

UNIVERSIDAD LUTERANA SALVADOREÑA

FACULTAD DE CIENCIAS DEL HOMBRE Y SU NATURALEZA



CÁTEDRERA:

Base de datos 1

CATEDRÁTICO:

LICENCIADO: David Clímaco

CICLO:

I/ 2020

ESTUDIANTES:

Yovany Alexander García Castro

Francisco Javier Mejía Ortiz

Daniel Alonso Muñoz Márquez

Marlon Miguel Meléndez Mejía

TABLA DE CONTENIDO

INTRODUCCION	1
OBJETIVOS	2
PERFIL DE LA EMPRESA.....	3
LUGAR DE UVICACION DE LA TIENDA.....	3
¿QUÉ ES CRUD?.....	4
FRAMEWORKS CRUD: CAPA DE ACCESO A LAS BASES DE DATOS	5
HTML	5
CSS	6
¿CÓMO FUNCIONAN?.....	7
TIPOS DE HOJAS DE ESTILO CSS.....	7
¿CÓMO AFECTA A MI WEB EL USO DE LAS HOJAS DE ESTILO?	8
XAMPP	8
¿PARA QUÉ SIRVE?.....	9
PHP EL PHP	9
CONCLUSION	10
ANEXOS.....	11
CICLO DE TRABAJO	11
DIAGRAMA	11
DIAGRAMA ENTIDAD RELACIÓN	12
REFERENCIAS.....	18

INTRODUCCION

En el presente trabajo se detalla las herramientas que se utilizan para la elaboración de un CRUD sus resultados al digitar el código adecuado, además documentación del proyecto en su elaboración y finalización, el proyecto está dirigido también a todas las personas que estén interesadas en comenzar el mundo de la programación ya que la elaboración de este no es muy compleja, además este documento detalla una breve descripción de Cada programa que se utilizará para la elaboración de este CRUD, además este documento no es una guía exacta para aprender a programar directamente, ya que se limita solo a explicar directamente las herramientas que se utilizaran y también la descripción de la micro mediana empresa que se detallaran sus diferentes funciones en lo laboral y este crud tendrá con fin aportar una herramienta que se capas de guardar almacenar, borrar, editar, actualizar datos de la tienda de productos, que más adelante se detallaran más afondo.

OBJETIVOS

Objetivo general.

- Realizar un CRUD que realice las funciones básicas como lo son agregar, eliminar, actualizar y editar.

Objetivo específico

- Obtener un programa utilizando PHP, MYSQL, HTML, CSS para así cumplirlas funciones básicas de un C.R.U.D.
- Obtener un CRUD que funcione correctamente, a la vez que sea atractivo visualmente obteniendo la satisfacción del usuario.

PERFIL DE LA EMPRESA.

La micro mediana empresa se dedica. A la venta de productos como sandalias Zapatos cerrados par hombre y mujer también cada uno de los estilos tiene diferente precio muy accesible para todos los clientes. Aparte de tenis juvenil, también tenemos una serie de tenis para niños desde 1 año hasta los 12 E igual para niña, con una gran variedad de estilos los cuales no solo están disponibles para los clientes que visitan la tienda también los productos los puedes encontrar en línea ya que también además nos dedicamos a la venta de ropa de mujer ya sea blusas o jeans las venta varían según las temporadas. Ya que al principio de enero se tuvo más de 200 bolsones escolares tanto para jóvenes o señoritas.

LUGAR DE UVICACION DE LA TIENDA

Estamos ubicados en "Santiago Nonualco " departamento de la paz. Calle principal 1° avenida panamericana frente a casa de la cultura de Santiago nonualco

¿QUÉ ES CRUD?

El concepto CRUD está estrechamente vinculado a la gestión de datos digitales. CRUD hace referencia a un acrónimo en el que se reúnen las primeras letras de las cuatro operaciones fundamentales de aplicaciones persistentes en sistemas de bases de datos:

Create (Crear registros)

Read bzw. Retrieve (Leer registros)

Update (Actualizar registros)

Delete bzw. Destroy (Borrar registros)

En pocas palabras, CRUD resume las funciones requeridas por un usuario para crear y gestionar datos. Varios procesos de gestión de datos están basados en CRUD, en los que dichas operaciones están específicamente adaptadas a los requisitos del sistema y de usuario, ya sea para la gestión de bases de datos o para el uso de aplicaciones. Para los expertos, las operaciones son las herramientas de acceso típicas e indispensables para comprobar, por ejemplo, los problemas de la base de datos, mientras que para los usuarios, CRUD significa crear una cuenta (create) y utilizarla (read), actualizarla (update) o borrarla (delete) en cualquier momento. Dependiendo de la configuración regional, las operaciones CRUD pueden implementarse de diferentes maneras, como lo muestra la siguiente tabla.

CRUD-Operation SQL RESTful HTTP XQuery

Create INSERT POST, PUT insert Read

SELECT GET, HEAD copy/modify/return Update

UPDATE PUT, PATCH replace, rename Delete DELETE delete

FRAMEWORKS CRUD: CAPA DE ACCESO A LAS BASES DE DATOS

Si los objetos individuales son visualizados por medio de una interfaz gráfica y modificados con las llamadas operaciones CRUD, entonces se habla de un framework CRUD o de un CRUD grid. Por lo general, se trata de interfaces HTML. Un framework CRUD demanda varios pasos de transacción, de forma que los datos no se recogen una vez se han introducido, sino que es necesario pulsar la opción “Guardar” o “Continuar”. Las operaciones de un framework CRUD pueden aplazarse para ser ejecutadas en diferentes plazos, sin que los datos de dichos periodos de tiempo se bloqueen para otros usuarios.

Este hecho resulta de gran importancia para sistemas multiusuario, pues permite que varias personas lean los mismos datos al mismo tiempo. Para llevar a cabo las operaciones se utilizan las denominadas capas de persistencia, que, por lo general, pueden ser implementadas o están contenidas en forma de extensiones (módulos) en el framework. Estas rompen con la representación relacional y tabular de la totalidad de los datos para, en su lugar, presentarlos en un nivel orientado a objetos. Los frameworks CRUD facilitan el acceso al sistema de bases de datos y son utilizados tanto en el desarrollo como en el uso de aplicaciones. Existen numerosos frameworks con un concepto CRUD basados en diferentes lenguajes y platafor

HTML

Documentos para Hipertexto. HTML es un lenguaje de marcado que se utiliza para el desarrollo de páginas de Internet. Se trata de la sigla que corresponde a HyperText Markup Lenguaje, es decir, Lenguaje de Marcas de Hipertexto, que podría ser traducido como Lenguaje de Formato de Se trata de un formato abierto que surgió a partir de las etiquetas SGML (Standard Generalized Markup Language).

Concepto traducido generalmente como «Estándar de Lenguaje de Marcado Generalizado» y que se entiende como un sistema que permite ordenar y etiquetar diversos documentos dentro de una lista. Este lenguaje es el que se utiliza para especificar los nombres de las etiquetas que se utilizarán al ordenar, no 6 EL HTML se encarga de desarrollar una descripción sobre los contenidos

que aparecen como textos y sobre su estructura, complementando dicho texto con diversos objetos (como fotografías, animaciones, etc).

Es un lenguaje muy simple y general que sirve para definir otros lenguajes que tienen que ver con el formato de los documentos. El texto en él se crea a partir de etiquetas, también llamadas tags, que permiten interconectar diversos conceptos y formatos. Para la escritura de este lenguaje, se crean etiquetas que aparecen especificadas a través de corchetes o paréntesis angulares: < y >. Entre sus componentes, los elementos dan forma a la estructura esencial del lenguaje, ya que tienen dos propiedades (el contenido en sí mismo y sus atributos). Por otra parte, cabe destacar que el HTML permite ciertos códigos que se conocen como scripts, los cuales brindan instrucciones específicas a los navegadores que se encargan de procesar el lenguaje. Entre los scripts que pueden agregarse, los más conocidos y utilizados son JavaScript y PHP.

El marcado estructural es el que estipula la finalidad del texto, aunque no define cómo se verá el elemento. El marcado presentacional, por su parte, es el que se encarga de señalar cómo se verá el texto más allá de su función. Para conocer el código HTML que utiliza una página web, hay que seleccionar Ver código fuente en nuestro navegador (como Internet Explorer o Mozilla Firefox). Al elegir esta opción, se abrirá el editor de texto con el código HTML de la página que se está visualizando.

CSS

Las hojas de estilo en cascada o CSS, son un simple mecanismo que describe cómo se va a mostrar un documento en la pantalla, cómo se va a imprimir o incluso cómo se debe llevar a cabo su lectura desde dispositivos habilitados para ello. Es decir, son líneas de código HTML o XML que están orientadas a darle un estilo específico a los elementos de un documento HTML, las utilizaremos para personalizar nuestra web (colores, backgrounds, fuentes, enlaces...) con la finalidad de mantener limpio y estructurado el código fuente sin carga de código que dificulte su lectura o modificación.

Las especificaciones de las hojas de estilo han sido creadas por W3C (World Wide Web consortium), organización encargada de definir todos los estándares de la web y creada en 1994 por Tim Berners Lee, con la finalidad de validar el

código en HTML y conseguir un buen posicionamiento en buscadores. Asimismo define los estándares que un sitio web debe seguir para alcanzar un diseño responsive.

¿Cómo funcionan?

Como ya hemos comentando CSS se usan para dar estilo a documentos HTML y XML, separando lo que es el contenido propiamente dicho de la presentación del mismo. CSS permitirá a los desarrolladores controlar el estilo y el formato de múltiples páginas Web al mismo tiempo, como se indica en W3C,

Hasta la aparición de las hojas de estilo, los desarrolladores web debían revisar todo el código HTML de cada una de las páginas en las que querían modificar, por pequeña fuera, la presentación de un determinado elemento, lo que conllevaba entre otras cosas una gran pérdida de tiempo y costes. Con el nacimiento de CSS, desaparecen las decenas de líneas repetitivas en el código fuente, necesarias para darle estilo a una determinada etiqueta.

TIPOS DE HOJAS DE ESTILO CSS

Existen tres tipos diferentes de hojas de estilo:

- Externas: Están definidas en un documento aparte con extensión .css y que se vincula mediante la etiqueta <link> en la sección <head> del documento html.
- Internas: Se definen en el interior del documento al que se quiere dar estilo utilizando el elemento <style> dentro de la sección <head>, de manera que los estilos sean reconocidos antes de que la página se cargue por completo.
- Inline: Se define el estilo directamente sobre aquellos elementos que lo permiten a través del atributo <style> y dentro de <body>. En este caso se perdería la ventaja que ofrecen los otros tipos de CSS, de hacer un código reutilizable y limpio.

Pero hablar de CSS no es simplemente comprender el manejo de colores, imágenes y transparencias para montar y estructurar páginas web, va más allá,

se trata de reducir el tamaño de una web, disminuir el tiempo de carga en los diferentes dispositivos con lo que se puede visualizar una página, reducir el procesamiento en los navegadores así como mejorar la accesibilidad y el orden y legibilidad del código.

¿CÓMO AFECTA A MI WEB EL USO DE LAS HOJAS DE ESTILO?

Los beneficios de usar CSS son varios, por un lado evitamos hacer archivos demasiados pesados (adiós a la espera en la carga de nuestro sitio web en el navegador, que tan tediosa resulta para el usuario final), ahora definimos el estilo de un sitio entero sin necesidad de hacerlo para cada una de las etiquetas de la web, lo que nos lleva a otra de las ventajas, la reutilización del código y la mejora en la legibilidad, orden y estructuración de los archivos que forman un proyecto de desarrollo web.

Por otro lado trabajamos con estándares, lo que nos garantiza que la mayoría de navegadores disponibles en el mercado van a ajustarse a los mismos, mostrando nuestra página web tal y como la hemos diseñado, presentando al usuario final nuestro “*sello*” sin cambios en la estructura, colores o maquetación del entorno.

¿QUÉ ES XAMPP Y PARA QUÉ SIRVE?

XAMPP

Es un servidor independiente de plataforma de código libre. Te permite instalar de forma sencilla Apache en tu propio ordenador, sin importar tu sistema operativo (Linux, Windows, MAC o Solaris). Y lo mejor de todo es que su uso es gratuito.

XAMPP incluye además servidores de bases de datos como MySQL y SQLite con sus respectivos gestores phpMyAdmin y phpSQLiteAdmin. Incorpora también el intérprete de PHP, el intérprete de Perl, servidores de FTP como ProFTPD ó FileZilla FTP Serve, etc. entre muchas cosas más.

Si alguna vez has intentado instalar Apache, sabes que no es una tarea fácil, sin embargo con XAMPP todo es diferente. Una de las ventajas de usar XAMPP es que su instalación es de lo más sencilla, basta descargarlo, extraerlo y comenzar a usarlo.

¿PARA QUÉ SIRVE?

Como ya lo debes haber deducido, XAMPP es una herramienta de desarrollo que te permite probar tu trabajo (páginas web o programación por ejemplo) en tu propio ordenador sin necesidad de tener que acceder a internet. Si eres un desarrollador que recién está comenzando, tampoco debes preocuparte sobre las configuraciones ya que XAMPP te provee de una configuración totalmente funcional desde el momento que lo instalas (básicamente lo extraes). Sin embargo, es bueno acotar que la seguridad de datos no es su punto fuerte, por lo cual no es suficientemente seguro para ambientes grandes o de producción. Paquetes que vienen con XAMPP Paquetes básicos: Apache, el servidor Web más famoso.

MySQL, una excelente base de datos de código libre. PHP y Perl: lenguajes de programación. ProFTPD: un servidor FTP. OpenSSL: para soporte a la capa de sockets segura. Paquetes gráficos: GD (Graphics Draw): la librería de dibujo de gráficos. libpng: la librería oficial de referencia de PNG. libjpeg: la librería oficial de referencia de JPEG. ncurses: la librería de gráficos de caracteres 8

PHP EL PHP

Es un lenguaje de scripting de código abierto, del lado del servidor, con programación HTML integrada que se utiliza para crear páginas web dinámicas.

Las ventajas de PHP son su flexibilidad y su alta compatibilidad con otras bases de datos. Además, PHP es considerado como un lenguaje fácil de aprender. Las funciones de PHP se relacionan con los scripts del lado del servidor. PHP puede realizar cualquier tarea que cualquier programa CGI (Common Gateway Interface) puede hacer y maneja el intercambio de datos entre el servidor y el software. Por lo tanto, PHP puede recopilar datos o crear webs de contenido dinámico. Tres funciones principales distinguen en particular el PHP:

- Programación del lado del servidor: esto requiere tres componentes, que son un analizador PHP, un navegador web y un servidor web. Este último está conectado con una instalación PHP. El navegador web se utiliza para acceder al programa PHP. El navegador muestra páginas web que se almacenan en el servidor y se generan con PHP. La programación del lado del servidor es ideal para webs de prueba antes de su publicación en la web.

CONCLUSION

En este proyecto se mostró la elaboración de un crud el cual es una aplicación muy buena para adentrarse en mundo de la programación ya que por medio de este programa se pueda un mejor control para la venta de un producto y desempeñarse , mejor en el trabajo o para fines educativos ir de la mano con la tecnología hace toda actividad más dinámica y se mejora el ambiente con los que están a nuestro alrededor la tecnología es una herramienta fundamental para dar a conocer un determinado producto que se desea comercializar. Por otro lado también Se obtuvo el mayor conocimiento en los programas y herramientas que se utilizaron para dicho proyecto

Y parte de su descripción de las herramientas utilizadas en la elaboración del crud. Se pudo comprender que para poder crear un crud se necesita trabajar con HTML o XHTML, CSS, PHP, LIBRERIAS DE CODIGO, MYSQL y se caracterizan por su relación entre unas y otras para poder llevar a cabo un mejor trabajo y tener un mejor resultado.

ANEXOS

Ciclo de trabajo

2020	CALENDARIO DE TRABAJO 2020 CICLO DE TRABAJO DEL PROYECTO DE BASE DE DATOS I																																				
	L	M	M	J	V	S	D	L	M	M	J	V	S	D	L	M	M	J	V	S	D	L	M	M	J	V	S	D	L	M	M	J	V	S	D	L	M
Marzo							1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Abril			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30					
Mayo				1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31			
Junio	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30							

En presente cuadro se marcan los días en el que el grupo trabajo arduamente en el proyecto.

DIAGRAMA

Usando el programa workbench creamos la base de datos y las tablas. Según las relaciones que tenga la base de datos tendremos una diferente distribución de foreign keys para relacionar las tablas.

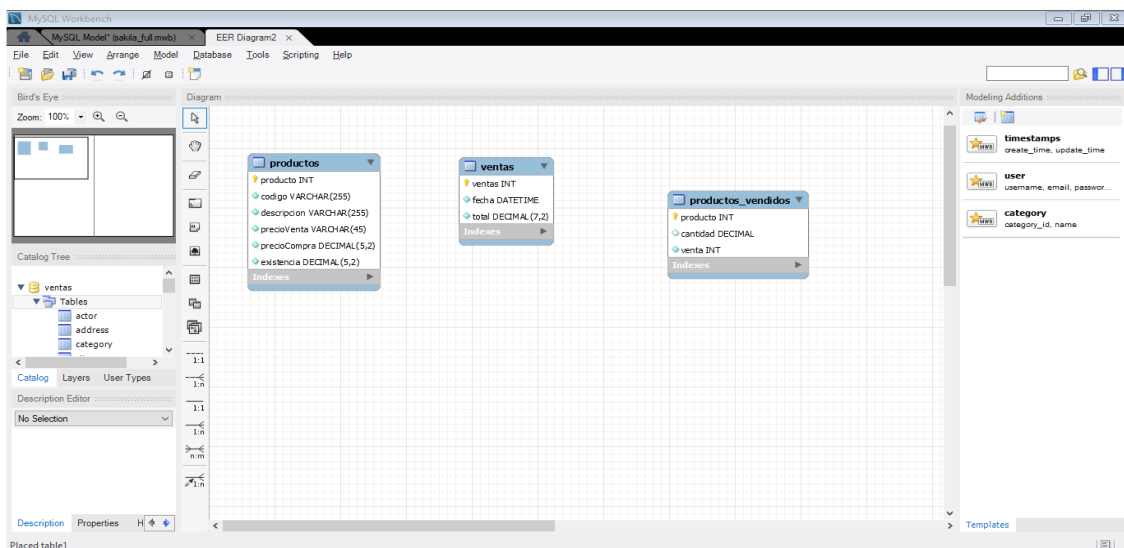
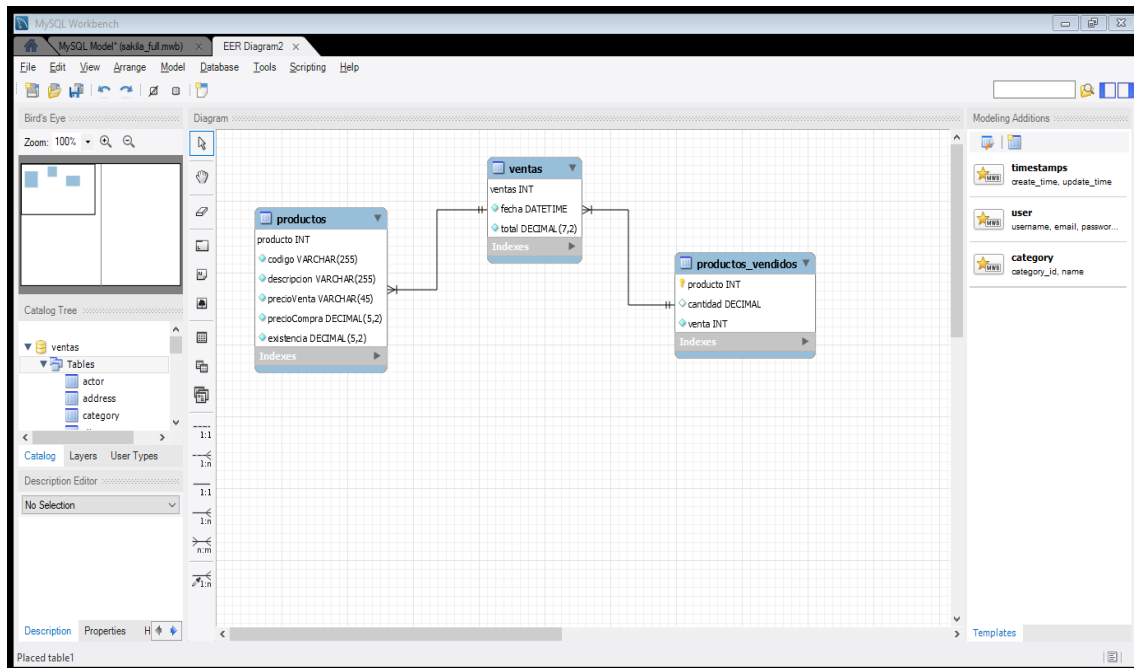


DIAGRAMA ENTIDAD RELACIÓN

Tablas entidad relación: aquí tenemos la relación que existe ente las tres tablas lo primero que se hizo fue crear la base de datos con su nombre ventas y luego las tablas y sus nombres "producto", "ventas", "productos_vendidos" a cada una se le creo sus campos con lo que se trabajaran para ingresar datos.



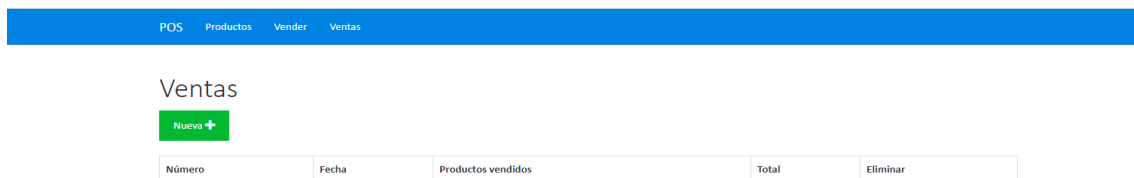
En la captura se muestra parte de la base de datos y sus respectivas tablas y sus campos que esta se llenaran de datos también la relación entre tablas también se muestra como insertar datos a la tabla productos los cuales son algunos de los productos como zapatos y sus características.

```
db_schema.sql (ventas_pdo-master) - Brackets
Ver  Navegación  Desarrollo  Ayuda  Java  Emmet
1  DROP DATABASE IF EXISTS ventas;
2  CREATE DATABASE IF NOT EXISTS ventas;
3  USE ventas;
4  CREATE TABLE productos(
5      id BIGINT UNSIGNED NOT NULL AUTO_INCREMENT,
6      codigo VARCHAR(255) NOT NULL,
7      descripcion VARCHAR(255) NOT NULL,
8      precioVenta DECIMAL(5, 2) NOT NULL,
9      precioCompra DECIMAL(5, 2) NOT NULL,
10     existencia DECIMAL(5, 2) NOT NULL,
11     PRIMARY KEY(id)
12 ) ENGINE = InnoDB DEFAULT CHARACTER SET = utf8;
13
14 CREATE TABLE ventas(
15     id BIGINT UNSIGNED NOT NULL AUTO_INCREMENT,
16     fecha DATETIME NOT NULL,
17     total DECIMAL(7,2),
18     PRIMARY KEY(id)
19 ) ENGINE = InnoDB DEFAULT CHARACTER SET = utf8;
20
21 CREATE TABLE productos_vendidos(
22     id BIGINT UNSIGNED NOT NULL AUTO_INCREMENT,
23     id_producto BIGINT UNSIGNED NOT NULL,
24     cantidad BIGINT UNSIGNED NOT NULL,
25     id_venta BIGINT UNSIGNED NOT NULL,
26     PRIMARY KEY(id),
27     FOREIGN KEY(id_producto) REFERENCES productos(id) ON DELETE CASCADE,
28     FOREIGN KEY(id_venta) REFERENCES ventas(id) ON DELETE CASCADE
29 ) ENGINE = InnoDB DEFAULT CHARACTER SET = utf8;
30
31 INSERT INTO productos(id, codigo, descripcion, precioVenta, precioCompra, existencia)
32 VALUES
33 (1, '1', 'NIKE AIR FORCE NUMERO DEL 7 AL 40', 50, 49, 100),
34 (2, '2',
35 Botas trekking Chiruca Xacobeo 13 GRIS/AZUL HOMBRE
36 , 80, 65, 100),
37 (3, '3', 'ZAPATILLAS MARRONES ARCHIVE TRAIL KUNA CREST DE TH', 70, 55, 100),
38 (4, '4', 'SANDALIAS CON LAZO', 35, 20, 100),
39 (5, '5', 'BOTAS TIMBERLAND DE CUERO MARRON', 60, 50, 100);
40
41 # Correcto
42
```

Y aquí tenemos la conexión de la base de datos.

```
base_de_datos.php (ventas_pdo-master) - Brackets
Navegación Desarrollo Ayuda Emmet
listar.php base_de_datos.php
1 <?php
2 /*
3 */
4 $contraseña = "";
5 $usuario = "root";
6 $nombre_base_de_datos = "ventas";
7 try{
8     $base_de_datos = new PDO('mysql:host=localhost;dbname=' . $nombre_base_de_datos, $usuario, $contraseña);
9     $base_de_datos->query("set names utf8;");
10    $base_de_datos->setAttribute(PDO::ATTR_EMULATE_PREPARES, FALSE);
11    $base_de_datos->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
12    $base_de_datos->setAttribute(PDO::ATTR_DEFAULT_FETCH_MODE, PDO::FETCH_OBJ);
13 }catch(Exception $e)
14
15 {
16     echo "Ocurrió algo con la base de datos: " . $e->getMessage();
17 }
18 ?>
```

Parte de la estructura del crud aquí tenemos la ventana que servirá para las ventas que se hicieron, que se irán guardando o incluso hacer una nueva venta está en proceso.



Parte de la estructura del código.

```

5     productos_vendidos_id_venta = ventas.id INNER JOIN productos ON productos.id = productos_vendidos.id_producto GROUP BY ventas.id ORDER BY
6     ventas.id);
7     $ventas = $sentencia->fetchAll(PDO::FETCH_OBJ);
8     ?>
9     <div class="col-xs-12">
10    <h1>Ventas</h1>
11    <div>
12    <a class="btn btn-success" href="./vender.php">Nueva <i class="fa fa-plus"></i></a>
13    </div>
14    <br>
15    <table class="table table-bordered">
16    <thead>
17    <tr>
18    <th>Número</th>
19    <th>Fecha</th>
20    <th>Productos vendidos</th>
21    <th>Total</th>
22    <th>Eliminar</th>
23    </tr>
24    </thead>
25    <tbody>
26    <?php foreach($ventas as $venta){ ?>
27    <tr>
28    <td><?php echo $venta->id ?></td>
29    <td><?php echo $venta->fecha ?></td>
30    <td>
31    <table class="table table-bordered">
32    <thead>
33    <tr>
34    <th>Código</th>
35    <th>Descripción</th>
36    <th>Cantidad</th>
37    </tr>
38    </thead>
39    <tbody>
40    <?php foreach(explode("___", $venta->productos) as $productosConcatenados){
41    $producto = explode(".", $productosConcatenados)
42    ?>
43    <tr>
44    <td><?php echo $producto[0] ?></td>
45    <td><?php echo $producto[1] ?></td>
46    <td><?php echo $producto[2] ?></td>

```

En esta parte del código tenemos donde se listan las ventas también tenemos una tabla y por cada producto hacemos un foreach, por los productos que se listen se imprimirán sus detalles.

```

7
8 <div class="col-xs-12">
9 <h1>Productos</h1>
10 <div>
11 <a class="btn btn-success" href="./formulario.php">Nuevo <i class="fa fa-plus"></i></a>
12 </div>
13 <br>
14 <table class="table table-bordered">
15 <thead>
16 <tr>
17 <th>ID</th>
18 <th>Código</th>
19 <th>Descripción</th>
20 <th>Precio de compra</th>
21 <th>Precio de venta</th>
22 <th>Existencia</th>
23 <th>Editar</th>
24 <th>Eliminar</th>
25 </tr>
26 </thead>
27 <tbody>
28 <?php foreach($productos as $producto){ ?>
29 <tr>
30 <td><?php echo $producto->id ?></td>
31 <td><?php echo $producto->codigo ?></td>
32 <td><?php echo $producto->descripcion ?></td>
33 <td><?php echo $producto->precioCompra ?></td>
34 <td><?php echo $producto->precioVenta ?></td>
35 <td><?php echo $producto->existencia ?></td>
36 <td><a class="btn btn-warning" href="<?php echo "editar.php?id=" . $producto->id?>"><i class="fa fa-edit">
37 </i></a></td>
38 <td><a class="btn btn-danger" href="<?php echo "eliminar.php?id=" . $producto->id?>"><i class="fa fa-trash">
39 </i></a></td>
40 </tr>
41 </tbody>
42 </table>

```


Si el producto no existe, lo indicamos. En caso de que sí, este formulario lo llenamos con el atributo `value` de la etiqueta `input`. Para el `textarea` es diferente, pues el contenido debe ir entre las etiquetas, no como atributo.

POS Productos Vender Ventas

Editar producto con el ID 4

Código de barras:

Descripción:

Precio de venta:

Precio de compra:

Existencia:

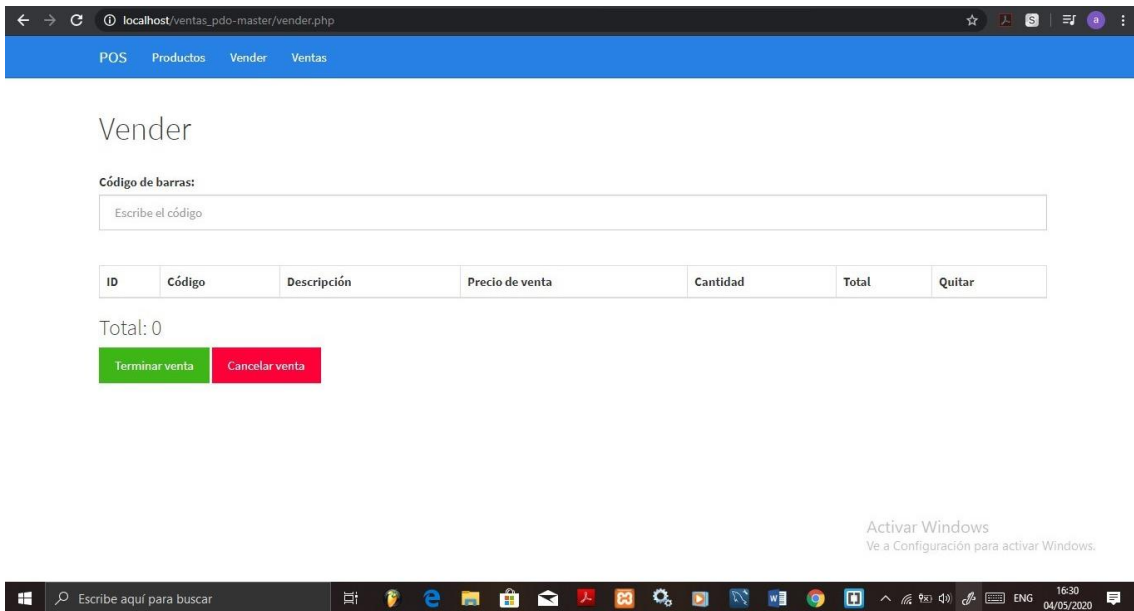
```
guardarDatosEditados.php (ventas_pdo-master) - Brackets
Navegación Desarrollo Ayuda Emmet
listar.php base_datos.php
1 <?php
2 #Salir si alguno de los datos no está presente
3
4 if(
5     isset($_POST["codigo"]) ||
6     isset($_POST["descripcion"]) ||
7     isset($_POST["precioCompra"]) ||
8     isset($_POST["precioVenta"]) ||
9     isset($_POST["existencia"]) ||
10    isset($_POST["id"])
11 ) exit();
12
13 #Si todo va bien, se ejecuta esta parte del código...
14
15 include_once "base_datos.php";
16 $id = $_POST["id"];
17 $codigo = $_POST["codigo"];
18 $descripcion = $_POST["descripcion"];
19 $precioCompra = $_POST["precioCompra"];
20 $precioVenta = $_POST["precioVenta"];
21 $existencia = $_POST["existencia"];
22
23 $sentencia = $base_datos->prepare("UPDATE productos SET codigo = ?, descripcion = ?, precioCompra = ?, precioVenta = ?,
24 existencia = ? WHERE id = ?");
25 $resultado = $sentencia->execute([$codigo, $descripcion, $precioCompra, $precioVenta, $existencia, $id]);
26
27 if($resultado === TRUE){
28     header("Location: ./listar.php");
29     exit();
30 }
31 else echo "Algo salió mal. Por favor verifica que la tabla exista, así como el ID del producto";
32 ?>
```

```

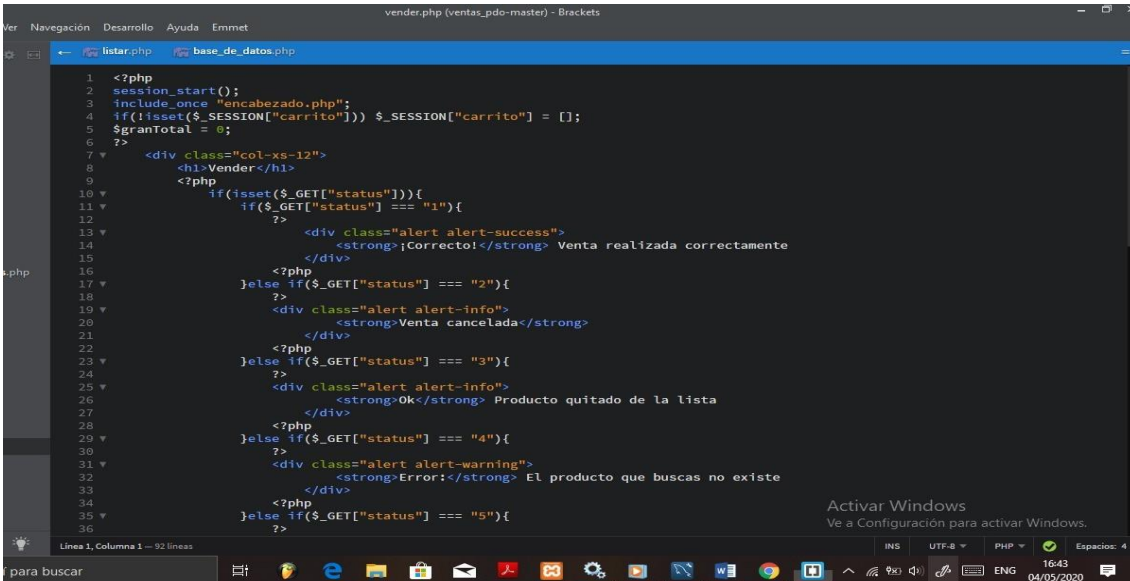
1 <?php
2 if(!isset($_GET["id"])) exit();
3 $id = $_GET["id"];
4 include_once "base_datos.php";
5 $sentencia = $base_datos->prepare("SELECT * FROM productos WHERE id = ?;");
6 $sentencia->execute([$id]);
7 $producto = $sentencia->fetch(PDO::FETCH_OBJ);
8 if($producto == FALSE){
9     echo "No existe algún producto con ese ID!";
10    exit();
11 }
12 }
13 }?>
14 <?php include_once "encabezado.php" ?>
15 <div class="col-xs-12">
16 <h1>Editar producto con el ID <?php echo $producto->id; ?></h1>
17 <form method="post" action="guardarDatosEditados.php">
18 <input type="hidden" name="id" value="<?php echo $producto->id; ?>">
19
20 <label for="codigo">Código de barras:</label>
21 <input value="<?php echo $producto->codigo ?>" class="form-control" name="codigo" required type="text" id="codigo"
22 placeholder="Escribe el código">
23
24 <label for="descripcion">Descripción:</label>
25 <textarea required id="descripcion" name="descripcion" cols="30" rows="5" class="form-control"><?php echo $producto->descripcion
26 ?></textarea>
27
28 <label for="precioVenta">Precio de venta:</label>
29 <input value="<?php echo $producto->precioVenta ?>" class="form-control" name="precioVenta" required type="number"
30 id="precioVenta" placeholder="Precio de venta">
31
32 <label for="precioCompra">Precio de compra:</label>
33 <input value="<?php echo $producto->precioCompra ?>" class="form-control" name="precioCompra" required type="number"
34 id="precioCompra" placeholder="Precio de compra">
35
36 <label for="existencia">Existencia:</label>
37 <input value="<?php echo $producto->existencia ?>" class="form-control" name="existencia" required type="number" id="existencia"
38 placeholder="Cantidad o existencia">
39
40 <br><br><input class="btn btn-info" type="submit" value="Guardar">
41 <a class="btn btn-warning" href="/listar.php">Cancelar</a>
42 </form>
43 </div>
44 <?php include_once "pie.php" ?>

```

Aquí se muestra otra plantilla que será de los productos que componen la venta. También muestra el total, que al inicio es 0. Y hasta abajo 2 botones que son para terminar de vender o para cancelar la venta. Igualmente hay unos ifs que muestran una alerta como “Producto inexistente” “en esta parte se esta trabando para que de resultado al final”



Comenzamos comprobando la existencia del producto, si no existe, regresamos a vender.php y le pasamos el status 5 que dirá que ya no hay existencias del producto.



```
1 <?php
2 session_start();
3 include_once "encabezado.php";
4 if(!isset($_SESSION["carrito"])) $_SESSION["carrito"] = [];
5 $granTotal = 0;
6 ?>
7 <div class="col-xs-12">
8 <h1>Vender</h1>
9 <?php
10 if(isset($_GET["status"])){
11     if($_GET["status"] === "1"){
12         ?>
13         <div class="alert alert-success">
14             <strong>¡Correcto!</strong> Venta realizada correctamente
15         </div>
16     <?php
17     }else if($_GET["status"] === "2"){
18         ?>
19         <div class="alert alert-info">
20             <strong>Venta cancelada</strong>
21         </div>
22     <?php
23     }else if($_GET["status"] === "3"){
24         ?>
25         <div class="alert alert-info">
26             <strong>Ok</strong> Producto quitado de la lista
27         </div>
28     <?php
29     }else if($_GET["status"] === "4"){
30         ?>
31         <div class="alert alert-warning">
32             <strong>Error:</strong> El producto que buscas no existe
33         </div>
34     <?php
35     }else if($_GET["status"] === "5"){
36         ?>
```

REFERENCIAS

<https://www.frikipandi.com/tecnologia/20170202/brackets-gran-editor-diseno-web/> (25/05/2020) (12:58)

<https://www.apachefriends.org/es/index.html> (25/05/2020) (12:58)

<https://www.php.net/manual/es/intro-what-is.php> (25/05/2020) (1:20)

<https://codigofacilito.com/articulos/que-es-html> (25/05/2020) (1:30)

<https://www.php.net/manual/es/intro.pdo.php> (25/05/2020) (1:40)