

Estanco 1997

Sobre nuestra tienda:

El Estanco 1977 lleva trabajando al servicio de los loreños desde 1977. Situado en Avenida Del Castillo nº 31, cuenta con una gran variedad de productos para el **fumador**.

Ahora también están especializados en artículos para CACHIMBAS Y EN VAPERS. ¡ Ven a conocernos a través de nuestra web !

Nos presentamos antes una página web dedicada a un estanco, en la que vemos una página de inicio en la que presentamos nuestros productos y las ofertas que tenemos en la página web, podemos encontrarnos un menú en la parte superior que contendrá las páginas de inicio, categoría y artículos. También en la parte superior derecha encontramos el carrito y el usuario que nos muestra si está activo, y si no lo está nos mandara a la página de login, en el apartado de las categorías tendremos un botón que nos permitirá navegar hasta los artículos de esa categoría, si pulsamos directamente en artículos nos mostrará todos los artículos que están disponibles para comprar, nuestra página de artículos contendrá a estos los cuales tendrán un botón de compra y para seleccionar la cantidad que quieres comprar, si lo añadimos al carrito podremos pasar al paso final para comprar y realizar el pedido para que nos llegue a nuestro domicilio.

Parte desarrollo web:

Nuestro proyecto está enfocado en un estanco en la cual tenemos un login y un register el cual nos permite tener nuestra propia cuenta para poder comprar, si no estamos logueados solo podremos ver los artículos que nos ofrece esa tienda pero no podremos comprarlos, el usuario contendrá un tipo que debe ser administrador o usuario normal, la diferencia es que el administrador tendrá las opciones de añadir, actualizar, borrar y ver todas las categorías y sus artículos, los usuarios normales solo tendrán la opción de comprar si están logueados y ver todos los artículos de nuestra tienda, al registrarse verificamos el usuario mediante nuestro correo electrónico. En cuanto a la base de datos utilizaremos MySQL mediante phpMyAdmin, mis tablas User, Category, Article, Order, Purchase. Además tendremos un buscador el cual nos filtra por el nombre de la cajetilla de tabaco y ordenaremos por el nombre o el precio que tenga la cajetilla de tabaco. Controlaremos los errores que nos puedan saltar a la hora de hacer las peticiones, las clases Category y Article contendrán un campo imagen el cual la utilizaremos en nuestra página web para ponerle imágenes a las categorías y sus artículos, el correo electrónico se enviará a la hora de verificar la cuenta.

//Crear la base de datos

```
CREATE DATABASE estanco1997AntonioCalzado;
USE `estanco1997AntonioCalzado`;
```

```
-- phpMyAdmin SQL Dump
-- version 5.2.0
-- https://www.phpmyadmin.net/
--
-- Host: mysqlDawes:3306
-- Generation Time: Jan 27, 2023 at 11:29 AM
-- Server version: 5.7.22
-- PHP Version: 8.0.19
```

```
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 */;
```

```
--
-- Database: `estanco1997AntonioCalzado`
--
```

```
-- -----
```

```
--
-- Table structure for table `category`
--
```

```
CREATE TABLE `category` (
  `id` int(11) NOT NULL,
  `name` varchar(50) DEFAULT NULL,
  `description` varchar(50) DEFAULT NULL,
  `image` varchar(500) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
-- -----
```

```
--
-- Table structure for table `material`
--
```

```
CREATE TABLE `tabaco` (  
  `id` int(11) NOT NULL,  
  `name` varchar(50) DEFAULT NULL,  
  `description` varchar(50) DEFAULT NULL,  
  `price` decimal(5,2) DEFAULT NULL,  
  `category` int(11) DEFAULT NULL,  
  `stock` int(50) DEFAULT '50',  
  `image` varchar(500) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

-- -----

```
--  
-- Table structure for table `pedido`  
--
```

```
CREATE TABLE `pedido` (  
  `id` int(11) NOT NULL,  
  `username` varchar(50) NOT NULL,  
  `date` datetime NOT NULL,  
  `iva` double NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

-- -----

```
--  
-- Table structure for table `purchase`  
--
```

```
CREATE TABLE `purchase` (  
  `id_order` int(50) NOT NULL,  
  `tabacol_code` int(11) NOT NULL,  
  `quantity` int(11) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

-- -----

```
--  
-- Table structure for table `user`  
--
```

```
CREATE TABLE `user` (  
  `username` varchar(50) NOT NULL,
```

```
`pass` varchar(200) DEFAULT NULL,  
`name` varchar(50) DEFAULT NULL,  
`email` varchar(50) DEFAULT NULL,  
`role` varchar(50) DEFAULT NULL,  
`verification_code` varchar(64) NOT NULL,  
`enabled` bit(1) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
--  
-- Indexes for table `category`  
--  
ALTER TABLE `category`  
  ADD PRIMARY KEY (`id`),  
  ADD UNIQUE KEY `name` (`name`);
```

```
--  
-- Indexes for table `material`  
--  
ALTER TABLE `tabaco`  
  ADD PRIMARY KEY (`id`),  
  ADD UNIQUE KEY `name` (`name`),  
  ADD KEY `fk_cat` (`category`);
```

```
--  
-- Indexes for table `pedido`  
--  
ALTER TABLE `pedido`  
  ADD PRIMARY KEY (`id`),  
  ADD KEY `order_fk_1` (`username`);
```

```
--  
-- Indexes for table `purchase`  
--  
ALTER TABLE `purchase`  
  ADD PRIMARY KEY (`id_order`, `tabacol_code`),  
  ADD KEY `fk_material` (`tabacol_code`),  
  ADD KEY `fk_Order` (`id_order`);
```

```
--  
-- Indexes for table `user`  
--  
ALTER TABLE `user`  
  ADD PRIMARY KEY (`username`);
```

```
--  
-- AUTO_INCREMENT for dumped tables  
--
```

```

--
-- AUTO_INCREMENT for table `category`
--
ALTER TABLE `category`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=13;

--
-- AUTO_INCREMENT for table `material`
--
ALTER TABLE `tabaco`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=104;

--
-- AUTO_INCREMENT for table `pedido`
--
ALTER TABLE `pedido`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=3;

--
-- Constraints for dumped tables
--

--
-- Constraints for table `material`
--
ALTER TABLE `tabaco`
  ADD CONSTRAINT `fk_cat` FOREIGN KEY (`category`) REFERENCES `category` (`id`)
  ON DELETE CASCADE;

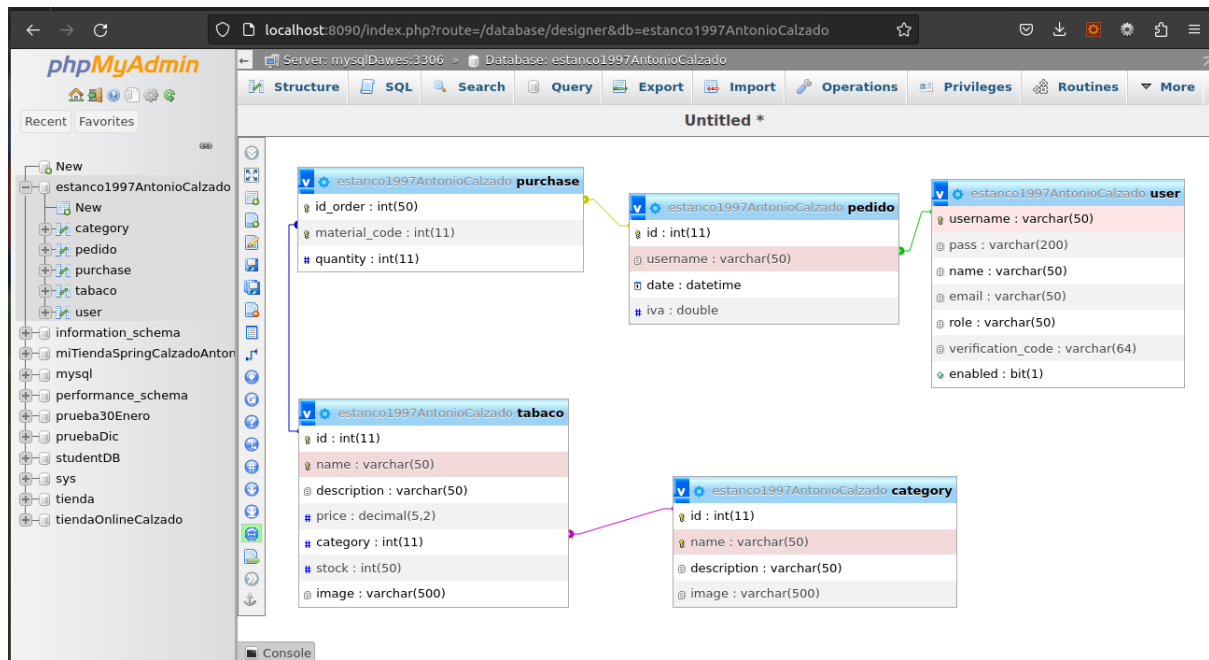
--
-- Constraints for table `pedido`
--
ALTER TABLE `pedido`
  ADD CONSTRAINT `order_fk_1` FOREIGN KEY (`username`) REFERENCES `user`
  (`username`) ON DELETE CASCADE ON UPDATE CASCADE;

--
-- Constraints for table `purchase`
--
ALTER TABLE `purchase`
  ADD CONSTRAINT `purchase_ibfk_1` FOREIGN KEY (`tabacol_code`) REFERENCES
  `tabaco` (`id`) ON DELETE CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `purchase_ibfk_2` FOREIGN KEY (`id_order`) REFERENCES
  `pedido` (`id`) ON DELETE CASCADE ON UPDATE CASCADE;
COMMIT;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;

```

```
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;  
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```



Este es el modelo relacional

Protocolo HTTP	URL	Acción	HTTP Status
GET	/users	Devuelve todos los usuarios	200 - OK 204 - Not Content: cuando no hay usuarios a listar
GET	/user/{idUser}	Devuelve un usuario con detalle	200 - OK 404 - Not Found: cuando no existe el usuario buscado
POST	/user	Añade un usuario	201 - Created
PUT	/user/{idUser}	Modifica un usuario completamente	200 - OK 400 - Bad request: cuando algún parámetro enviado no cumple con las reglas
PATCH	/user/{idUser}	Modifica solo los campos que nosotros pasamos	200 - OK 201 - Created: si el recurso a modificar no existe se crea en automático 400 - Bad request: cuando algún parámetro enviado no cumple con las reglas
DELETE	/user/{idUser}	Elimina el usuario	201 - Not Content:

		por el id	es el status standard a regresar en un verbo DELETE
--	--	-----------	-----------------------------------------------------

Protocolo HTTP	URL	Acción	HTTP Status
GET	/category	Devuelve todas las categorías	200 - OK 204 - Not Content: cuando no hay categorías a listar
GET	/category/{idCategory}	Devuelve una categoría con detalle	200 - OK 404 - Not Found: cuando no existe el categorías buscado
POST	/category	Añade una categoría	201 - Created
PUT	/category/{idCategory}	Modifica una categoría completamente	200 - OK 400 - Bad request: cuando algún parámetro enviado no cumple con las reglas
PATCH	/category/{idCategory}	Modifica solo los campos que nosotros pasamos	200 - OK 201 - Created: si el recurso a modificar no existe se crea en automático 400 - Bad request: cuando algún parámetro enviado no cumple con las reglas

DELETE	/category/{idCategory}	Elimina la categoría por el id	201 - Not Content: es el status standard a regresar en un verbo DELETE
--------	------------------------	--------------------------------	------------------------------------------------------------------------

Protocolo HTTP	URL	Acción	HTTP Status
GET	/tabacos	Devuelve todos los tabacos	200 - OK 204 - Not Content: cuando no hay usuarios a listar
GET	/tabacos/{idCategory}	Devuelve una lista de tabacos con el id de la category con detalle	200 - OK 404 - Not Found: cuando no existe el usuario buscado
POST	/tabaco	Añade un tabaco	201 - Created
PUT	/tabaco/{idTabaco}	Modifica un tabaco completamente	200 - OK 400 - Bad request: cuando algún parámetro enviado no cumple con las reglas
PATCH	/tabaco/{idTabaco}	Modifica solo los campos que nosotros pasamos	200 - OK 201 - Created: si el recurso a modificar no existe se crea en automático 400 - Bad request: cuando algún parámetro enviado no cumple con las reglas

DELETE	/tabaco/{idTabaco}	Elimina el tabaco por el id	201 - Not Content: es el status standard a regresar en un verbo DELETE

Protocolo HTTP	URL	Acción	HTTP Status
GET	/pedidos	Devuelve todos los pedidos	200 - OK 204 - Not Content: cuando no hay usuarios a listar
GET	/pedido/{idPedido}	Devuelve un pedido con detalle	200 - OK 404 - Not Found: cuando no existe el usuario buscado
POST	/pedido	Añade un pedido	201 - Created

Protocolo HTTP	URL	Acción	HTTP Status
GET	/compras	Devuelve todas las compras	200 - OK 204 - Not Content: cuando no hay usuarios a listar
GET	/compra/{idCompra}	Devuelve una compra con detalle	200 - OK 404 - Not Found: cuando no existe el usuario buscado
POST	/compra	Añade una compra	201 - Created

