

TABLA DE CONTENIDO

E.	MANUAL DEL PROGRAMADOR	1
E.1.	APLICACIÓN WEB	1
E.1.1.	Clase conexion	2
E.1.2.	Clase consulta	2
E.1.3.	Clase tabla	3
E.1.4.	Función Ist	4
E.1.5.	Función editar	4
E.1.6.	Función insertar	5
E.1.7.	Función mensaje	6
E.1.8.	Archivo index.php	6
E.1.9.	Archivo menu.php	6
E.1.10.	Archivo consultas.php	7
E.1.11.	Archivo busqueda.php	8
E.1.12.	Archivo passwd.php	8
E.1.13.	Archivo pedido.php	8
E.1.14.	Archivo datos.php	8
E.1.15.	Archivo salida.php	8
E.2.	APLICACIÓN WAP	9
E.2.1.	Clase buscarpedido	9
E.2.2.	Clase carrocompras	11
E.2.3.	Clase login	12
E.2.4.	Clase conexion	14
E.2.5.	Clase icliente	14
E.2.6.	Clase iagente	15
E.2.7.	Clase productos	15
E.2.8.	Clase listaproductos	16

E.2.9.	Clase confirmacion	16
E.2.10.	Clase VistasCarrito	17
E.2.11.	Clase PeticionCodigo	18
E.2.12.	Clase confirmacionenvio	18
E.2.13.	Función encabezado_WAP	19
E.2.14.	Archivo detalles.php	19
E.2.15.	Archivo detempresas.php	19
E.2.16.	Archivo empresas.php	19
E.2.17.	Archivo productos.php	20
E.2.18.	Archivo index.php	20
E.2.19.	Archivo opciones.php	20

LISTA DE TABLAS

<i>Tabla E-1. Operaciones de la clase conexión</i>	2
<i>Tabla E-2. Operaciones de la clase consulta</i>	3
<i>Tabla E-3. Operaciones de la clase tabla</i>	3
<i>Tabla E-4. Descripción de las funciones ejecutadas en el archivo Consulta.php</i>	7
<i>Tabla E-5. Operaciones de la clase buscardpedido</i>	10
<i>Tabla E-6. Descripción del switch contenido en la operación buscar</i>	11
<i>Tabla E-7. Operaciones de la clase carrocompras</i>	12
<i>Tabla E-8. Descripción del switch contenido en la operación carro</i>	12
<i>Tabla E-9. Operaciones de la clase carrocompras</i>	13
<i>Tabla E-10. Descripción del switch contenido en la operación carro</i>	14
<i>Tabla E-11. Operaciones de la clase conexión</i>	14
<i>Tabla E-12. Operaciones de la clase conexión</i>	15
<i>Tabla E-13. Operaciones de la clase conexión</i>	15
<i>Tabla E-14. Operaciones de la clase conexión</i>	16
<i>Tabla E-15. Operaciones de la clase conexión</i>	16
<i>Tabla E-16. Operaciones de la clase conexión</i>	17
<i>Tabla E-17. Operaciones de la clase conexión</i>	17
<i>Tabla E-18. Operaciones de la clase conexión</i>	18
<i>Tabla E-19. Operaciones de la clase conexión</i>	18
<i>Tabla E-20. Objetos creados en el archivo opciones.php</i>	21
<i>Tabla E-21. Operaciones ejecutadas en el archivo opciones.php</i>	22

E. MANUAL DEL PROGRAMADOR

E.1. APLICACIÓN WEB

Esta aplicación puede verse como un complemento del sistema móvil, permite que las empresas accedan y gestionen a la información concerniente a los pedidos que agentes y clientes envían utilizando la aplicación móvil. Además, facilita a la empresa modificar el estado del pedido, de forma tal que el cliente pueda tener información del progreso de atención del pedido.

Como la mayoría de aplicaciones de gestión, también permite que la empresa adicione, elimine o modifique los datos de sus clientes, agentes y productos; y cuenta con un sistema de validación de usuario, con el fin de proteger la información de cada empresa registrada en el sistema.

La aplicación está compuesta por un conjunto de clases y funciones distribuidas en los siguientes archivos:

- busqueda.php
- consultas.php
- datos.php
- funciones.php
- global.php
- index.php
- local.php
- menu.php
- passwd.php
- pedido.php
- salida.php

A continuación se describen las funciones y clases utilizadas, con sus respectivas operaciones atributos y argumentos, así como su ubicación en los archivos mencionados.

E.1.1. Clase conexion

Ubicación: funciones.php

Objetivo:

Permitir iniciar una conexión con una base de datos MySQL, crear un objeto de la clase consulta, con el cual posteriormente se ejecutarán las funciones del API de PHP para MySQL, y cerrar la conexión con la base de datos.

Atributos:

\$link: identificador de la conexión a la base de datos MySQL cuando la conexión ha sido exitosa, es utilizado para referenciar, posteriormente, la ejecución de funciones MySQL, a una conexión determinada.

Operaciones:

Nombre	Argumentos	Descripción	Retorna
conectar	nombre_servidor String, nombre_usuario String, base_de_datos String	Establece la conexión con el motor de la base de datos	
crear_consulta		Crea un objeto de la clase consulta	\$con, (objeto)
Cerrar		Termina la comunicación con el motor de la base de datos	

Tabla E-1. Operaciones de la clase conexión

E.1.2. Clase consulta

Ubicación: funciones.php

Objetivo:

Ejecutar las funciones del API de PHP para MySQL.

Atributos:

\$link: Identificador de la conexión a la base de datos MySQL a la cual se va a realizar la consulta.

\$query: Cadena que contiene la sentencia SQL a ejecutar.

\$result: Identificador de un conjunto resultado, producto de una ejecución de una sentencia SQL que contiene un SELECT.

Operaciones:

Nombre	Descripción	Retorna
ejecutar	Permite ejecutar sentencias SQL	\$result
No	Retorna el número de filas seleccionadas con una sentencia SELECT	\$rta, (entero)
liberar	Libera la memoria asociada a un conjunto resultado, producto de una sentencia SELECT	
siguiente	Retorna una fila de un conjunto de datos asociado a un identificador de resultado.	\$rta, (array)
imprimir_sql	Imprime una sentencia SQL a ejecutar.	
id	Retorna el último Id, (clave primaria) generado por un INSERT.	\$d, (entero)

Tabla E-2. Operaciones de la clase consulta

E.1.3. Clase tabla

Ubicación: funciones.php

Objetivo:

Crear la tablas de fondo, es decir, permite que una tabla anidada dentro de esta tabla, aparezca con un borde.

Operaciones:

Nombre	Argumentos	Descripción
tabla_inicio	ancho_sombra Int, sombra_tabla String, ancho_tabla Int, titulo String	Crea el inicio de la tabla
tabla_fin		Crea el final de la tabla

Tabla E-3. Operaciones de la clase tabla

E.1.4. Función Ist

Ubicación: funciones.php

Objetivo:

Crear una tabla a partir de una sentencia SQL, recibe como parámetros el objeto de la clase conexion, el número de columnas, los títulos como una cadena separados por el caracter “;” y los nombres de los estilos para los títulos y el texto de la tabla, según el tipo saca el listado solo o adiciona una columna con un enlace.

Argumentos:

\$sql: Objeto de la clase conexion.

\$ncol: Número de columnas SQL a mostrar, después de la posición 0 más la cantidad especificada en \$col_mas.

\$titulos: Títulos de la tabla separados por “;”.

\$classtitulo: Clase CSS para el estilo del título de las columnas.

\$classtexto: Clase CSS para el estilo del texto de la tabla.

\$enlaces: Lista señalizada de los enlaces de las celdas de la columna(s) adicional(es).

\$dmb_link: Lista señalizada de los títulos de las columnas adicionales.

Restricción:

La posición 0 de la tabla debe ser la clave primaria y no se tiene en cuenta para las columnas, solamente para el parámetro de los enlaces. La variable que se adiciona al enlace se llama \$lstid.

E.1.5. Función editar

Ubicación: funciones.php

Objetivo:

Crear el formulario de edición de los datos de una agente, cliente o producto. Para esto debe consultar los datos actuales y presentarlos en el formulario para que sean modificados.

Argumentos:

\$sql: Objeto de la clase conexion.

\$nombres: Nombres de los títulos de cada campo de datos.

\$classtexto: Clase CSS para el estilo del texto de la tabla.

\$lstid: Variable que almacena el código de agente, usuario o producto al cual se le deben realizar los cambios.

\$nom: Almacena el nombre de la tabla de la base de datos en la cual se deben realizar los cambios, (productos, agente o cliente).

E.1.6. Función insertar

Ubicación: funciones.php

Objetivo:

Crear el formulario para la introducción de un nuevo agente, cliente o producto.

Argumentos:

\$sql: Objeto de la clase conexion.

\$nombres: Nombres de los títulos de cada campo de datos.

\$classtexto: Clase CSS para el estilo del texto de la tabla.

\$est: Almacena el tipo de inserción de datos que el usuario esta realizando, (de agente, cliente o producto).

\$empresa: Almacena el código de la empresa que esta realizando al inserción de datos.

E.1.7. Función mensaje

Ubicación: funciones.php

Objetivo:

Desplegar al usuario mensajes de confirmación de inserción y modificación de datos.

Argumentos:

\$msg: Mensaje a desplegar

E.1.8. Archivo index.php

El Archivo contiene el esqueleto HTML para poder mostrar los datos requeridos por la empresa y los botones necesarios para su uso. En él se realiza la validación de usuario, la selección del tipo de menú que se debe mostrar y la adecuación del esqueleto de la página de acuerdo al tipo de usuario (registrado o no registrado).

Archivos incluidos:

funciones.php

menu.php

consultas.php

Objetos Creados:

\$con, de la clase conexion

\$table, de la clase tabla

E.1.9. Archivo menu.php

Selecciona y construye el menú lateral de la página de gestión, de acuerdo al tipo de usuario, (resgistrado o no registrado). En este archivo se hace una consulta a la BD, a la tabla *item_menu*, que selecciona los nombres de los botones según su categoría y posteriormente genera el menú en HTML con su nombre y su respectivo hipervínculo, de acuerdo a la categoría especificada en la variable *\$tipomenu*.

E.1.10. Archivo consultas.php

Este archivo contiene la lógica de selección del sistema, en él se trazan los caminos que debe seguir la aplicación cuando se selecciona alguno de los enlaces en la página principal de usuario registrado. Así, se selecciona la sentencia SQL adecuada para que se genere la Client page con la información solicitada. Para tal fin, este archivo contiene un *switch*, que de acuerdo al los valores tomados por las variables *\$opt* y *\$est* ejecuta una determinada función, como se describe en la siguiente tabla:

Case \$opt	Case \$est	Función utilizada	Genera
1	1	lst()	Lista de agentes
	2	lst()	Lista de repartidores
	3	lst()	Lista de Clientes
2	1	insertar()	Inserta agente
	2	insertar()	Inserta Repartidor
	3	insertar()	Inserta cliente
	4	insertar()	Inserta producto
3		lst()	Lista todos los productos
4		lst()	Lista los pedidos
edit		editar()	Edita los productos
edit1		editar()	Edita agente, cliente
pedido		ejecutar(update)	Cambiar estado pedido
grabar		ejecutar(update)	Actualiza datos
inserta		ejecutar (insert)	Inserta productos, agentes, cliente
passwd		ejecutar (update)	Cambiar contraseña
borrar		ejecutar(delete)	Borrar
buscar	1	lst()	Lista de agentes
	2	lst()	Lista de repartidores
	3	lst()	Lista de clientes
	4	lst()	Lista de productos
	5	lst()	Lista de pedidos
perfil		ejecutar(insert, delete)	
datos		include("datos.php")	Interfaz de modificación del perfil de Cliente.
local		include("local.php")	

Tabla E-4. Descripción de las funciones ejecutadas en el archivo Consulta.php

Archivos incluidos

pedido.php

passwd.php

busqueda.php

datos.php

local.php

E.1.11. Archivo busqueda.php

Contiene la estructura HTML que permite mostrar las cajas de texto necesarias para poder insertar los datos de búsqueda.

E.1.12. Archivo passwd.php

Contiene la construcción del formulario que contiene las cajas de texto que permiten al usuario modificar la contraseña.

E.1.13. Archivo pedido.php

En este archivo se genera la ficha de pedido con los datos del cliente, los productos pedidos, el valor y el estado del pedido. Por medio de este formulario la empresa puede modificar el estado del pedido, (pendiente, enviado, entregado o cancelado).

E.1.14. Archivo datos.php

Este archivo contiene estructura HTML, necesaria para que los cliente de las empresas, modifiquen su perfil, es decir, seleccionen las empresas a las cuales desean realizar pedidos. Para esto en el archivo se realiza una consulta a la base de datos para saber cuales son las empresas existentes. Además, cuenta con un formulario que permite al cliente ingresar su código y contraseña, con el fin de realizar su identificación y asegurar su validez como Cliente dentro del sistema.

E.1.15. Archivo salida.php

Desregistra, las variables de sesión y la destruye. Además, contiene la estructura HTML, de la página de confirmación de finalización de la sesión del usuario.

E.2. APLICACIÓN WAP

Esta aplicación fue desarrollada para un dispositivo móvil que soporte WAP, idealmente, el R280d de Ericsson. La aplicación permite a un usuario registrado como Cliente o Agente en la base de datos del sistema, realizar pedidos y almacenarlos de tal forma que la empresa distribuidora de los productos, pueda acceder a esta información utilizando la plataforma Web descrita anteriormente.

El código de la aplicación se encuentra distribuido en los siguientes archivos:

- buscar.php
- carro.php
- detalles.php
- detempresas.php
- empresas.php
- funciones.php
- index.php
- inicio.php
- login.php
- opciones.php
- productos.php

A continuación se describen las funciones y clases utilizadas, con sus respectivas operaciones atributos y argumentos, así como su ubicación en los archivos mencionados.

E.2.1. Clase buscarpedido

Ubicación: buscar.php

Objetivo:

Crear las interfaces necesarias para realizar una búsqueda de pedido, tanto para Clientes como para Agentes.

Operaciones:

Nombre	Argumentos	Descripción
buscar	tipo String, paso String, login String,	Crea las interfaces para recolectar la información necesaria para realizar una búsqueda de pedido, dependiendo del tipo de

	pass String, iniemp String, emp Int, tipobusqueda String, codpedido Int, codproducto Int, codcliente Int, pedido Int.	usuario, (cliente o agente) y del tipo de búsqueda, (por código o listado de los 5 últimos pedidos). Posteriormente, realiza y despliega la búsqueda.
--	--	---

Tabla E-5. Operaciones de la clase buscarpedido

La operación buscar esta constituida por un switch, que de acuerdo a la variable \$paso construye las interfaces necesarias para realizar la recolección de información para una búsqueda de pedido, utilizando el dispositivo móvil. Su estructura se describe a continuación:

\$paso	\$tipo ¹	Construye	Paso siguiente	Paso de retorno
1	c	Card que solicita la introducción de la inicial de la empresa a la que realizó el pedido que está buscando.	\$paso=2	Interfaz inicial de cliente
1	a	Card que solicita la introducción del código del cliente que realizó el pedido que está buscando.	\$paso=cliente	Interfaz inicial de agente
2	c	Card que contiene la lista de las empresas a las cuales el cliente esta suscrito e inician con los caracteres introducidos en el paso anterior.	\$paso=3	\$paso=1
3	c	Card que permite escoger al cliente el criterio de búsqueda, (por código de transacción o listar los últimos 5 pedidos)	Búsqueda por código: \$paso=4 Listado de los últimos 5 pedidos \$paso=lista	\$paso=2
3	a	Card que permite escoger al cliente el criterio de búsqueda, (por código de transacción o listar los últimos 5 pedidos)	Búsqueda por código: \$paso=4 Listado de los últimos 5 pedidos \$paso=lista	\$paso=1
4	a ó c	Card que permite al usuario introducir el código de la transacción que le fue enviado	\$paso=5	\$paso=3

¹ La variable \$tipo almacena el tipo de usuario que se encuentra realizando el pedido, \$tipo="c", en el caso de un cliente y \$tipo="a" en el caso de un agente.

		cuando realizó el pedido que esta buscando		
5	a ó c	Card que contiene los detalles del pedido encontrado.	Si el estado del pedido es "Pendiente", (\$estado="Pendiente"), \$paso="cancelar"	Si el tipo de búsqueda es por código, (\$tipobusqueda='c'), \$paso=4; si corresponde a la lista de los últimos 5 pedidos, (\$tipobusqueda='lista'), \$paso='lista'
lista	a ó c	Card que contiene la lista de los últimos 5 pedidos realizados a la empresa seleccionada por el cliente o a la que pertenece el agente.	\$paso=3	\$paso=5
det	a ó c	Card que contiene la información detallada acerca de un producto seleccionado en el card que contiene la información detallada del pedido encontrado		\$paso=5
cliente	a	Card que contiene el nombre del cliente que corresponde al código que el agente introdujo, en el paso 1	\$paso=3	
cancelar	a ó c	Card que contiene la confirmación de que el pedido ha sido cambiado de estado "Pendiente" a "Cancelado".	\$paso=5	

Tabla E-6. Descripción del switch contenido en la operación buscar

E.2.2. Clase carrocompras

Ubicación: carro.php

Objetivo:

Crear las interfaces necesarias para construir, desplegar, modificar y enviar el carrito de compras.

Operaciones:

Nombre	Argumentos	Descripción	Retorna
carro	tipovista String, carro Array, codigoproducto Int, proc String,	Crea las interfaces necesarias para que el usuario pueda ver los productos contenidos en el	

	codemp Int, idemp Int, idproducto Int,	carrito de compras, así como, modificar su contenido o realizar el envío del mismo.	
obtenercodigos	\$carro	Obtiene los códigos de los productos contenidos en el carrito de compras, para armar una sentencia SQL	Una cadena que contiene los códigos de los productos contenidos en el carrito de compras, separados por la cadena " or "

Tabla E-7. Operaciones de la clase carrocompras

La operación carrocompras esta constituida por un switch, que de acuerdo a la variable *\$tipovista* construye las interfaces que permiten al usuario ver el carrito de compras, modificarlo o enviarlo como un pedido, su estructura se describe a continuación:

\$tipovista	Construye
i	Construye un deck que contiene la vista detallada de productos en el carrito de compras, "Vista por items".
det	Construye un deck que contiene dos cards, el primero permite ver los detalles del producto seleccionado en la "Vista por items" y un enlace, que accede al siguiente card, el cual incluye un formulario que permite modificar la cantidad del producto en el carrito de compras.
el	Construye la lista de empresas a las que pertenecen los productos que se encuentran en el carrito de compras, a esta opción solamente acceden los clientes, en este caso la variable <i>\$tipovista</i> , toma el valor "elim".
em	Construye la lista de empresas a las que pertenecen los productos que se encuentran en el carrito de compras, a esta opción solamente acceden los clientes, en este caso la variable <i>\$tipovista</i> , toma el valor "i".
elim.	Construye una lista de los productos pertenecientes a la empresa escogida en la vista por empresas. Permite eliminar el producto que se seleccione.
Borrrart	Construye una interfaz que pide confirmación al usuario para eliminar todos los productos que se encuentran en el carrito de compras.
Borrar	Construye una interfaz que pide al usuario confirmación para eliminar uno de los productos del carrito de compras.

Tabla E-8. Descripción del switch contenido en la operación carro

E.2.3. Clase login

Ubicación: login.php

Objetivo:

Validar al usuario que intenta ingresar al sistema y si se encuentra registrado, mostrarle la información que va de acuerdo con su perfil.

Atributos:

\$tipo: Variable que te determina el tipo de usuario, (Cliente o Agente).

\$codigo: Almacena el código del usuario que intenta registrarse.

\$registrado: Bandera que se enciende a verdadero cuando el usuario obtiene una validación exitosa.

Operaciones:

Nombre	Argumentos	Descripción
verTipo	login String, carro Array, password String	De acuerdo a la estructura del código introducido determina el tipo de usuario que esta intentando ingresar al sistema y posteriormente realiza la validación.

Tabla E-9. Operaciones de la clase carrocompras

La operación carrocompras esta constituida por un switch, que de acuerdo a la variable \$tipovista construye las interfaces que permiten al usuario ver el carrito de compras, modificarlo o enviarlo como un pedido, su estructura se describe a continuación:

\$tipovista	Construye
i	Construye un deck que contiene la vista detallada de productos en el carrito de compras, "Vista por items".
det	Construye un deck que contiene dos cards, el primero permite ver los detalles del producto seleccionado en la "Vista por items" y un enlace, que accede al siguiente card, el cual incluye un formulario que permite modificar la cantidad del producto en el carrito de compras.
el	Construye la lista de empresas a las que pertenecen los productos que se encuentran en el carrito de compras, a esta opción solamente acceden los clientes, en este caso la variable \$tipovista, toma el valor "elim".
em	Construye la lista de empresas a las que pertenecen los productos que se encuentran en el carrito de compras, a esta opción solamente acceden los clientes, en este caso la variable \$tipovista, toma el valor "i".
elim	Construye una lista de los productos pertenecientes a la empresa escogida en la vista por empresas. Permite eliminar el producto que se seleccione.

borrart	Construye una interfaz que pide confirmación al usuario para eliminar todos los productos que se encuentran en el carrito de compras.
borrar	Construye una interfaz que pide al usuario confirmación para eliminar uno de los productos del carrito de compras.

Tabla E-10. Descripción del switch contenido en la operación carro

E.2.4. Clase conexion

Ubicación: funciones.php

Objetivo:

Permitir iniciar una conexión con una base de datos MySQL, crear un objeto de la clase consulta, con el cual posteriormente se ejecutarán las funciones del API de PHP para MySQL, y cerrar la conexión con la base de datos.

Atributos:

\$link: identificador de la conexión a la base de datos MySQL cuando la conexión ha sido exitosa, es utilizado para referenciar, posteriormente, la ejecución de funciones MySQL, a una conexión determinada.

Operaciones:

Nombre	Argumentos	Descripción	Retorna
conectar	nombre_servidor String, nombre_usuario String, base_de_datos String	Establece la conexión con el motor de la base de datos	
cerrar		Termina la comunicación con el motor de la base de datos	

Tabla E-11. Operaciones de la clase conexión

E.2.5. Clase icliente

Ubicación: funciones.php

Objetivo:

Construir la interfaz inicial de un cliente registrado, consultando, en la base de datos, cuales son las empresas en las que se encuentra registrado.

Operaciones:

Nombre	Argumentos	Descripción
ConsultarEmpresas	CodigoCliente String	Realiza una consulta en la base de datos, de las empresas en las que el cliente con el código cliente se encuentra registrado, y con el resultado construye un conjunto de <i>options</i> que contienen los nombres y códigos de dichas empresas.

Tabla E-12. Operaciones de la clase conexión

E.2.6. Clase iagente**Ubicación:** funciones.php**Objetivo:**

Construir la interfaz inicial de un agente registrado, consultando, en la base de datos, cuales son los productos que pertenecen a la empresa en la que el agente se encuentra registrado..

Operaciones:

Nombre	Argumentos	Descripción
ConsultarProductos	CodigoAgente String	Realiza una consulta en la base de datos, de los productos que pertenecen a la empresa en la cual el cliente se encuentra registrado, y con el resultado construye un conjunto de <i>options</i> que contienen los nombres y códigos de dichos productos.

Tabla E-13. Operaciones de la clase conexión

E.2.7. Clase productos**Ubicación:** funciones.php**Objetivo:**

Construye una interfaz que contiene la información detallada de un producto, seleccionado en la lista de productos de una empresa.

Operaciones:

Nombre	Argumentos	Descripción
DetallesProductos	CodigoProducto String	Construye dos cards: el primero contiene la información detallada del producto, la cual se construye a partir de una consulta a la base de datos; el segundo contiene un input por medio del cual el usuario puede introducir la cantidad del producto que desea adicionar al carrito de compras.

Tabla E-14. Operaciones de la clase conexión

E.2.8. Clase listaproductos**Ubicación:** funciones.php**Objetivo:**

Construye la interfaz que contiene la lista de productos que pertenecen a una empresa seleccionada previamente en la lista de empresas de un usuario registrado.

Operaciones:

Nombre	Argumentos	Descripción
ListarProductos	CodigoEmpresa String codigo String carro Array	Construye una interfaz en la cual por medio de un select se le muestran al usuario los productos pertenecientes a una empresa..

Tabla E-15. Operaciones de la clase conexión

E.2.9. Clase confirmacion**Ubicación:** funciones.php**Objetivo:**

Construir las interfaces en las que al usuario se le muestra información corta que no necesitan de interacción con él para continuar, tal como, confirmación de ejecución de operaciones.

Operaciones:

Nombre	Argumentos	Descripción
confirmar	tipo String url String	Un card que contiene un enlace y un timer que llevan al usuario a la página en la que estaba antes de la confirmación.

Tabla E-16. Operaciones de la clase conexión

La operación confirmar esta constituida por un switch, que de acuerdo a la variable *\$tipo* construye las interfaces con la confirmación y el enlace adecuado:

<i>\$tipo</i>	Mensaje
1	Articulo insertado
2	Cantidad actualizada
3	Producto eliminado
4	El carrito de compras está vacío
5	Código de cliente no asignado
6	No se han encontrado coincidencias
7	Ha salido exitosamente del sistema
8	Pedido cancelado

E.2.10. Clase VistasCarrito

Ubicación: funciones.php

Objetivo:

Construye la interfaz que contiene la lista de productos que pertenecen a una empresa seleccionada previamente en la lista de empresas de un usuario registrado.

Operaciones:

Nombre	Argumentos	Descripción
vistas	carro Array, proc String login String pass String codemp Int idemp Int	Construye la interfaz que permite al usuario escoger el tipo de vista del carrito de compras. De acuerdo al valor tomado por la variable \$proc, define la URL a la que debe retornar, es decir, el lugar de donde se llamó el carrito de compras, (lista de empresas o lista de productos)

Tabla E-17. Operaciones de la clase conexión

E.2.11. Clase PeticionCodigo

Ubicación: funciones.php

Objetivo:

Construir la interfaz que solicita al agente el código del cliente, al cual está representando, en el momento de enviar un pedido o realizar un búsqueda.

Operaciones:

Nombre	Argumentos	Descripción
codigo_cliente	proc String codemp Int id Int idprod Int idemp Int	Construye un card que contiene una etiqueta input en la cual el agente debe introducir el nombre del cliente al cual representa.

Tabla E-18. Operaciones de la clase conexión

E.2.12. Clase confirmacionenvio

Ubicación: funciones.php

Objetivo:

Construir la interfaz que solicita al usuario la confirmación de envío del contenido del carrito de compras como un pedido.

Operaciones:

Nombre	Argumentos	Descripción
confirmar	proc String codemp Int id Int idprod Int idemp Int	Construye un card que pide al usuario que confirme el envío del carrito de compras como un pedido.

Tabla E-19. Operaciones de la clase conexión

E.2.13. Función encabezado_WAP

Ubicación: funciones.php

Objetivo:

Imprimir el encabezado del documento WML, (header y prólogo WML).

E.2.14. Archivo detalles.php

A este archivo tienen acceso los usuarios no registrados, en el se encuentra el código que construye los detalles de un producto perteneciente a una empresa seleccionada, previamente.

Archivos incluidos:

funciones.php

Objetos Creados:

\$con, de la clase conexion

E.2.15. Archivo detempresas.php

A este archivo tienen acceso los usuarios no registrados, en el se encuentra el código que construye los detalles de una empresa, seleccionada de la lista de empresas de registradas en el sistema.

Archivos incluidos:

funciones.php

Objetos Creados:

\$con, de la clase conexion

E.2.16. Archivo empresas.php

A este archivo tienen acceso los usuarios no registrados, en el se encuentra el código que construye lista de empresas de registradas en el sistema.

Archivos incluidos:

funciones.php

Objetos Creados:

\$con, de la clase conexion

E.2.17. Archivo productos.php

A este archivo tienen acceso los usuarios no registrados, en el se encuentra el código que construye la lista de productos pertenecientes a una de las empresas seleccionada, previamente, de la lista de empresas registradas en el sistema.

Archivos incluidos:

funciones.php

Objetos Creados:

\$con, de la clase conexion

E.2.18. Archivo index.php

Construye la interfaz inicial del sistema, las opciones para usuarios registrados y no registrados.

Archivos incluidos:

funciones.php

E.2.19. Archivo opciones.php

A este archivo se realizan todas las peticiones de ejecución de funciones del sistema, y de acuerdo a la variable \$acc, dirige la petición hacia la clase o función indicada. Además aquí se realiza el registro de las variables de sesión de un usuario registrado.

Archivos incluidos:

funciones.php

login.php

carro.php

buscar.php

Objetos Creados:

Objeto	Clase
\$con	conexión
\$productos	listaproductos
\$detalles	productos
\$doLogin	login
\$confirma	confirmacion
\$ver	vistascarrito
\$carrito	carrocompras
\$confirmacio	confirmacionenvio
\$cod_cliente	peticioncodigo
\$búsqueda	buscapedido

Tabla E-20. Objetos creados en el archivo opciones.php

Los objetos creados y las funciones ejecutadas en este archivo, dependen del valor tomado por la variable \$acc. De esta forma el archivo contiene un switch que evalúa esta variable, como se muestra a continuación:

\$acc	Ejecuta
prod	Crea una instancia de la clase listaproductos y ejecuta la operación listarproductos
det	Crea una instancia de la clase productos y ejecuta la operación detallesproductos
emp	Crea una instancia de la clase login y ejecuta la operación verTipo
add	Adiciona un producto al carrito de compras, si el producto ya esta en el carrito de compras simplemente aumenta la cantidad del ya existente. Finalmente, crea una instancia de la clase confirmación y ejecuta la operación confirmar, (\$tipo=1).
vis	Crea una instancia de la clase vistascarrito ejecuta la operación vistas.
car	Crea una instancia de la clase carrocompras y ejecuta la operación carro.
edit	Modifica la cantidad de un producto dentro del carrito de compras, crea una

	instancia de la clase confirmacion y ejecuta la operación confirmar, (\$tipo=2).
opel	elimina un producto del carrito de compras, verifica si ya no quedan más productos pertenecientes a la misma empresa del último producto eliminado o si el carrito de compras se encuentra vacío, posteriormente, crea una instancia de la clase confirmacion y ejecuta la operación confirmar, (\$tipo=3ò \$tipo=4).
elt	Elimina todos los productos del carrito de compras, desregistrando la variable de sesión, luego, crea una instancia de la clase confirmacion y ejecuta la operación confirmar, (\$tipo=4).
envio	Si se trata de un agente, crea una instancia de la clase peticioncodigo y ejecuta la operación codigo_cliente; de lo contrario crea una instancia de la clase confirmacionenvio y ejecuta la operación confirmar.
pedir	Realiza las consultas necesarias para registrar el pedido en la base de datos, dependiendo de si se trata de un agente o de un cliente, verifica si todos los productos del carrito de compras han sido enviados y finalmente recupera el código de la transacción generado y lo presenta al usuario.
buscar	Crea una instancia de la clase buscarpedido y ejecuta la operación buscar.
salir	Elimina todas las variables registradas en la sesión, la destruye y finalmente crea una instancia de la clase confirmacion y ejecuta la operación confirmar, (\$tipo=7).

Tabla E-21. Operaciones ejecutadas en el archivo opciones.php