



Universidad de Valladolid

E. U. DE INFORMÁTICA (SEGOVIA)
Grado en Ingeniería Informática de Servicios y Aplicaciones

TIENDA VIRTUAL DE ELECTRODOMÉSTICOS

“Electrokuisland”

Alumno: Alberto Pastor Muñoz

Tutor: Fernando Díaz Gómez

| | |
|---|--------|
| 1. IDENTIFICACIÓN DEL PROYECTO..... | 12 |
| 2. VISION GENERAL DEL DOCUMENTO..... | 12 |
| 3. MOTIVACIÓN..... | 12 |
| 4. OBJETIVOS..... | 12 |
| 5. ARQUITECTURA Y ENTORNO DEL SISTEMA..... | 14 |
| 6. ASPECTOS METODOLÓGICOS..... | 15 |
| 7. CONSIDERACIONES..... | 16 |
| 7.1. CONSIDERACIONES VARIAS..... | 16 |
| 7.2. CONSIDERACIONES SOBRE LA BASE DE DATOS..... | 19 |
| 7.3. CONSIDERACIONES SOBRE EL SERVIDOR..... | 19 |
| 7.4. DIFICULTADES ENCONTRADAS..... | 19 |
| 8. PLANIFICACIÓN Y PRESUPUESTO..... | 20 |
| 8.1. PLANIFICACIÓN: FASES DE TRABAJO Y ESTIMACIÓN TEMPORAL..... | 20 |
| 8.2. PRESUPUESTO: ESTIMACIÓN MEDIANTE COCOMO..... | 20 |
| 8.3. PRESUPUESTO: COSTES DEL HARDWARE..... | 22 |
| 8.4. PRESUPUESTO: COSTES DEL SOTWARE..... | 22 |
| 8.5. PRESUPUESTO: DESARROLLO DEL PROYECTO..... | 23 |
| 8.6. PRESUPUESTO: TOTALES..... | 23 |
| 9. AMPLIACIONES..... | 23 |
| 9.1. GESTION DE VENDEDORES MÁS COMPLETA..... | 23 |
| 9.2. CONTROL DE STOCKS..... | 24 |
| 9.3. GESTIÓN DE CONTABILIDAD..... | 24 |
| 9.4. CREACIÓN DE CAMPAÑAS DE OFERTAS..... | 24 |
| 9.5. OFERTAS DINÁMICAS PERSONALIZADAS..... | 24 |
| 9.6. DIFERENTES SECCIONES, DIFERENTES ADMINISTRADORES..... | 24 |
| 9.7. SUBIR FOTOS DE USUARIO AL SERVIDOR..... | 24 |
| 9.8. ENVÍO DE MAILS..... | 24 |
| 9.9. SEGUIMIENTO DE PEDIDOS..... | 25 |
| 10. CONCLUSIONES..... | 25 |
| 11. ACRÓNIMOS..... | 25 |
| 12. BIBLIOGRAFÍA..... | 26 |
| 12.1. Comercio electrónico:..... | 26 |
| 12.2. Metodología Orientada a Objetos:..... | 26 |
| 12.3. DHTML:..... | 26 |
| 12.4. Sistema Gestor De Base De Datos:..... | 26 |
| 12.5. Lenguaje Orientado a Objetos:..... | 27 |
| 12.6. Servidor Apache:..... | 27 |
| SECCIÓN 2 : MANUAL TÉCNICO..... | 28 |
| REQUISITOS ESPECÍFICOS..... | 30 |
| 1. GESTIÓN DE INFORMACIÓN DE CLIENTES..... | 30 |
| 2. GESTIÓN DE INFORMACIÓN DE VENDEDORES..... | 31 |
| 3. GESTIÓN DE INFORMACIÓN DEL ADMINISTRADOR..... | 32 |
| DIAGRAMA DE CASOS DE USO DEL SISTEMA..... | 35 |
| ESPECIFICACIÓN DE LOS CASOS DE USO..... | 36 |
| 1. Caso de uso “Consultar catálogo”..... | 36 |
| 2. Caso de uso “Gestionar datos cliente” include (Validar usuario)..... | 37 |
| 2.1. Subcaso de uso “Consulta datos”..... | 37 |
| 2.2. Subcaso de uso “Modifiación datos”..... | 38 |
| 3. Caso de uso “Alta cliente”..... | 39 |
| 4. Caso de uso “Baja cliente”..... | 39 |
| 5. Caso de uso “Gestión carrito de compra”..... | 40 |
| 6. Caso de uso “Comprar”..... | 41 |
| 7. Caso de uso “Pagar”..... | 42 |
| 8. Caso de uso “Validar usuario”..... | 42 |
| 9. Caso de uso “Gestionar datos vendedor”: include (Validar usuario)..... | 43 |
| 9.1. Subcaso de uso “Consulta datos vendedor”..... | 44 |

| | |
|---|-----|
| 9.2. Subcaso de uso “Modificación datos vendedor” | 45 |
| 9.3. Subcaso de uso “Baja vendedor” | 45 |
| 10. Caso de uso “Gestionar persona contacto”: include(Validar usuario) | 46 |
| 10.1. Subcaso de uso “Alta persona contacto” | 46 |
| 10.2. Subcaso de uso “Consulta persona contacto” | 47 |
| 10.3. Subcaso de uso “Modificación persona contacto” | 48 |
| 10.4. Subcaso de uso “Baja persona contacto” | 49 |
| 11. Caso de uso “Gestionar productos Vendedor”: include (Validar usuario) | 49 |
| 11.1. Subcaso de uso “Alta producto” | 50 |
| 11.2. Subcaso de uso “Consulta producto” | 51 |
| 11.3. Subcaso de uso “Modificación producto” | 51 |
| 11.4 Subcaso de uso “Baja producto” | 52 |
| 12. Caso de uso “Gestionar productos”: include (Validar usuario) | 53 |
| 12.1. Subcaso de uso “Alta producto” | 53 |
| 12.2. Subcaso de uso “Consulta producto” | 54 |
| 12.3. Subcaso de uso “Modificación producto” | 55 |
| 12.4 Subcaso de uso “Baja producto” | 55 |
| 12. Caso de uso “Gestionar categorías productos”: include (Validar usuario) | 56 |
| 13.1. Subcaso de uso “Alta categoría” | 57 |
| 13.2. Subcaso de uso “Consulta categoría” | 58 |
| 13.3. Subcaso de uso “Modificación categoría” | 58 |
| 13.4 Subcaso de uso “Baja categoría” | 59 |
| 14. Caso de uso “Gestionar datos Administrador”: include (Validar usuario) | 60 |
| 14.1. Subcaso de uso “Consulta datos administrador” | 60 |
| 14.2. Subcaso de uso “Modificación datos administrador” | 61 |
| DIAGRAMAS DE INTERACCIÓN CONCEPTUALES | 63 |
| DIAGRAMA DE CLASES FINAL | 97 |
| DISEÑO DETALLADO: DIAGRAMA DE PAQUETES | 102 |
| ANÁLISIS Y DISEÑO DE LA BASE DE DATOS | 102 |
| PASO A TABLAS DEL MODELO E-R | 106 |
| DICCIONARIO DE DATOS | 107 |
| | |
| SECCIÓN 3: MANUAL DE USUARIO | 111 |
| | |
| 1. MANUAL DE INSTALACIÓN | 113 |
| PASO 1: Instalación de Apache 2.2.22 | 113 |
| PASO 2: Instalación de PHP5 | 113 |
| PASO 3: Instalación de MySQL | 116 |
| 2. MANUAL DE LA APLICACIÓN | 117 |
| 2.1. SECCIÓN CLIENTE NO REGISTRADO | 117 |
| 2.2. REGISTRO DE CLIENTES | 117 |
| 2.3. BÚSQUEDA DE PRODUCTOS | 118 |
| 2.4. CATÁLOGO DE PRODUCTOS | 119 |
| 2.5. IDENTIFICACIÓN DE USUARIO | 120 |
| 2.6. SECCIÓN DE CLIENTE | 121 |
| 2.7. SECCIÓN DE VENDEDOR | 125 |
| 2.8. SECCIÓN DE ADMINISTRADOR | 129 |

*SECCIÓN 1 : MEMORIA DEL
PROYECTO WEB*

1. IDENTIFICACIÓN DEL PROYECTO.

Título: Tienda Virtual “*Electrokuiland*”.

Autor: Alberto Pastor Muñoz.

Fecha: 25-06-2013.

2. VISION GENERAL DEL DOCUMENTO.

La documentación consta de tres secciones y un CD-ROM, que contienen la siguiente información:

- Memoria del proyecto.

En esta sección se detallan cuestiones generales sobre la aplicación, tales como criterios de desarrollo, tecnología utilizada y consideraciones varias.

- Manual técnico.

En esta sección se tratan de manera detallada aspectos sobre los requisitos específicos, y pruebas que se han realizado para el desarrollo de la aplicación.

- Manual de usuario e instalación.

En esta sección se explica de manera clara y precisa los pasos necesarios para poder tener corriendo la aplicación en el servidor, su puesta en funcionamiento y los pasos para el manejo de la misma.

- CD-ROM.

En él se adjuntan los siguientes directorios:

1.- “htdocs”: contiene la carpeta con la aplicación, que se debe copiar al servidor.

2.- “Documentación”: contiene el presente documento en formato PDF de la aplicación.

3. MOTIVACIÓN.

Los avances tecnológicos de la computación y las comunicaciones por Internet han ido evolucionando para facilitar las actividades de las personas, así como la forma de hacer negocios.

Internet se ha consolidado como la plataforma ideal para el desarrollo de pequeñas y grandes empresas que quieren iniciar o mejorar su presencia en Internet con un diseño acorde a su imagen corporativa o para crear un sitio de referencia para los clientes que buscan sus productos "para comprarlos".

El usuario del supermercado virtual tiene una clara necesidad de ahorro de tiempo, y trata de satisfacerla precisamente eludiendo la visita a la tienda física.

4. OBJETIVOS.

El objetivo del proyecto es el desarrollo de un portal web de una tienda de electrodomésticos, que sea accesible desde Internet.

El sistema estará administrado por tres perfiles de usuarios, diferenciándose tres perfiles de usuario: usuario cliente, usuario vendedor y usuario administrador. Para el caso de usuario cliente se cubre la posibilidad de que el cliente sea tanto particular como empresa.

Las operaciones para cada perfil de usuario serán las siguientes:

Usuario cliente:

Usuario cliente no registrado,

- Podrá navegar por la web.
- Realizar consultas acerca de los diferentes productos.
- Realizar búsquedas acerca de los diferentes productos.

- Posibilidad de registro en el sistema, previa validación de sus datos personales.

Usuario cliente registrado, además de lo anterior:

- Se le permitirá gestionar sus datos (modificarlos y/o consultarlos).
- Realizar pedidos.
- Podrá darse de baja del sistema.

Usuario vendedor:

- Realizar consultas acerca de los diferentes productos.
- Realizar búsquedas acerca de los diferentes productos.
- Se le permitirá gestionar sus datos (modificarlos y/o consultarlos).
- Podrá llevar a cabo la gestión de los productos que en principio podría suministrar a la tienda.
- Podrá llevar a cabo la gestión de la persona de contacto con la que cuenta para que le soliciten el suministro de uno o varios productos.
- Podrá darse de baja del sistema.

Usuario administrador:

- Realizar consultas acerca de los diferentes productos.
- Realizar búsquedas acerca de los diferentes productos.
- Gestión de categorías de productos.
- Gestión de productos.

Además de los indicados, como objetivos de primer orden se considerarán los siguientes puntos:

- **Ayuda:** El sistema guiará en diferentes aspectos al usuario de forma que el navegar por las diferentes páginas de la aplicación sea lo más fácil posible, poniendo especial énfasis en cuestiones como realizar un pedido o la gestión de datos personales.
- **Entorno amigable:** El sistema ha de tener un entorno sencillo y amigable, para proporcionar a los usuarios la mayor comodidad y facilidad posibles.
- **Ejecución en Internet:** El sistema ha de poder ejecutarse en Internet, ya que si no, no sería posible el uso del comercio electrónico.

5. ARQUITECTURA Y ENTORNO DEL SISTEMA.

El sistema se desarrollará bajo la arquitectura cliente-servidor, basada en el paso de mensajes entre una máquina (cliente) que solicita peticiones de servicio a otra en la que residen los datos y los programas de aplicación (servidor).

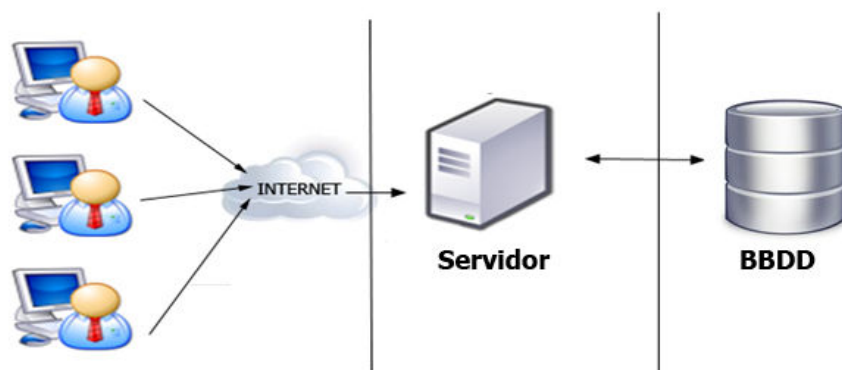


Figura 5. Arquitectura cliente-servidor.

En esta arquitectura la capacidad de proceso está repartida entre el servidor y los clientes. Se pueden distinguir 3 capas o niveles:

- Gestor de Base de Datos (Nivel de almacenamiento).
- Procesador de aplicaciones o reglas del negocio (Nivel lógico).
- Interface del usuario (Nivel de presentación).

La infraestructura sobre la que se va a implantar el sistema será la siguiente:

Puestos Cliente:

El entorno hardware del cliente deberá estar formado por un equipo con las características mínimas actuales.

El entorno software del cliente estará compuesto por lo siguiente:

- Sistema operativo Microsoft Windows XP Profesional ó similar.
- Navegador: Microsoft Internet Explorer 6.0 ó similar, como Mozilla 1.7.7.

Puesto Servidor:

El entorno hardware del servidor deberá tener los siguientes requisitos mínimos:

- Pentium IV a 3.2 Ghz.
- 512 MB de memoria RAM.
- 500 MB de espacio libre en el disco duro.
- Unidad de CD-ROM o DVD.

El entorno software del servidor estará compuesto por lo siguiente:

- Sistema operativo Microsoft Windows XP Profesional.
- Servidor web Apache 2.0.5.
- Módulo PHP 5.5 ó superior.

- Sistema gestor de base de datos MySQL 5.0.

El servidor de la aplicación utiliza las siguientes tecnologías:

- **HTML.** El lenguaje HTML (*HyperText Markup Language*) se usa para la creación de páginas web en Internet, HTML es un lenguaje muy sencillo que permite definir documentos de hipertexto a base de ciertas etiquetas que marcan partes del documento dándoles una estructura o jerarquía, y que permite presentar el texto de una manera ordenada y agradable, con enlaces que conducen a otros documentos o fuentes de información relacionadas, y con inserciones multimedia.
- **Javascript.** Es un "lenguaje de script" orientado a documento. Surgió para extender las capacidades del lenguaje HTML. Los "scripts" de *Javascript* pueden ser introducidos en páginas web. Esta tecnología permite dar respuesta a eventos iniciados por el usuario, tales como la entrada de un formulario o pinchar un determinado enlace. Esto sucede de tal forma que la verificación y validación de datos del usuario se desarrollan en el puesto cliente, no siendo necesario la transmisión de estos datos al servidor para su validación. Las entradas, después de chequearse en el cliente pueden ser transmitidas al servidor.
- **PHP.** Acrónimo recursivo de "*PHP: Hypertext Preprocessor*", originado inicialmente del nombre PHP Tools, o *Personal Home Page Tools*) es un lenguaje de programación interpretado. Se utiliza entre otras cosas para la programación de páginas web activas, y se destaca por su capacidad de mezclarse con el código HTML. Aunque su creación y desarrollo se da en el ámbito de los sistemas libres, bajo la licencia GNU, existe además un compilador comercial denominado (Zend Optimizer).
El uso más extendido del lenguaje PHP, es el de formar parte de una página web con el propósito de agregarle características dinámicas. Su interpretación y ejecución se da en el servidor en el cual se encuentra almacenada la página y el cliente solo recibe el resultado de la ejecución. Cuando el cliente hace una petición al servidor para que le envíe una página web, enriquecida con código PHP, el servidor interpretará las instrucciones mezcladas en el cuerpo de la página y las sustituirá con el resultado de la ejecución antes de enviar el resultado a la computadora del cliente. Además es posible utilizarlo para generar archivos PDF, Flash o JPG, entre otros.
- **SQL.** El lenguaje SQL (*SQL, Structured Query Language*) sirve para interactuar con las bases de datos relacionales. Se ha optado por diseñar e implementar un modelo de datos relacional debido a sus características de robustez y consistencia.

6. ASPECTOS METODOLÓGICOS.

La herramienta utilizada en el diseño conceptual de la base de datos es el modelo de datos Entidad-Relación (E/R), para después pasar el modelo conceptual obtenido al modelo lógico, siguiendo las pautas que marca el Modelo Relacional (MR). En cuanto al lenguaje de definición de datos para la especificación del modelo físico se utilizará el estándar SQL aplicado a *MySQL*. El diseño del diagrama Entidad-Relación será realizado con *DBDesigner 4.0*, al ofrecer un entorno más amigable que otras herramientas como *EASY CASE*.

Para la fase de implementación del sistema se utilizará código *HTML*, *JavaScript* y *PHP* en su versión 5.1. Cabe destacar que esta versión está provista de una mayor orientación a objetos que versiones anteriores como son *PHP 4* y *PHP 3*, con notables carencias en cuanto a clonación de objetos se refiere. Para solventar este problema *PHP 5* hace uso de los manejadores de objetos (Object handles), que son una especie de punteros que apuntan hacia los espacios en memoria donde residen los objetos. Cuando se asigna un manejador de objetos o se pasa como parámetro en una función, se duplica el propio object handle y no el objeto en si.

Además de lo expuesto anteriormente, *PHP 5* incorpora las siguientes características:

- Nombres fijos para los constructores y destructores.
- Acceso public, private y protected a propiedades y métodos.

- Posibilidad de uso de interfaces.
- Métodos y clases final.
- Atributos y métodos static.
- Clases y métodos abstractos.
- Funciones que especifican la clase que reciben por parámetro.

7. CONSIDERACIONES.

En este punto se desarrollarán cuestiones de diversa índole que se consideran relevantes y a tener en cuenta para una mejor comprensión de lo implica el desarrollo de una herramienta software de estas características.

Para las consideraciones se hará una división en etapas así como el desarrollo del proyecto lo ha requerido, dividiéndose en distintos apartados para cada una de estas etapas.

7.1. CONSIDERACIONES VARIAS.

7.1.1. JERARQUÍA DE USUARIOS.

Para el análisis de la aplicación se tuvo en cuenta que aunque no registrado, un usuario “nace” en el momento en que visualiza la página principal de la aplicación. A partir de ahí podrá realizar actividades propias de usuario registrado o no, pero igualmente podrá interactuar con la aplicación sin necesidad de estarlo. Es por esto que se creó una jerarquía de usuarios, en la que operaciones comunes para toda la jerarquía serán realizadas por el usuario ancestro y operaciones particulares, por cada uno de los subtipos de usuarios.

7.1.2. PAGO DEL PEDIDO REALIZADO.

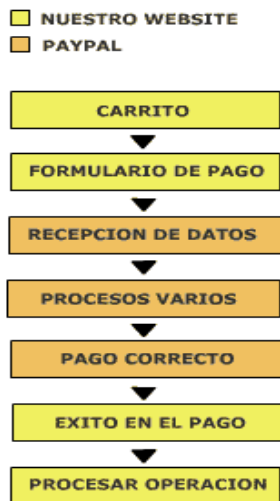
Una manera de que el cliente realice el pago antes de abandonar nuestra página es mediante el uso de una pasarela de pagos. Esto último lo conseguiremos integrando el carrito de la compra con una Plataforma de Pagos como Paypal. Para ello debemos sacar previamente una cuenta en <https://www.paypal.com/>, donde nos pedirán, entre otros requisitos, una dirección de correo electrónico que nos servirá para que el sistema de Paypal nos identifique como vendedores cuando un comprador realice una compra desde nuestro website.

Debemos tener en cuenta que, en el ejemplo que presentaremos, mientras se realice el proceso de pago el comprador oscilará entre nuestro servidor y el servidor de Paypal. Independientemente del protocolo que use nuestro servidor para las transacciones web (lo más probable es que use protocolo HTTP), Paypal usa protocolo HTTPS, que es más apropiado para el tráfico de información sensible, como por ejemplo números de tarjetas de crédito.

El esquema de trabajo que utilizaremos es el siguiente: en nuestro website procesaremos la selección de los artículos y calcularemos los precios. Luego derivaremos esa información y a nuestro cliente a Paypal. Ya en Paypal y bajo el paraguas del protocolo HTTPS, el cliente se identificará, verá los productos que ha seleccionado en nuestro website junto con el importe de los mismos, y elegirá alguna alternativa de pago o suspenderá la compra o el sistema le indicará que no tiene fondos suficientes o una larga lista de etcéteras. En cualquiera de todos estos casos, al terminar, Paypal redireccionará nuevamente al comprador a nuestro website.

En resumen, en el esquema descrito pueden presentarse básicamente dos alternativas: que el pago se concrete o que el pago no se concrete.

Primera alternativa: Se concreta el pago.



Si el pago se realiza correctamente, Paypal derivará al cliente a una página de nuestro site que nosotros le indiquemos para los casos de éxito en el pago, devolviéndonos al mismo tiempo algunas variables que nos servirán para realizar nuestros procesos: por ejemplo, si vendimos un acceso a un servicio web por 3 meses, actualizar la base de datos dándole permisos al comprador por ese período o enviar un mail a alguien diciéndole que debe entregar x detalle de productos a x cliente, descontar del stock, etc.

Segunda alternativa: No se concreta el pago.



Si el pago no se concreta, Paypal devolverá al cliente a una página de nuestro site que nosotros le

indiquemos para los casos de fracaso en el pago. En esta oportunidad no deberemos procesar nada: sólo deberemos notificar al cliente que por alguna razón el pago no pudo concretarse.

7.1.3. EL PATRÓN MVC.

La implementación de la aplicación se ha realizado siguiendo el criterio del patrón de diseño *MVC* (*Modelo-Vista-Controlador*). El patrón MVC descompone una aplicación interactiva en tres grandes bloques:

- El *modelo* contiene los datos y la funcionalidad de la aplicación. Es independiente de la representación de los datos.
- Las *vistas* muestran la información al usuario de una cierta forma. Existen todas las que se necesite definir.
- Cada *vista* tiene un *controlador* asociado. Los controladores reciben entradas en forma de eventos que responden a mandos realizados por el usuario a través del ratón o del teclado. El control traduce estos eventos a peticiones a la *vista* o al *modelo*.

7.1.4. DIVERSIDAD DE CLIENTES.

Debido a que los tipos de clientes que soporta la aplicación son dos (cliente empresa y cliente particular) los formularios de recogida de datos para el registro como cliente no serán iguales, lo que implica que tampoco lo serán las validaciones previas de datos y su almacenamiento en el sistema será particular para cada uno.

El usuario administrador será único en el sistema, esto se debe a que según la especificación de requisitos considerada sólo deberá existir una persona que utilice la aplicación con permisos de “súper-usuario”, con fines administrativos.

7.1.5. MÉTODOS DE CLASE.

Todas aquellas clases que necesitaron de un listado de elementos en la aplicación y una visualización individual por pantalla incorporaron a su implementación métodos de clase concebidos para cumplir esa función de visualización o listado de elementos almacenados en la base de datos, sin necesidad de instanciar objetos de dicha clase para realizarlo.

7.1.6. CLASE PARA EL ACCESO A LA BASE DE DATOS.

Esta clase fue incluida ya en el diseño de la aplicación para realizar el acceso a la base de datos. Es una clase a la que todas las demás clases de la aplicación tienen acceso para la ejecución de sus métodos, que serán todos ellos métodos de clase.

7.1.7. CLASE CONTROLADOR.

La clase controlador fue concebida para que el control sobre las demás clases estuviera centralizado en una clase. Esta clase luego no se llevó a efecto como tal, puesto que el controlador quedó disgregado a lo largo de todo el código de la aplicación, como se muestra a continuación:

```
$letra = $_POST["letra"]
if (($letra < "A") || ($letra > "Z")) {
    include "letranovalida.php";
} else {
    $productos = obtener_productos ($letra);
    include "listadoproductos.php";}
```

`letranovalida.php` y `listadoproductos.php` son parte de la vista. El segundo fichero hace uso de la variable `$productos` para mostrar el listado. Como se puede ver, el controlador no se mete con HTML; se limita a obtener el listado de la base de datos y pasárselo, tal cual, a la vista.

De este modo, el código del modelo de negocio está perfectamente separado del código dedicado a crear HTML (interfaz), apreciando el papel que juega el controlador.

7.2. CONSIDERACIONES SOBRE LA BASE DE DATOS.

7.2.1. RELACIONES 1:N Y 1:1.

Las relaciones de tipo 1:N y 1:1 con atributos se transformaron en tablas como lo harían las de tipo N:M. Se optó por esta opción para no perder semántica y en previsión de que en un futuro se pudieran convertir en tablas N:M.

7.2.2. IDENTIFICADORES PRINCIPALES AUTONUMÉRICOS.

Los identificadores principales de determinadas tablas que escapan al control administrativo del administrador como son los pedidos. De esta forma será la base de datos por su diseño la encargada de asignar identificadores a las tuplas y no el administrador.

Esto no ocurre así con las tablas que entran dentro de la tarea de gestión y administración que ejerce el administrador, como son las categorías de productos, etc cuyos identificadores principales serán asignados por el administrador en el momento de su creación.

7.2.3. DISTINTOS PERMISOS , DISTINTAS CONEXIONES.

La base de datos de la aplicación fue diseñada con usuarios con permisos distintos. Un usuario cliente potencial, que visite la página en la sección del catálogo de productos no podrá utilizar la conexión asociada a los permisos de creación, borrado y modificación de datos, para que se muestren los productos por pantalla. Esto es así por motivos de seguridad y por seguir un orden lógico, pues tales permisos deberían estar asignados a usuarios registrados, para la administración de datos diversos.

7.3. CONSIDERACIONES SOBRE EL SERVIDOR.

7.3.1. PERMISO DE LISTADO DENEGADO.

El directorio raíz fue configurado para que no se permita el listado de contenido del mismo en el navegador, de forma que no se pueda a priori extraer contenido. Esto es así al inhabilitar la opción *Indexes* del fichero de configuración de Apache. Mediante esta directiva el módulo *mod_autoindex* respondería con un listado del contenido al no encontrar un fichero *index* especificado.

7.3.2. SIRVIENDO SIN HOST VIRTUALES.

Los *VirtualHost* son mecanismos de los que Apache nos provee para mantener múltiples dominios o *hostnames* en una misma máquina. Para el desarrollo del proyecto no fue necesario tener en cuenta este concepto en el fichero de configuración de Apache, puesto que sólo es un dominio el que tenemos que mantener en nuestro caso.

7.4. DIFICULTADES ENCONTRADAS.

En el desarrollo de este proyecto software, la familiarización con las tecnologías involucradas y el aprendizaje de sus características para su utilización resultó a veces un gran inconveniente. Puesto que la experiencia en la utilización de dichas tecnologías era escasa, se encontraron dificultades que se fueron superando algunas veces con la ayuda de la bibliografía suministrada, y otras con la consulta de distintos foros, etc..

Puede que el mayor problema haya sido el no saber a priori el trabajo que supondría la realización del presente proyecto, lo que llevó en un principio a hacer una estimación “a ciegas”. El cumplimiento de este calendario establecido fue lo que en gran parte impulsó un mayor tesón en el desarrollo del proyecto, siendo clave la coordinación para la realización de la implementación y documentación para la obtención de un producto final que cumpla con lo establecido en un principio. Todo ello ha permitido que el tiempo de desarrollo no se alargara más.

Finalmente cabe destacar que una parte importante del esfuerzo ha sido dedicado a la interfaz de la aplicación, intentando que esta fuera en todo momento clara, sencilla de usar y que tuviera lo imprescindible para que todo tipo de clientes puedan utilizarla sin problemas.

8. PLANIFICACIÓN Y PRESUPUESTO.

En este apartado se realizará una planificación temporal tenida en cuenta para el desarrollo de la aplicación. Posteriormente y a raíz de esto se elaborará un presupuesto del proyecto desarrollado como tal mediante la metodología establecida por COCOMO, obtenido como la suma del importe de los costes sobre los recursos humanos, el porcentaje establecido sobre el total de los costes de hardware y el porcentaje establecido sobre el total de los costes de software. La jornada laboral estimada, considerando un ordenador por persona, es de ocho horas.

8.1. PLANIFICACIÓN: FASES DE TRABAJO Y ESTIMACIÓN TEMPORAL.

Las fases de trabajo establecidas a priori y la planificación supuesta realizadas son las siguientes:

FASES DE TRABAJO:

- Estudio de aplicaciones similares.
- Estudio de las tecnologías involucradas en el proyecto.
- Análisis del sistema mediante metodología orientada a objetos.
- Diseño mediante metodología orientada a objetos.
- Implementación del sistema.
- Realización de pruebas para corrección de posibles errores.
- Generación de documentación.

PLANIFICACIÓN TEMPORAL:

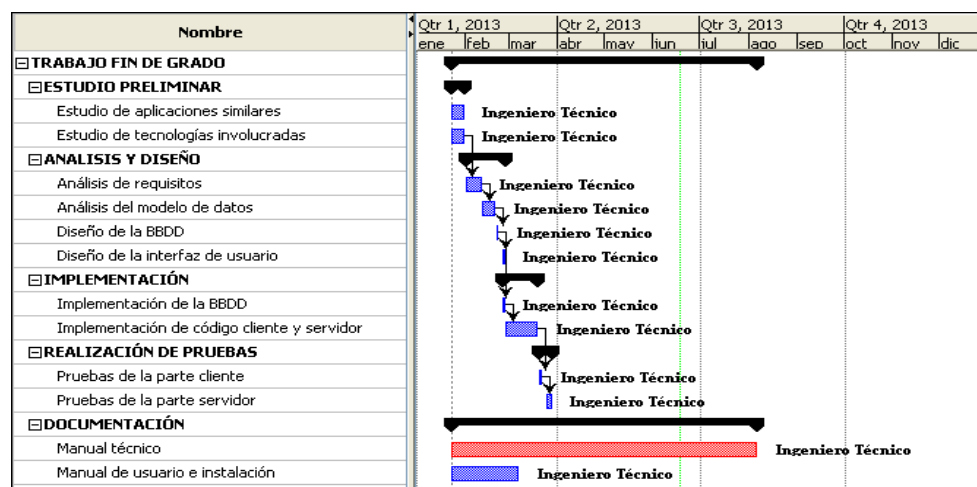


Figura 8.1 Planificación temporal.

8.2. PRESUPUESTO: ESTIMACIÓN MEDIANTE COCOMO.

Una de las tareas de mayor importancia en la planificación de proyectos de software es la estimación, la cual consiste en determinar, con cierto grado de certeza, los recursos de hardware y software, costo, tiempo y esfuerzo necesarios para el desarrollo de los mismos.

Este punto describe el modelo de estimación COCOMO, propuesto por Barry Boehm. Este modelo permite realizar estimaciones en función del tamaño del software, y de un conjunto de factores de coste y de escala, siendo el modelo de estimación de costes más utilizado.

A continuación se muestran los tipos de desarrollo y los valores para las fórmulas de esfuerzo y tiempo:

| MODO DE DESARROLLO | PERSONAS-MES | TIEMPO DE DESARROLLO |
|--------------------|-------------------------------|-----------------------------|
| Orgánico | PM = 3,2 KLDC ^{1,05} | TD = 2,5 PM ^{0,38} |
| Empotrado | PM = 3,0 KLDC ^{1,12} | TD = 2,5 PM ^{0,35} |
| Semi-libre | PM = 2,8 KLDC ^{1,2} | TD = 2,5 PM ^{0,32} |

Tabla 8.2 Alternativas de desarrollo en COCOMO.

Cada modalidad de desarrollo implica diferentes aspectos en cuanto a requisitos, entorno y plazos:

- Orgánico: desarrollado en un entorno estable, con poca innovación técnica con pocas presiones de tiempo y tamaño relativamente pequeño (<50 KLDC).
- Empotrado: desarrollo de software con requisitos muy restrictivos, con gran volatilidad de requisitos, complejo en un entorno con gran innovación técnica.
- Semi-libre: situaciones intermedias entre los dos modos anteriores.

Nuestro caso será el de desarrollo Semi-libre, puesto que a pesar de movernos en un entorno estable y tamaño relativamente pequeño, los plazos a cumplir para poder concluir en fecha el presente TFG hacen que se presente una restricción importante.

Así pues, pasaremos a realizar una estimación mediante COCOMO básico en primera instancia, aplicando sólo valores nominales, para después pasar al COCOMO intermedio, suponiendo una especificación de requisitos más o menos terminada.

COCOMO BÁSICO:

$$\text{Esfuerzo Nominal (PM)} = 2,8 * \text{KLDC } 1,2 = 2,8 * 4,910^{1,2} = 18,89 \approx 19 \text{ Personas/Mes}$$

COCOMO INTERMEDIO:

Para pasar al COCOMO intermedio hay que aplicar un factor para el esfuerzo. Este factor son 15 atributos del proyecto agrupados en cuatro grandes grupos: Atributos del producto, atributos de la computadora, atributos del personal y atributos del proyecto. Cada atributo se mide en varios grados y cada grado toma un valor.

Tras obtener el valor de este factor, de manera inmediata obtendremos el tiempo de desarrollo y el número de personas necesarias para hacerlo.

$$\text{Esfuerzo} = \text{Esfuerzo Nominal} * \text{Factor(es) Ajuste Esfuerzo} = 18,89 * 0,71 \text{ (analistas)} * 0,91 \text{ (experiencia con el tipo de aplicación)} * 0,95 \text{ (experiencia con lenguaje de programación)} * 0,86 \text{ (Calidad programadores)} * 0,91 \text{ (Técnicas modernas de programación)} = 9,07 \approx 9 \text{ personas/mes}$$

$$\text{Tiempo de Desarrollo} = 2,5 * \text{PM}^{0,32} = 2,5 * 9,07^{0,32} = 5,05 \text{ Meses}$$

| FACTORES | Valor de los factores | | | | | |
|---------------------------------------|-----------------------|------|-------|------|----------|-------|
| | Muy bajo | Bajo | Medio | Alto | Muy alto | Extra |
| Fiabilidad requerida | 0,75 | 0,88 | 1,00 | 1,15 | 1,4 | |
| Tamaño de la base de datos | | 0,94 | 1,00 | 1,08 | 1,16 | |
| Complejidad del software | 0,70 | 0,85 | 1,00 | 1,15 | 1,30 | 1,65 |
| Restricciones de tiempo de ejecución | | | 1,00 | 1,11 | 1,30 | 1,66 |
| Restricciones de memoria | | | 1,00 | 1,06 | 1,21 | 1,56 |
| Volatilidad del hardware | | 0,87 | 1,00 | 1,15 | 1,30 | |
| Restricciones de tiempo de respuesta | | 0,87 | 1,00 | 1,07 | | |
| Calidad de los analistas | 1,46 | 1,19 | 1,00 | 0,86 | 0,71 | |
| Experiencia con el tipo de aplicación | 1,29 | 1,13 | 1,00 | 0,91 | 0,82 | |
| Experiencia con el hardware | 1,21 | 1,10 | 1,00 | 0,90 | | |
| Exp. con lenguaje de programación. | 1,14 | 1,07 | 1,00 | 0,95 | | |
| Calidad de los programadores | 1,42 | 1,17 | 1,00 | 0,86 | 0,70 | |
| Técnicas modernas de programación | 1,24 | 1,10 | 1,00 | 0,91 | 0,82 | |
| Empleo de herramientas | 1,24 | 1,10 | 1,00 | 0,91 | 0,83 | |
| Restric. a la duración del proyecto | 1,23 | 1,08 | 1,00 | 1,04 | 1,10 | |

Tabla 8.3. Tabla de factores de coste para COCOMO.

8.3. PRESUPUESTO: COSTES DEL HARDWARE.

Para desarrollar el proyecto se necesitarán medios Hardware y Software cuyo coste proporcional al uso que se le dará en el proyecto hay que introducir en el presupuesto.

El hardware utilizado para el desarrollo del proyecto ha contado con las siguientes características:

- Ordenador para el desarrollo del sistema, la implantación y pruebas del mismo, la generación de la documentación y para el análisis.
- Conexión a internet para obtención de información y descarga de software.
- Impresora para imprimir la documentación e información.

| HARDWARE | USO (%) | COSTE TOTAL (€) | COSTE (€) |
|---------------------|---------|-----------------|--|
| Ordenador personal | 90 % | 1200€ | 1.080 € |
| Conexión a internet | 100 % | 50 €/mes | 50 €/mes * 7,54 meses /proyecto = 720 € / proyecto |
| Impresora | 20% | 70€ | 14 € / proyecto |
| TOTAL: | | | 1814,00 € |

8.4. PRESUPUESTO: COSTES DEL SOTWARE.

En cuanto a software se refiere, se utilizaron las siguientes herramientas con sus costes asociados:

| SOFTWARE | USO (%) | COSTE TOTAL (€) | COSTE (€) |
|---|---------|-----------------|-----------|
| Windows 7 Ultimate | 100% | 227,63 € | 227,63 € |
| Microsoft Office 2007 (Versión completa) | 10% | 709,00 € | 70,9 € |
| OpenProject | 5% | Gratuito | N/A |
| Rational Rose Enterprise Edition 7.0 (licencia de prueba 30 días) | 15% | Gratuito | N/A |

| | | | |
|----------------|-----|----------|-----------------|
| Apache | 40% | Gratuito | N/A |
| MySQL | 20% | Gratuito | N/A |
| PHP | 40% | Gratuito | N/A |
| Smarty | 40% | Gratuito | N/A |
| NetBeans 6.9.1 | 50% | Gratuito | N/A |
| TOTAL: | | | 297,53 € |

8.5. PRESUPUESTO: DESARROLLO DEL PROYECTO.

En la siguiente tabla se encuentra un desglose de las tareas necesarias para llevar a cabo el proyecto y la duración estimada de las mismas en horas de trabajo. Teniendo en cuenta que se ha estimado que los trabajos durarán 5 meses, y que cada día se pretende trabajar 8 horas (solo días laborables, de lunes a viernes, festivos no incluidos):

Tenemos 5,05 meses lo que resulta en **808 horas**, estimando 160 horas mensuales de trabajo (8 horas /día * 5 días /semana * 4 semanas /mes)

Multiplicando por el sueldo por hora de un Titulado de Grado en Informática, se obtiene el coste total de la mano de obra:

Estableciendo un precios de 60 €/hora para el análisis y de 40 €/hora para la programación, estimaremos una media de 50 €/hora.

| | TIEMPO | | COSTE |
|------------------|--------|-------|-------------|
| Ingeniero | 808,8 | HORAS | 50 € / Hora |

TOTAL: 40.400 €

8.6. PRESUPUESTO: TOTALES.

La estimación del presupuesto total es la suma de los presupuestos que hemos estimado anteriormente.

| | COSTE |
|--------------|--------------------|
| Hardware | 1.814,00 € |
| Software | 297,53 € |
| Desarrollo | 40.400,00 € |
| TOTAL | 42.511,53 € |

9. AMPLIACIONES.

Una característica fundamental de una herramienta software es la posibilidad de que se deje abierta a la realimentación, es decir, que no sea una herramienta cerrada, sin posibilidad de dar cabida a modificaciones o ampliaciones que puedan surgir.

En este punto se desarrollarán someramente algunas de las ampliaciones que podrían realizarse sobre la aplicación, explicando brevemente las características de cada una de ellas.

9.1. GESTION DE VENDEDORES MÁS COMPLETA.

La sección dedicada a vendedores podría tener una función más definida e importante en la aplicación. En el sistema la información de vendedores servirá al administrador para tener un listado de vendedores que suministran determinados productos de su interés, y también las personas a través de las cuales se pondría en contacto con esos vendedores.

La ampliación posible en este punto sería que los vendedores comunicaran mediante una solicitud al sistema su interés por ingresar como posibles vendedores, y fuera el administrador el que decidiera si los acepta o no a través de la aplicación, comunicándose después vía email, por ejemplo.

9.2. CONTROL DE STOCKS.

Actualmente el stock de productos lo controlaría el administrador como venía haciéndolo habitualmente. Esto se podría automatizar de forma que la aplicación controlara el número de productos contenidos en el almacén y en función de esto realizara un informe de petición de productos determinados a los vendedores que el administrador designara en el sistema para suministrarles.

De esta forma y junto con lo citado en el punto 9.1 quedaría completada una gestión de vendedores más formal.

9.3. GESTIÓN DE CONTABILIDAD.

La gestión de contabilidad podría realizarse mediante el control de los pedidos pagados realizados por los clientes y el gasto que supondría los pedidos a vendedores. De esta forma se obtendrían los beneficios como resultado de la diferencia de los ingresos menos los costes. Esto supondría un primer acercamiento a lo que sería una gestión completa de la contabilidad en la aplicación.

9.4. CREACIÓN DE CAMPAÑAS DE OFERTAS.

Debido a que la aplicación desarrollada se trata de una tienda, como tal podría incorporar campañas de ofertas gestionadas por el administrador de forma que apareciese la figura de la “Campaña de Oferta”, algo así como una categoría de productos que agrupa una serie de ellos por su característica de estar rebajados de precio.

9.5. OFERTAS DINÁMICAS PERSONALIZADAS.

Las ofertas podrían ser dinámicas e individualizadas para cada cliente, de forma que el sistema recogiera todos los contenidos de pedidos pagados realizados en un intervalo de tiempo fijo, y determinara los productos “preferidos” del cliente (si es que existen coincidencias de productos en sus pedidos), o bien mediante cookies se determinara los productos más visitados por cada cliente y creara un carrito de la compra virtual con una selección de dichos productos que se le ofrecería al cliente por un mensaje al correo de este a precio rebajado.

9.6. DIFERENTES SECCIONES, DIFERENTES ADMINISTRADORES.

Actualmente la aplicación cuenta con un sólo administrador que gestiona todos los aspectos de toda la aplicación. Una posible ampliación sería crear un usuario capaz de crear “usuarios administradores de sección”, con permisos distintos, los cuales estarían cada uno de ellos al cargo de cada una de las secciones: contabilidad, control de stock, clientes, etc.

De esta forma se filtraría información del sistema que a determinados usuarios administradores no les interesa para realizar su labor y se obtendría una aplicación distribuida por secciones, cada una de ellas con su usuario administrador. Todo esto sería más orientado a una sección de la aplicación que sería una página de Intranet.

9.7. SUBIR FOTOS DE USUARIO AL SERVIDOR.

Se podría habilitar al usuario para que subiera ficheros de tipo imagen al servidor, de forma que en el formulario de inscripción se le diera la opción de añadir una foto, y esta quedara registrada como foto de usuario, que sería mostrada en la ficha del usuario, al igual que las fotos de los productos y sus fichas.

9.8. ENVÍO DE MAILS.

La comunicación entre los usuarios de una aplicación web es fundamental de cara a que el flujo de información no se limite a que un usuario, (como puede ser el administrador) se limite a observar los cambios producidos en el sistema debido a la acción de otros usuarios. Por ello el envío de mails mediante el uso de un servidor SMTP de correo saliente facilitaría esta comunicación.

9.9. SEGUIMIENTO DE PEDIDOS.

El usuario administrador podría realizar el seguimiento de los pedidos realizados, de forma que pudiera controlar cuáles de ellos han sido pagados, cuales no, etc y dar de baja aquellos que considerara oportuno (con direcciones absurdas por ejemplo).

10. CONCLUSIONES.

El presente proyecto representa una síntesis de los conocimientos adquiridos durante el estudio de la documentación y los conocimientos adquiridos en su desarrollo. Todos los aspectos han sido tratados con la mayor rigurosidad posible para obtener lo que sería un producto final software de calidad.

El desarrollo del presente proyecto me ha hecho ampliar las características y el manejo de diversas tecnologías, algunas de ellas sobre todo en el aspecto práctico. Este aprendizaje me ha supuesto una gran ampliación de conocimientos en muchas materias, y me ha demostrado que es necesario ser autodidacta para adquirir conocimientos o al menos ampliarlos.

Por último cabe destacar que comprar en Internet es seguro, siempre y cuando se realice la compra en sitios web que utilicen el protocolo de seguridad SSL, que nos garantiza la encriptación de los datos intercambiados con la tienda virtual y con la pasarela de pagos través de ese canal seguro. Si bien es cierto que hay posibilidades de que se produzca un fraude en este tipo de comercio, también se puede sufrir en el comercio tradicional y por muchas más vías que con el comercio electrónico.

11. ACRÓNIMOS.

CASE: Computer Aided Software Engineering. (Ingeniería de Software Asistida por Ordenador)

CD-ROM: *CD-read-only memory*. (Compact Disc de solo lectura).

DHTML: HTML Dinámico.

DVD: Digital Versatile Disc. (Disco Digital Versátil).

GNU: Acrónimo recursivo de "*GNU No es Unix*".

HTML: HyperText Markup Language. (Lenguaje de Marcación de Hipertexto).

MVC: Patrón de diseño Modelo-Vista-Controlador.

PDF: Portable Document Format. (Formato de Documento Portátil).

PHP: Acrónimo recursivo de "*PHP: Hypertext Preprocessor*", originado inicialmente del nombre PHP Tools.

PHP Tools : Personal Home Page Tools. (Herramienta para Páginas Personales).

POO: Programación Orientada a Objetos.

RAM: Random Acces Memory. (Memoria de Acceso Aleatorio).

SMTP: Simple Mail Transfer Protocol. (Protocolo Simple de Transferencia de Correo electrónico).

SQL: Structured Query Language. (Lenguaje de Consulta Estructurado) .

SSL: Secure Sockets Layer. (Capa de Sockets Segura).

TPV: Terminal Punto de Venta.

UML: Unified Modeling Language. (Lenguaje Unificado de Modelado).

URL: Uniform Resource Locator. (Localizador Uniforme de Recurso).

12. BIBLIOGRAFÍA.

12.1. Comercio electrónico:

URLs:

- **Agencia certificadora española.**
<http://www.ace.es/>
- **Pasarela de pagos PAYPAL.**
<https://paypal.com/>

12.2. Metodología Orientada a Objetos:

- **El lenguaje unificado de modelado**
Booch, Grady
Rumbaugh, James
Jacobson, Ivar
ADDISON WESLEY, 1997.
- **Patrones de Diseño**
Gamma, Erich
ADDISON WESLEY, 2003.

URLs:

- <http://www.rational.com>
- <http://www.phppatterns.com>

12.3. DHTML:

URLs:

- **W3C**
<http://www.w3c.org>

12.4. Sistema Gestor De Base De Datos:

- **Guía de aprendizaje: MySQL**
Ullman, Larry.
PEARSON EDUCACIÓN, 2003.
- **MYSQL**
Gutiérrez Gallardo, Juan Diego.
ANAYA MULTIMEDIA, 2004.

URLs:

- **MySQL**
<http://www.mysql.com>

12.5. Lenguaje Orientado a Objetos:

- **PHP 5**
Mercer, Dave; Kent, Allan; Nowicki, Steven D.
ANAYA MULTIMEDIA, 2005.
- **PHP 5 A TRAVÉS DE EJEMPLOS**
Gutiérrez, A. y Bravo, G.
RA-MA, 2005.

URLs:

- **PHP**
• <http://www.php.net>

12.6. Servidor Apache:

- **La biblia de Servidor Apache**
Kabir, Mohamed J.
ANAYA MULTIMEDIA, 1999.

URLs:

- **Apache.**
• <http://httpd.apache.org/docs/2.2/index.html>

*SECCIÓN 2 : MANUAL
TÉCNICO*

REQUISITOS ESPECÍFICOS.

1. GESTIÓN DE INFORMACIÓN DE CLIENTES.

Cualquier cliente potencial que desee realizar un pedido necesita estar registrado como cliente. Se registrarán por cada cliente los siguientes datos identificativos:

- ✓ Login de usuario.
- ✓ Contraseña.
- ✓ Nombre y apellidos (este último se omitirá para las empresas).
- ✓ DNI (CIF para las empresas).
- ✓ Dirección.
- ✓ Ciudad.
- ✓ Provincia.
- ✓ Código postal.
- ✓ Teléfono.
- ✓ Fax.
- ✓ Fecha de registro como cliente.
- ✓ Número de cuenta corriente.
- ✓ E-mail.

En el momento de registrarse como cliente, el usuario cliente deberá introducir una contraseña personal que servirá junto con su nombre de usuario de cliente para poder identificarse ante el sistema.

Si un cliente quiere tanto modificar, consultar sus datos como darse de baja del sistema, será el cliente mismo el que realice todo el proceso.

1.1. ACCEDIENDO AL SISTEMA COMO CLIENTE.

Para hacer uso de su “cuenta cliente”, el usuario cliente procede a introducir los siguientes datos:

- ✓ Nombre de usuario.
- ✓ Contraseña.

A continuación tendrá ya la posibilidad de realizar las acciones propias de usuario cliente registrado:

- Visitar el catálogo de productos, con la posibilidad de realizar distintos filtrados de información según preferencias.
- Realizar búsquedas de productos por las distintas categorías de productos.
- Gestionar sus datos personales de registro.
- Darse de baja del sistema como usuario cliente.
- Realizar un pedido.

1.2. VISITAR EL CATÁLOGO DE PRODUCTOS.

Esta característica, que se le brinda también al usuario cliente no registrado, permite la visualización del catálogo de los distintos productos ofertados por la tienda. La visualización se puede realizar estableciendo dos tipos de filtrados o preferencias de información:

- Listado por categorías:
Mediante este tipo de preferencia se mostrarán en primera instancia todas las categorías de productos de la tienda y posteriormente los productos que contienen cada una de ellas cuando sean seleccionadas.
- Listado por productos:
Mediante este tipo de preferencia se mostrarán todos los productos que la tienda ofrece sin agrupar según criterio, como podría ser el de categorías de productos citado anteriormente.

1.3. GESTIÓN DE DATOS PERSONALES.

El usuario cliente podrá realizar las labores propias de gestión de sus datos personales como son la consulta y modificación. La condición indispensable para causar baja en el sistema será no tener ningún pedido pendiente en la fecha actual del intento de baja.

1.4. GESTIÓN DE PEDIDOS DE CLIENTES.

Un cliente puede realizar uno o más pedidos de una sola vez. Cuando un cliente realiza un pedido el sistema debe asegurarse de que no tiene pendiente el pago de ningún pedido anterior. Si existen pagos pendientes de efectuar, el cliente debe realizarlos antes de efectuar otro pedido.

Una vez el usuario se haya identificado como tal y haya realizado un pedido, se generará un informe de pedido junto con sus datos personales que constará de lo siguiente:

- ✓ Datos personales del cliente.
- ✓ Referencia de producto.
- ✓ Descripción de producto.
- ✓ Número de unidades.
- ✓ Precio unitario.
- ✓ Importe total del pedido.

Cabe destacar que el método de representación virtual elegido para un pedido será el de un *carrito de la compra virtual*.

2. GESTIÓN DE INFORMACIÓN DE VENDEDORES.

Los vendedores que consten en el sistema deberán ser dados de alta por el usuario administrador, de forma que deba haber un acuerdo previo entre ambas partes para la incorporación del vendedor al sistema. Alguna información referente a usuarios vendedores será de carácter informativo, como son las personas de contacto. El usuario vendedor podrá establecer qué productos desea suministrar a la tienda a través del alta de productos. Para concretar el suministro del producto el usuario administrador entrará en contacto con la/s persona/s de contacto que consten en el sistema para ese usuario vendedor.

Se registrarán por cada vendedor los siguientes datos identificativos:

- ✓ Login de usuario.
- ✓ Contraseña.
- ✓ Nombre representativo del vendedor.
- ✓ CIF.
- ✓ Dirección.
- ✓ Ciudad.
- ✓ Provincia.
- ✓ Código postal.
- ✓ Teléfono.
- ✓ Fax.
- ✓ E-mail.

Debido a que será el usuario administrador el que dé de alta a los vendedores, será responsabilidad de este el rellenar los datos señalados anteriormente, incluidos el nombre de usuario y contraseña. Esto implica que en primera instancia estos dos datos serán establecidos por el usuario administrador, y remitidos al vendedor para que haciendo uso de su “cuenta vendedor” sea este el que establezca los valores preferidos, a excepción del nombre de usuario.

Si un vendedor quiere tanto modificar, consultar sus datos como darse de baja del sistema, será este el que realice todo el proceso.

2.1. ACCEDIENDO AL SISTEMA COMO VENDEDOR.

Para hacer uso de su “cuenta vendedor”, el usuario vendedor procede a introducir los siguientes datos:

- ✓ Nombre de usuario.
- ✓ Contraseña.

A continuación tendrá ya la posibilidad de realizar las acciones propias de usuario vendedor:

- Visitar el catálogo de productos, con la posibilidad de realizar distintos filtrados de información según preferencias.
- Realizar búsquedas de productos por las distintas categorías de productos.
- Gestionar sus datos personales y su persona de contacto.
- Gestionar las productos que estaría en disposición de proveer.

2.2. GESTIÓN DE DATOS PERSONALES.

El usuario vendedor podrá realizar las labores propias de gestión de sus datos personales como son la consulta, modificación y baja en el sistema.

2.3. GESTIÓN DE LA PERSONA DE CONTACTO .

El usuario vendedor podrá realizar la gestión de los datos personales de personas de contacto que el sistema almacenará con carácter informativo, para que en el caso de que el usuario administrador quiera que se le suministre un determinado producto, se ponga en contacto con esa persona a través de los datos almacenados de ella.

Los datos almacenados por cada una de estas personas serán:

- ✓ Nombre de contacto.
- ✓ Apellidos de contacto.
- ✓ Email de contacto.
- ✓ Teléfono de contacto.

2.4. GESTIÓN DE LOS PRODUCTOS.

El usuario vendedor podrá realizar la gestión de los datos referentes a los productos que estaría en disposición de proveer a la tienda.

Los datos almacenados por cada una de los productos serán:

- ✓ Identificador de producto.
- ✓ Nombre.
- ✓ Descripción.
- ✓ Imagen.
- ✓ Estado de suspensión.
- ✓ Unidades disponibles en el almacén.
- ✓ Categoría a la que pertenecen.
- ✓ Fecha de suministro.

3. GESTIÓN DE INFORMACIÓN DEL ADMINISTRADOR.

La manipulación de la aplicación con privilegios de usuario administrador correrá a cargo por parte de un único usuario de tipo administrador. La aplicación por defecto definirá un sólo usuario de este tipo de forma que no se ofrece la posibilidad de dar de alta nuevos usuarios administradores.

El sistema almacenará los datos personales del usuario administrador que podrían ser utilizados en el envío de correos electrónicos a clientes y vendedores en un futuro, así como la posible generación de facturas.

Los datos almacenados del usuario administrador serán comunes a los descritos para el cliente, a excepción del código de cuenta corriente, la fecha de registro y el tipo de cliente.

3.1. ACCEDIENDO AL SISTEMA COMO ADMINISTRADOR.

Para hacer uso de su “cuenta administrador”, el usuario administrador procede a introducir los siguientes datos:

- ✓ Login de usuario.
- ✓ Contraseña.

A continuación tendrá ya la posibilidad de realizar las acciones propias de usuario administrador:

- Visitar el catálogo de productos, con la posibilidad de realizar distintos filtrados de información según preferencias.
- Realizar búsquedas de productos por las distintas categorías de productos.
- Gestionar sus datos personales.
- Gestionar los productos y las categorías por las que se agrupan.

3.2. GESTIÓN DE PRODUCTOS Y CATEGORÍAS DE PRODUCTOS.

El usuario administrador realizará en este punto las labores de gestión de datos tanto para productos como para categorías de productos como son el alta, consulta, modificación y baja de cada uno de ellos.

ESPECIFICACIÓN DE LOS CASOS DE USO

1. Caso de uso “Consultar catálogo”.

●**Objetivo:**

Mostrar información de productos disponibles según las preferencias y filtrados de información establecidas en el criterio de visualización. Las preferencias establecerán la posibilidad de realizar la visualización por categorías de productos y posteriormente de los productos de estas o bien la visualización de productos directamente de acuerdo a los filtrados de información establecidos.

●**Actor:**

Usuario cliente, usuario vendedor, usuario administrador.

●**Flujo de eventos principal:**

1. El usuario solicita la visualización de los datos de productos.
2. El sistema solicita las preferencias de búsqueda y los posibles filtros de información.
3. El usuario establece unas determinadas preferencias y filtrados de información.
4. El sistema recoge la petición y muestra una relación de productos, o categorías en su caso, que cumplen con las condiciones establecidas.

●**Flujo de eventos alternativo:**

- 1.4 <Cancelación de proceso> : El usuario puede cancelar el proceso en cualquier momento, reiniciando de esta manera el caso de uso.
- 4.1. El usuario solicita al sistema que visualice los detalles de un determinado producto.
- 4.2. El sistema recoge la petición, mostrando los detalles de dicho producto.

●**Flujo de eventos excepcional:**

- <Fallo de conexión> *común para ambos flujos*: El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.
- 4.<No se encontraron productos> *flujo principal*: El sistema muestra un mensaje indicando que no se han encontrado productos que cumplan las condiciones especificadas, y le permite al usuario realizar una nueva búsqueda.
- 4.2.<No se encontraron detalles de producto> *flujo alternativo*: El sistema muestra un mensaje indicando que no existen detalles para el producto especificado.

●**Escenarios:**

- 1.1 Listar productos según condiciones.

2. Caso de uso “Gestionar datos cliente” incluye (Validar usuario).

•Objetivo:

Permite al usuario realizar las operaciones de gestión de datos personales modificación y consulta.

•Actor:

Usuario cliente registrado.

•Pasos:

1. El usuario podrá crear una consulta referente a sus datos personales.
2. El usuario podrá realizar una modificación de sus datos personales.

Este es un caso de uso abstracto que se introduce para englobar las operaciones de gestión de datos de cliente registrado que se explotan en subcasos de uso que se muestran en la siguiente figura:

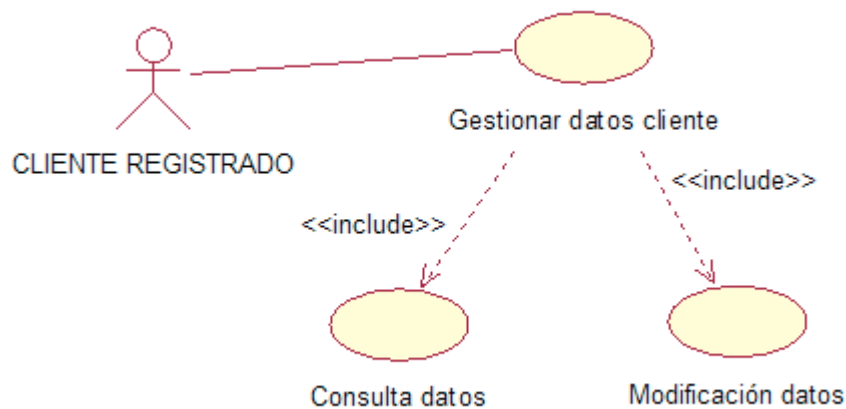


Figura 1. Diagrama de Caso de uso “Gestionar datos cliente”

2.1. Subcaso de uso “Consulta datos”.

•Objetivo:

Permite al usuario cliente visualizar sus datos personales almacenados en el sistema.

•Actor:

Usuario cliente registrado.

•Flujo de eventos principal:

1. El usuario solicita al sistema realizar la consulta de sus datos personales.
2. El sistema muestra los datos actuales del usuario.

•**Flujo de eventos alternativo:**

1.2 <Cancelación de proceso> : El usuario cliente puede cancelar el proceso en cualquier momento, reiniciando de esta manera el caso de uso.

•**Flujo de eventos excepcional:**

<Fallo de conexión> *común para ambos flujos*: El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

•**Escenarios:**

2.1 Visualización de datos personales.

2.2. Subcaso de uso “Modificación datos”.

•**Objetivo:**

Permite al usuario modificar sus datos personales almacenados en el sistema.

•**Actor:**

Usuario cliente registrado.

•**Flujo de eventos principal:**

1. El usuario solicita al sistema modificar sus datos personales.
2. El sistema muestra los datos actuales del usuario.
3. El usuario introduce los nuevos datos personales
4. El sistema los recoge, previa validación de los mismos.
5. El sistema modifica los datos correspondientes, introducidos por el usuario.

•**Flujo de eventos alternativo:**

1.4 <Cancelación de proceso> : El usuario puede cancelar el proceso en cualquier momento, reiniciando de esta manera el caso de uso.

•**Flujo de eventos excepcional:**

<Fallo de conexión> *común para ambos flujos*: El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

4. <Dato obligatorio en blanco>: El sistema muestra un mensaje de error y se le da al usuario la posibilidad de introducir el dato que faltaba.

3. <Formato de dato no válido>. El sistema muestra un mensaje de error indicando el dato cuyo formato no es válido, ofreciendo al usuario la posibilidad de modificarlo.

•**Escenarios:**

2.2 Modificación de datos personales.

3. Caso de uso “Alta cliente”.

●**Objetivo:**

Registrar como usuario cliente a un usuario no registrado y por lo tanto darle de alta en el sistema.

●**Actor:**

Usuario cliente no registrado.

●**Flujo de eventos principal:**

1. El usuario no registrado hace una petición de registro al sistema.
2. El sistema solicita al usuario que introduzca los datos.
3. El usuario introduce los datos requeridos.
4. El sistema recoge los datos, previa validación y da de alta al usuario en el sistema.

●**Flujo de eventos alternativo:**

1.4 <Cancelación de proceso> *flujo principal*: El usuario cliente puede cancelar el proceso en cualquier momento, reiniciando de esta manera el caso de uso.

●**Flujo de eventos excepcional:**

1.4.<Fallo de conexión> : El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

4. <Dato obligatorio en blanco>: El sistema muestra un mensaje de error y retorna al usuario a la introducción de los datos para que rellene el dato obligatorio.

4. <Formato no válido>: El sistema muestra un mensaje de error indicando el dato cuyo formato no es válido, ofreciendo al usuario la posibilidad de modificarlo.

4. <Login ya existe>: El sistema muestra un mensaje de error indicando que el login ya existe y da la posibilidad al usuario de introducir un nuevo login.

4. <Password incorrecto>: El sistema muestra un mensaje de error indicando que el password introducido es inválido por contener caracteres especiales, y da al usuario posibilidad de volver a introducir el password.

●**Escenarios:**

3.1. Alta cliente.

4. Caso de uso “Baja cliente”.

●**Objetivo:**

Dar de baja a un usuario cliente del sistema.

●**Actor:**

Usuario cliente registrado.

●**Flujo de eventos principal:**

1. El usuario registrado hace una petición para darse de baja del sistema como cliente.
2. El sistema da de baja al cliente.

●**Flujo de eventos alternativo:**

1..2. <Cancelación de proceso> *flujo principal*: El usuario cliente puede cancelar el proceso en cualquier momento, reiniciando de esta manera el caso de uso.

●**Flujo de eventos excepcional:**

1..2. <Fallo de conexión> : El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

●**Escenarios:**

4.1 Baja cliente.

5. Caso de uso “Gestión carrito de compra”.

●**Objetivo:**

Permitir al usuario cliente registrado añadir productos a su carrito de la compra “virtual”, modificar las unidades que desea de un producto, sacar productos del carrito o vaciar el carrito de la compra. Además, se ofrece la posibilidad de ver los productos que tiene actualmente en su carrito de la compra.

●**Actor:**

Usuario cliente registrado.

●**Flujo de eventos principal:**

El usuario añade los productos al carrito de la compra indicando el número de unidades que desea. El sistema añade la relación de productos al carrito de la compra. El usuario solicita al sistema realizar el pedido correspondiente al contenido de su carrito.

●**Flujo de eventos alternativos:**

- 1.1 El usuario solicita al sistema ver el contenido de su carrito de la compra.
- 1.2 El sistema muestra al usuario el contenido de su carrito.
- 1.3 El usuario solicita al sistema vaciar por completo su carrito de la compra.
- 1.4 El sistema vacía el carrito de la compra.
- 1.5 El usuario solicita modificar el contenido de su carrito de la compra.
- 1.6 El sistema muestra el contenido actual del carrito de la compra.
- 1.7 El usuario lleva a cabo las modificaciones oportunas.
- 1.8 El sistema actualiza el carrito de la compra.

1.3 <Cancelación de proceso>: El usuario cliente puede cancelar el proceso de búsqueda en cualquier momento, reiniciando de esta manera el caso de uso.

●**Flujo de eventos excepcional:**

<Fallo de conexión> *común para ambos flujos*: El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

2, 3.8. <Cantidad negativa o cero>. En caso de que el usuario introduzca como cantidad de producto a añadir al carrito un número.

●**Escenarios:**

- 5.1. Añadir un producto al carrito.
- 5.2. Visualizar el contenido del carrito.
- 5.3. Modificar las unidades de un producto del carrito.
- 5.4. Vaciar el carrito.
- 5.5. Borrar un producto del carrito.

6. Caso de uso "Comprar". include (Validar usuario), include (Gestion carrito de compra), include (Pagar).

●**Objetivo:**

Permitir al usuario cliente registrado realizar un pedido.

●**Actor:**

Usuario cliente registrado.

●**Flujo de eventos principal:**

1. El usuario solicita al sistema realizar la compra de los productos existentes en su carrito de la compra.
2. El sistema aplica los descuentos correspondientes y muestra el pedido actual con los datos del usuario y del carrito de la compra.
3. El sistema modifica si es necesario la categoría de cliente.
4. El sistema genera un informe que será mandado al cliente con los datos de su pedido.

●**Flujo de eventos alternativos:**

- 1.4 <Cancelación de proceso>: El usuario cliente puede cancelar el proceso de compra en cualquier momento.
- 2.1. El usuario decide modificar en este punto la composición de su compra.

●**Flujo de eventos excepcional:**

- 1.4. <Fallo de conexión>: El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.
1. <Carrito de la compra vacío>: El sistema muestra un mensaje informando al usuario de que su carrito está vacío y le retorna a la opción de compra de productos.

●**Escenarios:**

- 6.1 Compra de pedido.

7. Caso de uso “Pagar”.

•Objetivo:

Permitir al usuario registrado, que va a llevar a cabo un pedido, elegir una forma de pago concreta.

•Actor:

Usuario cliente registrado.

•Flujo de eventos principal:

1. El usuario solicita al sistema indicar el método de pago.
2. El sistema solicita al usuario que indique el método de pago.
3. El usuario establece el método de pago.
4. El sistema guarda la forma de pago.

•Flujo de eventos alternativos:

1..2 <Cancelación de proceso>: El usuario cliente puede cancelar el proceso de compra en cualquier momento.

•Flujo de eventos excepcional:

1..2. <Fallo de conexión>: El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

1. <Carrito de la compra vacío>: El sistema muestra un mensaje informando al usuario de que su carrito está vacío y le retorna a la opción de compra de productos.

•Escenarios:

7.1. Establecer forma de pago.

8. Caso de uso “Validar usuario”.

•Objetivo:

Comprobar que el usuario que intenta acceder al sistema está registrado como tal en este, y por tanto tiene acceso a las operaciones propias correspondientes a su categoría.

•Actor:

Usuario cliente, usuario vendedor, usuario administrador.

•Flujo de eventos principal:

1. Un usuario (registrado o no) realiza una petición de acceso al sistema.
2. El sistema requiere la autenticación de este.
3. El usuario se identifica ante el sistema.
4. El sistema comprueba los datos, permitiendo su acceso al sistema.

●**Flujo de eventos alternativos:**

1.4 <Cancelación de proceso> : El usuario cliente puede cancelar el proceso en cualquier momento.

●**Flujo de eventos excepcional:**

<Fallo de conexión> *común para ambos flujos*: El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

4. <No existe usuario>: El sistema muestra un mensaje informando de que el login introducido no se corresponde con ningún usuario y se le da al usuario la posibilidad de volver a intentar la identificación, para el acceso al sistema.

4. <Contraseña incorrecta>: El sistema muestra un mensaje de error que indica que el password introducido no es el correcto para ese login, y se le da al usuario una nueva oportunidad para introducir su password.

●**Escenarios:**

8.1. Validación de usuario.

9. Caso de uso “Gestionar datos vendedor”: incluye (Validar usuario).

●**Objetivo:**

El objetivo final es permitir al usuario vendedor realizar las operaciones propias de gestión de datos personales: modificación, visualización y baja en el sistema.

●**Actor:**

Usuario vendedor.

●**Pasos:**

1. El usuario vendedor podrá crear una consulta referente a sus datos personales.
2. El usuario vendedor podrá realizar una modificación de sus datos personales.
3. El usuario vendedor podrá darse de baja como vendedor en el sistema.

Este es un caso de uso abstracto que se introduce para englobar las operaciones de gestión de los datos de vendedor, que se explotan en subcasos de uso que se muestran en la siguiente figura:

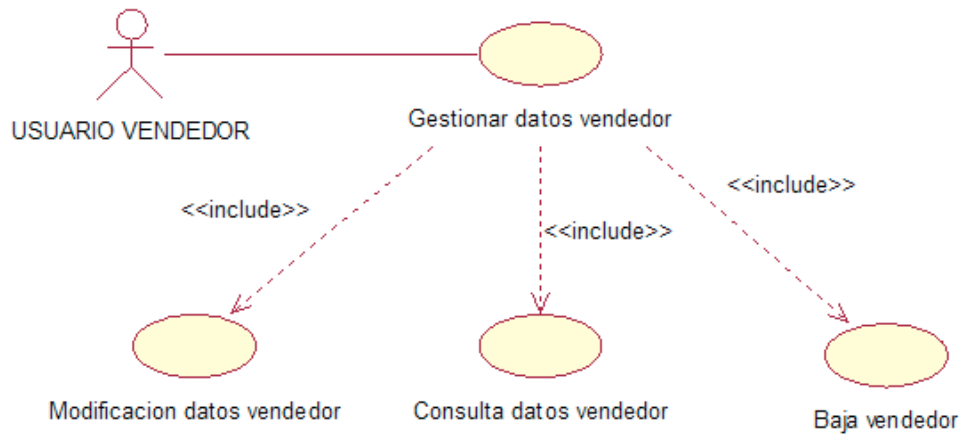


Figura 2. Diagrama de Caso de uso “Gestionar datos vendedor”.

9.1. Subcaso de uso “Consulta datos vendedor”.

●**Objetivo:**

El objetivo es permitir al usuario visualizar los datos registrados en el sistema.

●**Actor:**

Usuario vendedor.

●**Flujo de eventos principal:**

1. El usuario solicita al sistema realizar una consulta sobre sus datos.
2. El sistema solicita las preferencias de visualización y los posibles filtros de información.
3. El usuario establece unas determinadas preferencias y filtrados de información.
4. El sistema recoge la petición y muestra la información requerida.

●**Flujo de eventos alternativos:**

1.4 <Cancelación de proceso> : El usuario cancela el proceso de compra en cualquier momento.

●**Flujo de eventos excepcional:**

<Fallo de conexión> *común para ambos flujos*: El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

3. <Rango especificado no válido>: El sistema muestra un mensaje indicando que el rango especificado de búsqueda no es válido, lo que se puede deber a que la fecha de registro como vendedor no se encuentra dentro del rango.

4. <No hay resultados en la búsqueda>: El sistema muestra un mensaje indicando que la búsqueda en el historial de vendedores no produjo ningún resultado de acuerdo con el rango establecido.

●**Escenarios:**

9.1. Visualización datos vendedor.

9.2. Subcaso de uso "Modificación datos vendedor".

●Objetivo:

El objetivo es permitir al usuario vendedor modificar sus datos registrados en el sistema.

●Actor:

Usuario vendedor.

●Flujo de eventos principal:

1. El vendedor solicita al sistema realizar una modificación sobre sus datos.
2. El sistema muestra los datos actuales del vendedor y solicita los nuevos.
3. El vendedor introduce los nuevos datos personales
4. El sistema los recoge, previa validación de los mismos.
5. El sistema modifica los datos correspondientes, introducidos por el usuario.

●Flujo de eventos alternativos:

1.5 <Cancelación de proceso>: El usuario vendedor cancela el proceso de compra en cualquier momento.

●Flujo de eventos excepcional:

1.5. <Fallo de conexión> : El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

4. <Dato obligatorio en blanco>: El sistema muestra un mensaje de error y se le da al usuario la posibilidad de introducir el dato que faltaba.

5. <Formato de dato no válido>: El sistema muestra un mensaje de error indicando el dato cuyo formato no es válido, ofreciendo al usuario la posibilidad de modificarlo.

●Escenarios:

9.2. Modificación de datos vendedor.

9.3. Subcaso de uso "Baja vendedor".

●Objetivo:

El objetivo es permitir al usuario vendedor darse de baja dentro del sistema.

●Actor:

Usuario vendedor.

●Flujo de eventos principal:

1. El usuario solicita al sistema realizar una baja en el sistema.
2. El sistema realiza la baja del vendedor.

●Flujo de eventos alternativos:

1..2. <Cancelación de proceso> *flujo principal*: El usuario puede cancelar el proceso en cualquier momento, reiniciando de esta manera el caso de uso.

●**Flujo de eventos excepcional:**

1..2. <Fallo de conexión>: El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

●**Escenarios:**

9.3. Causar baja en el sistema.

10. Caso de uso “Gestionar persona contacto”: include(Validar usuario).

●**Objetivo:**

El objetivo de este caso de uso es ofrecer al usuario vendedor la posibilidad de gestionar los datos relativos a la persona de contacto que constará en el sistema como contacto con el vendedor. Se podrán realizar las operaciones de alta, modificación, visualización y baja en el sistema.

●**Actor:**

Usuario vendedor.

●**Pasos:**

1. El usuario vendedor podrá dar de alta a un contacto, la cual figurará en el sistema como persona de contacto.
2. El usuario vendedor podrá realizar una consulta acerca de la persona de contacto.
3. El usuario vendedor podrá realizar una modificación de los datos de la persona de contacto.
4. El usuario vendedor podrá dar de baja en el sistema a la persona de contacto.

Este es un caso de uso abstracto que se introduce para englobar las operaciones de gestión de la persona de contacto que se explotan en subcasos de uso que se muestran en la siguiente figura:

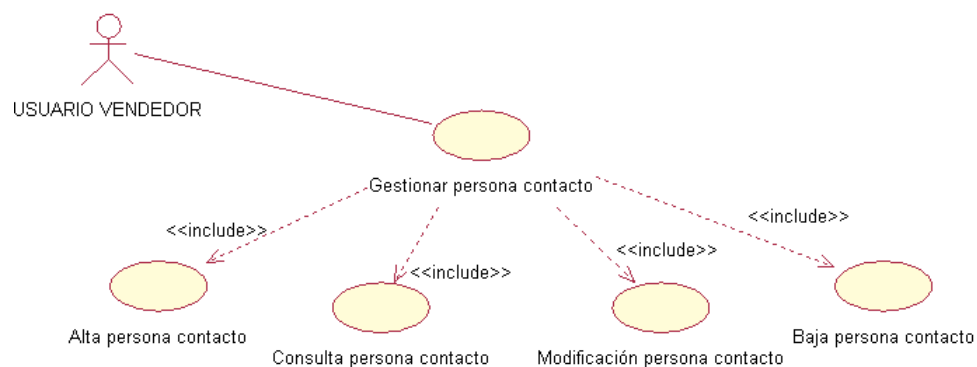


Figura 3. Diagrama de Caso de uso “Gestionar persona contacto”.

10.1. Subcaso de uso “Alta persona contacto”

●**Objetivo:**

El objetivo es permitir al usuario vendedor dar de alta a una persona de contacto dentro del sistema.

●**Actor:**

Usuario vendedor.

●**Flujo de eventos principal:**

1. El vendedor solicita al sistema dar de alta a una persona de contacto en el sistema.
2. El sistema solicita los datos de dicha persona de contacto.
3. El vendedor introduce al sistema los datos de dicha persona.
4. El sistema los recoge previa validación de los mismos.
5. El sistema da de alta a la persona de contacto.

●**Flujo de eventos alternativos:**

1.4 <Cancelación de proceso> *flujo principal*: El usuario vendedor puede cancelar el proceso en cualquier momento, reiniciando de esta manera el caso de uso.

●**Flujo de eventos excepcional:**

1.4 <Fallo de conexión> : El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

4.<Dato obligatorio en blanco> : El sistema muestra un mensaje de error y se le da al usuario la posibilidad de introducir el dato que faltaba.

4. <Formato de dato no válido>. El sistema muestra un mensaje de error indicando el dato cuyo formato no es válido, ofreciendo al usuario la posibilidad de modificarlo.

●**Escenarios:**

10.1. Alta Persona de contacto.

10.2. Subcaso de uso “Consulta persona contacto”

●**Objetivo:**

El objetivo es permitir al usuario vendedor consultar los datos de la persona de contacto.

●**Actor:**

Usuario vendedor.

●**Flujo de eventos principal:**

1. El usuario vendedor solicita al sistema la visualización de los datos de la persona de contacto.
2. El sistema muestra los resultados, dando la posibilidad al usuario de mostrar el detalle de los datos de la persona de contacto.

●**Flujo de eventos alternativos:**

1..2 <Cancelación de proceso> *flujo principal*: El usuario vendedor puede cancelar el proceso en cualquier momento, reiniciando de esta manera el caso de uso.

●**Flujo de eventos excepcional:**

1..2 <Fallo de conexión> : El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

●**Escenarios:**

10.2. Visualización de datos persona contacto.

10.3. Subcaso de uso “Modificación persona contacto”.

●**Objetivo:**

El objetivo es permitir al usuario vendedor modificar los datos de la persona de contacto.

●**Actor:**

Usuario vendedor

●**Flujo de eventos principal:**

1. El vendedor hace una petición para modificar los datos relativos a la persona de contacto.
2. El sistema muestra los datos actuales de la persona de contacto y solicita los nuevos.
3. El vendedor introduce los nuevos datos.
4. El sistema los recoge, previa validación de los mismos.
5. El sistema modifica los datos correspondientes, introducidos por el usuario.

●**Flujo de eventos alternativos:**

1..4 <Cancelación de proceso> *flujo principal*: El usuario vendedor puede cancelar el proceso en cualquier momento, reiniciando de esta manera el caso de uso.

●**Flujo de eventos excepcional:**

1..4 <Fallo de conexión> : El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

●**Escenarios:**

11.3. Modificación de datos persona contacto.

10.4. Subcaso de uso “Baja persona contacto”.

●**Objetivo:**

El objetivo es permitir al usuario vendedor poder dar de baja a la persona de contacto en el sistema.

●**Actor:**

Usuario vendedor.

●**Flujo de eventos principal:**

1. El vendedor hace una petición para dar de baja a la persona de contacto.
2. El sistema da de baja a la persona de contacto.

●**Flujo de eventos alternativos:**

1.2 <Cancelación de proceso> *flujo principal*: El usuario vendedor puede cancelar el proceso en cualquier momento, reiniciando de esta manera el caso de uso.

●**Flujo de eventos excepcional:**

1.2 <Fallo de conexión> : El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

●**Escenarios:**

11.4. Causar baja en el sistema persona contacto.

11. Caso de uso “Gestionar productos Vendedor”: **include (Validar usuario)** .

●**Objetivo:**

Permitir al usuario vendedor realizar las operaciones de alta, consulta, modificación y baja de sus productos del sistema.

●**Actor:**

Usuario vendedor.

●**Pasos:**

1. El usuario podrá dar de alta en el sistema productos.
2. El usuario podrá realizar la consulta de los datos de productos.
3. El usuario podrá realizar la modificación de los datos de productos.
4. El usuario podrá dar de baja en el sistema a los productos.

Este es un caso de uso abstracto que se introduce para englobar las operaciones de gestión de productos que se explotan en subcasos de uso que se muestran en la siguiente figura:

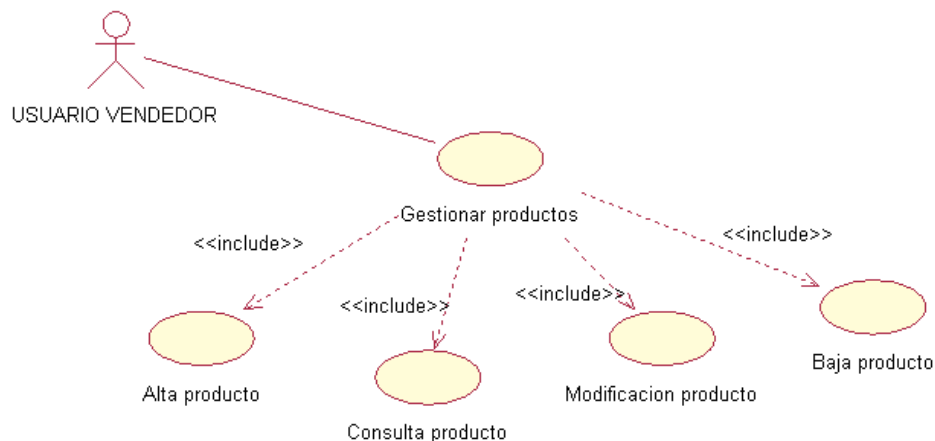


Figura 4. Diagrama de Caso de uso “Gestionar productos vendedor”.

11.1. Subcaso de uso “Alta producto” .

●**Objetivo:**

Permitir al usuario vendedor dar de alta productos.

●**Actor:**

Usuario vendedor.

●**Flujo de eventos principal:**

1. El usuario solicita al sistema realizar el alta de un nuevo producto.
2. El sistema pide los datos referentes al producto.
3. El usuario introduce los datos requeridos.
4. El sistema recoge los datos y los valida..
5. El sistema da de alta el producto en el sistema.

●**Flujo de eventos alternativo:**

1..4 <Cancelación de proceso> *flujo principal*: El usuario cliente puede cancelar el proceso en cualquier momento, reiniciando de esta manera el caso de uso.

●**Flujo de eventos excepcional:**

<Fallo de conexión> *flujo principal*: El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

4. <Dato obligatorio en blanco>: El sistema muestra un mensaje de error indicando la existencia del dato en blanco, y retorna al usuario a la introducción de datos del producto.

4. <Formato de dato no válido>: El sistema muestra un mensaje de error indicando el dato cuyo formato no es válido, ofreciendo al usuario la posibilidad de modificarlo.

4. <Producto ya existe >: El sistema muestra un mensaje de error indicando que el producto que se quiere dar de alta ya existe en el sistema.

●**Escenarios:**

11.1. Alta productos vendedor.

11.2. Subcaso de uso “Consulta producto” .

●**Objetivo:**

Permitir al usuario vendedor consultar los datos de los productos.

●**Actor:**

Usuario vendedor.

●**Flujo de eventos principal:**

1. El usuario vendedor solicita al sistema la visualización de datos de productos.
2. El sistema muestra los resultados, dando la posibilidad al usuario de mostrar el detalle de los datos para un determinado producto.

●**Flujo de eventos alternativo:**

1.2 <Cancelación de proceso> : El usuario cliente puede cancelar el proceso en cualquier momento, reiniciando de esta manera el caso de uso.

●**Flujo de eventos excepcional:**

<Fallo de conexión> *flujo principal*: El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.
2.<No hay resultados en la búsqueda> : El sistema muestra un mensaje al usuario administrador de que la búsqueda no produjo ningún resultado.

●**Escenarios:**

11.2. Visualización de datos de producto.

11.3. Subcaso de uso “Modificación producto” .

●**Objetivo:**

Permitir al usuario vendedor modificar los datos de los productos.

●**Actor:**

Usuario vendedor.

●**Flujo de eventos principal:**

1. El usuario solicita al sistema realizar la modificación de los datos de un producto.
2. El sistema muestra los datos actuales del producto y solicita los nuevos.
3. El usuario introduce los nuevos datos.
4. El sistema los recoge, previa validación de los mismos.
5. El sistema modifica los datos correspondientes, introducidos por el usuario.

●**Flujo de eventos alternativo:**

1..3 <Cancelación de proceso> : El usuario cliente puede cancelar el proceso en cualquier momento, reiniciando de esta manera el caso de uso.

●**Flujo de eventos excepcional:**

<Fallo de conexión> *flujo principal*: El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

2.<No hay resultados en la búsqueda> : El sistema muestra un mensaje al usuario administrador de que la búsqueda no produjo ningún resultado.

3. <Dato obligatorio en blanco>. El sistema muestra un mensaje de error indicando la existencia del dato en blanco, y retorna al usuario a la introducción de datos del producto.

3. <Formato de dato no válido> : El sistema muestra un mensaje de error indicando el dato cuyo formato no es válido, ofreciendo al usuario la posibilidad de modificarlo.

●**Escenarios:**

11.3. Modificación productos.

11.4 Subcaso de uso “Baja producto” .

●**Objetivo:**

Permitir al usuario dar de baja en el sistema a los productos.

●**Actor:**

Usuario vendedor.

●**Flujo de eventos principal:**

1. El usuario realiza la petición de baja al sistema para dar de baja un producto.
2. El sistema da de baja el producto.

●**Flujo de eventos alternativo:**

1..2 <Cancelación de proceso> : El usuario cliente puede cancelar el proceso en cualquier momento, reiniciando de esta manera el caso de uso.

●**Flujo de eventos excepcional:**

<Fallo de conexión> *flujo principal*: El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

●**Escenarios:**

14.4. Baja productos.

12. Caso de uso “Gestionar productos”: incluye (Validar usuario) .

•Objetivo:

Permitir al usuario administrador realizar las operaciones de alta, consulta, modificación y baja de productos del sistema.

•Actor:

Usuario administrador.

•Pasos:

1. El usuario podrá dar de alta en el sistema productos.
2. El usuario podrá realizar la consulta de los datos de productos.
3. El usuario podrá realizar la modificación de los datos de productos.
4. El usuario podrá dar de baja en el sistema a los productos.

Este es un caso de uso abstracto que se introduce para englobar las operaciones de gestión de productos que se explotan en subcasos de uso que se muestran en la siguiente figura:

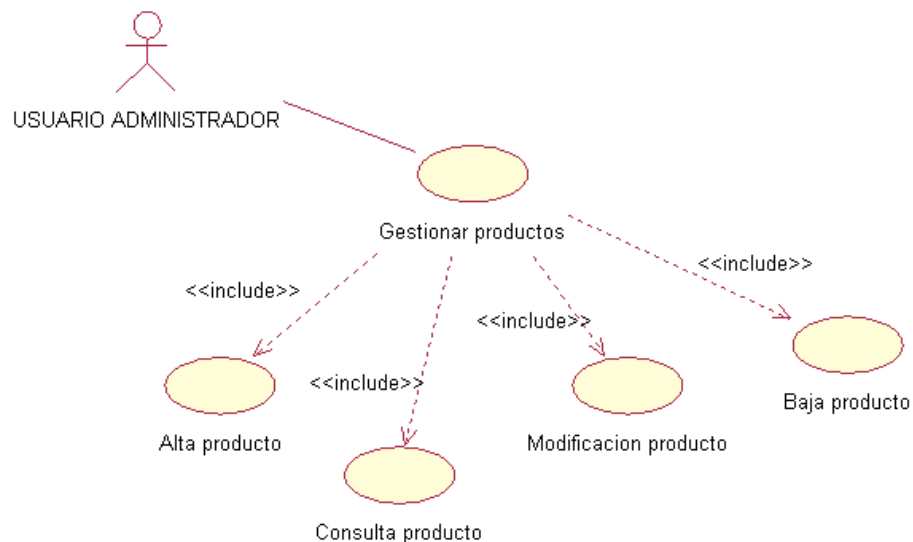


Figura 5. Diagrama de Caso de uso “Gestionar productos”.

12.1. Subcaso de uso “Alta producto” .

•Objetivo:

Permitir al usuario administrador dar de alta productos.

•Actor:

Usuario administrador.

•Flujo de eventos principal:

1. El usuario solicita al sistema realizar el alta de un nuevo producto.
2. El sistema pide los datos referentes al producto.
3. El usuario introduce los datos requeridos.
4. El sistema recoge los datos y los valida..
5. El sistema da de alta el producto en el sistema.

●**Flujo de eventos alternativo:**

1.4 <Cancelación de proceso> *flujo principal*: El usuario cliente puede cancelar el proceso en cualquier momento, reiniciando de esta manera el caso de uso.

●**Flujo de eventos excepcional:**

<Fallo de conexión> *flujo principal*: El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

4. <Dato obligatorio en blanco>: El sistema muestra un mensaje de error indicando la existencia del dato en blanco, y retorna al usuario a la introducción de datos del producto.

4. <Formato de dato no válido>: El sistema muestra un mensaje de error indicando el dato cuyo formato no es válido, ofreciendo al usuario la posibilidad de modificarlo.

4. <Producto ya existe >: El sistema muestra un mensaje de error indicando que el producto que se quiere dar de alta ya existe en el sistema.

●**Escenarios:**

12.1. Alta productos.

12.2. Subcaso de uso “Consulta producto” .

●**Objetivo:**

Permitir al usuario consultar los datos de los productos.

●**Actor:**

Usuario administrador.

●**Flujo de eventos principal:**

1. El usuario administrador solicita al sistema la visualización de datos de productos.
2. El sistema muestra los resultados, dando la posibilidad al usuario de mostrar el detalle de los datos para un determinado producto.

●**Flujo de eventos alternativo:**

1..2 <Cancelación de proceso> : El usuario cliente puede cancelar el proceso en cualquier momento, reiniciando de esta manera el caso de uso.

●**Flujo de eventos excepcional:**

<Fallo de conexión> *flujo principal*: El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

2.<No hay resultados en la búsqueda> : El sistema muestra un mensaje al usuario administrador de que la búsqueda no produjo ningún resultado.

●**Escenarios:**

12.2. Visualización de datos de producto.

12.3. Subcaso de uso “Modificación producto” .

●**Objetivo:**

Permitir al usuario modificar los datos de los productos.

●**Actor:**

Usuario administrador.

●**Flujo de eventos principal:**

1. El usuario solicita al sistema realizar la modificación de los datos de un producto.
2. El sistema muestra los datos actuales del producto y solicita los nuevos.
3. El usuario introduce los nuevos datos.
4. El sistema los recoge, previa validación de los mismos.
5. El sistema modifica los datos correspondientes, introducidos por el usuario.

●**Flujo de eventos alternativo:**

1.3 <Cancelación de proceso> : El usuario cliente puede cancelar el proceso en cualquier momento, reiniciando de esta manera el caso de uso.

●**Flujo de eventos excepcional:**

<Fallo de conexión> *flujo principal*: El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

2.<No hay resultados en la búsqueda> : El sistema muestra un mensaje al usuario administrador de que la búsqueda no produjo ningún resultado.

3. <Dato obligatorio en blanco>. El sistema muestra un mensaje de error indicando la existencia del dato en blanco, y retorna al usuario a la introducción de datos del producto.

3. <Formato de dato no válido> : El sistema muestra un mensaje de error indicando el dato cuyo formato no es válido, ofreciendo al usuario la posibilidad de modificarlo.

●**Escenarios:**

12.3. Modificación productos.

12.4 Subcaso de uso “Baja producto” .

●**Objetivo:**

Permitir al usuario dar de baja en el sistema a los productos.

●**Actor:**

Usuario administrador.

●**Flujo de eventos principal:**

1. El usuario realiza la petición de baja al sistema para dar de baja un producto.
2. El sistema da de baja el producto.

●**Flujo de eventos alternativo:**

1..2 <Cancelación de proceso> : El usuario cliente puede cancelar el proceso en cualquier momento, reiniciando de esta manera el caso de uso.

●**Flujo de eventos excepcional:**

<Fallo de conexión> *flujo principal*: El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

●**Escenarios:**

12.4. Baja productos.

12. Caso de uso “Gestionar categorías productos”: include (Validar usuario) .

●**Objetivo:**

Realizar las operaciones propias de gestión con las categorías de productos de ofertas: alta, visualización, modificación y baja.

●**Actor:**

Usuario administrador.

●**Pasos:**

1. El usuario podrá dar de alta a una categoría.
2. El usuario podrá consultar los datos relativos a categorías.
3. El usuario podrá modificar los datos de las categorías.
4. El usuario podrá dar de baja en el sistema a categorías.

Este es un caso de uso abstracto que se introduce para englobar las operaciones de gestión de categorías de productos que se explotan en subcasos de uso que se muestran en la siguiente figura :

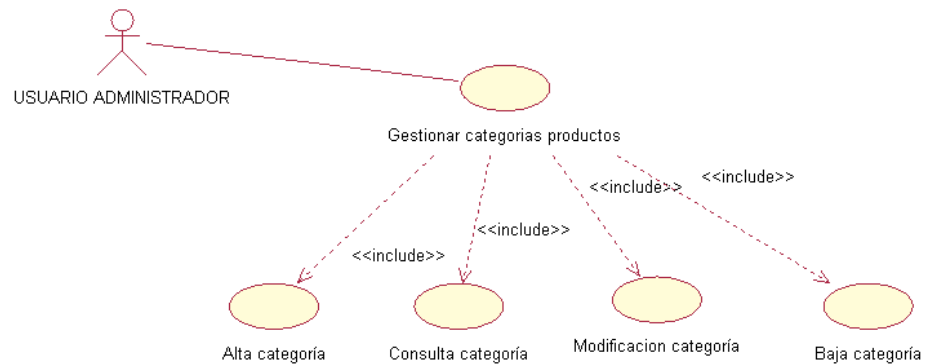


Figura 6. Diagrama de Caso de uso “Gestionar categorías productos”.

13.1. Subcaso de uso “Alta categoría” .

•Objetivo:

Permitir al usuario administrador dar de alta categorías de productos.

•Actor:

Usuario administrador.

•Flujo de eventos principal:

1. El usuario solicita al sistema realizar el alta de una nueva categoría.
2. El sistema pide los datos referentes a la categoría.
3. El usuario introduce los datos requeridos.
4. El sistema recoge los datos y los valida.
5. El sistema da de alta la categoría en el sistema.

•Flujo de eventos alternativo:

1.4 <Cancelación de proceso> *flujo principal*: El usuario cliente puede cancelar el proceso en cualquier momento, reiniciando de esta manera el caso de uso.

•Flujo de eventos excepcional:

<Fallo de conexión> *flujo principal*: El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

4. <Dato obligatorio en blanco>: El sistema muestra un mensaje de error indicando la existencia del dato en blanco, y retorna al usuario a la introducción de datos del producto.

4. <Formato de dato no válido>: El sistema muestra un mensaje de error indicando el dato cuyo formato no es válido, ofreciendo al usuario la posibilidad de modificarlo.

4. <Categoría ya existe >: El sistema muestra un mensaje de error indicando que la categoría que se quiere dar de alta ya existe en el sistema.

●**Escenarios:**

13.1. Alta de categoría.

13.2. Subcaso de uso “Consulta categoría”.

●**Objetivo:**

Permitir al usuario consultar los datos de las categorías.

●**Actor:**

Usuario administrador.

●**Flujo de eventos principal:**

1. El usuario administrador solicita al sistema la visualización de datos de categorías.
2. El sistema muestra los resultados, dando la posibilidad al usuario de mostrar el detalle de los datos para una determinada categoría.

●**Flujo de eventos alternativo:**

1.2 <Cancelación de proceso> : El usuario cliente puede cancelar el proceso en cualquier momento, reiniciando de esta manera el caso de uso.

●**Flujo de eventos excepcional:**

<Fallo de conexión> *flujo principal*: El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

2.<No hay resultados en la búsqueda> : El sistema muestra un mensaje al usuario administrador de que la búsqueda no produjo ningún resultado.

●**Escenarios:**

13.2. Visualización de categoría.

13.3. Subcaso de uso “Modificación categoría”.

●**Objetivo:**

Permitir al usuario modificar los datos de las categorías.

●**Actor:**

Usuario administrador.

●**Flujo de eventos principal:**

1. El usuario solicita al sistema realizar la modificación de los datos de una categoría.
2. El sistema muestra los datos actuales de la categoría y solicita los nuevos.
3. El usuario introduce los nuevos datos.
4. El sistema los recoge, previa validación de los mismos.

5. El sistema modifica los datos correspondientes, introducidos por el usuario.

●**Flujo de eventos alternativo:**

1..3 <Cancelación de proceso> : El usuario cliente puede cancelar el proceso en cualquier momento, reiniciando de esta manera el caso de uso.

●**Flujo de eventos excepcional:**

<Fallo de conexión> *flujo principal*: El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

2.<No hay resultados en la búsqueda>: El sistema muestra un mensaje al usuario administrador de que la búsqueda no produjo ningún resultado.

3. <Dato obligatorio en blanco>: El sistema muestra un mensaje de error indicando la existencia del dato en blanco, y retorna al usuario a la introducción de datos del producto.

3. <Formato de dato no válido>: El sistema muestra un mensaje de error indicando el dato cuyo formato no es válido, ofreciendo al usuario la posibilidad de modificarlo.

●**Escenarios:**

13.3. Modificación de datos de categoría.

13.4 Subcaso de uso “Baja categoría” .

●**Objetivo:**

Permitir al usuario dar de baja en el sistema a las categorías.

●**Actor:**

Usuario administrador.

●**Flujo de eventos principal:**

1. El usuario realiza la petición de baja al sistema para dar de baja una categoría.
2. El sistema da de baja la categoría.

●**Flujo de eventos alternativo:**

1..2 <Cancelación de proceso> : El usuario cliente puede cancelar el proceso en cualquier momento, reiniciando de esta manera el caso de uso.

●**Flujo de eventos excepcional:**

<Fallo de conexión> *flujo principal*: El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

2. <Categoría contiene productos> : El sistema muestra un mensaje de que no se puede eliminar la categoría por contener todavía en ese momento productos.

•Escenarios:

13.4. Baja de categoría.

14. Caso de uso “Gestionar datos Administrador”: include (Validar usuario).

•Objetivo:

El objetivo final es permitir al usuario administrador realizar las operaciones propias de gestión de datos personales: modificación, visualización y baja en el sistema.

•Actor:

Usuario administrador.

•Pasos:

1. El usuario administrador podrá crear una consulta referente a sus datos personales.
2. El usuario administrador podrá realizar una modificación de sus datos personales.
3. El usuario administrador podrá darse de baja como vendedor en el sistema.

Este es un caso de uso abstracto que se introduce para englobar las operaciones de gestión de los datos de vendedor, que se explotan en subcasos de uso que se muestran en la siguiente figura:

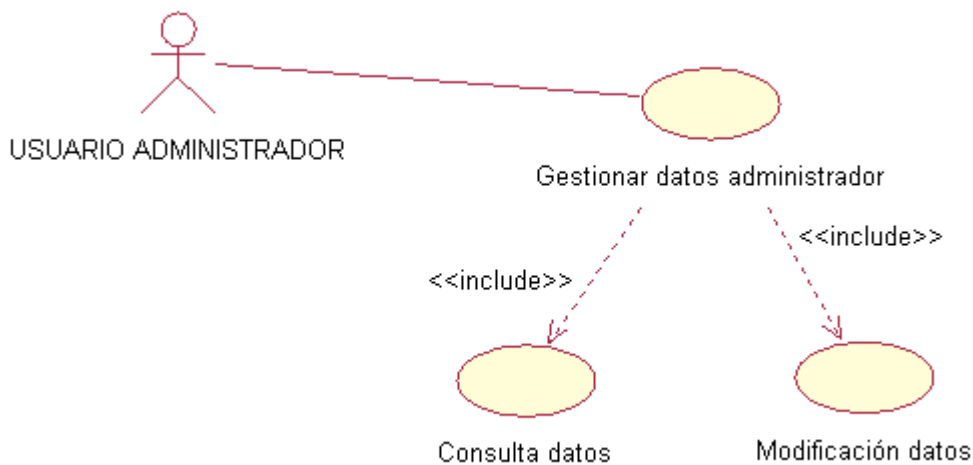


Figura 6. Diagrama de Caso de uso “Gestionar datos administrador”.

14.1. Subcaso de uso “Consulta datos administrador”.

•Objetivo:

El objetivo es permitir al usuario visualizar los datos registrados en el sistema.

•Actor:

Usuario administrador.

●**Flujo de eventos principal:**

5. El usuario solicita al sistema realizar una consulta sobre sus datos.
6. El sistema solicita las preferencias de visualización y los posibles filtros de información.
7. El usuario establece unas determinadas preferencias y filtrados de información.
8. El sistema recoge la petición y muestra la información requerida.

●**Flujo de eventos alternativos:**

1.4 <Cancelación de proceso> : El usuario cancela el proceso de compra en cualquier momento.

●**Flujo de eventos excepcional:**

<Fallo de conexión> *común para ambos flujos*: El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

3. <Rango especificado no válido>: El sistema muestra un mensaje indicando que el rango especificado de búsqueda no es válido, lo que se puede deber a que la fecha de registro como vendedor no se encuentra dentro del rango.

4. <No hay resultados en la búsqueda>: El sistema muestra un mensaje indicando que la búsqueda en el historial de vendedores no produjo ningún resultado de acuerdo con el rango establecido.

●**Escenarios:**

14.1. Visualización datos administrador.

14.2. Subcaso de uso “Modificación datos administrador”.

●**Objetivo:**

El objetivo es permitir al usuario administrador modificar sus datos registrados en el sistema.

●**Actor:**

Usuario administrador.

●**Flujo de eventos principal:**

1. El usuario solicita al sistema realizar una modificación sobre sus datos.
2. El sistema muestra los datos actuales del administrador y solicita los nuevos.
3. El administrador introduce los nuevos datos personales
4. El sistema los recoge, previa validación de los mismos.
5. El sistema modifica los datos correspondientes, introducidos por el usuario.

●**Flujo de eventos alternativos:**

1.5 <Cancelación de proceso>: El usuario vendedor cancela el proceso de compra en cualquier momento.

•**Flujo de eventos excepcional:**

1..5. <Fallo de conexión> : El sistema muestra este mensaje de error si falla la conexión con el servidor en el momento de realizar la operación.

4. <Dato obligatorio en blanco>: El sistema muestra un mensaje de error y se le da al usuario la posibilidad de introducir el dato que faltaba.

5. <Formato de dato no válido>: El sistema muestra un mensaje de error indicando el dato cuyo formato no es válido, ofreciendo al usuario la posibilidad de modificarlo.

•**Escenarios:**

- 14.2. Modificación de datos vendedor.

DIAGRAMAS DE INTERACCIÓN CONCEPTUALES.

Escenario 1.1 “Listar productos según condiciones”.

- **Caso de uso:** 1. “Consultar productos”.
- **Escenario:** 1.1. Listar productos según condiciones.
- **Objetivo:** Permitir al usuario cliente listar productos según las preferencias y filtrados de información establecidas en el criterio de visualización.
- **Descripción:** El usuario solicita la visualización de los datos de productos y el sistema solicita las preferencias de búsqueda y los posibles filtros de información. El usuario establece unas determinadas preferencias y filtrados de información y el sistema recoge la petición y muestra una relación de productos, o categorías en su caso, que cumplen con las condiciones establecidas.

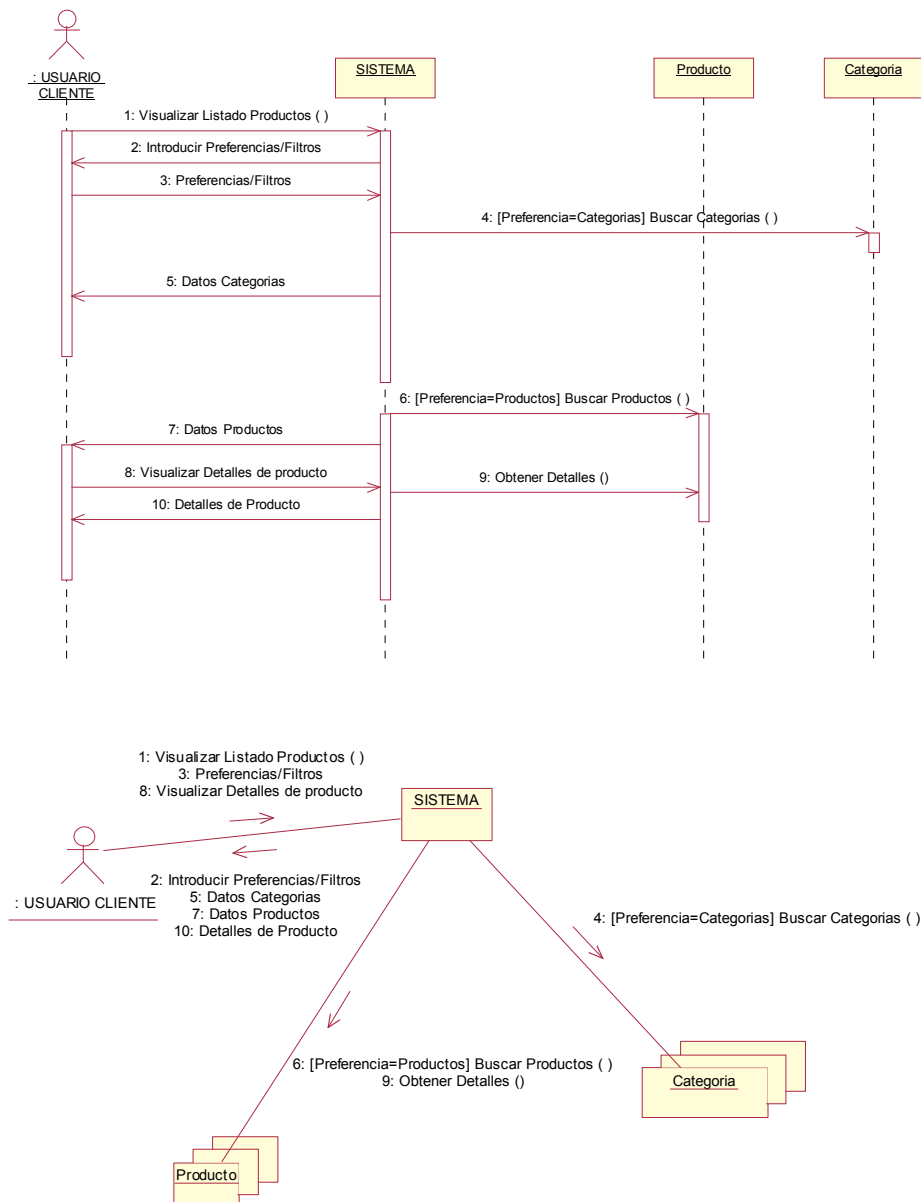


Figura 1.1. Escenario “Listar productos según condiciones”.

Escenario 2.1 “Visualización de datos personales”.

- **Caso de uso:** 2. “Gestionar datos cliente”.
- **Escenario:** 2.2. Visualización de datos personales.
- **Objetivo:** Permitir al usuario cliente visualizar sus datos personales que constan en el sistema.
- **Descripción:** El usuario solicita al sistema realizar la consulta de sus datos personales y el sistema muestra los datos actuales del usuario.

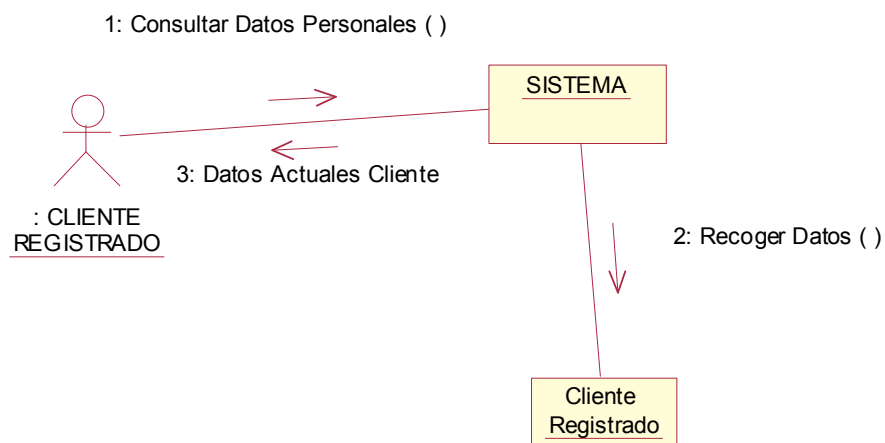
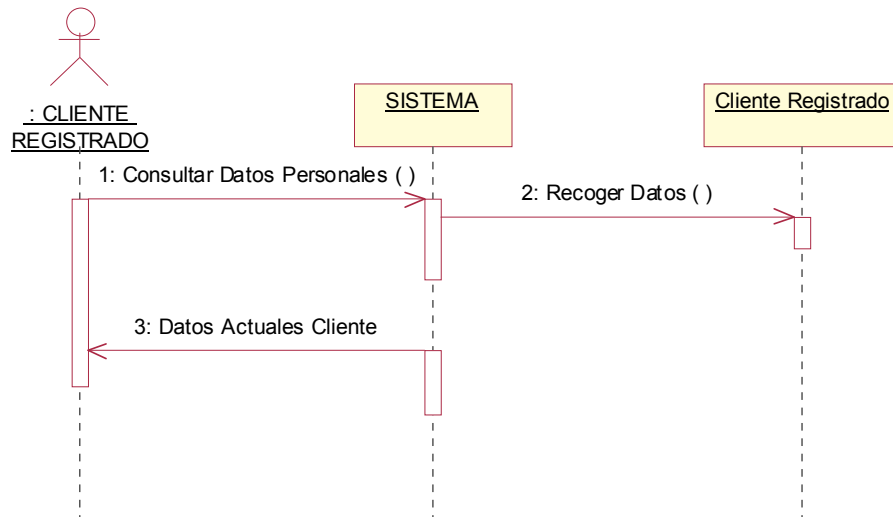


Figura 2.1. Escenario “Visualización de datos personales”.

Escenario 2.2 "Modificación de datos personales".

- **Caso de uso:** 2. "Gestionar datos cliente".
- **Escenario:** 2.1. Modificación de datos personales.
- **Objetivo:** Permitir al usuario cliente modificar sus datos personales.
- **Descripción:** El usuario solicita al sistema modificar sus datos personales. El sistema muestra los datos actuales del usuario y el usuario introduce los nuevos datos personales. El sistema los recoge, previa validación de los mismos y modifica los datos correspondientes, introducidos por el usuario.

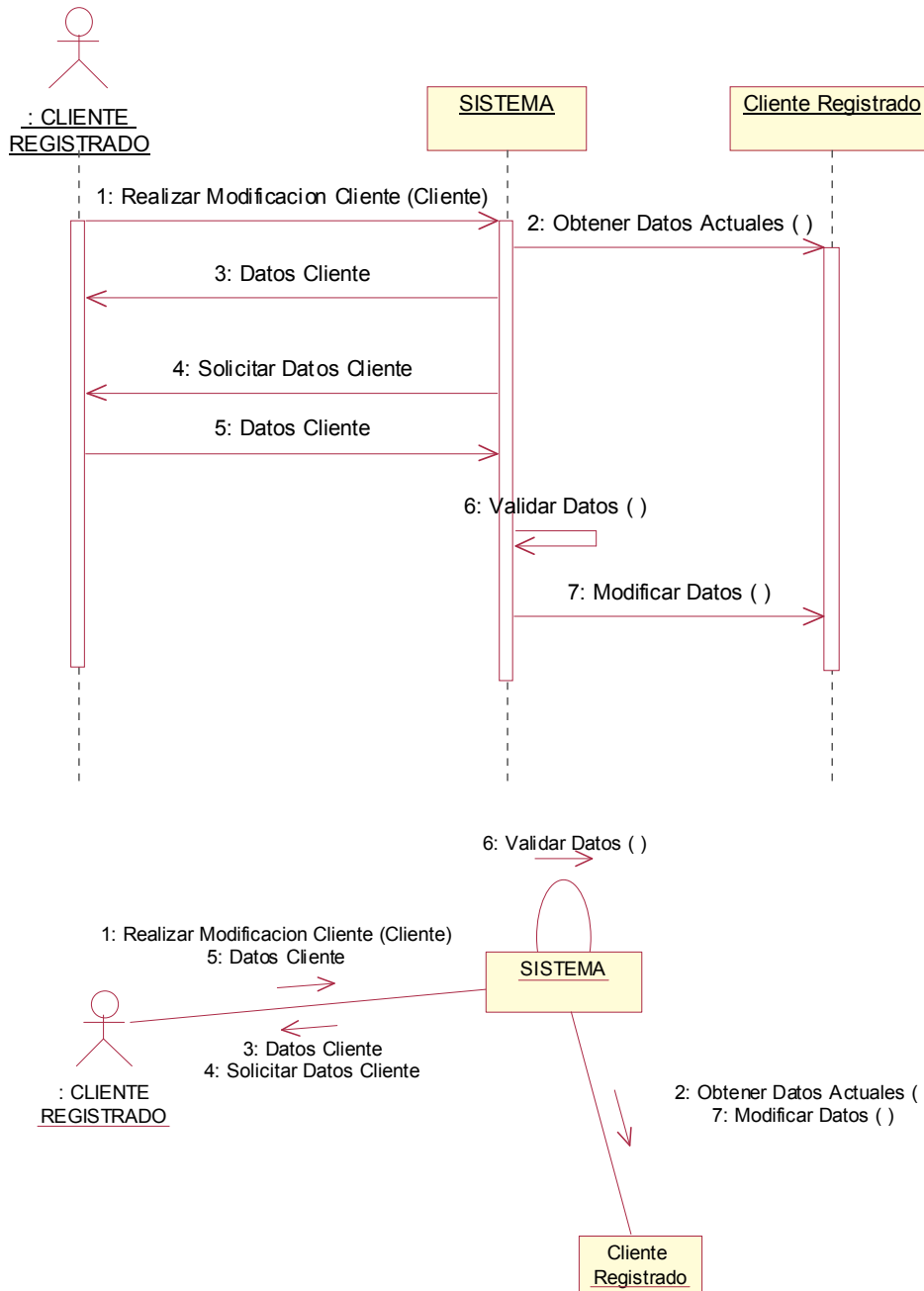


Figura 2.2. Escenario "Modificación de datos personales".

Escenario 3.1 “Alta cliente”.

- **Caso de uso:** 3.”Registrar cliente”.
- **Escenario:** 3.1. Registrar cliente.
- **Objetivo:** Registrar como usuario cliente a un usuario no registrado y por lo tanto darle de alta en el sistema.
- **Descripción:** El usuario no registrado hace una petición de registro al sistema y el sistema solicita al usuario que introduzca los datos. El usuario introduce los datos requeridos y el sistema recoge los datos, previa validación y da de alta al usuario en el sistema.

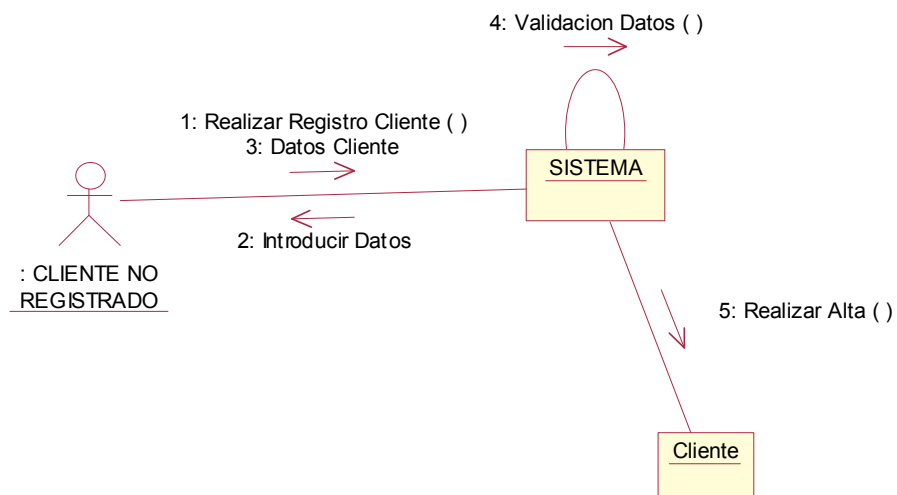
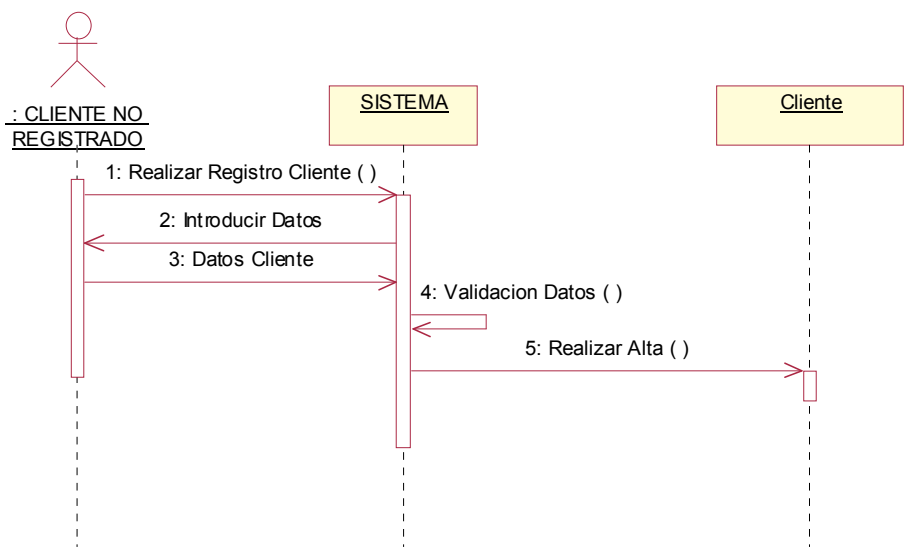


Figura 3.1. Escenario “Alta de cliente”.

Escenario 4.1 "Baja cliente".

- **Caso de uso:** 4. "Gestionar datos cliente".
- **Escenario:** 2.3. Causar baja en el sistema.
- **Objetivo:** Permitir al usuario cliente darse de baja del sistema.
- **Descripción:** El usuario registrado hace una petición para darse de baja del sistema como cliente y el sistema da de baja al cliente.

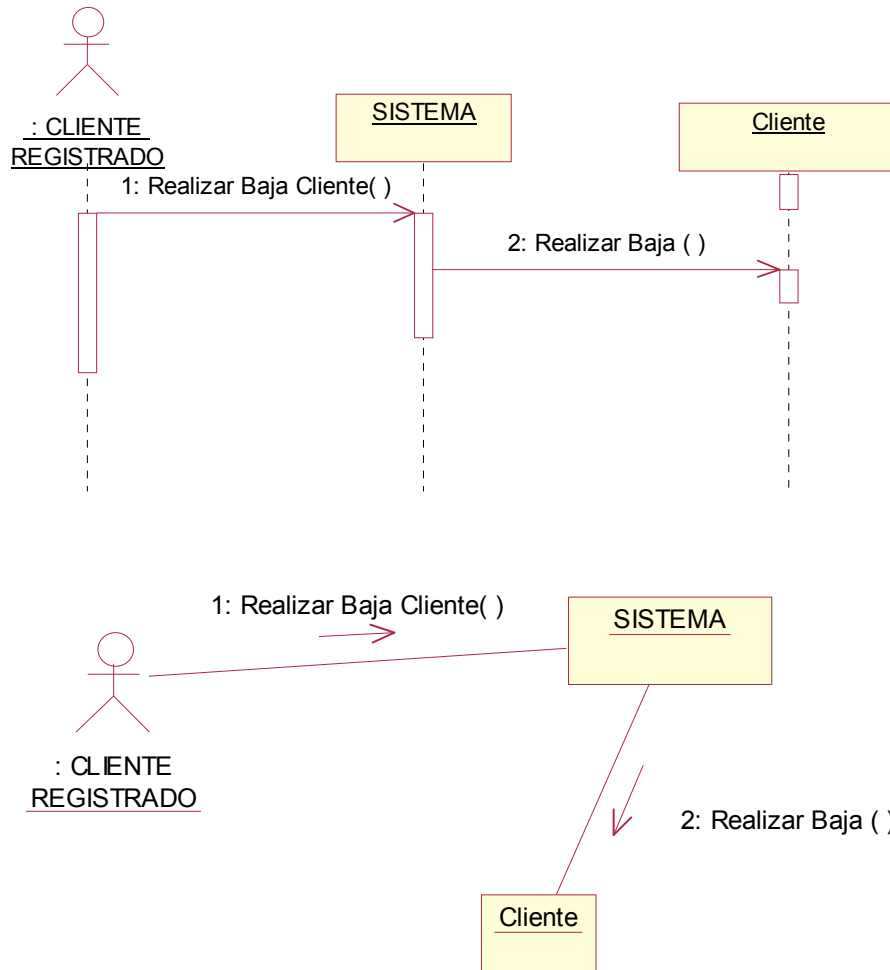


Figura 4.1. Escenario "Baja de cliente".

Escenario 5.1 “Añadir un producto al carrito”.

- **Caso de uso:** 5. “Gestión carrito de compra”.
- **Escenario:** 5.1. Añadir un producto al carrito.
- **Objetivo:** Permitir al usuario cliente añadir un producto a su carrito de la compra
- **Descripción:** El usuario añade los productos al carrito de la compra indicando el número de unidades que desea. El sistema añade la relación de productos al carrito de la compra.

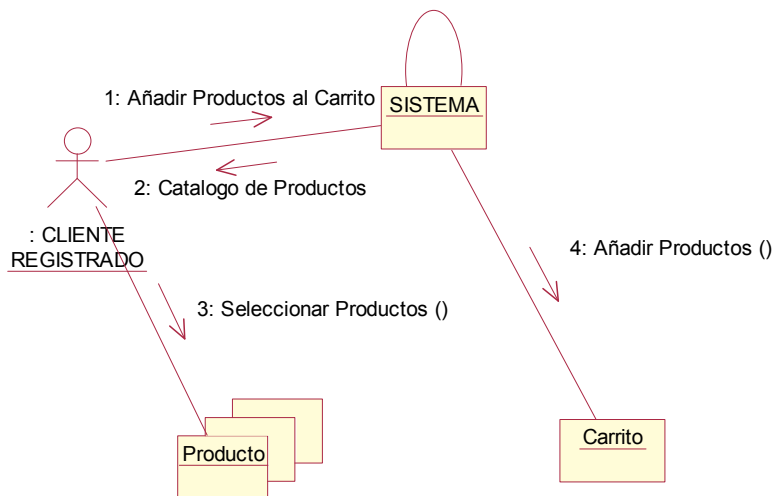
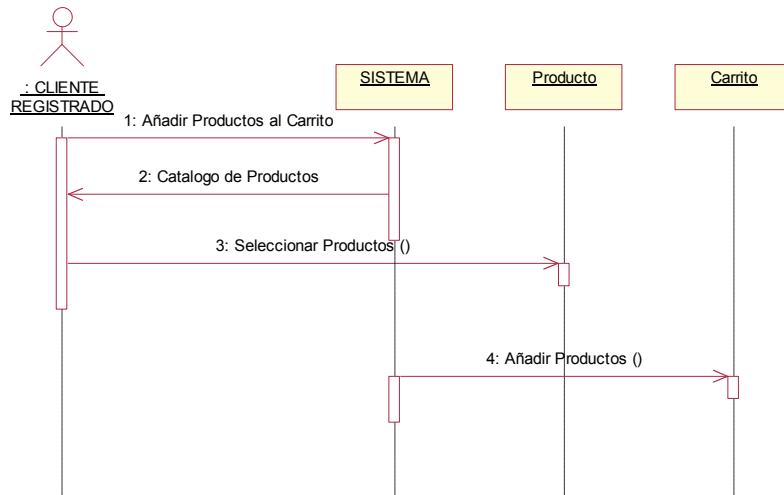


Figura 5.1. Escenario “Añadir un producto al carrito”.

Escenario 5.2 "Visualizar el contenido del carrito".

- **Caso de uso:** 5."Gestión carrito de compra".
- **Escenario:** 5.2. Visualizar el contenido del carrito.
- **Objetivo:** Permitir al usuario cliente visualizar el contenido actual del carrito en ese momento.
- **Descripción:** El usuario solicita al sistema ver el contenido de su carrito de la compra y e sistema muestra al usuario el contenido de su carrito.

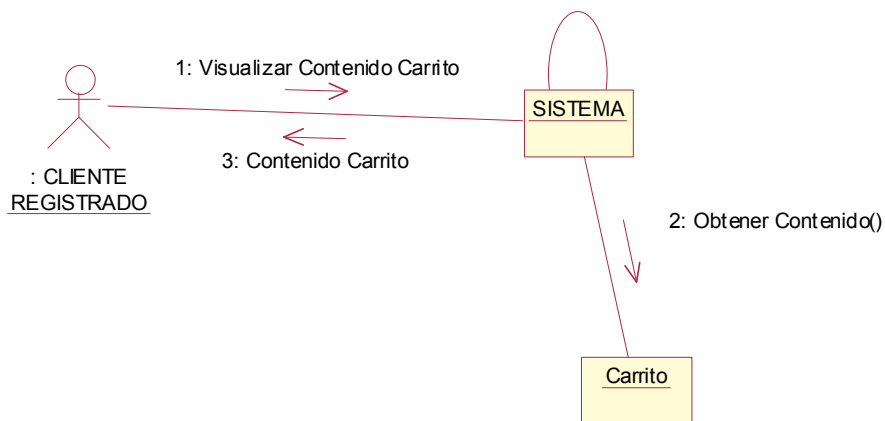
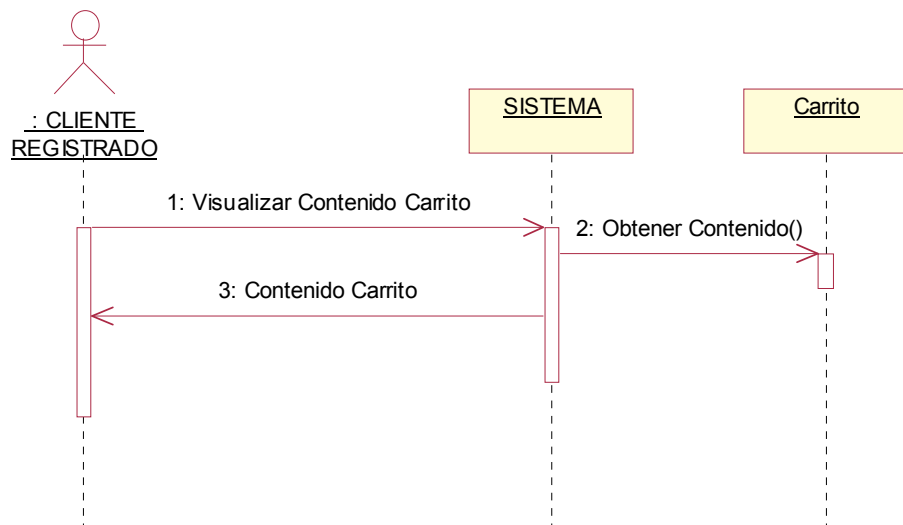


Figura 5.2. Escenario "Visualizar contenido del carrito".

Escenario 5.3 “Modificar las unidades de un producto del carrito.”

- **Caso de uso:** 5. “Gestión carrito de compra”.
- **Escenario:** 5.3. Modificar las unidades de un producto del carrito.
- **Objetivo:** Permitir al usuario cliente modificar el número de unidades de un determinado producto del carrito de la compra.
- **Descripción:** El usuario solicita modificar el contenido de su carrito de la compra y el sistema muestra el contenido actual del carrito de la compra. El usuario lleva a cabo las modificaciones oportunas y el sistema actualiza el carrito de la compra.

Escenario 5.4 “Vaciar el carrito”.

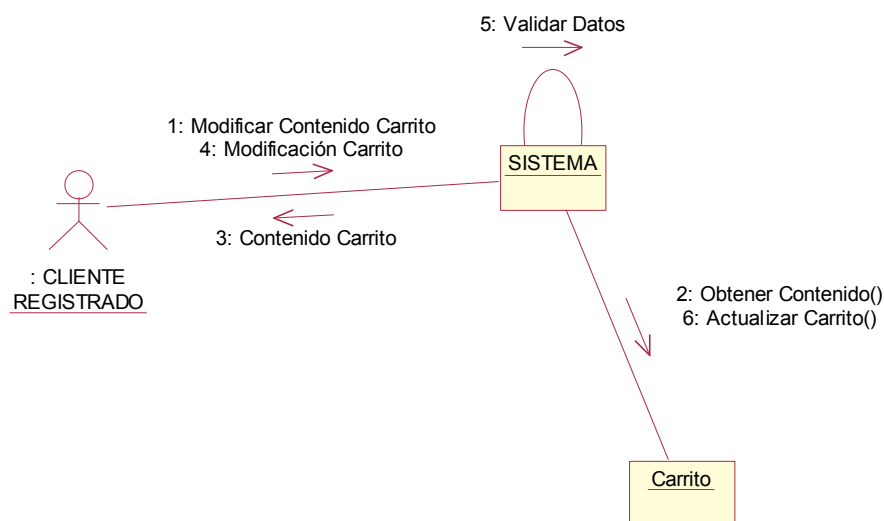
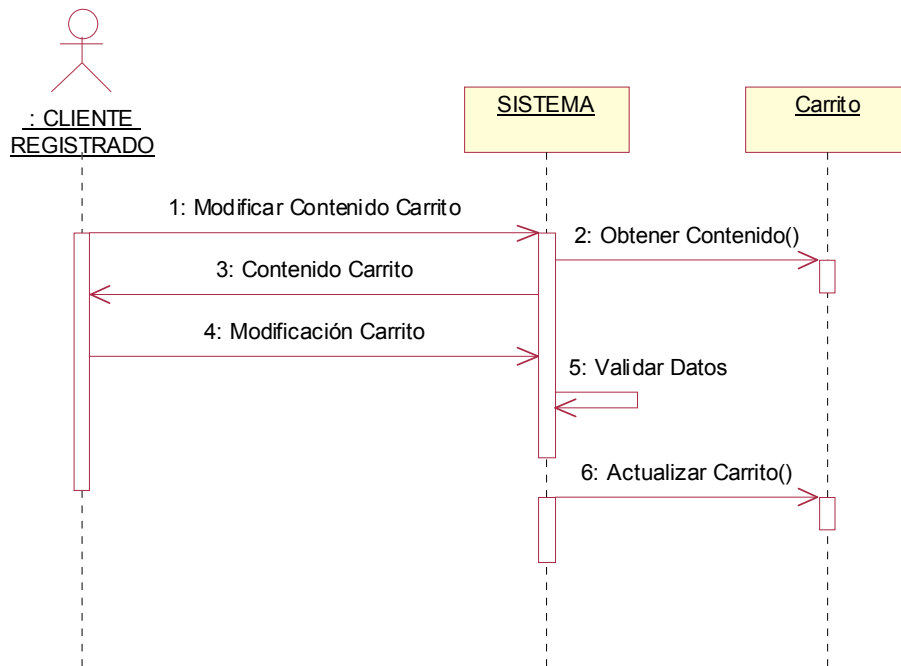


Figura 5.3. Escenario “Modificación de unidades de carrito”.

- **Caso de uso:** 5. "Gestión carrito de compra".
- **Escenario:** 5.4. Vaciar el carrito.
- **Objetivo:** Permitir al usuario cliente vaciar por completo el carrito de la compra.
- **Descripción:** El usuario solicita al sistema vaciar por completo su carrito de la compra y el sistema vacía el carrito de la compra.

Escenario 5.5 "Borrar un producto del carrito".

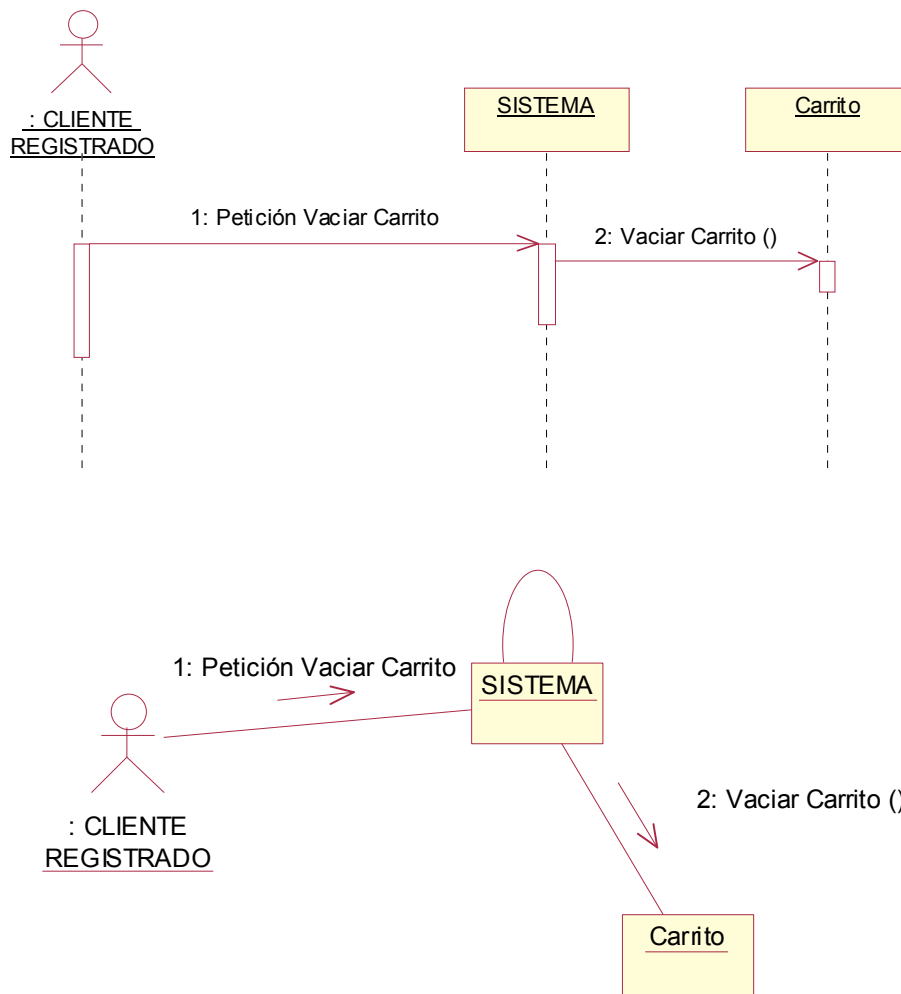


Figura 5.4. Escenario "Vaciar carrito".

- **Caso de uso:** 5. “Gestión carrito de compra”.
- **Escenario:** 5.5. Borrar un producto del carrito.
- **Objetivo:** Permitir al usuario cliente eliminar un producto del carrito de la compra.
- **Descripción:** El usuario solicita al sistema eliminar de su carrito de la compra un producto, y el sistema lo elimina.

Escenario 6.1 “Compra de pedido”.

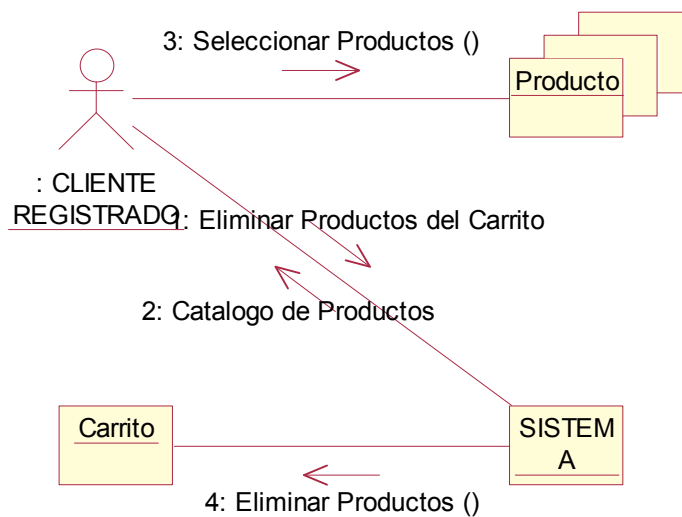
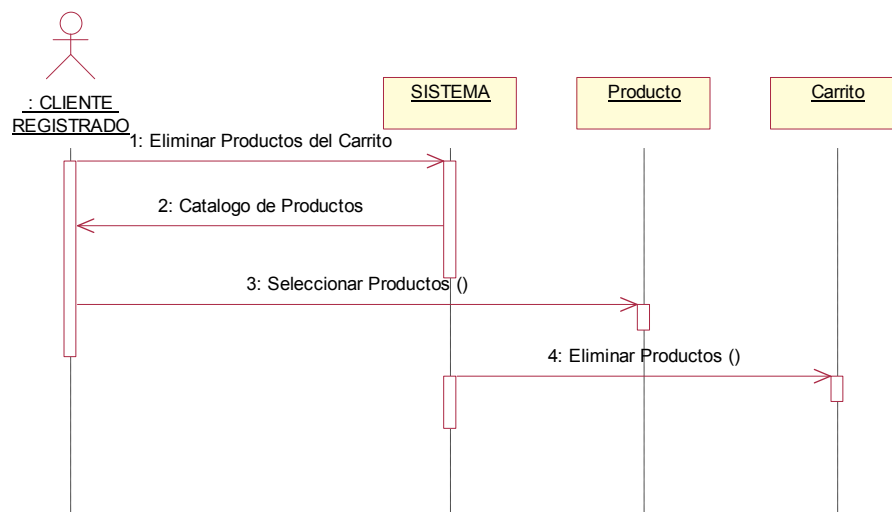


Figura 5.5. Escenario “Eliminar producto del carrito”.

- **Caso de uso:** 6. “Comprar”.
- **Escenario:** 6.1. Compra de pedido.
- **Objetivo:** Permitir al usuario cliente registrado realizar un pedido.
- **Descripción:** El usuario solicita al sistema realizar la compra de los productos existentes en su carrito de la compra. El sistema aplica los descuentos correspondientes y muestra el pedido actual con los datos del usuario y del carrito de la compra, después modifica si es necesario la categoría de cliente y genera un informe que será mandado al cliente con los datos de su pedido.

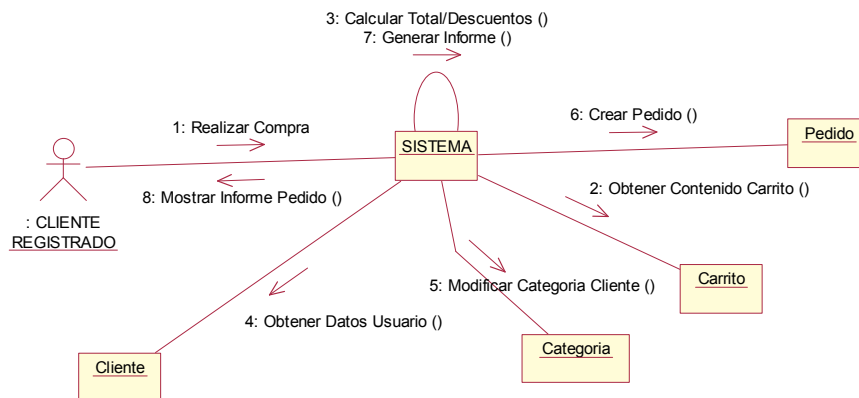
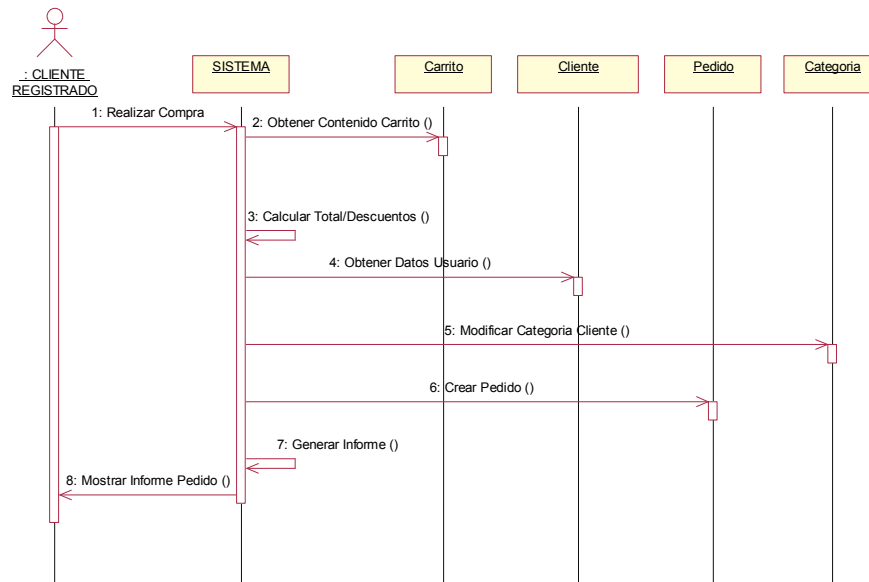


Figura 6.1. Escenario “Compra de pedido”.

Escenario 7.1 “Establecer Forma de Pago”.

- **Caso de uso:** 7. “Pagar”.
- **Escenario:** 7.1. Establecer Forma de Pago.
- **Objetivo:** Permitir al usuario registrado, que va a llevar a cabo un pedido, elegir una forma de pago concreta.
- **Descripción:** El usuario solicita al sistema indicar el método de pago y el sistema solicita al usuario que indique el método de pago. El usuario establece el método de pago y el sistema guarda la forma de pago.

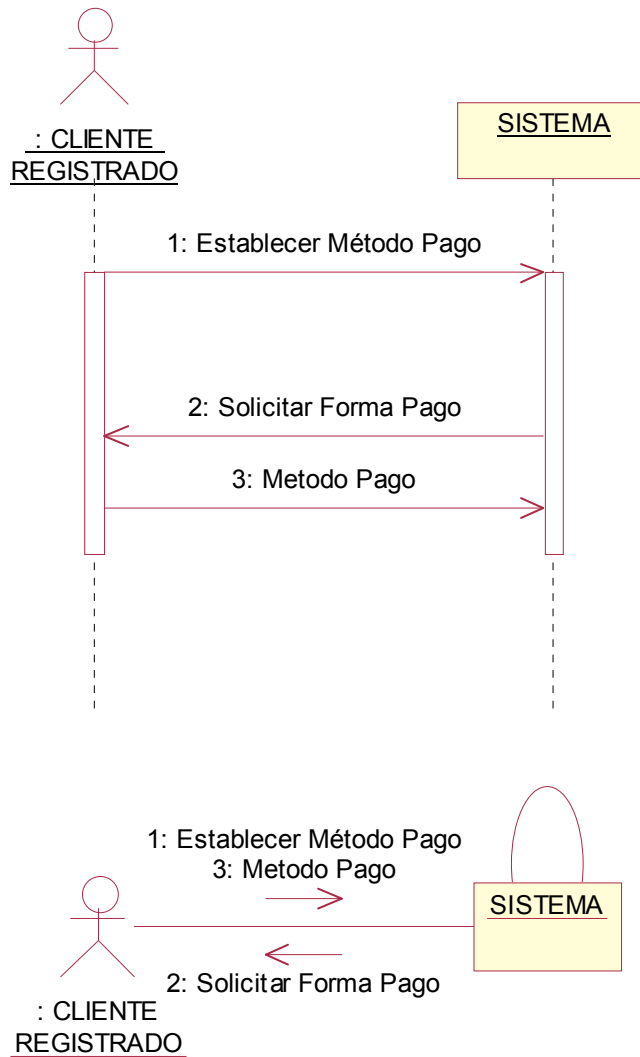


Figura 7.1. Escenario “Establecer forma de pago”.

Escenario 8.1 "Validación de usuario".

- **Caso de uso:** 8. "Validar usuario".
- **Escenario:** 8.1. Validación de usuario.
- **Objetivo:** Comprobar que el usuario cliente está registrado como tal en el sistema y por tanto tiene acceso a las operaciones propias de usuario cliente.
- **Descripción:** Un usuario realiza una petición de acceso al sistema. El sistema requiere la autenticación de este. El usuario se identifica ante el sistema y el sistema comprueba los datos, permitiendo su acceso al sistema.

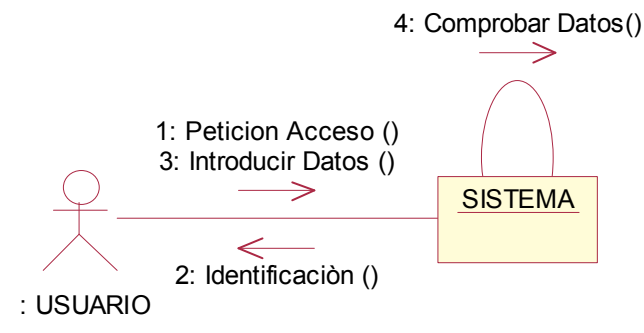
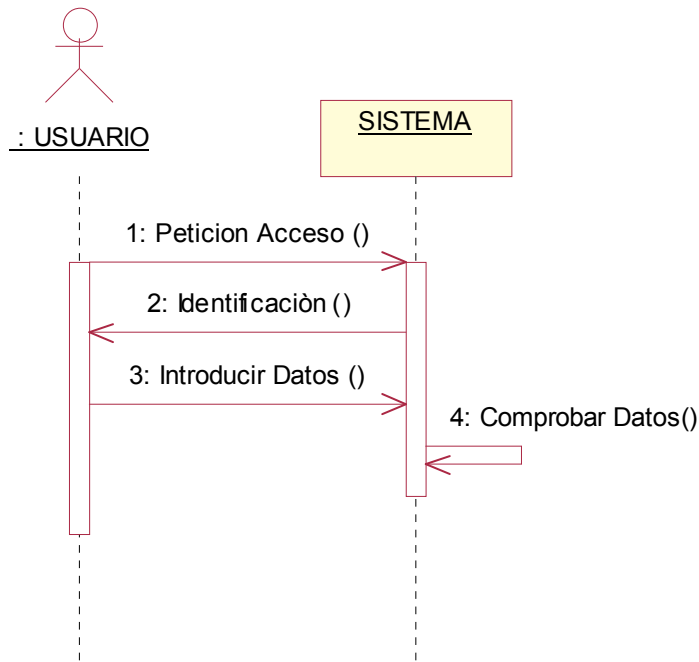


Figura 8.1. Escenario "Validación de usuario".

Escenario 9.1 “Visualización datos vendedor”.

- **Caso de uso:** 9. “Gestionar datos vendedor”.
- **Escenario:** 9.1. Visualización de datos vendedor.
- **Objetivo:** Permitir al usuario vendedor visualizar los datos personales que constan en el sistema acerca de él.
- **Descripción:** El usuario solicita al sistema realizar una consulta sobre sus datos. El sistema solicita las preferencias de visualización y los posibles filtros de información. El usuario establece unas determinadas preferencias y filtrados de información y el sistema recoge la petición y muestra la información requerida.

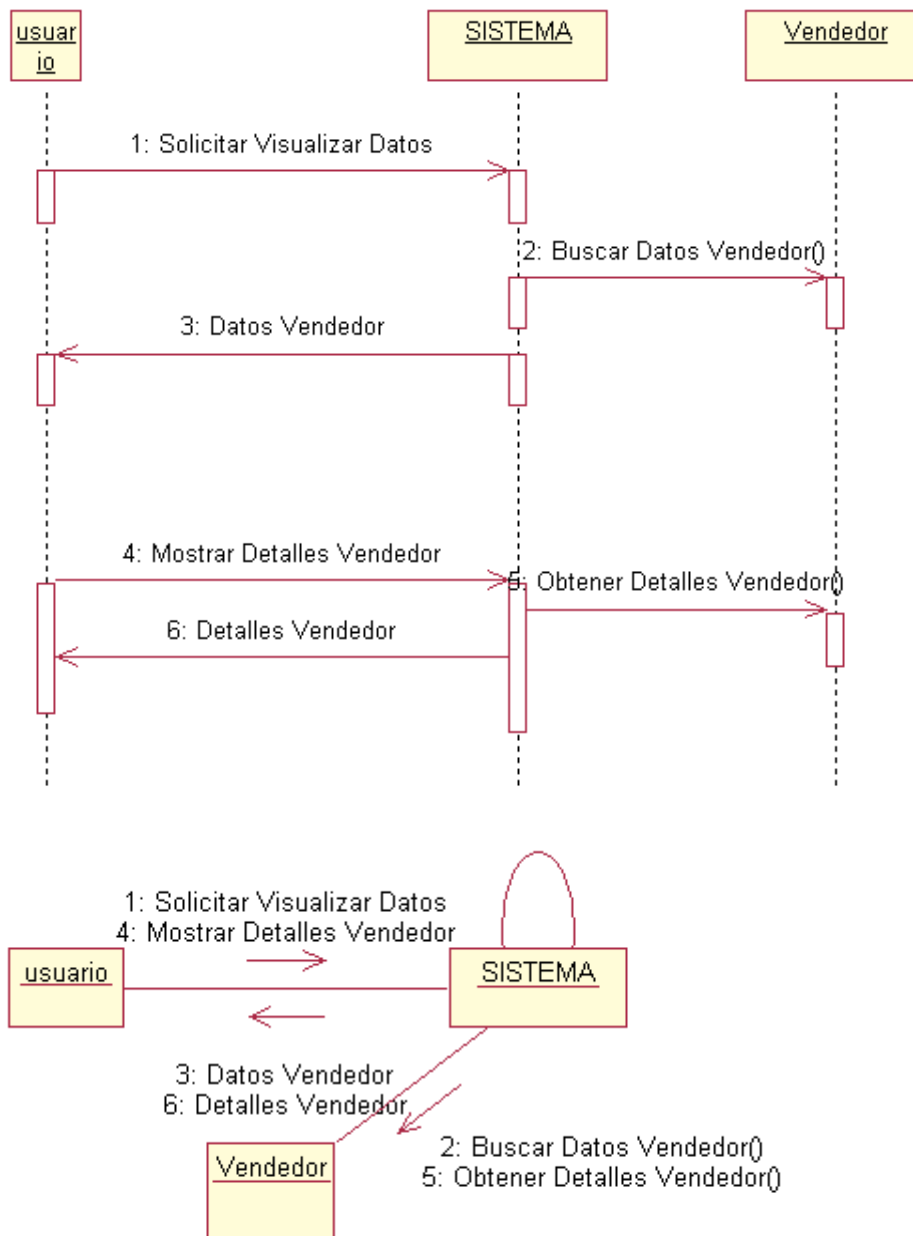


Figura 9.1. Escenario “Visualización de datos de vendedor”.

Escenario 9.2 “Modificación de datos vendedor”.

- **Caso de uso:** 9. “Gestionar datos vendedor”.
- **Escenario:** 9.2. Modificación de datos vendedor.
- **Objetivo:** Permitir al usuario vendedor modificar sus datos personales.
- **Descripción:** El vendedor solicita al sistema realizar una modificación sobre sus datos. El sistema muestra los datos actuales del vendedor y solicita los nuevos. El vendedor introduce los nuevos datos personales y el sistema los recoge, previa validación de los mismos. Finalmente el sistema modifica los datos correspondientes, introducidos por el usuario.

Escenario 9.3 “Causar baja en el sistema”.

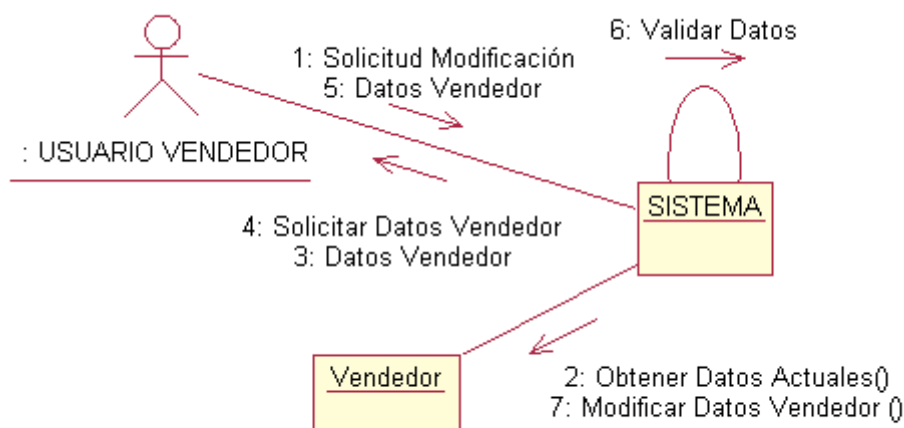
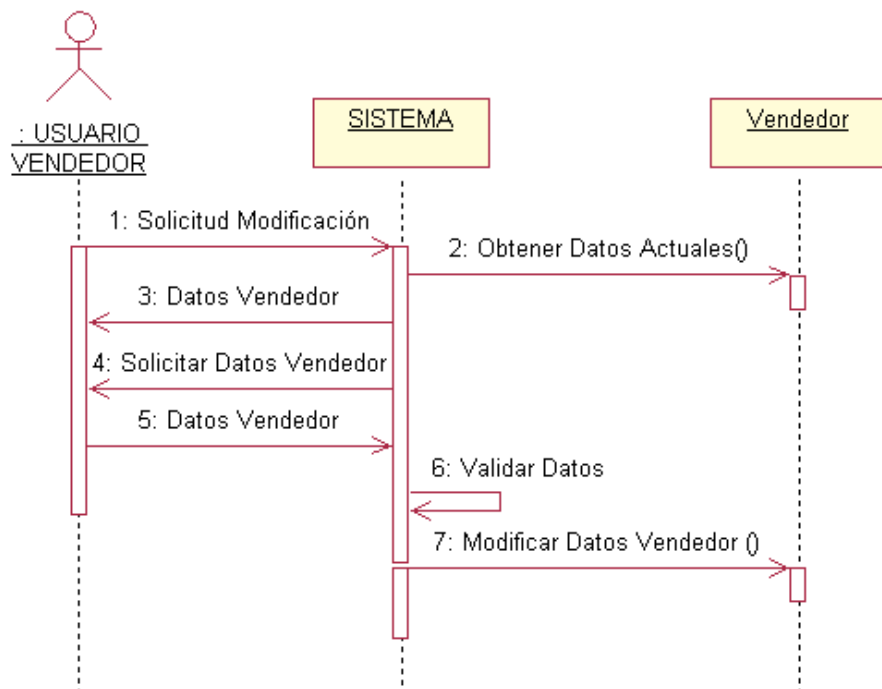


Figura 9.2. Escenario “Modificación de datos de vendedor”.

- **Caso de uso:** 9. “Gestionar datos vendedor”.
- **Escenario:** 9.3. Causar baja en el sistema.
- **Objetivo:** Permitir al usuario vendedor causar baja en el sistema.
- **Descripción:** El usuario solicita al sistema realizar una baja en el sistema y el sistema realiza la baja del vendedor.

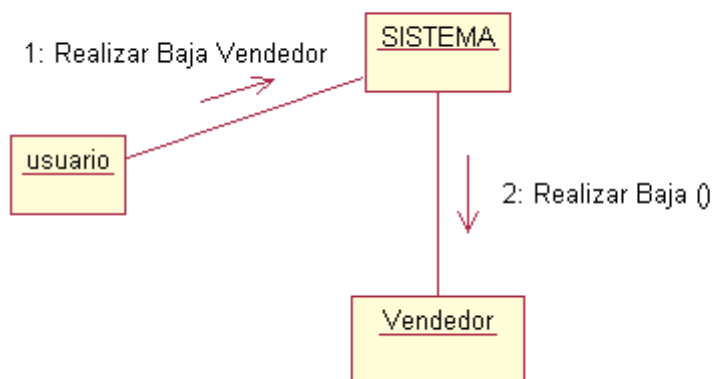
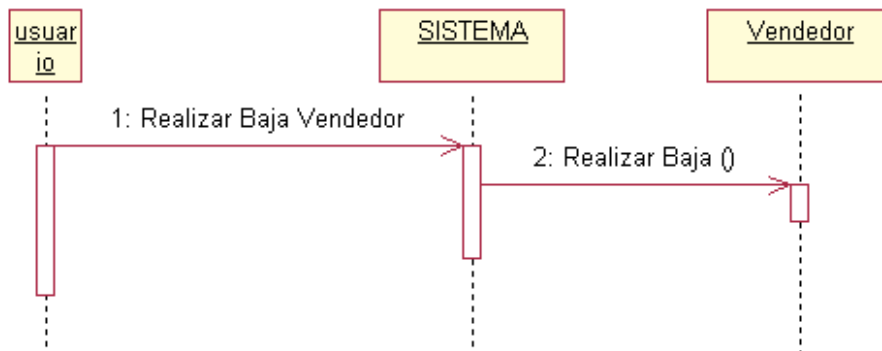


Figura 9.3. Escenario “Causar baja en el sistema”.

Escenario 10.1 “Alta persona contacto”.

- **Caso de uso:** 10. “Gestionar persona contacto”.
- **Escenario:** 10.1. Modificación de datos personales persona contacto.
- **Objetivo:** El objetivo es permitir al usuario vendedor dar de alta a una persona de contacto dentro del sistema.
- **Descripción:** El vendedor solicita al sistema dar de alta a una persona de contacto en el sistema y el sistema solicita los datos de dicha persona de contacto. El vendedor introduce al sistema los datos de dicha persona y el sistema los recoge previa validación de los mismos finalmente el sistema da de alta a la persona de contacto.

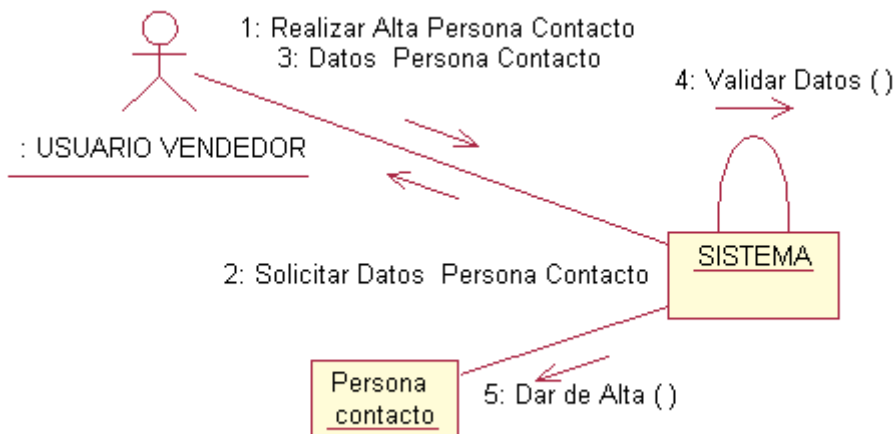
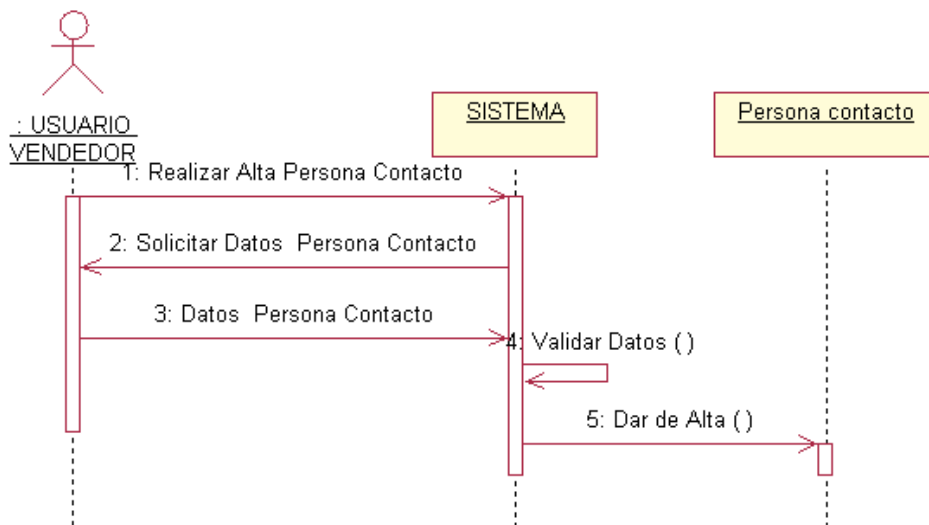


Figura 10.1. Escenario “Alta persona de contacto”.

Escenario 10.2 “Visualización de datos personales persona contacto”.

- **Caso de uso:** 10. “Gestionar persona contacto”.
- **Escenario:** 10.2. Visualización de datos personales persona contacto.
- **Objetivo:** Permitir al usuario vendedor visualizar los datos personales que constan en el sistema acerca de la persona de contacto.
- **Descripción:** El usuario vendedor solicita al sistema la visualización de los datos de la persona de contacto y el sistema muestra los resultados, dando la posibilidad al usuario de mostrar el detalle de los datos de la persona de contacto.

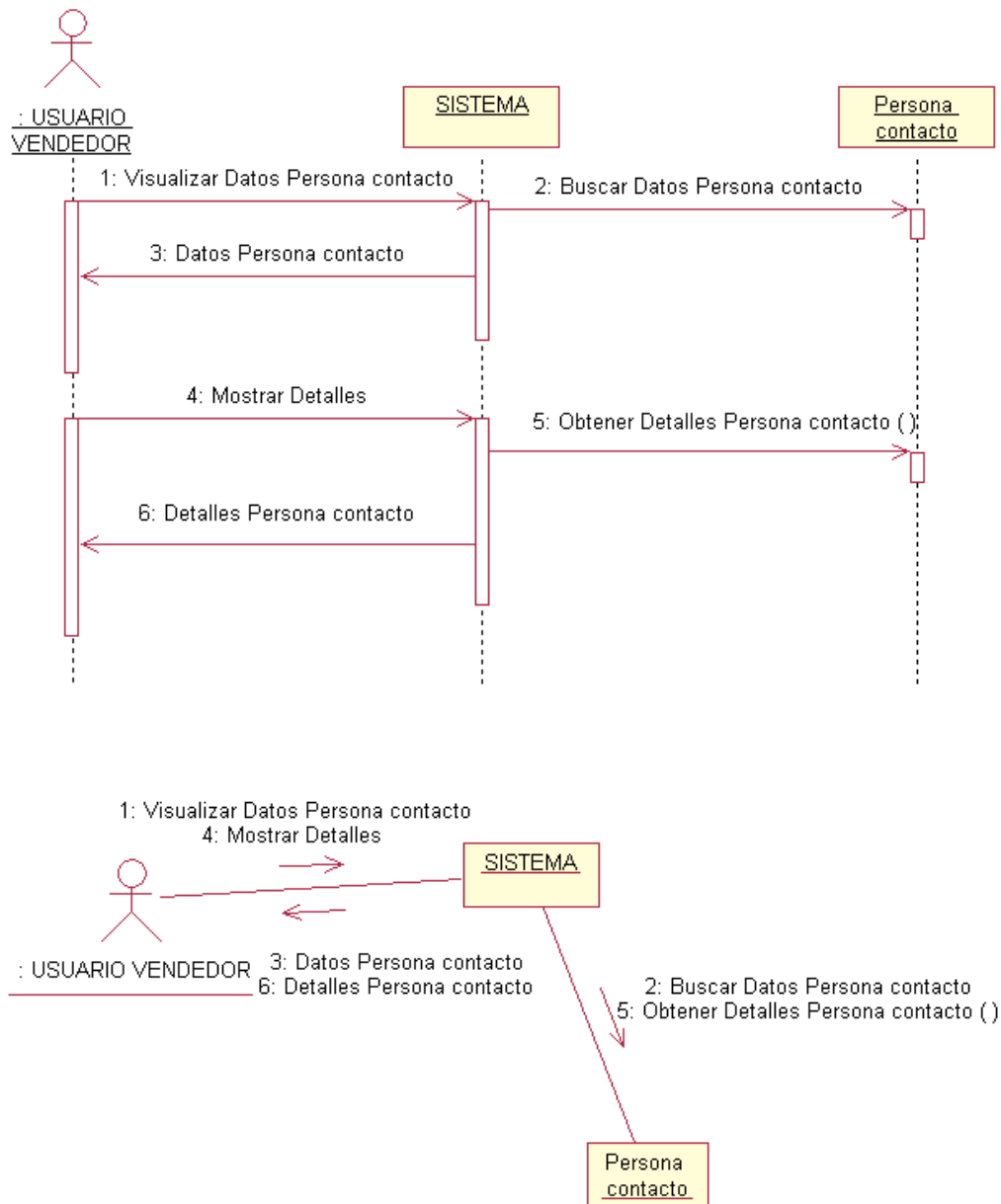


Figura 10.2. Escenario “Visualización de datos de persona de contacto”.

Escenario 10.3 "Modificación de datos personales persona contacto".

- **Caso de uso:** 10. "Gestionar persona contacto".
- **Escenario:** 10.3. Modificación de datos personales persona contacto.
- **Objetivo:** Permitir al usuario vendedor modificar los datos personales de la persona de contacto de su empresa que constan en el sistema.
- **Descripción:** El vendedor solicita al sistema realizar una modificación sobre los datos de la persona de contacto. El sistema muestra los datos actuales de la persona de contacto y solicita los nuevos. El vendedor introduce los nuevos datos y el sistema los recoge, previa validación de los mismos. Finalmente el sistema modifica los datos correspondientes, introducidos por el usuario.

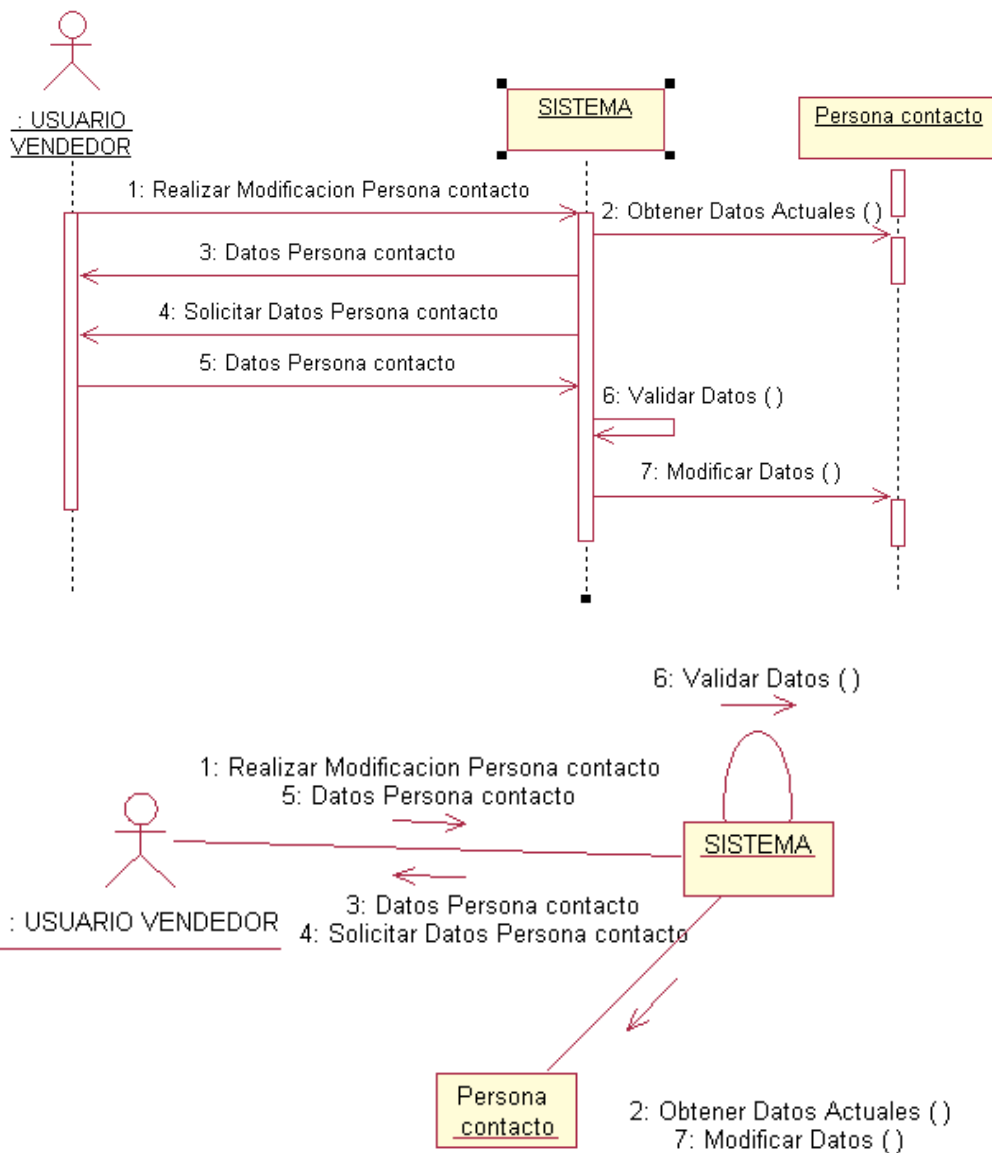


Figura 10.3. Escenario "Modificación de datos personales de persona de contacto".

Escenario 10.4 “Causar baja en el sistema persona contacto”.

- **Caso de uso:** 10. “Gestionar persona contacto”.
- **Escenario:** 10.4. Causar baja en el sistema persona contacto.
- **Objetivo:** Permitir al usuario vendedor dar de baja en el sistema a la persona de contacto.
- **Descripción:** El usuario solicita al sistema realizar una baja en el sistema y el sistema realiza la baja de la persona de contacto.

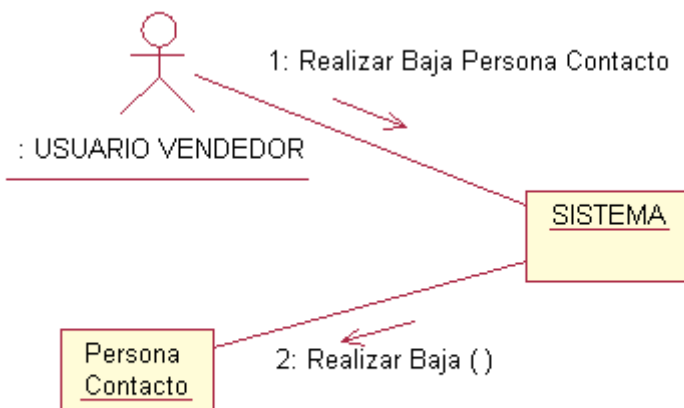
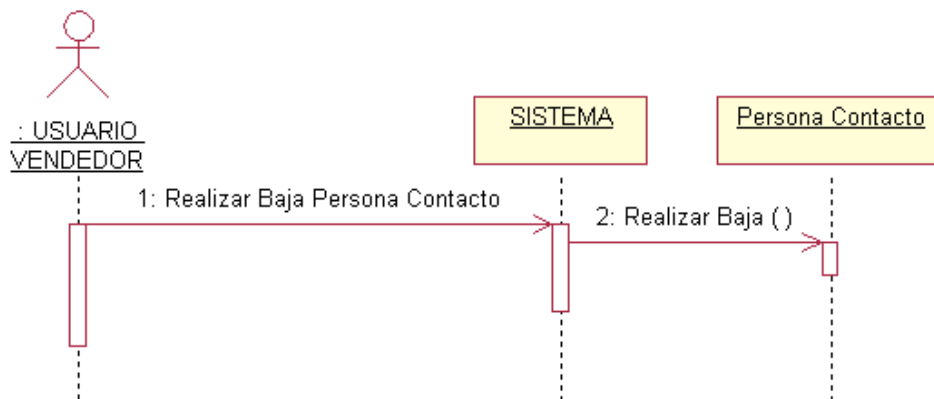


Figura 10.4. Escenario “Causar baja en el sistema persona de contacto”.

Escenario 11.1 "Alta productos".

- **Caso de uso:** 11. "Gestionar productos vendedor".
- **Escenario:** 11.1. Alta productos vendedor.
- **Objetivo:** Permitir al usuario vendedor dar de alta un producto.
- **Descripción:** El usuario solicita al sistema realizar el alta de un nuevo producto, el sistema pide los datos referentes al producto y el usuario introduce los datos requeridos. El sistema recoge los datos, los valida y da de alta el producto en el sistema.

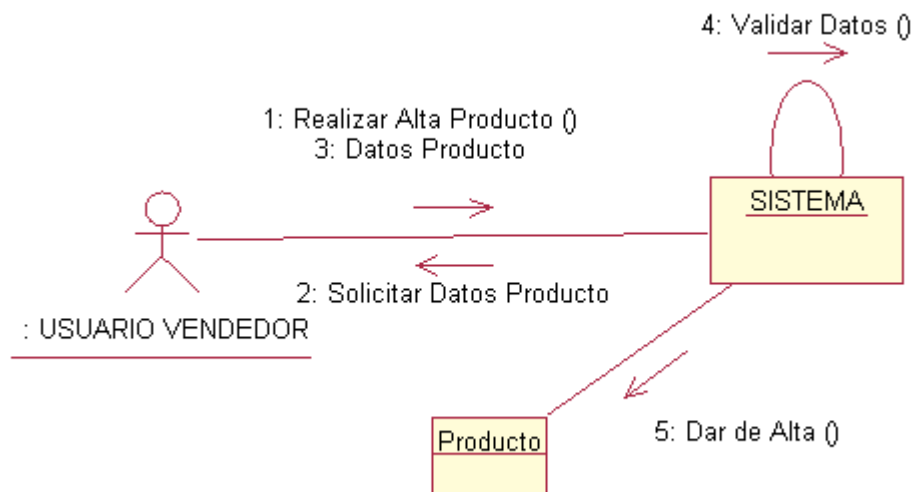
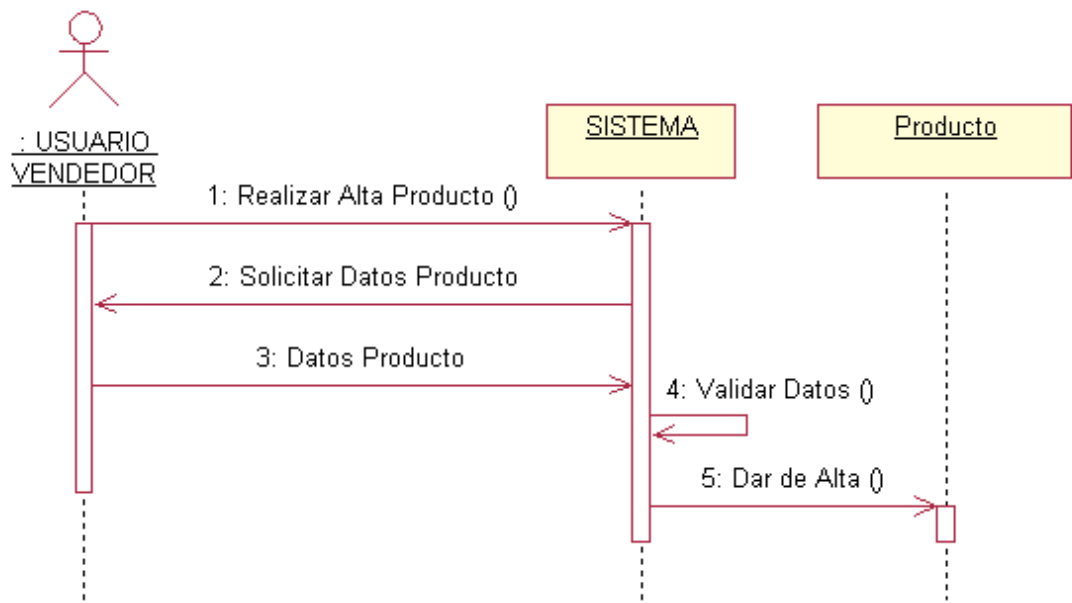


Figura 11.1. Escenario "Alta de producto de vendedor".

Escenario 11.2 “Visualización de datos de producto”.

- **Caso de uso:** 11. “Gestionar productos Vendedor”.
- **Escenario:** 11.2. Visualización de datos de producto.
- **Objetivo:** Permitir al usuario vendedor visualizar los datos de un determinado producto.
- **Descripción:** El usuario vendedor solicita al sistema la visualización de datos de productos y el sistema muestra los resultados, dando la posibilidad al usuario de mostrar el detalle de los datos para un determinado producto.

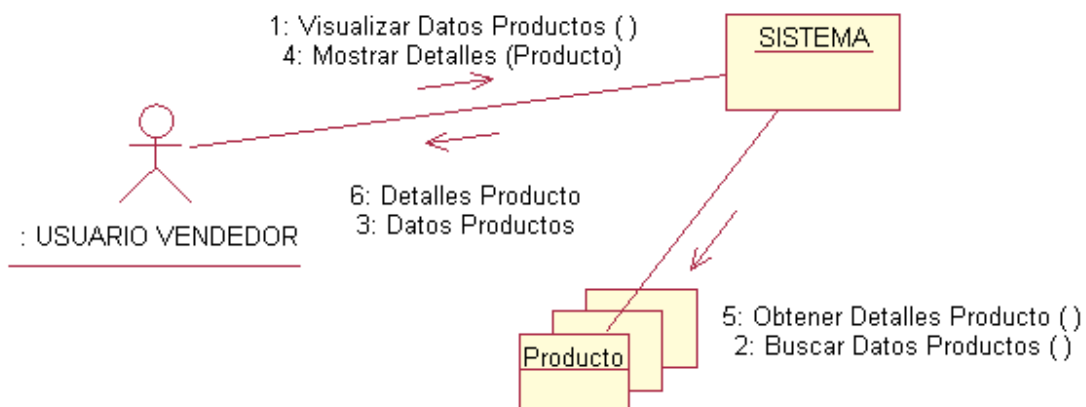
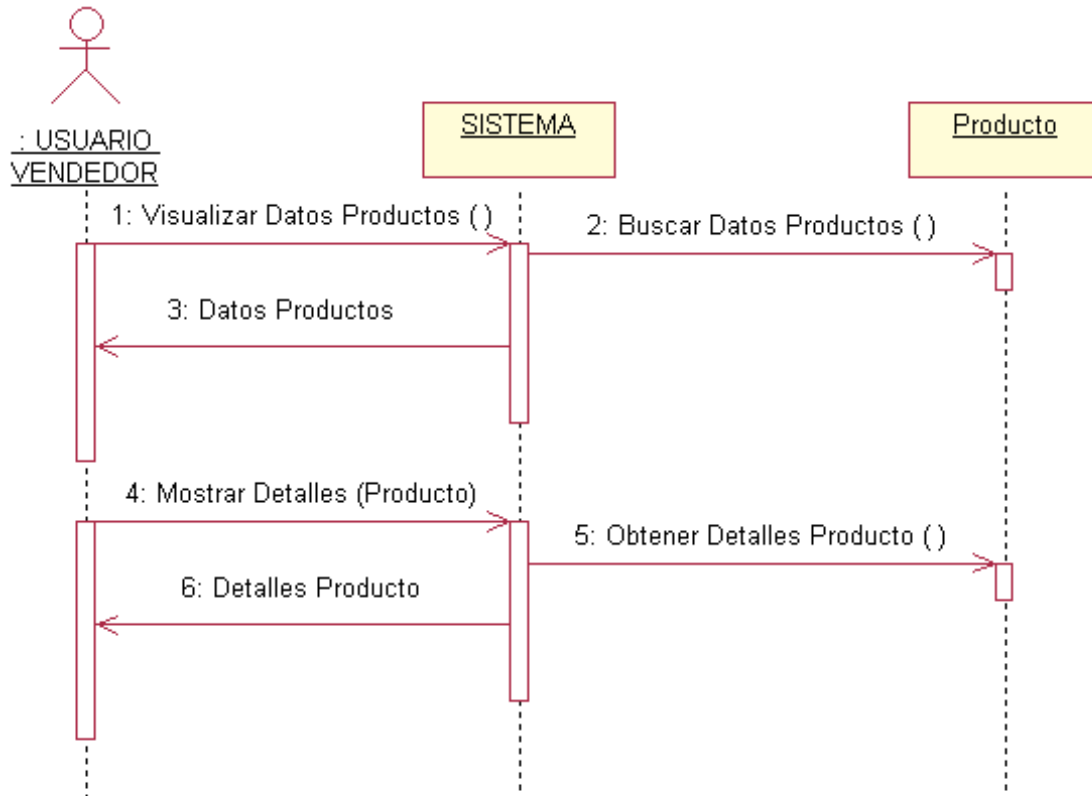


Figura 11.2. Escenario “Visualización de datos de producto de vendedor”.

Escenario 11.3 "Modificación productos".

- **Caso de uso:** 11. "Gestionar productos vendedor".
- **Escenario:** 11.3. Modificación productos vendedor.
- **Objetivo:** Permitir al usuario vendedor modificar los datos relativos a un determinado producto.
- **Descripción:** El usuario vendedor solicita al sistema realizar la modificación de los datos de un producto y el sistema muestra los datos actuales del producto y solicita los nuevos. El usuario introduce los nuevos datos y el sistema los recoge, previa validación de los mismos.

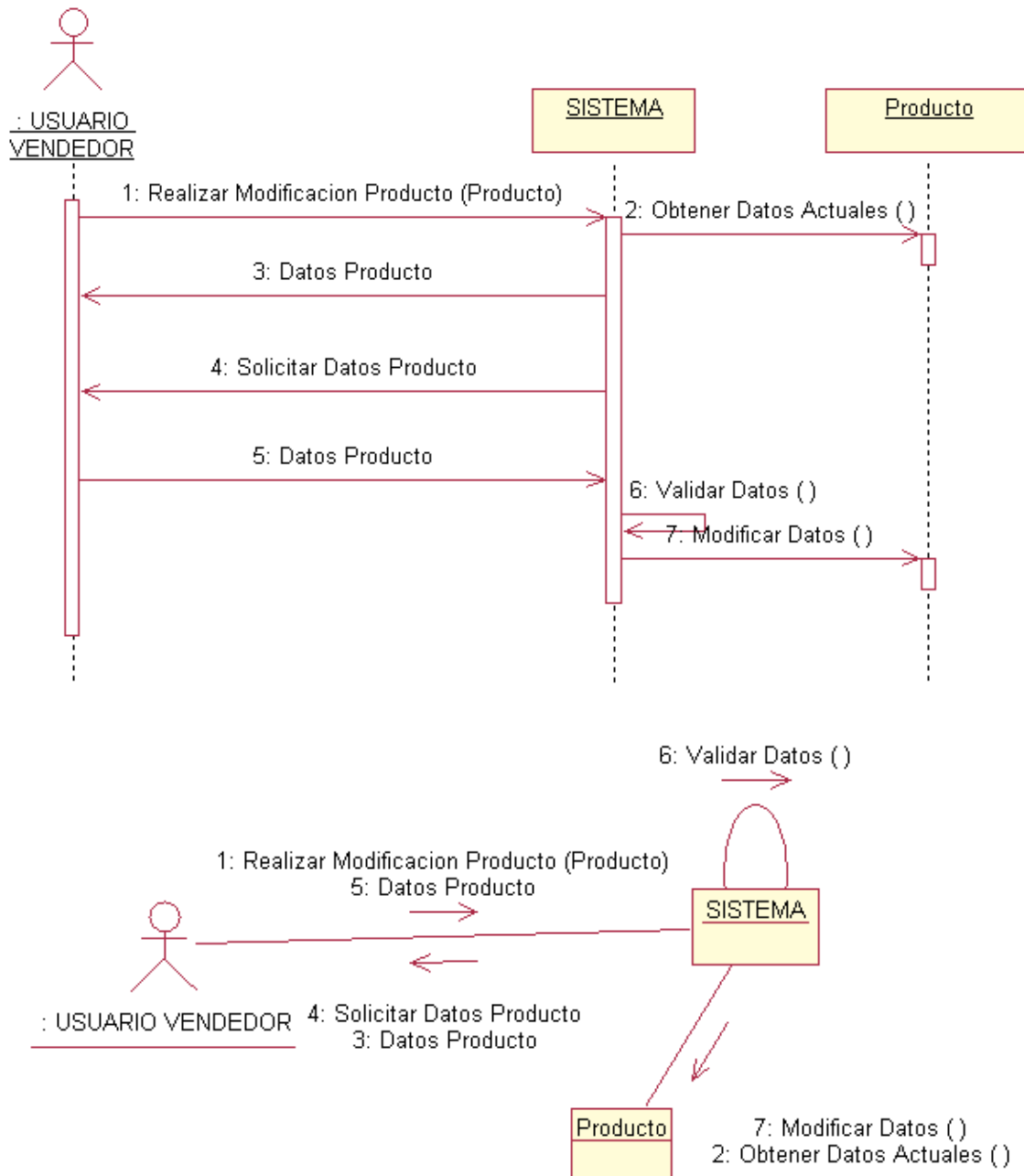


Figura 11.3. Escenario "Modificación de datos de producto de vendedor".

Escenario 11.4 “Baja productos”.

- **Caso de uso:** 11. “Gestionar productos vendedor”.
- **Escenario:** 11.2. Baja productos vendedor.
- **Objetivo:** Permitir al usuario vendedor dar de baja un producto.
- **Descripción:** El usuario vendedor realiza la petición de baja al sistema para dar de baja un producto y el sistema da de baja el producto.

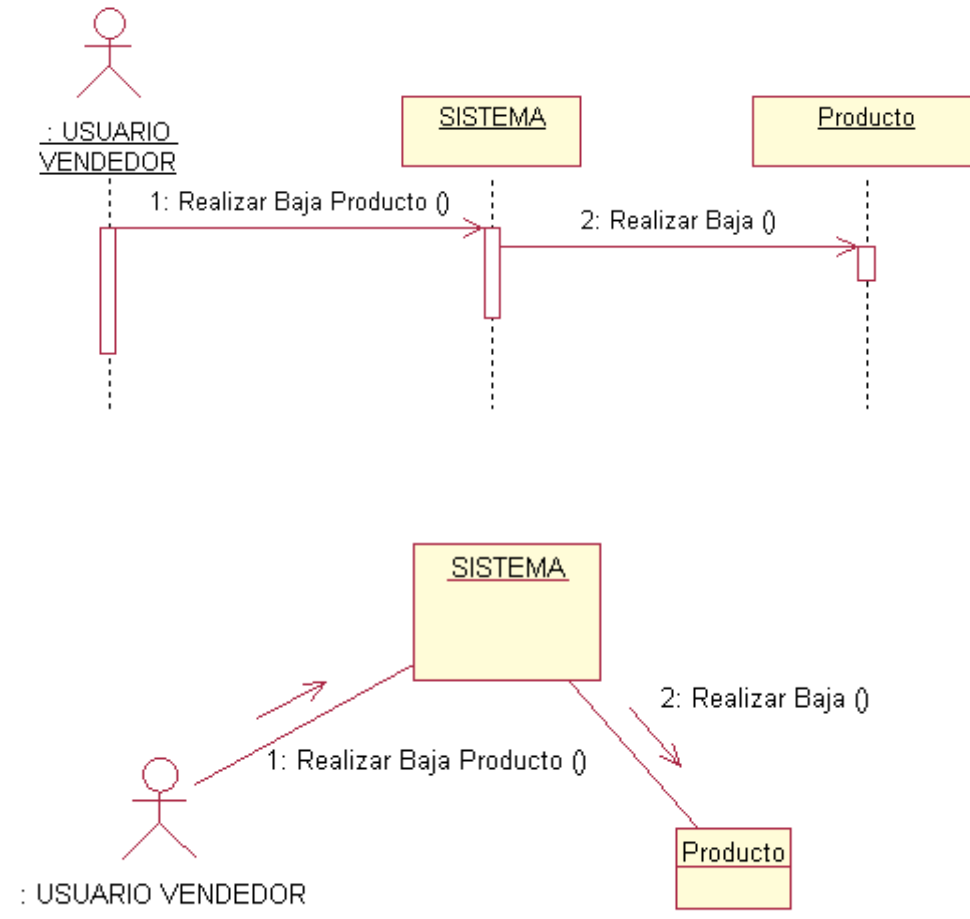


Figura 11.1. Escenario “Baja de producto de vendedor”.

Escenario 12.1 "Alta productos".

- **Caso de uso:** 12. "Gestionar productos".
- **Escenario:** 12.1. Alta productos.
- **Objetivo:** Permitir al usuario administrador dar de alta un producto.
- **Descripción:** El usuario solicita al sistema realizar el alta de un nuevo producto, el sistema pide los datos referentes al producto y el usuario introduce los datos requeridos. El sistema recoge los datos, los valida y da de alta el producto en el sistema.

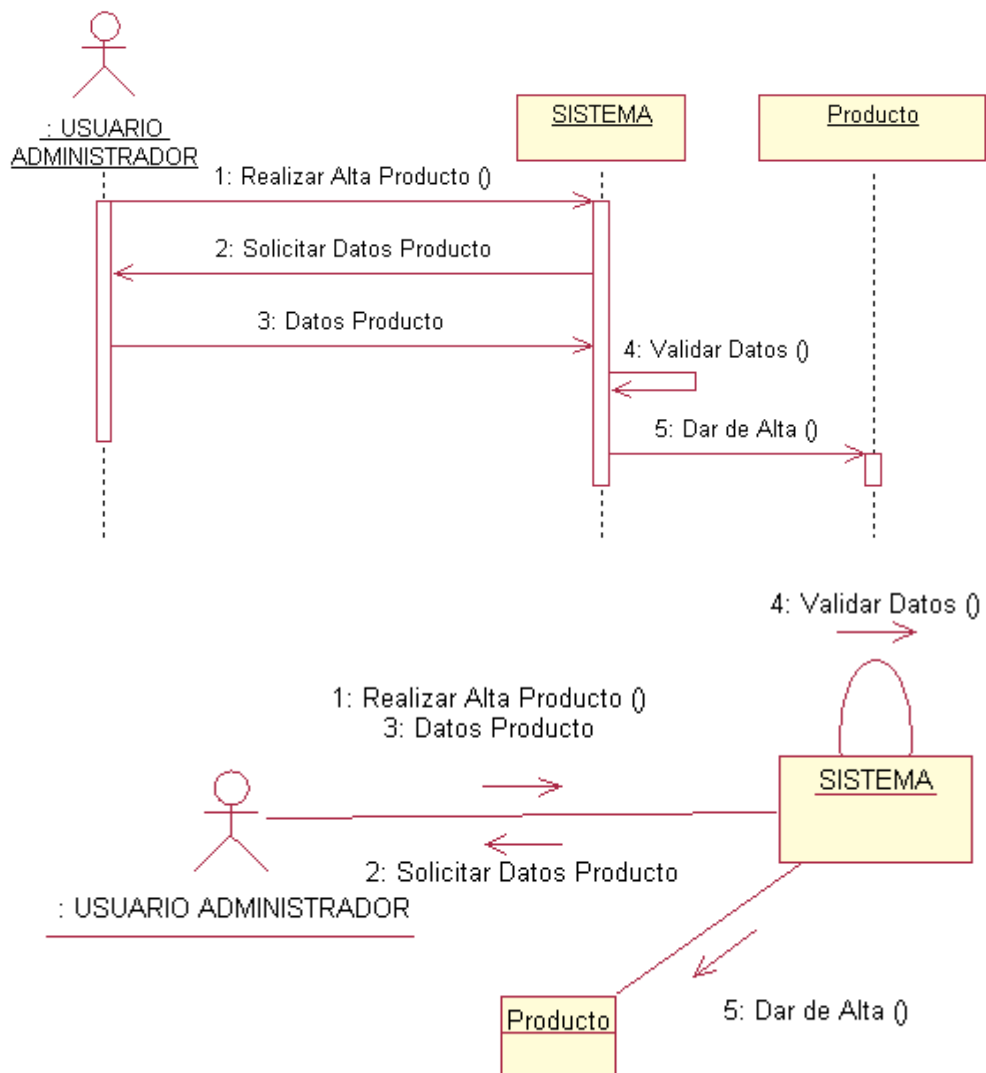


Figura 12.1. Escenario "Alta de producto".

Escenario 12.2 “Visualización de datos de producto”.

- **Caso de uso:** 12. “Gestionar productos”.
- **Escenario:** 12.2. Visualización de datos de producto.
- **Objetivo:** Permitir al usuario administrador visualizar los datos de un determinado producto.
- **Descripción:** El usuario administrador solicita al sistema la visualización de datos de productos y el sistema muestra los resultados, dando la posibilidad al usuario de mostrar el detalle de los datos para un determinado producto.

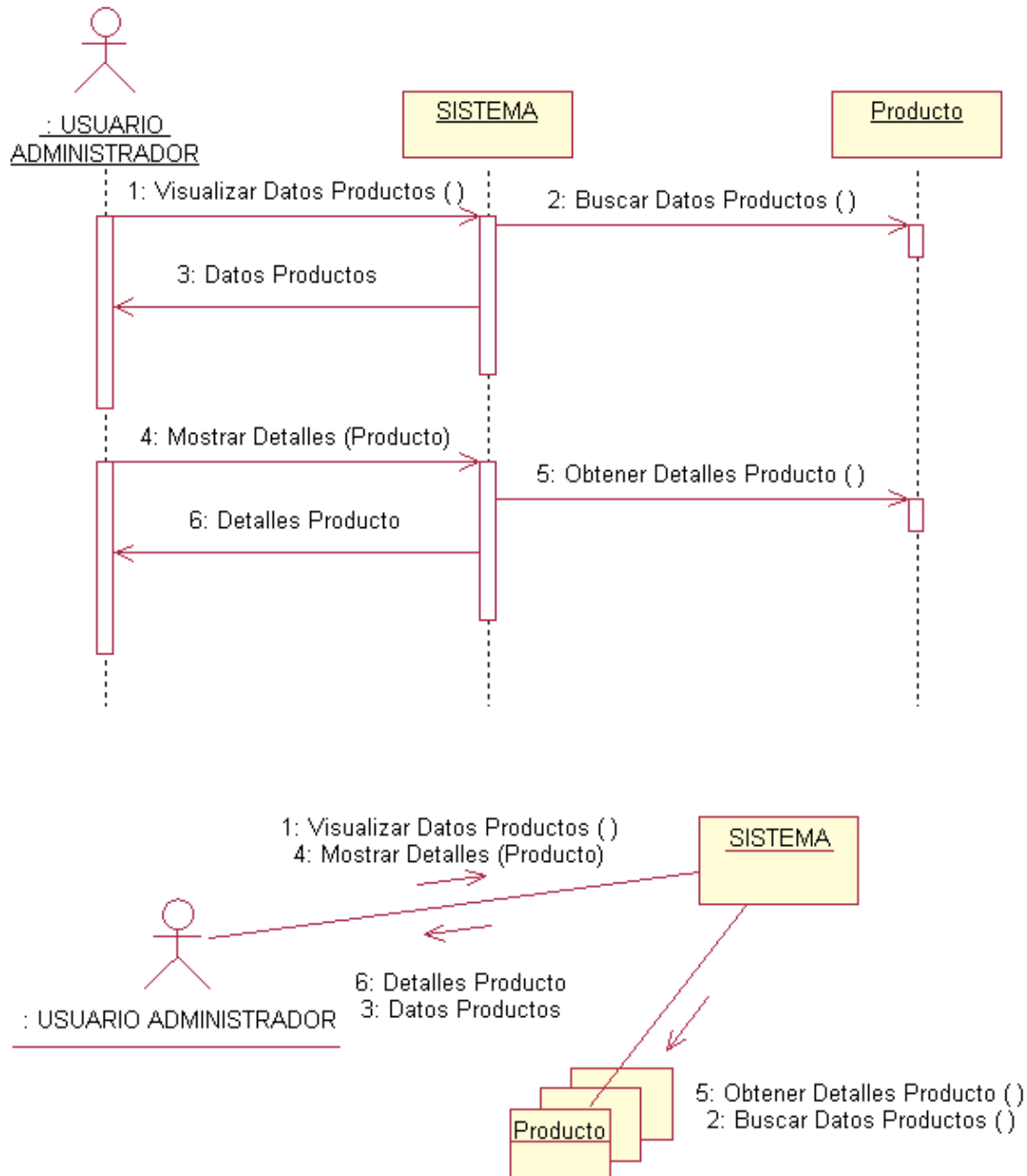


Figura 12.2. Escenario “Visualización de datos de producto”.

Escenario 12.3 "Modificación productos".

- **Caso de uso:** 12. "Gestionar productos".
- **Escenario:** 12.3. Modificación productos.
- **Objetivo:** Permitir al usuario administrador modificar los datos relativos a un determinado producto.
- **Descripción:** El usuario solicita al sistema realizar la modificación de los datos de un producto y el sistema muestra los datos actuales del producto y solicita los nuevos. El usuario introduce los nuevos datos y el sistema los recoge, previa validación de los mismos.

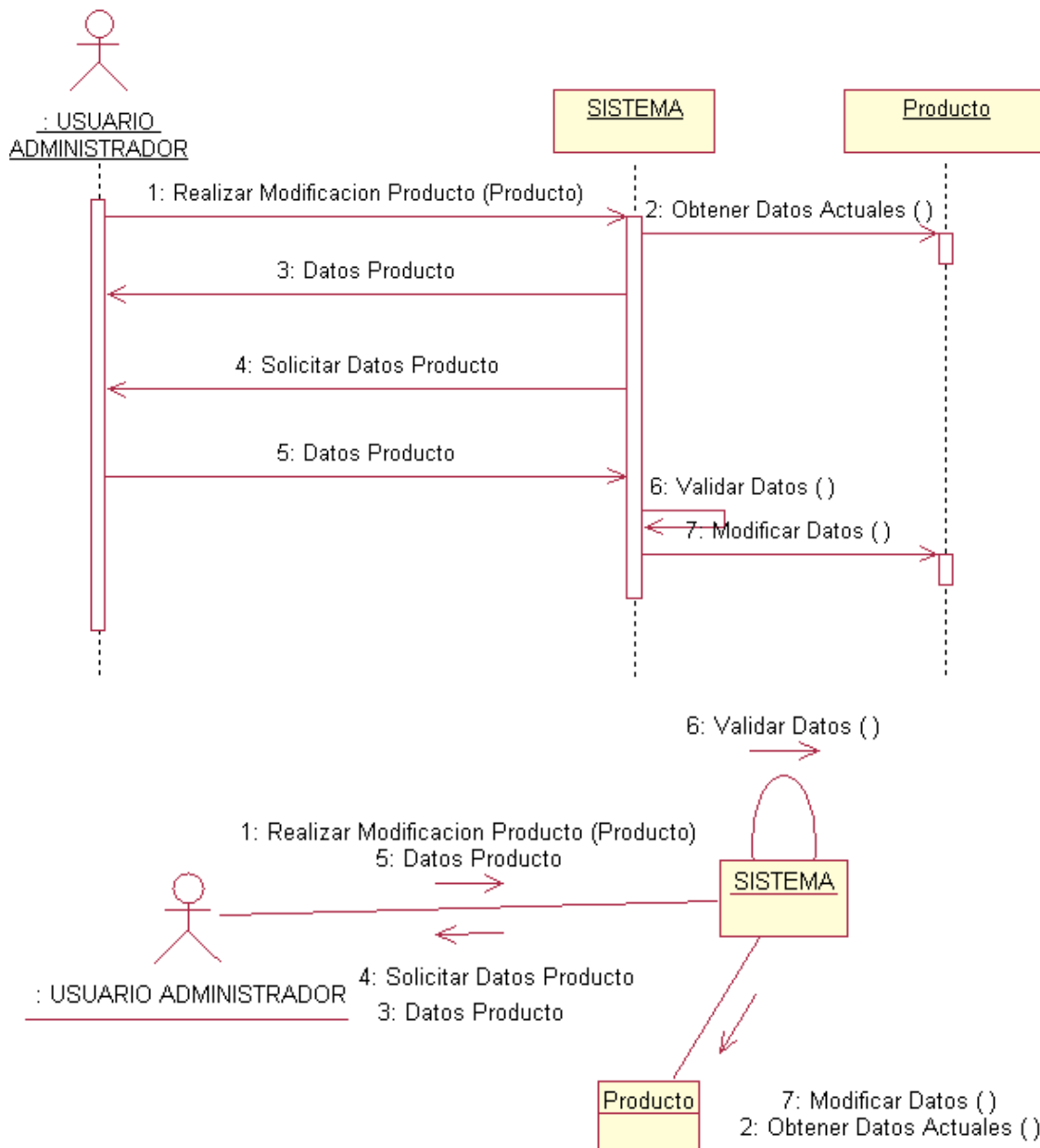


Figura 12.3. Escenario "Modificación de datos de producto".

Escenario 12.4 “Baja productos”.

- **Caso de uso:** 12. “Gestionar productos”.
- **Escenario:** 12.4. Baja productos.
- **Objetivo:** Permitir al usuario administrador dar de baja un producto.
- **Descripción:** El usuario realiza la petición de baja al sistema para dar de baja un producto y el sistema da de baja el producto.

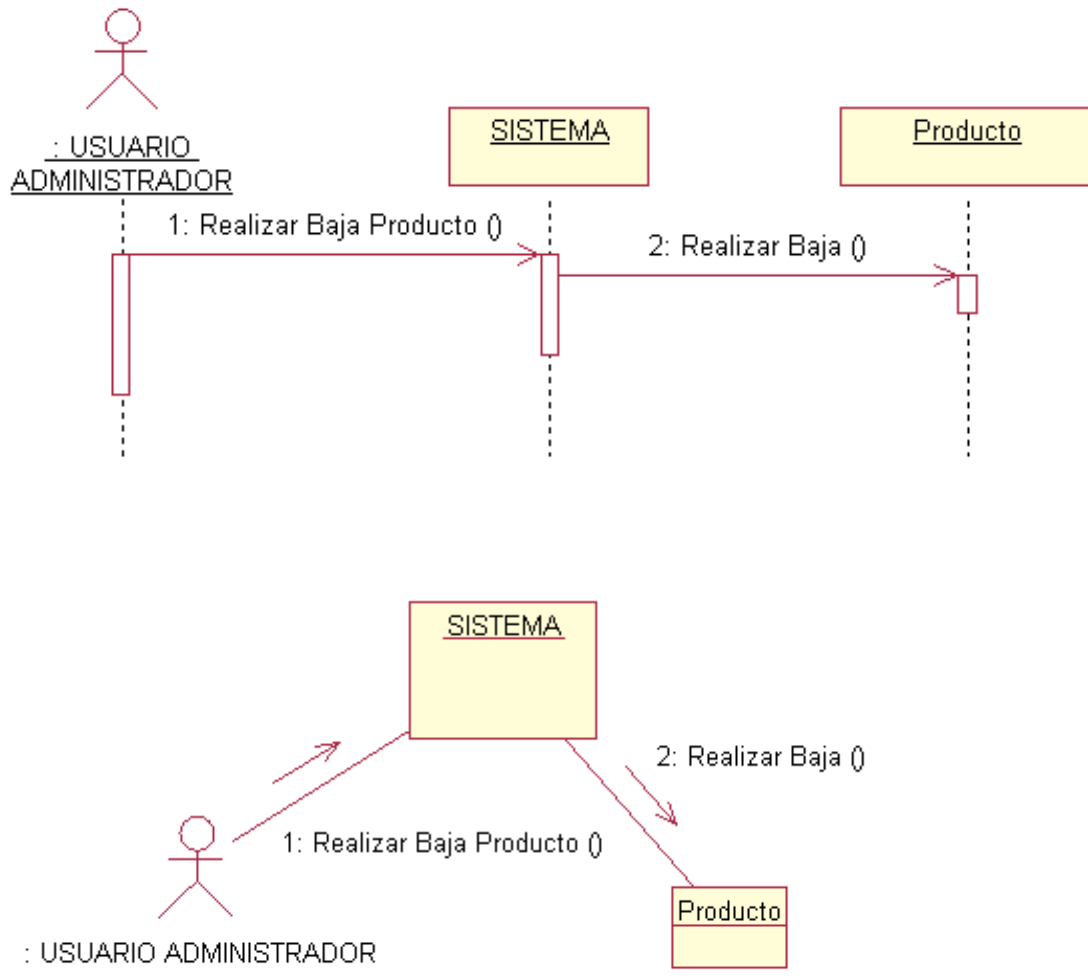


Figura 12.4. Escenario “Baja de producto”.

Escenario 13.1 "Alta de categoría".

- **Caso de uso:** 13. "Gestionar categorías productos".
- **Escenario:** 13.1. Alta de categoría.
- **Objetivo:** Permitir al usuario administrador dar de alta en el sistema a una categoría.
- **Descripción:** El usuario solicita al sistema realizar el alta una categoría, el sistema pide los datos referentes a la categoría y el usuario introduce los datos requeridos. El sistema recoge los datos, los valida y da de alta la categoría en el sistema.

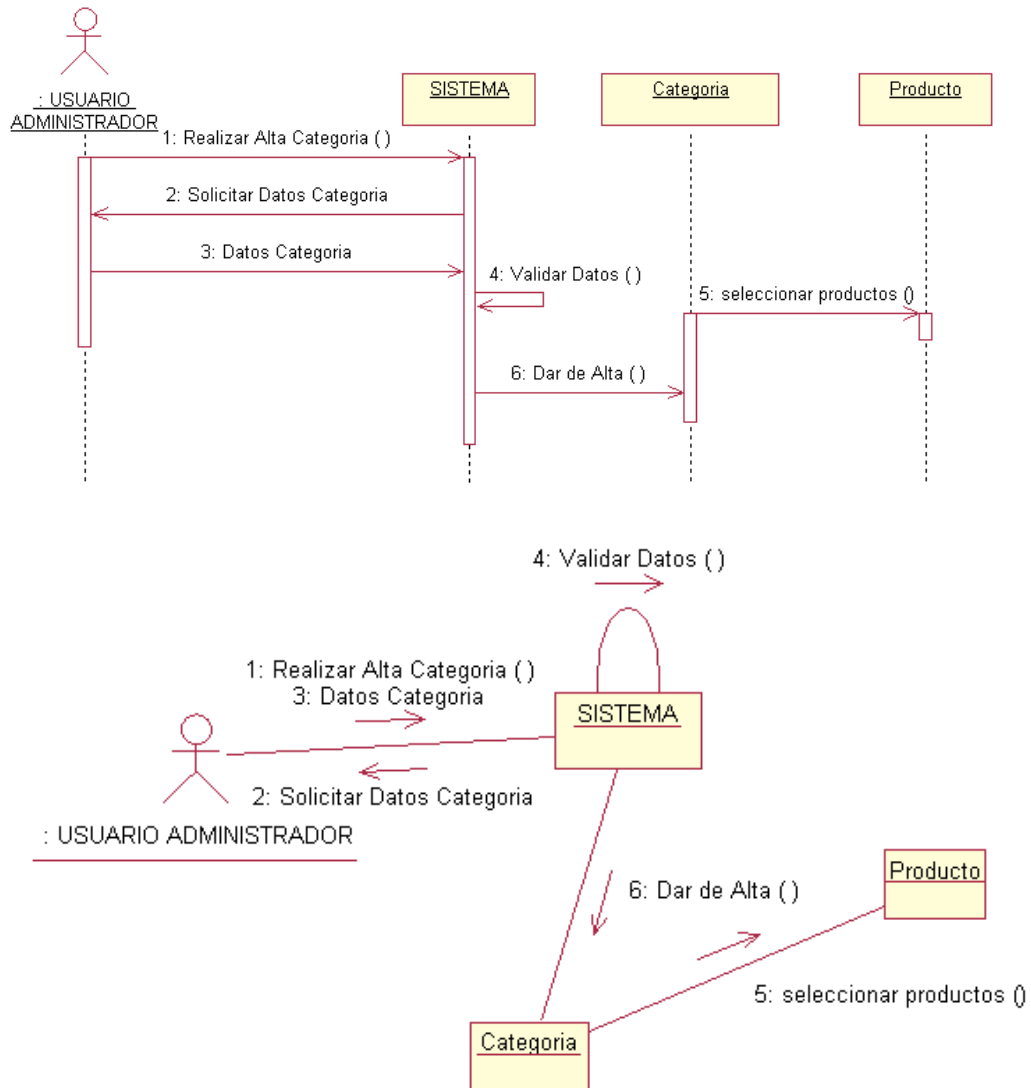
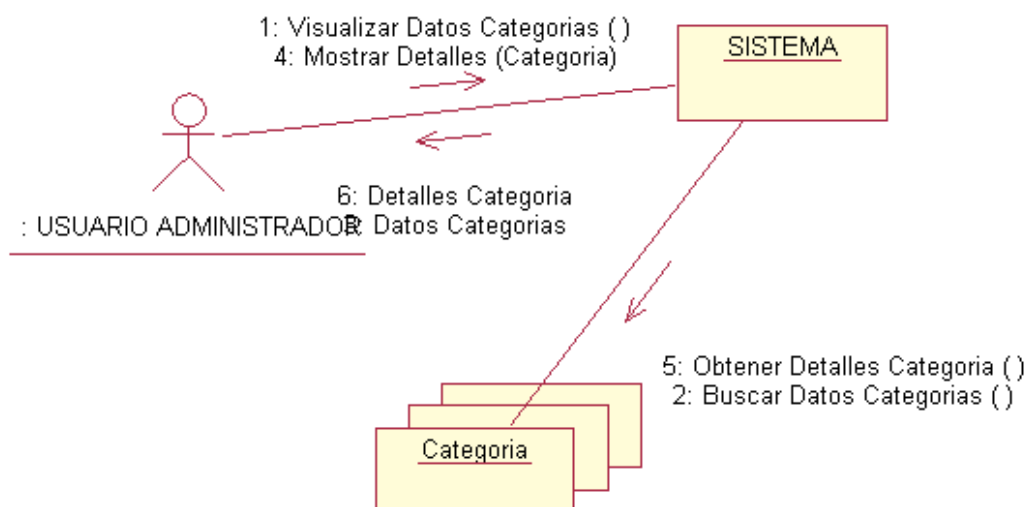
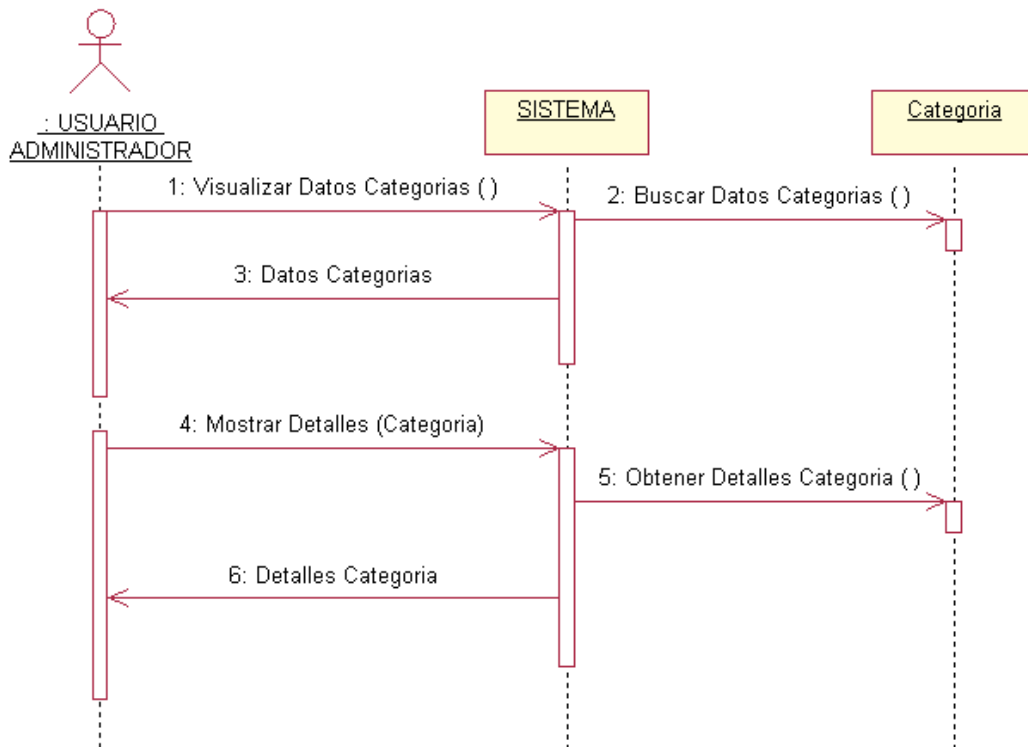


Figura 13.1. Escenario "Alta de categoría".

Escenario 13.2 “Visualización de categoría”.

- **Caso de uso:** 13. “Gestionar categorías productos”.
- **Escenario:** 13.2. Visualización de categoría.
- **Objetivo:** Permitir al usuario administrador la visualización de los datos propios de una determinada categoría.
- **Descripción:** El usuario administrador solicita al sistema la visualización de datos de categorías y el sistema muestra los resultados, dando la posibilidad al usuario de mostrar el detalle de los datos para una determinada categoría.



Escenario 13.3 "Modificación de datos de categoría".

- **Caso de uso:** 13. "Gestionar categorías productos".
- **Escenario:** 13.3. Modificación de datos de categoría.
- **Objetivo:** Permitir al usuario administrador modificar los datos de una categoría.
- **Descripción:** El usuario solicita al sistema realizar la modificación de los datos de una categoría y el sistema muestra los datos actuales de la categoría y solicita los nuevos. El usuario introduce los nuevos datos y el sistema los recoge, previa validación de los mismos.

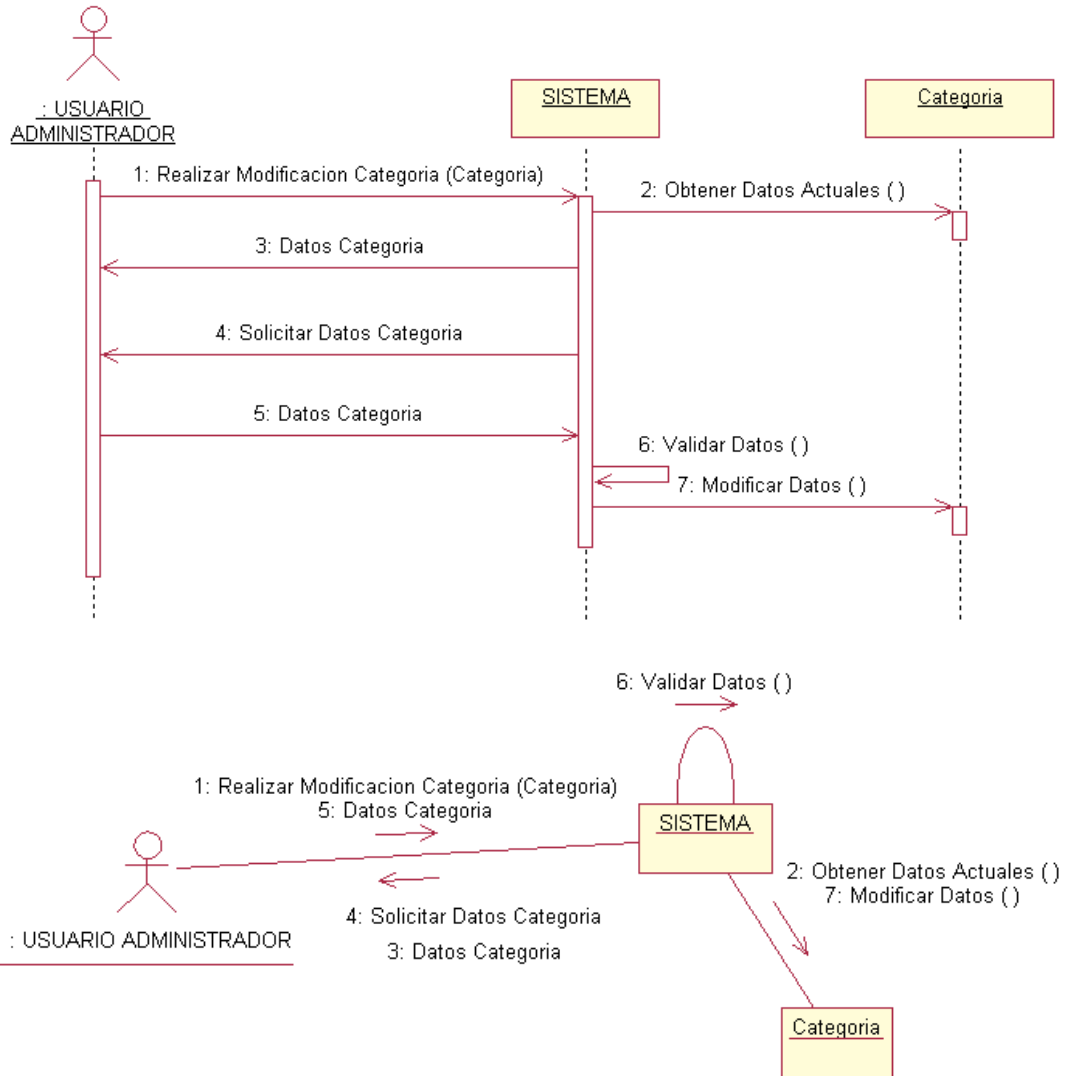


Figura 13.2. Escenario "Visualización de categoría".

Escenario 13.4 “Baja de categoría”.

- **Caso de uso:** 13. “Gestionar categorías productos”.
- **Escenario:** 13.4. Baja de categoría.
- **Objetivo:** Permitir al usuario administrador dar de baja una categoría en el categoría.
- **Descripción:** El usuario realiza la petición de baja al sistema para dar de baja una categoría y el sistema da de baja la categoría.

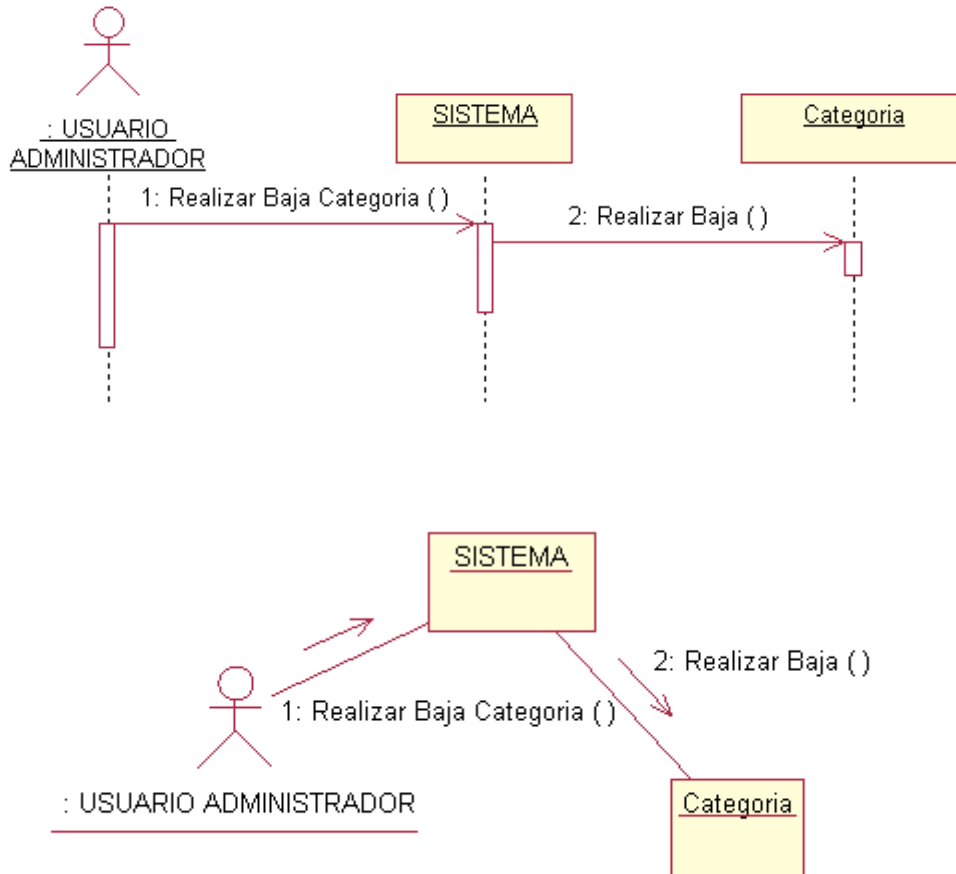


Figura 13.3. Escenario “Baja de categoría”.

Escenario 14.1 “Visualización datos administrador”.

- **Caso de uso:** 14. “Gestionar datos administrador”.
- **Escenario:** 14.1. Visualización de datos administrador.
- **Objetivo:** Permitir al usuario administrador visualizar los datos personales que constan en el sistema acerca de él.
- **Descripción:** El usuario solicita al sistema realizar una consulta sobre sus datos. El sistema solicita las preferencias de visualización y los posibles filtros de información. El usuario establece unas determinadas preferencias y filtrados de información y el sistema recoge la petición y muestra la información requerida.

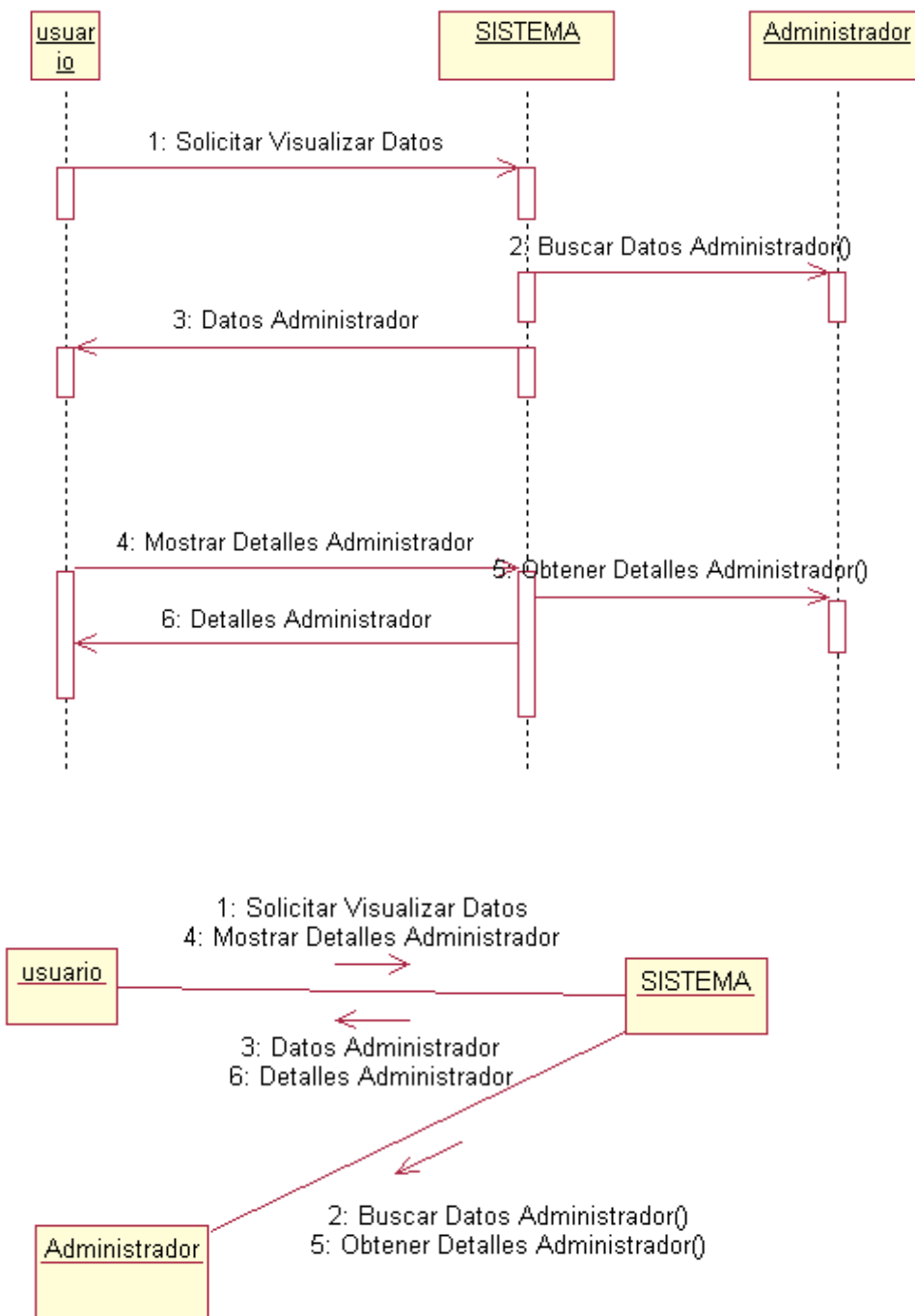


Figura 14.1. Escenario “Visualización de datos de administrador”.

Escenario 14.2 “Modificación de datos administrador”.

- **Caso de uso:** 14. “Gestionar datos administrador”.
- **Escenario:** 14.2. Modificación de datos administrador.
- **Objetivo:** Permitir al usuario administrador modificar sus datos personales.

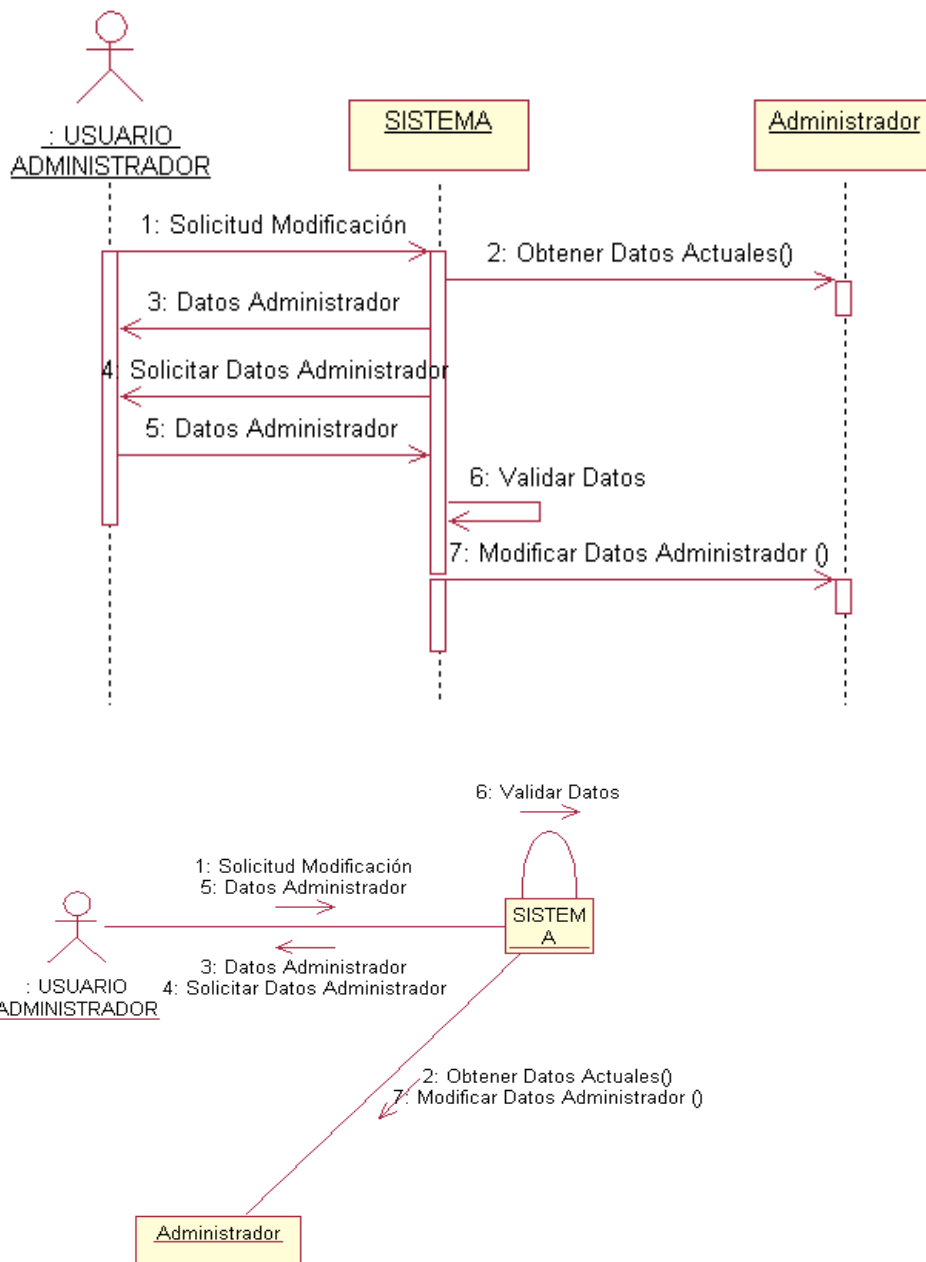
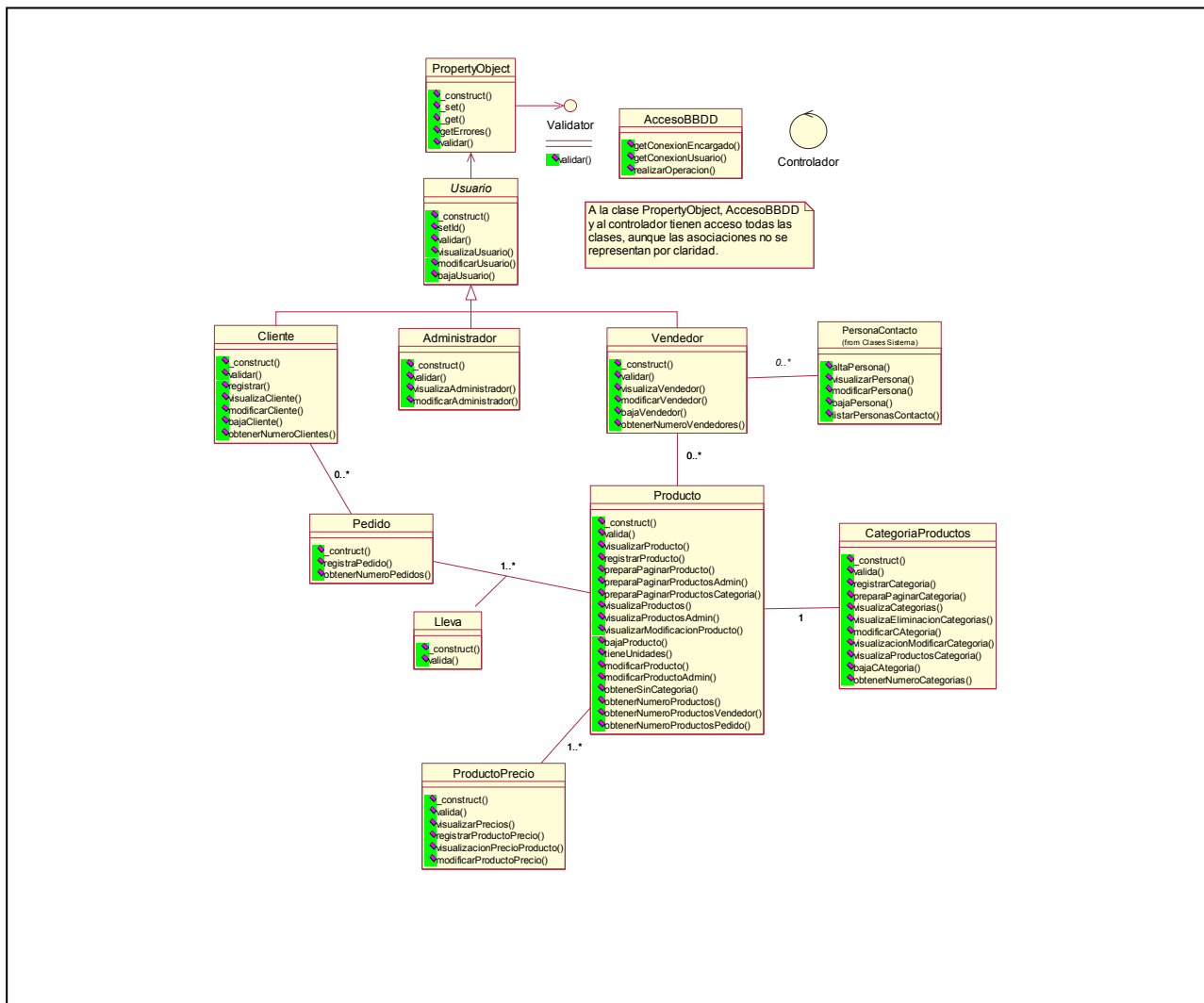


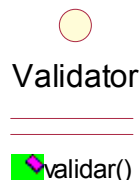
Figura 14.2. Escenario “Modificación de datos de administrador”

DIAGRAMA DE CLASES FINAL.

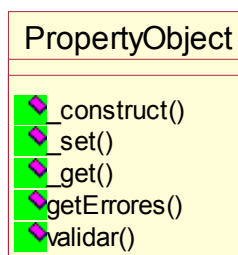


CLASES DEL SISTEMA

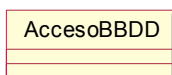
Interface Validador: Esta interface define la funcionalidad de validación de todas las clases de la aplicación con el método validar() que implementará cada clase que la necesite.



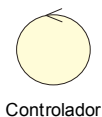
Clase PropertyObject: Esta clase es abstracta ya que no proporciona una implementación para el método de validación. De esta clase Heredan todas la demás de la aplicación, pues en ella se establecen los métodos *set()* y *get()* de atributos de cada clase de la aplicación, aparte de la validación de los datos correspondientes a los atributos y el almacenamiento de los posibles errores que se produzcan como consecuencia de una mala edición o inserción de datos en el formulario que corresponda.



Clase Acceso a base de datos: esta clase nace ante la necesidad de diversos métodos que se puedan ejecutar desde cualquier clase, pero que no llegan a ser propios de ninguna de ellas. Es por ello que se denomina una clase puramente funcional y que por lo tanto, establece métodos de clase.

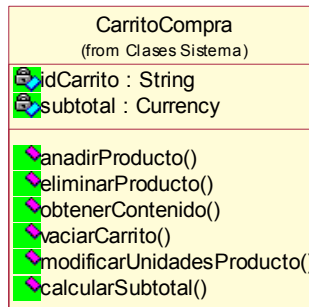


Clase Controlador: esta clase que tiene acceso a todas las clases del diagrama, representa al elemento encargado de recibir los eventos generados en la vista de la aplicación y generar las relaciones con el modelo que respondan a dicho evento, actualizando posteriormente la vista con los resultados obtenidos. Las asociaciones no se representan para no entorpecer el seguimiento del diagrama de clases.

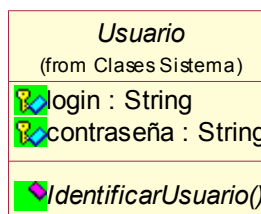


Controlador

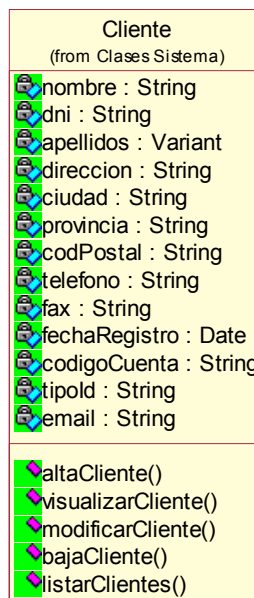
Clase Carrito de la compra: esta clase es la que contendrá una relación de todos los productos que se van seleccionando a la hora de llevar a cabo la compra en la página web.



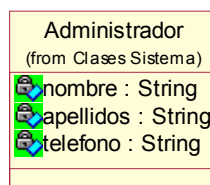
Clase Usuario: se trata de la clase padre de la jerarquía de usuarios de la aplicación, y es una clase abstracta, lo que implica que no se generarán clases de este tipo durante el desarrollo normal del programa.



Clase Cliente: esta clase representa al comprador de productos. Hereda de la clase usuario.



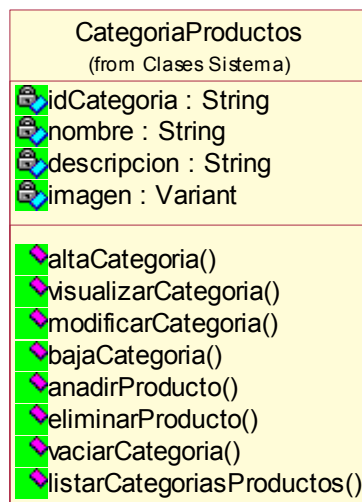
Clase Administrador: esta clase representa al administrador de la aplicación. Hereda de la clase usuario.



Clase Vendedor: esta clase representa a cada uno de los vendedores dentro de la aplicación. Hereda de la clase usuario.



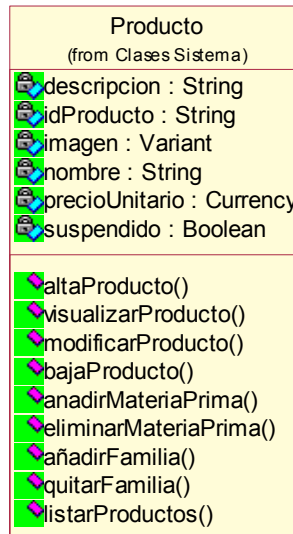
Clase Categoría de Productos: representa grupos de productos que se ordenan siguiendo una lógica establecida por el administrador.



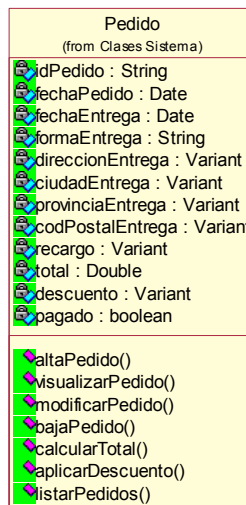
Clase Lleva: esta clase es utilizada para representar el número de unidades de un determinado producto que van en un pedido.



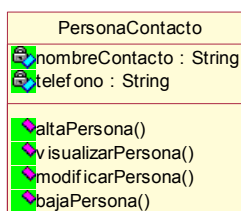
Clase producto: es la clase encargada de representar a cada uno de los productos que son creados en la tienda virtual.



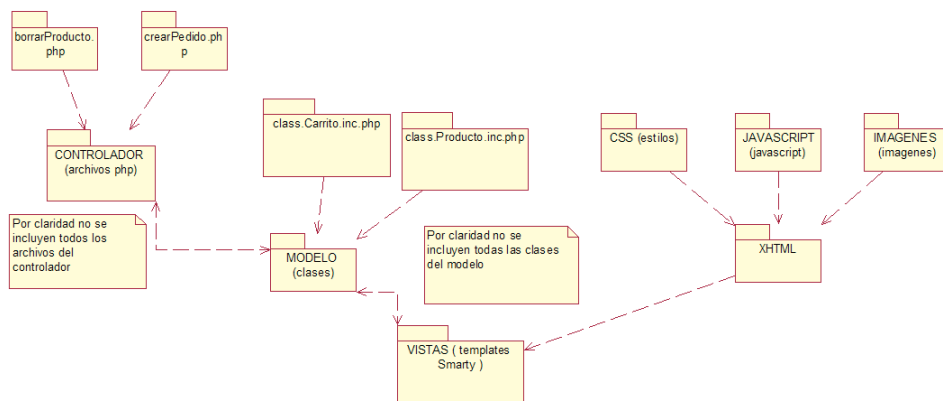
Clase pedido: representa la materialización, en compra, de un carrito que contiene productos y al que se le han añadido los datos del comprador en cuestión.



Clase persona de contacto: esta clase es utilizada para almacenar información acerca del representante o representantes de un determinado vendedor que ha sido dado de alta en el sistema.



DISEÑO DETALLADO: DIAGRAMA DE PAQUETES.



En el presente apartado se muestra el diagrama de paquetes donde se observan las relaciones de los componentes de la aplicación que al final se convierten en directorios. En los mismos quedan representados el Modelo, en el directorio *Clases*, la Vista, que es el directorio de plantillas de Smarty, que a su vez utiliza los directorios de imágenes, CSS y Javascript para junto con el XHTML y el código propio de la plantillas conformar la vista, y el Controlador, todos los archivos que recogen las acciones propias de los menús como son `listarProductos.php`, por ejemplo, y desencadenan las acciones que permiten acceder a los modelos, a la base de datos y actualizar la vista correspondiente.

ANÁLISIS Y DISEÑO DE LA BASE DE DATOS.

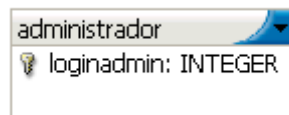
Por considerarse una parte fundamental de la aplicación, el análisis y diseño de la base de datos se tratarán a continuación en un apartado independiente de la sección 2.

El diagrama Entidad-Relación no forma parte de la metodología orientada a objetos utilizada, pero es necesario para diseñar la base de datos relacional del sistema.

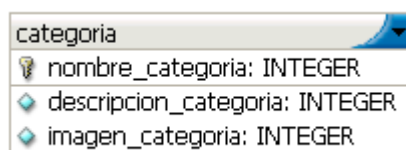
A través del Modelo Entidad-Relación se muestra el modelado de la base de datos a nivel conceptual. Mediante este diagrama se representa la información del sistema a alto nivel, es decir, se muestran los datos que son precisos almacenar, pero no la forma en que serán almacenados. Además se presentarán de forma individual las entidades que forman el diagrama y los atributos que incorporan. Todas las entidades y relaciones que aparecen se corresponderán con clases ya citadas en el diagrama de clases, debido a que ambos diagramas son equivalentes y proporcionan casi la misma información.

El diagrama Entidad-Relación que se muestra se presenta especificado hasta el nivel de entidad, para favorecer la nitidez del mismo.





Entidad ADMINISTRADOR:






Entidad CATEGORIA:















Entidad CLIENTE:

| cliente | |
|---|-----------------------|
|  | logincliente: INTEGER |
|  | fecha: INTEGER |
|  | cuenta: INTEGER |
|  | tipoCliente: INTEGER |






Relación LINEA_PEDIDO:

| linea_pedido | |
|---|----------------------|
|  | idproducto: INTEGER |
|  | idpedido: INTEGER |
|  | numunidades: INTEGER |










Entidad PEDIDO:

| pedido | |
|---|----------------------------|
|  | idpedido: INTEGER |
|  | fecha_pedido: INTEGER |
|  | fecha_entrega: INTEGER |
|  | forma_entrega: INTEGER |
|  | direccion_entrega: INTEGER |
|  | ciudad_entrega: INTEGER |
|  | provincia_entrega: INTEGER |
|  | cp__entrega: INTEGER |
|  | total: INTEGER |
|  | descuento: INTEGER |
|  | pagado: INTEGER |
|  | cliente: INTEGER |



Entidad PERSONA CONTACTO:

| persona_contacto | |
|---|-----------------------------|
|  | login_ven: INTEGER |
|  | telefono: INTEGER |
|  | nombre_contacto: INTEGER |
|  | apellidos_contacto: INTEGER |
|  | email_contacto: INTEGER |













Entidad PRODUCTO:

| producto | |
|---|--|
|  idproducto: INTEGER | |
|  nombre_producto: INTEGER | |
|  descripcion_producto: INTEGER | |
|  imagen_producto: INTEGER | |
|  suspendido: INTEGER | |
|  unidades_disponibles: INTEGER | |
|  nombre_cat: INTEGER | |
|  fabricante: INTEGER | |
|  fecha_suministro: INTEGER | |



Entidad PRODUCTO_PRECIO:

| producto_precio | |
|---|--|
|  idprod: INTEGER | |
|  precio: INTEGER | |

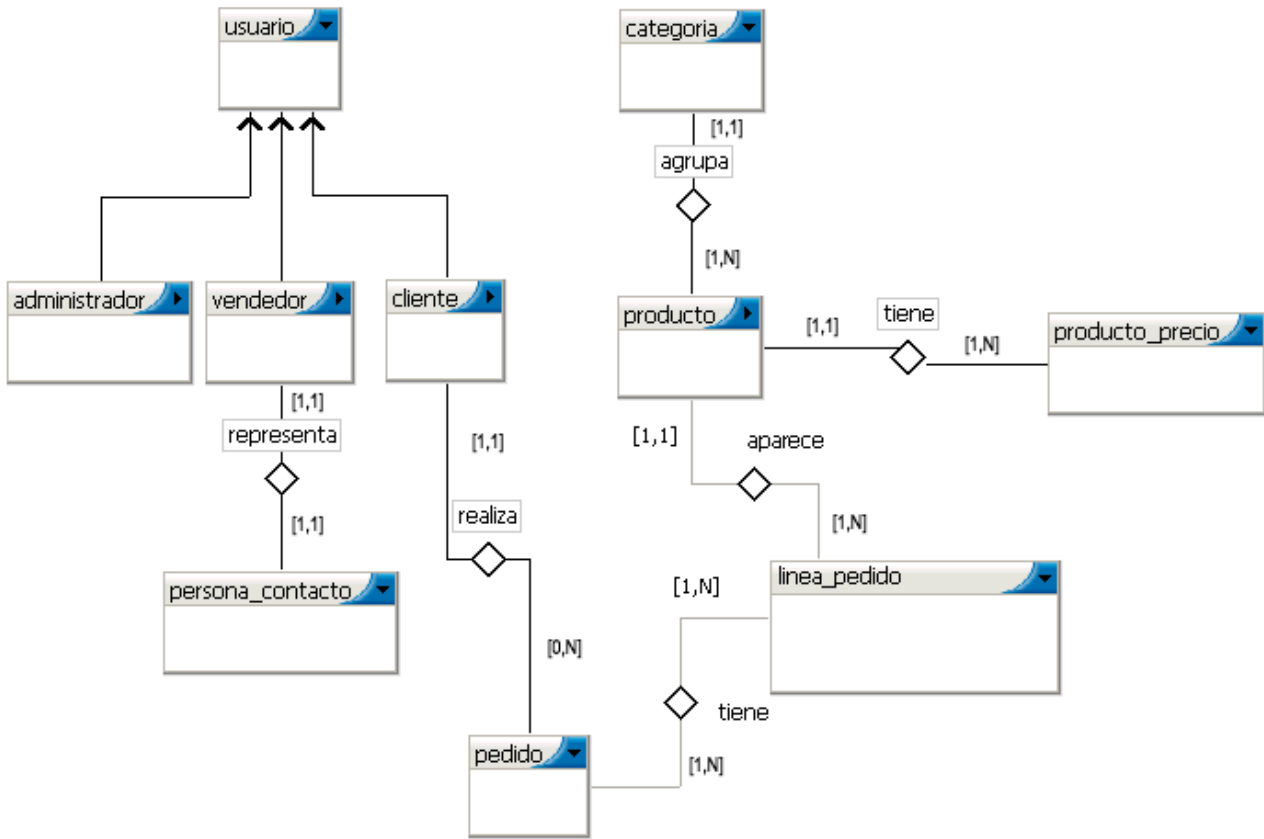
Entidad USUARIO:

| usuario | |
|---|--|
|  login: INTEGER | |
|  password_2: INTEGER | |
|  tipoUsuario: INTEGER | |
|  nombre: INTEGER | |
|  apellidos: INTEGER | |
|  dniOcif: INTEGER | |
|  email: INTEGER | |
|  direccion: INTEGER | |
|  ciudad: INTEGER | |
|  provincia: INTEGER | |
|  codigoPostal: INTEGER | |
|  telefono: INTEGER | |

Entidad VENDEDOR:

| vendedor | |
|--|--|
|  loginvendedor: INTEGER | |
|  fecha_registro_vendedor: INTEGER | |

Modelo Entidad-Relación de la aplicación.



PASO A TABLAS DEL MODELO E-R.

En este apartado de la sección dedicada a la base de datos se presenta el paso al Modelo Relacional o paso a tablas del modelo Entidad-Relación. Posteriormente se mostrará el diccionario de datos correspondiente a la base de datos del sistema.

ADMINISTRADOR

| Campo | Tipo | Nulo | Predeterminado | Enlaces a |
|-------------------|-------------|------|----------------|----------------|
| <i>loginadmin</i> | varchar(15) | No | | usuario->login |

CATEGORÍA

| Campo | Tipo | Nulo | Predeterminado | Enlaces a |
|-------------------------|-------------|------|----------------|-----------|
| <i>nombre_categoria</i> | varchar(30) | No | | |
| descripcion_categoria | text | No | | |
| imagen_categoria | varchar(80) | No | | |

CLIENTE

| Campo | Tipo | Nulo | Predeterminado | Enlaces a |
|---------------------|-------------------------------|------|----------------|----------------|
| <i>logincliente</i> | varchar(15) | No | | usuario->login |
| fecha | date | No | | |
| cuenta | varchar(20) | No | | |
| tipoCliente | enum('particular', 'empresa') | No | | |

LINEA_PEDIDO

| Campo | Tipo | Nulo | Predeterminado | Enlaces a |
|-------------------|-------------|------|----------------|--------------------------|
| <i>idproducto</i> | varchar(15) | No | | producto ->idproducto |
| <i>idpedido</i> | int(10) | No | | pedido->idpedido |
| numunidades | varchar(3) | No | | |

PEDIDO

| Campo | Tipo | Nulo | Predeterminado | Enlaces a |
|-------------------|------------------------------|------|----------------|-----------------------|
| <i>idpedido</i> | int(10) | No | | |
| fecha_pedido | date | No | | |
| fecha_entrega | date | No | | |
| forma_entrega | enum('domicilio', 'recoger') | No | domicilio | |
| direccion_entrega | varchar(50) | Sí | NULL | |
| ciudad_entrega | varchar(20) | Sí | NULL | |
| provincia_entrega | varchar(20) | Sí | NULL | |
| cp_entrega | varchar(5) | Sí | NULL | |
| total | varchar(10) | No | | |
| descuento | varchar(5) | No | | |
| pagado | enum('pagado', 'a deber') | Sí | NULL | |
| cliente | varchar(15) | Sí | NULL | cliente->logincliente |

PERSONA CONTACTO

| Campo | Tipo | Nulo | Predeterminado | Enlaces a |
|--------------------|-------------|------|----------------|-----------------------------|
| <i>login_ven</i> | varchar(15) | No | | vendedor ->loginvendedor |
| <i>telefono</i> | varchar(9) | No | | |
| nombre_contacto | varchar(15) | No | | |
| apellidos_contacto | varchar(30) | No | | |
| email_contacto | varchar(30) | No | | |

PRODUCTO

| Campo | Tipo | Nulo | Predeterminado | Enlaces a |
|----------------------|--------------|------|----------------|---------------------------------|
| <i>idproducto</i> | varchar(15) | No | | |
| nombre_producto | varchar(20) | No | | |
| descripcion_producto | text | No | | |
| imagen_producto | varchar(150) | No | | |
| suspendido | tinyint(1) | No | 0 | |
| unidades_disponibles | varchar(2) | No | | |
| nombre_cat | varchar(40) | Sí | NULL | Categoría ->nombre_categoria |
| fabricante | varchar(15) | Sí | NULL | vendedor ->loginvendedor |
| fecha_suministro | date | No | | |

PRODUCTO_PRECIO

| Campo | Tipo | Nulo | Predeterminado | Enlaces a |
|---------------|-------------|------|----------------|--------------------------|
| <i>idprod</i> | varchar(15) | No | | producto ->idproducto |
| <i>precio</i> | varchar(8) | No | | |

USUARIO

| Campo | Tipo | Nulo | Predeterminado | Enlaces a |
|--------------|--|------|----------------|-----------|
| <i>login</i> | varchar(15) | No | | |
| password | varchar(15) | No | | |
| tipoUsuario | enum('administrador', 'vendedor', 'cliente') | No | | |
| nombre | varchar(20) | No | | |
| apellidos | varchar(40) | Sí | NULL | |
| dniOcif | varchar(10) | No | | |
| email | varchar(40) | No | | |
| direccion | varchar(30) | No | | |
| ciudad | varchar(20) | No | | |
| provincia | varchar(20) | No | | |
| codigoPostal | varchar(5) | No | | |
| telefono | varchar(9) | No | | |

VENDEDOR

| Campo | Tipo | Nulo | Predeterminado | Enlaces a |
|-------------------------|-------------|------|----------------|----------------|
| <i>loginvendedor</i> | varchar(15) | No | | usuario->login |
| fecha_registro_vendedor | date | No | | |

DICCIONARIO DE DATOS.

ADMINISTRADOR

| Campo | Comentarios |
|-------------------|--|
| <i>loginadmin</i> | Código alfanumérico que describe la clave ajena c respecto al usuario y que describe la clave principal c administrador. |

CATEGORÍA

| Campo | Comentarios |
|-------------------------|--|
| <i>nombre_categoria</i> | Código alfanumérico que representa la clave primaria de la categoría de productos. |

SECCIÓN 2: MANUAL TÉCNICO

| | |
|-----------------------|---|
| descripcion_categoria | Texto con la descripción de la categoría. |
| imagen_categoria | Código alfanumérico que indica la imagen de la categoría. |

CLIENTE

| Campo | Comentarios |
|---------------------|--|
| logincliente | Código alfanumérico que describe la clave primaria del cliente. |
| fecha | Fecha de antigüedad del cliente. |
| cuenta | Número de código cuenta corriente. |
| tipoCliente | Tipo enumerado que describe el tipo de cliente (particular o empresa). |

LINEA PEDIDO

| Campo | Comentarios |
|-------------------|---|
| idproducto | Código alfanumérico que describe la clave ajena con respecto al producto y correpresenta la clave primaria de la tabla. |
| idpedido | Código numérico que describe la clave ajena con respecto al pedido y correpresenta la clave primaria de la tabla. |
| numunidades | Código numérico que indica los el número de unidades de un producto en un pedido. |

PEDIDO

| Campo | Comentarios |
|-------------------|--|
| idpedido | Código numérico que describe la clave primaria del pedido. |
| fecha_pedido | Código de tipo fecha que indica la fecha de realización de pedido. |
| fecha_entrega | Código de tipo fecha que indica la fecha aproximada de entrega del pedido. |
| forma_entrega | Atributo enumerado que indica la forma de entrega (domicilio o recoger) . |
| direccion_entrega | Código alfanumérico que indica la dirección de entrega. |
| ciudad_entrega | Código alfanumérico que indica la ciudad de entrega. |
| provincia_entrega | Código alfanumérico que indica la provincia de entrega. |
| cp_entrega | Código alfanumérico que indica el código postal del lugar de entrega. |
| total | Código numérico que indica el recargo que tiene el pedido. |
| descuento | Código numérico que indica el total del coste del pedido. |
| pagado | Código numérico que indica el descuento obtenido en el pedido. |
| cliente | Atributo enumerado que indica si el pedido está pagado o pendiente. |

PERSONA CONTACTO

| Campo | Comentarios |
|--------------------|--|
| login_ven | Código alfanumérico que describe la clave ajena con respecto al vendedor y correpresenta la clave principal de la persona. |
| telefono | Código numérico que describe la clave ajena con respecto al vendedor y correpresenta la clave principal de la persona. |
| nombre_contacto | Código alfanumérico que describe el nombre de la persona. |
| apellidos_contacto | Código alfanumérico que describe los apellidos de la persona. |
| email_contacto | Código alfanumérico que describe el email de la persona. |

PRODUCTO

| Campo | Comentarios |
|----------------------|---|
| idproducto | Código alfanumérico que describe la clave primaria del producto. |
| nombre_producto | Código alfanumérico que indica el nombre del producto |
| descripcion_producto | Breve descripción del producto. |
| imagen_producto | Imagen que representa el producto en cuestión. |
| suspendido | Código numérico que indica la suspensión del producto. |
| unidades_disponibles | Código numérico de las unidades disponibles del producto. |
| nombre_cat | Código alfanumérico que describe la clave ajena con respecto a la categoría. |
| fabricante | Código alfanumérico que describe la clave ajena con respecto al vendedor. |
| fecha_suministro | Código de tipo fecha que describe la fecha de suministro del producto por parte del vendedor. |

PRODUCTO_PRECIO

| Campo | Comentarios |
|---------------|---|
| idprod | Código alfanumérico que describe la clave ajena con respecto al producto y correpresenta la clave primaria de la tabla. |
| precio | Código alfanumérico que correpresenta la clave primaria de la tabla. |

USUARIO

| Campo | Comentarios |
|--------------|---|
| login | Código alfanumérico representa la clave primaria del usuario. |
| password | Código alfanumérico que indica la password del usuario. |
| tipoUsuario | Tipo enumerado que indica el tipo de usuario (administrador, cliente o vendedor). |
| nombre | Código alfanumérico que indica el nombre del usuario. |
| apellidos | Código alfanumérico que indica los apellidos del usuario. |
| dniOcif | Código alfanumérico que indica el dni o el cif del usuario. |
| email | Código alfanumérico que indica el email del usuario. |
| direccion | Código alfanumérico que indica la dirección del usuario. |
| ciudad | Código alfanumérico que indica la ciudad del usuario. |
| provincia | Código alfanumérico que indica la provincia del usuario. |
| códigoPostal | Código alfanumérico que indica el código postal del usuario. |
| telefono | Código alfanumérico que indica el telefono del usuario. |

VENDEDOR

| Campo | Comentarios |
|-------------------------|--|
| loginvendedor | Código alfanumérico que describe la clave primaria de la tabla vendedor. |
| fecha_registro_vendedor | Fecha de registro del vendedor. |

*SECCIÓN 3: MANUAL DE
USUARIO*

1. MANUAL DE INSTALACIÓN.

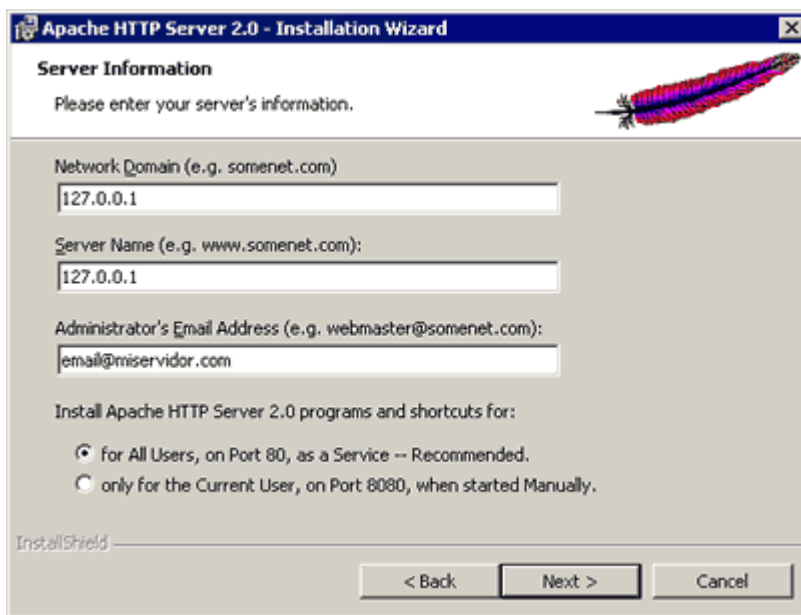
A continuación se detallará el proceso a seguir para la instalación y configuración del software utilizado en el desarrollo del proyecto, de forma que el usuario final sea capaz de poder implantar la aplicación en cualquier ordenador que haga de servidor.

PASO 1: Instalación de Apache 2.2.22.

Para esta labor, en la fecha de realización del proyecto la última versión disponible era la 2.0.50, por lo que el archivo que tenemos que bajar puede ser:

- **apache_2.2.22-win32-x86-no_ssl.exe, disponible en la dirección:**
<http://archive.apache.org/dist/httpd/binaries/win32/>

Una vez bajado el archivo, se ejecutará el instalador y se recorrerán las distintas pantallas hasta llegar al punto donde se solicitan los datos que se muestran a continuación, en cuyo caso pondremos:
Network Domain: 127.0.0.1 (o también en nuestro caso electrokuisland.es).
Server Name: 127.0.0.1 (o también en nuestro caso www.electrokuisland.es).
Administrator's Email Address: un correo e-mail, aunque no es totalmente necesario.
For All Users, on Port 80, as a Service: se selecciona esta opción.



La IP 127.0.0.1 es la dirección IP asociada a nuestra máquina, es decir el host local o vulgarmente conocido como **Localhost**. Cabe destacar que a la hora de probar los scripts en modo local, da igual referirse a 127.0.0.1 o a localhost.

PASO 2: Instalación de PHP5.

Procedemos a descargar el PHP para Windows disponible en la dirección <http://www.php.net/> de la sección *downloads* del menú superior.

Una vez descargado todo el ZIP, se deberá crear una carpeta en el sitio donde se quiera instalar los archivos del servidor (PHP, MySQL...), por ejemplo una carpeta en la raíz del disco duro, de tal forma que quede así: **C:\Servidor**. Dentro de esta carpeta se creará otra carpeta llamada PHP. Luego se procede a extraer los archivos del ZIP dentro de esa carpeta PHP, tal que los contenidos del ZIP quedarán dentro de la ruta **C:\Servidor\PHP**

Posteriormente hay que coger el todos los archivos DLL (a excepción de los archivos DLL contenidos en la carpeta EXT) localizados en la carpeta principal **C:\Servidor\PHP** y copiarlos al directorio **System32** de la carpeta de Windows.

CONFIGURACIÓN DEL ARCHIVO **php.ini**.

El siguiente paso es configurar el archivo **php.ini**. Se renombrará el archivo **C:\Servidor\PHP\php.ini-dist** por **php.ini**. A continuación será necesario editarlo con un editor de texto. A continuación se tendrá que indicar a PHP dónde se guardan las distintas extensiones. Dentro del archivo **php.ini** buscamos **extension_dir** y le ponemos la carpeta que contiene los archivos **php_xxx.dll**, que por defecto es la carpeta **"/ext"** dentro de PHP. La línea correspondiente del fichero debe quedar así:

```
# Directory in which the loadable extensions (modules) reside.  
extension_dir = C:/Servidor/PHP/ext/
```

Además se podrán activar las extensiones disponibles del listado que se crea que se vayan a utilizar, para lo cual se buscará **"Windows Extensions"** en el archivo y para cargar las extensiones se prescindirá del **“;”** de delante. Por ejemplo, si se quisiera cargar la extensión **gd2.dll** (para manejar las funciones PHP relativas a imágenes) deberíamos cambiar **;extension=php_gd2.dll** por **extension=php_gd2.dll**.

Es importante resaltar que hay algunas extensiones que requieren de librerías extra que no vienen en el paquete completo de PHP, y para hacerlas funcionar serán necesario buscar dichas librerías. En concreto la librería **php_mysql.dll** es la que permite manejar las funciones relacionadas con MySQL, y por defecto en PHP5 viene desactivada, por tanto se tendrá que activar de la forma indicada antes: se busca **;extension=php_mysql.dll** y se quita el **;** de delante.

Para trabajar con sesiones, se deberá especificar un directorio donde se guarden los archivos temporales. Al igual que antes, se localiza la opción **session.save_path** y le damos el valor de un directorio que exista (o que se vaya a crear):

```
session.save_path = C:/Servidor/PHP/sessions/
```

A continuación se copia el archivo **php.ini** a la carpeta **C:\Windows**.

Después de esto el servidor deberá “ser consciente” de la existencia de PHP 5.

CONFIGURACIÓN DEL ARCHIVO **httpd.conf** DE APACHE.

En esta ocasión se debe editar el archivo **httpd.conf** que se encuentra en la carpeta **Conf** dentro del directorio de Apache.

Buscamos **Dynamic Shared Object (DSO) Support** que es donde se cargan los módulos. Ahí se cargará el módulo de PHP para Apache, dando la dirección del archivo **php5apache2.dll** de PHP5 que lo contiene. Encontramos el siguiente ejemplo:

```
# Example:  
# LoadModule foo_module modules/mod_foo.so  
#
```

Y ahora añadimos debajo:

```
LoadModule php5_module C:/Servidor/PHP/php5apache2.dll, de tal  
forma que aparezca lo siguiente:
```

```
# Example:  
# LoadModule foo_module modules/mod_foo.so  
#
```

```
LoadModule php5_module C:/Servidor/PHP/php5apache2.dll.
```

El directorio predeterminado para guardar nuestras páginas es el `htdocs` de Apache:

C:\Archivos de programa\Apache Group\Apache\htdocs

Este directorio se puede cambiar fácilmente de forma que busquemos **DocumentRoot** y aparece esto:

```
#
# DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this
# directory, but
# symbolic links and aliases may be used to point to other
# locations.
#
DocumentRoot "C:/Archivos de programa/Apache Group/Apache/htdocs"
```

Esta carpeta se puede sustituir por la elegida para servir los distintos archivos de nuestra aplicación. Todos los archivos que vayan a ser probados con Apache deberán estar localizados en esta carpeta. Por ejemplo nos creamos una carpeta "paginas" dentro del directorio del servidor:

```
DocumentRoot "C:/Servidor/ paginas /".
```

Por lo que construiremos toda nuestra página dentro de ese directorio, que equivale al directorio raíz de nuestro host local.

Ahora buscamos el **DirectoryIndex** y aparece algo como esto:

```
#
# DirectoryIndex: sets the file that Apache will serve if a
# directory
# is requested.
#
# The index.html.var file (a type-map) is used to deliver content-
# negotiated documents. The MultiViews Option can be used for the
# same purpose, but it is much slower.
#
DirectoryIndex index.php index.html index.htm index.html.var
```

Se pueden usar más nombres si se prefiere, de forma que se ejecutarán por orden: si no encuentra el primero, pasa al segundo; si no está el segundo, pasa al tercero, y así sucesivamente hasta que si no encuentra ninguno entonces da error. Si estamos instalando PHP5 también podemos añadir la extensión `.php5` además de las que aparecen:

```
AddType application/x-httpd-php .php .php3 .php4 .phtml
AddType application/x-httpd-php-source .phps
```

La primera indica las extensiones que serán interpretadas por Apache. Por ejemplo podemos añadir la extensión `.htm` o `.html` para que el Apache ejecute el código PHP contenido en esas páginas (es decir, así el uso de código PHP no está limitado exclusivamente a archivos `*.PHP`). La segunda sirve para que si se entra en una página **xxxxx.php**s entonces se muestra el código PHP a color, muy útil si se quiere por ejemplo mostrar el código fuente a color en una página. De tal forma que al final todo queda así:

```
#
# DirectoryIndex: sets the file that Apache will serve if a
# directory
# is requested.
#
# The index.html.var file (a type-map) is used to deliver content-
# negotiated documents. The MultiViews Option can be used for the
# same purpose, but it is much slower.
#
```

SECCIÓN 3: MANUAL DE USUARIO

```
DirectoryIndex index.php index.html index.htm index.php3
index.php4 index.phtml index.html.var
AddType application/x-httpd-php .php .phtml .inc .php3 .inc .php4
.phtml
AddType application/x-httpd-php-source .phps
```

Después de esto se puede ya guardar los cambios del archivo `httpd.conf`.

PROBANDO EL SERVIDOR.

Ahora se deberá probar el Apache y PHP, para lo cual deberemos arrancar el Apache, accederemos desde los accesos directos que se crean en el menú de inicio al acceso directo **Monitor Apache Service** y aparecerá un icono al lado del reloj. Pinchando en el icono, sale un menú desde donde podremos iniciar, apagar y reiniciar el Apache. Si Apache se instaló como servicio de Windows, y elegimos inicialización automática, no será necesario inicializarlo explícitamente, puesto que esto será automático cuando encendamos la máquina, a la vez que Windows. Si ya estaba encendido le daremos a Restart (útil si cambiamos alguna configuración de PHP o del fichero de configuración de Apache) para que al reiniciarlo surtan efecto los cambios que se han realizado.

Si todo ha ido bien, podemos crear un archivo llamado por ejemplo **info.php** dentro de la carpeta de nuestra web cuyo contenido sea exclusivamente el siguiente:

```
<?
phpinfo();
?>
```

Si accedemos mediante nuestro navegador a la dirección **http://127.0.0.1/info.php** o **http://localhost/info.php** (o bien **www.electrokuisland.es/info.php**) con el Apache activo deberemos visualizar una página de información de PHP. Si no sale, algo ha fallado, y se deberán revisar los pasos dados hasta ahora.

Salvo que se edite el fichero `host`, que cuenta con la ruta: `C:\WINDOWS\system32\drivers\etc`, para acceder a nuestro servidor local y así poder probar nuestros script y las páginas de nuestra web, siempre accederemos con la dirección **127.0.0.1** o **localhost**. El citado fichero sirve para hacer creer a la máquina que debe salir a Internet a buscar la página solicitada, pero al realizarse la resolución de nombres y encontrar en este fichero la línea:

```
127.0.0.1 www.electrokuisland.es
```

será equivalente a poner **localhost** o **127.0.0.1**.

En general, para visualizar o acceder a cualquier archivo (PHP, HTML, TXT, ZIP, GIF, JPG, etc.) que tengamos en nuestra carpeta de la web, lo haremos de la siguiente forma:

http://localhost/CARPETA/ARCHIVO.EXT, salvo que se haya configurado la opción de no permitir el listado de ficheros en Apache, con lo que no se podría visualizar nada.

Donde CARPETA es la carpeta (o serie de carpetas) relativas a la raíz en que se encuentra el ARCHIVO.EXT

Por tanto resaltar que no sólo podemos hacer llamadas a script PHP, sino también a páginas HTML, descarga de archivos, imágenes, etc., tan solo poniendo la dirección correcta en el navegador y siempre que la configuración lo permita. De esta forma, podemos tener nuestra web funcionando en nuestro propio ordenador para construirla desde cero o bien hacer las pruebas que necesitemos sin tener que estar conectados a Internet

PASO 3: Instalación de MySQL .

Se realizará la instalación de una versión 5.X, por lo que en este caso buscaremos la versión que corresponda en los downloads para Windows en la dirección: <http://dev.mysql.com/downloads/mysql/5.0.html>.

En la fecha de realización del proyecto, la versión utilizada fue la 5.5.

La ruta elegida en el programa de instalación será **C:\Servidor\MySQL** para tener el PHP y MySQL en la misma carpeta. Una vez finalizada la instalación, solo hay que llamar al `winmysqladmin.exe` que estaría en **C:\Servidor\MySQL\bin** y la primera vez que lo ejecutemos nos pide un nombre de usuario y contraseña, (si es que la versión más actual no lo hace al acabar el

proceso de instalación) que introduciremos sin más. Por ejemplo podemos poner el usuario "**Pepe**" y la contraseña "**hola**", pero se pueden dejar el usuario y contraseña que vienen especificados por defecto con **root**.

Como se ha dicho, por defecto, MySQL crea un usuario de nombre **root** y sin contraseña, y éste es el usuario que usaremos en nuestros.

Notas Finales.

Es importante destacar que estamos trabajando con programas servidores, con lo cual estos programas tienen que estar activos cuando se vayan a hacer pruebas. Esto quiere decir que el Apache debe estar conectado, que el script debe estar dentro de la carpeta especificada anteriormente en DocumentRoot, además también MySQL debe estar funcionando para trabajar con bases de datos.

2. MANUAL DE LA APLICACIÓN.

En esta sección se procederá a presentar el manual de la aplicación, de forma que se recojan las acciones principales que se pueden realizar en las distintas secciones de cada uno de los tipos de usuario. Este manual se dividirá en cuatro partes; la primera dedicada a la sección de cliente no registrado, la segunda dedicada a la sección de cliente registrado, la tercera dedicada a la sección de vendedor y la cuarta dedicada a la sección del administrador. Cada una de las secciones cuenta con apartados comunes en sus respectivos menús principales, lo cual propicia que dichas secciones sean descritas una única vez.

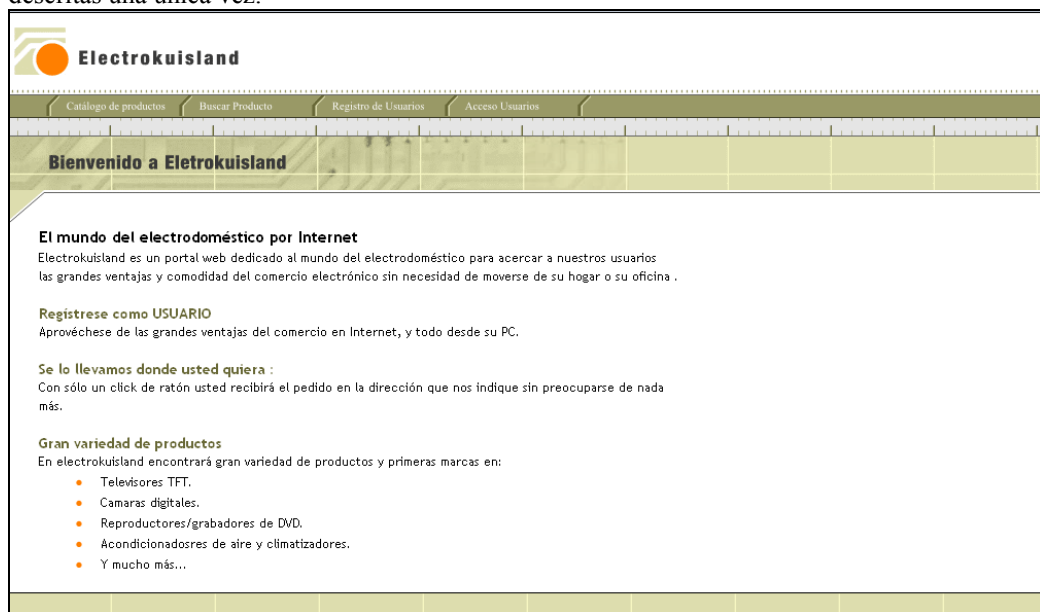


Figura 1. Página principal.

Estos apartados ya no serán citados en cada una de las secciones siguientes, aunque estarán presentes en sus menús.

2.1. SECCIÓN CLIENTE NO REGISTRADO.

Esta sección será en la que todo cliente potencial de la tienda podrá navegar por la página principal, ya sea accediendo a las secciones comunes anteriormente citadas o registrándose como cliente.

2.2. REGISTRO DE CLIENTES.

Esta opción del menú de la página principal permite a cualquier usuario potencial registrarse como usuario cliente de la tienda en una de sus dos modalidades: usuario cliente particular y usuario cliente empresa. Como ya se ha citado en diferentes ocasiones a lo largo de este documento, estos

dos tipos de clientes tienen diferente formulario de registro, y su almacenaje es distinto en el sistema, pues a pesar de ser ambos clientes registrados, los datos identificativos no son los mismos.

Electrokuisland

Eleccion de tipo de cliente | Volver al inicio

Registro de Usuarios

INSTRUCCIONES DE REGISTRO ⓘ
FORMATO DE LOS CAMPOS ⓘ
CONDICIONES DE USO ⓘ

Acepto No Acepto

CONDICIONES DE USO
 El cliente se compromete a posibilitar la entrega del pedido solicitado facilitando una dirección en la que pueda ser entregado el pedido dentro del horario habitual de entrega de mercancías. En caso de incumplimiento por parte del cliente de esta obligación, se podrá considerar anulado el orden y devolver el posible importe ya pagado. En caso de que el cliente lo desee se puede enviar el pedido con la empresa de mensajería, a la nave más cercana a la habitación del cliente; éste procederá a retirar allí el paquete presentando un carné de identidad y el número de albarán del envío.

Formulario de cliente particular

Login: Password:

Nombre: Apellidos:

DNI: Dirección:

Ciudad: Provincia:

Código Postal: Teléfono:

Email: Código de cuenta:

Figura 2. Registro de cliente.

Una vez elegido el tipo de formulario adecuado al tipo de cliente que somos, se procederá a rellenar los campos de dicho formulario teniendo en cuenta los formatos y valores válidos especificados en la fase de pruebas anterior, teniendo en cuenta que todos los campos son requeridos. Una vez cumplimentados todos los datos de forma correcta, la aplicación nos lleva a la zona de usuario registrado, donde podremos realizar las acciones propias de un cliente.

Si en el proceso de registro alguno de los campos es erróneo, la aplicación lanzará un mensaje de error indicándonoslo.

Electrokuisland

Eleccion de tipo de cliente | Volver al inicio

Registro de Usuarios

INSTRUCCIONES DE REGISTRO ⓘ
FORMATO DE LOS CAMPOS ⓘ
CONDICIONES DE USO ⓘ

Acepto No Acepto

EXISTEN ERRORES DE REGISTRO

| | |
|--------------------------------|-------------------------------------|
| Login incorrecto. | Password incorrecto. |
| Nombre incorrecto. | Apellidos incorrectos. |
| DNI o CIF incorrecto. | La dirección es obligatoria. |
| La ciudad es obligatoria. | La provincia es obligatoria. |
| Código postal incorrecto. | Teléfono incorrecto. |
| Correo electrónico incorrecto. | Código cuenta corriente incorrecto. |

Formulario de cliente particular

Login: Password:

Nombre: Apellidos:

DNI: Dirección:

Ciudad: Provincia:

Código Postal: Teléfono:

Email: Código de cuenta:

Figura 4. Errores de registro.

2.3. BÚSQUEDA DE PRODUCTOS.

En este apartado el usuario podrá realizar la búsqueda de productos en las distintas categorías que componen el catálogo de la tienda. Para ello el usuario deberá escoger la categoría sobre la que quiere realizar la búsqueda, introducir la(s) palabra(s) clave para la misma y hacer clic en el botón de envío de la consulta. Después de procesar la consulta, aparecerán los resultados si es que existen.

algunos que Mostrar, de lo contrario la aplicación nos mostrará un mensaje informativo acerca de que no se encontró ningún resultado que mostrar.



Fig

ura 5. Búsqueda de productos.

2.4. CATÁLOGO DE PRODUCTOS.

En cuanto al catálogo de productos, el usuario podrá realizar una visualización por categorías y posteriormente una visualización de productos de esta, o bien una visualización de todos los productos del catálogo, sin realizar ningún tipo de filtrado.

Las categorías se mostrarán en el catálogo, apareciendo en este el nombre de la categoría y una imagen representativa, posteriormente si se hace clic sobre la imagen de la categoría, se despliegan todos los productos que la forman.

Tanto si el listado de productos se realiza por categorías o por productos, aparecerá en primera instancia información del nombre y la imagen del producto y si se hace clic sobre la imagen, se abre una nueva ventana con toda la información relativa al producto.

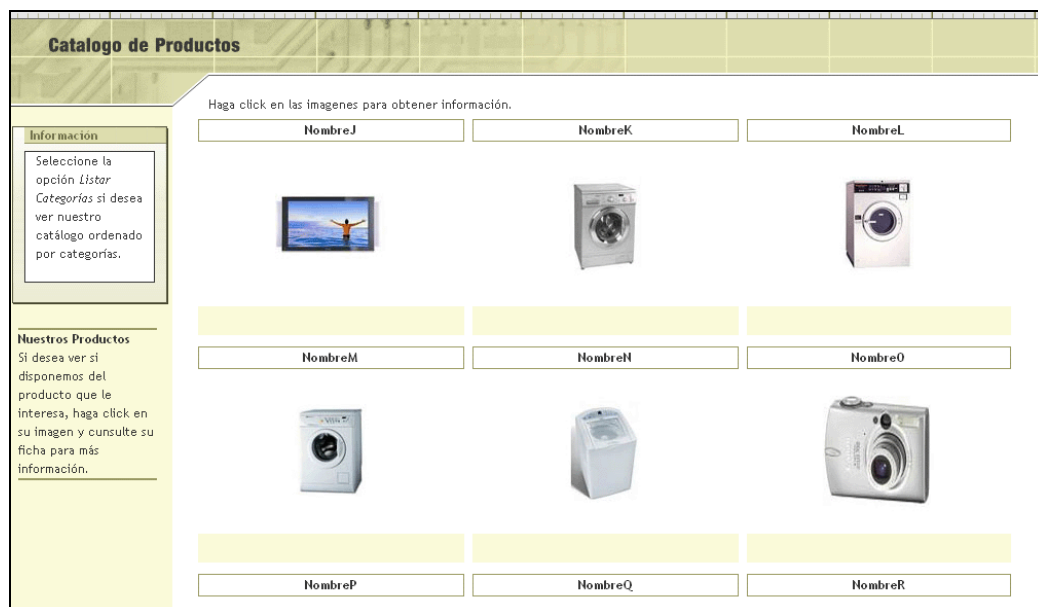


Figura 6. Catálogo de productos.



Figura 7. Detalle de producto.

2.5. IDENTIFICACIÓN DE USUARIO.

En el caso de ser ya un usuario registrado en el sistema, debemos proceder a identificarnos como usuario, sea cual sea el tipo, para poder acceder a nuestra cuenta de usuario. Tras introducir nuestro nombre de usuario y contraseña y si todo es correcto, accederemos directamente a la sección que nos corresponda, dependiendo del tipo de usuario que seamos.

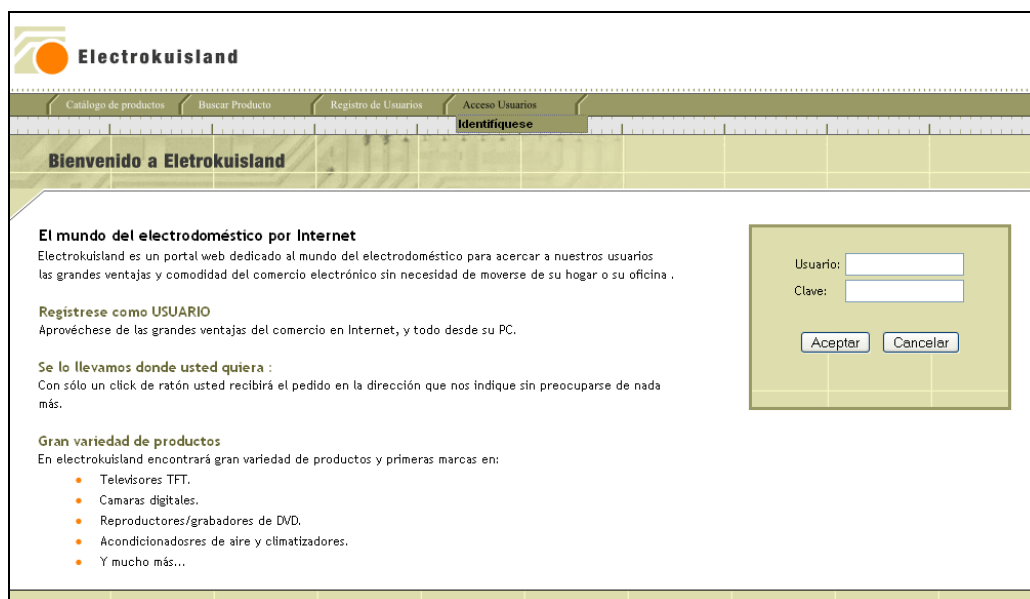


Figura 6. Identificación de usuario.

Si por el contrario hemos cometido algún error en el formulario de identificación, ya sea porque el login o la contraseña son incorrectos, la aplicación lanza un mensaje de error, que nos advierte de este hecho, dándonos una nueva oportunidad de identificarnos.



Figura 7. Error de identificación.

2.6. SECCIÓN DE CLIENTE.

En la página principal de la sección de cliente se pueden encontrar, además de las acciones comunes ya citadas, las siguientes opciones:

- Administrar Cuenta.
Esta opción de menú se divide en otras tres posibles acciones:
 1. Ver Datos Cliente.
 2. Modificar Datos Cliente.
 3. Causar Baja.
- Comprar.
- Salir.



Figura 8. Inicio usuario cliente.

1. Administrar Cuenta.

En esta opción el usuario cliente puede proceder a gestionar sus datos personales almacenados en el sistema, además de ofrecerse la posibilidad de causar baja en el mismo. Las opciones posibles son:

- Consulta de datos personales.
- Modificación de datos personales.
- Causar Baja

1.1. Consulta de datos personales.

Si el usuario cliente hace clic sobre la opción de consulta, el sistema procede a mostrar sus datos personales almacenados.

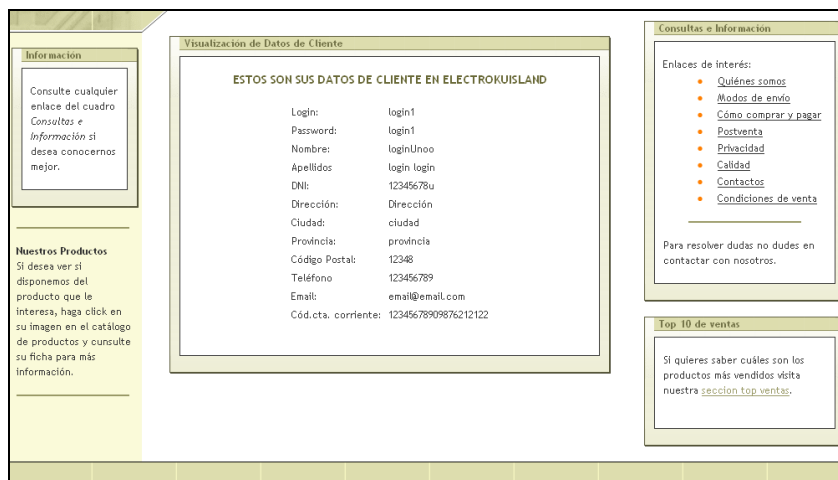


Figura 9. Consulta de datos de cliente.

1.2. Modificación de datos personales.

En este caso si el usuario cliente hace clic sobre la opción de modificación, el sistema le mostrará un formulario de modificación en el que los valores por defecto mostrados serán los que constan actualmente acerca de él. Para evitar inconsistencias y efectos laterales, el sistema no permitirá modificar el nombre de usuario o login. Los datos a modificar podrán ser cualquiera de los mostrados. Si los datos introducidos son correctos, en caso de haber variado alguno, o si se aceptan los datos que constaban, el sistema muestra un mensaje de éxito.

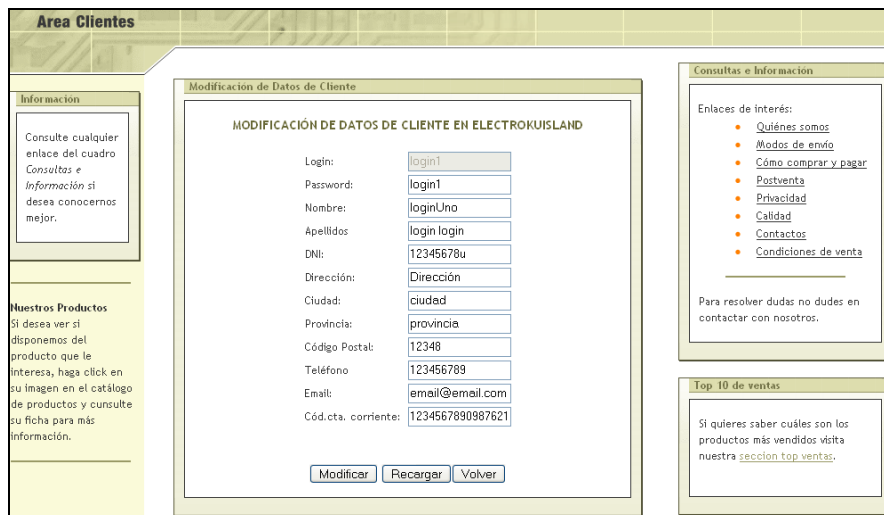


Figura 10. Modificación de datos de cliente.



Figura 11. Éxito en modificación de datos de cliente.

1.3. Causar baja en el sistema.

Esta opción es la que permite al usuario cliente causar baja en el sistema. Cuando se produce esta selección el sistema muestra un mensaje de advertencia y ofrece la posibilidad de confirmar la baja. Si se confirma esta opción, el sistema procederá a dar de baja al usuario.

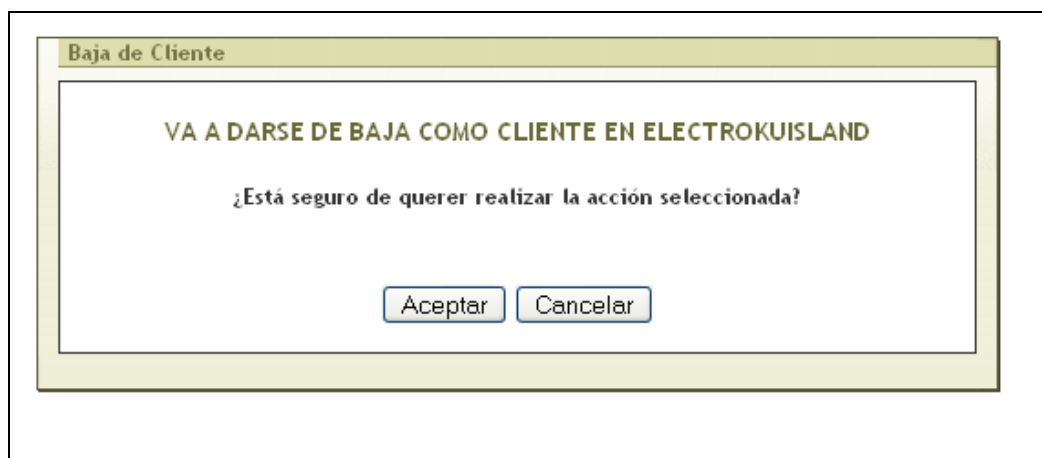


Figura 12. Mensaje de baja de cliente.



Figura 13. Resultado de baja de cliente.

2. Comprar.

Esta opción es la que da sentido a la tienda virtual, pues dar la posibilidad de realizar una compra desde Internet es el cometido principal de la aplicación.

Para realizar la compra, se debe seleccionar primero esta opción en el menú de usuario. Una vez hecho esto, se despliega ante el usuario un listado de los productos activos actualmente en la tienda, preparados para ser añadidos al carrito de la compra. Si queremos añadir un producto a nuestro carrito, deberemos hacer click en el espacio habilitado para ello en cada cuadro de producto. En el momento en que el producto es añadido, pasa a formar parte del carrito de la compra y la opción de añadir al carrito se transformará por la de eliminar dicho producto del carrito. La forma en que es modificado el contenido del carrito de la compra será descrita en el siguiente punto.

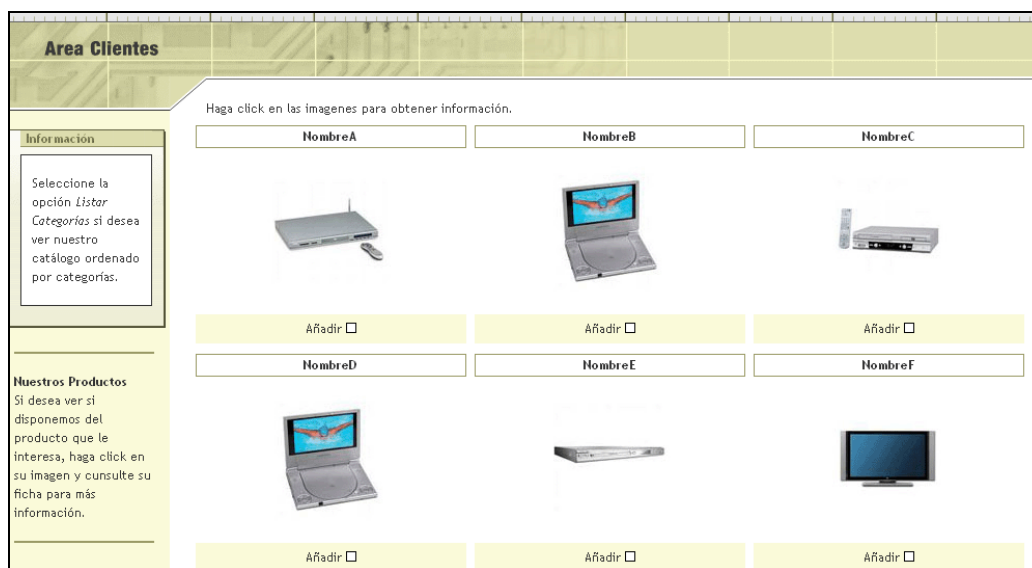


Figura 14. Inicio del proceso de compra.

2. Modificar el carrito.

Mediante esta opción el usuario podrá modificar el número de unidades de cada producto del carrito, eliminar cada uno de ellos o vaciar por completo el carrito de la compra.

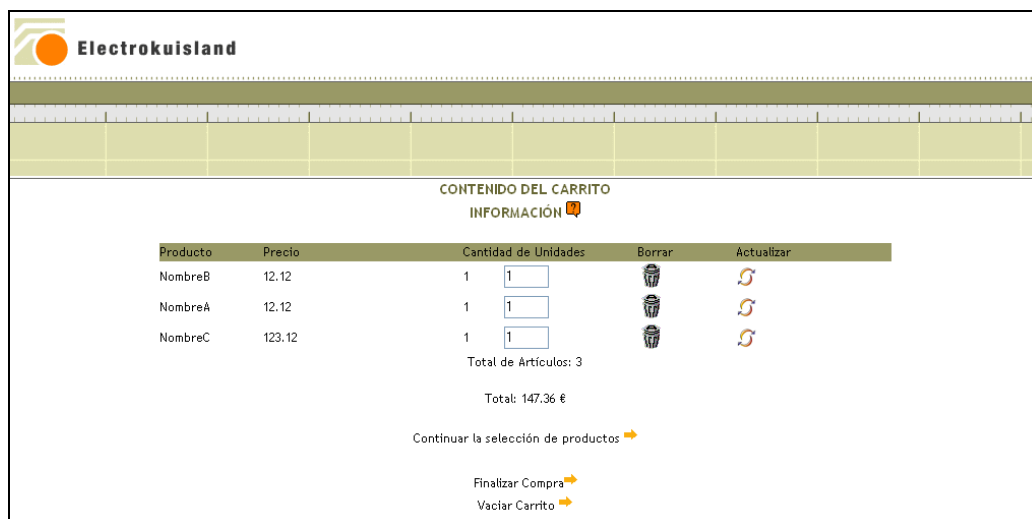


Figura 15. Modificar contenido del carrito.

3. Pagar.

Mediante esta opción el usuario podrá confirmar su pedido y especificar la forma de pago en la ventana que se muestra cuando se decide formalizar el pedido. Se mostrará un informe al usuario con los datos de entrega del pedido. Para formalizar el pedido existen dos formas de entrega y dos formas de pago.

En cuanto a las formas de entrega se dan las posibilidades de entrega a domicilio y entrega a recoger en local. En caso de ser elegida la segunda opción, el pedido se formalizará sin almacenar dirección, ciudad, etc de entrega.

En cuanto a las formas de pago, si la opción elegida es la de pago con tarjeta de crédito, la aplicación se conecta con la pasarela de pagos virtual para formalizar el pedido. Si la opción elegida es la de pago contra reembolso, la aplicación formaliza el pedido como pendiente. En cualquier caso después de formalizar el pedido, la aplicación muestra un mensaje de éxito en caso de que todo fuese bien, o de fracaso en caso de ocurrir algún error.

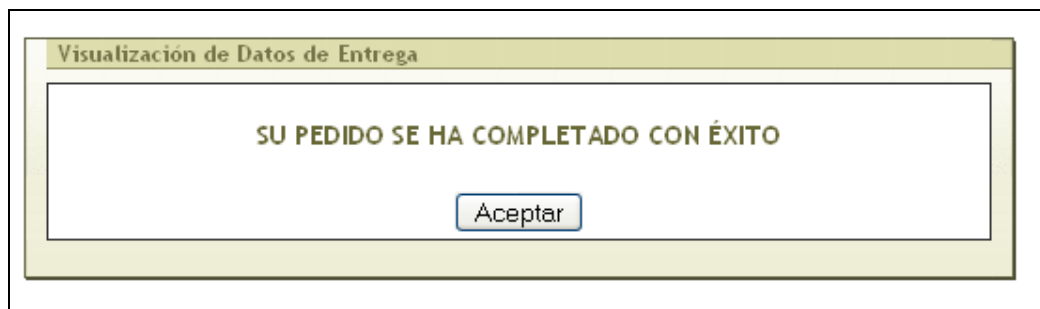


Figura 16.Éxito en pedido.

2.7. SECCIÓN DE VENDEDOR.

En la página principal de la sección de vendedor se puede encontrar el menú principal con las siguientes opciones, además de las comunes ya descritas:

- Administrar Cuenta.

Esta opción de menú se divide en otras tres posibles acciones:

1. Ver Datos Cliente.
2. Modificar Datos Cliente.
3. Causar Baja.

- Administrar Productos.

Esta opción de menú se divide en otras tres posibles acciones:

1. Añadir Producto.
2. Editar Producto.
3. Eliminar Producto.



Figura 17. Inicio usuario vendedor.

1. Administrar Cuenta.

En esta opción el usuario cliente puede proceder a gestionar sus datos personales almacenados en el sistema, además de ofrecerse la posibilidad de causar baja en el mismo. Las opciones posibles son:

- Consulta de datos personales.
- Modificación de datos personales.
- Causar Baja

1.2. Consulta de datos personales.

Si el usuario vendedor hace clic sobre la opción de consulta, el sistema procede a mostrar sus datos personales, y los de la persona de contacto, almacenada en el sistema. Estos datos serán los que en su momento introdujo el usuario administrador en el alta del usuario vendedor, datos facilitados por el usuario vendedor al usuario administrador.

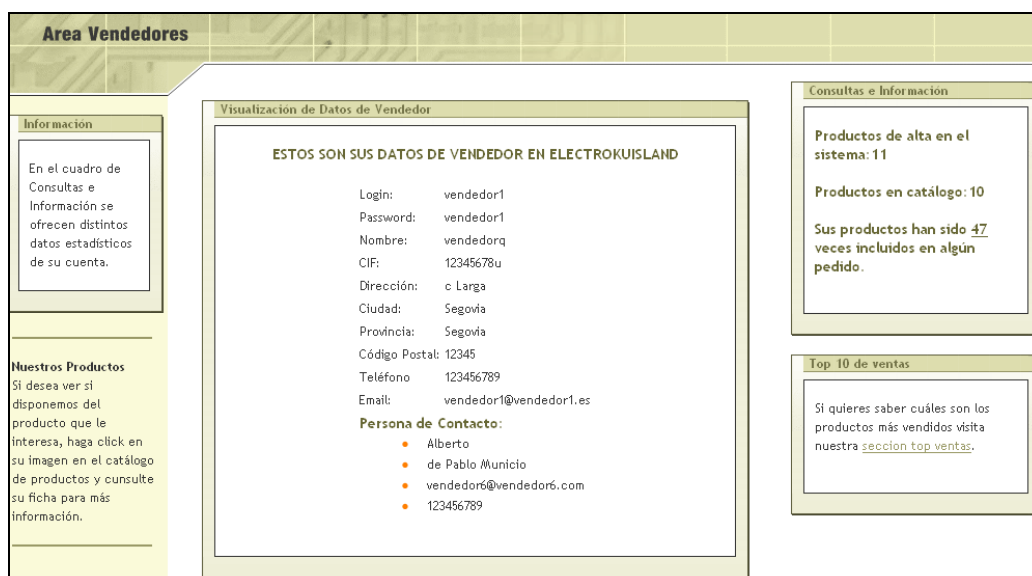


Figura 18. Consulta de datos de vendedor.

1.1. Modificación de datos personales.

Si el usuario vendedor hace clic sobre el icono de modificación, el sistema le mostrará un formulario de modificación en el que los valores por defecto mostrados serán los que constan actualmente acerca de él. Como ya ocurría en el caso del usuario cliente, el sistema no permitirá modificar el nombre de usuario o login. Los datos a modificar podrán ser cualquiera de los mostrados. Si los datos introducidos son correctos, en caso de haber variado alguno, o si se aceptan los datos que constaban, el sistema muestra un mensaje de éxito. Cabe destacar que se ofrece la posibilidad de variar también los datos de la persona de contacto, si se hace clic sobre el enlace que aparece en el formulario de edición del vendedor, como se muestra en la siguiente imagen.

The image shows a web form titled "MODIFICACIÓN DE DATOS DE PERSONA DE CONTACTO" within the "Electrokuisland" interface. The form contains the following fields and buttons:

- Nombre:
- Apellidos:
- Teléfono:
- Email:
- Buttons:

Figura 19. Modificación de datos de persona de contacto.

1.3. Causar baja.

Esta opción es la que permite al usuario vendedor causar baja en el sistema. Cuando se produce esta selección el sistema muestra un mensaje de advertencia y ofrece la posibilidad de confirmar la baja. Si se confirma esta opción, el sistema procederá a dar de baja al usuario vendedor y sus datos desaparecen del sistema.

2. Administrar productos.

El usuario vendedor podrá realizar la gestión de productos en este punto. Las opciones posibles serán:

- Alta de productos.
- Edición de productos.
- Baja de productos.

2.1. Alta de productos.

En esta opción se ofrece al vendedor la opción de dar de alta en el sistema un producto que suministre a la tienda, y que pasará a formar parte del catálogo, siempre y cuando el administrador así lo desee, desbloqueando dicho producto, puesto que el estado en el que queda almacenado el producto es de “suspendido”, no apareciendo por tanto en el catálogo.

The screenshot shows a web interface for sellers. The main area is titled 'Insercion de Producto' and contains a form for adding a new product. The form has the following fields: 'Identificador' (text input), 'Nombre' (text input), 'Imagen' (text input with an 'Examinar...' button), 'Unidades' (text input), 'Precio' (text input), 'Nombre de categoría' (dropdown menu showing 'Acondicionadores de Aire'), and 'Descripcion' (text area). At the bottom of the form are 'Aceptar' and 'Limpiar' buttons. On the left, there is an 'Información' sidebar with text about consulting data and a 'Nuestros Productos' section. On the right, there is a 'Consultas e Información' sidebar showing statistics like 'Productos de alta en el sistema: 27' and 'Productos en catálogo: 25'.

Figura 20. Alta de productos de vendedor.

2.2. Editar productos.

Si el usuario vendedor elige la opción de edición, el sistema le mostrará un listado de los productos que constan en el sistema y de los que el es el proveedor, así como las unidades que quedan del producto en el almacén, la categoría a la que pertenece, y la fecha de suministro.

Posteriormente a que se seleccione un producto del listado se habilitará el botón que da paso al formulario de modificación del mismo, en el que se pueden visualizar los datos que constan acerca del producto actualmente en el sistema.

The screenshot shows a form for editing product data. At the top, it says 'MODIFICACIÓN DE DATOS DE PRODUCTO' and features an image of a laptop. The form fields are: 'Identificador' (text input with 'ProductoB'), 'Nombre' (text input with 'NombreB'), 'Cambiar Imagen' (radio buttons for 'Si' and 'No', with 'No' selected), 'Imagen' (text input), 'Unidades' (text input with '1'), 'Precio' (text input with '12.12' and a '€' symbol), 'Nombre de categoría' (dropdown menu with 'Reproductores de DVD'), and 'Descripcion' (text area). At the bottom are 'Modificar', 'Recargar', and 'Volver' buttons.

Figura 21. Formulario de edición de producto.

2.3. Eliminación de productos.

La baja de productos por parte del usuario vendedor ofrece la posibilidad de que el mismo de d baja productos del sistema, siempre y cuando se cumpla la condición de que no existan productos en el almacén. El sistema implementado para ello es similar al de edición de productos, apareciendo en primer lugar un listado de productos, y una vez escogido uno y si se cumple la condición expresada, se da de baja del sistema el producto en cuestión.

2.8. SECCIÓN DE ADMINISTRADOR.

En la página principal de la sección de administrador se puede encontrar el menú principal con las siguientes opciones, además de las comunes ya descritas:

- Administrar Cuenta.

Esta opción de menú se divide en otras tres posibles acciones:

1. Ver Datos Cliente.
2. Modificar Datos Cliente.

- Administrar Productos.

Esta opción de menú se divide en otras tres posibles acciones:

1. Añadir Producto.
2. Editar Producto.
3. Eliminar Producto.

- Administrar Categorías.

Esta opción de menú se divide en otras tres posibles acciones:

1. Añadir Categoría.
2. Editar Categoría.
3. Eliminar Categoría.

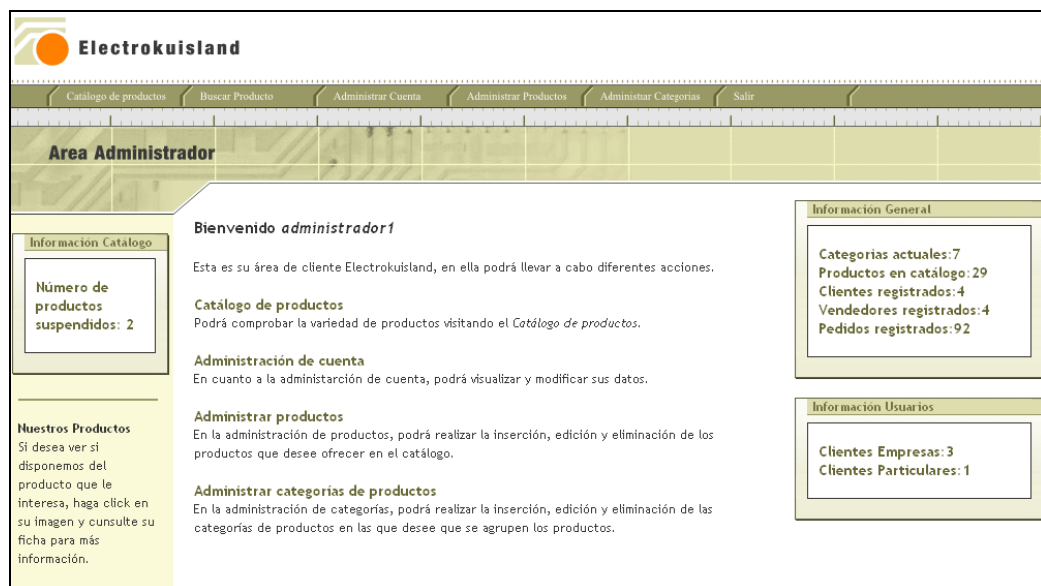


Figura 22. Inicio de usuario administrador.

1. Administrar Cuenta.

En esta opción el usuario administrador puede proceder a gestionar sus datos personales almacenados en el sistema. Las opciones posibles son:

- Consulta de datos personales.
- Modificación de datos personales.

1.2. Consulta de datos personales.

Si el usuario administrador hace clic sobre la opción de consulta, el sistema procede a mostrar sus datos personales almacenados en el sistema.

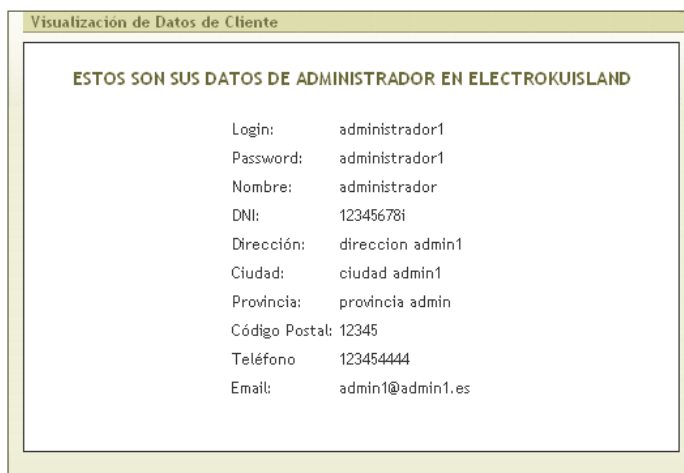


Figura 23. Consulta de datos de administrador.

1.1. Modificación de datos personales.

Si el usuario administrador hace clic sobre el icono de modificación, el sistema le mostrará un formulario de modificación en el que los valores por defecto mostrados serán los que constan actualmente acerca de él. Como ya ocurría en el caso de los otros usuarios, el sistema no permitirá modificar el nombre de usuario o login. Los datos a modificar podrán ser cualquiera de los mostrados.

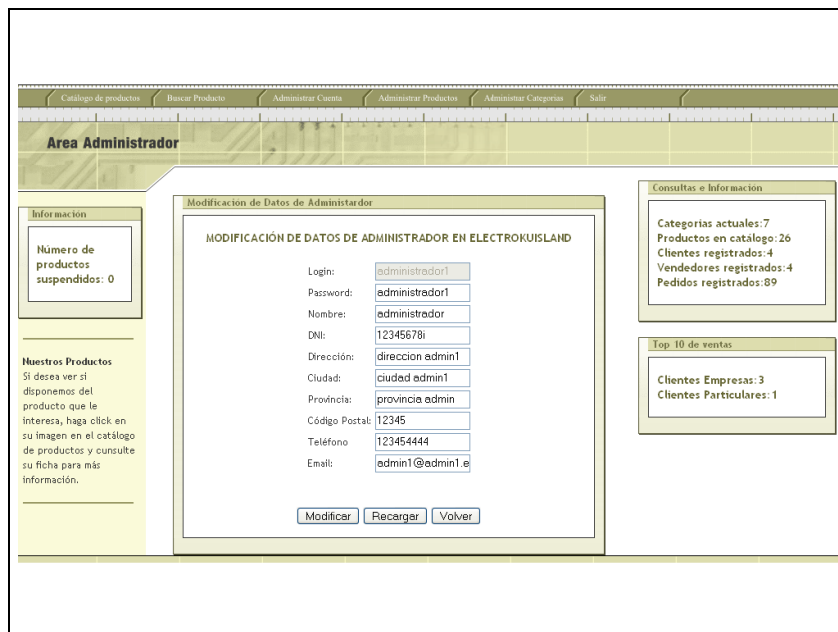


Figura 24. Modificación de datos de administrador.

2. Administrar productos.

El usuario administrador podrá realizar la gestión de productos en este punto. Las opciones posibles serán:

- Alta de productos.
- Edición de productos.
- Baja de productos.