

---

# ESPECIFICACIÓN DE COMPONENTES

para

*Galería de productos web*  
*Calzados Lara Paillaco®*

Versión 1.1

**Autores:**

Eduardo Hopperdietzel Ralph  
Sebastián Lara Barría

**Supervisor:**

Cristian Olivares Rodríguez

**Universidad Austral de Chile**

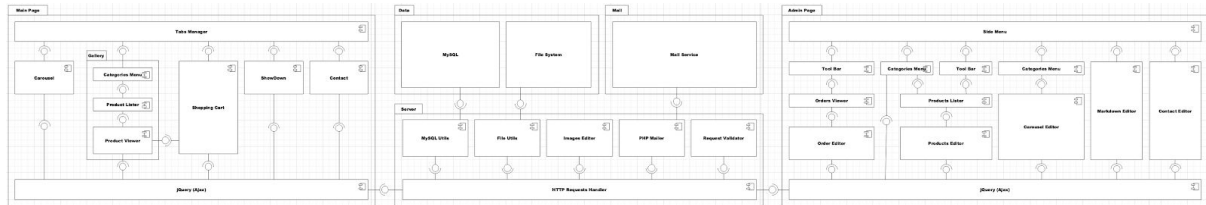
27 de junio de 2020

# Index

<b>1 Diagrama de Componentes</b>	<b>2</b>
1.1 Diagrama completo	2
1.2 Main Page	3
1.3 Server-Data-Mail	3
1.4 Admin Page	4
<b>2 Descripción de Componentes</b>	<b>4</b>
2.1 Tabs Manager	4
2.2 Carousel	5
2.3 Categories Menu	5
2.4 Product Lister	6
2.5 Product Viewer	7
2.6 ShowDown	8
2.7 Contacto	9
2.8 jQuery ( Ajax )	10
2.9 Side Bar	10
2.10 MySQL Utils	11
2.11 Validator	12
2.12 PHP Mailer	13
2.13 File Utils	14
2.14 Images Editor	14
2.15 HTTP Request Handler	14
2.16 Tool Bar	14
2.17 Orders Viewer	14
2.18 Orders Editor	14
2.19 Products Editor	14
2.20 Carousel Editor	14
2.21 MarkDown Editor	15
2.22 Contact Editor	15

# 1 Diagrama de Componentes

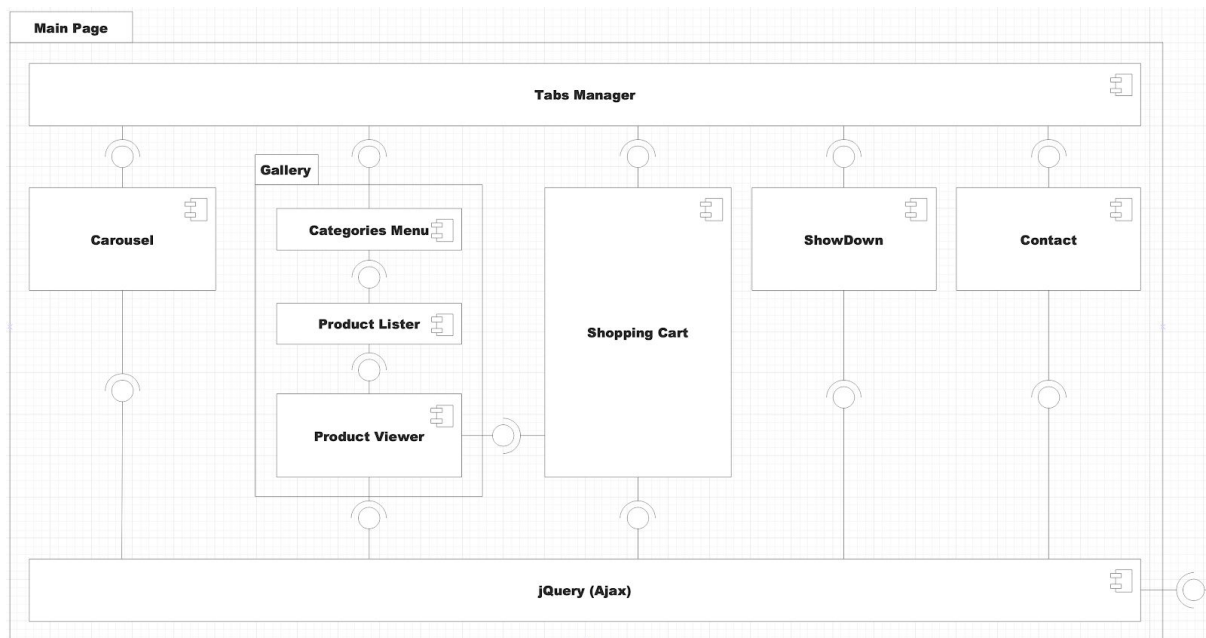
## 1.1 Diagrama completo



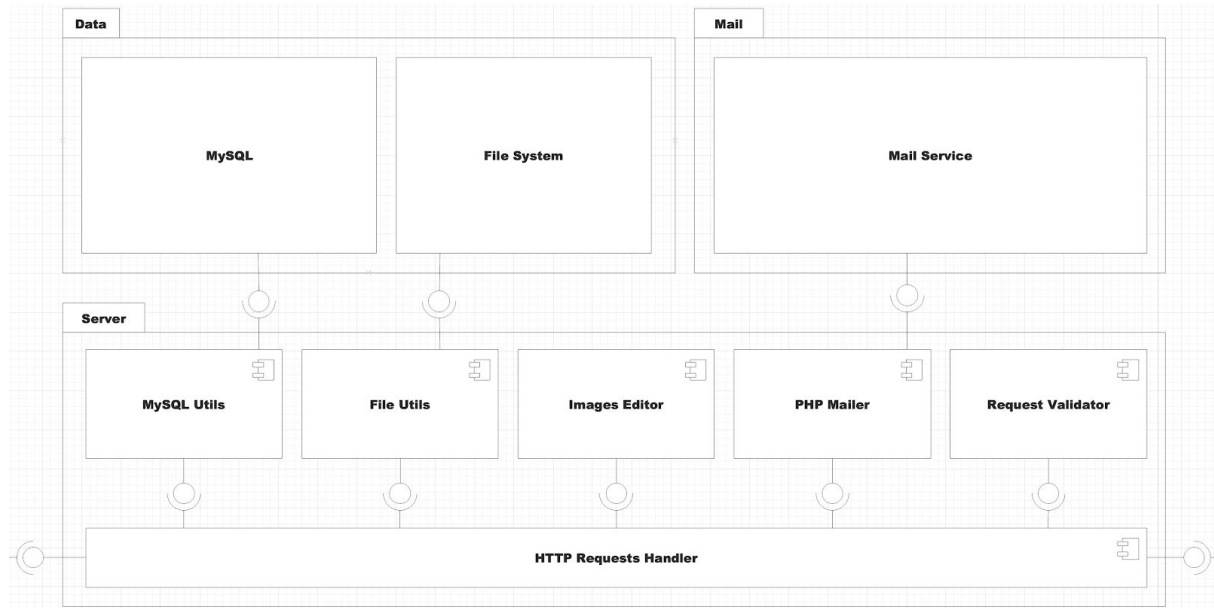
Hemos dividido el diagrama en 3 secciones con el fin de facilitar su visualización:

- **Main Page:** Muestra los componentes utilizados en la página principal.
- **Server-Data-Mail:** Muestra los componentes y servicios utilizados por el servidor.
- **Admin Page:** Muestra los componentes usados en el panel de administrador.

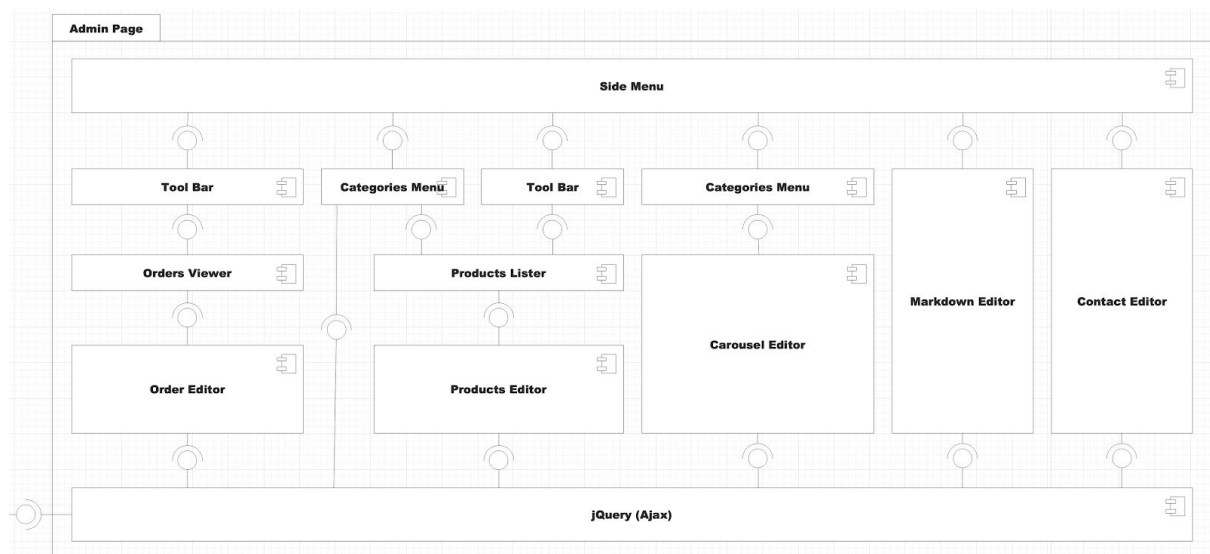
## 1.2 Main Page



### 1.3 Server-Data-Mail



## 1.4 Admin Page



## 2 Descripción de Componentes

### 2.1 Tabs Manager

<b>Lenguaje</b>	JavaScript
<b>Descripción</b>	Automatiza la creación de pestañas de navegación, permitiendo asignar funciones al evento Click de cada una.
<b>Propósito</b>	Manejar las secciones de la página principal.
<b>Dependencias</b>	jQuery

Interfaces:

<b>Interfaz</b>	createTabs( tabs : Object ) : void
<b>Descripción</b>	Crea las pestañas.
<b>Entrada</b>	<p>Objeto con la siguiente estructura:</p> <pre>{   containerId : ( ID del contendor ),   tabs: [     {       title: ( Título de la pestaña ),       onClick: ( función al hacer click )     },     ...   ] }</pre>

## 2.2 Carousel

<b>Lenguaje</b>	JavaScript
<b>Descripción</b>	Permite crear un slider automático de imágenes.
<b>Propósito</b>	Crear el slider de fotos de la sección presentación.
<b>Dependencias</b>	jQuery

Interfaces:

<https://github.com/SSilence/simple.carousel>

## 2.3 Categories Menu

<b>Lenguaje</b>	JavaScript
<b>Descripción</b>	Crea una lista vertical de pestañas responsive, a las cuales se les puede asignar una función al hacerles Click.
<b>Propósito</b>	Permite manejar las subsecciones de una sección, facilitando la visualización en dispositivos móviles.
<b>Dependencias</b>	jQuery

Interfaces:

<b>Interfaz</b>	createMenu( tabs : Object ) : void
<b>Descripción</b>	Crea las pestañas.
<b>Entrada</b>	<p>Objeto con la siguiente estructura:</p> <pre>{   containerId : ( ID del contendor ),   tabs: [     {       title: ( Título de la pestaña ),       onClick: ( función al hacer click )     },     ...   ] }</pre>

## 2.4 Product Lister

<b>Lenguaje</b>	JavaScript
<b>Descripción</b>	Permite imprimir una malla de tarjetas con imagen e información sobre un producto.
<b>Propósito</b>	Facilitar el despliegue de productos.
<b>Dependencias</b>	jQuery

Interfaces:

<b>Interfaz</b>	setCards( cards : Object ) : void
<b>Descripción</b>	Crea las tarjetas.
<b>Entrada</b>	<p>Objeto con la siguiente estructura:</p> <pre>{   containerId : ( ID del contendor ),   cards: [     {       id: ( id del producto),       img: ( Url de la imagen ),       title: ( Título de la tarjeta ),       dec: ( Descripción de la tarjeta ),       custom: ( Indica si el producto es customizable ),       onClick: ( función al hacer click )     },     ...   ] }</pre>

## 2.5 Product Viewer

<b>Lenguaje</b>	JavaScript
<b>Descripción</b>	Muestra la información completa de un producto, junto con un panel de customización, y un botón para añadir al carrito.
<b>Propósito</b>	Facilitar el despliegue y customización de un producto, por parte de los clientes.
<b>Dependencias</b>	jQuery

### Interfaces:

<b>Interfaz</b>	setInfo( info : Object ) : void
<b>Descripción</b>	Muestra la información del producto.
<b>Entrada</b>	<p>Objeto con la siguiente estructura:</p> <pre>{   containerId : ( ID del contendor ),   cart: ( Referencia al objeto carrito ),   info:   {     id: ( id del producto ),     title: ( Título ),     dec: ( Descripción del producto ),     custom: ( Indica si el producto es customizable ),     onAddToCart: ( función al hacer click ),     options: [       ( Arreglo de objetos con las customizaciones )     ]   } }</pre>



## 2.6 ShowDown

<b>Lenguaje</b>	JavaScript
<b>Descripción</b>	Convierte código Markdown a HTML.
<b>Propósito</b>	Permite mostrar la información de la sección historia creada por el administrador en formato Markdown.
<b>Dependencias</b>	None

Interfaces:

<b>Interfaz</b>	makeHTML( markdown : String ) : String
<b>Descripción</b>	Convierte código Markdown a HTML.
<b>Entrada</b>	String con el código markdown.

<https://github.com/showdownjs/showdown>

## 2.7 Contacto

<b>Lenguaje</b>	JavaScript
<b>Descripción</b>	Crea una lista de métodos de contacto ( Whatsapp, Instagram, Facebook, etc ) con sus respectivos iconos y enlaces.
<b>Propósito</b>	Mostrar los métodos de contacto del negocio.
<b>Dependencias</b>	jQuery

Interfaces:

<b>Interfaz</b>	create( info : Object ) : void
<b>Descripción</b>	Crea los contactos.
<b>Entrada</b>	Objeto con la siguiente estructura:  { containerId : ( ID del contendor ), social: [

	<pre> {   type: ( Whatsapp, Facebook, etc ),   text: ( @usuario, número, etc ),   href: ( url usuario o teléfono) }, ... ] } </pre>
--	-------------------------------------------------------------------------------------------------------------------------------------

## 2.8 jQuery ( Ajax )

<b>Lenguaje</b>	JavaScript
<b>Descripción</b>	Permite realizar llamados HTTP.
<b>Propósito</b>	Facilitar la comunicación con la API.
<b>Dependencias</b>	jQuery

Interfaces:

<https://api.jquery.com/jquery.ajax/>

## 2.9 Side Bar

<b>Lenguaje</b>	JavaScript
<b>Descripción</b>	Permite automatizar la creación de menús laterales, “escondibles”.
<b>Propósito</b>	Manejar las secciones de la página de administrador.
<b>Dependencias</b>	jQuery

Interfaces:

<b>Interfaz</b>	createMenu( tabs : Object ) : void
<b>Descripción</b>	Crea el menú.
<b>Entrada</b>	Objeto con la siguiente estructura: <pre> { </pre>

	<pre> containerId : ( ID del contendor ), tabs: [   {     icon: ( Url del ícono ),     title: ( Título de la pestaña ),     onClick: ( función al hacer click )   },   ... ] } </pre>
--	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## 2.10 MySQL Utils

<b>Lenguaje</b>	PHP
<b>Descripción</b>	Set de funciones para simplificar los llamados a la base de datos.
<b>Propósito</b>	Simplificar los llamados a la base de datos.
<b>Dependencias</b>	None

### Interfaces:

<b>Interfaz</b>	mysqlConnect( ) : void
<b>Descripción</b>	Se conecta a la base de datos, con la configuración descrita en conf.json.
<b>Entrada</b>	Credenciales MySQL desde el archivo conf.json

<b>Interfaz</b>	sqlSearchToJSON( \$sql : String ) : String
<b>Descripción</b>	Realiza una consulta a la base de datos y convierte el resultado a formato JSON.
<b>Entrada</b>	String con la consulta SQL.
<b>Salida</b>	Resultado de la consulta en formato JSON.

<b>Interfaz</b>	sqlQuery( \$sql : String , \$multi : bool = false) : Status
<b>Descripción</b>	Realiza una o múltiples consultas a la base de datos.
<b>Entrada</b>	String con la consulta SQL, y una variable para indicar si el string contiene múltiples consultas.
<b>Salida</b>	El estado final de la consulta.

<b>Interfaz</b>	sqlSearchToArray( \$sql : String ): Array
<b>Descripción</b>	Realiza consultas a la base de datos y retorna resultados en formato arreglo.
<b>Entrada</b>	String con la consulta SQL.
<b>Salida</b>	Arreglo con la respuesta.

## 2.11 Validator

<b>Lenguaje</b>	PHP
<b>Descripción</b>	Set de funciones para validar los llamados a la API REST.
<b>Propósito</b>	Añadir seguridad a la API.
<b>Dependencias</b>	None

### Interfaces:

<b>Interfaz</b>	httpError( \$error : String ) : void
<b>Descripción</b>	Termina la ejecución del llamado, generando una respuesta HTTP con código 500.
<b>Entrada</b>	String con la descripción del error.

<b>Interfaz</b>	httpOk( \$msg ) : void
<b>Descripción</b>	Termina la ejecución del llamado, generando una respuesta HTTP con código 200.
<b>Entrada</b>	String con para añadir un mensaje extra.

<b>Interfaz</b>	checkAdminLogin() : void
<b>Descripción</b>	Verifica la sesión del administrador.
<b>Entrada</b>	None

<b>Interfaz</b>	requiredParams( \$required : String Array, \$method = "POST") : void
<b>Descripción</b>	Valida que existan todos los parámetros de un cierto tipo de llamado.
<b>Entrada</b>	Arreglo de strings con los nombres de los parámetros requeridos, y otro string indicando el tipo de request.
<b>Salida</b>	Termina el programa utilizando el método httpError, si falta un parámetro.

## 2.12 PHP Mailer

<b>Lenguaje</b>	PHP
<b>Descripción</b>	Permite enviar correos programáticamente.
<b>Propósito</b>	Automatizar el envío de correos.
<b>Dependencias</b>	PHPMailer Exception / PHPMailer SMTP

Interfaces:

<b>Interfaz</b>	sendmail(\$to,\$nameto,\$subject,\$message,\$altmess)
<b>Descripción</b>	Envía un correo utilizando credenciales definidas previamente.

## 2.13 File Utils

Por definir.

## 2.14 Images Editor

Por definir.

## 2.15 HTTP Request Handler

Definido en la API.

## 2.16 Tool Bar

Por definir.

## 2.17 Orders Viewer

Por definir.

## 2.18 Orders Editor

Por definir.

## 2.19 Products Editor

Por definir.

## 2.20 Carousel Editor

Por definir.

## 2.21 MarkDown Editor

Por definir.

## 2.22 Contact Editor

Por definir.