|

SISTEMA

**Índice**

**Contenido**

[**INTRODUCCIÓN**](https://docs.google.com/document/d/16aJ9OI5VTBFLOyg0dxn0-tGmWnSzYqju4YgNNRvyJaY/edit#heading=h.vzeeeznv1vkj)[**2**](https://docs.google.com/document/d/16aJ9OI5VTBFLOyg0dxn0-tGmWnSzYqju4YgNNRvyJaY/edit#heading=h.vzeeeznv1vkj)

[**OBJETIVOS**](https://docs.google.com/document/d/16aJ9OI5VTBFLOyg0dxn0-tGmWnSzYqju4YgNNRvyJaY/edit#heading=h.1fob9te)[**2**](https://docs.google.com/document/d/16aJ9OI5VTBFLOyg0dxn0-tGmWnSzYqju4YgNNRvyJaY/edit#heading=h.1fob9te)

[**DESARROLLO**](https://docs.google.com/document/d/16aJ9OI5VTBFLOyg0dxn0-tGmWnSzYqju4YgNNRvyJaY/edit#heading=h.hhemfxz2hvww)[**2**](https://docs.google.com/document/d/16aJ9OI5VTBFLOyg0dxn0-tGmWnSzYqju4YgNNRvyJaY/edit#heading=h.hhemfxz2hvww)

# INTRODUCCIÓN

La empresa Burgers SRL desea implementar un sitio web para recibir pedidos en modalidad takeaway. Una persona realiza la compra desde la página web, eligiendo los productos, seleccionando la sucursal, y abonando en efectivo en el local al retirar el pedido u online a través de una pasarela de pago (mercadopago).

Desde un panel administrativo los vendedores podrán visualizar los pedidos entrantes con el estado “Pendiente” y modificar los mismos con los estados “En preparación”, “Entregado” y “Cancelado”

# REQUISITOS

**Requisitos sitio web institucional**

*Enumerar y detallar los distintos requisitos sobre lo que debe realizar el sitio institucional.*

En home del sitio debe mostrar Banner de hamburguesas, domicilio de todas las sucursales desde la BBDD con número de teléfono, dirección y botón con link a google maps (al menos 4 sucursales).

El sitio debe tener las siguientes opciones de menú: Inicio, Takeaway, Nosotros, Contacto, Mi cuenta.

La página de Takeaway debe mostrar un listado de todos los productos, cada uno con su título, descripción y precio. Todo debe ser traído de la BBDD. También debe tener cada producto un input para elegir la cantidad y botón para agregar al carrito. Los productos agregados al carrito deben ser guardados también en la BBDD.

La página de Nosotros contendrá descripción de la empresa y un banner con un título. Además, contará con un subtítulo con un slider de bootstrap con frases rotativas. Luego una sección de “Trabajá con nosotros” con un formulario que solicite nombre, apellido, teléfono y un input para adjuntar el cv y el botón de enviar donde llevará  a un enlace con el mensaje “Gracias, pronto estarás recibiendo noticias nuestras”. Este contacto deberá registrarse en la base de datos como una postulación.

La página de Contacto debe tener  Formulario con nombre y apellido, celular, mail, localidad, y un área de texto donde esté la consulta o mensaje y botón enviar de tipo “submit”

La página de mi cuenta mostrará los datos del usuario tales como nombre, apellido, celular, correo y una tabla con los pedidos activos.

**Requisitos para el panel administrativo**

*Enumerar y detallar los distintos requisitos sobre lo que debe realizar el panel administrativo.*

El sistema debe permitir realizar el alta, baja, consulta y modificación de Productos, con los datos: nombre, cantidad, precio, imagen, categoría.

El sistema debe permitir realizar el alta, baja, consulta y modificación de Clientes, con los datos: nombre, apellido, mail, dni, teléfono, clave.

El sistema debe permitir realizar el alta, baja, consulta y modificación de Pedidos, con los datos: fecha, descripción, cantidad del pedido, nombre de producto, sucursal, cliente, total.

El sistema debe permitir realizar el alta, baja, consulta y modificación de Postulaciones, con los datos: nombre, apellido, teléfono, mail y curriculum.

El sistema debe permitir realizar el alta, baja, consulta y modificación de Sucursales, con los datos: dirección, teléfono, link mapa.

**Páginas de la competencia**

|  |  |
| --- | --- |
| **URL** | **Alumno** |
| https://cervezaeliotness.com.ar/ |  |
| https://www.mostazaweb.com.ar/ |  |
| https://www.mcdonalds.com.ar/ |  |

**Redes sociales**

No tiene.

**Plantilla para la web (bootstrap)**

*Buscar en Internet plantillas compatibles con bootstrap, se deja estos recursos de ejemplo:*

[*https://themewagon.com/theme-framework/bootstrap-5/*](https://themewagon.com/theme-framework/bootstrap-5/)

[*https://bootstrapmade.com/*](https://bootstrapmade.com/)

|  |  |
| --- | --- |
| **URL** | **Alumno** |
| <https://es.wix.com/website-template/view/html/1806?originUrl=https%3A%2F%2Fes.wix.com%2Fwebsite%2Ftemplates%2Fhtml%2Frestaurants-food%2Frestaurant&tpClick=view_button&esi=5a78a099-2ef5-41a7-9b34-298a4e3206c2>  https://themewagon.com/themes/free-bootstrap-4-html5-restaurant-website-template-feane/ |  |
| <https://plantillashtmlgratis.com/todas-las-plantillas/plantilla/plantilla-html-gratis-spicyo/>  --------------------------------------------------------------  [*https://elements.envato.com/es/soup-restaurant-with-online-ordering-system-DEVQJ6*](https://elements.envato.com/es/soup-restaurant-with-online-ordering-system-DEVQJ6)  --------------------------------------------------------------https://themewagon.com/themes/free-bootstrap-4-html5-restaurant-website-template-feane/ |  |

# 

# DESARROLLO

**Tablas del sistema**

**Base de datos**

|  |  |
| --- | --- |
| Alumnos que cursan en vivo (en la nube) | Alumnos libres (en su localhost) |
| Host: 167.114.86.211  Usuario: burgers\_062021  Clave: BURG.4579 (SOLO COPIAR, NO CORTAR)  Puerto: 3310  Nombre BBDD: burgers\_062021 | Host: 127.0.0.1  Usuario: root  Clave:  Puerto: 3306  Nombre BBDD: sistema |

**Repositorio**

actualizar .gitignore y vaciar logs

git config --global user.email "nelson.tarche@mail.com"

git config --global user.name "ntarche"

git clone https://github.com/depcsuite/burgers.git

**Panel de administración**

[http://127.0.0.1:8000/admin/logi](http://127.0.0.1:8000/admin/login)

Usuario: admin

Clave: 1234

**Archivo de configuración .env**

|  |  |
| --- | --- |
|  | APP\_NAME="Burgers SRL"  APP\_ENV=local  APP\_KEY=base64:xggbsM2IZ+yTAwTfsZT5iUKg033uMLT+aNCB1MVEVss  APP\_DEBUG=true  APP\_LOG\_LEVEL=debug  APP\_URL=http://127.0.0.1:8000  APP\_PATH="/Users/nelson/Desktop/Proyectos FS/broker-seguros"  DB\_CONNECTION=mysql  DB\_HOST=167.114.86.211  DB\_PORT=3310  DB\_DATABASE=burgers\_042021  DB\_USERNAME=burgers\_042021  DB\_PASSWORD=BR8K.8496  CACHE\_EXPIRE = -1  LOG\_CHANNEL=none  BROADCAST\_DRIVER=log  CACHE\_DRIVER=file  SESSION\_DRIVER=file  SESSION\_LIFETIME=120  QUEUE\_DRIVER=sync  REDIS\_HOST=127.0.0.1  REDIS\_PASSWORD=null  REDIS\_PORT=6379  MAIL\_DRIVER=smtp  MAIL\_HOST=null  MAIL\_PORT=25  MAIL\_USERNAME=null  MAIL\_PASSWORD=null  MAIL\_ENCRYPTION=tls  MAIL\_FROM\_NAME=null  MAIL\_FROM\_ADDRESS=null  MAIL\_NOREPLY=null  PUSHER\_APP\_ID=  PUSHER\_APP\_KEY=  PUSHER\_APP\_SECRET= |

**Menú administrador**

|  |  |  |
| --- | --- | --- |
| **Menú padre**  **(Con icono)** | **Submenú** | **URL** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Menú sitio institucional**

|  |  |  |
| --- | --- | --- |
| **Menú padre**  **(Con icono)** | **Submenú** | **URL** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

<form id="form1" method="POST" enctype="multipart/form-data">

public function guardar(Request $request)

{

try {

//Define la entidad servicio

$titulo = "Guardar producto";

$entidad = new Producto();

$entidad->cargarDesdeRequest($request);

if ($\_FILES["archivo"]["error"] === UPLOAD\_ERR\_OK) {//Se adjunta imagen

$nombre = date("Ymdhmsi") . ".jpg";

$archivo = $\_FILES["archivo"]["tmp\_name"];

move\_uploaded\_file($archivo, env('APP\_PATH') . "/public/files/$nombre"); //guardaelarchivo

$entidad->imagen = $nombre;

}

//validaciones

if ($entidad->nombre == "") {

$msg["ESTADO"] = MSG\_ERROR;

$msg["MSG"] = "Complete todos los datos";

} else {

if ($\_POST["id"] > 0) {

$productAnt = new Producto();

$productAnt->obtenerPorId($entidad->idproducto);

if($\_FILES["archivo"]["error"] === UPLOAD\_ERR\_OK){

//Eliminar imagen anterior

@unlink(env('APP\_PATH') . "/public/files/$productAnt->imagen");

} else {

$entidad->imagen = $productAnt->imagen;

}

//Es actualizacion

$entidad->guardar();

$msg["ESTADO"] = MSG\_SUCCESS;

$msg["MSG"] = OKINSERT;

} else {

//Es nuevo

$entidad->insertar();

$msg["ESTADO"] = MSG\_SUCCESS;

$msg["MSG"] = OKINSERT;

}

$\_POST["id"] = $entidad->idproducto;

return view('producto.producto-listar', compact('titulo', 'msg'));

}

} catch (Exception $e) {

$msg["ESTADO"] = MSG\_ERROR;

$msg["MSG"] = ERRORINSERT;

}

$id = $entidad->idproducto;

$producto = new Producto();

$producto->obtenerPorId($id);

return view('producto.producto-nuevo', compact('msg', 'producto', 'titulo')) . '?id=' . $producto->idproducto;

}

**Agregar en formulario-cliente.php**

if(isset($\_GET["do"]) && $\_GET["do"] == "cargarGrilla"){

$idCliente = $\_GET['idCliente'];

$request = $\_REQUEST;

$entidad = new Domicilio();

$aDomicilio = $entidad->obtenerFiltrado($idCliente);

$data = array();

$inicio = $request['start'];

$registros\_por\_pagina = $request['length'];

if (count($aDomicilio) > 0)

$cont=0;

for ($i=$inicio; $i < count($aDomicilio) && $cont < $registros\_por\_pagina; $i++) {

$row = array();

$row[] = $aDomicilio[$i]->tipo;

$row[] = $aDomicilio[$i]->provincia;

$row[] = $aDomicilio[$i]->localidad;

$row[] = $aDomicilio[$i]->domicilio;

$cont++;

$data[] = $row;

}

$json\_data = array(

"draw" => intval($request['draw']),

"recordsTotal" => count($aDomicilio), //cantidad total de registros sin paginar

"recordsFiltered" => count($aDomicilio),//cantidad total de registros en la paginacion

"data" => $data

);

echo json\_encode($json\_data);

exit;

}

**Agregar en entidad domicilio**

public function obtenerFiltrado($idCliente){

$mysqli = new mysqli(Config::BBDD\_HOST, Config::BBDD\_USUARIO, Config::BBDD\_CLAVE, Config::BBDD\_NOMBRE);

$request = $\_REQUEST;

$columns = array(

0 => 'B.nombre',

1 => 'D.nombre',

2 => 'C.nombre',

3 => 'A.domicilio'

);

$sql = "SELECT

A.iddomicilio,

A.fk\_tipo,

B.nombre AS tipo,

A.fk\_idlocalidad,

C.nombre AS localidad,

D.idprovincia,

D.nombre AS provincia,

A.domicilio

FROM domicilios A

INNER JOIN tipo\_domicilios B ON A.fk\_tipo = B.idtipo

INNER JOIN localidades C ON C.idlocalidad = A.fk\_idlocalidad

INNER JOIN provincias D ON D.idprovincia = C.fk\_idprovincia

WHERE 1=1 AND A.fk\_idcliente = $idCliente

";

//Realiza el filtrado

if (!empty($request['search']['value'])) {

$sql.=" AND ( B.nombre LIKE '%" . $request['search']['value'] . "%' ";

$sql.=" OR D.nombre LIKE '%" . $request['search']['value'] . "%' ";

$sql.=" OR C.nombre LIKE '%" . $request['search']['value'] . "%' ";

$sql.=" OR A.domicilio LIKE '%" . $request['search']['value'] . "%' )";

}

$sql.=" ORDER BY " . $columns[$request['order'][0]['column']] . " " . $request['order'][0]['dir'];

$resultado = $mysqli->query($sql);

$lstRetorno = array();

while ($fila = $resultado->fetch\_assoc()) {

$domicilio = new Domicilio();

$domicilio->iddomicilio = $fila["iddomicilio"];

$domicilio->fk\_tipo = $fila["fk\_tipo"];

$domicilio->tipo = $fila["tipo"];

$domicilio->fk\_idlocalidad = $fila["fk\_idlocalidad"];

$domicilio->localidad = $fila["localidad"];

$domicilio->idprovincia = $fila["idprovincia"];

$domicilio->provincia = $fila["provincia"];

$domicilio->domicilio = $fila["domicilio"];

$lstRetorno[] = $domicilio;

}

return $lstRetorno;

}

Nicolas: Página gracias postulacion

Alejandro: fontawesome, En Mi cuenta mostrar los datos del cliente logueado y que pueda cambiarlos

Maximo: Eliminar del carrito

Matías: Corregir importe del pedido en el admin, mostrar productos del pedido detalle, y vaciar carrito

Nicolás: Promociones

Lucas: Sponsor