

SENA

3197815

Instructora

Javier Leonardo Pineda Uribe

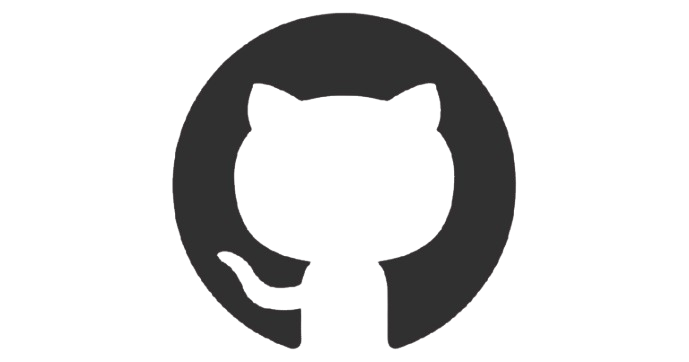
Aprendiz

Juan Pablo Hernández Cadenas

Competencia

Aplicación de la calidad de software

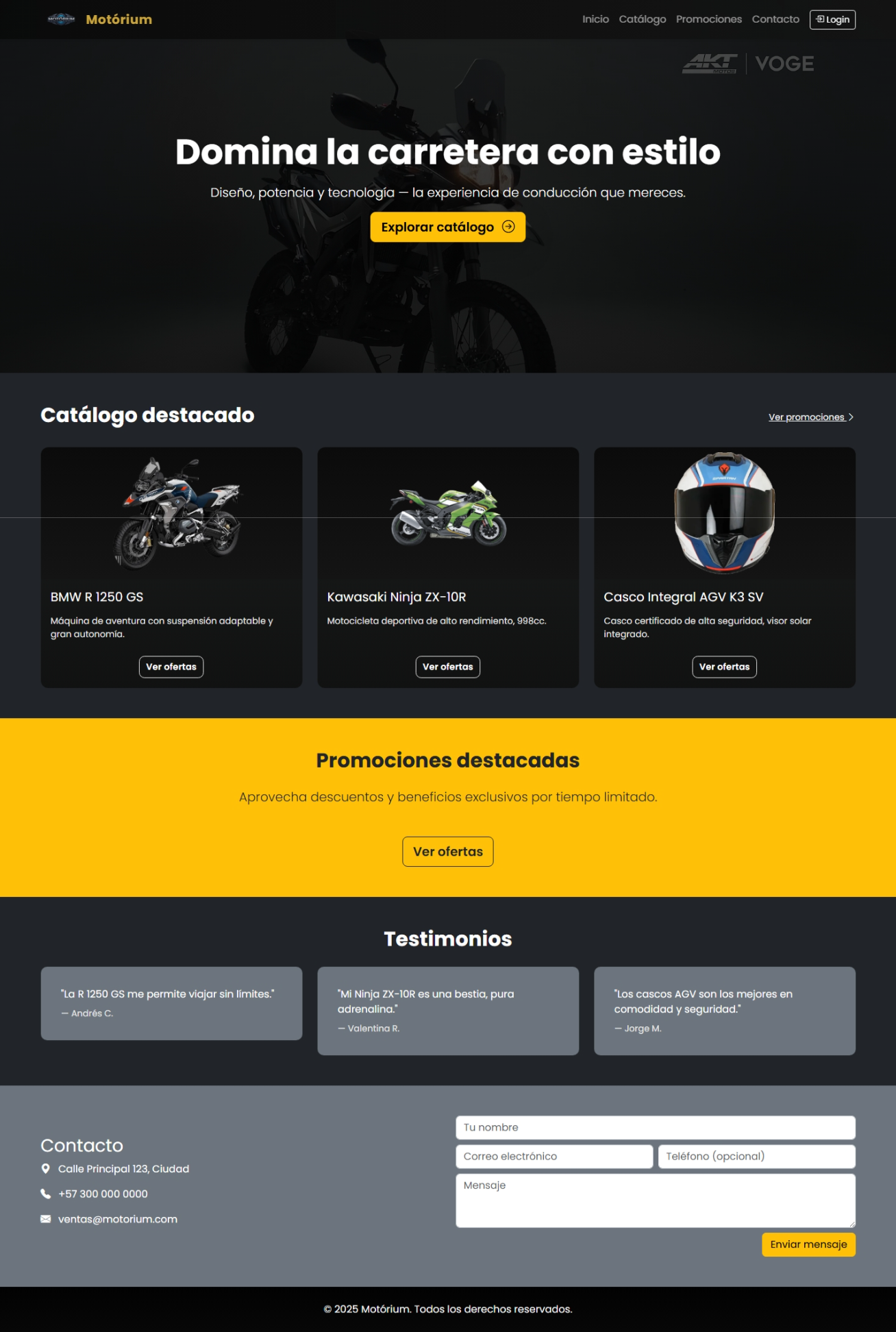
19/10/2025

Repositorio Git Hub   
[](https://acortar.link/q2L6Tq)

## 1. INTRODUCCIÓN Y OBJETIVO

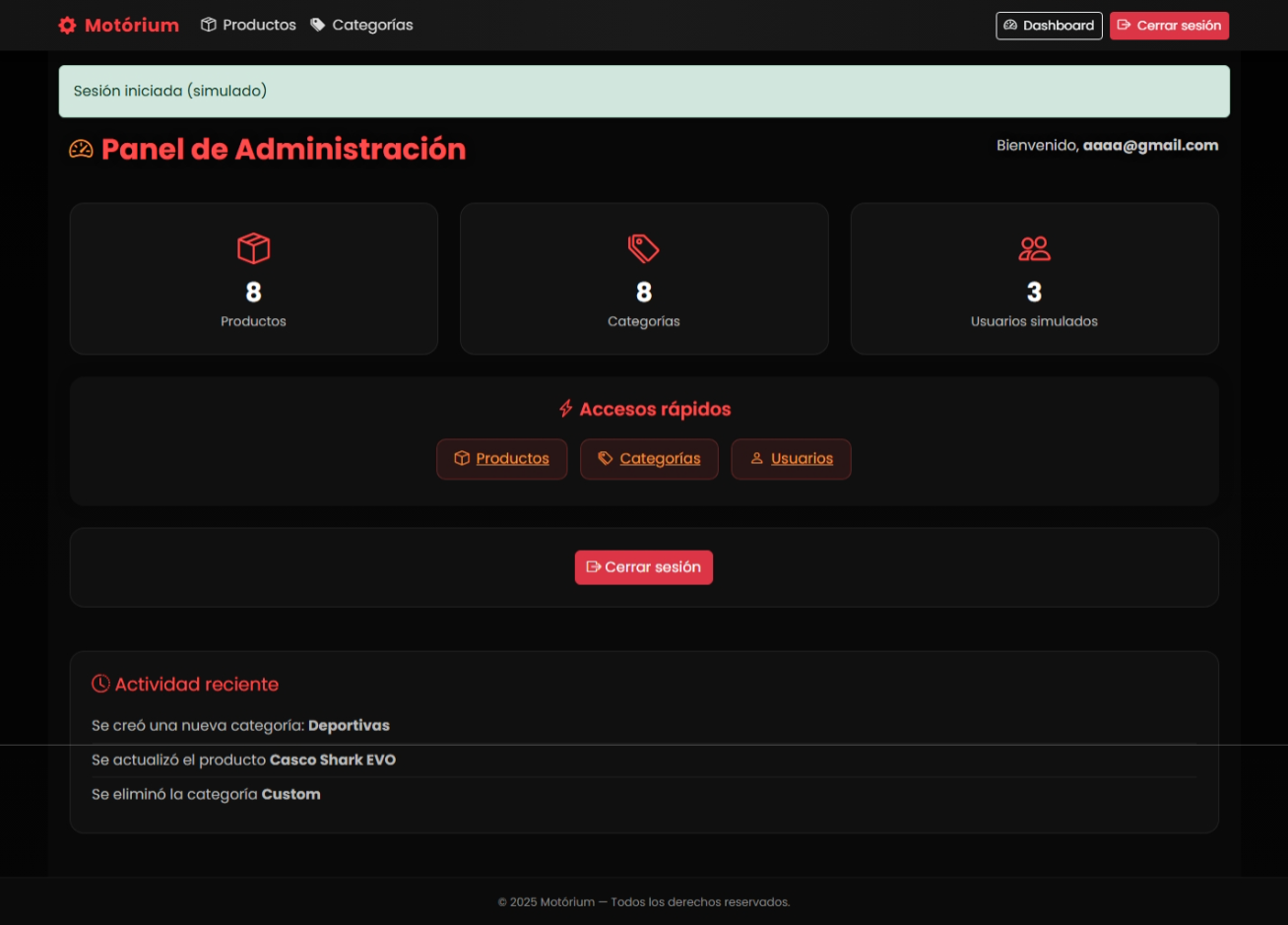
El presente documento detalla el análisis del proyecto Motorium el cual está diseñado para venta de motos con crud incorporada de MySQL y además utilizando el framework de laravel.

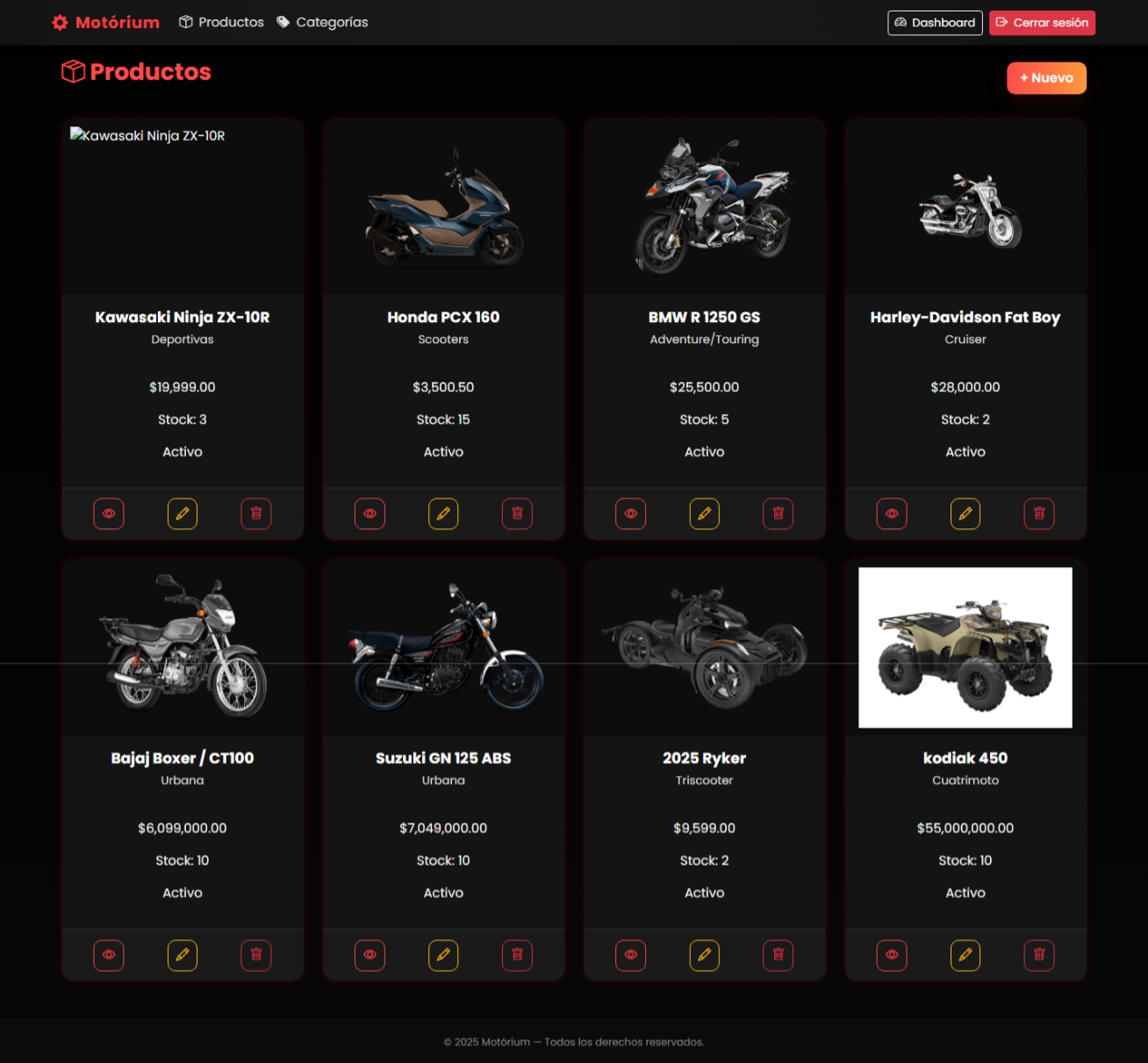
# Confirmación de funcionalidad

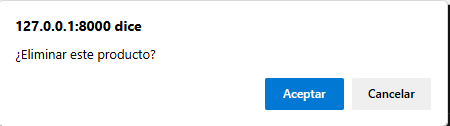
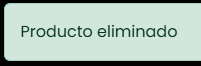
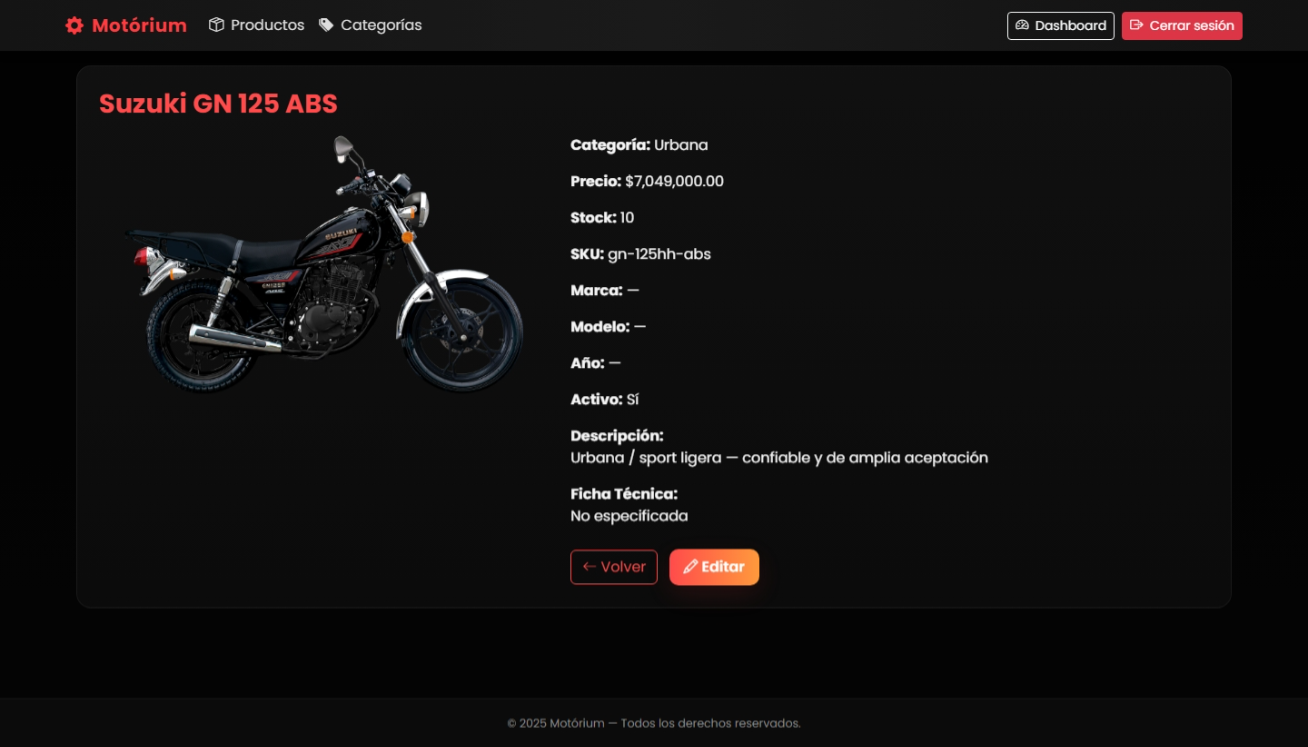
Imágenes de la página Motorium en la web desde servidor local.  
Index



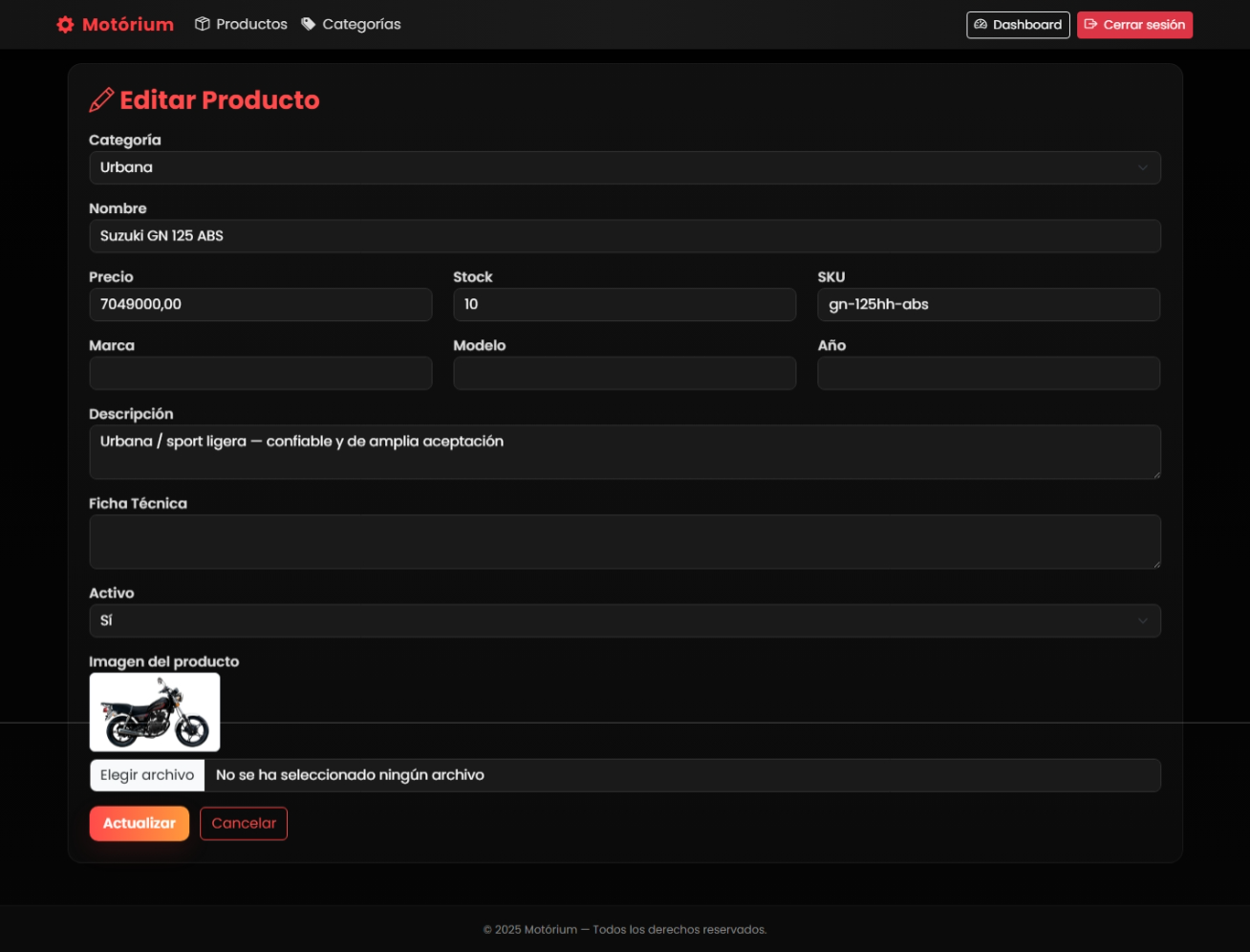
Login

Dashboard

CRUD-PRODUCTOS   
index

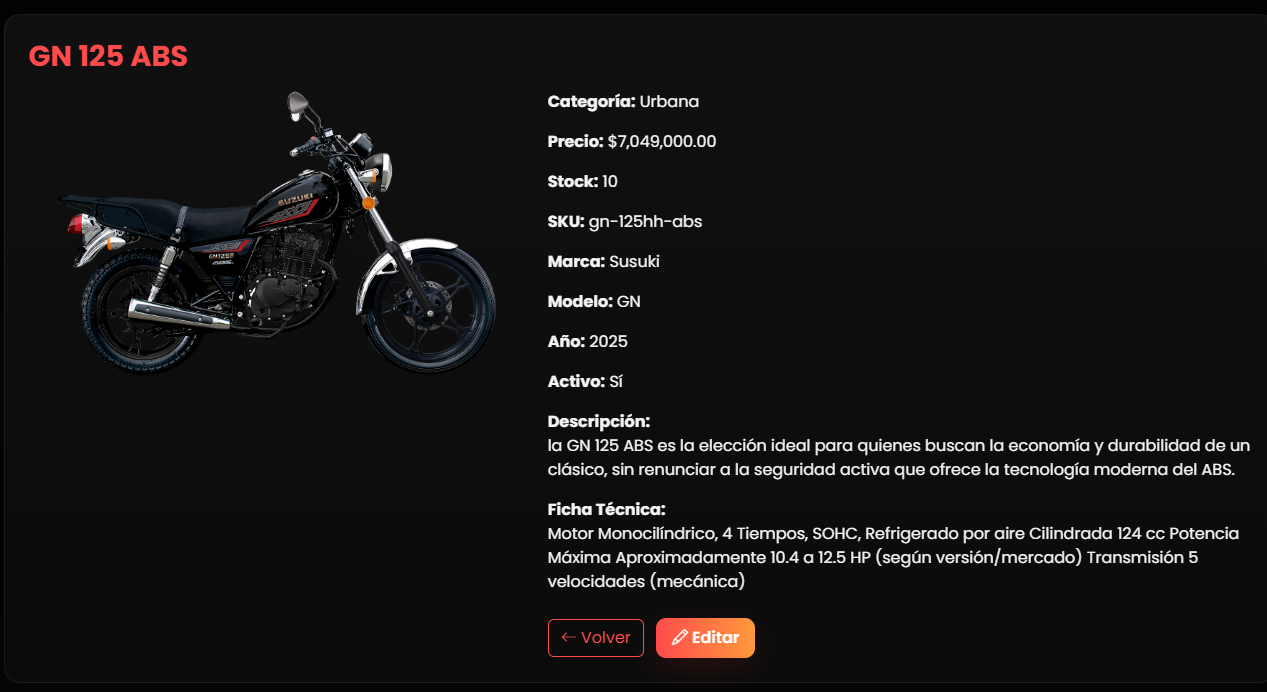
Vista Eliminar  


Editor



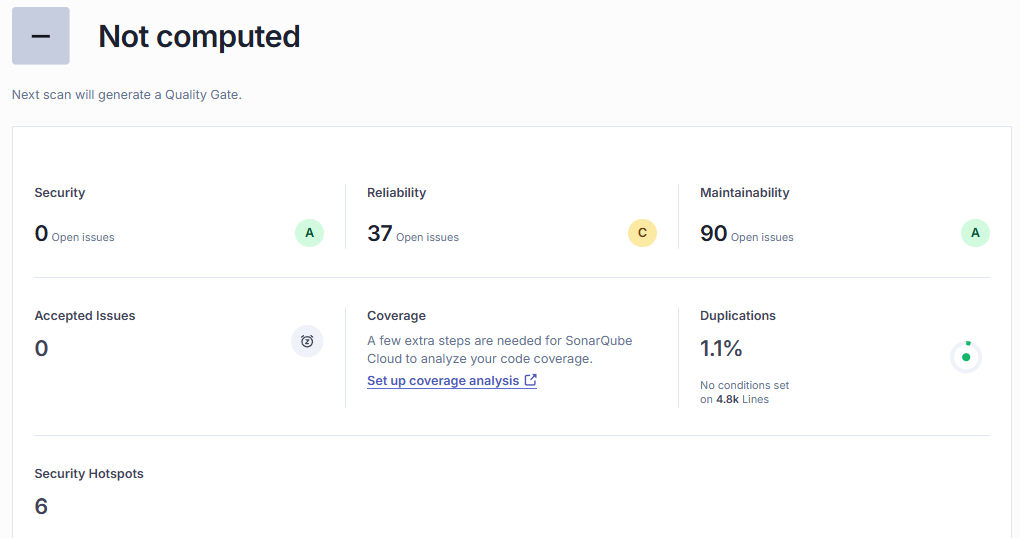
Se hará el cambio del nombre a GN 125 ABS





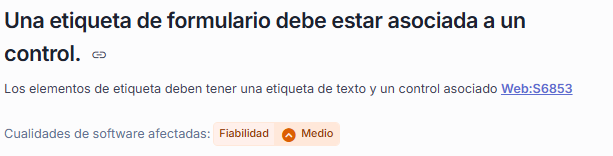
# Escaneo con Sonar Cloud

**Primer escaneo de Motorium en Sonar Cloud**

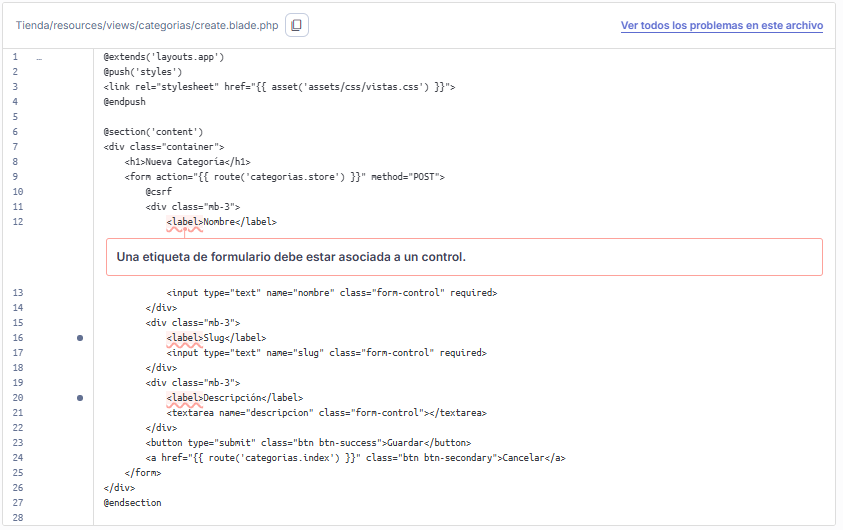
Como podemos ver en mi primer análisis de Motorium podemos sacar las siguientes conclusiones:

En seguridad: se encontró un numero de cero con calificación en A, lo cual significa que mi proyecto Motorium no tiene ningún error, bug, falencia o código que pongo en vulnerabilidad a mi proyecto.  
En fiabilidad tenemos 37 asuntos abiertos con calificación en C, lo cual **Sonar,** indica que mi código puede tener al menos un error importante, esto significa que en esos 37 asuntos abiertos pueden tener una falencia de error dentro del código.  
En mantenibilidad dio como resultado 90 asuntos abiertos en calificación A, esto según los documentos de métricas de sonar se basa en que le deuda técnica(Technical Debt) o en el radio de deuda técnica (Technical Debt Ratio) se estarán cambiando o actualizando periódicamente lo cual significa mas tiempo de deudo técnica y menos porcentaje de aceptación.  
Duplicidad: El proyecto presenta una **duplicidad de código extremadamente baja del 1.1%**. Este resultado es muy positivo y confirma el uso de del principio **DRY (Don't Repeat Yourself)** según Sonar cloud

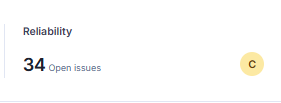
# Análisis

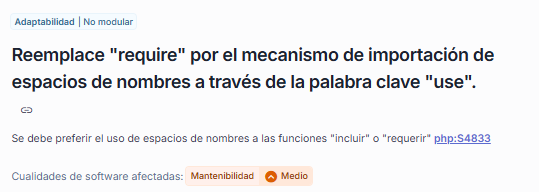


Comenzaremos con un error de Fiabilidad donde nos muestra un code smells que se encuentra en create.blade.php y edit.blade.php de la carpeta categoría lo que significa que hay un error en su diseño.



El error proviene de la etiqueta <label>, donde nos dice que este debe estar asociado a un controlador, esto se corrige con la función for dentro del <label>.

 Aca como nos sersioramos que los ajustes fueron tomados pasando de 37 a 34 errores de rentabilidad.



En el segundo nos vamos a uno de Mantenibilidad donde nos menciona que debemos utilizar la clave “use” de importacion en vez de “require”.



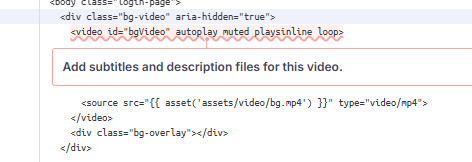
Según lo que investige estos errores que se muestran en la correcion es que hay una diferencia entre laravel y sonar el cual nos pide utiliza use en vez de require pero laravel necesita require para el llamado de scripts que utiliza para su funcionamiento.



Aca se muestra que se hizo un cambio pasando de 90 a 89 issues



Este error es de mantenibilidad el cual nos dice que le debemos hacer una descripcion al video que se reproduce en la pagina login.blade.php



Entonces para corregir este error agregue “label” con una breve descripciòn donde habla un poco aserca del video.

