



Manual del programador

Jeanluca Chaves Flores cf190725

Carlos Eduardo Siliézar Colocho SC202203

José Ángel Cartagena Rivera CR190362

Angel Guillermo Sanchez Mangandi SM192656

1. Objetivo

- Permitir que la llantería tenga presencia en Internet, lo que nos ayuda a que todos los clientes puedan encontrar información sobre los productos y servicios.
- Mostrar una serie completa de los productos que ofrece la llantería, incluyendo diferentes marcas, modelos y tamaños de productos.

2. Funcionalidad

Esta página web de llantería nos ofrece un catálogo donde le permite al cliente una variedad de búsquedas como marca, modelo, tamaño y más.

También cuenta con la función de carrito donde la permite agregar he eliminar producto dándole una facilidad y comodidad al cliente, cuenta con diversos métodos de pago ya sea tarjeta de débito o crédito y PayPal. Ofrece la opción de obtener informes detallados del inventario. El desarrollo de la aplicación se basa en el lenguaje de PHP, que es un lenguaje orientado a servidores y el cual es ampliamente utilizado para el desarrollo web y por su simplicidad es eficiente implementarlo dentro de la aplicación.

3. Instalación

Los requisitos para la instalación de la aplicación web son: primero la instalación de wamp, el cual es un paquete de software que facilita la instalación de un entorno de desarrollo web en Windows. Contiene Apache como servidor web, MySQL como base de datos y PHP como lenguaje de programación del lado del servidor, lo que permite crear y probar aplicaciones web localmente antes de publicarlas en un servidor en línea. Para el editor de código se utilizará como preferencia visual studio code, esto por su amplia gama de extensiones y lenguajes de programación que el editor puede soportar, y para finalizar con la instalación de componentes, se utilizara MySQL como servidor de base de datos relacional para alojar los datos que se utilizaran dentro de la aplicación.

Para la instalación de la aplicación primero se necesita iniciar la base de datos, para eso abrimos el script db_llanteria.sql

```
db_llanteria.sql -- GUILLERMO (57)
-- phpMyAdmin SQL Dump
-- version 5.2.0
-- https://www.phpmyadmin.net/
--
-- Servidor: 127.0.0.1:3306
-- Tiempo de generación: 16-08-2024 a las 16:27:21
-- Versión del servidor: 8.0.31
-- Versión de PHP: 8.0.26

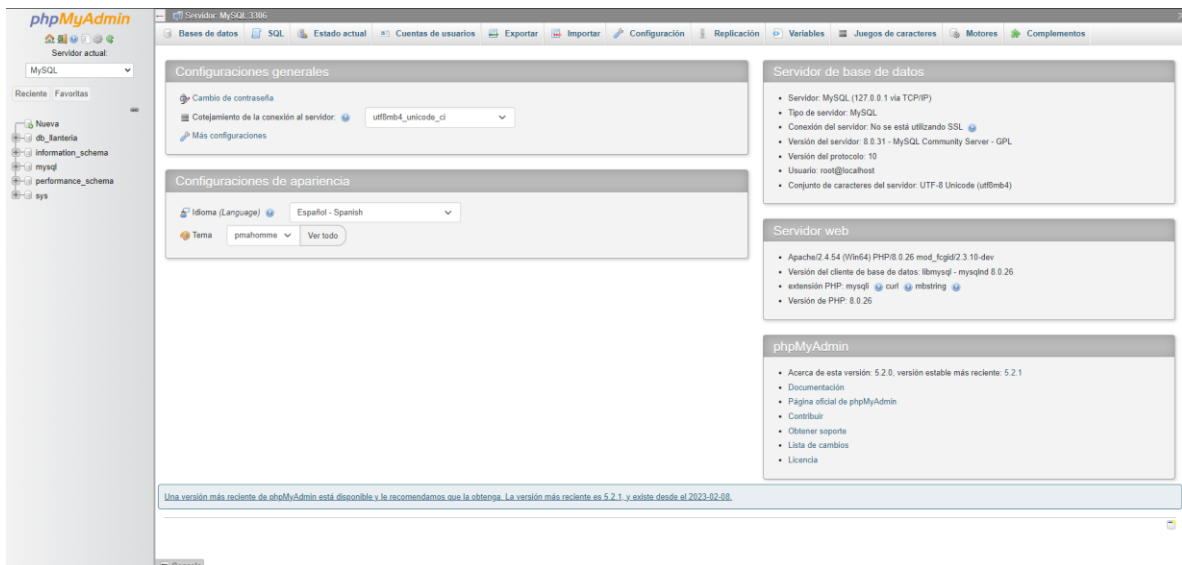
SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

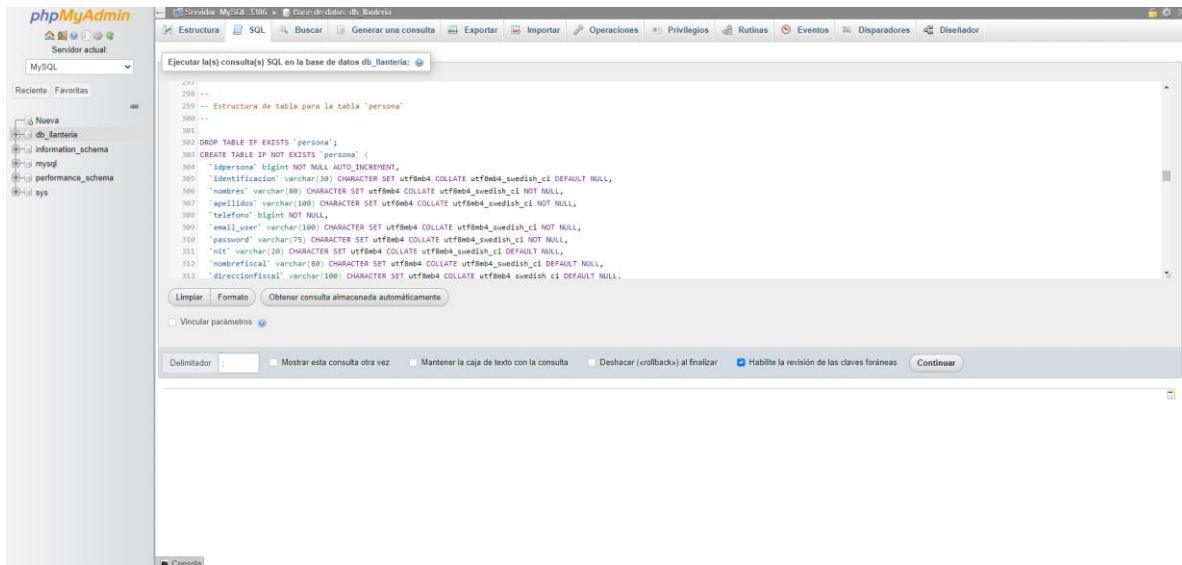
--
-- Base de datos: 'db_llanteria'
--
--
-- Estructura de tabla para la tabla 'categoria'
--
DROP TABLE IF EXISTS `categoria`;
CREATE TABLE IF NOT EXISTS `categoria` (
  `idcategoria` bigint(20) NOT NULL AUTO_INCREMENT,
  `nombre` varchar(255) CHARACTER SET utf8mb4 COLLATE utf8mb4_swedish_ci NOT NULL,
  `descripcion` text CHARACTER SET utf8mb4 COLLATE utf8mb4_swedish_ci NOT NULL,
  `portada` varchar(100) CHARACTER SET utf8mb4 COLLATE utf8mb4_swedish_ci NOT NULL,
  `datecreated` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
  `ruta` varchar(255) CHARACTER SET utf8mb4 COLLATE utf8mb4_swedish_ci NOT NULL,
  `status` int(11) NOT NULL DEFAULT '1',
  PRIMARY KEY (`idcategoria`)
) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_swedish_ci;

--
-- Volcado de datos para la tabla 'categoria'
--
INSERT INTO `categoria` (`idcategoria`, `nombre`, `descripcion`, `portada`, `datecreated`, `ruta`, `status`) VALUES
(1, 'Dunlop', 'Llantas Dunlop', 'img_55da462591e876d51666d9b03979b05.jpg', '2021-08-20 03:04:04', 'Dunlop', 1),
(2, 'Falken', 'Llantas Falken', 'img_72343178699c48544758a9a2aae1d599.jpg', '2021-08-21 00:47:18', 'Falken', 1),
(3, 'Sumitomo', 'Llantas Sumitomo', 'img_0f9818181f28ca382939274df21f5c59.jpg', '2021-08-21 00:47:18', 'Sumitomo', 1);
```

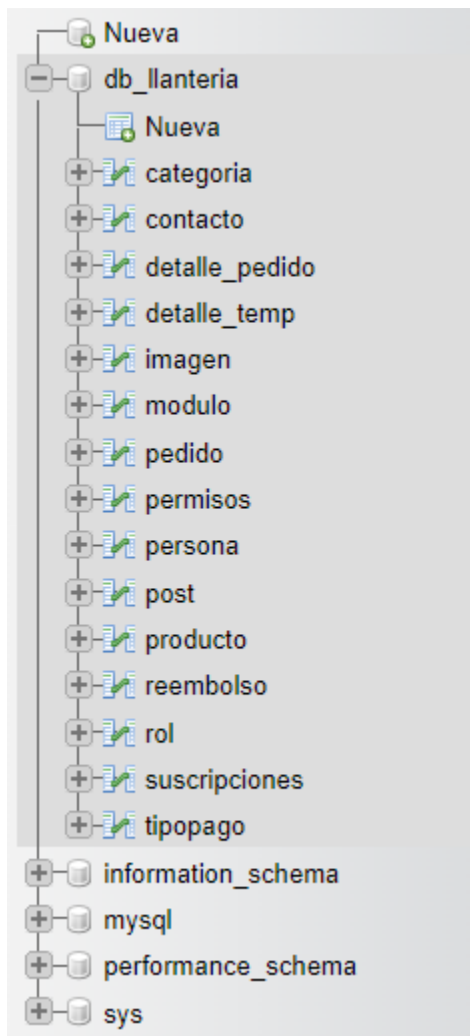
Luego copiamos el script MySQL y nos dirigimos a PHPmyAdmin, el cual ya viene incluido en la paquetería de instalación de WAMP , esto para crear una base de datos con el mismo nombre del archivo



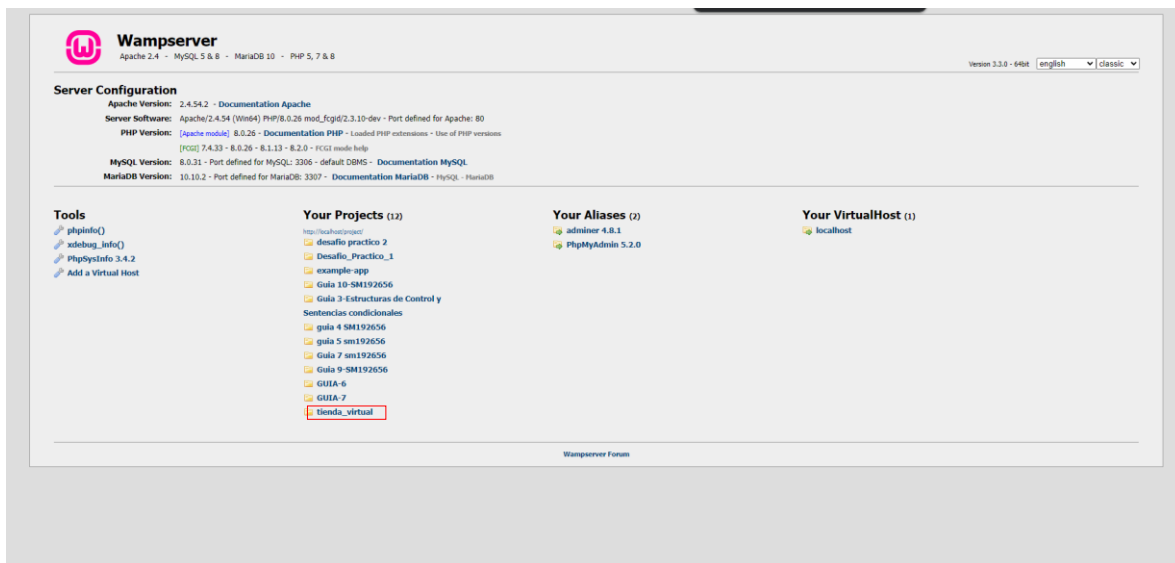
Y en la pestaña sql pegamos el script recién copiado y ejecutamos



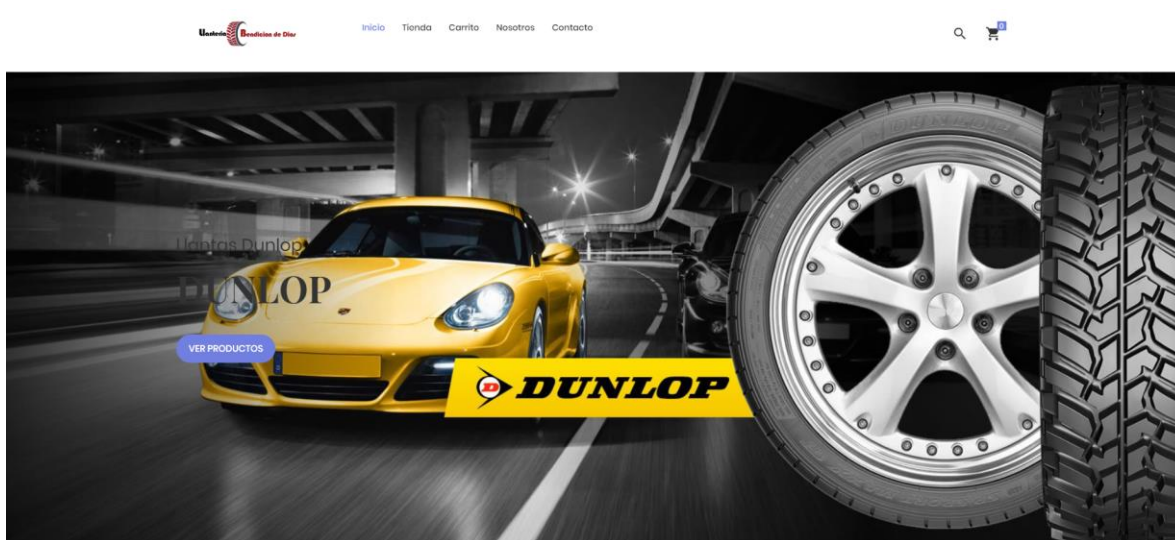
La base de datos quedaría algo así:



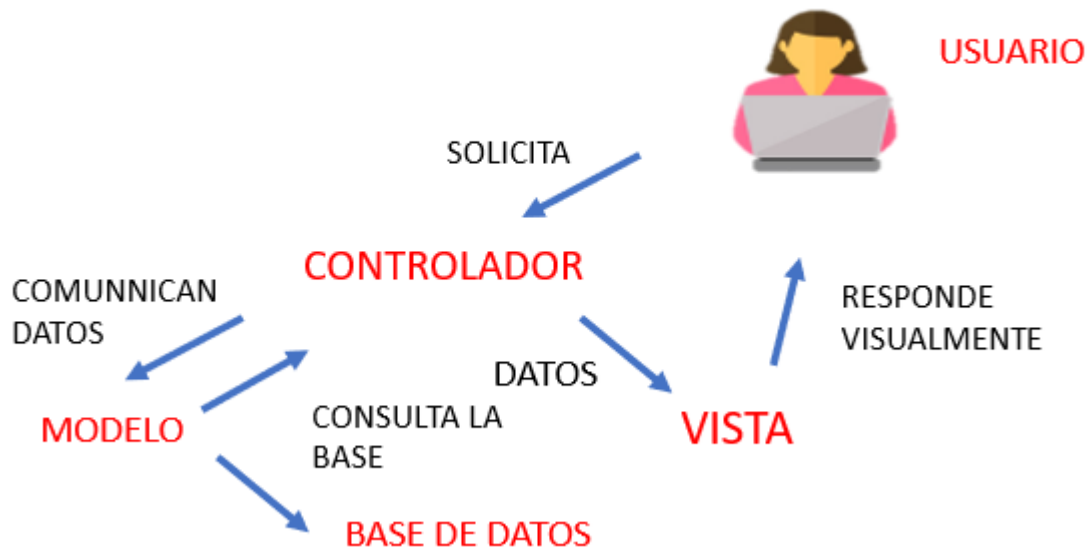
Ahora se puede colocar el proyecto en la aplicación de WAMP, y se abre desde localhost



Y al ejecutarlo ya se puede acceder a la interfaz de la aplicación.



Debido a que el programa fue hecho con el lenguaje PHP, esta aplicación pertenece a MVC (Modelo-Vista-Controlador) en el cual los lenguajes JavaScript y PHP funcionan como lenguaje en back-end funcionando en los controladores y modelos para mandar a llamar la base de datos, mientras que para las vistas su utiliza el lenguaje HTML dentro de PHP, el cual permite el orden y la escalabilidad en e proyecto.



3. Librerías y dependencias

Las librerías y dependencias utilizadas, tanto para funciones como diseño fueron:

- Bootstrap: es un framework de diseño css para el front-end que permite un desarrollo web más rápido y fácil, dentro del cual se implementaron dentro del proyecto el uso de botones, carruseles, modales, formularios y diseños de imágenes para una funcionalidad del código en distintas plataformas y con diseño amigable.
- DatePicker: es un selector de fecha, calendario emergente, selector de fecha y hora o selector de hora es un widget de interfaz gráfica de usuario que permite al usuario seleccionar una fecha de un calendario y/o una hora de un intervalo de tiempo.
- FontAwesome: Es un framework que permite modificar los estilos de fuente dentro del proyecto implementando código css que manda a llamar desde su base de datos las fuentes que necesitemos en el proyecto.
- JQuery: es una biblioteca multiplataforma de JavaScript, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web.
- Popper Js: es un motor de posicionamiento, su finalidad es calcular la posición de un elemento para permitir colocarlo cerca de un elemento de referencia dado.
- Sweetalert: Es un framework de tipo modal que permite mandar a llamar alertas con funciones distintas, estas son: "advertencia", "error", "éxito" e "info".
- JsBarcode: Es un framework generador de códigos de barras.
- TinyMCE: es un editor de texto WYSIWYG para HTML de código abierto que funciona completamente en JavaScript, y se ejecuta en el navegador de internet.
- map-custom: Es un framework que sirve para mandar a llamar un mapa dentro de la aplicación.

4. Actualización

Para actualizar los frameworks y la aplicación en si primero es necesario tener una copia de seguridad del proyecto, para luego pasar a verificar las actualizaciones de cada framework, pero para eso se debe buscar manualmente cada uno para tener en optimo estado la aplicación. Las versiones actuales de WAMP y PHPmyAdmin son 3.3.0 y 2.4.54.2 respectivamente. En caso de haberse actualizado, realizar las pruebas post-actualización y verificar QUE TODO FUNCIONE CORRECTAMENTE.