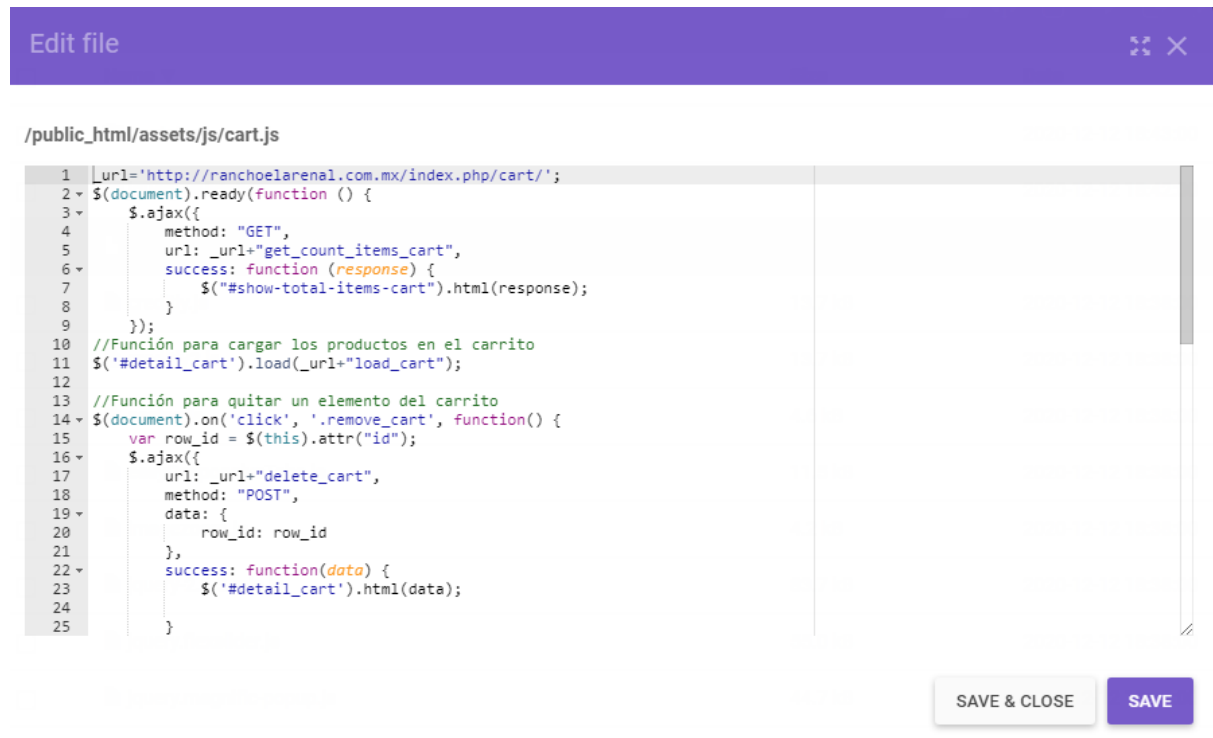


El día de hoy se continuó trabajando en la plataforma de ventas. A continuación se muestra una lista sobre las acciones realizadas:

- Se configuraron las urls para el llamado de los servicios en la plataforma de ventas que se subió al servidor.



```
/public_html/assets/js/cart.js

1 |_url='http://ranchoelarenal.com.mx/index.php/cart/';
2 |$(document).ready(function () {
3 |    $.ajax({
4 |        method: "GET",
5 |        url: _url+"get_count_items_cart",
6 |        success: function (response) {
7 |            $("#show-total-items-cart").html(response);
8 |        }
9 |    });
10 |    //Función para cargar los productos en el carrito
11 |    $('#detail_cart').load(_url+"load_cart");
12 |
13 |    //Función para quitar un elemento del carrito
14 |    $(document).on('click', '.remove_cart', function() {
15 |        var row_id = $(this).attr("id");
16 |        $.ajax({
17 |            url: _url+"delete_cart",
18 |            method: "POST",
19 |            data: {
20 |                row_id: row_id
21 |            },
22 |            success: function(data) {
23 |                $('#detail_cart').html(data);
24 |            }
25 |        })
    }
```

- Se configuró la función de registro de clientes para enviar correo electrónico para activación de cuenta en el servidor.

```
80 $config=array(  
81     'protocol'=>"smtp",  
82     'smtp_host'=> "ssl://smtp.gmail.com",  
83     'smtp_port'=>465,  
84     'smtp_user'=>"pruebasqrosystem@gmail.com",  
85     'smtp_pass'=>"qrosystem123",  
86     // 'newline'=>"\r\n",  
87     'mailtype'=>"html",  
88     'charset'=>"utf-8",  
89     //'crlf'=>"\r\n",  
90     'wordwrap'=>TRUE,  
91     //'priority'=>3  
92 );  
93  
94 $this->load->library('email',$config);  
95 $this->email->set_newline("\r\n");  
96 $this->email->set_header('Carrito de compras','Rancho el Arenal');  
97 $this->email->from("pruebasqrosystem@gmail.com", "Rancho el Arenal");  
98 $this->email->to($email);  
99 $this->email->subject('Verifica tu correo electrónico en Rancho el Arenal');  
100 $data= array('id'=>$dataResponse['key'],'token' =>$token );  
101 $body = $this->load->view('send_mail',$data,TRUE);  
102 $this->email->message($body);
```

- Se creó una función para evitar que el cliente seleccione fechas atrasadas al momento de realizar el pedido.

```
var today = new Date();  
var dd = today.getDate()+1;  
var mm = today.getMonth()+1; //January is 0!  
var yyyy = today.getFullYear();  
if(dd<10){  
    dd='0'+dd  
}  
if(mm<10){  
    mm='0'+mm  
}  
  
today = yyyy+'-'+mm+'-'+dd;  
document.getElementById("order_date").setAttribute("min", today);
```

Fragmento de código de función para desactivar las fechas ya pasadas.

Reporte del día
Lunes 14 de diciembre de 2020

Fecha de pedido*

15/12/2020

diciembre de 2020

do.	lu.	ma.	mi.	ju.	vi.	sá.
29	30	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	1	2
3	4	5	6	7	8	9

Hoy

Resultado obtenido.

- Se agregó la condición para verificar si hay productos registrados y en caso de que no haya se muestre el mensaje de que no hay productos en venta.

Nuestros Productos

⚠ No hay productos en venta

Buscar..

Nombre del producto... >

Precio

Resultado obtenido.

- Se modificó el tamaño de las tarjetas en donde se muestran los productos.



Resultado obtenido.

- Se agregaron nuevos banners con las fotografías tomadas.



Resultado visual de banner agregado.

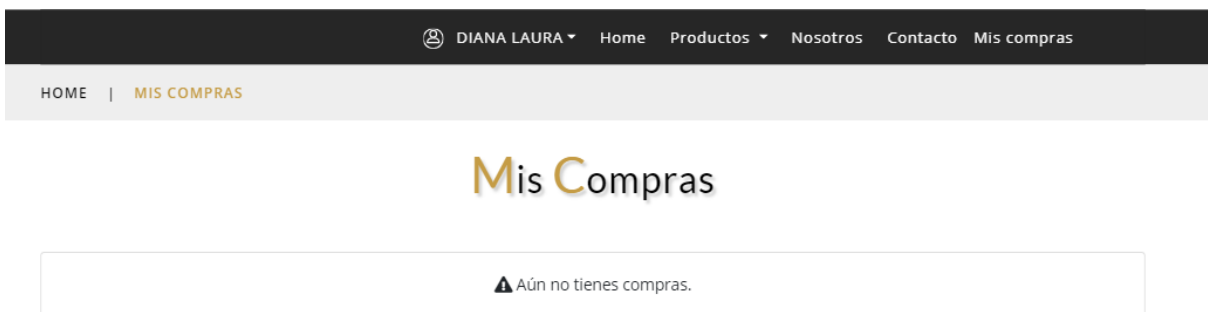


Resultado visual de banner agregado.



Resultado visual de banner agregado.

- Se agregó condición para mostrar la leyenda de cuando aún no tiene compras o pedidos.

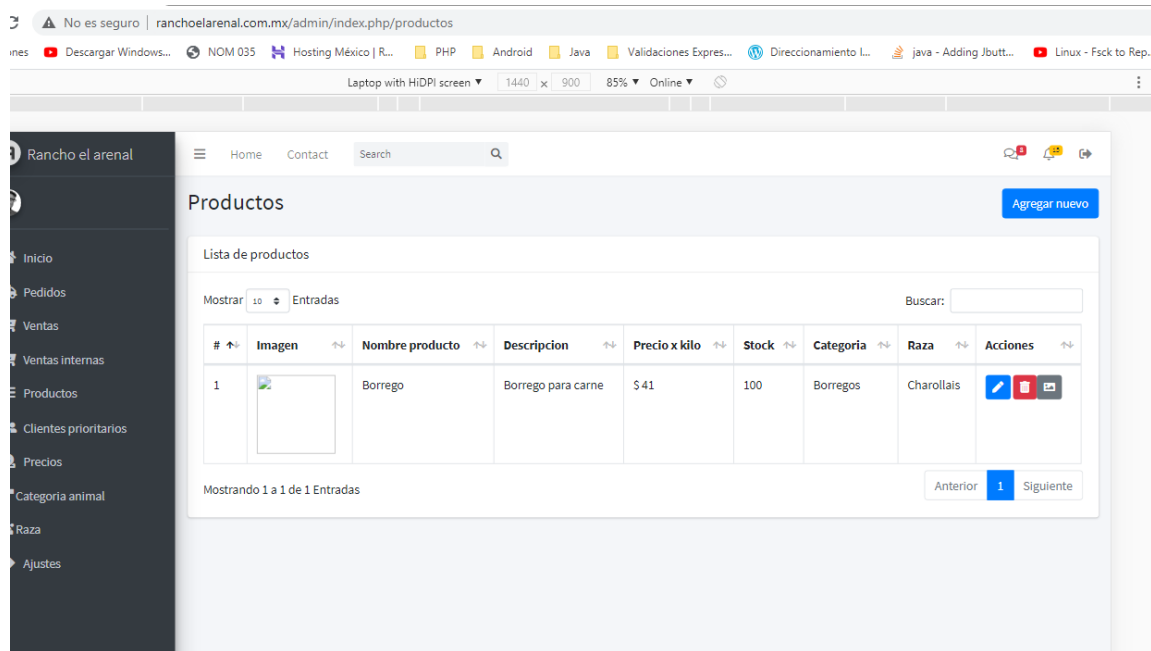


Resultado visual obtenido.

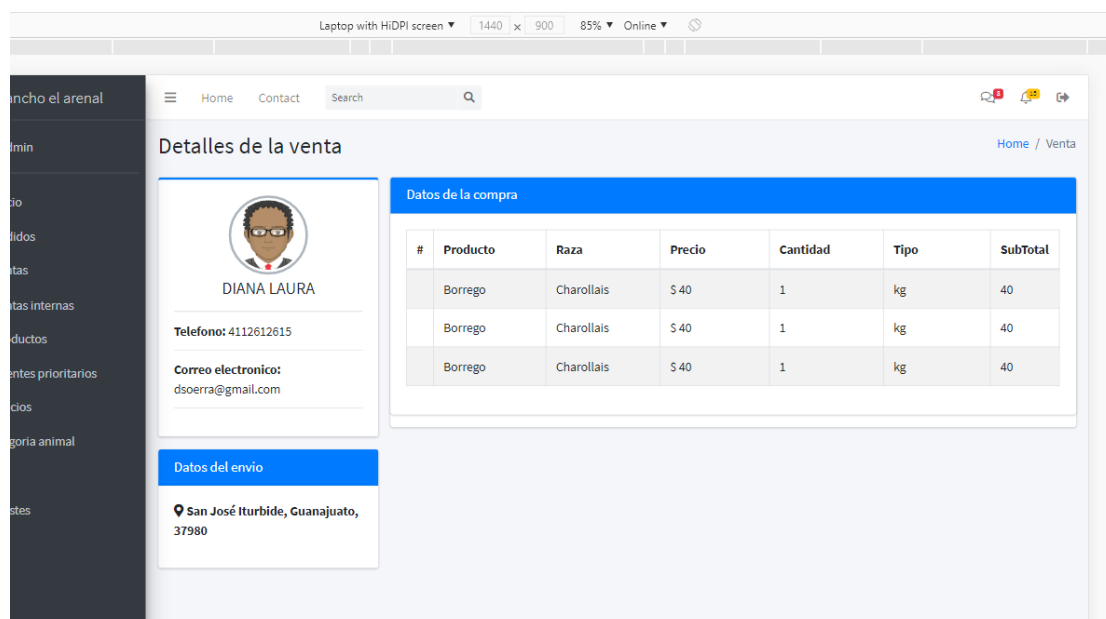
Reporte del día

Lunes 14 de diciembre de 2020

El día de hoy se configuraron las url de el proyecto debido a que ya se implemento en el servidor y es necesario actualizarlas ya que anteriormente se trabajaba localmente y son necesarias para el llamado de las funciones.



Una vez configuradas las url se realizaron pruebas en la parte del admin para verificar que los procesos se han realizado correctamente.



Reporte del día

Lunes 14 de diciembre de 2020

Una vez que se hicieron unas pruebas se vaciaron las tablas de la base de datos para poder hacer pruebas reales de las ventas internas.



Se ha decidido que la categoría sea la que contenga una fotografía del animal o producto que se va a mostrar, esto con la finalidad de no saturar el servidor con imágenes repetidas, y solo se mostrará la imagen de la categoría.













```
//agregar categorias
function categorias_post()
{
    $data = $this->post();
    if (count($data) == 0 || count($data) > 15) {
        $response = array(
            "status" => "error",
            "status_code" => 409,
            "message" => count($data) == 0 ? 'No se recibieron datos' : 'Demasiados datos fueron enviados',
            "data" => count($data),
            "validations" => array(
                "nombre" => "Requerido",
                "imagen_cat" => "Requerido"
            )
        );
    } else {
        $this->form_validation->set_data($data);
        $this->form_validation->set_rules('nombre', 'nombre', 'required');
        $this->form_validation->set_rules('imagen_cat', 'imagen_cat', 'required');









        if ($this->form_validation->run() == FALSE) {
            $response = array(
                "status" => "error",
                "status_code" => 409,
                "message" => "validations error",
                "data" => null,
                "validations" => $this->form_validation->error_array()
            );
        } else {
            $data = array(
                "nombre_categoria" => $this->post('nombre'),
                "imagen_cat" => $this->post('imagen_cat')
            );
            $dataResponse = $this->DAO->saveOrUpdateItem('categorias', $data, null, true);
        }
    }
}
```






Se modificó el método para poder editar la imagen de la categoría



```
function categorias_put()
{
    $id = $this->get('id');
    if ($id) {
        $dataExists = $this->DAO->selectEntity('categorias', array('id_categoria' => $id), TRUE);
        if ($dataExists) {
            if (count($this->put()) > 4) {
                $response = array(
                    "status" => "error",
                    "status_code" => 409,
                    "message" => "Demasiados datos fueron enviados",
                    "validations" => array(
                        "nombre" => "Requerido",
                        "imagen_cat" => "Requerido"
                    ),
                    "data" => null
                );
            } else if (count($this->put()) == 0) {
                $response = array(
                    "status" => "error",
                    "status_code" => 409,
                    "message" => "Ningun dato fue enviado",
                    "validations" => array(
                        "nombre" => "Requerido"
                    ),
                    "data" => null
                );
            } else {
                $this->form_validation->set_data($this->put());
                $this->form_validation->set_rules('nombre', 'nombre', 'required');
                $this->form_validation->set_rules('imagen_cat', 'imagen_cat', 'required');
                if ($this->form_validation->run() == FALSE) {
                    $response = array(
                        "status" => "error",
                        "status_code" => 400
                    );
                }
            }
        }
    }
}
```

Se actualizo la base de datos para poder almacenar el nombre de la imagen ya que la imagen se guarda en el servidor.



#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
<input type="checkbox"/>	1	id_categoria	int(11)		No	Ninguna		AUTO_INCREMENT	  Más
<input type="checkbox"/>	2	nombre_categoria	varchar(100)	latin1_swedish_ci	No	Ninguna			  Más
<input checked="" type="checkbox"/>	3	imagen	text	latin1_swedish_ci	No	Ninguna			  Más
<input type="checkbox"/>	4	categoria_create	timestamp		No	current_timestamp()			  Más
<input type="checkbox"/>	5	categoria_edit	timestamp		No	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()	  Más
<input type="checkbox"/>	6	status	int(11)		No	1	1 activo, 0 inactivo		  Más

 Para los elementos que están marcados:       

 1 columna(s) después de status 

Índices

Acción	Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
 	PRIMARY	BTREE	Sí	No	id_categoria	2	A	No	

Se actualizo el formulario para el registro de las categorias, se agrego el campo para seleccionar la fotografia o imagen.

Agregar categoria

(*) Campos requeridos.

Nombre*

Fotografia

Ningún archivo seleccionado



Se actualizo la vista de la tabla principal de categorías donde se puede observar la imagen correspondiente a la categoría, esta imagen será la que se muestre para todos los productos que se registren bajo esta categoría.

Categorías

Lista de categorias

Mostrar 10 Entradas

Buscar:

#	Nombre	Imagen	Acciones
1	Ovinos		<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
2	Pescado		<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>

Mostrando 1 a 2 de 2 Entradas

Anterior

1

Siguiente

Reporte del día

Lunes 14 de diciembre de 2020

En el formulario de agregar cliente prioritario se ha agregado el campo para poder ingresar el RFC del cliente.

Agregar Nuevo cliente

CURP

VESGDGHH323MS32

R.F.C

UPU800825569

Nombre(s) *

John

Apellido Paterno *

Smith

Apellido Materno *

Ahn

Teléfono *

12345678901

Correo electrónico *

example@gmail.com

Contraseña *

Repite Contraseña *

INE O PASAPORTE

Seleccionar archivo

Ningún archivo seleccionado

Guardar