

PROYECTO FINAL GRUPAL - Comisión 9, Grupo 4

Integrantes:

- DIAZ ROSSINI JUAN JOSE - Legajo 61744
- NAVARRO VICTOR LEANDRO - Legajo 61550
- GALLO GENARO - Legajo 61909
- LIACOPLO GERONIMO EMILIANO - Legajo 61306

Título del Proyecto: E-Commerce para Ferretería

Descripción:

Este proyecto consiste en el desarrollo de una plataforma web orientada a la gestión y venta de productos de ferretería en línea (E-Commerce). }

El sistema permite registrar usuarios, administrar productos, visualizar categorías, generar ventas y gestionar carritos de compra.

Tecnologías Utilizadas:

Backend:

- Node.js con Express.js como framework principal
- mysql2 como cliente para conexión a base de datos MySQL
- cors para habilitar el acceso entre dominios
- nodemon para reinicio automático del servidor en desarrollo

Frontend:

- HTML5
- CSS3 (diseño responsivo y visual amigable)
- JavaScript puro para manejar interacciones y peticiones asincrónicas (axios)

FUNCIONALIDADES DEL SISTEMA

El sistema está dividido en dos experiencias principales: la experiencia del usuario administrador y la del usuario cliente. Ambas comparten una misma base de datos MySQL, pero acceden a rutas y funciones diferenciadas según su rol.

El ingreso de los usuarios se realiza desde el archivo login.html, donde se controla el acceso mediante verificación contra la tabla de usuarios. Si el usuario posee rol "admin", se redirige automáticamente a admin.html. Caso contrario, se redirige a index.html, donde inicia su experiencia como comprador. El archivo register.html permite registrar nuevos usuarios desde la interfaz pública. Ambos archivos (login.html y register.html) contienen validaciones y control de sesiones a través de la tabla sesiones ...

GESTIÓN ADMINISTRATIVA

Una vez en admin.html, el administrador dispone de tres módulos CRUD bien definidos:

- Gestión de Empleados

Permite crear, ver, actualizar y eliminar empleados. Cada acción impacta directamente sobre la tabla empleados. Se pueden visualizar imágenes, contactos y correos de los empleados registrados.

- Gestión de Productos

Permite crear nuevos productos, asignarles una categoría, establecer su precio, stock e imagen. El administrador puede actualizar cualquier dato del producto, o eliminarlo en caso de ser necesario. Todos estos datos se guardan en la tabla de productos.

- Gestión de Usuarios

Similar a la gestión de empleados, este módulo permite mantener actualizada la información de los usuarios. No se permite la edición de roles a través de la interfaz por razones de seguridad.

Adicionalmente, el módulo de ventas permite al administrador visualizar todas las compras realizadas por los usuarios, con sus respectivos productos, cantidades y totales. Si bien puede eliminar ventas para depuración o corrección, no puede crear ni modificar ventas manualmente, ya que éstas provienen exclusivamente de interacciones reales con el carrito de compras.

Cada uno de estos módulos implementa una barra de búsqueda por ID, permitiendo filtrar de manera ágil los registros. Esta funcionalidad está presente en todos los listados administrativos.

EXPERIENCIA DEL USUARIO COMÚN

El usuario común accede a la tienda desde index.html, donde se le presenta inicialmente un carrusel de imágenes, diseñado para mostrar memes o publicidades visuales relacionadas con el rubro ferretero. Este carrusel es completamente funcional y cambia automáticamente cada 5 segundos, además de contar con botones de navegación manual.

Debajo del carrusel, se renderizan todos los productos disponibles de forma dinámica y en formato de cards. Cada card muestra:

- Imagen del producto
- Nombre
- Categoría
- Stock disponible
- Precio
- Botón para añadir al carrito

La información del producto proviene directamente desde la base de datos, a través del endpoint /productos, y la categoría se resuelve mediante un join a la tabla categorías.


El botón para añadir al carrito inserta directamente el producto correspondiente en la tabla carrito, haciendo uso del ID_Usuario actualmente activo (obtenido por sesión). Si el producto ya está presente en el carrito, simplemente se incrementa la cantidad, evitando duplicados.

INTERFAZ Y NAVEGACIÓN

El sistema cuenta con un header fijo en todas las vistas (login.html, register.html, index.html) que contiene:

- Enlace a "Iniciar sesión"
- Enlace a "Inicio" que redirige al catálogo
- Enlace "Contacto" que desplaza al footer con información de contacto

Si el usuario ha iniciado sesión, se muestran adicionalmente:

- Su nombre y foto de perfil
- Un botón de engranaje que redirige a perfil.html con el ID del usuario por URL (ejemplo: perfil.html?id=3)
- Un botón con ícono de carrito  ubicado en el header, a la derecha del todo, que permite acceder al carrito sin riesgo de ser destruido por la renderización del main. Esta decisión técnica se tomó para preservar persistencia y evitar colisiones de interfaz durante la carga de la SPA.


EDICIÓN DE PERFIL



La vista perfil.html permite al usuario modificar sus datos personales: nombre, contraseña, teléfono, correo electrónico e imagen de perfil. Los campos se actualizan en tiempo real en una vista previa ubicada al costado del formulario. La edición se realiza mediante método PUT hacia la ruta /admin/usuarios/update/:id, excluyendo el campo rol por seguridad. Un botón "Cancelar" permite volver a la tienda sin guardar los cambios.

CARRITO DE COMPRAS (SPA)

El carrito de compras fue implementado como una Single Page Application (SPA), es decir, se renderiza dinámicamente dentro de la misma vista index.html sin redirección. Esta decisión técnica permitió:

- Mayor velocidad de carga
- Flujo continuo sin interrupciones
- Control total del DOM para añadir o quitar productos

Desde el botón " Ir al carrito" ubicado en el header, el usuario accede a su carrito actual. El sistema consulta todos los productos de la tabla carrito asociados al ID_Usuario activo. Los productos se listan con:

- Imagen
- Nombre
- Precio
- Cantidad actual
- Subtotal
- Botones  y  para aumentar o reducir la cantidad

El botón "Finalizar compra" inicia un modal flotante que oscurece el fondo y muestra un comprobante de los productos que serán comprados, incluyendo sus cantidades y precios. El usuario puede:

- Cancelar, lo que cierra el modal y lo deja en el carrito
- Confirmar, lo que:
 - Inserta los datos en la tabla ventas
 - Calcula totales
 - Muestra un alert de confirmación
 - Vacía el carrito completamente

Toda esta lógica está conectada con rutas como `/admin/ventas/add`, `/carrito/delete/:id`, `/carrito/update/:id`, `/carrito/clear` y es completamente persistente en MySQL.

CONCLUSIÓN

El sistema desarrollado ofrece una solución integral para la gestión de una ferretería tanto a nivel administrativo como desde la experiencia del usuario comprador. Se implementaron buenas prácticas como separación de responsabilidades, modularidad en el código, consumo de API REST y diseño responsivo. El uso de sesiones, autenticación basada en roles, y control de acceso garantiza un entorno seguro, robusto y escalable.

Fecha de documentación: 2025-06-29