## TRATAMIENTO DE ENTIDADES

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Create** | **Read** | **Update** | **Delete** |
| **Usuario** | X | X | X | X |
| **Dirección** | X X | X X | X | X |
| **Método de Pago** | X | X X | X | X |
| **Pedido** | X | X |  |  |
| **Producto** | X | X X X | X | X |
| **Producto Montable** | X | X |  |  |

## PUNTOS DONDE SE MANIPULAN ENTIDADES

1. USUARIO:
   1. CREAR:
      1. Localización: register.php
      2. Acceso: Haciendo clic en el botón Login/Register, ya sea en el incio de la página o al final de un pedido.
      3. Casos de uso: CU02.
   2. LEER:
      1. Localización: account.php
      2. Acceso: Loguearse. Después, acceder a “My Account”.
      3. Casos de uso: CU04.
   3. ACTUALIZAR:
      1. Localización: account.php
      2. Acceso: Loguearse. Después, acceder a “My Account”, introducir valores válidos y pinchar en “Update Account”.
      3. Casos de uso: CU06.
   4. BORRAR:
      1. Localización account.php
      2. Acceso: Acceso: Loguearse. Después, acceder a “My Account”, pinchar en “Delete Account”.
      3. Casos de uso: CU15.
2. DIRECCIÓN:
   1. CREAR 1:
      1. Localización: addresses.php.
      2. Acceso: Estando el usuario logueado, debe acceder a “Mi Cuenta”, y luego a “Add address”.
      3. Casos de uso: CU16.
   2. CREAR 2:
      1. Localización: register.php.
      2. Acceso: Haciendo clic en el botón Login/Register, y luego en “Register”, y rellenando el formulario.
      3. Casos de uso: CU02.
   3. LEER 1:
      1. Localización: account.php
      2. Acceso: Loguearse. Después, acceder a “My Account”.
      3. Casos de uso: CU04.
   4. LEER 2:
      1. Localización: order.php
      2. Acceso: Crear un pedido, loguearse y acceder a “Cart”.
      3. Casos de uso: CU05.
   5. ACTUALIZAR:
      1. Localización: addresses.php
      2. Acceso: Loguearse. Después, acceder a “My Account”, pinchar en “Update Address”, introducir valores válidos y pinchar en “Update Address”.
      3. Casos de uso: CU18.
   6. BORRAR:
      1. Localización: address.php
      2. Acceso: Loguearse. Después, acceder a “My Account”, pinchar en “Delete Address”.
      3. Casos de uso: CU17.
3. PRODUCTO:
   1. CREAR:
      1. Localización: manageProduct.php
      2. Acceso: Tras logearse como administrador accedemos a “My Account” y pinchamos en “Add Product”.
      3. Casos de uso: CU12.
   2. LEER 1:
      1. Localización: manageProduct.php.
      2. Acceso: Tras logearse como administrador accedemos a “My Account” y pinchamos en “Edit Product” o “Delete Product”.
      3. Casos de uso: CU03.
   3. LEER 2:
      1. Localización: order.php.
      2. Acceso: Entramos a “build computer” y construimos un ordenador. Aceptamos y damos a “Confirm order”. El sistema buscará los productos en la base de datos para calcular los precios.
      3. Casos de uso: CU10.
   4. LEER 3:
      1. Localización: account.php
      2. Acceso: Loguearse. Después, acceder a “My Account”. Navegar hasta “My Orders”.
      3. Casos de uso: CU04.
   5. ACTUALIZAR:
      1. Localización: manageProduct.php.
      2. Acceso: Tras logearse como administrador accedemos a “My Account” y pinchamos en “Edit Product”.
      3. Casos de uso: CU13.
   6. BORRAR:
      1. Localización: manageProduct.php
      2. Acceso: Tras logearse como administrador accedemos a “My Account” y pinchamos en “Delete Product”.
      3. Casos de uso: CU14.
4. METODO DE PAGO:
   1. CREAR:
      1. Localización: paymentMethod.php
      2. Acceso: Estando el usuario logueado, debe acceder a “My Account”, y luego a “Add Payment Method”.
      3. Casos de uso: CU07.
   2. LEER 1:
      1. Localización: manageAccount.php
      2. Acceso: Loguearse. Después, acceder a “My Account”.
      3. Casos de uso: CU04.
   3. LEER 2:
      1. Localización: order.php
      2. Acceso: Crear un pedido, loguearse y acceder a “Cart”.
      3. Casos de uso: CU05.
   4. ACTUALIZAR
      1. Localizacion: paymentMethod.php
      2. Acceso: Loguearse. Después, acceder a “My Account”, pinchar en “Update Payment Method”, introducir valores válidos y pinchar en “Update Payment Method”.
      3. Casos de uso: CU09.
   5. BORRAR:
      1. Localización: paymentMethod.php
      2. Acceso: Loguearse. Después, acceder a “My Account”, pinchar en “Delete Address”.
      3. Casos de uso: CU08.
5. PRODUCTO MONTABLE:
   1. CREAR:
      1. Localización: order.php.
      2. Acceso: Entramos a “build computer” y construimos un ordenador. Aceptamos y damos a “Confirm order”.
      3. Casos de uso: CU10.
   2. LEER:
      1. Localización: account.php
      2. Acceso: Loguearse. Después, acceder a “My Account”. Navegar hasta “My Orders”.
      3. Casos de uso: CU04.
6. PEDIDO:
   1. CREAR:
      1. Localización: order.php.
      2. Acceso: Entramos a “build computer” y construimos un ordenador. Aceptamos y damos a “Confirm order”.
      3. Casos de uso: CU10.
   2. LEER:
      1. Localización: account.php
      2. Acceso: Loguearse. Después, acceder a “My Account”. Navegar hasta “My Orders”.
      3. Casos de uso: CU04.

## APORTACIONES JAVASCRIPT

1. Validación de los campos de formularios que entran en contacto con los usuarios clientes:
   1. Objetivo/Justificación: Mejorar la experiencia de usuario a la hora de rellenar los campos de los formularios, de modo que el usuario sepa exactamente qué campos son válidos y cuáles no, y por qué, antes de clicar en “Enviar”.
   2. Implementación: Se ha llevado a cabo con un plugin jQuery[[1]](#footnote-1) que permite especificar para cada input restricciones y mensajes de error específicos para cada una de ellas.
2. Campo de búsqueda de productos con filtrado para los administradores:
   1. Objetivo/Justificación: Facilita el trabajo del administrador a la hora de buscar un producto para editarlo y borrarlo ya que filtra dependiendo de lo que se introduzca en el campo.
   2. Implementación: Dependiendo del campo de búsqueda que estamos rellenando, filtramos la tabla según tipo de producto o categoría recorriendo la tabla de productos.
3. Validación de los métodos de pago de los clientes:
   1. Objetivo/Justificación: Evita que los introduzcan tarjetas de crédito falsas o caducadas.
   2. Implementación: Se ha llevado a cabo con un plugin de jQuery[[2]](#footnote-2) que comprueba cada uno de los campos para comprobar que la tarjeta es correcta, además se ha modificado para adaptarlo a las necesidades del proyecto.
4. Confirmación de acciones de borrado irreversibles:
   1. Objetivo/Justificación: Añade una capa más de protección frente a acciones de borrado e irreversibles que puedan ser indeseadas, para mejorar la experiencia de usuario.
   2. Implementación: Se ha llevado a cabo empleando la funcionalidad “confirm” de Javascript en el momento de enviar los formularios.

## APORTACIONES EXTRAORDINARIAS

### USOS DE CSS

1. Diseño responsive: Para realizar un diseño atractivo y que se adapte a cualquier dispositivo, se ha empleado la plantilla CSS ofrecida por W3School[[3]](#footnote-3). En concreto, la aplicación se mostrará en modo compacto si la anchura de la ventana es menor a 800 píxeles.

### USOS DE HTML5

1. Elementos semánticos <header>, <footer>, <article> y <section>: ayudan a organizar la información claramente en la página y a aplicar el estilo apropiado.
   1. Localización: Usados en todas las páginas de la aplicación
2. Atributos para los formularios, como “number” y “required” entre otros: usados como primera línea de defensa para mejorar la experiencia de usuario.
   1. Localización: Usados en todas las páginas que contienen formularios.
3. Atributo para los inputs de formulario “form”, que permite colocar inputs fuera de la etiqueta del formulario, pero que apunten a él. Se ha empleado para hacer una tabla donde cada fila es un formulario, respetando la condición de que <form> no puede estar dentro de <tr>.
   1. Localización: Usado en manageProducts.php.

### USOS DE JQUERY

1. Validación de formularios: La segunda línea de defensa para mejorar la experiencia de usuario se realiza con validaciones de formularios usando un plugin de jQuery, que indica exactamente en qué falla el formulario cuando el usuario lo deselecciona.
2. Validación de tarjetas de crédito: Se utiliza un plugin[[4]](#footnote-4) para asegurarnos de que se solamente se introducen tarjetas de VISA, MASTERCARD Y AMEX. Además, se ha modificado el plugin para adaptarlo a las necesidades del proyecto.
3. Confirmación de acciones de borrado irreversibles: Se han empleado selectores y funciones jQuery para facilitar la implementación de esta funcionalidad.

### TECNOLOGÍAS NO VISTAS EN LA ASIGNATURA

1. Git/GitHub: El equipo ha empleado el sistema de control de versiones Git para controlar y compartir el trabajo, alojando el proyecto el un repositorio remoto de GitHub. El repositorio puede consultarse en: <https://github.com/mridpin/PA_Group7_Project>

1. El plugin está disponible en <https://jqueryvalidation.org/>

   2 El plugin está disponible en <https://tutorialzine.com/2016/11/simple-credit-card-validation-form> [↑](#footnote-ref-1)
2. [↑](#footnote-ref-2)
3. <https://www.w3schools.com/w3css/default.asp> [↑](#footnote-ref-3)
4. [↑](#footnote-ref-4)