

**ANEXO 3**

**CREACION DEL SERVICIO**

**CENTRO COMERCIAL VIRTUAL COMO SOPORTE PARA EL  
DESARROLLO SOCIOECONOMICO DEL DEPARTAMENTO DEL  
CAUCA**

**JHON EDWIN PRADO PALTA  
WILLINTHON SILVA CAICEDO**

**DIRECTOR: ING. DIEGO MAURICIO LOPEZ GUTIERREZ**

**UNIVERSIDAD DEL CAUCA  
FACULTAD DE INGENIERÍA ELECTRÓNICA Y  
TELECOMUNICACIONES  
POPAYÁN  
2002**

## **CENTRO COMERCIAL VIRTUAL PARA EL PORTAL TAMPU**

### **ANEXO 3.- CREACION DEL SERVICIO**

#### **TABLA DE CONTENIDO**

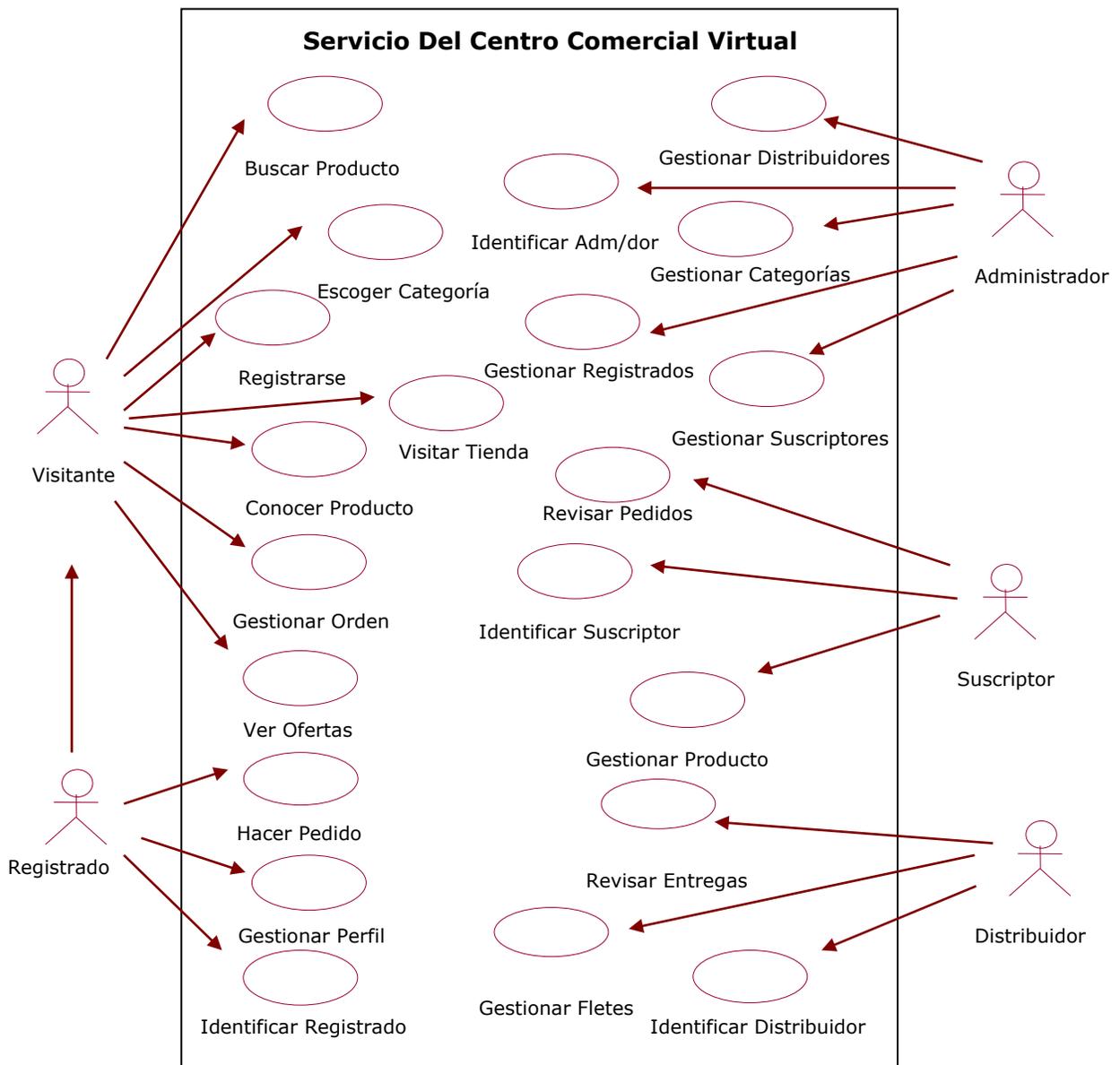
	<b>Pág.</b>
Subproducto Esencial No. 1: Modelos Completos	91
Subproducto Esencial No. 1A: Modelo de Casos de Uso del Servicio	91
Subproducto Esencial No. 1B: Modelo de Análisis del Servicio	115
Subproducto Esencial No. 1C: Modelo de Diseño del Servicio	128
Subproducto Esencial No. 1D: Modelo de Despliegue del Servicio	151
Subproducto Esencial No. 1E: Modelo de Implementación del Servicio	152
Subproducto Esencial No. 1F: Modelo de Pruebas del Servicio	154
Subproducto Esencial No. 2: Descripción de Arquitectura del Servicio	157
Subproducto Esencial No. 4: Lista de Riesgos Actualizada	157
Subproducto Esencial No 5: Manual de usuario y Material de Apoyo	160
Subproducto Esencial No. 6: Plan del Proyecto	160
Subproducto Esencial No. 7: Caso del Negocio	162
Subproducto Complementario No 1	165
Subproducto Complementario No 2	167
Subproducto Complementario No 3	176

### FASE 3: CREACIÓN DEL SERVICIO DEL CENTRO COMERCIAL VIRTUAL

#### Subproducto Esencial No. 1: Modelos Completos

#### Subproducto Esencial No. 1A: Modelo de Casos de Uso del Servicio

#### 1. Diagrama de Casos de Uso del Servicio



## 1. Descripción de los Escenarios de los casos de uso.

### **Caso de Uso Buscar 1**

**Iniciador:** Visitante.

**Precondición:** ninguna.

#### **Flujo de eventos**

#### **Flujo principal**

1. El Visitante ingresa a la opción de buscar y digita el criterio de búsqueda.
2. El servicio procesa su petición y le entrega la información de los productos relacionados con su criterio de búsqueda indicando nombre, categoría, empresa a la cual pertenece el producto y el enlace a la descripción del producto y su respectiva imagen.
3. El Visitante selecciona un enlace a seguir según el producto que desee.
4. El servicio le entrega (despliega) la respectiva petición.

#### **Poscondiciones**

Visitante con posibilidad de seguir navegando a través del servicio o estableciendo búsquedas.

#### **Flujos alternativos**

En 2 Es posible que el servicio no encuentre ningún producto que coincida con las palabras procesadas, de tal manera que se le entrega un mensaje al visitante que indique que no se encontró ninguna coincidencia.

En 3 el Suscriptor puede iterar sus búsquedas.

#### **Excepciones**

Fallas de comunicación.

Decisión por parte del Visitante de interrumpir alguna actividad.

#### **Recursos especiales utilizados**

Base de datos: Para consultar y ubicar la información.

#### **Guis relacionadas:**

GUI\_Consultas

GUI\_Resultado\_Consulta

## **Caso de Uso Identificar Suscriptor 2**

**Iniciador:** Suscriptor.

**Precondición:** ninguna.

### **Flujo de eventos**

#### **Flujo principal**

1. El Suscriptor accede a la página de Identificación del Suscriptor.
2. El Suscriptor se identifica con su login y password e indica al servicio que ha ingresado los datos.
3. El servicio valida el Suscriptor y de ser válido le permite ingresar a la pagina Principal (Menú Principal)
4. El Suscriptor recibe la página de menú principal de Suscriptor.

#### **Poscondiciones**

Sesión de Suscriptor iniciada.

#### **Flujos alternativos**

En 3 puede haber variación debido a la validación del Suscriptor, por que al no reconocerlo como valido le reenvía una pagina solicitándole el login y password de nuevo

#### **Excepciones**

Fallos de comunicación.  
Abandono del proceso por parte del Suscriptor.

#### **Recursos especiales utilizados**

Base de datos de usuarios: Consulta para verificación.

#### **Guis relacionadas:**

GUI\_IU\_Identificacion\_Suscriptor  
GUI\_Sesion\_Suscriptor

### **Caso de Uso Gestionar Producto 3**

**Iniciador:** Suscriptor.

**Precondición:** Sesión de Suscriptor iniciada.

#### **Flujo de eventos**

##### **Flujo principal**

1. Desde la página principal de Suscriptor del servicio, el Suscriptor escoge la opción que le permite gestionar un Producto. El servicio entonces le entrega la página de Gestión de Productos.
2. En la página de Gestión de Productos tiene la opción de modificar datos de un producto o adicionar un producto nuevo a su tienda. El Suscriptor escoge la opción que desee.
3. El servicio entrega al Suscriptor una página donde este puede llenar los campos de Nombre, foto, categoría, Empresa, Peso, valor, Comentarios del Producto y un campo donde pueda adicionar otras características que dependerán del tipo de producto.
4. El Suscriptor ingresa la totalidad de los datos anteriormente mencionados y le informa al servicio que ha ingresado los datos mediante un botón.
5. El servicio registra los datos y actualiza los datos.

##### **Poscondiciones**

Productos actualizados en características o ubicación.

##### **Flujos alternativos**

No definidos

##### **Excepciones**

No definidos

##### **Recursos especiales utilizados**

Registro de datos de Producto: Para consultar y ubicar la información.

##### **Guis relacionadas:**

GUI\_IAMenúPrincipal

GUI\_IAResultadoProducto

#### **Caso de Uso Gestionar Perfil 4**

**Iniciador:** Registrado

**Precondición:** Sesión de Registrado iniciada.

#### **Flujo de eventos**

##### **Flujo principal**

1. El Registrado accede por medio de un vínculo a Gestionar su Perfil
2. El servicio suministra al Registrado el formulario para modificación de datos con los ítems posible a cambiar.
3. El Registrado modifica y los confirma.
4. El servicio verifica que los datos sean válidos, los incorpora y envía una confirmación al Suscriptor.
5. El Registrado recibe la confirmación y el servicio actualiza los datos

##### **Postcondiciones**

Datos de Registrado actualizado, con posibilidad de seguir navegando por el servicio.

##### **Flujos alternativos**

En 4 si el Registrado no llenó todos los datos requeridos o son inconsistentes entonces se le envía un mensaje al Registrado con la inconsistencia y le entrega el formulario a llenar conservando los datos validos (para no volverlos a llenar) iterando la secuencia hasta validar.

##### **Excepciones**

Fallas de comunicación.

Decisión del Registrado de abandonar el proceso de actualización de datos.

##### **Recursos especiales utilizados**

Base de datos Actualizada.

GUI'S Requeridas:

GUI\_Formulario\_Registrado

### **Caso de Uso Registrarse 5**

**Iniciador:** Visitante

**Precondición:** ninguna.

#### **Flujo de eventos**

##### **Flujo principal**

1. El visitante navega a través del servicio, teniendo la posibilidad de registrarse en cualquier momento
2. Presiona la opción de registrarse
3. El servicio entrega el formulario de registro.
4. Ingresa los datos respectivos (identificación, datos personales, código, facultad).
5. El servicio valida los datos respectivos y de ser consistentes incorpora al nuevo Registrado y le presenta la serie de facilidades que el servicio tiene para los usuarios registrados.
6. El nuevo Registrado podrá acceder a la totalidad de los servicios que se ofrecen.

##### **Postcondiciones**

Nuevo Registrado en el servicio.  
Sesión de Registrado activada.

##### **Flujos alternativos**

En 5 el servicio al validar los datos encuentra opciones que invalida la solicitud y devuelve un mensaje al Visitante con la inconsistencia y le entrega el formulario a llenar conservando los datos validos (para no volverlos a llenar) iterando la secuencia hasta validar.

Las posibilidades por las cuales no pueden ser validados son:

- Campos requeridos vacíos:
- Correo electrónico ya utilizado.

##### **Excepciones**

Fallas de comunicación o inestabilidad del servicio eventual.  
Visitante cierra la página.

##### **Recursos especiales utilizados**

Bases de datos para actualizarla.

##### **GUI'S Requeridas:**

GUI\_Formulario\_Registrado

### **Caso de Uso Identificar Administrador 6**

**Iniciador:** Administrador

**Precondición:** ninguna.

#### **Flujo de eventos**

##### **Flujo principal**

1. El Administrador ejecuta la página de administración del servicio.
2. El Administrador se identifica con su login y password.
3. El servicio valida el Administrador y de ser valido le permite iniciar una sesión de Administrador ubicándolo en el menú de inicio de Administrador del servicio.

##### **Postcondiciones**

Sesión de Administrador iniciada.

##### **Flujos alternativos**

En 3 puede haber variación debido a la validación del Administrador, por que al no reconocerlo como valido le reenvía una pagina solicitándole el login y password de nuevo.

##### **Excepciones**

Fallos de comunicación.  
Abandono del proceso por parte del Administrador.

##### **Recursos especiales utilizados**

Servicio de identificación de Administrador.

GUI'S Requeridas:  
GUI\_IU\_Identificacion\_Administrador  
GUI\_Sesion\_Administrador

### **Caso de Uso Conocer Producto 7**

**Iniciador:** Visitante

**Precondición:** Haber ubicado un producto.

#### **Flujo de eventos**

##### **Flujo principal**

1. El Visitante selecciona ver características de un producto en particular
2. El servicio le entrega (despliega) la información solicitada
3. El Visitante selecciona algunas características modificables del producto a su preferencia, como color del producto, intensidad horaria del servicio, etc.

##### **Postcondiciones**

Visitante con posibilidad de seguir navegando a través del servicio y eligiendo productos a ver.

Visitante con posibilidad de adicionar el producto a su orden de compras

##### **Flujos alternativos**

En 2 la información entregada dependerá de cada producto en específico, por lo general se incluirá una descripción, una foto y el precio por unidad.

En 3 algunos productos no tendrán la opción de modificar las características o es posible que aun cuando la tenga la opción el Visitante no desee seleccionar características, entonces se incluirán en la orden las características por defecto.

##### **Excepciones**

Fallas de comunicación.

Decisión por parte del Visitante de interrumpir alguna actividad.

##### **Recursos especiales utilizados**

Base de datos: Para acceder a un producto en especial.

Paginas del sitio web: para navegar buscando productos.

GUI'S Requeridas:

GUI\_Producto

### **Caso de Uso Visitar Tienda 8**

**Iniciador:** Visitante

**Precondición:** ninguna.

#### **Flujo de eventos**

##### **Flujo principal**

1. El Visitante selecciona ver una de las tiendas que aparecen en la página inicial del servicio.
2. El servicio le presenta la descripción de la tienda solicitada y las categorías de productos que esta vende.
3. El Visitante accede a una de las categorías que maneja la tienda
4. El sistema le despliega la lista de productos de la categoría seleccionada.

##### **Postcondiciones**

Visitante con posibilidad de seguir navegando a través de las opciones del servicio

##### **Flujos alternativos**

En 2 si la tienda no tiene categorías se presentará, además de la descripción, la lista de los productos.

##### **Excepciones**

Fallas de comunicación.

Decisión por parte del Visitante de interrumpir alguna actividad.

##### **Recursos especiales utilizados**

Base de datos: Para acceder a una tienda.

##### **GUI'S Requeridas:**

GUI\_Tienda

### **Caso de Uso Gestionar Orden 9**

**Iniciador:** Visitante

**Precondición:** Haber adicionado al menos un producto en su orden de compras o estar dentro de la interfaz que le permite conocer un producto.

#### **Flujo de eventos**

##### **Flujo principal**

1. El Visitante elige la opción gestionar orden.
2. El servicio le entrega una lista con los productos que ha adicionado a su orden de compras, con un espacio para que el visitante pueda cambiar la cantidad de productos, el precio, además de otras opciones como eliminar producto de la orden y pagar.
3. El servicio le da la oportunidad al visitante para confirmar que desea comprar lo que ha adicionado a su orden.
4. El Visitante confirma que está lista su orden para pagarla.

##### **Postcondiciones**

Visitante con posibilidad de seguir navegando a través del servicio y eligiendo productos para conocer.

##### **Flujos alternativos**

En 1 el Visitante también puede llegar a este caso de uso cuando adiciona un producto a la orden de compra.

En 3 el Visitante puede salir de esta de esta interfaz para seguir conociendo productos e incluso puede abandonar el portal o cerrar el explorador, el sistema guardará su orden para cuando vuelva a entrar al servicio.

##### **Excepciones**

Fallas de comunicación.

Decisión por parte del Visitante de interrumpir alguna actividad.

##### **Recursos especiales utilizados**

Base de datos: Para acceder a un producto en especial.

Paginas del sitio web: para navegar buscando productos.

##### **GUI'S Requeridas:**

GUI\_Orden

### **Caso de Uso Gestionar Suscriptores 10**

**Iniciador:** Administrador

**Precondición:** Sesión de Administrador iniciada

#### **Flujo de eventos**

##### **Flujo principal**

1. El Administrador solicita al servicio la interfaz de gestión de Suscriptores.
2. El servicio despliega una interfaz con un listado de los Suscriptores de los cuales se puede seleccionar al que se vaya a modificar o eliminar, además de la opción que permite adicionar un nuevo Suscriptor
3. El Administrador escoge la opción modificar Suscriptor
4. El servicio presenta los campos que se requieren con los respectivos datos del Suscriptor.
5. El Administrador acepta los datos
6. El servicio valida y/o actualiza la información de los Suscriptores.

##### **Postcondiciones**

Administrador con posibilidad de seguir accediendo a las opciones de gestión de Suscriptor  
Base de datos actualizada.

##### **Flujos alternativos**

En 3 el Administrador escoge la opción eliminar Suscriptor y éste queda eliminado de la base de datos y de la interfaz. También puede escoger la opción ingresar nuevo Suscriptor, entonces el servicio le entregará el formulario de Suscriptor, el Administrador llena los campos y acepta.  
En 5 el servicio no valida los datos por que falta algún campo necesario o el Suscriptor ya existía, el servicio devolverá el formulario con los datos válidos llenos.

##### **Excepciones**

Fallas de comunicación.  
Decisión por parte del Administrador de interrumpir alguna actividad.

##### **Recursos especiales utilizados**

Base de datos: Para acceder a un Suscriptor.

##### **GUI'S Requeridas:**

GUI\_Gestión\_Suscriptor  
GUI\_Formulario\_Suscriptor

### **Caso de Uso Gestionar Distribuidores 11**

**Iniciador:** Administrador

**Precondición:** Sesión de Administrador iniciada

#### **Flujo de eventos**

#### **Flujo principal**

1. El Administrador solicita al servicio la interfaz de gestión de Distribuidores.
2. El servicio despliega una interfaz con un listado de los Distribuidores de los cuales se puede seleccionar al que se vaya a modificar o eliminar, además de la opción que permite adicionar un nuevo Distribuidor
3. El Administrador escoge la opción modificar Distribuidor
4. El servicio presenta los campos que se requieren con los respectivos datos del Distribuidor.
5. El Administrador acepta los datos
6. El servicio valida y/o actualiza la información de los Distribuidores.

#### **Postcondiciones**

Administrador con posibilidad de seguir accediendo a las opciones de gestión de Distribuidor

Base de datos actualizada.

#### **Flujos alternativos**

En 3 El Administrador escoge la opción eliminar Distribuidor y éste queda eliminado de la base de datos y de la interfaz. También puede escoger la opción ingresar nuevo Distribuidor, entonces el servicio le entregará el formulario de Distribuidor, el Administrador llena los campos y acepta.

En 5 el servicio no valida los datos por que falta algún campo necesario o el Distribuidor ya existía, el servicio devolverá el formulario con los datos válidos llenos.

#### **Excepciones**

Fallas de comunicación.

Decisión por parte del Administrador de interrumpir alguna actividad.

#### **Recursos especiales utilizados**

Base de datos: Para acceder a un Distribuidor.

GUI'S Requeridas:

GUI\_Gestión\_Distribuidor

GUI\_Formulario\_Distribuidor

## **Caso de Uso Escoger categoría 12**

**Iniciador:** Visitante

**Precondición:** ninguna.

### **Flujo de eventos**

#### **Flujo principal**

1. El visitante accede al Centro Comercial Virtual a través de la URL de Tampu
2. El sistema muestra los logos de cada una de las tiendas que se encuentren suscritas al servicio y los vínculos a las diferentes opciones de este como buscar un producto o escoger una categoría.
3. El visitante escoge una de las categorías de productos
4. El servicio le muestra los diferentes productos o subcategorías.

#### **Postcondiciones**

Visitante con posibilidad de seguir navegando a través del servicio y eligiendo productos para conocer.

#### **Flujos alternativos**

En 1 También se puede inicializar este caso de uso en el interior de una de las tiendas

#### **Excepciones**

Fallas de comunicación.  
Decisión por parte del Visitante de interrumpir alguna actividad.

#### **Recursos especiales utilizados**

Base de datos: Para extraer el árbol de categorías.

#### **GUI'S Requeridas:**

GUI\_Principal

GUI\_Tienda

GUI\_Resultado\_Consulta

### **Caso de Uso Identificar Registrado 13**

**Iniciador:** Registrado

**Precondición:** ninguna.

#### **Flujo de eventos**

#### **Flujo principal**

1. Durante el recorrido del Centro, el servicio le presenta al visitante la oportunidad de registrarse o identificarse como Registrado.
2. El visitante introduce su login y password en las casillas destinadas para ello.
3. El servicio valida el Registrado y se inicia una sesión de Registrado ubicándolo en el Formulario de Registrado para corroborar la información personal.

#### **Postcondiciones**

Registrado con posibilidad de seguir navegando a través del servicio y eligiendo productos para conocer.

#### **Excepciones**

Fallas de comunicación.

Decisión por parte del Registrado de interrumpir alguna actividad.

#### **Recursos especiales utilizados**

Base de datos: Para comparar datos de usuario

#### **GUI'S Requeridas:**

GUI\_Identificacion\_Registrado

### **Caso de Uso Gestionar Fletes 14**

**Iniciador:** Distribuidor

**Precondición:** Esta validado dentro del servicio como Distribuidor

#### **Flujo de eventos**

#### **Flujo principal**

1. El Distribuidor solicita al servicio gestionar fletes
2. El servicio le entrega un listado con las diferentes ciudades y países.
3. El Distribuidor elige un país o ciudad
4. El servicio muestra el valor del flete y la oportunidad de modificarlo
5. El Distribuidor modifica el valor del flete y acepta los cambios
6. El servicio modifica la base de datos y entrega al Distribuidor el mensaje correspondiente de actualización de los datos.

#### **Postcondiciones**

Distribuidor con posibilidad de seguir accediendo a las opciones de Distribuidor.

Base de datos actualizada.

#### **Flujos alternativos**

En 5 el Distribuidor puede no modificar los datos y simplemente salir a buscar otro país o ciudad.

#### **Excepciones**

Fallas de comunicación.

Decisión por parte del Distribuidor de interrumpir alguna actividad.

#### **Recursos especiales utilizados**

Base de datos: Para acceder y modificar los valores de los fletes.

#### **GUI'S Requeridas:**

GUI\_Sesion\_Distribuidor

GUI\_Fletes

### **Caso de Uso Gestionar Registrados 15**

**Iniciador:** Administrador

**Precondición:** Esta validado dentro del servicio como Administrador

#### **Flujo de eventos**

#### **Flujo principal**

1. El Administrador solicita al servicio gestionar Registrados
2. El servicio le entrega un listado con los Registrados actuales con las opciones ver detalles y eliminar registro.
3. El Administrador elimina del servicio al Registrado que desee
4. El servicio envía un correo al Registrado eliminado
5. El servicio modifica la base de datos
6. El servicio entrega al Administrador el mensaje correspondiente de actualización de los datos.

#### **Postcondiciones**

Administrador con posibilidad de seguir accediendo a las opciones de Administrador.  
Base de datos actualizada.

#### **Flujos alternativos**

En 3 el Administrador puede no eliminar Registrado alguno.

#### **Excepciones**

Fallas de comunicación.  
Decisión por parte del Administrador de interrumpir alguna actividad.

#### **Recursos especiales utilizados**

Base de datos: Para acceder y modificar los datos.

GUI'S Requeridas:

GUI\_Sesion\_Administrador

GUI\_Registrados

GUI\_Informacion\_Cliente

### **Caso de Uso Gestionar Categorías 16**

**Iniciador:** Administrador

**Precondición:** Administrador válido dentro del servicio

#### **Flujo de eventos**

#### **Flujo principal**

1. El Administrador solicita al servicio gestionar Categorías
2. El servicio le entrega un listado con los nombres de las categorías y la opción de eliminar o adicionar una nueva categoría
3. El Administrador adiciona una nueva categoría
4. El servicio modifica la base de datos
5. El servicio entrega al Administrador el mensaje correspondiente de actualización de los datos.

#### **Postcondiciones**

Administrador con posibilidad de seguir accediendo a las opciones de Administrador.  
Base de datos actualizada.

#### **Flujos alternativos**

En 3 el Administrador puede eliminar una categoría. Si esta tiene productos asociados no podrá ser eliminada.

#### **Excepciones**

#### **Fallas de comunicación.**

Decisión por parte del Administrador de interrumpir alguna actividad.

#### **Recursos especiales utilizados**

Base de datos: Para acceder y modificar los datos.

GUI'S Requeridas:

GUI\_Sesion\_Administrador

GUI\_Gestionar\_Categorias

### **Caso de Uso Revisar Entregas 17**

**Iniciador:** Distribuidor

**Precondición:** Distribuidor válido dentro del servicio

#### **Flujo de eventos**

##### **Flujo principal**

1. El Distribuidor solicita al servicio revisar las entregas
2. El servicio le entrega un listado con los últimos pedidos, señalando los nuevos y la oportunidad de marcarlos como despachados
3. El Distribuidor marca los pedidos despachados
4. El servicio modifica la base de datos
5. El servicio entrega al Distribuidor el mensaje correspondiente de actualización de los datos

##### **Postcondiciones**

Ninguna

##### **Flujos alternativos**

Ninguno

##### **Excepciones**

Fallas de comunicación.  
Decisión por parte del Distribuidor de interrumpir alguna actividad.

##### **Recursos especiales utilizados**

Base de datos: Para acceder y modificar los datos.

GUI'S Requeridas:  
GUI\_Sesion\_Distribuidor

### **Caso de Uso Hacer pedido 18**

**Iniciador:** Registrado

**Precondición:** El Registrado con al menos un producto en su orden de compra

#### **Flujo de eventos**

##### **Flujo principal**

1. El Registrado decide hacer el pedido de la orden de compra que ha gestionado
2. El servicio entrega el formulario con la información del Registrado con el fin que este la valide
3. El Registrado cambia y/o valida la información
4. El servicio pide al Registrado la información concerniente al pago
5. El Registrado paga con tarjeta de crédito
6. El servicio se comunica con el banco para que este valide la información del Registrado
7. El servicio actualiza la base de datos.
8. El servicio entrega al Registrado el mensaje correspondiente a que se ha efectuado exitosamente la transacción.

##### **Postcondiciones**

Ninguna

##### **Flujos alternativos**

En 2, es posible que sea un Visitante quien quiera utilizar este caso de uso, entonces se le presentará el formulario vacío para que lo llene con sus datos personales y quede Registrado en el servicio.

En 4, el Registrado desea pagar en la forma contra-entrega.

En 6, el banco no valida los datos, se le entrega al Registrado el mensaje indicando que debe reescribir la información de pago.

##### **Excepciones**

Fallas de comunicación.

Decisión por parte del Registrado de interrumpir alguna actividad.

##### **Recursos especiales utilizados**

Base de datos: Para acceder y modificar los datos.

Servicio de información del Banco.

##### **GUI'S Requeridas:**

GUI\_Pago

### **Caso de Uso Revisar Pedidos 19**

**Iniciador:** Suscriptor

**Precondición:** Suscriptor válido dentro del servicio

**Flujo de eventos**

**Flujo principal**

1. Una vez realizado el pedido el servicio entrega al Suscriptor la notificación de nuevo pedido con toda la información necesaria.
2. El Suscriptor autoriza la entrega

**Postcondiciones**

Ninguna.

**Flujos alternativos**

Ninguno.

**Excepciones**

Fallas de comunicación.  
Decisión por parte del Suscriptor de interrumpir alguna actividad.

**Recursos especiales utilizados**

Base de datos: Para acceder y modificar los datos.

**GUI'S Requeridas:**

GUI\_Pedidos

## **Caso de Uso Ver Ofertas 20**

**Iniciador:** Visitante

**Precondición:** Ninguna

### **Flujo de eventos**

#### **Flujo principal**

1. Una vez el Visitante acceda al servicio, este entrega al interfaz con una las últimas ofertas.
2. El Visitante decide conocer la oferta y se inicializa el caso de uso conocer producto o conocer la Tienda que ofrece el producto en oferta y se inicializa el caso de uso visitar tienda.

#### **Postcondiciones**

Ninguna

#### **Flujos alternativos**

En 2, El Visitante decide no conocer la oferta presentada y continúa navegando por el sitio.

#### **Excepciones**

Fallas de comunicación.

#### **Recursos especiales utilizados**

Base de datos: Para acceder y modificar los datos.

#### **GUI'S Requeridas:**

GUI\_Oferta

## **Caso de Uso Gestionar Escogencias Gráficas 22**

**Iniciador:** Suscriptor

**Precondición:** ser Suscriptor válido dentro del Servicio

### **Flujo de eventos**

#### **Flujo principal**

1. El Distribuidor se encuentra en el menú principal de Distribuidor y decide editar o adicionar una escogencia.
2. El servicio despliega una interfaz con un listado de las escogencias gráficas con las opciones de modificar, eliminar y adicionar una nueva escogencia
3. El Suscriptor escoge la opción modificar Escogencia
4. El servicio presenta los campos para modificar la información de las escogencia, estos son, nombre e imagen.
5. El Suscriptor acepta los datos
6. El servicio valida y/o actualiza la información de la escogencia.

#### **Postcondiciones**

Sesión de Distribuidor iniciada.

#### **Flujos alternativos**

En 3 El suscriptor puede elegir agregar una nueva escogencia, caso en el cual se le presentará el mismo formulario pero con los campos vacíos.

#### **Excepciones**

Fallos de comunicación.  
Abandono del proceso por parte del Distribuidor.

#### **Recursos especiales utilizados**

Base de datos de usuarios: Consulta para verificación.

#### **GUI'S Requeridas:**

GUI\_Gestionar\_Escogencias

### **Caso de Uso Identificar Distribuidor 23**

**Iniciador:** Distribuidor

**Precondición:** ninguna.

#### **Flujo de eventos**

##### **Flujo principal**

1. El Distribuidor accede a la página de Identificación del Distribuidor.
2. El Distribuidor se identifica con su login y password e indica al servicio que ha ingresado los datos.
3. El servicio valida el Distribuidor y de ser válido le permite ingresar a la sesión Distribuidor

##### **Postcondiciones**

Sesión de Distribuidor iniciada.

##### **Flujos alternativos**

En 3 puede haber variación debido a la validación del Distribuidor, por que al no reconocerlo como valido le reenvía una pagina solicitándole el login y password de nuevo.

##### **Excepciones**

Fallos de comunicación.  
Abandono del proceso por parte del Distribuidor.

##### **Recursos especiales utilizados**

Base de datos de usuarios: Consulta para verificación.

##### **GUI'S Requeridas:**

GUI\_IU\_Identificacion\_Distribuidor  
GUI\_Sesion\_Distribuidor

## 2. Bosquejo de las interfaces gráficas de usuario.

Con el fin de explicar mejor el concepto de Escogencia gráfica se presentan las siguientes interfases:

### Editar Escogencia

---

ID Escogencia: 13

Nombre:

Imagen: 

### Gestionar Escogencia Gráfica

Categoría:

Especificaciones:

Palabras claves:

Escogencia:

Opción 1:

### Gestionar Producto

• [Todo](#)  
• [Ropa](#)  
• [Jovias](#)  
• [Alimentos](#)  
• [Servicios](#)  
• [Viajes](#)  
• [Artesanías](#)  
• [Calzado](#)

**S**  
**E**  
**D**  
**A**

Para toda ocasión, le saluda con estos zapatos vaqueros que ya no usa, y lo mejor de todo es que es completamente natural, no se mató ni un animalito para que usted pudiera verse 100% elegante. Se puede encontrar en color caqui, blanco y anaranjado, con diseños diferentes como malla, estrella, entre otros, esta es una oportunidad que usted no puede desperdiciar, además por este mes de octubre está por mitad de precio.

Comprelo YA!!!

**Especificaciones:** Bufanda de seda de excelente calidad. Diseños y colores surtidos.  
**Peso:** 1 Kilos por cada 4 unidades

**Advertencia:**  
Se hará el despacho de mercancía solo para compras mayores de 24 unidades

\$10.00 cada uno

**Opciones de Bufanda de Seda**

Escogencia:

malla1       malla2

mb5-06       rs3-05

Otras:      color:

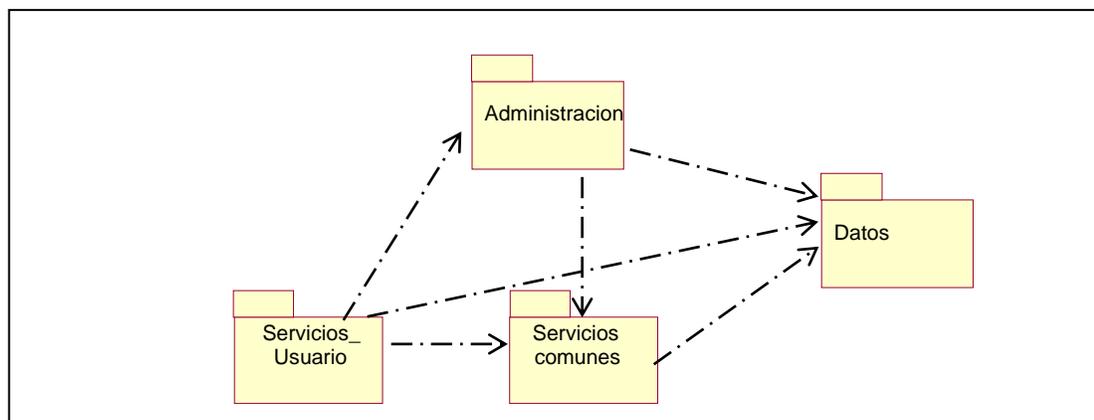
[Regresar](#)      [Agregar al Carrito](#)

Conocer Producto

## Subproducto Esencial No. 1B: Modelo de Análisis del Servicio

### Descripción de Paquetes y Clases de Análisis del Servicio

#### 1.1 Diagrama de Paquetes de Análisis esenciales



## 1.2 Descripción de las clases de Análisis esenciales

### Clases de Análisis de Tipo Control

#### ControlTiendas

**Responsabilidades:**

Adquirir los logotipos e imágenes de la base de datos de cada empresa para desplegarlas en la interfaz GUI\_Principal.

Se encarga de invocar la página de entrada de cada tienda.

#### ControlPedidos

**Responsabilidades:**

Hacer los cálculos correspondientes al valor total y parcial de los pedidos.

Hacer los cálculos correspondientes al valor agregado por concepto de fletes.

Modificar los registros de la base de datos de sus pedidos respectivos.

#### ControlEncabezado

**Responsabilidades:**

Desplegar la información necesaria para que el Visitante acceda a todas los diferentes servicios que ofrece el Centro.

#### ControlPagos

**Responsabilidades:**

Invocar la interfaz GUI\_Pago

Realizar las operaciones correspondientes a cada forma de pago.

Si el pago se efectúa con tarjeta de Crédito, esta clase se encarga de conectarse con la base de datos del banco utilizando los protocolos de seguridad requeridos, hacer el depósito y modificar la base de datos. Si el número de la tarjeta es errado, se encarga de entregar el mensaje correspondiente.

#### ControlIngresoDistribuidor

**Responsabilidades:**

Validar el Ingreso del Distribuidor al Servicio.

Invocar el despliegue de la Interfaz GUI\_Sesion\_Distribuidor si el ingreso es Valido.

Invocar el despliegue de un mensaje de error de no ser así.

#### ControlIngresoRegistrado

**Responsabilidades:**

Verifica si el usuario está registrado o no en el servicio (chequeo).

Validar el Ingreso de los Registrados al Servicio.

Invocar el despliegue de la Interfaz GUI\_Formulario\_Registrado con el fin de que confirme sus datos.

<b>ControlIngresoAdministrador</b>
<b>Responsabilidades:</b> Validar el Ingreso del Administrador al Servicio. Invocar el despliegue de la Interfaz GUI_Sesion_Administrador.
<b>ControlIngresoSuscriptor</b>
<b>Responsabilidades:</b> Validar el Ingreso de los Suscriptores al Servicio. Invocar el despliegue de la Interfaz GUI_Sesion_Suscriptor
<b>ControlVistaProductos</b>
<b>Responsabilidades:</b> Se encarga de la visualización del producto y las características de este, extrayendo esta información de la base de datos, generando así la interfaz GUI_Producto
<b>GestionBusquedas</b>
<b>Responsabilidades:</b> Es la clase encargada de controlar todo el proceso relacionado con la búsquedas de productos, toma los parámetros entregados por la clase GUI_Consultas realiza la búsqueda consultando sobre la clase TarjetaProducto y los despliega utilizando GUI_Resultado_Consulta.
<b>GestionRegistro</b>
<b>Responsabilidades:</b> Gestionar el proceso de inclusión o modificación de la información de un nuevo Registrado. Transporta la información de la base de datos para desplegarla en la interfaz GUI_Formulario_Registrado Revisar y validar los datos provenientes de la Clase GUI_Formulario_Registrado. Insertar dichos datos en la clase Tarjeta_Registrado Desplegar los diferentes mensajes de error cuando hayan ocurrido.
<b>GestionRegistrados</b>
<b>Responsabilidades:</b> Consulta la Tarjeta_Registrado para desplegar la lista de los Registrados en el servicio invocando la clase GUI_Registrados Realizar las funciones de Borrado de un Registrado en el servicio. Modificar la Tarjeta_Registrado una vez el Administrador acepte los cambios.
<b>GestionProducto</b>
<b>Responsabilidades:</b> Realizar las funciones de Suscriptor de Edición, Borrado o Inclusión de un nuevo producto en el servicio. Se encarga de desplegar el formulario de inclusión de un nuevo producto, interfaz GUI_Formulario_Producto. Validar dichos datos, verificar si son coherentes y que se han llenado los campos requeridos Incluir la nueva información de la base de datos.

#### **GestionFletes**

**Responsabilidades:**

Realizar las funciones de modificación o Inclusión de precios de fletes  
Hacer la modificación de la base de datos.

#### **GestionCategorias**

**Responsabilidades:**

Realizar las funciones de eliminar o Incluir una nueva categoría al servicio.  
Invocar la interfaz GUI\_Gestionar\_Categorías que presenta una lista de categorías al Administrador.  
Hacer la modificación de la base de datos.

#### **GestionEscogencias**

**Responsabilidades:**

Realizar las funciones de eliminar o Incluir una nueva escogencia gráfica.  
Invocar la interfaz GUI\_Gestionar\_Escogencias que presenta una lista de escogencias al Suscriptor.  
Hacer la modificación de la base de datos.

#### **GestionOrden**

**Responsabilidades:**

Realizar las funciones de incluir los productos en la Tarjeta\_Orden a medida que el visitante solicita, invocar la interfaz GUI\_Orden y realizar los cálculos correspondientes a las cuentas total y parcial de la orden.

#### **GestionSuscriptores**

**Responsabilidades:**

Realizar las funciones de Administrador de Edición, Borrado o Inclusión de un nuevo Suscriptor (tienda) en el servicio.  
Se encarga de desplegar el formulario de inclusión de un nuevo suscriptor, interfaz GUI\_Formulario\_Suscriptor.  
Validar dichos datos, verificar si son coherentes y que se han llenado los campos requeridos  
Incluir la nueva información de la base de datos.

#### **ControlMensajes**

**Responsabilidades:**

Tiene como principal función la de desplegar los mensajes de error o confirmación de eventos realizados por cada uno de los actores.

#### **Clases de Análisis de Tipo Frontera**

#### **GUI\_Consultas**

**Responsabilidades:**

Presenta el formulario que permite hacer la búsqueda de un producto.  
Se relaciona con las clases, Tarjeta\_producto y GestionBusquedas.

#### **GUI\_Resultado\_Consulta**

**Responsabilidades:**

Presenta los resultados de la consulta en una lista de productos con sus respectivas imágenes, empresas que los ofrecen vínculos.  
Se relaciona con las clases, Tarjeta\_producto y GestionBusquedas.

#### **GUI\_Identificacion\_Registrado**

**Responsabilidades:**

Proveer al Registrado de la interfaz que contiene los campos para que este ingrese su login, password y a partir de esto ingresar como Registrado al servicio.  
Se relaciona con ControlIngresoSuscriptor a la cual pasa los parámetros capturados de login y password.

#### **GUI\_Identificacion\_Suscriptor**

**Responsabilidades:**

Proveer al Suscriptor de la interfaz que contiene los campos para que este ingrese su login y password y a partir de esto ingresar al servicio.  
Se relaciona con ControlIngresoSuscriptor a la cual pasa los parámetros capturados de login y password.

#### **GUI\_Identificacion\_Administrador**

**Responsabilidades:**

Proveer al Administrador de la interfaz que contiene los campos para que este ingrese su login y password y a partir de esto ingresar al servicio.  
Se relaciona con ControlIngresoAdministrador a la cual pasa los parámetros capturados de login y password.

#### **GUI\_formulario\_Suscriptor**

**Responsabilidades:**

Proveer de un formulario al Administrador donde este pueda ingresar o editar la información de un Suscriptor (tienda).  
Se relaciona con la clase GestiónSuscriptores y Tarjeta\_Suscriptor.

#### **GUI\_Formulario\_Producto**

**Responsabilidades:**

Proveer de un formulario al Suscriptor donde este pueda ingresar o modificar la información de un producto que ofrezca en su tienda.  
Se relaciona con la clase GestiónProducto. Que contiene los métodos que permitan hacer estas consultas y modificaciones en la Tarjeta\_producto.

#### **GUI\_Formulario\_Registrado**

**Responsabilidades**

Proveer de un formulario al Visitante o Registrado donde este pueda registrarse o modificar su información personal, respectivamente.  
Se relaciona con la clase GestiónRegistro. Que contiene los métodos que permitan hacer estas consultas y modificaciones en la Tarjeta\_Registrado.

#### **GUI\_Sesion\_Administrador**

**Responsabilidades:**

Proveer de un menú de opciones de Administrador.

#### **GUI\_Sesion\_Suscriptor**

**Responsabilidades:**

Proveer de un menú de opciones de Suscriptor.

#### **GUI\_Sesion\_Distribuidor**

**Responsabilidades:**

Proveer de un menú de opciones de Distribuidor.

#### **GUI\_Producto**

**Responsabilidades:**

Interfaz que presenta las características de producto como nombre, imagen, empresa, descripción, peso, precio, etc.

Se relaciona con las clases Tarjeta\_producto y ControlVistaProducto.

#### **GUI\_Tienda**

**Responsabilidades:**

Despliega la información de la tienda con los productos asociados a ella.

Se relaciona con ControlTiendas, Tarjeta\_Suscriptor y Tarjeta\_productos.

#### **GUI\_Principal**

**Responsabilidades:**

Es la interfaz que le muestra al visitante las diferentes opciones y facilidades del servicio, como ingreso a las categorías y tiendas.

Se relaciona con la clase ControlEncabezado

#### **GUI\_Orden**

**Responsabilidades:**

Entregar una lista con los productos solicitados por el visitante, con los valores parcial y total.

Se relaciona con las clases GestionOrden y Tarjeta\_Orden.

#### **GUI\_Pago**

Es la interfaz encargada de entregar las opciones de las diferentes formas de pago, el formulario necesario para efectuar la cancelación por tarjeta de crédito.

Se relaciona con la clase

### **GUI\_Gestión\_Suscriptor**

**Responsabilidades:**

Es la interfaz encargada de desplegar el formulario para agregar o modificar la información de un Suscriptor.  
Se relaciona con las clases GestionSuscriptores y Tarjeta\_Suscriptor.

### **GUI\_Fletes**

**Responsabilidades:**

Es la clase encargada de proporcionar la lista de los diferentes países y regiones con su respectivo valor de gastos de envío y con las opciones correspondientes para modificar dichos valores o agregar un nuevo lugar.  
Esta relacionada con GestionFletes y Tarjeta\_fletes.

### **GUI\_Registrados**

**Responsabilidades:**

Presenta un listado con los Registrados actuales y la opción de eliminarlos del servicio  
Se relaciona con las clases GestiónRegistrados y Tarjeta\_Registrado

### **GUI\_Gestionar\_Categorías**

**Responsabilidades:**

Presentar al Administrador una lista de las categorías existentes y la oportunidad de eliminar alguna o agregar una nueva.  
Se relaciona con las clases de GestionCategorías y Tarjeta\_categoria

### **Clases de Análisis de Tipo Entidad**

#### **Tarjeta\_Registrado**

**Responsabilidades:**

Es el almacén de información que resguarda los datos de los Registrado, su contraseña, su información de contacto, etc. es utilizada por GestionRegistro para inclusión u modificación de la información por parte del usuario, también es usada por GestionRegistros que tiene métodos que trabajan a más bajo nivel, es decir directamente con la base de datos, para listar los Registrados existentes y eliminar registros. También es usada por ControlIngresoRegistrado que valida los datos de ingreso al servicio.

#### **Tarjeta\_categoria**

**Responsabilidades:**

Contiene la lista de las categorías. Es utilizada por ControlCategoría para desplegarlas.

#### **Tarjeta\_producto**

**Responsabilidades:**

Esta clase se encarga de almacenar los datos relacionados con los productos. Se relaciona con ControlTiendas, ControlPedidos, ControlEntregas, ControlVistaProducto, estionBusquedas y GestionProducto.

#### **Tarjeta\_Suscriptor**

**Responsabilidades:**

Almacena los nombres y datos de cada empresa suscripta al servicio. Es utilizada por ControlIngresoSuscriptor para validar su entrada al servicio y por GestionSuscriptores

#### **5. Tarjeta\_Orden**

**Responsabilidades:**

Esta clase se encarga de almacenar la información pertinente a los pedidos realizados por los Registrados

#### **5. Tarjeta\_Preferencias**

**Responsabilidades:**

Almacena la lista de las preferencias que el usuario puede seleccionar en el momento de su registro.

### **1.3 Relación de las clases contenidas en los diferentes paquetes**

#### **Administración**

GestionCategoría

GestionProducto

GestionFletes

GestionSuscriptores

GestionRegistrados

GestionProducto

GUI\_Gestión\_Suscriptor

GUI\_Fletes

GUI\_Registrados

GUI\_Gestionar\_Categorías

GUI\_Sesion\_Administrador

GUI\_Sesion\_Suscriptor

GUI\_Sesion\_Distribuidor

GUI\_formulario\_Suscriptor

GUI\_Formulario\_Producto

GUI\_Identificacion\_Suscriptor

GUI\_Identificacion\_Administrador

### **Servicios de usuario**

GestionBusquedas  
ControlTiendas  
ControlPedidos  
GestionOrden  
ControlPagos  
ControlIngresoRegistrado  
ControlVistaProducto  
GestionRegistro  
GUI\_Consultas  
GUI\_Resultado\_Consulta  
GUI\_Identificacion\_Registrado  
GUI\_Formulario\_Registrado  
GUI\_Producto  
GUI\_Tienda  
GUI\_Principal  
GUI\_Orden  
GUI\_Pago  
GUI\_Pedidos  
GUI\_Oferta

### **Servicios Comunes**

ControlEncabezado  
ControlIngresoAdministrador  
ControlIngresoSuscriptor  
ControlMensajes

### **Datos**

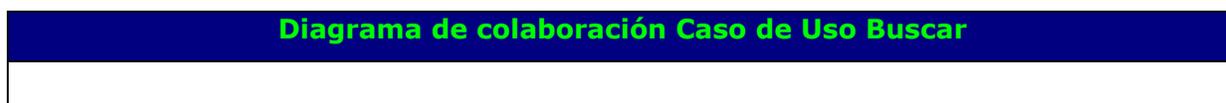
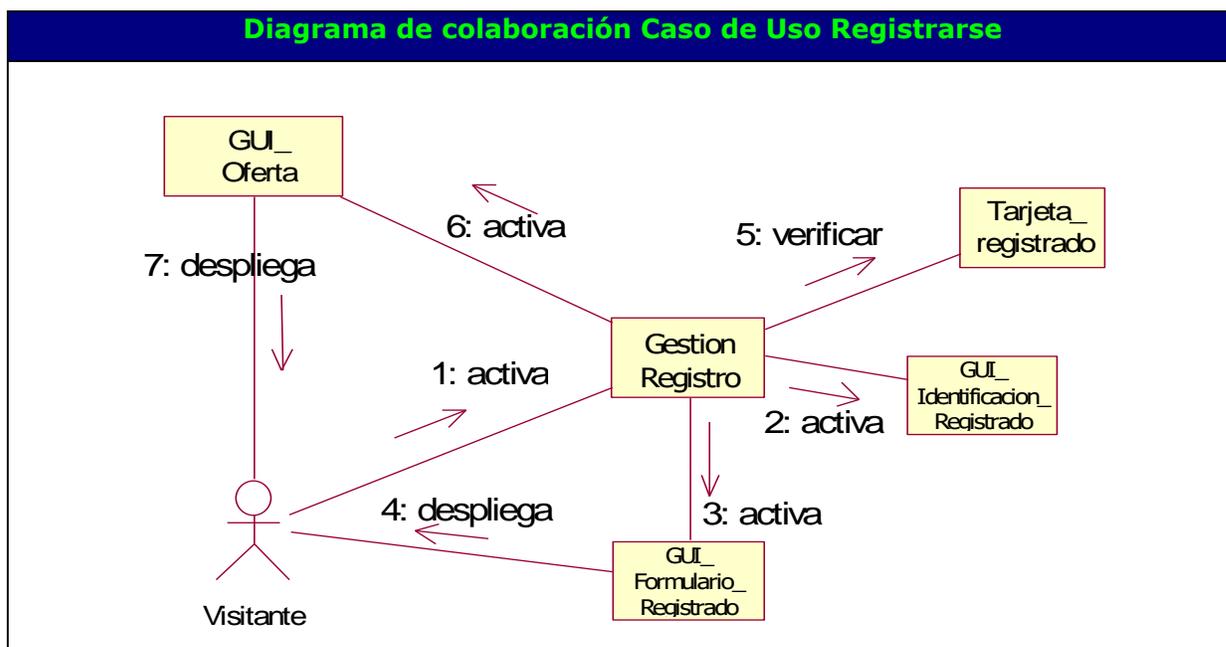
Tarjeta\_registrado  
Tarjeta\_Administrador  
Tarjeta\_categoria  
Tarjeta\_producto  
Tarjeta\_suscriptor  
Tarjeta\_Orden  
Tarjeta\_fletes

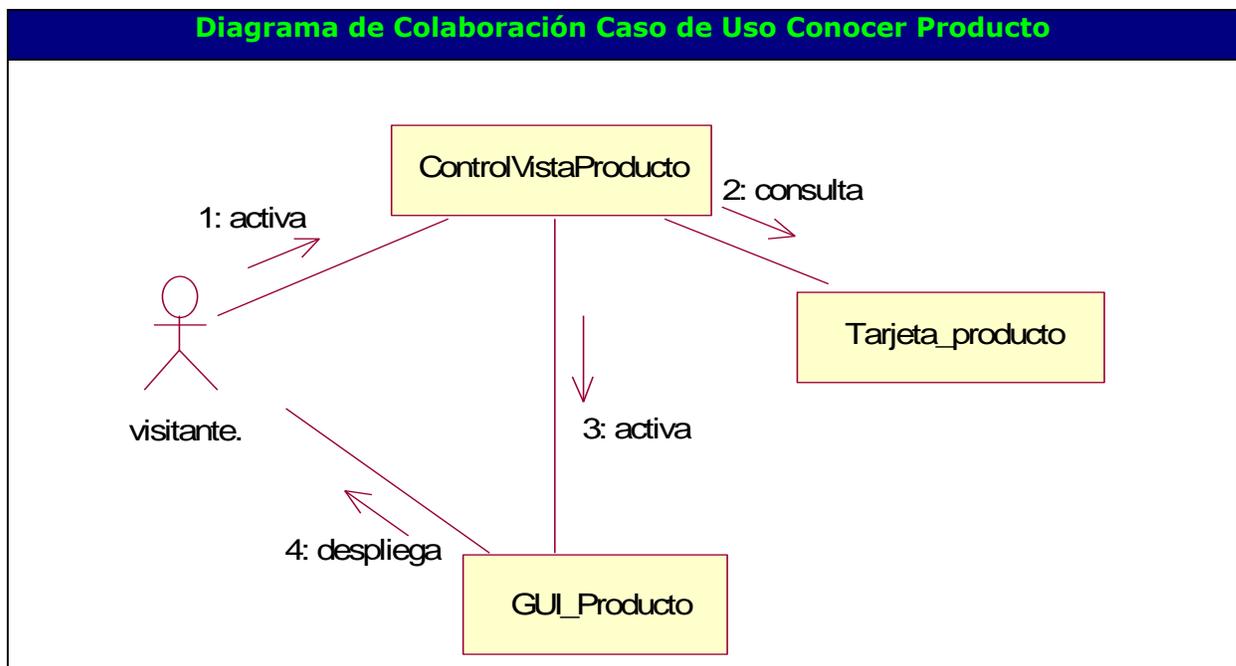
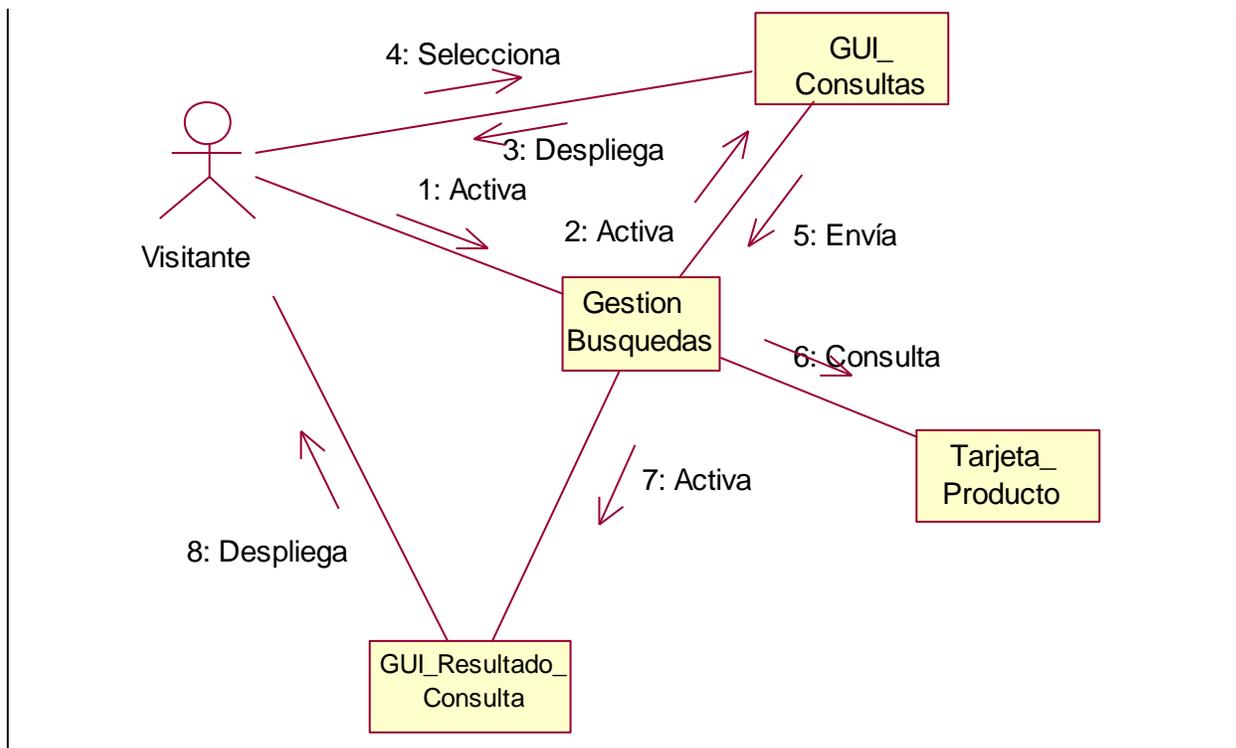
### Diagramas de Colaboración para los Casos de Uso del Servicio

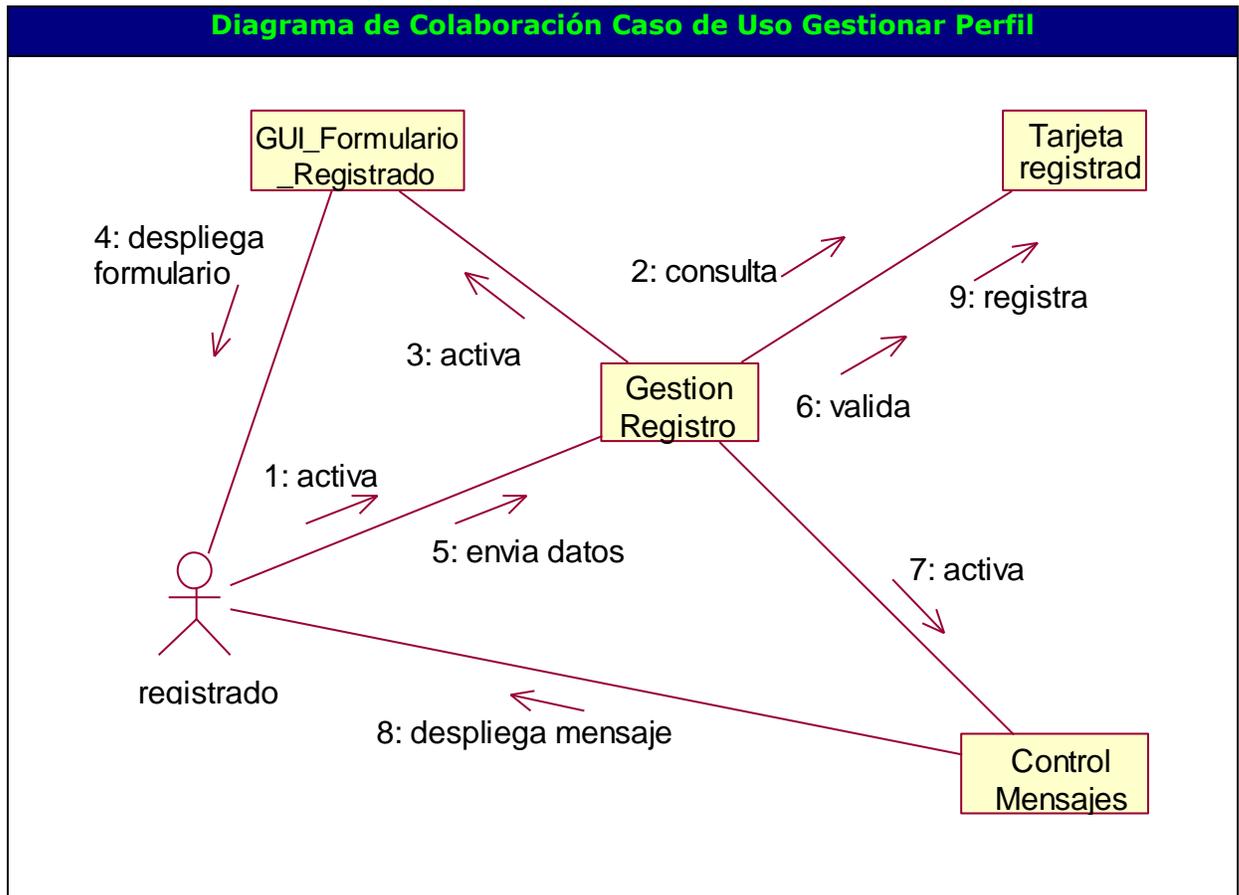
Estos diagramas se pueden revisar en el Anexo 4.

Los diagramas mostrados a continuación están relacionados con los siguientes casos de uso:

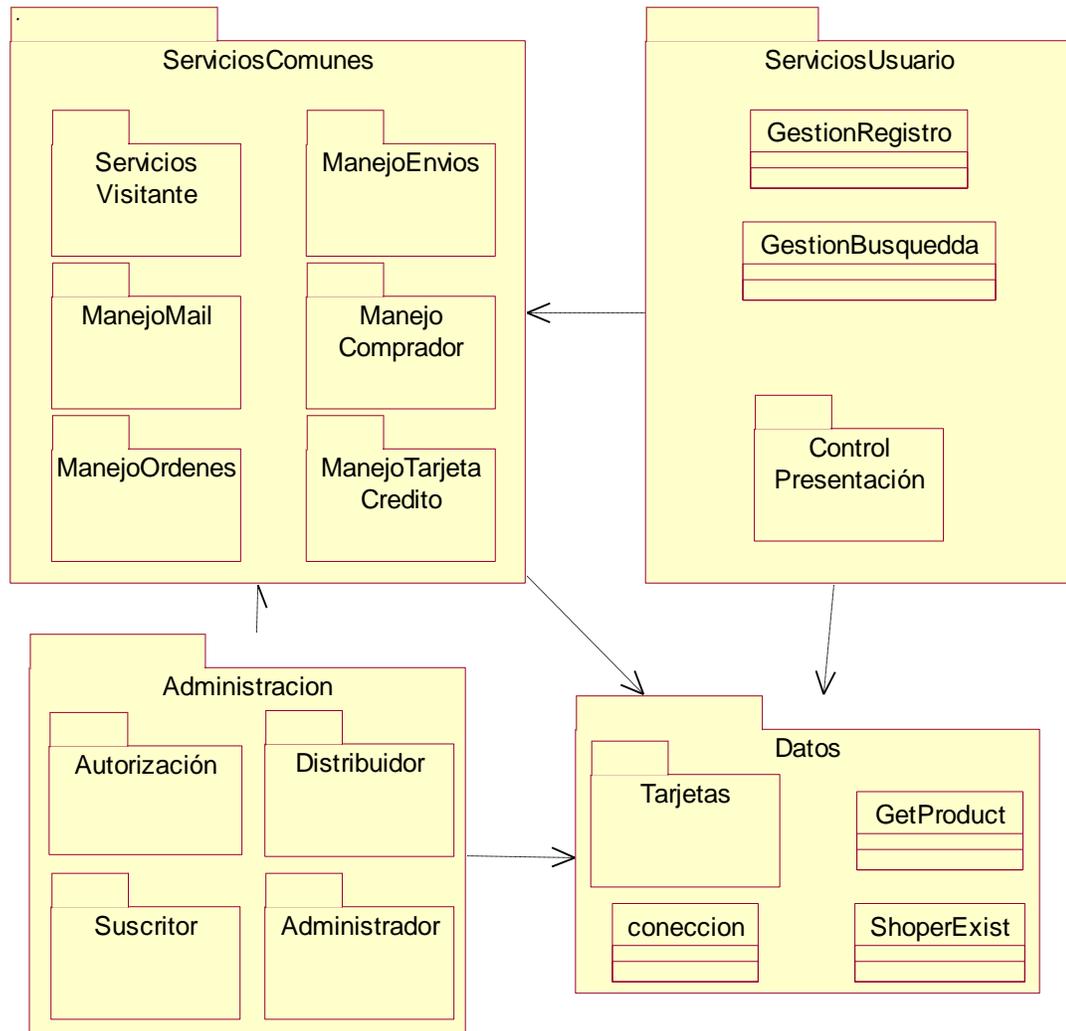
- Caso de Uso Registrarse
- Caso de Uso Buscar
- Conocer producto
- Gestionar Perfil







### Diagrama de Clases de Análisis del Servicio



## Subproducto Esencial No. 1C: Modelo de Diseño del Servicio

### 1.1 Funcionalidad de Usuario

#### 1.1.1. Descripción de Clases de Diseño del Servicio

A continuación se realiza una descripción de las clases de diseño que intervienen en la funcionalidad de usuario.

Posteriormente se establece como estas clases interactúan según cada caso de uso y adicionalmente se explican los mecanismos de seguridad propuestos.

La plataforma sobre la que se trabaja está orientada a desarrollar la lógica del lado del servidor, razón por la cual el cliente no necesita un software especializado.

#### Páginas de Usuario

**specialFooter:** Genera el despliegue de las ofertas utilizando la función `reduceImagen` para mostrar una vista previa de los productos en oferta.

**getProduct:** extrae de la base de datos toda la información de la Tarjeta Producto a partir del código de producto, además extrae el nombre de las opciones elegidas por el usuario cuando ordena un producto para adicionarlo al carrito de compras.

**product:** Presenta la información del producto, utiliza diferentes funciones orientadas a la presentación de información `stringOptionName` y `viewImage`, presenta el formulario `productForm` encargado de mandar al carrito de compras la información necesaria para adicionar el producto.

**paginaPrincipal:** Presenta la entrada a las tiendas utilizando la función `tiendaExpo`, además inicia la sesión de visitante con la función `session_start`

*Caso de uso:* Todos en los que intervenga el usuario (Visitante / Registrado).

**account:** Incluye los formularios `dirForm` y `dirPagoForm` que presenta los campos para la información de usuario, se encarga de identificar si la sesión se usuario pertenece a un

Registrado en la base de datos, en cuyo caso entrega en el formulario la información del registrado. Valida los datos y los almacena una vez halla recibido la confirmación por parte del usuario para actualizar los datos del Registrado o crear uno nuevo.

**Order:** Presenta el carrito de compras y la posibilidad de gestionar la orden, necesita la función *remove\_product para eliminar una* y el formulario *orderForm*  
Caso de uso: Gestionar Orden.

**orderRetrieve:** Presenta la información de una orden realizada con anterioridad en forma de recibo donde detalla el valor y características de la orden en forma de recibo de compra.  
Caso de uso: Gestionar Orden.

**remove\_product:** Se encarga de eliminar un item de la orden  
Caso de uso: Gestionar Orden

**tienda:** Presenta la información de cada tienda, logo, nombre, descripción, y si los tiene, un vínculo al correo-e y página web. Utiliza la función *generarCategorias* y *listing* que lista los productos encontrados.  
Caso de uso: Visitar Tienda

**shopperCheck:** Pide la identificación del registrado o al visitante que se registre. Aparte del formulario *regForm* la función *shopperExists*, si esta retorna verdadero, el servicio lo reconoce como usuario Registrado re-diseccionándolo a una interfaz con los datos de Registrado llenos.  
Caso de uso: Hacer pedido.

**getPwd** Permite enviar la clave correspondiente al registrado a partir del correo suministrado

**login:** Valida los datos de identificación de registrado. Si no son válidos devuelve el formulario *regForm* con un mensaje de error.

**shipAddressInfo:** Presenta el formulario que contiene la información del envío, El registrado debe confirmar esta información para continuar el trámite de transacción, valida la información y actualiza las tablas en la BD.

**shipmethodinfo:** Entrega un formulario donde el registrado deberá escoger el método de pago.

**search:** Presenta un formulario de búsqueda avanzada.

**searchListing:** Identifica la naturaleza de la búsqueda, palabras, frases, coincidencia y aplica un algoritmo para solicitar la búsqueda bajo las condiciones establecidas por el usuario; para finalmente presentar los resultados de la búsqueda con la función productList. Si se ha elegido buscar Empresas, la página se redirecciona a searchSuscrip.

**searchSuscrip:** Aplica un algoritmo para solicitar la búsqueda bajo las condiciones establecidas por el usuario; para finalmente presentar los resultados de la búsqueda con la función tiendaExpo.

**listing:** Presenta el resultado de la búsqueda por categorías, utiliza la función productList.

<b>Funciones comunes</b>
--------------------------

**Conectar:** Se conecta con la base de datos realiza la consulta entregada, cierra la conexión y devuelve el resultado de la consulta.

**generarCategorias** Se encarga de extraer de la base de datos y presentar las categorías existente en el centro.

*Caso de uso:* Todos en los que intervenga el usuario (Visitante / Registrado).

**mailFuctions:** Presenta todos los métodos necesarios para formatear y enviar correos electrónicos.

**stringSelectBox:** Recupera la cadena de opciones que se han dado para el producto y las presenta en una caja de selección HTML

**viewImage:** Extrae el archivo de imagen de la base de datos y la presenta en la interfaz.

**reduceImage:** Entrega en una matriz un nuevo tamaño de imagen para aplicarlo cuando se desee dar una vista previa de esta.

**selectBox:** Construye una caja de selección HTML a partir de un resultado de consulta en la base de datos basado en los parámetros que se le proporciona.

**shopperExists:** Revisa si el ID de visitante, es decir el ID de la sesión abierta pertenece a un registrado en la base de datos

**getShopper:** Toma a partir del atributo ID de registrado la información del Registrado en la base de datos

**getProduct:** Toma la información necesaria de la base de datos a partir del código del producto para generar la Orden

**resultSetNav:** Función que presenta los resultados de una consulta en una lista grupos discretos de acuerdo a un número establecido por el administrador. Por ejemplo si el número establecido es 10, se presentarán las listas de diez en diez, si la consulta arroja más de diez resultados, la función se encarga de presentar el vínculo de adelante y/o atrás.

**header :** Como característica importante se tiene que debe dar un mismo formato a toda la información de usuario que se presente. Contiene los menús que el visitante tendrá a la mano durante su recorrido por el centro. Utiliza la función generarCategorias. Genera (si aún no existe) y recupera el identificador de la sesión de visitante con la función session\_id.

*Caso de uso:* Todos en los que intervenga el usuario (Visitante / Registrado).

**footer:** Permite visualizar en las páginas de usuario las ofertas de extrayendo las ultimas de estas desde la base de datos dependiendo del perfil del Registrado.

## Formularios

**dirForm:** Obtiene los datos la dirección donde desembocará el envío

<<input>> *shopper\_ship\_name*: Nombre de la persona que recibirá el envío  
<<input>> *shopper\_ship\_company*: Empresa de la persona que recibirá el envío  
<<input>> *shopper\_ship\_phone*: Número telefónico de la persona que recibirá el envío  
<<input>> *shopper\_ship\_fax*: Fax de la persona que recibirá el envío  
<<input>> *shopper\_ship\_addr\_1*: Primera línea de la dirección  
<<input>> *shopper\_ship\_addr\_2*: Segunda línea de la dirección  
<<input>> *shopper\_ship\_city*: Ciudad de la persona que recibirá el envío  
<<input>> *shopper\_ship\_country*: País de la persona que recibirá el envío  
<<submit>> *Enviar*: Envía los datos a ser procesados.

**dirPagoForm:** Obtiene los datos la dirección donde se pagará el envío

<<input>> *shopper\_bill\_email*: Nombre de la persona que pagará el envío:  
<<input>> *shopper\_bill\_name*: Empresa de la persona que pagará el envío:  
<<input>> *shopper\_bill\_company*: Número telefónico de la persona que pagará el envío:  
<<input>> *shopper\_bill\_phone*: Fax de la persona que pagará el envío:  
<<input>> *shopper\_bill\_fax*: Primera línea de la dirección :  
<<input>> *shopper\_bill\_addr\_1*: Segunda línea de la dirección:  
<<input>> *shopper\_bill\_addr\_2*: Ciudad de la persona que pagará el envío:  
<<input>> *shopper\_bill\_city*: País de la persona que pagará el envío:  
<<input>> *shopper\_bill\_country*: Envía los datos a ser procesados.:

**productForm:** Formulario presente en la página de GUI\_Producto, se encarga de capturar datos acerca de las características del producto, información necesaria para el caso de uso Gestionar Orden (carrito de compras).

### Atributos

<<hidden>> *eCommID* : Etiqueta que identifica el usuario  
<<hidden>> *product\_id* : Etiqueta que transporta el id del producto  
<<hidden>> *unitCost*: Valor unitario del producto

### Métodos

OpcionesProducto: Genera las cajas de selección HTML correspondientes a las diferentes opciones del producto configurables desde el caso de uso Gestionar Producto

<<submit>> *enviar*: Envía los datos a Order.

**Caso de uso:** Conocer producto.

**orderForm:** Formulario presente en la página de GUI\_Orden, se encarga de capturar la orden, para hacer la transacción.

#### **Atributos**

<<hidden>> *eCommID*: Etiqueta que identifica el usuario

<<hidden>> *product\_id*: Etiqueta que transporta el id del producto

<<input>> *quant\_n*: Caja de texto que captura la cantidad de unidades que el usuario desea del artículo n de la orden.

#### **Métodos**

<<submit>> *update*: Actualiza el subtotal de acuerdo a la cantidad de los artículos solicitada por el Visitante.

<<submit>> *checkout*: Envía los datos a shopperCheck

**Caso de uso:** Gestionar Orden.

**regForm:** Formulario que presenta los campos necesarios para registrarse o identificarse como registrado.

#### **Atributos**

<<hidden>> *eCommID*: Etiqueta que identifica la sesión del usuario

<<hidden>> *nextPage*: información que adquiere la página con fines de redireccionamiento

<<input>> *shopper\_email*: Donde el usuario introduce su correo electrónico

<<input>> *shopper\_password*: Donde el usuario introduce su password

<<input>> *confirm\_password*: Donde el visitante introduce su password de nuevo (en el caso que no esté registrado en el servicio)

#### **Métodos**

<<submit>> *enviar*: Envía los datos a login para ser procesados

**searchForm:** Presenta los campos para hacer una búsqueda avanzada.

#### **Atributos**

<<hidden>> *eCommID*: Etiqueta que identifica el usuario

<<input>> *sf*: Donde el usuario escribe la palabra que se desea encontrar

<<select>> *critério*: Con el fin de elegir si se busca una empresa o un producto (por defecto está en buscar todo)

<<radio>> *wm*: Para elegir si se desea buscar la palabra exacta o palabras parecidas

<<radio>> *wu*: Para elegir si se desea buscar todas o cualquiera de las palabras.

### **Métodos**

<<submit>> *buscar*: Envía los datos a searchListing para ser procesados

## **1.1.2 Descripción de Diagramas de Secuencia y clases para los Casos de Uso del Servicio**

### **Precondiciones:**

Durante todas las páginas las GUI utilizan la clase ControlEncabezado para formatear la página.

Cuando una clase tipo control necesita comunicarse con la Base de Datos utiliza la función conectar.

### **Caso de Uso Buscar**

Este caso de uso se inicia desde cualquier página de usuario o desde la interfaz *GUI\_Consultas*. El formulario de búsqueda, *searchForm* tiene el campo para ingresar el criterio, y entradas que especifican el tipo de búsqueda deseada, producto, empresa, coincidencia, palabras usadas. Desde el formulario de búsqueda se envían (submit) los datos a la página de servidor *GestionBusquedas*, ésta recibe el parámetro, arma y realiza la consulta en la *Tarjeta\_Producto* dependiendo del tipo de consulta y con los datos retornados genera la interfaz *Resultado\_Consulta*, que al igual que todas las páginas de usuario contiene el *header* y el listado de los records obtenidos en la consulta, utiliza la clase *resultSetNav* para mostrar los resultados dependiendo del número de records establecidos en el archivo de configuración.

### **Caso de Uso Visitar Tienda**

Se inicia al estar en *GUI\_Principal* o *GUI\_Resultado\_Consulta*, el usuario pide al servicio entrar en una tienda, *ControlTiendas* genera la interfaz *GUI\_Tienda* consultando la *Tarjeta Suscriptor* y *Tarjeta Categoría* y *Tarjeta Producto*, Mostrando así la información de

la tienda, las categorías que ésta maneja y una vista previa de los productos usando la función *reduceImage* dependiendo la categoría que el usuario halla seleccionado.

### Caso de Uso Gestionar Perfil

Se inicia cuando el Registrado decide acceder a revisar su cuenta de usuario, *ControlIngresoRegistrado* verifica si el usuario es Registrado capturando el identificador de sesión con la función *session\_id*; usando la función *shoperExist* hace la consulta a la base de datos. *GestionRegistro* genera la *GUI\_Formulario\_Registrado* que contiene el formulario *RegForm*; obtiene la información del Registrado a partir del id de Registrado mediante la función *getShoper*, con estos datos se llena *RegForm* para que el Registrado modifique los datos que desee, una vez el Registrado acepte los cambios *GestionRegistro* verifica que los datos sean válidos y los campos requeridos estén diligenciados, si es así se hace la modificación de los datos en la *tarjeta Registrado*.

### Caso de Uso Ver Ofertas

Este caso de uso se inicia cuando el Registrado decide conocer la oferta desplegada por *specialFooter* y se inicializa el caso de uso conocer producto.

### Caso de Uso Registrarse

Inicia desde *UI\_Identificacion\_Registrado* donde se solicita al Visitante, por medio del formulario *checkForm*, ingresar su correo electrónico y Password para con esto datos quedar registrado en la *Tarjeta Registrado* una vez *ControlIngresoRegistrado* halla verificado que el correo electrónico no esté registrado en el servicio. Se le entrega al nuevo Registrado la interfaz *GUI\_Formulario\_Registrado* que contiene el formulario *RegForm* donde aparecen los campos para incluir la información personal en la base de datos del servicio, junto a un botón de envío. El listado de preferencias es generado gracias a una consulta a la base de datos (*Tarjeta Preferencias*). Una vez llenado se envía (submit) a *GestionRegistro* que recibe todos los datos y los valida (que existan datos), actualiza la base de datos par finalmente crear la sesión de Registrado

### Escoger Categoría

*Header* genera la lista de categorías, el usuario hace click sobre una de ellas y se despliega la lista de tiendas pertenecientes a esta categoría.

### **Caso de Uso Conocer Producto**

Inicia en donde se encuentre ubicada una lista de productos ya sea de una tienda específica (*tienda*) o de un resultado de consulta (*searchListing*), se puede ejecutar alguno de los links ya sea desde la imagen asociada o desde el link en el texto que indica el nombre del producto, al ejecutarse, *ControlVistaProductos* despliega la visualización del producto y las características de este, extrayendo esta información de la base de datos, generando así la interfaz *GUI\_Producto*, utiliza *viewImage* y *opcionesProducto*.

### **Caso de Uso Gestionar Orden**

Se inicia cuando el visitante decide ver la lista de los productos que tiene en el carrito de compras, link que se presenta en todas las páginas de usuario gracias a *header*, o cuando decide ordenar algún producto desde la interfaz *GUI\_Producto*. Cuando se adiciona a la orden el primer producto, *GestionOrden* registra el objeto de la orden con fines de persistencia por medio de una variable de sesión llamada *order* utilizando la función *session\_register*. Para cerciorarnos que tenemos siempre la información más reciente de la orden se borra el caché desde la información de cabecera. Se usa la función *getProduct* que captura de la base de datos la información de cada producto en la orden con sus características escogidas por el Visitante, esta información se utiliza para generar la interfaz *GUI\_Orden* que tabula cada producto con sus características, opciones y tienda proveedora. Se utiliza la función *removeProduct* para eliminar un producto de la orden cuando el Visitante lo solicite. Se incluye en la interfaz el formulario *orderForm* con el cual cada producto cuenta con una caja de texto para modificar la cantidad (la cantidad por defecto es = 1). También se le mostrará al Registrado las órdenes efectuadas con anterioridad a través de la función *orderRetrieve*.

### **Caso de Uso Hacer Pedido**

Se inicia a partir de la interfaz *GUI\_Orden*, cuando el visitante decide hacer el pedido de todo lo reunido en el carrito de compras presiona el botón Hacer Pedido, *ControlPedidos* le presenta la el formulario *dirForm* con los datos de la dirección de envío para que se

confirman, se pide que elija un Distribuidor y un método de pago, (si es tarjeta de crédito debe introducir el número). Una vez aceptado (submit) se envía la información al Registrado, Suscriptor y Distribuidor por correo con la clase *mailFuctions*.

### Caso de Uso Identificar Registrado

Se inicia cuando el visitante requiera identificarse como Registrado en el servicio *GestionRegistro* presenta el formulario *checkForm* en donde el visitante debe ingresar su login, password, revisa si los campos están llenos y comprueba si los datos concuerdan con una fila en la base de datos, si es así se le da al Registrado su id de sesión para ser identificado en todas las páginas de usuario.

## 1.2 Funcionalidad de Administrador

### 1.2.1. Descripción de Subsistemas y Clases de Diseño del Servicio

#### Páginas de Usuario

**editListingSuscrip:** Presenta la lista de suscriptores dentro del formulario *GestionSuscriptores* para eliminar cada suscriptor o para actualizar los datos de este, hace la advertencia de eliminación y pide la confirmación de esta.

**EditListingShipper:** Presenta la lista de suscriptores dentro del formulario *GestionRegistros* para eliminar cada Distribuidor o para actualizar los datos de este.

**indice\_Administrador:** Presenta la lista de facilidades que el servicio le ofrece al Administrador, tales como: gestionar suscriptores distribuidores y categorías, se presenta en todas las páginas de Administrador.

**editSections:** Presenta una lista de las categorías dentro del formulario *GestionCategorias* que permite eliminar o cambiar el nombre de las categorías. Utiliza la función *resultSetNav*.

**addSuscrip:** Presenta el formulario *formSuscriptor* con el fin de adicionar un nuevo suscriptor (tienda), valida y almacena la información en la base de datos.

**editSuscrip:** Presenta el formulario *formSuscriptor* con los campos llenos de un Suscriptor con el fin de modificar la información, la valida y la almacena en la base de datos.

**addShipper:** Presenta el formulario *formDistribuidor* con el fin de adicionar un nuevo Distribuidor, valida y almacena la información en la base de datos.

**editSuscrip:** Presenta el formulario *formDistribuidor* con los campos llenos de un Distribuidor con el fin de modificar la información, la valida y la almacena en la base de datos.

**editShoppers:** Presenta el formulario *gestionRegistrados* con los campos llenos de un Distribuidor con el fin de modificar la información, la valida y la almacena en la base de datos.

### Funciones comunes

**authorize:** Presenta el formulario de identificación para el administrador, suscriptor o distribuidor, valida los datos comparándolos con la base de datos (previa encriptación), para finalmente crear una sesión de usuario que valide a estos mientras navegan por sus diferentes interfases de gestión, este debe ser un archivo que se incluya en todas las páginas de gestión.

**header:** Presenta el menú de Administrador en todas las páginas de Administrador.

### Clases de Formulario

**formchallenge:** Presenta los campos necesarios para realizar la identificación del Administrador.

#### Atributos

<<input>> *PHP\_AUTH\_USER\**: Login del Administrador

<<input>> *PHP\_AUTH\_PW\**: Password del Administrador

#### Métodos

<<submit>> *Aceptar*: Envía los datos para ser procesados.

**formSuscriptor**: Presenta los campos en los que se incluirá la información de un Suscriptor.

#### **Atributos**

<<input>> *suscrip\_name\**: Nombre de la empresa

<<input>> *suscrip\_image*: Logo

<<input>> *suscrip\_frase*: Frase corporativa de la empresa

<<input>> *suscrip\_desc*: Descripción de la empresa

<<input>> *categorías*: Las categorías con las que funcionará la tiendas

<<input>> *suscrip\_keywords*: Palabras claves que se usarán en la búsqueda

<<input>> *suscrip\_correo*: Correo electrónico

<<input>> *suscrip\_password*: Palabra de paso para entrar a la interfaz de suscriptor.

<<input>> *suscrip\_telefono*: Teléfono de contacto

<<input>> *suscrip\_web*: Página web de la empresa (alojada en otro servidor)

<< checkbox >> *suscrip\_active*: Por defecto está activado si no se activa no aparecerá en el servicio, pero se mantendrá en la base de datos.

#### **Métodos**

<<submit>> *Adicionar\_Suscriptor*: Envía los datos para ser procesados.

**gestionSuscriptores**: Presenta los campos necesarios para eliminar los suscriptores.

#### **Atributos**

<<checkbox>> *delItem[]*: Aparece para cada Suscriptor sustraído de la BD para que el Administrador seleccione cuales desea eliminar.

Nota: Para una mayor gestión de cada Suscriptor aparece un vínculo hacia el formulario de Suscriptor.

#### **Métodos**

<<submit>> *Eliminar\_items*: Elimina los items seleccionados una vez hecha la confirmación.

**gestionRegistrados**: Presenta los campos necesarios para eliminar los Registrados.

#### **Atributos**

<<checkbox>> *delItem[]*: Aparece para cada Registrado sustraído de la BD para que el Administrador seleccione cuales desea eliminar.

#### **Métodos**

<<submit>> *Eliminar\_items*: Elimina los items seleccionados una vez hecha la confirmación.

**gestionCategorias:** Presenta los campos necesarios para eliminar, renombrar o adicionar categorías.

#### **Atributos**

<<checkbox>> *delItem[]*: Aparece para cada categoría para que el Administrador seleccione cuáles desea eliminar.

<<input>> *new\_name[]*: Campo para colocar un nuevo nombre de la categoría.

<<input>> *new\_id[]*: Campo para colocar un nuevo id de la categoría.

<<input>> *NC*: Campo para agregar una nueva categoría.

Nota: Cambiar el id o eliminar una categoría puede ocasionar daños en el funcionamiento del Centro.

#### **Métodos**

<<submit>> *Aplicar cambios*: Elimina los items seleccionados y aplica los cambios en la base de datos.

**formDistribuidor:** Presenta los campos en los que se incluirá la información de un Distribuidor.

#### **Atributos**

<<input>> *shipper\_name\**: Nombre del Distribuidor (empresa distribuidora)

<<input>> *shipper\_email*: correo electrónico del Distribuidor

<<input>> *shipper\_web*: página web del Distribuidor

<<input>> *shipper\_password*: password del Distribuidor

<<select>> *shipper\_active*: campo que indica si el distribuidor está activo en el Servicio.

#### **Métodos**

<<submit>> *Adicionar\_Suscriptor*: Envía los datos para ser procesados.

**gestionDistribuidores:** Presenta los campos necesarios para eliminar los distribuidores.

#### **Atributos**

<<checkbox>> *delItem[]*: Aparece para cada Distribuidor sustraído de la BD para que el Administrador seleccione cuales desea eliminar.

Nota: Para una completa gestión de cada Distribuidor aparece un vínculo hacia el formulario de Distribuidor.

#### **Métodos**

<<submit>> *Eliminar\_items*: Elimina los items seleccionados una vez hecha la confirmación.

### 1.2.2. Descripción de Diagramas de Secuencia y clases para los Casos de Uso del Servicio

#### Gestionar Distribuidores

*GestionDistribuidores* hace una consulta a la base de datos a través de *editListingShipper* para desplegar una lista de los distribuidores activos y no activos en el servicio, con estos datos se crea el formulario *gestiónDistribuidores* que se encarga de eliminar los distribuidores que el administrador desee previa confirmación también realizada por *GestionSuscriptores*. Adicionalmente se le presenta al Administrador el vínculo hacia *editShopper* para una gestión más detallada de cada Distribuidor, allí se presenta el formulario *formDistribuidor* con los campos llenos para ser modificados.

#### Identificar Administrador

*ControIngresoAdministrador* se encarga de solicitar la identificación del administrador, mientras no halla una sesión abierta de este, en todas las páginas de Administrador mediante la función *challenge*, que presenta el formulario de identificación de Administrador *formChallenge*. Compara los datos obtenidos del formulario con los almacenados en la *Tarjeta\_Administrador*, si coinciden inicia sesión de administrador, permitiendo entrar a todas las páginas de este, si no es así presenta de nuevo el formulario *formChallenge*.

#### Gestionar Categorías

Este caso de uso se inicia cuando desde el menú principal de administrador escoge Gestionar Categorías, entonces se le envía a la página *editSections* donde *GestionCategoría* se encarga de hacer una consulta a la base de datos para desplegar una lista dentro del formulario *gestionCategorias* que le da la oportunidad al Administrador de eliminar, renombrar o crear nuevas categorías en cualquiera de las opciones elegidas, *GestionCategoría* hace las modificaciones pertinentes, en caso de

elegir eliminar categorías, se le anunciará al Administrador el riesgo que representaría en el funcionamiento del Centro y las sugerencias pertinentes.

### Gestionar Suscriptores

*GestionSuscriptores* hace una consulta a la base de datos a través de *editListing* para desplegar una lista de los suscriptores activos y no activos en el servicio, con estos datos se crea el formulario *gestiónSuscriptores* que se encarga de eliminar los suscriptores que el administrador desee previa confirmación también realizada por *GestionSuscriptores*. Adicionalmente se le presenta al Administrador el vínculo hacia *editSuscrip* para una gestión más detallada de cada Suscriptor, allí se presenta el formulario *formSuscriptor* con los campos llenos para ser modificados.

### Gestionar Registrados

*GestionRegistrados* hace una consulta a la base de datos a través de *editShoppers* para desplegar una lista de los registrados en el servicio, con estos datos se crea el formulario *gestionRegistrados* que se encarga de eliminar los registrados que el administrador desee previa confirmación también realizada por *GestionRegistrados*.

## 1.3 Funcionalidad de Suscriptor

### 1.3.1. Descripción de Subsistemas y Clases de Diseño del Servicio

#### Páginas de Usuario

**indice\_Suscriptor:** Presenta la lista de facilidades que el servicio le ofrece al suscriptor, tales como: gestionar productos, sus categorías y su perfil. Se presenta en todas las páginas de Suscriptor.

**editSections:** Presenta una lista de las categorías del Suscriptor dentro del formulario *GestionCategorias* que permite eliminar o cambiar el nombre de las categorías. Utiliza la función *resultSetNav*.

**editMe:** Presenta el formulario formSuscriptor con los campos llenos con su información de Suscriptor con el fin de modificar la información, la valida y la almacena en la base de datos.

**editListing:** Presenta la lista de productos dentro del formulario GestionSuscriptor para eliminar cada producto o para actualizar los datos de este a través de un vínculo hacia el formulario de producto.

**addProduct:** Presenta el formulario FormProducto con el fin de adicionar un nuevo producto, valida y almacena la información en la base de datos.

**editProduct:** Presenta el formulario FormProductor con los campos llenos con la información del producto con el fin de modificar la información, la valida y la almacena en la base de datos.

**addColor:** Presenta el formulario formEscogencia con el fin de adicionar una nueva escogencia gráfica, valida y almacena la información en la base de datos.

**editColor:** Presenta el formulario formEscogencia con los campos llenos con la información de la escogencia con el fin de modificar la información, la valida y la almacena en la base de datos.

**editColors:** Presenta la lista de escogencias dentro del formulario GestionEscogencia para eliminar cada producto o para actualizar los datos de este a través de un vínculo hacia el formulario formEscogencia.

<b>Funciones comunes</b>
--------------------------

**header:** Presenta el menú de Suscriptor en todas las páginas de Suscriptor.

<b>Clases de Formulario</b>
-----------------------------

**formchallenge:** Presenta los campos necesarios para realizar la identificación del Suscriptor.

### **Atributos**

<<input>> *PHP\_AUTH\_USER\**: Login del Suscriptor

<<input *PHP\_AUTH\_PW\**: Password del Suscriptor

### **Métodos**

<<submit>> *Aceptar*: Envía los datos para ser procesados.

**formSuscriptor**: Presenta los campos para modificar la información del Suscriptor.

### **Atributos**

<<input>> *suscrip\_name\**: Nombre de la empresa

<<input>> *suscrip\_image*: Logo

<<input>> *suscrip\_frase*: Frase corporativa de la empresa

<<input>> *suscrip\_desc*: Descripción de la empresa

<<input>> *categorías*: Las categorías con las que funcionará la tiendas

<<input>> *suscrip\_keywords*: Palabras claves que se usarán en la búsqueda

<<input>> *suscrip\_correo*: Correo electrónico

<<input>> *suscrip\_password*: Palabra de paso para entrar a la interfaz de suscriptor.

<<input>> *suscrip\_repassword*: Palabra de paso para entrar a la interfaz de suscriptor.

<<input>> *suscrip\_telefono*: Teléfono de contacto

<<input>> *suscrip\_web*: Página web de la empresa (alojada en otro servidor)

### **Métodos**

<<submit>> *Aceptar Cambios*: Envía los datos para ser procesados.

**gestionCategorias**: Presenta los campos necesarios para eliminar, renombrar o adicionar categorías.

### **Atributos**

<<checkbox>> *delItem[]*: Aparece para cada categoría para que el Suscriptor seleccione cuáles desea eliminar.

<<input>> *new\_name[]*: Campo para colocar un nuevo nombre de la categoría.

<<input>> *new\_id[]*: Campo para colocar un nuevo id de la categoría.

<<input>> *NC*: Campo para agregar una nueva categoría.

Nota: Cambiar el id o eliminar una categoría puede ocasionar daños en el funcionamiento del Centro.

### **Métodos**

<<submit>> *Aplicar cambios*: Elimina los items seleccionados y aplica los cambios en la base de datos.

**formEscogencia:** Presenta los campos en los que se incluirá la información de una escogencia gráfica.

**Atributos**

<<input>> *color\_name*: Nombre de la escogencia

<<file>> *color\_image*: Foto de la escogencia

**Métodos**

<<submit>> *Adicionar\_Escogencia*: Envía los datos para ser procesados.

**formProducto:** Presenta los campos en los que se incluirá la información de un producto.

**Atributos**

<<input>> *product\_id\**: Referencia del producto

<<input>> *product\_name*: Nombre del producto

<<file>> *product\_image*: Foto del producto

<<input>> *product\_option\_one*: Opción número 1 (formato nombreOpcion: primera, segunda, tercera, etc)

<<input>> *product\_option\_two*: Opción número 2

<<input>> *product\_option\_three*: Opción número 3

<<input>> *product\_option\_four*: Opción número 4

<<input>> *product\_desc*: Descripción del

<<select>> *categorías*: Utiliza la función SelectBox para presentar cada categoría con las que trabaja el suscriptor que está adicionando el producto.

<<input>> *suscrip\_keywords*: Palabras claves que se usarán en la búsqueda

<<input>> *product\_turnaround*: Peso del producto.

<<input>> *product\_disclaimer*: Especificaciones acerca del producto

<<input>> *product\_price*: Precio del producto

<<input>> *\$product\_special\_price*: Precio especial del producto (oferta)

<<checkbox >> *product\_active*: Por defecto está activado si no se activa no aparecerá en el servicio, pero se mantendrá en la base de datos.

**Métodos**

<<submit>> *Adicionar\_Producto*: Envía los datos para ser procesados.

**GestionProductos:** Presenta los campos necesarios para eliminar los productos del suscriptor que ha iniciado sesión.

**Atributos**

<<checkbox>> *delItem[]*: Aparece para cada producto sustraído de la BD para que el Suscriptor seleccione cuales desea eliminar.

Nota: Para una mayor gestión de cada Producto aparece un vínculo hacia el formulario de Producto.

#### **Métodos**

<<submit>> *Eliminar\_items*: Elimina los items seleccionados una vez hecha la confirmación.

**GestionEscogencia:** Presenta los campos necesarios para eliminar las escogencias del suscriptor que ha iniciado sesión.

#### **Atributos**

<<checkbox>> *delItem[]*: Aparece para cada Escogencia sustraído de la BD para que el Suscriptor seleccione cuales desea eliminar.

#### **Métodos**

<<submit>> *Eliminar\_items*: Elimina los items seleccionados una vez hecha la confirmación.

### **1.3.2. Descripción de Diagramas de Secuencia y clases para los Casos de Uso del Servicio**

#### **Gestionar Categorías**

Este caso de uso se inicia cuando desde el menú principal de Suscriptor escoge Gestionar Categorías, entonces se le envía a la página *editSections* donde *GestionCategoría* se encarga de hacer una consulta a la base de datos para desplegar una lista dentro del formulario *gestionCategorias* que le da la oportunidad de eliminar, renombrar o crear nuevas categoría, *GestionCategoría* hace las modificaciones pertinentes.

#### **Gestionar Escogencias Gráficas**

El Suscriptor decide gestionar escogencias gráficas, *GestionEscogencias* le permite, adicionar una nueva escogencia a través de *addColor*, eliminar escogencias a través de *editColors* que presenta una interfaz desde donde permite editar una escogencia existente a través de *editColor*.

### Gestionar Producto

Este caso de uso se inicia cuando desde el menú principal de Suscriptor se escoge Gestionar Producto, entonces se le envía a la página *editListing* donde *GestionProducto* se encarga de hacer una consulta a la base de datos para desplegar una lista dentro del formulario *gestionCategorias*. *GestionSuscriptores* hace una consulta a la base de datos a través de *editListing* para desplegar una lista de los productos activos y no activos en el servicio, con estos datos se crea el formulario *GestionProductos* que se encarga de eliminar los productos que el administrador desee previa confirmación.. Adicionalmente se le presenta al Administrador el vínculo hacia *editProduct* para una gestión más detallada de cada Producto, allí se presenta el formulario *formProducto* con los campos llenos para ser modificados.

El Suscriptor también pudo haber solicitado agregar un nuevo producto, entonces *GestionProductos* le presenta el formulario *formProducto* para que se llenen los datos del nuevo producto.

En cualquier caso *GestionProductos* valida y almacena los datos.

### Identificar Suscriptor

Se inicia cuando se intenta ingresar a alguna página de Suscriptor *ControlIngresoAdministrador* se encarga de solicitar la identificación del Suscriptor (mientras no halla una sesión abierta de este) mediante la función challenge, que presenta el formulario de identificación de Suscriptor *formChallenge*. Compara los datos obtenidos del formulario con los almacenados en la *Tarjeta\_Suscriptor*, si coinciden inicia sesión de Suscriptor, permitiendo entrar a todas las páginas de este, si no es así presenta de nuevo el formulario *formChallenge*.

### Revisar Pedidos

Cuando se halla hecho un nuevo pedido por parte de un usuario Registrado, *ControlPedidos* envía un correo a cada Suscriptor involucrado, este a su vez revisa el detalle del pedido para autorizarlo.

#### 1.4 Funcionalidad de Distribuidor

### 1.3.1. Descripción de Subsistemas y Clases de Diseño del Servicio

#### Páginas de Usuario

**indice\_Distribuidor:** Presenta la lista de facilidades que el servicio le ofrece al suscriptor, tales como: gestionar fletes, y gestionar Entregas. Se presenta en todas las páginas de Suscriptor.

**editFletes:** Presenta una lista de los países dentro del formulario GestionFletes que permite revisar y reasignar el valor de los fletes que el Distribuidor cobra por realizar el envío a cada país.

#### Funciones comunes

**header:** Presenta el menú de Suscriptor en todas las páginas de Suscriptor.

#### Clases de Formulario

**formchallenge:** Presenta los campos necesarios para realizar la identificación del Suscriptor.

##### Atributos

<<input>> *PHP\_AUTH\_USER\**: Login del Distribuidor

<<input>> *PHP\_AUTH\_PW\**: Password del Distribuidor

##### Métodos

<<submit>> *Aceptar*: Envía los datos para ser procesados.

**gestionFletes:** Presenta una caja de texto frente a cada país para modificar el valor del flete.

##### Atributos

<<input>> *flete[]*: Campo para colocar el valor del flete para cada país.

Nota: Cambiar el id o eliminar una categoría puede ocasionar daños en el funcionamiento del Centro.

##### Métodos

<<submit>> *Aplicar cambios* Aplica los cambios en la base de datos.

### 1.3.2. Descripción de Diagramas de Secuencia y clases para los Casos de Uso del Servicio

#### Gestionar Fletes

El Distribuidor decide revisar el valor de los fletes, *GestionFletes* le entrega la lista de los países con el valor del flete en un campo de texto para poder ser modificados con el formulario *gestionFletes*. El Distribuidor acepta los cambios y *GestionFletes* valida y los datos y actualiza la base.

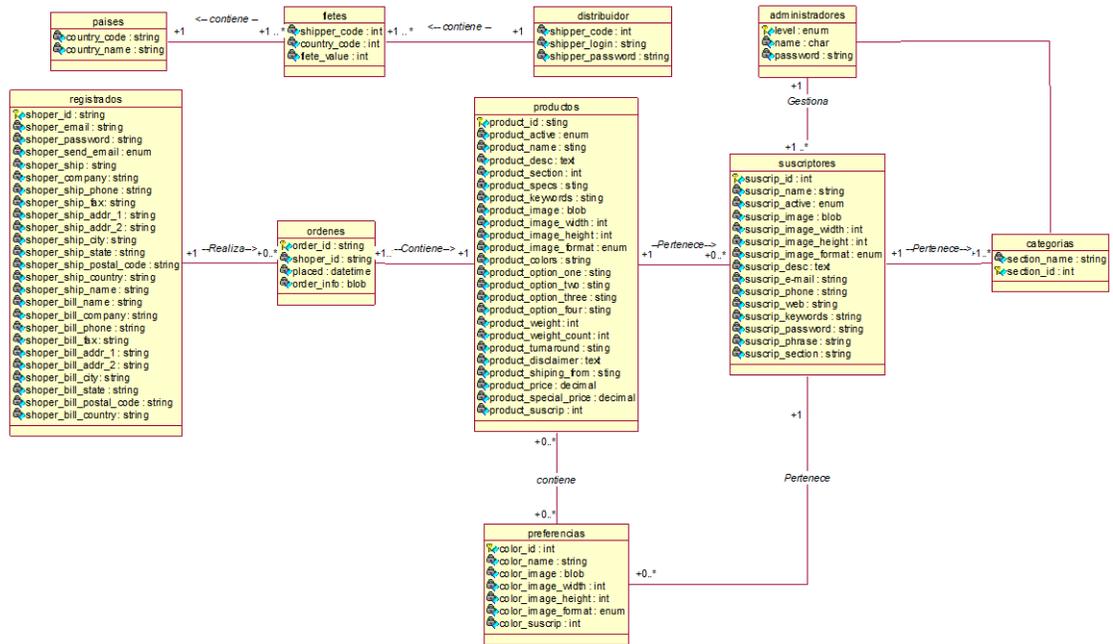
#### Identificar Distribuidor

*ControlIngresoDistribuidor* se encarga de solicitar la identificación del Distribuidor (mientras no halla una sesión abierta de este) mediante la función *challenge*, que presenta el formulario de identificación de Distribuidor *formChallenge*. Compara los datos obtenidos del formulario con los almacenados en la *Tarjeta\_Distribuidor*, si coinciden inicia sesión de Distribuidor, permitiendo entrar a todas las páginas de este, si no es así presenta de nuevo el formulario *formChallenge*.

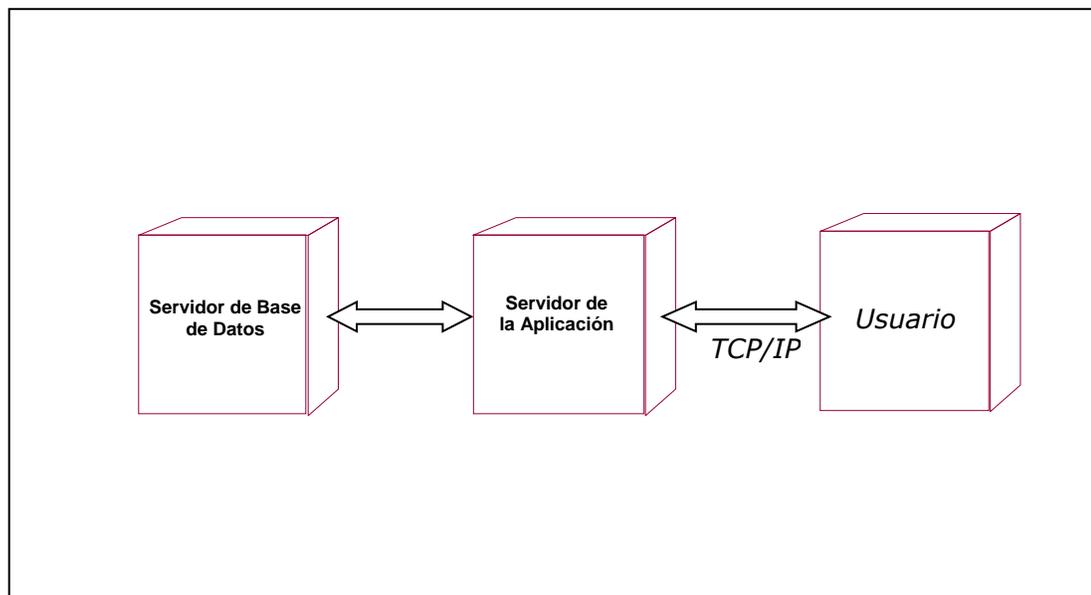
#### Revisar Entregas

Cuando se halla hecho un nuevo pedido por parte de un usuario Registrado, *ControlPedidos* envía un correo al Distribuidor elegido, este a su vez revisa el detalle del pedido para hacer efectiva la entrega.

## 2.- MODELO ENTIDAD RELACIÓN



## Subproducto Esencial No. 1D: Modelo de Despliegue del Servicio



### Usuario

El nodo usuario debe presentar los siguientes requerimientos mínimos:

- Navegador por ejemplo: Internet Explorer 6.0 en adelante y Navigator 5.0
- Hardware soporte de red

### Servidor de la Aplicación

Es el servidor que contiene la lógica del servicio, encargado de la comunicación con los clientes y sus solicitudes (Vía HTTP). El Centro Comercial Virtual para el portal Tampu será un sistema software de tipo cliente servidor basado en PHP.

### Servidor de Base de datos

Es el servidor que contendrá la información sobre los Productos y Servicios pertenecientes al Centro comercial virtual.

Estará soportado sobre MySQL.

Tanto el servidor de la aplicación como el de base de datos, deberá implantarse en el servidor Web de la Universidad del Cauca donde reside el dominio de Tampu.

### Subproducto Esencial No. 1E: Modelo de Implementación del Servicio.

Se presentan los aspectos más relevantes de la implementación y la distribución de archivos.

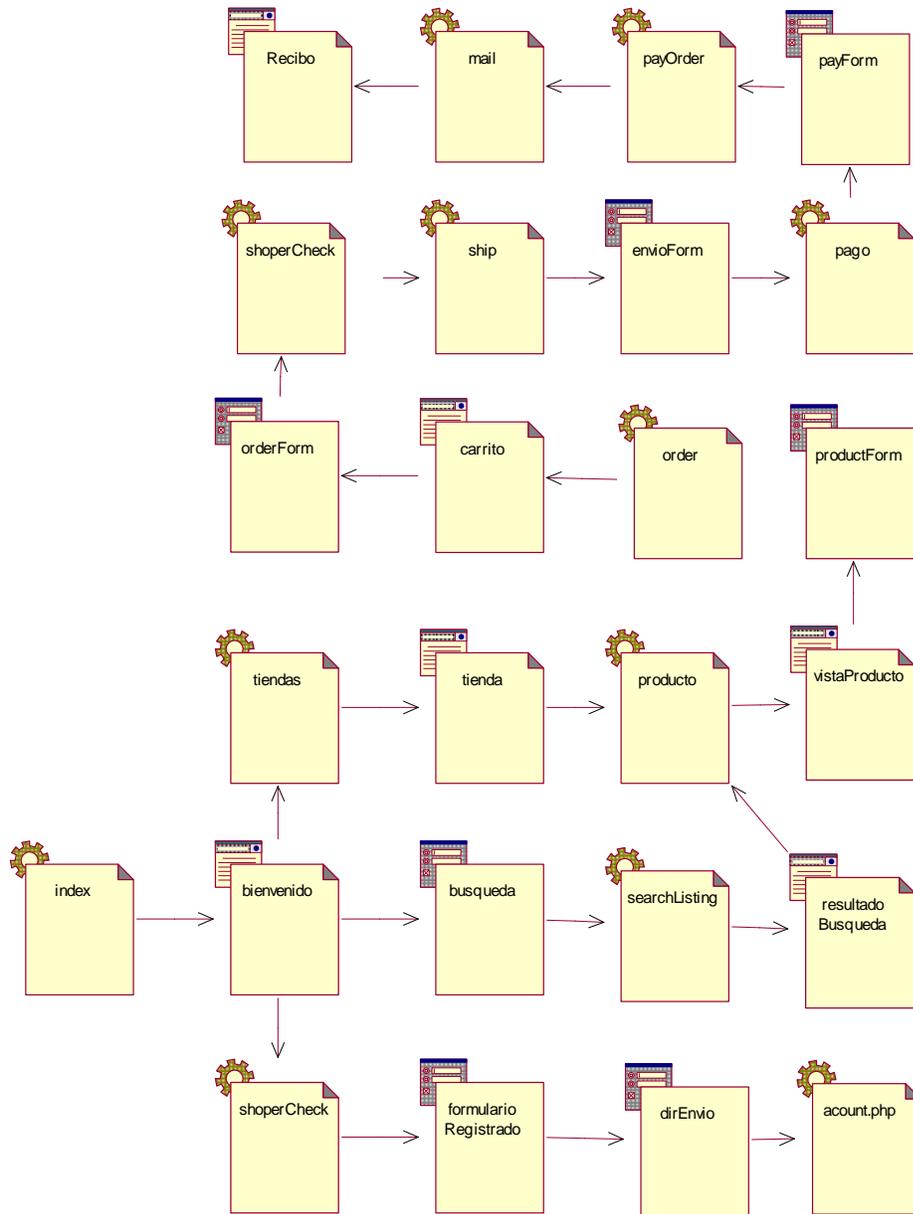
#### 1. Distribución funcional de la arquitectura

El servidor de Base de Datos contiene como un paquete de archivos llamado centrocomercial.

#### Archivos

<b>Servicios de usuario</b>	<b>Suscriptor</b>	<b>Distribuidor</b>
acoount billAddressInfo getPwd index listing login order orderDump orderRetrieve orderSummary placedOrders product receipt remove_product search searchListing searchSuscrip shipAddressInfo shipMethodInfo shopperCheck store viewImage	addColor addProduct editColor editColors editListing editMe editProduct editSections footer header index  <b>Administrador</b> addShiper addSuscrip editListing editListingShippers editSections editShipper editSuscrip footer header index editShoppers	fletes  <b>Comunes</b> admin.authorize.inc common.functions.inc config.inc credit_card.inc dbdefs.inc footer.inc header.inc mail.functions.inc order.functions.inc shipment.inc shopper.functions.inc  <b>Nota:</b> Extensión de todos los archivos: PHP

## 2. Esquema de Navegación



## **Subproducto Esencial No. 1F: Modelo de Pruebas del Servicio.**

Las pruebas se realizarán según las fechas establecidas en la planeación general.

### **Prueba 1: Sobre Acceso y Seguridad.**

#### **Problemas Potenciales**

- Un intruso accede a un lugar no autorizado (Sitio del Administrador)
- Un intruso suplanta identidad de un Registrado sin tener el login y password.
- Un Registrado a quien se le ha cancelado la cuenta intenta acceder.
- El sistema permite seguir navegando a un Intruso a través del sitio después de haber finalizado una sesión.
- Existen caracteres que permiten violar la seguridad del sistema.

#### **Estrategia**

- Investigar en la red o en documentos sobre formas de violar seguridad de sitios y aplicarlas al servicio. Establecer si el sitio es vulnerable frente a un ataque de ese tipo.
- Intentar violar la seguridad accediendo a través de cada una de las páginas del servicio y con diferentes parámetros en la cabecera, sin antes haber pasado por la página de ingreso.
- Intentar repetidas veces ingresar al servicio con diferentes logins y passwords.
- Ingresar con una cuenta registrada después de fallar múltiples veces y observar el comportamiento del sistema.

### **Prueba 2: Comprobación de Navegabilidad.**

#### **Problemas Potenciales**

- Existencia de alguna página que presenta errores si es accedida desde un lugar diferente al establecido pese a existir una sesión. Existencia de alguna página que reinicie las variables de sesión que llevan el estado del Suscriptor.

#### **Estrategia**

Navegar a través de todo el sitio, acceder a las páginas mencionadas de forma directa sin utilizar el vínculo que provee alguna página, al igual que utilizar parámetros en la cabecera y la evaluación de su respectiva respuesta.

### **Prueba 3: Comprobación de enlaces.**

#### **Problemas Potenciales**

- Existencia de algún enlace apuntando a una página no existente.
- Nombre no coherente entre el nombre de algún enlace y la función.

#### **Estrategia**

Se navega a través de todo el sitio y se evalúa cada enlace verificando que no existan incoherencias entre nombres de enlaces y enlaces efectivos.

### **Prueba 4: Sobre búsquedas.**

#### **Problemas Potenciales**

- Entrega de Datos Repetidos.
- Entrega de Datos Inconsistentes.
- No entrega de algún tipo de dato.
- Tiempo de Respuesta Excesivamente largo (Mayor a 30 Segundos).

#### **Estrategia**

Desde cualquier formulario de búsqueda, se realiza búsquedas varias (palabras, letras, números, signos) y se observa comportamiento.

Se comprueba que los problemas anteriores no se presenten.

### **Prueba 5: Sobre inclusión y modificación de Datos.**

#### **Problemas Potenciales**

- El sistema no compruebe que se ha llenado algún campo requerido.
- El sistema no compruebe si el usuario ya existe.
- El sistema no compruebe que el password y la confirmación coincidan.
- Para el caso de modificar password el sistema no compruebe si el dato del password actual es correcto.
- Existen caracteres especiales que impiden establecer lo anterior.

#### **Estrategia**

Evaluar si las anteriores situaciones existen y aplicar los correctivos.

Estos son los criterios de evaluación para cada formulario.

### **Prueba 6 Pago**

#### **Problemas Potenciales**

- Se intenta pagar con datos falsos de tarjeta de crédito

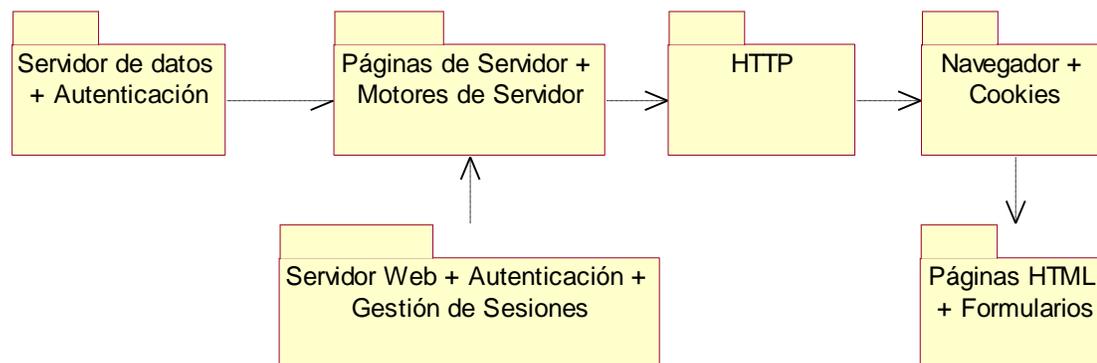
#### **Estrategia**

Se introducirán diferentes combinaciones de número de tarjetas de crédito.

### **Prueba 7 Consistencia de las Variables de Sesión.**

Se harán peticiones al servidor desde diferentes clientes y se comprobará la consistencia e integridad de los datos de sesión.

## Subproducto Esencial No. 2: Descripción de Arquitectura del Servicio



## Subproducto Esencial No. 4: Lista de Riesgos Actualizada.

### Riesgos de Tipo Crítico

#### 1. Metodología de Reutilización

**Descripción:**

El desconocimiento de herramientas que permitan acondicionar la información y módulos software existentes.

**Impacto:**

Calidad del servicio, insatisfacción del cliente.

**Estrategias:**

Revisar documentación, modelos, submódulos, clasificar la información existente y elegir la más conveniente para el proyecto.

#### 2. Infraestructura de Red

**Descripción:**

El desconocimiento de condiciones específicas de la red que puedan incidir negativamente en la prestación del servicio.

**Impacto:**

Altas limitaciones con respecto a la prestación del servicio.

**Estrategias:**

Implementar un prototipo que nos permita visualizar esas limitaciones.

**3. Compatibilidad de Herramientas**

**Descripción:**

Consistente en problemas que durante el desarrollo o integración se presenten entre las distintas herramientas con las que se generaran diversos componentes.

**Impacto:**

Redefinición de diseño y retraso en el proyecto.

**Estrategias:**

Documentarse al respecto de la compatibilidad de herramientas a usar, realizar pruebas sobre compatibilidad que no este clara, mantener consistencia con la plataforma de Java.

**Riesgo Tipo Significativo**

**4. Asesoría**

**Descripción:**

Consistente en encontrar y disponer de asesoría apropiada en algunos aspectos de implementación o de una tecnología en particular.

**Impacto:**

Podría ocasionar retraso en el proyecto.

**Estrategias:**

Revisar las alternativas en cuestión de asesoría o en últimas revisar alternativas de cambio de tecnología siempre y cuando no afecte la arquitectura.

**5. Eficiencia de los servidores y manejadores de bases de datos**

**Descripción:**

Consistente en la conformidad en cuanto a requerimientos de rapidez y rendimiento que el servicio necesita que es soportada por los servidores y la base de datos.

**Impacto:**

Baja calidad den la prestación del servicio, selección de nuevo de servidores o bases de datos.

**Estrategias:**

Documentarse al respecto sobre la eficiencia de servidores y bases de datos a usar, realizar pruebas preliminares, mantener varias opciones viables y compatibles.

**Riesgos de Tipo Ordinario**

**6. Falta de Coordinación con la Empresa**

**Descripción:**

Consistente en la falta de comunicación y lenguaje común con los representantes de la empresa.

**Impacto:**

Retraso en el proyecto, calidad del servicio.

**Estrategias:**

Hacer un compromiso firme con las directivas de la empresa en cuestión.

## Subproducto Esencial No 5 Manuales iniciales de usuario y Material de Apoyo.

Los manuales y material de apoyo se encuentran en el Anexo 5.

### Subproducto Esencial No. 6: Plan del Proyecto

#### 1. Actividades a realizar y duración

Actividad	Duración (días)	Esfuerzo (H-H)
1. Implantación en el entorno de ejecución.	2	15
2. Determinación de satisfacción de requerimientos.	5	20
3. Corrección de los defectos.	5	40
4. Verificación de modelos y descripción de arquitectura.	5	15
5. Evaluación del proyecto.	4	12

#### 2. Asignación de responsabilidades

Estas actividades caen sobre todo el equipo, donde el director y asesores serán parte de la comunidad de usuarios que prueban el servicio.

#### 3. Cronograma

Id	Nombre de tarea	Duración	15 enero					22 enero					29 enero											
			M	M	J	V	S	D	L	M	M	J	V	S	D	L	M	M	J	V	S	D	L	M
1	Implantación en el entorno de ejecución.	2d																						
2	Determinación de satisfacción de requerimientos.	5d																						
3	Corrección de los defectos.	5d																						
4	Verificación de modelos y descripción de arquitectura.	5d																						
5	Evaluación del proyecto.	3.5d																						

#### 4. Recursos requeridos y disponibilidad

Se mantienen los recursos y disponibilidad planteada en el punto 1 del subproducto Complementario 1

Se mantiene el PC como entorno de desarrollo y de ejecución.

Referente al software se disponen de versiones de prueba (actualmente).

Adicionalmente se contempla

Varios: \$200.000

## 5. Estimación de costos

Tomando como base las normas de depreciación de los recursos vigente al interior de la universidad y el tiempo estimado para el proyecto tenemos:

3 semanas de duración de Validación del servicio.

Horas estimadas de trabajo:

Desarrolladores 30 horas c/u por semana (1.5 puntos c/u)

Director del proyecto 2 horas por semana (2.5 puntos)

Asesores 2 horas en total por semana

PC costo ponderado \$1.200.000 uso 30 horas por semana

Costo punto \$6134 (según reajuste retroactivo al momento de generar este documento)

### Costos

Desarrolladores	\$ 1.656.180
Director	\$ 92.010
Asesores	\$ 92.010
PC1	\$ 522.720
Otros	\$ 200.000
<b>TOTAL</b>	<b>\$ 2.024.220</b>

## Subproducto Esencial No. 7: Caso del Negocio

### 1. Beneficios que se quieren lograr.

#### **Corto Plazo:**

- Apropiación de tecnologías para el desarrollo de servicios orientados a B2C
- Brindar la oportunidad a las empresas vinculadas al portal Tampu dar a conocer los bienes y servicios que estas ofrecen.
- Servicio de alta calidad.
- Un servicio amigable que permita acceder a todas sus prestaciones y que éstas se provean de forma fácil, rápida y segura.

Se debe tener en cuenta que el servicio se constituye como una construcción de una aplicación para la innovación, apropiando tecnologías, y finalmente generando conocimiento a partir de un mayor uso de los recursos disponibles en la Internet.

#### **Mediano Plazo:**

- Un servicio que permita la adición de una buena cantidad de nuevas tiendas.
- Un servicio que se pueda actualizar constantemente en sus contenidos.
- Un servicio configurable que permita realizar cambios en la presentación de contenidos sin que ello implique modificar la configuración o arquitectura del servicio.

#### **Largo Plazo:**

- El servicio debe estar diseñando y construido para evolucionar, esto implica que se debe implementar un sistema basado en componentes.
- Implementar una línea de base para el despliegue en el futuro de más y mejores características del servicio.
- Ofrecer una herramienta para promover el desarrollo económico y cultural del departamento del Cauca.

### 2. Costo estimado para construcción del servicio.

Según estimativa preliminar de la planeación la etapa de validación del proyecto tomara el valor de :	\$ 2.024.220
---	--------------

Según estimativa preliminar de la planeación incluyendo lo corrido del proyecto(Prefactibilidad, factibilidad y creación):	\$28.325.720
--	--------------

### 3. Duración estimada para el desarrollo del proyecto.

Según estimativa preliminar de la planeación del resto del proyecto se tiene un periodo de :	3 semanas
Periodo comprendido entre:	Abril 01 de 2002 Abril 19 de 2002

Según estimativa preliminar de resto del proyecto (incluyendo Prefactibilidad, factibilidad y creación):	45 semanas
Periodo comprendido entre:	Junio 03 de 2001 Abril 26 de 2002

### 4. Nivel de calidad del servicio que se aspira a lograr.

Desde la perspectiva del Usuario Final del Servicio, la calidad depende fundamentalmente de:

- 1.- La existencia de un ambiente amigable que le permita acceder a todas las prestaciones del servicio y que éstas se provean de forma fácil, rápida y segura.
- 2.- Facilitar al usuario el proceso de compras, permitiendo la gestión de estas.
- 3.- Proveer distintas alternativas de pago.

Desde la perspectiva del Administrador del Servicio, la calidad depende de:

- 1.- La mantenibilidad del servicio, es decir, la facilidad para adaptarse tanto a los cambios de la red de soporte como a los cambios en las necesidades de los Usuarios Finales.

- 2.- Del cumplimiento de las características sobre seguridad y registro de uso del servicio.
- 3.- De una buena calidad de la atención al Usuario Final, tener bajas probabilidades de que se deje de prestar el servicio por la falla de alguna de sus partes.

Desde la perspectiva del Suscriptor, la calidad depende de:

1. Mantener información actualizada de los últimos pedidos realizados por los usuarios.
2. Facilitar la gestión de la información de los bienes y servicios que ofrece.
3. Presentar al suscriptor la información del cliente necesaria para llevar estadísticas.
4. Soporte para diferentes formas de pago.

Desde la perspectiva del Distribuidor, la calidad depende de:

1. Entregar al distribuidor información clara y concisa acerca del envío.
2. Modificar las diferentes tarifas de envío.

Adicionalmente desde la perspectiva de los Desarrolladores del Servicio la calidad depende también de:

- 1.- Una entrega oportuna de cada uno de los prototipos.
- 2.- El logro de costos competitivos.
- 3.- La escalabilidad del servicio desarrollado

## **5. Apreciación sobre la conveniencia del proyecto.**

- Se considera conveniente pasar a la fase de validación.
- Se considere que el prototipo ha alcanzado los niveles de calidad requeridos.
- Se mantiene la conveniencia del servicio creado a pesar de los retrasos.
- Se actualiza el Cronograma y costos.

## **SUBPRODUCTOS COMPLEMENTARIOS - FASE DE CREACIÓN DEL SERVICIO**

### **Subproducto Complementario No 1: Declaración de recursos Disponibles**

#### **1. Descripción de recursos físicos y técnicos del proyecto para el proyecto**

Se encuentran disponibles totalmente para el proyecto actualmente:

##### **Logísticos y físicos**

Oficina 111 IPET Ambientes de Desarrollo

1 punto de red

1 Hub 8 puertos

##### **Técnicos**

##### **Hardware**

Computador asignado para el proyecto que será entorno de desarrollo y de ejecución

**PC1** Marca DTK, Procesador Pentium III de 450 Mhz, Memoria RAM de 64 MB, Disco duro de 8 Gbyte, Drive para disco 3.5" de 1.44 Mbyte, Multimedia IDE CD ROM, Tarjeta de red.

Un PC (**PC2**) adicional de apoyo (Disponibilidad 50%)

**PC2** Pentium MMX 200 MHz, Memoria 64 megas, HD 4 Gigas, Office 97.

##### **Software**

Sistema Operativo Windows 98

Office 2000

Documentación disponible en software

Rational Rose (versión de evaluación).

##### **Herramientas de desarrollo**

Apache

PHP4

MySQL

### **Bibliográfica**

Metodología Integral para Construcción de Servicios, Gustavo Adolfo Ramírez – Darío E. Melo

### **Otros**

Papelería

## **2. Descripción del Equipo del Proyecto**

Se mantiene los integrantes que participaron de la etapa anterior

Los desarrolladores: Jhon Edwin Prado y Willinthon Silva

El director del proyecto: Ing. Diego M. López

Los asesores: Ing. Mario Solarte e Ing. Gustavo A. Ramírez

## **Subproducto Complementario No 2**

### **Plan de Construcción Incremental del Prototipo Operacional del Servicio**

#### **1. Secuencia de incrementos a lograr (o de iteraciones a realizar).**

Las siguientes se esperan sean las iteraciones necesarias para alcanzar la capacidad operacional deseada:

- A. Análisis de Funcionalidad Usuario.
- B. Versión Inicial de Funcionalidad de Usuario.
- C. Funcionalidad de Usuario (sesiones y excepciones).
- D. Análisis de Funcionalidad Administrador.
- E. Versión Inicial de Funcionalidad de Administrador
- F. Funcionalidad de Administrador (sesiones, excepciones)
- G. Análisis de Funcionalidad Suscriptor.
- H. Versión Inicial de Funcionalidad de Suscriptor.
- I. Funcionalidad de Suscriptor (sesiones y excepciones).
- J. Análisis de Funcionalidad Distribuidor.
- K. Versión Inicial de Funcionalidad de Distribuidor.
- L. Funcionalidad de Distribuidor (sesiones y excepciones).

#### **2. Relación de tareas a realizar para cada iteración y su duración estimada.**

- A. Análisis de Funcionalidad usuario.
  - 1. Diagramación e implementación de prototipos de interfaces para usuario.
  - 2. Actualización del modelo de casos de Uso parte de Visitante y Registrado.
- B. Versión Inicial de Funcionalidad de Usuario
  - 3. Diseño, implementación y pruebas de páginas basados en prototipos (sin excepciones de los casos de uso).
  - 4. Actualización de Base de datos Preliminar
- C. Funcionalidad de usuario (sesiones y excepciones)
  - 5. Actualización de modelo de análisis y revisión preliminar de arquitectura.
  - 6. Diseño, implementación y pruebas de páginas basados en prototipos (con excepciones de los casos de uso).

7. Actualización de Modelo de Diseño, Implementación y despliegue.
  8. Elaboración y consecución del plan de pruebas parte de usuario.
- D. Análisis de Funcionalidad Administrador
9. Diagramación e implementación de prototipos de interfaces para Administrador.
  10. Actualización del modelo de casos de Uso.
- E. Versión Inicial de Funcionalidad de Administrador
11. Diseño, implementación y pruebas de páginas basados en prototipos (sin excepciones de los casos de uso).
  12. Actualización de Base de datos.
- F. Funcionalidad de Administrador (sesiones y excepciones)
13. Actualización de modelo de análisis y revisión preliminar de arquitectura.
  14. Diseño, implementación y pruebas de páginas basados en prototipos (con excepciones de los casos de uso).
  15. Actualización de Modelo de Diseño, Implementación y despliegue.
  16. Elaboración y consecución del plan de pruebas parte de Administrador.
- G. Análisis de Funcionalidad Suscriptor
17. Diagramación e implementación de prototipos de interfaces para Suscriptor.
  18. Actualización del modelo de casos de Uso.
- H. Versión Inicial de Funcionalidad de Suscriptor
19. Diseño, implementación y pruebas de páginas basados en prototipos (sin excepciones de los casos de uso).
  20. Actualización de Base de datos.
- I. Funcionalidad de Suscriptor (sesiones y excepciones)
21. Actualización de modelo de análisis y revisión preliminar de arquitectura.
  22. Diseño, implementación y pruebas de páginas basados en prototipos (con excepciones de los casos de uso).
  23. Actualización de Modelo de Diseño, Implementación y despliegue.
  24. Elaboración y consecución del plan de pruebas parte de Administrador.
- J. Análisis de Funcionalidad Distribuidor
25. Diagramación e implementación de prototipos de interfaces para Distribuidor.
  26. Actualización del modelo de casos de Uso.
- K. Versión Inicial de Funcionalidad de Distribuidor
27. Diseño, implementación y pruebas de páginas basados en prototipos (sin excepciones de los casos de uso).

28. Actualización de Base de datos.

L. Funcionalidad de Distribuidor (sesiones y excepciones)

29. Actualización de modelo de análisis y revisión preliminar de arquitectura.

30. Diseño, implementación y pruebas de páginas basados en prototipos (con excepciones de los casos de uso).

31. Actualización de Modelo de Diseño, Implementación y despliegue.

32. Elaboración y consecución del plan de pruebas parte de Administrador.

33. Evaluación de pruebas realizadas.

34. Evaluación de la etapa.

35. Planeación de la etapa de transición.

### 3. Asignación de responsabilidades.

Se establece la siguiente convención teniendo en cuenta los nombres de los desarrolladores se tomaran **E** y **W**, como responsable principal

Actividad	Res	Estimación	
		Días	H/H
<b>A. Análisis de Funcionalidad usuario.</b>			
1. Diagramación e implementación de prototipos de interfaces para usuario.	E	5	35
2. Actualización del modelo de casos de Uso parte de visitante y registrado.	WE	5	35
<b>B. Versión Inicial de Funcionalidad de Usuario</b>			
3. Diseño, implementación y pruebas de páginas basados en prototipos (sin excepciones de los casos de uso).	E	8	40
4. Actualización de Base de datos Preliminar	EW	2	12
<b>C. Funcionalidad de usuario (sesiones y excepciones)</b>			
5. Actualización de modelo de análisis y revisión preliminar de arquitectura.	W	5	30
6. Diseño, implementación y pruebas de páginas basados en prototipos (con excepciones de los casos de uso).	E	10	40
7. Actualización de Modelo de Diseño, Implementación y despliegue.	WE	5	30
8. Elaboración y consecución del plan de pruebas parte	E	3	18

de usuario.			
<b>D. Análisis de Funcionalidad Administrador</b>			
9. Diagramación e implementación de prototipos de interfaces para Administrador.	E	5	35
10. Actualización del modelo de casos de Uso.	WE	5	35
<b>E. Versión Inicial de Funcionalidad de Administrador</b>			
11. Diseño, implementación y pruebas de páginas basados en prototipos (sin excepciones de los casos de uso).	E	8	40
12. Actualización de Base de datos.	EW	2	12
<b>F. Funcionalidad de Administrador (sesiones y excepciones)</b>			
13. Actualización de modelo de análisis y revisión preliminar de arquitectura.	W	5	30
14. Diseño, implementación y pruebas de páginas basados en prototipos (con excepciones de los casos de uso).	E	10	40
15. Actualización de Modelo de Diseño, Implementación y despliegue.	WE	5	30
16. Elaboración y consecución del plan de pruebas parte de Administrador.	E	3	18
<b>G. Análisis de Funcionalidad Suscriptor</b>			
17. Diagramación e implementación de prototipos de interfaces para Suscriptor.	E	5	35
18. Actualización del modelo de casos de Uso.	WE	5	35
<b>H. Versión Inicial de Funcionalidad de Suscriptor</b>			
19. Diseño, implementación y pruebas de páginas basados en prototipos (sin excepciones de los casos de uso).	E	7	35
20. Actualización de Base de datos.	EW	2	12
<b>I. Funcionalidad de Suscriptor (sesiones y excepciones)</b>			
21. Actualización de modelo de análisis y revisión preliminar de arquitectura.	W	5	30
22. Diseño, implementación y pruebas de páginas basados en prototipos (con excepciones de los casos de uso).	E	7	28
23. Actualización de Modelo de Diseño, Implementación y	WE	5	30

despliegue.			
24. Elaboración y consecución del plan de pruebas parte de Administrador.	E	3	18
<b>J. Análisis de Funcionalidad Distribuidor</b>			
25. Diagramación e implementación de prototipos de interfaces para Distribuidor.	E	5	35
26. Actualización del modelo de casos de Uso.	WE	5	35
<b>K. Versión Inicial de Funcionalidad de Distribuidor</b>			
27. Diseño, implementación y pruebas de páginas basados en prototipos (sin excepciones de los casos de uso).	E	7	35
28. Actualización de Base de datos.	EW	2	12
<b>L. Funcionalidad de Distribuidor (sesiones y excepciones)</b>			
29. Actualización de modelo de análisis y revisión preliminar de arquitectura.	W	5	30
30. Diseño, implementación y pruebas de páginas basados en prototipos (con excepciones de los casos de uso).	E	7	28
31. Actualización de Modelo de Diseño, Implementación y despliegue.	WE	5	30
32. Elaboración y consecución del plan de pruebas parte de Distribuidor.	E	3	18
33. Evaluación de pruebas realizadas.	W	2	12
34. Evaluación de la etapa.	WE	3	18
35. Planeación de la etapa de transición.	WE	2	12

H/H: Horas Hombre

Res: Responsable

#### 4. Cronograma establecido para ejecución de las tareas.

##### Actividades 1-8

Id	Nombre de tarea	Duración	diciembre					ener
			25/11	02/12	09/12	16/12	23/12	30/12
1	Diagramación e Implementación de prototipos de interfaces para usuarios.	5 días		■				
2	Actualización de modelo de casos de uso parte visitante y registrado.	5 días		■				
3	Diseño, implementación y pruebas de paginas basadas en prototipos (sin excepciones c	8 días		■	■			
4	Actualización de bases de datos preliminar.	2 días			■			
5	Actualización del modelo de análisis y revisión preliminar de arquitectura.	5 días				■		
6	Diseño, implementación y pruebas de paginas basadas en prototipos (con excepciones	10 días				■	■	
7	Actualización de modelo de diseño, implementación y despliegue.	5 días					■	
8	Elaboración y consecución de pruebas parte visitante y registrado	3 días					■	

##### Actividades 9- 16 (incluido periodo de vacaciones)

Id	Nombre de tarea	Duración	enero					f
			30/12	06/01	13/01	20/01	27/01	
9	Diagramación e Implementación de prototipos de interfaces para Administrador.	5 días		■				
10	Actualización de modelo de casos de uso parte Administrador.	5 días		■				
11	Diseño, implementación y pruebas de paginas basadas en prototipos (sin excepciones c	8 días		■	■			
12	Actualización de bases de datos preliminar.	2 días			■			
13	Actualización del modelo de análisis y revisión preliminar de arquitectura.	5 días				■		
14	Diseño, implementación y pruebas de paginas basadas en prototipos (con excepciones	10 días				■	■	
15	Actualización de modelo de diseño, implementación y despliegue.	5 días					■	
16	Elaboración y consecución de pruebas parte Administrador.	3 días					■	

### Actividades 17 – 24

Id	Nombre de tarea	Duración	febrero				
			27/01	03/02	10/02	17/02	24/02
17	Diagramación e Implementación de prototipos de interfaces para Suscriptor.	5 días		■			
18	Actualización de modelo de casos de uso parte Suscriptor.	5 días		■			
19	Diseño, implementación y pruebas de paginas basadas en prototipos (sin manejo de sesiones)	7 días		■	■		
20	Actualización de bases de datos preliminar.	2 días			■		
21	Actualización del modelo de análisis y revisión preliminar de arquitectura.	5 días				■	
22	Diseño, implementación y pruebas de paginas basadas en prototipos (con excepciones de seguridad)	7 días			■	■	
23	Actualización de modelo de diseño, implementación y despliegue.	5 días					■
24	Elaboración y consecución de pruebas parte Suscriptor.	3 días					■

### Actividades 25 – 35

Id	Nombre de tarea	Duración	marzo				
			24/02	03/03	10/03	17/03	24/03
25	Diagramación e Implementación de prototipos de interfaces para Distribuidor.	5 días	■				
26	Actualización de modelo de casos de uso parte Distribuidor.	5 días	■				
27	Diseño, implementación y pruebas de paginas basadas en prototipos (sin manejo de sesiones)	7 días	■	■			
28	Actualización de bases de datos preliminar.	2 días		■			
29	Actualización del modelo de análisis y revisión preliminar de arquitectura.	5 días			■		
30	Diseño, implementación y pruebas de paginas basadas en prototipos (con excepciones de seguridad)	7 días		■	■		
31	Actualización de modelo de diseño, implementación y despliegue.	5 días				■	
32	Elaboración y consecución de pruebas parte Distribuidor.	3 días				■	
33	Evaluación de pruebas realizadas.	2 días					■
34	Evaluación de la etapa.	3 días					■
35	Planeación de la etapa de transición.	2 días					■

## 5. Descripción de los recursos requeridos y las respectivas ventanas de disponibilidad.

Se utilizaran todos los recursos mencionados en el subproducto complementario 1

- Como entorno de desarrollo se establece el denominado **PC1** (100% de disponibilidad en las actividades)

Actividades involucradas (según listado de iteraciones punto 2):

Actividades 1 – 3 – 4 – 6 – 7 -8 - 9 – 11 – 12 – 14 – 15 -16 - 17 – 19 – 20 -22 –  
23 -24 - 25 - 27 – 28 – 30 – 31 – 32

- Como apoyo a la labor de documentación se utilizara el **PC2** (50% de disponibilidad en las actividades)

Actividades involucradas (según listado de iteraciones punto 2):

Actividades 2 – 5 – 8 - 10 – 13 – 16 - 18 – 21 – 24 – 26 – 29 – 32 – 33 – 34 - 35

## **6. Estimación de los costos de construcción del prototipo operacional inicial.**

Tomando como base las normas de depreciación de los recursos vigente al interior de la universidad y el tiempo estimado para esta etapa tenemos:

14 semanas de duración

Horas estimadas de trabajo:

Desarrolladores 30 horas c/u por semana (1.5 puntos c/u)

Director del proyecto 2 horas por semana (2.5 puntos)

Asesores 2 horas en total por semana

PC1 costo ponderado \$1'200.000 uso 30 horas por semana

PC2 costo ponderado \$1.200.000 uso 40 horas por semana

Costo punto \$6.134

Desarrolladores	\$ 7.728.840
Director	\$ 429.380
Asesores	\$ 429.380
PC1	\$ 522.720
PC2	\$ 522.720
Otros	\$ 200.000
<b>TOTAL</b>	<b>\$ 9.653.040</b>

**Subproducto Complementario No. 3**  
**Lista de Comprobaciones para la fase de Creación del Servicio**

CRITERIO	SI	NO
<p><b>1. Satisfacción adecuada de los requerimientos establecidos por los Casos de Uso</b></p> <p>1. ¿Satisface cada iteración los requerimientos funcionales del conjunto de casos de uso respectivo?</p> <p>2. ¿Se satisfacen en cada iteración los requerimientos no funcionales correspondientes?</p>	3  3	
<p><b>2. Suficiencia del material de soporte para la fase de validación del servicio</b></p> <p>1. ¿Se tiene suficiente material escrito de soporte para los usuarios finales (guías de usuario, texto de ayuda, notas para entrega del producto, manuales de usuario, manuales de operador, etc.)?</p> <p>2. ¿Se tiene suficientes ayudas (diapositivas, notas, ejemplos, tutoriales, etc.) para la capacitación de los usuarios?</p>	3  3	
<p><b>3.- Adecuación del prototipo operacional inicial</b></p> <p>1. ¿Se tiene preparado un plan para la etapa de transición?</p> <p>2. ¿Se tienen los ejecutables en si que provean la capacidad operacional?</p> <p>3. ¿Se tiene los modelos del sistema completos?</p> <p>4. ¿Se tiene la descripción de arquitectura actualizada</p> <p>5. ¿Refleja el caso de negocio la situación al final de esta fase?</p>	3 3 3 3 3	