

Universidad Nacional Autónoma de México

Facultad de Estudios Superiores Cuautitlán



Seminario de Aplicaciones Web 2

Profesora:

Karina Simón Farfán

Grupo: 1901

Integrantes del equipo:

Hernández Cruz Sergio

Martínez Pérez Laura Estela

Manual de especificaciones técnicas

Aplicación Web:

"Tienda en línea: AnimaToys"

ÍNDICE

INTRODUCCIÓN.....	3
LENGUAJES DE PROGRAMACIÓN UTILIZADOS	4
SOFTWARE UTILIZADO.....	5
ARQUITECTURA	6
BASE DE DATOS.....	6
Modelo E-R.....	6
DICCIONARIO DE DATOS	7
Tabla cliente	7
Tabla Compra.....	7
Tabla Producto	8
Tabla Asocia.....	8
Tabla Administrador.....	9
DESCRIPCIÓN DE PROCESOS.....	10
Diagrama de Flujo	10
OPERACIONES PRINCIPALES.....	12
Registro de usuarios.....	12
Iniciar sesión como usuario	13
Agregar un producto al carrito de compras	14
Eliminar un producto del carrito de compras	14
Pagar un producto	15
Iniciar sesión como administrador.....	15
Agregar nuevo producto	16
Editar un producto	17
Eliminar un producto.....	17
Agregar disponibilidad un producto.....	18

INTRODUCCIÓN

Una tienda en línea o tienda virtual. Se refiere a una aplicación web que permite el comercio como la compra o venta entre uno o varios usuarios a través de Internet.

El presente manual tiene la finalidad de proporcionar al lector la información necesaria para conocer como ha sido creada la tienda en línea “AnimaToys” en términos técnicos.



Es necesario especificar cuáles fueron las herramientas que se utilizaron para la creación del sitio, así como explicar a detalle para qué fueron ocupadas estas herramientas específicamente, cabe aclarar que este manual no es para el aprendizaje de su utilización, es únicamente para proporcionar información acerca de cada una de las herramientas utilizadas para la elaboración del sitio web.

LENGUAJES DE PROGRAMACIÓN UTILIZADOS

LENGUAJE	DESCRIPCIÓN	UTILIZACIÓN
PHP (Hypertext PreProcessor)	Lenguaje de código abierto que puede ser incrustado en HTML.	Para que nuestro sitio web sea dinámico es necesario usar PHP, se ejecuta al lado del servidor y se incrusta dentro del código HTML.
HTML (HyperText Markup Language)	Lenguaje de programación que se utiliza para el desarrollo de páginas de Internet.	Para indicar como va ordenado el contenido de una página web mediante las marcas de hipertexto las cuales son las etiquetas.
Js (JavaScript)	Es un lenguaje de programación orientado a objetos, basado en prototipos, imperativo y dinámico.	Se utiliza del lado del cliente permitiéndonos crear efectos atractivos y dinámicos en la página web.
CSS (Cascading Style Sheets)	es el lenguaje utilizado para describir la presentación de documentos HTML o XML	Para organizar la presentación y aspecto de nuestra página web.
SQL (Structured Query Language)	lenguaje de dominio específico utilizado en programación, diseñado para administrar, y recuperar información de sistemas de gestión de bases de datos relacionales.	Para administrar información de dentro de una base de datos relacional, lo que nos permitirá almacenar información del sitio web.

SOFTWARE UTILIZADO

- 1) **XAMPP:** Es un servidor independiente de plataforma de código libre. Gracias a las herramientas que nos proporciona como Apache y Maria db lo utilizamos para poder crear nuestro sitio web y nuestra base de datos visualizando nuestra página web por medio del servidor remoto.
- 2) **Maria db:** Es un sistema gestor de bases de datos en la que se puede acceder a la información de la BD, en este caso accedemos a nuestra BD gracias a PHPMyAdmin, ya que es el acceso que XAMPP nos permite tener con Maria db.
- 3) **Apache:** El servidor HTTP Apache es un servidor web HTTP de código abierto, con el cuál pudimos crear y visualizar nuestro sitio web.
- 4) **Sublime text 3:** Es un editor de código que nos permite programar en diferentes lenguajes.
- 5) **Bootstrap:** Bootstrap es un framework que tiene como objetivo facilitar el diseño web. Permite crear de forma sencilla webs de diseño adaptable, es decir, que se ajusten a cualquier dispositivo y tamaño de pantalla, es decir nos permite ver nuestro sitio en modo responsive.
- 6) **Navegadores:** El sitio fue visualizado y durante su elaboración fue utilizado por los siguientes navegadores:
 - Mozilla Firefox
 - Opera
 - Google Chrome



ARQUITECTURA

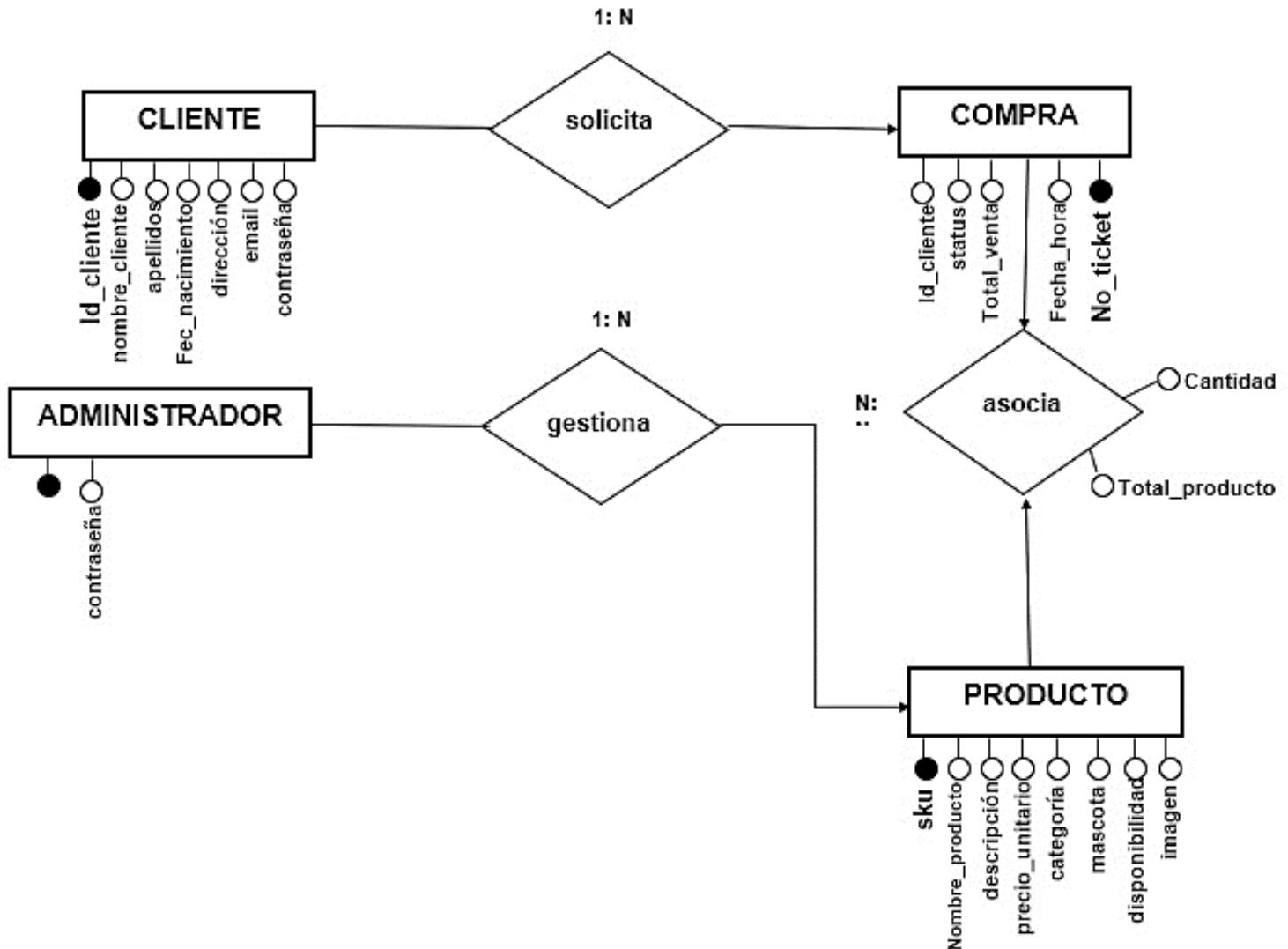
La arquitectura ocupada para elaborar esta tienda en línea es MVC

MVC: Es una propuesta de diseño de software utilizada para implementar sistemas donde se requiere el uso de interfaces de usuario.

Este diseño fue elegido con el objetivo de crear una aplicación de mayor calidad gracias a que nos proporciona facilidad de desarrollo y mantenimiento.

BASE DE DATOS

Modelo E-R.



DICCCIONARIO DE DATOS

Tabla cliente

NOMBRE	TIPO DE DATO	LONGITUD	TIPO DE DATO	DESCRIPCIÓN
ID	Int	11	PK	Identificador de cliente
Nombre	Varchar	150	-	Nombre del cliente
Apellidos	Varchar	150	-	Apellidos del cliente
Año_nac	Date		-	Año de nacimiento del cliente
Dirección	Varchar	100	-	Dirección completa del cliente
email	Varchar	70	-	Correo electrónico del cliente
contraseña	Varchar	20	-	Contraseña de usuario de AnimaToys del cliente

Tabla Compra

NOMBRE	TIPO DE DATO	LONGITUD	TIPO DE DATO	DESCRIPCIÓN
No_ticket	Varchar	6	PK	Identificador de una compra
Fecha_hora	Datetime		-	Fecha y hora en que se realizó la compra
Total_venta	Varchar	10	-	Cantidad total a pagar en la compra
Status	Varchar	20	-	Muestra si los productos comprados han sido pagados o no.

ID_cliente	int	11	FK	Número de identificador del cliente que ha hecho la compra
-------------------	-----	----	----	--

Tabla Producto

NOMBRE	TIPO DE DATO	LONGITUD	TIPO DE DATO	DESCRIPCIÓN
SKU	Varchar	6	PK	SKU perteneciente a un producto en específico
Nombre_producto	Varchar	80	-	Nombre del producto
Descripción	Varchar	200	-	Descripción completa del producto
Precio_unitario	Varchar	10	-	Precio del producto
Categoría	Varchar	25	-	Nombre de la categoría a la que pertenece ese producto
Mascota	Varchar	25	-	Muestra a qué especie de animal pertenece ese producto
disponibilidad	Int	10	-	Muestra cuántos productos están disponibles para su compra
imagen	Varchar	30	-	Muestra una imagen del producto
Nombre_usuario	varchar	18	FK	Nombre identificador de usuario de administrador

Tabla Asocia

NOMBRE	TIPO DE DATO	LONGITUD	TIPO DE DATO	DESCRIPCIÓN
No_ticket	Varchar	6	PK/FK	Identificador de una compra

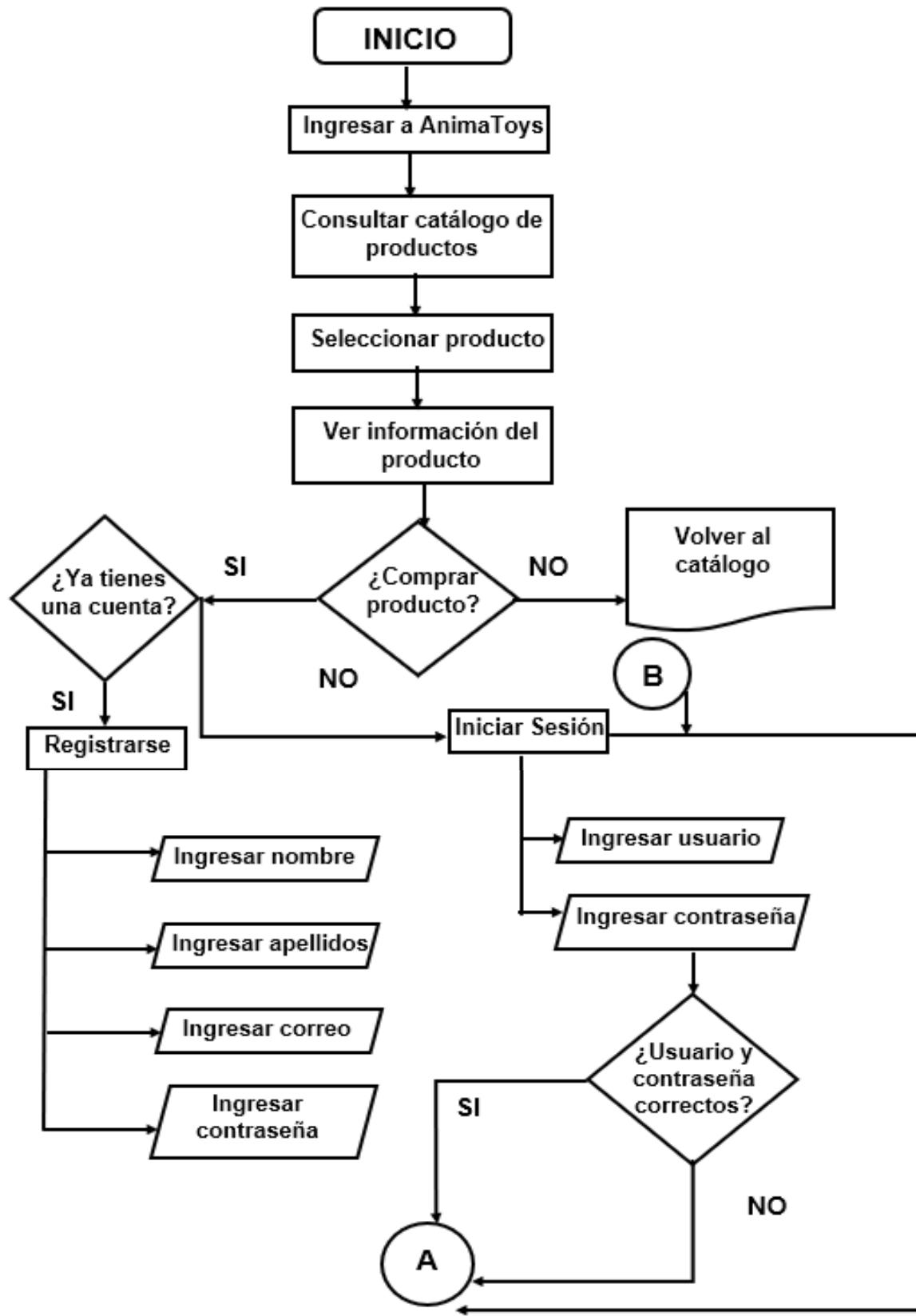
SKU	Varchar	6	PK/FK	Fecha en que se realizó la compra
Cantidad	int	11	-	Cantidad total a pagar en la compra
Total_producto	int	10	-	Número de identificador del cliente que ha hecho la compra

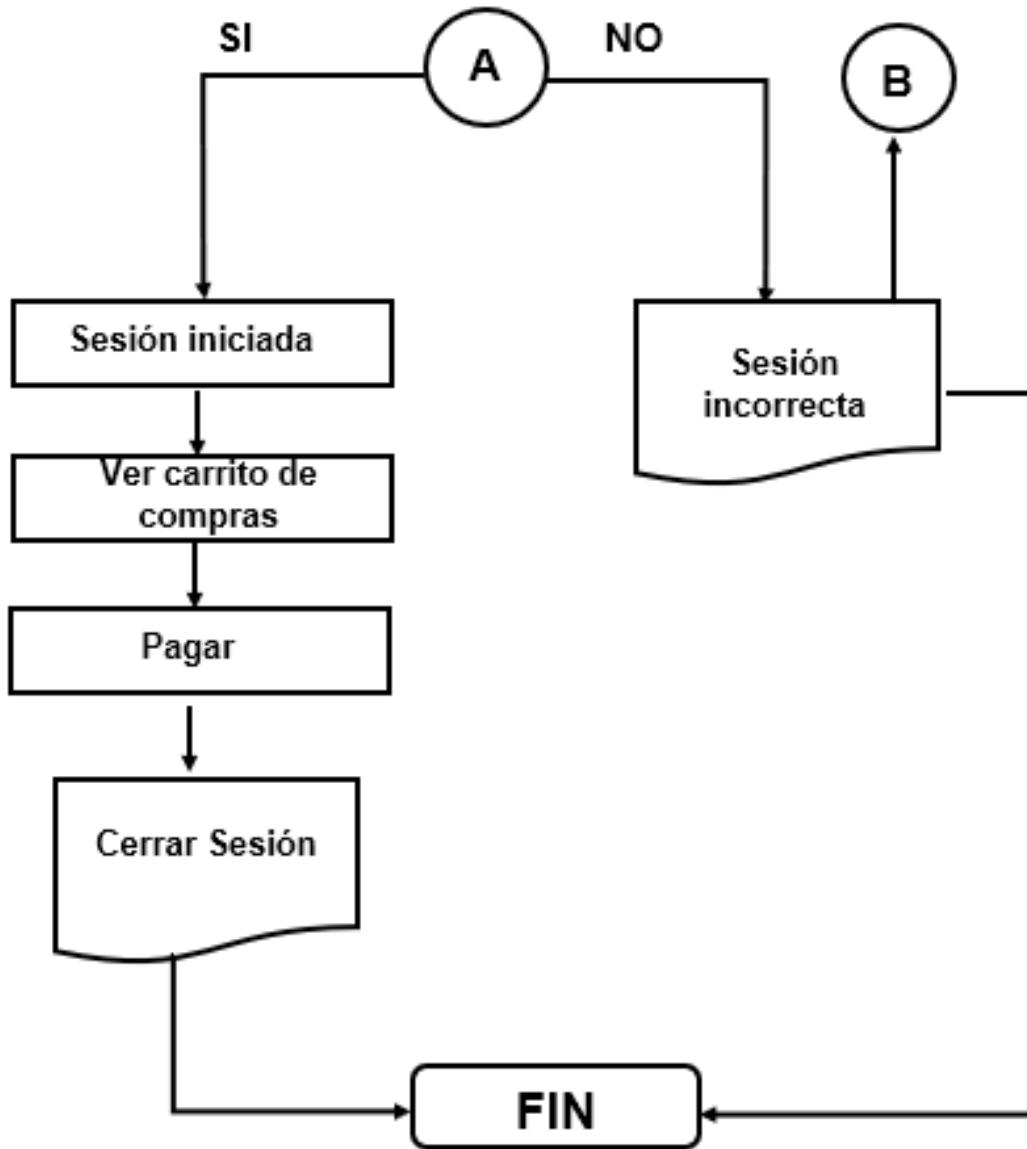
Tabla Administrador

NOMBRE	TIPO DE DATO	LONGITUD	TIPO DE DATO	DESCRIPCIÓN
Nombre_usuario	Varchar	18	PK	El nombre de usuario deberá ser único
Contraseña	Varchar	11	-	Contraseña de administrador

DESCRIPCIÓN DE PROCESOS

Diagrama de Flujo





OPERACIONES PRINCIPALES

Registro de usuarios

Se creó el documento llamado registrarse.php, el cual tiene como principal función recibir los datos que ingresará el usuario, tal como se muestra en la siguiente imagen:



```
registrarse.php
<?php
require_once 'conexion.php';
$conexion = conexion();

$nombre = $_POST['nombre'];
$apellidos = $_POST['apellido'];
$fec_nac = $_POST['nacimiento'];
$email = $_POST['correo'];
$contraseña = $_POST['pass'];

if ($nombre == '' || $apellidos == '' || $fec_nac == '' || $email == '' || $contraseña == '') {
    echo json_encode("Error");
}
else{
    $sql= "INSERT INTO cliente(nombre_cliente, apellidos, fec_nacimiento, email, contrasenia)
VALUES ('$nombre','$apellidos','$fec_nac','$email','$contraseña')";
$resultado = mysqli_query($conexion, $sql);

    if (!$resultado) {
        echo json_encode("Error");
    }
    else{
        echo json_encode("Correcto");
    }
}
?>
```

¡Registrate!

Nombre(s):

Apellidos:

Fecha de nacimiento:

Correo electronico:

Contraseña (8 caracteres):

Registrarse

Iniciar sesión como usuario

En caso de ya tener una cuenta existente, el usuario deberá dirigirse a la sección de Iniciar sesión, en donde ingresará su correo electrónico y contraseña, como se muestra a continuación:

Iniciar sesión

Correo electrónico:

ang_rmz@gmail.com

Contraseña:

.....

Iniciar sesión

La función que se encargará de darle función a esto es la siguiente:

```
iniciar_sesion.js
var iniciar_sesion = document.getElementById('iniciar_sesion');
var respuesta = document.getElementById('respuesta');

iniciar_sesion.addEventListener('submit', function(e){
    e.preventDefault();

    var datos = new FormData(iniciar_sesion);

    fetch('modelo/iniciar_sesion.php',{
        method: "POST",
        body: datos
    })

    .then(res => res.json())
    .then(data => {
        // console.log(data)
        if(data === 'Error'){
            respuesta.innerHTML = "<div class='alert alert-danger'>Error: Favor de verificar tus datos</div>";
        }
        else{
            window.location.href= "vista/php/inicio.php";
        }
    })
})
```

La función anterior detectará el correo y contraseña del usuario para permitirle el acceso a la cuenta, en dado caso de que la contraseña y/o el correo sean incorrectos aparecerá el mensaje "Error: favor de verificar tus datos".

Agregar un producto al carrito de compras

Para agregar un producto al carrito de compras, se ocupará la siguiente función:

```
function agregaprod(codigo, cantidad){  
    cadena= "producto="+codigo+"&cantidad="+cantidad;  
    $.ajax({  
        type: "POST",  
        url: "../../modelo/agregar_carrito.php",  
        data: cadena,  
        success: function(r){  
            if (r=1) {  
                alertify.success("El producto se ha agregado al carrito");  
            }else{  
                alertify.error("El producto ya ha sido agregado al carrito")  
            }  
        }  
    });  
}
```

En dónde se registrarán la cantidad de producto que el usuario requiere comprar, al momento de agregar un producto al carrito aparecerán los siguientes mensajes: mensaje de éxito: “El producto se ha agregado al carrito” y como error : “El producto ya ha sido agregado al carrito” dependiendo el caso.

Eliminar un producto del carrito de compras

Para eliminar un producto del carrito de compras será necesario usar la siguiente función:

```
function eliminar_datos(sku, mascota){  
    cadena= "sku="+sku;  
    $.ajax({  
        type:"POST",  
        url:"../../modelo/eliminar.php",  
        data: cadena,  
        success:function(r){  
            if (r=1) {  
                $("#tabla").load("cont_products.php?mascota="+mascota);  
                alertify.success("El producto se ha eliminado del carrito");  
            }  
            else{  
                alertify.error("Error con el servidor");  
            }  
        }  
    })  
}
```

En dónde por medio del sku del producto y el tipo de mascota, permitirá que este se elimine del carrito de compras del cliente.

Pagar un producto

Para pagar un producto se deberá ingresar el número de tarjeta del cliente, a la vez se generará un folio de compra, después se obtendrán estos con fecha y hora de pago, y finalmente aparecerá el mensaje “pagado”.

Iniciar sesión como administrador

Para iniciar sesión como administrador, es necesario ingresar primero a la cuenta, como se muestra a continuación:

Iniciar sesión

Usuario:

Contraseña:

Acceder

El administrador podrá acceder con la siguiente función:

```
var iniciar_sesion = document.getElementById('inicia_sesion');
var respuesta = document.getElementById('respuesta');

iniciar_sesion.addEventListener('submit', function(e){
    e.preventDefault();

    var datos = new FormData(iniciar_sesion);

    fetch '../../../../../modelo/ini_sesion.php',{
        method: "POST",
        body: datos
    })

    .then(res => res.json())
    .then(data => {
        console.log(data)
        if( data === 'Correcto'){
            window.location.href= '../../../../../index.php';
        }
        else{
            respuesta.innerHTML = "<div class='alert alert-danger'>Error: Favor de verificar tus datos</div>";
        }
    })
})

var registrarse = document.getElementById('registro');
var resp_registro = document.getElementById('resp_registro');

registrarse.addEventListener('submit', function(e){
    e.preventDefault();
```

```

var reg = new FormData(registrarse);
/*console.log(reg);
console.log(reg.get('nombre'))
console.log(reg.get('apellido'))*/
fetch('../modelo/registrar.php',{
    method: "POST",
    body: reg
})

.then(res => res.json())
.then (data=> {
    if (data == "correcto"){
        resp_registro.innerHTML =
            "<div class='alert alert-success'>Se ha registrado su correo <br> Ingrese con su cuenta</div>"
    }
    else{
        resp_registro.innerHTML=
            "<div class='alert alert-danger'>Error: ha ocurrido un problema <br>Verifique sus datos</div>"
    }
})

```

Esta función verificará si existe o no el correo y contraseña de usuario, de no ser así, mandará mensaje “verifique sus datos”

Agregar nuevo producto

Como administrador, se tienen ciertos privilegios, ya que es el encargado de administrar el sitio web, por lo que se agregaron funciones para agregar, modificar y eliminar productos.

Para agregar producto se requiere de la siguiente función:

```

if (isset($_POST['registrar'])) {
    if (strcmp($_REQUEST['cod'], "registro") == 0) {

        $producto = $_POST['producto'];
        $sku = $_POST['sku'];
        $desc = $_POST['desc'];
        $precio = $_POST['precio'];
        $disp = $_POST['disp'];
        $categoria = $_POST['categoria'];
        $mascota = $_POST['mascota'];

        $usuario = $_SESSION['id_admin'];

        if($_FILES['img']['error'] > 0){
            echo "<script language='javascript'>alert('Debe elegir una imagen'); window.history.back(); </script>";
        }else{
            $imagen = $_FILES['img']['name'];

            $sql = "INSERT INTO producto VALUES ('$sku', '$producto', '$desc', '$precio', '$categoria', '$mascota', '$disp',
                '$imagen', '$usuario')";

            if(saveImage($_FILES['img']) == 1){
                $result = mysqli_query($conexion, $sql);
                echo "<script language='javascript'>alert('El producto ha sido agregado exitosamente');
                    window.location.href='../vista/php/inicio.php';</script>";
            }else{
                /*echo "<script language='javascript'>alert('Ha ocurrido un error');
                    window.history.back();
                </script>"; */
            }
        }
    }
}

```

El cuál consiste en pasar los valores de las variables que se van a agregar con la ayuda de una instrucción en lenguaje SQL, en el momento en que se el administrador de clic en guardar aparecerá un mensaje de éxito “El producto ha sido agregado exitosamente”.

Editar un producto

Para editar/actualizar un producto, es necesario asignarles valores nuevos a las variables que ya existen, al momento de ser actualizado un producto, aparecerá un mensaje de éxito: “Actualizado con éxito” de lo contrario, mandará un error con el mensaje “Fallo el servidor”.

```
function actualizaDatos(){
    sku=$('#sku').val();
    nombre=$('#nombre').val();
    desc=$('#desc').val();
    precio=$('#precio').val();
    categoria=$('#categoria').val();
    mascota=$('#mascota').val();
    cadena="sku=" + sku + "&nombre=" + nombre + "&desc=" + desc + "&precio=" + precio + "&categoria=" + categoria + "&mascota=" + mascota;

    $.ajax({
        type: "POST",
        url:"../../../modelo/actualiza_datos.php",
        data: cadena,
        success:function(r){
            if(r=1){
                $('#tabla').load("cont_products.php?mascota="+mascota);
                alertify.success("Actualizada con éxito");
            }
            else{
                alertify.error("Fallo el servidor");
            }
        }
    })
}
```

Eliminar un producto

Para eliminar un producto de la tienda, usaremos la siguiente función:

```
function pregunta(sku, mascota){
    alertify.confirm('Eliminar datos', '¿Esta seguro de eliminar producto?', 
    function(){
        eliminar_datos(sku, mascota)
    },function(){
        alertify.error('Cancelado')
    });
}
```

En dónde se tomará el sku del producto y la mascota a la que pertenece ese producto, el cuál generará un query SQL ejecutando un DELETE.

```
<?php
require_once 'conexion.php';
$conexion =conexion();

$sku = $_POST['sku'];

$sql = "DELETE FROM producto WHERE sku= '$sku'";
$resultado = mysqli_query($conexion, $sql);
?>
```

Agregar disponibilidad un producto

El administrador podrá agregar disponibilidad de productos en la tienda con la siguiente función:

```
function sumar_disponibilidad(){
    sku=$('#skua').val();
    cantidad=$('#cantidad').val();
    cadena= "sku="+sku+"&cantidad="+cantidad;
    mascota= $('#mascotaa').val();

    $.ajax({
        type: "POST",
        url:"../../modelo/agregar_disp.php",
        data: cadena,
        success:function(r){
            if(r==1){
                $("#tabla").load("cont_products.php?mascota="+mascota);
                alertify.success("Actualizada con éxito");
            }
            else{
                alertify.error("Fallo el servidor");
            }
        }
    })
}
```

El cuál sumará la cantidad a la cantidad que ya se encuentra guardada en la BD en la variable "cantidad" dependiendo del sku del producto.