datos de prueba en la base de datos. Mostraré cómo se han generado datos de prueba para la tabla Users.



```
UserFactory.php
database > factories > UserFactory.php > ...
1  <?php
2
3  namespace Database\Factories;
4
5  use App\Models\Team;
6  use App\Models\User;
7  use Illuminate\Database\Eloquent\Factories\Factory;
8  use Illuminate\Support\Str;
9  use Laravel\Jetstream\Features;
10
11 class UserFactory extends Factory
12 {
13     /**
14      * The name of the factory's corresponding model.
15      *
16      * @var string
17      */
18     protected $model = User::class;
19
20     /**
21      * Define the model's default state.
22      *
23      * @return array
24      */
25     public function definition()
26     {
27         return [
28             'name' => $this->faker->name(),
29             'apellidos'=>$this->faker->name(),
30             'email' => $this->faker->unique()->safeEmail(),
31             'email_verified_at' => now(),
32             'password' => '$2y$10$92IXUNpkj00rQ5byMi.Ye4oKoEa3Ro9llC/.og/at2.uheWG/igi', // password
33
34             'rol'=>random_int(1,2), /* 1 = Usuario normal || 2 = Usuario Admin */
35             'direccion'=>$this->faker->address(), /* Genera una direccion aleatoria */
36
37             'remember_token' => Str::random(10),
38         ];
39     }
40 }
```

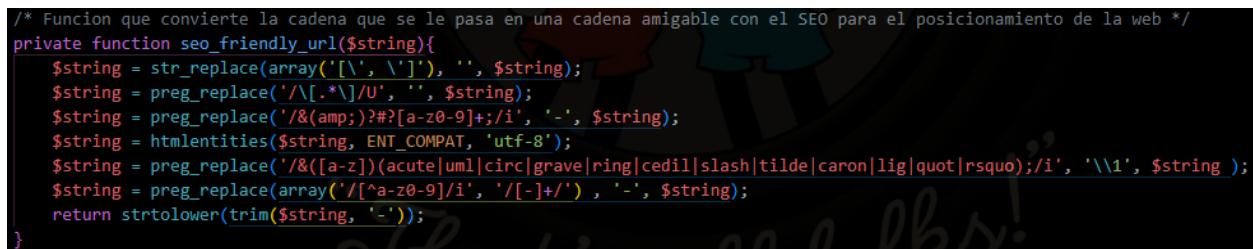
Configuración de controladores

Los controladores son una parte importante de la web ya que son los encargados de manejar las distintas acciones que se acontecen en la misma. Para este caso mostraré una parte del controlador de Productos, en mi opinión el más importante de todos.



```
ProductController.php
app > Http > Controllers > ProductController.php > ...
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use App\Models\Brand;
6 use App\Models\Product;
7 use Carbon\Carbon;
8 use Illuminate\Http\Request;
9 use Illuminate\Support\Facades\Storage;
10
11 class ProductController extends Controller
12 {
13     /**
14      * Display a listing of the resource.
15      *
16      * @return \Illuminate\Http\Response
17     */
18    public function index(Request $request)
19    {
20        //Aqui se llama para retornar la vista de index de productos.
21        try {
22            $productos=Product::sortable('id')->nombre($request->nombres)->tipos($request->tipos)->paginate(10)->withQueryString();
23        } catch (\Kyslik\ColumnSortable\Exceptions\ColumnSortableException $e) {
24            dd($e);
25        }
26
27        $tipos=['Embrague', 'Transmision', 'Valvula', 'Freno', 'Rueda', 'Cambio', 'Pedal', 'Espejo', 'Motor', 'Turbo', 'Supercargador', 'Radiador', 'Amortiguador'];
28
29        return view('adminDirectory.productos.indexProductos', compact('productos', 'tipos', 'request'));
30    }
31
32    /**
33     * Show the form for creating a new resource.
34     *
35     * @return \Illuminate\Http\Response
36     */
37    public function create()
38    {
39        $tipos=['Embrague', 'Transmision', 'Valvula', 'Freno', 'Rueda', 'Cambio', 'Pedal', 'Espejo', 'Motor', 'Turbo', 'Supercargador', 'Radiador', 'Amortiguador'];
40
41        $marcas=Brand::orderBy('id', 'asc')->get();
42        //Retornamos una vista del createProduct
43        return view('adminDirectory.productos.createProductos', compact('tipos', 'marcas'));
44    }
45
46    /**
47     * Store a newly created resource in storage.
48     *
49     * @param \Illuminate\Http\Request $request
50     */
51    public function store(Request $request)
52    {
53        $product = new Product();
54        $product->fill($request->all());
55        $product->save();
56
57        return redirect()->route('admin.productos.index');
58    }
59}
```

Quiero mostrar también esta última función, importante para el posicionamiento SEO debido a que se encarga de tratar el campo “slug” mostrado anteriormente en la migración de Productos para crear una URL amigable y mejorar el posicionamiento de la web debido a que es un formato que Google apremia mucho.



```
/*
 * Funcion que convierte la cadena que se le pasa en una cadena amigable con el SEO para el posicionamiento de la web */
private function seo_friendly_url($string){
    $string = str_replace(array('[\'', '\']'), '', $string);
    $string = preg_replace('/[.*]U', '', $string);
    $string = preg_replace('/&(amp;)?#[a-z0-9]+;/i', '-', $string);
    $string = htmlentities($string, ENT_COMPAT, 'utf-8');
    $string = preg_replace('/&([a-z])(acute|uml|circ|grave|ring|cedil|slash|tilde|caron|lig|quot|rsquo);/i', '\\1', $string );
    $string = preg_replace(array('/[a-z0-9]/i', '/[-]+/'), '-', $string);
    return strtolower(trim($string, '-'));
}
```

Esta función recibe una cadena y la modifica quitando los caracteres especiales y reemplazando todo lo que no sea letra y número por un guión, generando URLs como la siguiente

(i) 127.0.0.1:8000/tienda/tempora-molestias-autem-inventore-dolores-ratione-eligendi-dignissimos

Como bien he comentado antes, este formato de URL es muy premiado por Google, mejorando mucho nuestro posicionamiento en la web.

Archivo de rutas

Este archivo es sumamente importante ya que es el que nos permite navegar por la web. El archivo es el siguiente:

```
routes > web.php > ...
routes > web.php > ...
22 */
23
24 Route::get('/', function () {
25     /* Desde laravel 5.3 es posible pasarle un array al where. En este caso queremos que se escogen 8 elementos aleatorios de entre todos los que cumplen las 2 condiciones */
26     $piezasEnEscasez=Product::where([
27         ['cantidad', '<=', 3],
28         ['cantidad', '!=', 0]
29     ])->get()->random(10);
30 }
31
32     /* Carbon es una api que facilita las cosas con el formato DateTime */
33     $piezasNuevas=Product::where('cantidad', '!=', 0)->whereDate('fecha_venta', '<=', Carbon::now()->add(-10, 'day')->format('Y-m-d'))->orderBy('fecha_venta', 'desc')->get();
34
35     $piezasInteresantes=Product::where('cantidad', '!=', 0)->take(100)->get()->random(10);
36
37     return view('dashboard', compact('piezasEnEscasez', 'piezasInteresantes', 'piezasNuevas'));
38 })->name('index');

39 /* Vistas de la tienda principal */
40 Route::get('/tienda', [TiendaController::class, 'tienda'])->name('tienda');
41 Route::get('/tienda/{product:slug}', [TiendaController::class, 'tiendaProducto'])->name('tienda.producto');
42 Route::middleware(['auth:sanctum', 'verified'])->get('/tienda/addDeseo/{product:slug}', [TiendaController::class, 'addDeseo'])->name('tienda.addDeseo');
43 Route::middleware(['auth:sanctum', 'verified'])->get('/tienda/addCarrito/{products:slug}', [TiendaController::class, 'addCarrito'])->name('tienda.addCarrito');
44 Route::middleware(['auth:sanctum', 'verified'])->get('/listaDeseos', [TiendaController::class, 'listaDeseos'])->name('tienda.deseos');
45 Route::get('/comprarProducto/{product}', [TiendaController::class, 'comprarProducto'])->name('tienda.comprarProducto');
46 Route::post('/comprarProducto/{product}', [TiendaController::class, 'procesarCompra'])->name('tienda.procesarProducto');
47
48 /* Envio de formulario de contacto */
49 Route::get('/contacto', [ContactoController::class, 'pintarFormulario'])->name('contacto.pintar');
50 Route::post('/contacto', [ContactoController::class, 'procesarFormulario'])->name('contacto.procesar');
51 /*
52 */
53 /* Zona donde colocaremos los enlaces donde solo podrán acceder los admins ademas de tener que verificar la cuenta */
54
55 Route::middleware(['role', 'verified'])->resource('products', ProductController::class); # Carga todas las rutas de Products
56
57
58 Route::middleware(['role', 'verified'])->get('/facturas', [FacturaController::class, 'index'])->name('facturas.index');
59 Route::middleware(['role', 'verified'])->get('/facturas/{factura}', [FacturaController::class, 'show'])->name('facturas.show');
60 Route::middleware(['role', 'verified'])->get('/facturas/csv', [FacturaController::class, 'exportarCsv'])->name('facturas.exportar');
61
62 Route::middleware(['role', 'verified'])->resource('/admin/brands', BrandController::class); # Carga todas las rutas de Brand
63
64 Route::middleware(['role', 'verified'])->resource('/admin/users', UserController::class); # Carga todas las rutas de Users
65 Route::middleware(['role', 'verified'])->get('/admin/users/rol/{user}', [UserController::class, 'rol'])->name('users.rol'); #Llama al metodo para cambiar su rol
66
67 Route::middleware(['role', 'verified'])->get('/admin', function () {
68     return view('adminDirectory.index'); # Carga la vista del panel de administrador del admin
69 })->name('admin');
```

Manual de usuario

Página principal

En cuanto a la parte de usuario, no hay mucha novedad respecto a otras páginas web. De primeras el usuario se encontrará con lo siguiente

Listado de productos

Piezas a punto de agotarse

Imagen	Nombre	Descripción	Precio	Estado
	Cambio	et quo sed nam voluptas dignissimos iste nesciunt	3534.01 €	Solo quedan 2 en stock!!
	Rueda	tempora molestias autem inventore dolores ratione eligendi dignissimos	4824.1 €	Solo quedan 1 en stock!!
		rem dicta rerum eveniet in impedit dolor facere	3267.89 €	Cambio
	Freno	mollitia voluptatem blanditiis consectetur et aliquam sit voluptas	3123.76 €	Solo quedan 1 en stock!!
	Embrague	et tempora alias qui dolor enim rerum rerum	6325.05 €	Solo quedan 2 en stock!!

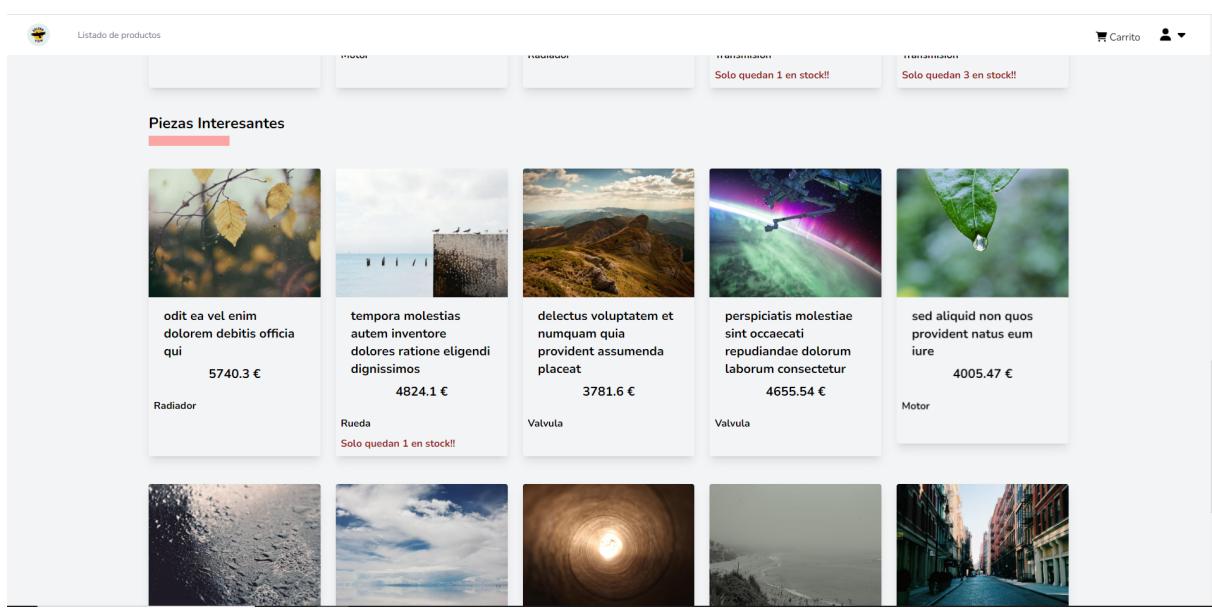
pariatur quis dolores tempore aut commodi distinctio esse nesciunt distinctio et soluta sint doloribus

Listado de productos

Piezas Nuevas

Imagen	Nombre	Descripción	Precio	Estado
	Amortiguador	unde eligendi velit et repellendus qui similique excepturi	1905.25 €	Solo quedan 1 en stock!!
	Cambio	ratione optio maiores et est est ipsam molestias	6231.56 €	
	Amortiguador	praesentium earum quia iure sit quia quaerat ut	4589.46 €	
	Espejo	veniam distinctio quo necessitatibus odio ipsam est exercitationem	8217.43 €	
	Radiador	aut nostrum nam modi molestiae ut ea quaerat	6153.66 €	

debitis sunt adipisci odio illo rem iure porro unde animi nisi nam asperiores odit voluptas consequatur eligendi quod iure sapiente culpa sunt accusantium eum est omnis dignissimos nesciunt distinctio et eius tenetur tempore



Los productos se encuentran separados en 3 secciones. En la primera se muestran unos productos aleatorios a punto de agotarse. En la siguiente sección se muestran los productos que se han añadido recientemente y por último, una selección variada de productos. (Se ha puesto el tipo de cada pieza debido a que son datos de prueba).

El usuario podrá hacer click en cualquiera de ellos y se le redirigirá a la página del producto donde podrá realizar diversas acciones.

La visualización de un producto es la siguiente

et quo sed nam voluptas dignissimos iste nesciunt
3534.01 €

Añadir al carrito Comprar

Compatible con las siguientes marcas

Toyota Honda Peugeot Renault Mercedes-Benz Opel Citroen Seat Dodge Ford Mercedes-AMG Fiat Audi Suzuki Dacia

Año de fabricación: 2022

Descripción del producto.
Hic saepe quidem dicta ab enim. Dolorum et sit sint voluptatem autem et. Quia cum omnis ad et.

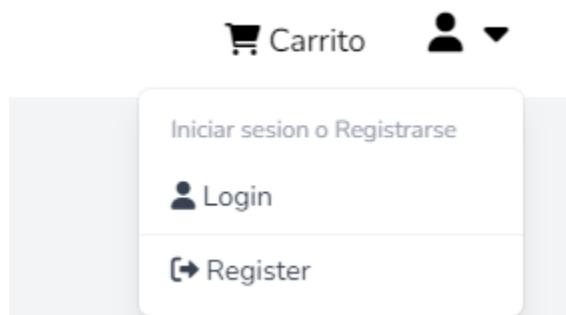
Señalar la URL que utiliza, como bien mencioné anteriormente es una buena práctica para el posicionamiento web.

Login y register

El login y el registro es una parte fundamental de la web ya que nos permite tener una base de compradores. En la web de CarMotor las acciones de Login y Register se encuentran de la siguiente forma



Al hacer click nos aparece lo siguiente



Aquí el usuario podrá loguearse o registrarse.

El formulario de login es el siguiente

A screenshot of a login form. At the top, there is a decorative image of a car wheel. Below it is a white input field labeled "Email" with a placeholder "Email" and a blue border. Underneath is a password input field labeled "Password" with a blue border. At the bottom left is a "Remember me" checkbox. At the bottom right are two buttons: "Forgot your password?" and a dark "LOG IN" button.

(El logo está pendiente de cambio)

Por otro lado el register es el siguiente

Nombre

Apellidos

Email

Calle

CP

Poblacion

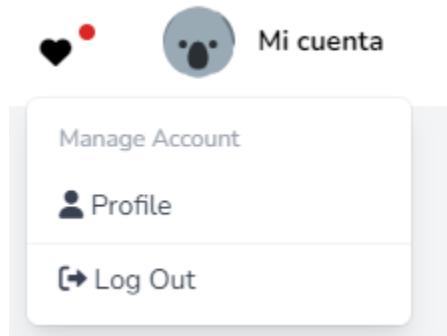
Provincia

Contraseña

Confirmar Contraseña

[Already registered?](#) **REGISTER**

Una vez el usuario se ha registrado o ha iniciado sesión, donde antes se encontraban los botones anteriores ahora aparecen los siguientes



Se cambia el ícono anterior por la foto de perfil del usuario y aparecen las siguientes acciones, estas son para ver el perfil de usuario y por supuesto, cerrar sesión. (También aparece un corazón que al hacer click nos llevará a los productos que dicho usuario tiene en la lista de deseos, el punto rojo indica que hay productos en dicha lista)

Al meternos en el perfil nos saldrá lo siguiente

A screenshot of a profile editing form. At the top left is the heading "Informacion del perfil" and a note "Actualizar la cuenta's profile information and email address.". The main area contains fields for "Foto" (with a placeholder image of a koala), "Nombre" (with the value "Carlos"), "Apellidos" (with the value "adminadmin"), and "Email" (with the value "carlosjmsanchez@gmail.com"). Below these fields are two buttons: "SELECCIONAR NUEVA FOTO" and "BORRAR FOTO". At the bottom right is a large "GUARDAR" button.

Update Direction

Change your direction

Direccion Actual: macaco n7, 04230, almeria, almeria

Calle
Calle, N°

CP
Codigo Postal

Poblacion
Poblacion

Provincia
Provincia

SAVE

También podremos cambiar nuestra contraseña, entre otras funciones.

Por otro lado y ya para culminar este apartado, el usuario tiene la posibilidad de ponerse en contacto con nosotros mediante un pequeño formulario ubicado en el pie de página.



Al hacer click, el usuario será llevado a un formulario donde deberá de poner su nombre, correo electrónico y la pregunta o problema que pueda tener

Formulario de contacto

Nombre
Carlos

Email
carlosjmsanchez@gmail.com

Indicanos el motivo de tu consulta

Enviar **Regresar**

Si el usuario está logueado, los campos de nombre y correo serán rellenados automáticamente, en caso de no estar logueado se deberán de llenar.

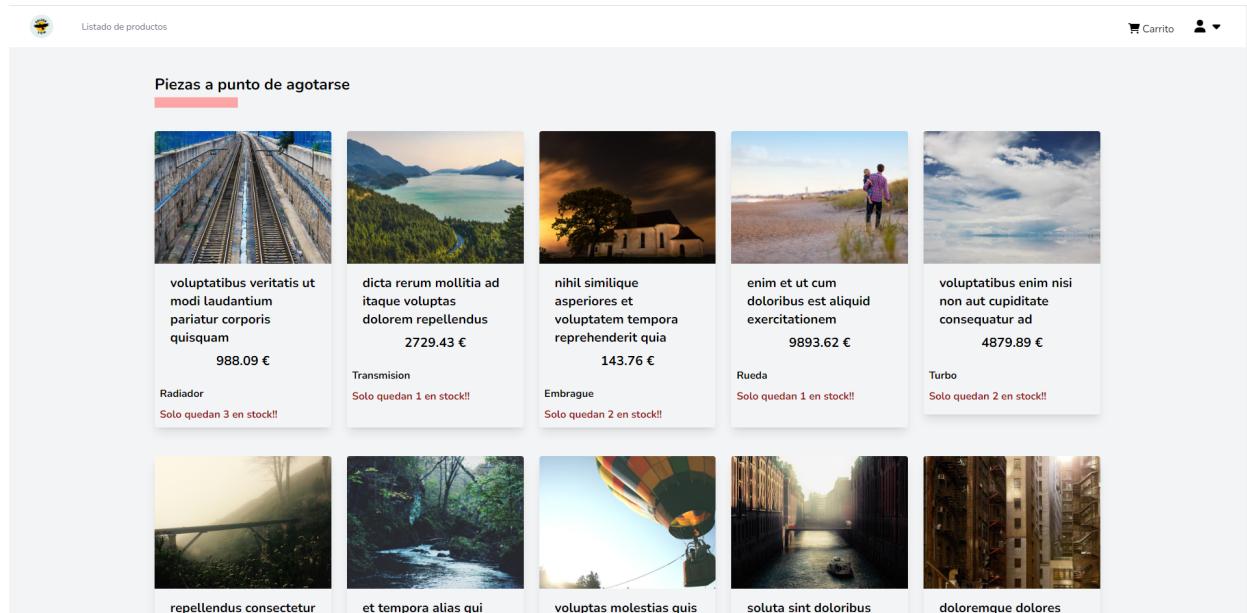
Diseño

Maquetación

La maquetación es un aspecto importante de la web ya que nos permite adaptarla a distintos dispositivos. En la web de CarMotor la web es responsive la parte del usuario. Esto es debido a que el administrador trabajará en una pantalla de escritorio, pero en cambio el usuario normal podrá navegar tanto en un ordenador como en un dispositivo móvil.

Mostraré el diseño responsive de la página principal y de la visualización de un producto.

Vista normal



The screenshot shows a responsive product listing page. At the top, there's a header with a user icon, 'Listado de productos', a shopping cart icon labeled 'Carrito', and a profile icon. Below the header, a red bar highlights the category 'Piezas a punto de agotarse' (Parts at risk of running out). The main content area displays a grid of five product cards in a row, with two more rows below it. Each card contains a small image, a title, a description, a price, and a stock status message.

Imagen	Título	Descripción	Precio	Stock
	Radiador	voluptatibus veritatis ut modi laudantium pariatur corporis quisquam	988.09 €	Solo quedan 3 en stock!!
	Transmision	dicta rerum mollitia ad itaque voluptas dolorem repellendus	2729.43 €	Solo quedan 1 en stock!!
	Embrague	nihil similique asperiores et voluptatem tempora reprehenderit quia	143.76 €	Solo quedan 2 en stock!!
	Rueda	enim et ut cum doloribus est aliquid exercitationem	9893.62 €	
	Turbo	voluptatibus enim nisi non aut cupiditate consequatur ad	4879.89 €	Solo quedan 2 en stock!!
		repellendus consectetur		
		et tempora alias qui		
		voluptas molestias quis		
		soluta sint doloribus		
		dolore que dolores		

Vista en pantallas anchas

Dimensions: Reactivo ▾ 1019 x 768 100% Sin limitaciones

Listado de productos Carrito

Piezas a punto de agotarse

 <p>voluptatibus veritatis ut modi laudantium pariatur corporis quisquam</p> <p>988.09 €</p> <p>Radiador <i>Solo quedan 3 en stock!!</i></p>	 <p>dicta rerum mollitia ad itaque voluptas dolorem repellendus</p> <p>2729.43 €</p> <p>Transmision <i>Solo quedan 1 en stock!!</i></p>	 <p>nihil similique asperiores et voluptatem tempora reprehenderit quia</p> <p>143.76 €</p> <p>Embrague <i>Solo quedan 2 en stock!!</i></p>	 <p>enim et ut cum doloribus est aliquid exercitationem</p> <p>9893.62 €</p> <p>Rueda <i>Solo quedan 1 en stock!!</i></p>
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

== //

Vista en pantallas medianas

Dimensiones: Reactivo ▾ 765 × 768 100% ▾ Sin limitaciones ▾

Listado de productos Carrito

Piezas a punto de agotarse



voluptatibus
veritatis ut modi
laudantium
pariatur corporis
quisquam

988.09 €

Radiador

Solo quedan 3 en stock!!



dicta rerum
mollitia ad itaque
voluptas dolorem
repellendus

2729.43 €

Transmision

Solo quedan 1 en stock!!



nihil similique
asperiores et
voluptatem
tempora
reprehenderit quia

143.76 €

Embrague

Solo quedan 2 en stock!!



Y por último, vista en pantalla de móvil

Dimensiones: Samsung Galaxy S20 Ultra ▾ 412 x 915 92% ▾ Sin limitaciones ▾

Piezas a punto de agotarse

	
voluptatibus veritatis ut modi laudantium pariatur corporis quisquam	dicta rerum mollitia ad itaque voluptas dolorem repellendus
988.09 €	2729.43 €
Radiador	Transmision
Solo quedan 3 en stock!!	Solo quedan 1 en stock!!





Ahora mostraré la vista de un producto cualquiera.

Vista normal

The screenshot shows a product listing for a car. The main image is a scenic view of a lake surrounded by mountains. The title above the image reads "dicta rerum mollitia ad itaque voluptas dolorem repellendus". Below the title is the price "2729.43 €". To the left of the price is a heart icon, and to the right is a "Añadir al carrito" button. A large blue "Comprar" button is positioned at the bottom right of the listing. Below the listing, there is a section titled "Compatible con las siguientes marcas" with buttons for "Audi", "Citroen", and "Dacia". The text "Año de fabricación: 2022" is also present. At the bottom of the page, there is a "Descripción del producto" section with placeholder text: "Facilis et ex fugiat beatae corporis. Deleniti necessitatibus animi voluptas necessitatibus." The top navigation bar includes a user icon, "Listado de productos", and a "Carrito" icon. The footer contains the CarMotor.es logo, social media links, and copyright information: "© 2022 CarMotor. Todos los derechos Reservados."

Vista en pantallas anchas

This screenshot shows the same product listing as the previous one, but it is displayed on a wider screen. The horizontal scroll bars at the bottom indicate that the page content continues beyond the visible area. The layout is identical to the "Vista normal" version, featuring the scenic image, title, price, and purchase buttons. The "Descripción del producto" section and footer are also present. The top navigation bar and footer are consistent with the first screenshot.

Y por último, vista en pantalla de móvil

Dimensiones: Samsung Galaxy S20 Ultra ▾ 412 x 915 75% ▾ Sin limitaciones ▾

dicta rerum mollitia ad
itaque voluptas dolorem
repellendus

2729.43 €

Comprar

Compatible con las siguientes marcas

Audi Citroen Dacia

Año de fabricación: 2022

Descripción del producto.

Facilis et ex fugiat beatae corporis. Deleniti

Multimedia

Los ficheros multimedia de la web son únicamente imágenes en formato JPG de los productos aunque el generador de datos falsos de PHP ha generado imágenes de paisajes. Es por esto que en la web se ha indicado el tipo de cada producto para así poder diferenciarlos.

Usabilidad

En cuanto a la gama de colores, ya se ha podido comprobar en las imágenes anteriores. En la web se predomina el color blanco, aunque en algunas secciones hay otros colores como por ejemplo aquí

The screenshot shows a user interface for a product listing. On the left, there is a sidebar with three sections: 'Precio' (Price) with 'Menor a mayor' (Lower to higher) and 'Mayor a menor' (Higher to lower) options; 'Pieza' (Part) with a list of vehicle components like 'Embrague', 'Transmisión', 'Valvula', 'Freno', 'Rueda', 'Cambio', 'Pedal', 'Espejo', 'Motor', 'Turbo', 'Supercargador', 'Radiador', and 'Amortiguador'; and 'Marcas' (Brands) with a list of car brands including 'Audi', 'Citroen', 'Dacia', 'Dodge', 'Fiat', and 'Ford'. The main area displays four product cards. The first card is for a 'Supercargador' (Supercharger) with placeholder text: 'neque dolor iste asperiores qui ea rerum commodi' and price '9988.88 €'. It shows 'Solo quedan 3 en stock!!' and brand logos for Audi and Citroen. The second card is for a 'Rueda' (Wheel) with placeholder text: 'enim et ut cum doloribus est aliquid exercitationem' and price '9893.62 €'. It shows 'Solo quedan 1 en stock!!' and brand logos for Audi, Citroen, Dacia, Dodge, Fiat, Ford, Honda, Mercedes-AMG, Mercedes-Benz, and Opel. The third card is for a 'Supercargador' (Supercharger) with placeholder text: 'qui vel quia aliquid et facilis qui assumenda' and price '9866.46 €'. It shows 'Solo quedan 1 en stock!!' and brand logos for Audi, Citroen, Dacia, Dodge, and Fiat. The fourth card is for a 'Transmisión' (Transmission) with placeholder text: 'saepe qui incident et laudantium voluptates beatae corrupti' and price '9858.85 €'. It shows brand logos for Audi, Citroen, Dacia, Dodge, and Fiat. Below the cards, there are small thumbnail images of other products.

También en la visualización del producto



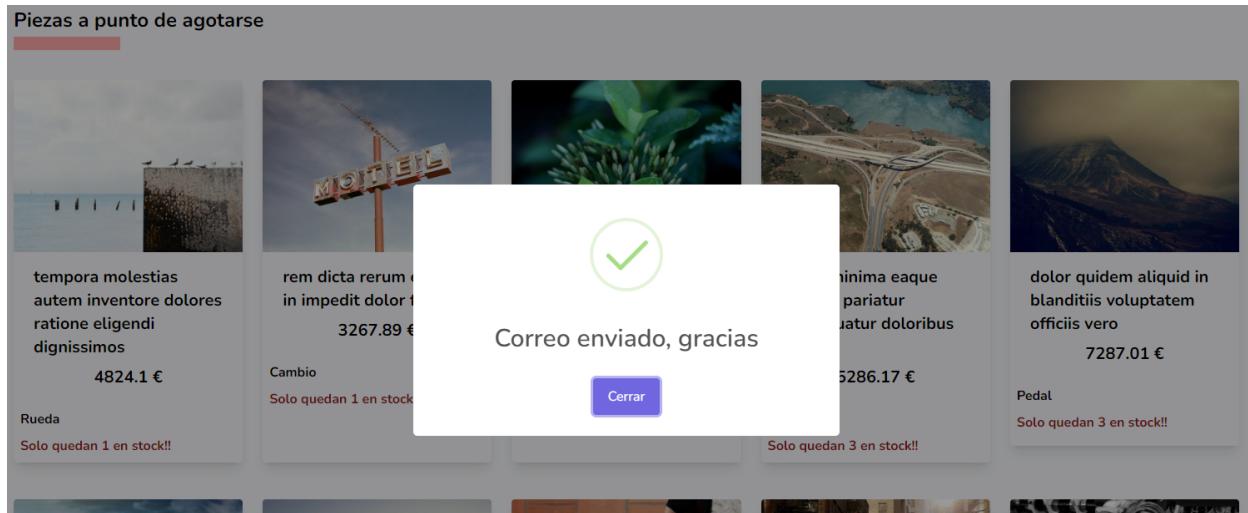
Al colocar el raton encima del corazon o del carrito, este cambiará de color dependiendo de algunos factores.

En cuanto a las reglas de usabilidad, las iré mencionando una a una.

Visibilidad del estado del sistema

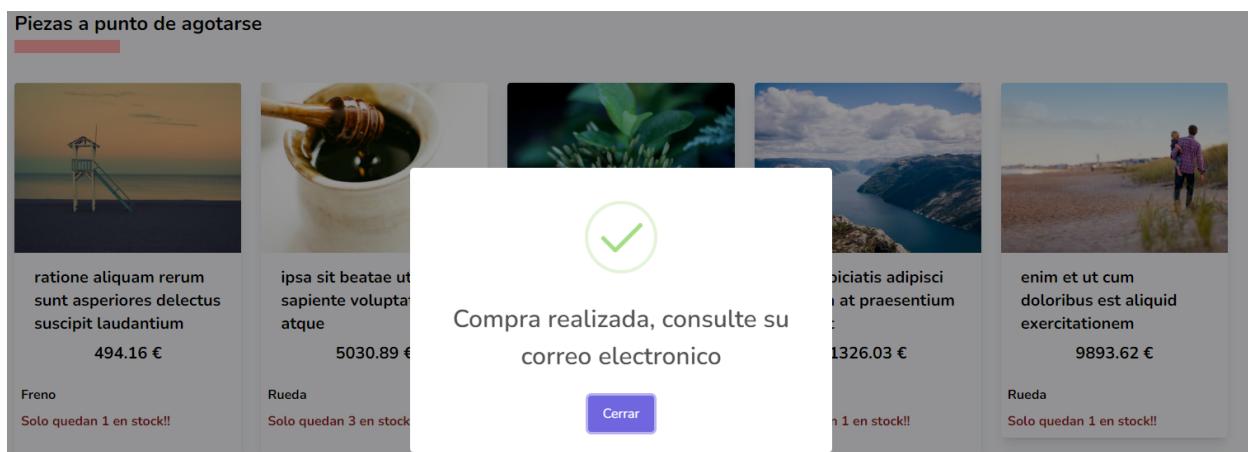
“Mantener informado al usuario de lo que sucede en la web”

Mostraré dos ejemplos. El primero es lo que aparece cuando algún usuario envía un formulario de contacto, al volver a la página inicial aparecerá lo siguiente



En caso de que no se haya podido enviar, será notificado igualmente.

El otro ejemplo sucede cuando se realiza una compra correctamente, mostrando el siguiente mensaje



Relación con el mundo real

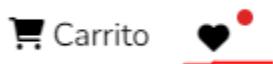
Este punto se cumple porque la web trata de piezas de algo que vemos todos los días como puede ser un vehículo. El problema es que como los datos que hay ahora son de prueba, no se llega a reflejar correctamente.

Control y libertad del usuario

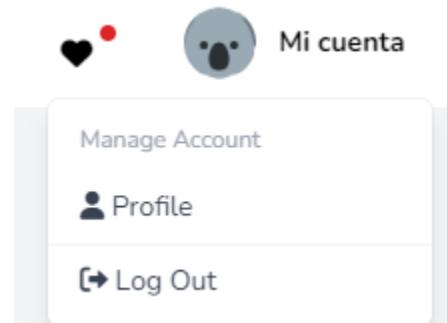
Este punto se cumple ya que el usuario sabe perfectamente dónde está cada cosa. ¿Qué quiere acceder al catálogo de productos? En la barra de navegación está la opción



Por otro lado, si quiere ver su lista de deseos, dicho menú se encuentra también en la barra de navegación



Si quiere configurar su perfil, también puede acceder a dicho menú desde la barra de navegación



Por último, en la ventana del catálogo de productos, el usuario puede filtrar por diversos campos

The screenshot shows a user interface for a product catalog. On the left, there are three vertical filter panels:

- Precio**: Radio buttons for "Menor a mayor" and "Mayor a menor".
- Pieza**: Radio buttons for various car parts: Embrague, Transmision, Valvula, Freno, Rueda, Cambio, Pedal, Espejo, Motor, Turbo, Supercargador, Radiador, and Amortiguador.
- Marcas**: Radio buttons for car brands: Audi, Citroen, Dacia, Dodge, Fiat, and Ford.

At the top center is a search bar with placeholder text "Buscar un producto..." and a magnifying glass icon. Below the search bar are four product cards, each with a thumbnail image, a title, a price, a status message, and a list of compatible brands:

Imagen	Título	Precio	Estado	Brands
	neque dolor iste asperiores qui ea rerum commodi	9988.88 €	Supercargador Solo quedan 3 en stock!!	Audi, Citroen
	enim et ut cum doloribus est aliquid exercitationem	9893.62 €	Rueda Solo quedan 1 en stock!!	Audi, Citroen, Dacia Dodge, Fiat, Ford Honda, Mercedes-AMG Mercedes-Benz, Opel
	qui vel quia aliquid et facilis qui assumenda	9866.46 €	Supercargador Solo quedan 1 en stock!!	Audi, Citroen, Dacia Dodge
	saepe qui incident et laudantium voluptates beatae corrupti	9858.85 €	Transmision	Audi, Citroen, Dacia Dodge, Fiat

Se puede filtrar por precio, por tipo de pieza y por marcas, además de poder buscar por un nombre concreto.

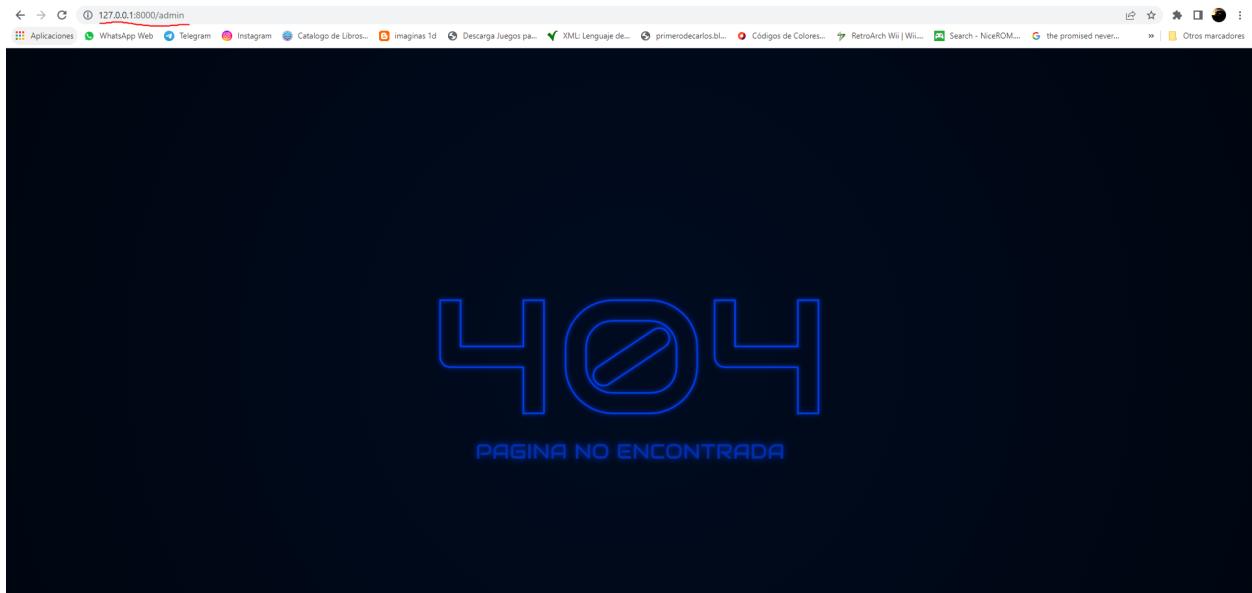
Prevención de errores

En este punto he hecho especial hincapié. El usuario normal no puede acceder por medios normales al panel de administrador. ¿Pero qué pasa si por casualidad prueba rutas hasta encontrar este panel? En este caso sucede lo siguiente

The screenshot shows a browser window with the URL `127.0.0.1:8000/admin`. The page title is "Listado de productos" and the sub-header is "Panel de Administracion". There is a navigation menu with links: "Aplicaciones", "WhatsApp Web", "Telegram", "Instagram", "Catalogo de Libros...", "imagenes 1d", "Descarga Juegos pa...", "XML: Lenguaje de...", "primerodecarlos.bl...", and "Códigos de". Below the menu is a horizontal navigation bar with tabs: "Productos", "Marcas", "Usuarios", and "Facturas".

Bienvenido al panel de administracion de
CarMotor.es

Este usuario si puede acceder al panel de administrador, la ruta existe, por lo que un usuario normal no debe de acceder a este sitio. Veamos lo que ocurre cuando un usuario normal intenta acceder



Esto ocurre por el siguiente motivo.

En el archivo de rutas se ha creado una condición especial basándose en un campo de la tabla de usuarios. Básicamente esta condición comprueba si el usuario que ha iniciado sesión tiene privilegios para acceder a la página, si es así entonces cargará la página, en cualquier otro caso (haya usuario logueado o no) se mostrará el error de página no encontrada. Lo lógico sería mostrar un error 500 pero cuanto menos sepa el usuario normal de lo que sucede a nivel de servidor mejor.

Reconocimiento antes que recuerdo

En la web disponemos de un registro de usuarios para facilitar la navegación por la web.



Aquí por ejemplo el usuario sabe que hace cada botón y además al colocar el ratón encima de cada ícono, se le mostrará un texto indicando qué significa.

Una de las ventajas de estar registrado es que al momento de llenar formularios, toda la información del usuario será rellenada automáticamente, agilizando mucho cualquier proceso. Esta es una baza importante para que más usuarios se acaben registrando en la web ya que estamos jugando con el tiempo del usuario.

Flexibilidad y eficiencia de uso

El usuario dispone a la hora de buscar un producto de una variada selección de filtros junto con una barra de búsqueda

The screenshot shows a user interface for searching products. At the top left is a sidebar with filtering options:

- Precio:
 - Menor a mayor
 - Mayor a menor
- Pieza:
 - Embrague
 - Transmision
 - Valvula
 - Freno
 - Rueda
 - Cambio
 - Pedal
 - Espejo
 - Motor
 - Turbo
 - Supercargador
 - Radiador
 - Amortiguador
- Marcas:
 - Audi
 - Citroen
 - Dacia
 - Dodge
 - Fiat
 - Ford

At the top center is a search bar with placeholder text "Buscar un producto..." and a magnifying glass icon.

The main area displays four product cards:

- Supercargador**:
Image: A close-up of a white, frosty, crystalline structure.
Description: neque dolor iste asperiores qui ea rerum commodi
Price: 9988.88 €
Stock: Solo quedan 3 en stock!!
Brands: Audi, Citroen, Dacia, Dodge, Ford, Honda, Mercedes-AMG, Mercedes-Benz, Opel
- Rueda**:
Image: A person walking on a sandy beach at sunset.
Description: enim et ut cum doloribus est aliquid exercitationem
Price: 9893.62 €
Stock: Solo quedan 1 en stock!!
Brands: Audi, Citroen, Dacia, Dodge, Fiat, Ford, Honda, Mercedes-AMG, Mercedes-Benz, Opel
- Supercargador**:
Image: A train moving through a dark, misty landscape.
Description: qui vel quia aliquid et facilis qui assumenda
Price: 9866.46 €
Stock: Solo quedan 1 en stock!!
Brands: Audi, Citroen, Dacia, Dodge, Fiat, Ford, Honda, Mercedes-AMG, Mercedes-Benz, Opel
- Transmision**:
Image: A green coffee mug on a desk.
Description: saepe qui incident et laudantium voluptates beatae corrupti
Price: 9858.85 €
Brands: Audi, Citroen, Dacia, Dodge, Fiat

Below the cards are three small, partially visible images: a building, a white surface, and a forest.

Por otro lado, los administradores también disponen de una barra de búsqueda en la sección de administración.

Estética y diseño minimalista

A lo largo de las diferentes capturas se ha podido ver el diseño de la web. En la web principal predomina el blanco y los únicos elementos que hay son los productos divididos en 3 secciones.

Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de errores

La web está lo suficientemente testada como para evitar algún error. En cuanto a los mensajes de error mostraré lo que ocurre cuando se ponen datos incorrectos en la zona para actualizar la dirección del usuario

The screenshot shows a 'Update Direction' form with the following fields and errors:

- Calle:** The input field contains "Calle, N°". Below it, a red error message says "The calle field is required."
- CP:** The input field contains "Codigo Postal". Below it, a red error message says "The cp field is required."
- Poblacion:** The input field contains "Poblacion". Below it, a red error message says "The poblacion field is required."
- Provincia:** The input field contains "Provincia". Below it, a red error message says "The provincia field is required."

At the bottom right of the form is a dark blue 'SAVE' button.

Ayuda y documentación

Como mencionamos anteriormente, la web dispone de un formulario de contacto donde los usuarios podrán aclarar sus dudas o solucionar algún problema que tengan con la web.

Despliegue en AWS

Durante el despliegue realicé una pequeña guía con todos los pasos a realizar. Dicha guía la subí a google drive y la pueden ver aquí: [Tutorial](#)

No puedo poner aquí la dirección de la web debido a que cada vez que se arranca la máquina su dirección cambia. Es por esto que he configurado un ubuntu server con el proyecto para que se pueda visualizar.

La OVA la podeis encontrar aqui: [OVA](#)

Las credenciales son las siguientes:

- Usuario: carlos
- Password: 2302

Con estas credenciales ya podéis acceder a la máquina. Acto seguido se visualizará la IP de la misma, únicamente lo que se deberá de hacer es introducir dicha ip en el navegador y ya se mostrará la página web.

Para el profesorado se le ha creado un correo con derechos de administrador para que puedan investigar mejor la web. Dichas credenciales son las siguientes:

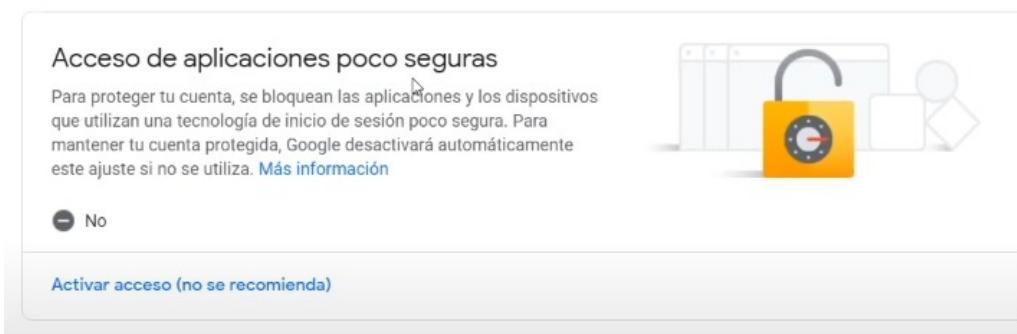
- Correo: profesorado@gmail.com
- password: password

Conclusiones finales

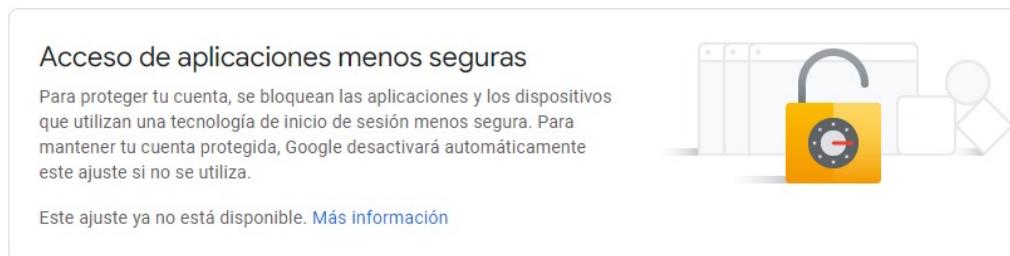
Problemas encontrados y soluciones

- Este problema está relacionado con el envío de correos. En un principio tenía planeado probar el servicio de envío de correo usando Mailtrap para después, una vez terminado el proyecto, cambiarlo por el servicio de correo de Google. El problema ha venido cuando he ido a informarme sobre como implementarlo y he descubierto que este servicio dejó de estar disponible el 30 de mayo de 2022.

Para implementar este servicio había que irse a los ajustes de la cuenta de google y en el apartado de seguridad buscar y activar lo siguiente:



Cuando me he dispuesto a buscarlo en mi cuenta me he encontrado con esto:



Dandole a ‘Más información’ nos encontramos esto:

Aplicaciones menos seguras y la cuenta de Google

Para proteger tu cuenta, a partir del 30 de mayo del 2022, Google dejará de admitir aplicaciones y dispositivos de terceros que te pidan que inicies sesión en tu cuenta de Google usando solo tu nombre de usuario y contraseña.

Importante: Este plazo no se aplica a los clientes de Google Workspace o de Google Cloud Identity. La fecha en que el cambio se implementará para estos clientes se anunciará en el blog de Workspace más adelante.

Para obtener más información, sigue leyendo.

Si una aplicación o un sitio no cumple con nuestros [estándares de seguridad](#), podemos bloquear el acceso a cualquiera que intenten iniciar sesión en tu cuenta desde esa aplicación o ese sitio. Las aplicaciones menos seguras pueden facilitar el acceso de los hackers a tu cuenta, por lo que bloquear los inicios de sesión procedentes de estas aplicaciones contribuye a proteger la cuenta.

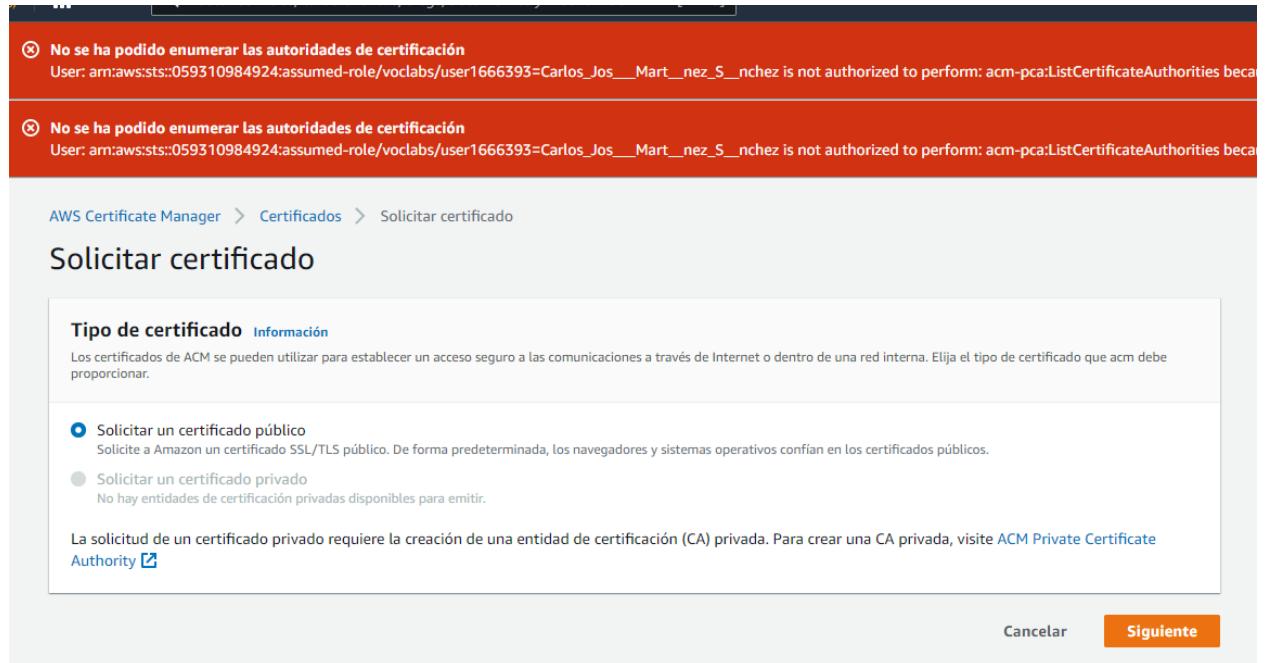
Según he investigado, la única opción que queda es mediante el servicio de Google Workspace, el cual tiene un precio de 5€ al mes. Debido a esto seguiré usando el servicio de Mailtrap.

2. Otro problema que he tenido tiene que ver con el certificado SSL de la web. Resulta que para implementar dicho certificado (que es gratuito) es necesario tener previamente un nombre de dominio, por ejemplo www.carmotor.es. Bien pues resulta que la cuenta que dispongo de AWS no me permite crear un dominio que además es de pago. Lo iré mostrando a continuación.

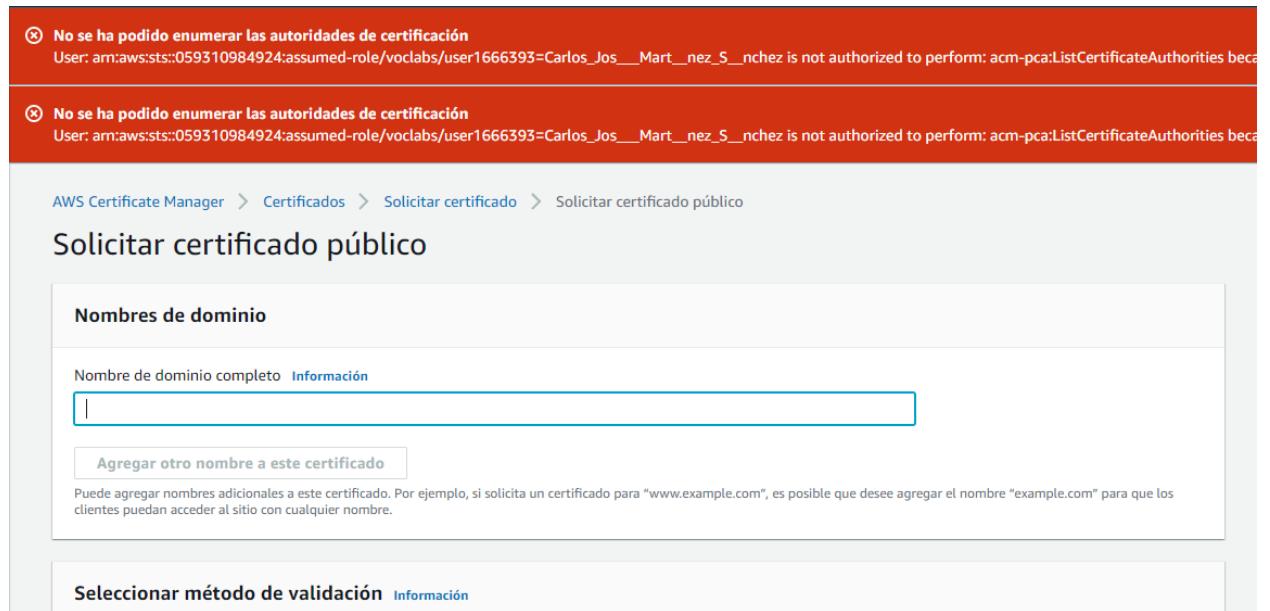
The screenshot shows the AWS Certificate Manager (ACM) interface. At the top, there is a red error message: "No se ha podido enumerar las autoridades de certificación" and "User: arn:aws:sts::059310984924:assumed-role/voclabs/user1666393=Carlos_Jos__Mart_nez_S_nchez is not authorized to perform: acm-pca>ListCertificateAuthorities because no identity-based policy allows the acm-pca>ListCertificateAuthorities action". Below this, the main heading is "AWS Certificate Manager" with the subtext "AWS Certificate Manager (ACM) facilita el aprovisionamiento, la administración, la implementación y la renovación de certificados SSL/TLS.". To the right, there is a sidebar titled "Nuevo certificado administrado por ACM" with options: "Solicite un certificado público de Amazon o un certificado privado de la entidad de certificación (CA) de su organización.", "Importar certificados obtenidos fuera de AWS", "Importar un certificado", and "Cree jerarquías de entidades de certificación (CA) privadas para su organización.".

En esta sección es donde podemos solicitar un certificado, ya de primeras me van saliendo errores.

Ahora seleccionamos el público



Y a continuación nos pide introducir el nombre del dominio



Como no tenemos ninguno, nos disponemos a intentar crearlo

The screenshot shows the AWS Route 53 service interface. On the left, there's a sidebar with various options like 'Panel', 'Zonas hospedadas', 'Comprobaciones de estado', etc. Under 'Dominios', it says 'Dominios registrado:' and 'Solicitudes pendientes'. The main area has a search bar and tabs for 'Registrar un dominio', 'Transferir dominios', and 'Informe de facturación de dominios'. A prominent red error message box says: 'Lo sentimos, pero no dispone de los permisos suficientes para realizar esta acción.' Below the message, the heading 'Dominios registrados' is visible.

ya de primeras nos dice que no podemos, pero aun así si nos vamos a Registrar un dominio...

This screenshot shows the 'Elegir un nombre de dominio' (Choose a domain name) step in the AWS Domain Registration wizard. It features a search input field and a dropdown menu titled 'Popular' showing various domain extensions with their prices: '.com - 12,00 \$', '.net - 11,00 \$', '.org - 12,00 \$', '.com.au - 15,00 \$', '.co.uk - 9,00 \$', '.io - 71,00 \$', '.info - 23,00 \$', '.co - 25,00 \$', and '.nl - 9,00 \$'. There are 'Comprobar' (Check) and 'Continuar' (Continue) buttons at the bottom right.

Ya nos salen los distintos precios que hay.

Por lo tanto no puedo crearle a la web un certificado SSL, así que la forma de acceder a ella seguirá siendo mediante HTTP.

3. Otro problema que he tenido ha sido también con AWS. Resulta que cada vez que arranco la máquina virtual, su IP pública cambia. Esto provocaba que si me metía a la web tras arrancarla, las imágenes no cargaban debido a que la dirección de la web era diferente.

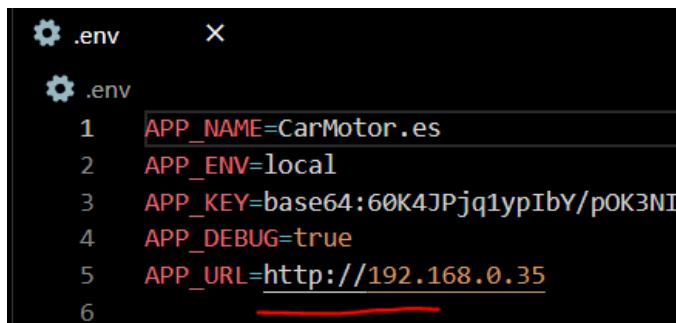
Lo que tenía que hacer nada más arrancar la máquina era conectarme a ella por ssh e irme al archivo .env del proyecto y cambiar la URL por la actual.

Estuve investigando sobre como solucionarlo, intenté probar con la DNS pública pero esta también cambia cada vez que se arranca la máquina, lo que no cambia de la máquina es la IP privada y la DNS privada, pero para que carguen las imágenes necesito la publica.

Para solucionarlo hice lo siguiente. Resulta que si la URL del archivo .env está vacío, por defecto cargará otra página

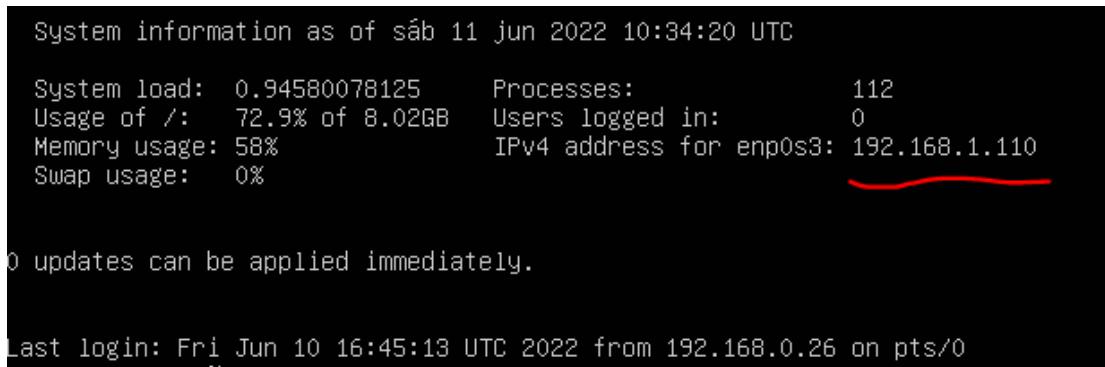
Iré mostrandolo con imágenes.

Este es el archivo de configuración (.env)



```
⚙️ .env      X
⚙️ .env
1 APP_NAME=CarMotor.es
2 APP_ENV=local
3 APP_KEY=base64:60K4JPjq1ypIbY/pOK3NI
4 APP_DEBUG=true
5 APP_URL=http://192.168.0.35
6 _____
```

Pero la IP de la máquina es la siguiente



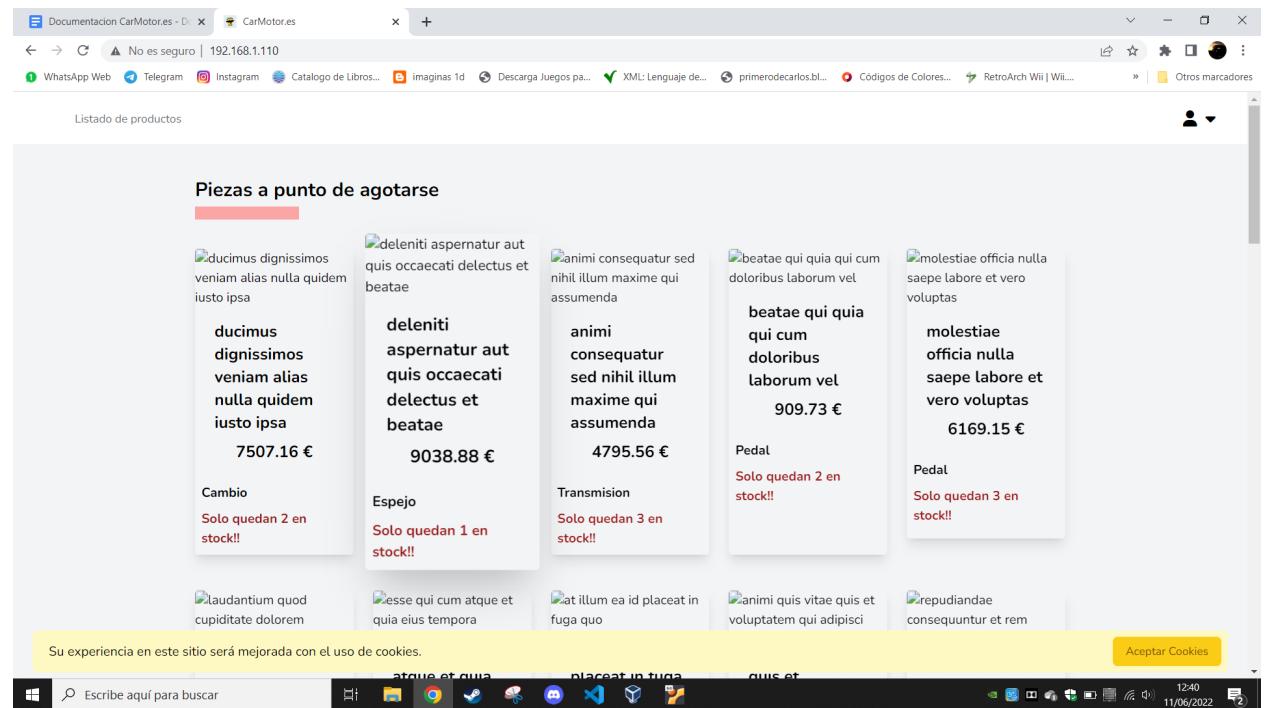
```
System information as of sáb 11 jun 2022 10:34:20 UTC

System load: 0.94580078125    Processes:           112
Usage of /: 72.9% of 8.02GB   Users logged in:      0
Memory usage: 58%              IPv4 address for enp0s3: 192.168.1.110
Swap usage:  0%                _____
```

0 updates can be applied immediately.

```
Last login: Fri Jun 10 16:45:13 UTC 2022 from 192.168.0.26 on pts/0
_____
```

Esto provoca que la web se vea así

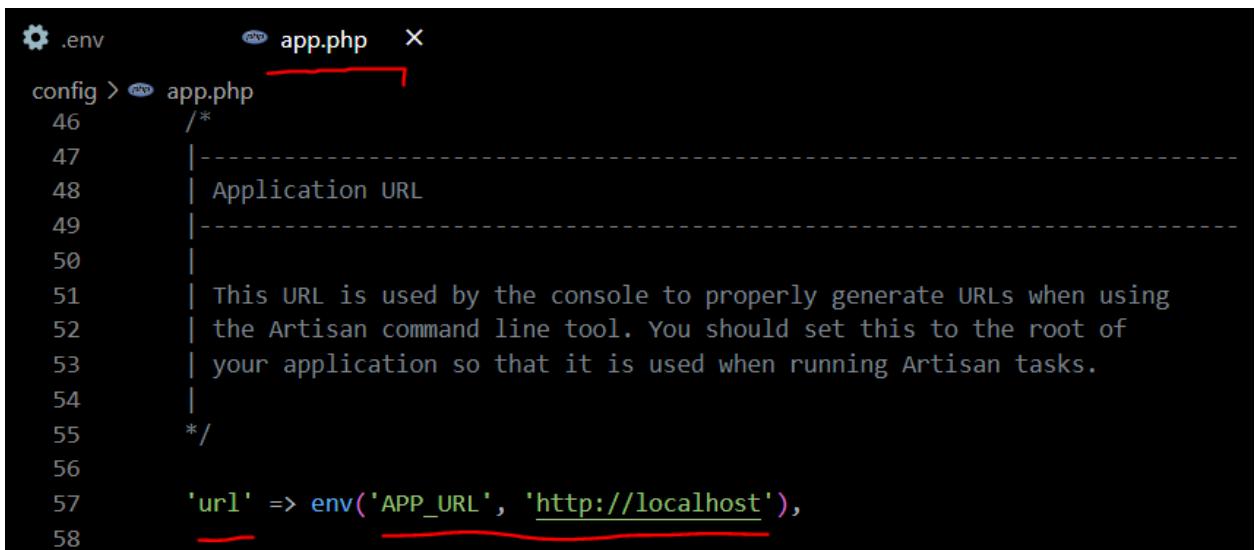


Para solucionar esto hice lo siguiente.

Primero dejé vacía la URL del .env

```
⚙️ .env X  
⚙️ .env  
1 APP_NAME=CarMotor.es  
2 APP_ENV=local  
3 APP_KEY=base64:60K4JPjq1ypIb...  
4 APP_DEBUG=true  
5 APP_URL=
```

Ahora nos vamos al archivo app.php de la carpeta config

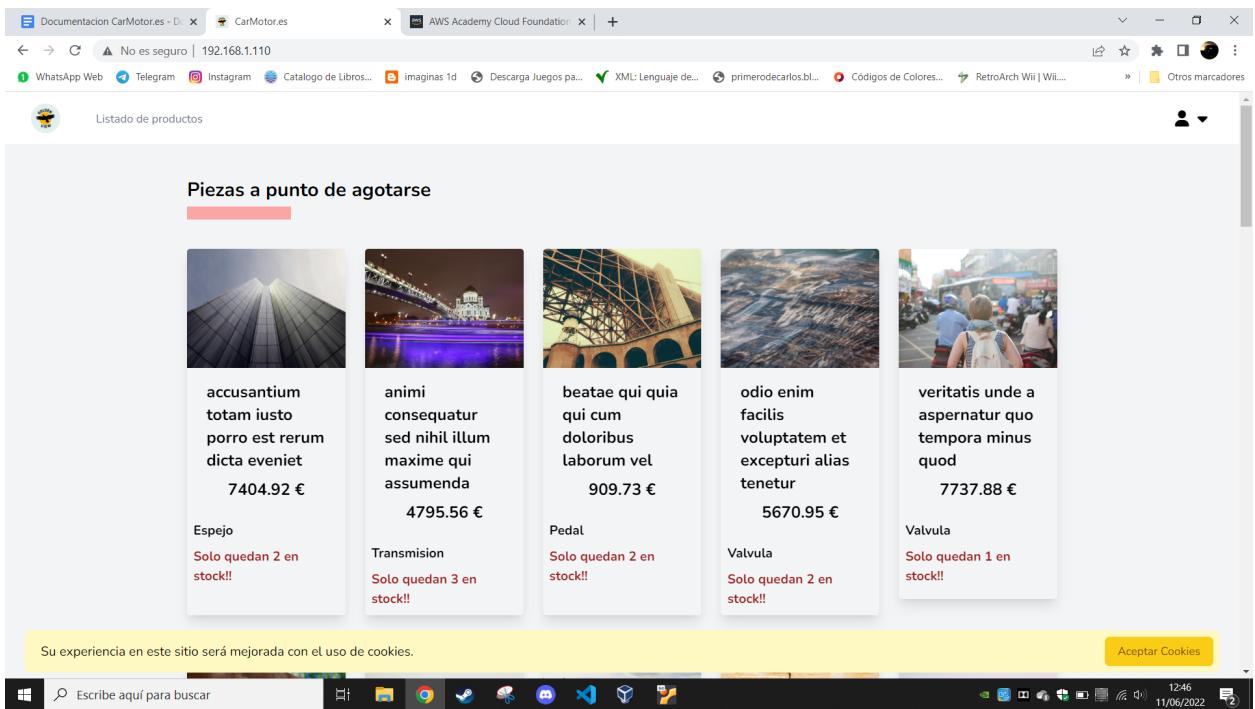


```

.env app.php X
config > app.php
46     /*
47     |
48     | Application URL
49     |
50     |
51     | This URL is used by the console to properly generate URLs when using
52     | the Artisan command line tool. You should set this to the root of
53     | your application so that it is used when running Artisan tasks.
54     |
55     */
56
57     'url' => env('APP_URL', 'http://localhost'),
58

```

Lo dejamos así y ahora volvemos a la web y ya carga perfectamente.



Considero que no es una buena práctica dejar la URL del .env vacía, pero no se me ocurre otra solución para esto que no pase por conectarme a la máquina y cambiar la dirección manualmente.

Anexo

- Web de stack overflow: [Enlace](#)
- Documentación oficial de Laravel: [Enlace](#)
- Documentación de Tailwind: [Enlace](#)
- Diversos componentes de Tailwind: [Enlace](#)
- Flowbite, una web con más componentes para Tailwind: [Enlace](#)
- SweetAlert: [Enlace](#)
- Documentación de AWS: [Enlace](#)
- Videoguía larga de la web: [Enlace](#)