



# Universidad Rey Juan Carlos

Escuela Técnica Superior de Ingeniería Informática

Ingeniería del Software II

## Práctica Obligatoria

*Documento de Diseño Técnico*

Radu Tom Vlad  
David Rufo Valero  
Jesús Sánchez-Oro Calvo  
Adrián Santalla Romero de Ávila

asantalla@siliconkernel.com

5º de Ingeniería Informática

16 de enero de 2010

# Índice

<b>1. DSI 1: Definición de la Arquitectura del Sistema</b>	<b>4</b>
1.1. DSI 1.1: Definición de Niveles de Arquitectura . . . . .	4
1.2. DSI 1.2: Identificación de Requisitos de Diseño y Construcción . . . . .	5
1.3. DSI 1.3: Especificación de Excepciones . . . . .	6
1.4. DSI 1.4: Especificación de Estándares y Norma de Diseño y Construcción . . . . .	6
1.5. DSI 1.5: Identificación de Subsistemas de Diseño . . . . .	7
1.6. DSI 1.6: Especificación del Entorno Tecnológico . . . . .	8
1.7. DSI 1.7: Especificación de Requisitos de Operación y Seguridad . . . . .	10
<b>2. DSI 2: Diseño de la Arquitectura de Soporte</b>	<b>11</b>
2.1. DSI 2.1: Diseño de Subsistemas de Soporte . . . . .	11
2.2. DSI 2.2: Identificación de Mecanismos Genéricos de Diseño . . . . .	11
<b>3. DSI 3: Diseño de Casos de Uso Reales</b>	<b>12</b>
3.1. DSI 3.1: Identificación de Clases Asociadas a un Caso de Uso . . . . .	12
3.2. DSI 3.2: Diseño de la Realización de los Casos de Uso . . . . .	13
3.3. DSI 3.3: Revisión de la Interfaz de Usuario . . . . .	28
3.4. DSI 3.4: Revisión de Subsistemas de Diseño e Interfaces . . . . .	30
<b>4. DSI 4: Diseño de Clases</b>	<b>34</b>
4.1. DSI 4.1: Identificación de Clases Adicionales . . . . .	34
4.2. DSI 4.2: Diseño de Asociaciones y Agregaciones . . . . .	34
4.3. DSI 4.3: Identificación de Atributos de las Clases . . . . .	35
4.4. DSI 4.4: Identificación de Operaciones de las Clases . . . . .	36
4.5. DSI 4.5: Diseño de la Jerarquía . . . . .	37
4.6. DSI 4.7: Especificación de Necesidades de Migración y Carga Inicial de Datos . . . . .	38
<b>5. DSI 6: Diseño Físico de Datos</b>	<b>39</b>
5.1. DSI 6.1: Diseño del Modelo Físico de Datos . . . . .	39
5.2. DSI 6.2: Especificación de los Caminos de Acceso a los Datos . . . . .	41
5.3. DSI 6.3: Optimización del Modelo Físico de Datos . . . . .	41
5.4. DSI 6.4: Especificación de la Distribución de Datos . . . . .	41
<b>6. DSI 8: Generación de Especificaciones de Construcción</b>	<b>42</b>
6.1. DSI 8.1: Especificación del Entorno de Construcción . . . . .	42
6.2. DSI 8.2: Definición de Componentes y Subsistemas de Construcción . . . . .	42
6.3. DSI 8.3: Elaboración de Especificaciones de Construcción . . . . .	43
6.4. DSI 8.4: Elaboración de Especificaciones del Modelo Físico de Datos . . . . .	44
<b>7. DSI 9: Diseño de la Migración y Carga Inicial de Datos</b>	<b>47</b>
7.1. DSI 9.1: Especificación del Entorno de Migración . . . . .	47
7.2. DSI 9.2: Diseño de Procedimientos de Migración y Carga Inicial . . . . .	47
7.3. DSI 9.3: Diseño Detallado de Componentes de Migración y Carga Inicial . . . . .	47
7.4. DSI 9.4: Revisión de la Planificación de la Migración . . . . .	48
<b>8. DSI 10: Especificación Técnica del Plan de Pruebas</b>	<b>49</b>
8.1. DSI 10.1: Especificación del Entorno de Pruebas . . . . .	49
8.2. DSI 10.2: Especificación Técnica de Niveles de Prueba . . . . .	49
8.3. DSI 10.3: Revisión de la Planificación de Pruebas . . . . .	50
<b>9. DSI 11: Establecimiento de Requisitos de Implantación</b>	<b>51</b>
9.1. DSI 11.1: Especificación de Requisitos de Documentación de Usuario . . . . .	51
9.2. DSI 11.2: Especificación de Requisitos de Implantación . . . . .	51

## Índice de figuras

1.	Diagrama de despliegue del sistema. . . . .	4
2.	Diagrama de paquetes. . . . .	8
3.	Diagrama de secuencia de login (correcto). . . . .	13
4.	Diagrama de secuencia de login (incorrecto). . . . .	14
5.	Diagrama de secuencia de logout (correcto). . . . .	15
6.	Diagrama de secuencia de logout (incorrecto). . . . .	15
7.	Diagrama de secuencia de registro (correcto). . . . .	16
8.	Diagrama de secuencia de registro (incorrecto). . . . .	17
9.	Diagrama de secuencia de atención al cliente genérica. . . . .	18
10.	Diagrama de secuencia información corporativa. . . . .	18
11.	Diagrama de secuencia de tareas pendientes. . . . .	19
12.	Diagrama de secuencia de seguimiento de envíos. . . . .	20
13.	Diagrama de secuencia de historico de envíos. . . . .	20
14.	Diagrama de secuencia de obtención de justificantes. . . . .	21
15.	Diagrama de secuencia de apadrinamiento. . . . .	22
16.	Diagrama de secuencia de asociaciones. . . . .	22
17.	Diagrama de secuencia de atención al cliente. . . . .	23
18.	Diagrama de secuencia de colecciones filatélicas. . . . .	24
19.	Diagrama de secuencia de gestión online (correcto). . . . .	25
20.	Diagrama de secuencia de gestión online (incorrecto). . . . .	25
21.	Diagrama de secuencia de búsqueda de sucursales. . . . .	26
22.	Diagrama de secuencia de envío (correcto). . . . .	27
23.	Diagrama de secuencia de envío (incorrecto). . . . .	27
24.	Captura de la interfaz principal de particulares. . . . .	28
25.	Captura de la interfaz principal de empresas. . . . .	29
26.	Captura de la interfaz principal de oficina virtual. . . . .	30
27.	Diagrama de clases del sistema. . . . .	34
28.	Diagrama de componentes. . . . .	43

# 1. DSI 1: Definición de la Arquitectura del Sistema

## 1.1. DSI 1.1: Definición de Niveles de Arquitectura

El particionamiento físico que será realizado en el sistema se muestra un esquema en la figura siguiente:

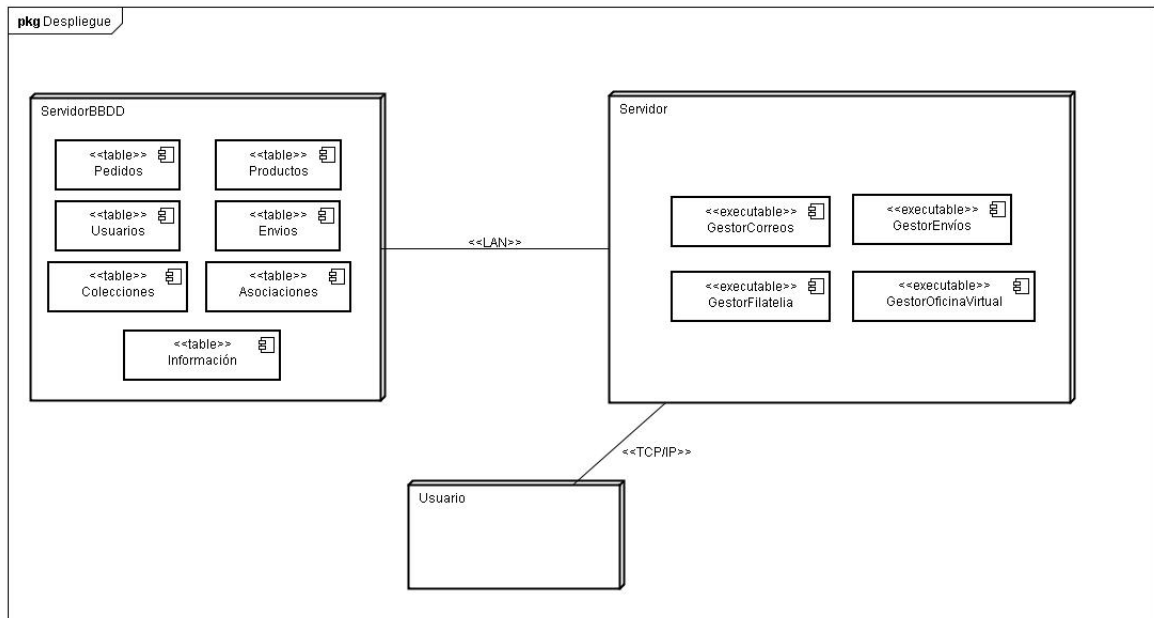


Figura 1: Diagrama de despliegue del sistema.

Los elementos o nodos de la figura se explican a continuación.

Por un lado el sistema tendrá una **base de datos** en la que se almacenará la siguiente información:

- *Tabla Usuarios*: se guardará datos de los clientes registrados, como por ejemplo su Nombre Personal, Apellidos, DNI, Nombre de Usuario, Contraseña, Dirección Particular, Teléfonos de contactos, etc. El campo de la contraseña no será una cadena simple de texto sino que se almacenará con una cierta codificación para la seguridad del usuario.
- *Tabla Pedidos*: cualquier cliente previamente registrado puede realizar un pedido a través del portal, una vez hecha tal acción se guardarán los datos necesarios, como pueden ser la información del Usuario que está realizando el pedido, el nombre del producto junto con una referencia asociada a dicho producto (esta referencia será la misma que tenga ese producto en la tabla correspondiente a los productos), etc.
- *Tabla Colecciones*: se almacenan las colecciones filatélicas. Los datos a guardar son el nombre de la colección, una descripción de ésta, código,...
- *Tabla Productos*: en esta tabla se almacenarán todo tipo de información que describa un producto, es decir, su nombre, código o referencia, descripción, precio, etc.
- *Tabla Envíos*: en ella se guardarán los datos de los envíos, como pueden ser la información del Usuario registrado que realizó el envío, la localización del envío, el código o referencia, descripción o información referente al envío, dirección, etc.
- *Tabla Asociaciones*: almacenamiento de asociaciones referentes a las asociaciones filatélicas. Luego se guardarán teléfonos de esas asociaciones, direcciones, etc. En la aplicación se mostrará un mapa para que el usuario registrado o no puede ver las asociaciones filatélicas más cercanas a su situación, aunque también podrá ver todas las demás.

- *Tabla Información*: en la que se guardarán los datos referentes al resto de la aplicación, es decir, la información de las paginas web estáticas y los demás datos que no se pueden agrupar en una tabla específica.

Ya en la implementación se decidirá que campos son los realmente básicos en cada una de las tablas.

Los datos de las tablas irán creciendo con el paso del tiempo, aunque también el administrador del sistema irá haciendo desaparecer cierta información que ya no sea necesaria desde ese momento. Por lo que el volumen de la base de datos continuará creciendo día a día.

En el servidor es donde se encuentra la lógica de negocio. En él se encontrarán los distintos gestores (de Correos, de Envíos, de Filatelia y de la Oficina Virtual) a través de los cuales se accederá a la base de datos, en el caso de ser necesario.

El usuario realizará las operaciones que él desee dependiendo de si es un usuario Anónimo, Registrado o un usuario referente a una empresa. Y todas estas acciones establecerán una comunicación con el servidor para que éste le responda con la información adecuada a la consulta realizada por el usuario.

Por lo tanto las comunicaciones entre los distintos nodos son:

- El Servidor realizará consultas al Servidor de Base de Datos y éste le responderá con la información adecuada a la consulta realizada. La comunicación entre Servidor y ServidorBBDD, será mediante una conexión de red local (LAN).
- El Usuario a través de las distintas opciones de la página web principal de Correos ([www.correos.es](http://www.correos.es)) irá realizando las consultas a la información que desea o llevar a cabo los operaciones o transacciones que considere oportunas en el caso de ser un usuario registrado. Esta comunicación, entre Usuario y Servidor, será realizada a través de Internet (TCP/IP) por lo que un requisito fundamental en la parte del cliente será la conexión a Internet para el uso de esta aplicación y el requisito fundamental del Servidor es estar en un dominio con una accesibilidad pública para su localización y uso.

Para evitar la saturación del servidor se utilizarán varios de éstos, ya que la media de acceso al portal es de 400.000 (cuatrocientas mil) visitas al mes. Así se evita que los usuarios no vean un cierto retraso a la hora de responder el servidor sobre una determinada acción pedida por el cliente.

Los problemas de concurrencia que podrían surgir cuando se dispone de varios servidores y cada uno tenga accesos al servidor de bases de datos serán solucionados con las técnicas de programación adecuadas para la consistencia de los datos. Por ejemplo, una solución a este problema podría ser el evitar que más de un servidor acceda a un registro de una tabla, con esto se evitaría la inconsistencia de los datos entre servidores a la hora de responder con datos al cliente.

Respecto a la seguridad, las comunicaciones una vez el usuario ha sido registrado y ha entrado correctamente en la aplicación, se usarán conexiones seguras con el fin de no tener problemas con otros usuarios o con usuarios cuyas acciones son mal intencionadas.

En el caso de usuarios con discapacidades también tendrán la posibilidad de utilizar esta aplicación web debido a que se incluirán los métodos apropiados para que estas personas puedan utilizar la aplicación.

## 1.2. DSI 1.2: Identificación de Requisitos de Diseño y Construcción

Los requisitos relacionados con el diseño de la aplicación son:

- El *lenguaje* utilizado en la web será un lenguaje común para que cualquier usuario pueda ser capaz de leer y entender correctamente la información que desea este usuario, tanto los anónimos como los registrados. Por lo que éstos no deberán tener un nivel educativo y cultural demasiado amplio para su uso.

- Las operaciones o transacciones que pueda realizar un cliente registrado serán bastante accesibles y con la suficiente información para su realización correcta. Aunque durante el horario establecido de atención al cliente siempre podrá llamar el usuario para resolver cualquier duda.
- El *rendimiento* ofrecido por los *servidores* será el máximo posible de cada uno de ellos para evitar el colapso de éstos y que la aplicación no funcione debido a éstos problemas. El rendimiento máximo de estos servidores dependerá del uso actual de la web, es decir, según el número de usuarios que en cada momento estén utilizando la aplicación. Estos servidores guardarán la lógica de negocio de la aplicación.
- Además el *rendimiento* ofrecido por el *servidor de base de datos* será también máximo dependiendo de su interacción con los otros servidores. En este servidor se guardarán todos los datos de la aplicación de forma permanente y con sus correspondientes copias de seguridad cada un cierto período, el cual todavía no se ha establecido, ya que dependerá de su volumen.
- Los servidores y el servidor de base de datos estarán ubicados en el mismo lugar ya que su comunicación se realizará a través de una conexión de red local (LAN).
- La conexión entre Usuario y Servidor será a través de Internet (TCP/IP).

### 1.3. DSI 1.3: Especificación de Excepciones

Los comportamientos no habituales de la aplicación, que puede darse en algún momento, son:

- *Caída de los servidores*: los usuarios no podrán utilizar la aplicación. Los elementos afectados serán los propios a la lógica de negocio. El motivo de esta situación puede surgir por una caída de luz en las instalaciones o una renovación de los equipos, aunque esto último será realizado poco a poco para poder evitar dicha situación.
- *Caída del Servidor de Base de Datos*: los usuarios podrán utilizar la aplicación pero no podrán realizar acciones que requieran de información de la base de datos. Por ejemplo, usuarios registrados no podrán entrar en la aplicación, las consultas de tarifas de los productos no podrán ser vistas por los usuarios,... El elemento afectado será la base de datos. Su motivo puede ser, al igual que antes, una caída de la luz o la actualización de la base de datos.
- *Comunicación entre Servidores y Servidor de BBDD*: puede ocurrir la situación de que la conexión de red local entre ellos caiga debido bien a la ruptura de un cable o bien por desenchufar el servidor de BBDD o algún otro servidor del switch con el que tienen comunicación estos nodos. El elemento afectado es la conexión de red local.
- Puede suceder que el usuario en un campo donde sólo deberían introducirse números inserte algún carácter no válido y la aplicación produzca la excepción de caracteres no válidos. Por lo tanto este tipo de excepciones serán tratadas para que la aplicación avise al usuario del tipo de información a rellenar en cada campo para evitar estas situaciones.
- El usuario también deberá tener especial cuidado cuando introduzca fechas ya que no estará permitido por la aplicación la inclusión de fechas anteriores al día actual. Por ejemplo, si el usuario pretende realizar un envío no podrá proporcionar la fecha inicial anterior al día actual. Además se indicará el orden de la fecha, por ejemplo Día/Mes/Año.

### 1.4. DSI 1.4: Especificación de Estándares y Norma de Diseño y Construcción

La aplicación será construida siguiendo estos estándares:

- **W3C**: mediante este estándar se consigue que la aplicación sea multiplataforma y accesible desde la mayor parte de los navegadores, ya que una vez que la aplicación pasa el test del validador de la W3C (<http://validator.w3.org>), ésta puede usarse en cualquier navegador que siga dicho estándar.

- **Nivel A de Conformidad con las Directrices de Accesibilidad para el Contenido Web 1.0 (WCAG 1.0):** las Directrices de Accesibilidad para el Contenido Web 1.0 explican cómo hacer el contenido Web accesible para las personas con discapacidad. Ajustándose a estas Directrices ayudará a hacer la Web más accesible a los usuarios con discapacidades y beneficiará a todos los usuarios.

Referente a la legislación, la principal ley que se debe tener en cuenta es la **Ley Orgánica de Protección de Datos de Carácter Personal**, que como indica en su artículo 1, tiene como objetivo *garantizar y proteger, en lo que concierne al tratamiento de los datos personales, las libertades públicas y los derechos fundamentales de las personas físicas, y especialmente de su honor e intimidad personal y familiar*. Esto es de vital importancia en cualquier tipo de aplicación que maneje datos con información privada de sus usuarios, por lo que el nuevo portal deberá tenerlo en cuenta.

### 1.5. DSI 1.5: Identificación de Subsistemas de Diseño

Según la especificación descrita en el documento de Análisis Funcional, en nuestro sistema podemos encontrar cuatro grandes subsistemas de diseño:

- **Gestor de Correos:** Es el encargado de la gestión de usuarios, y realiza acciones tales como registrar y validar del sistema a usuarios de correos. Además se encarga de obtener la información buscada por los clientes. Por lo que interactuará con la base de datos de usuarios del sistema y con las bases de datos referentes a la información (asociaciones, colecciones y productos).
- **Gestor de Envíos:** Es el encargado de la realización, cancelación y seguimiento de los envíos realizados por los clientes. Por lo que interactuará con la base de datos de envíos y de usuarios del sistema.
- **Gestor de Filatelia:** Es el encargado de realizar todas las gestiones que el sistema debe realizar referentes al apartado filatélico. Por lo que interactuará con las bases de datos de productos, colecciones y pedidos.
- **Gestor de Oficina Virtual:** Es el encargado de realizar todas las gestiones que el sistema debe realizar referentes a la Oficina Virtual de correos. Por lo que interactuará con las bases de datos de usuarios y pedidos.

El diagrama de paquetes que muestra las relaciones entre los diferentes subsistemas es el siguiente:

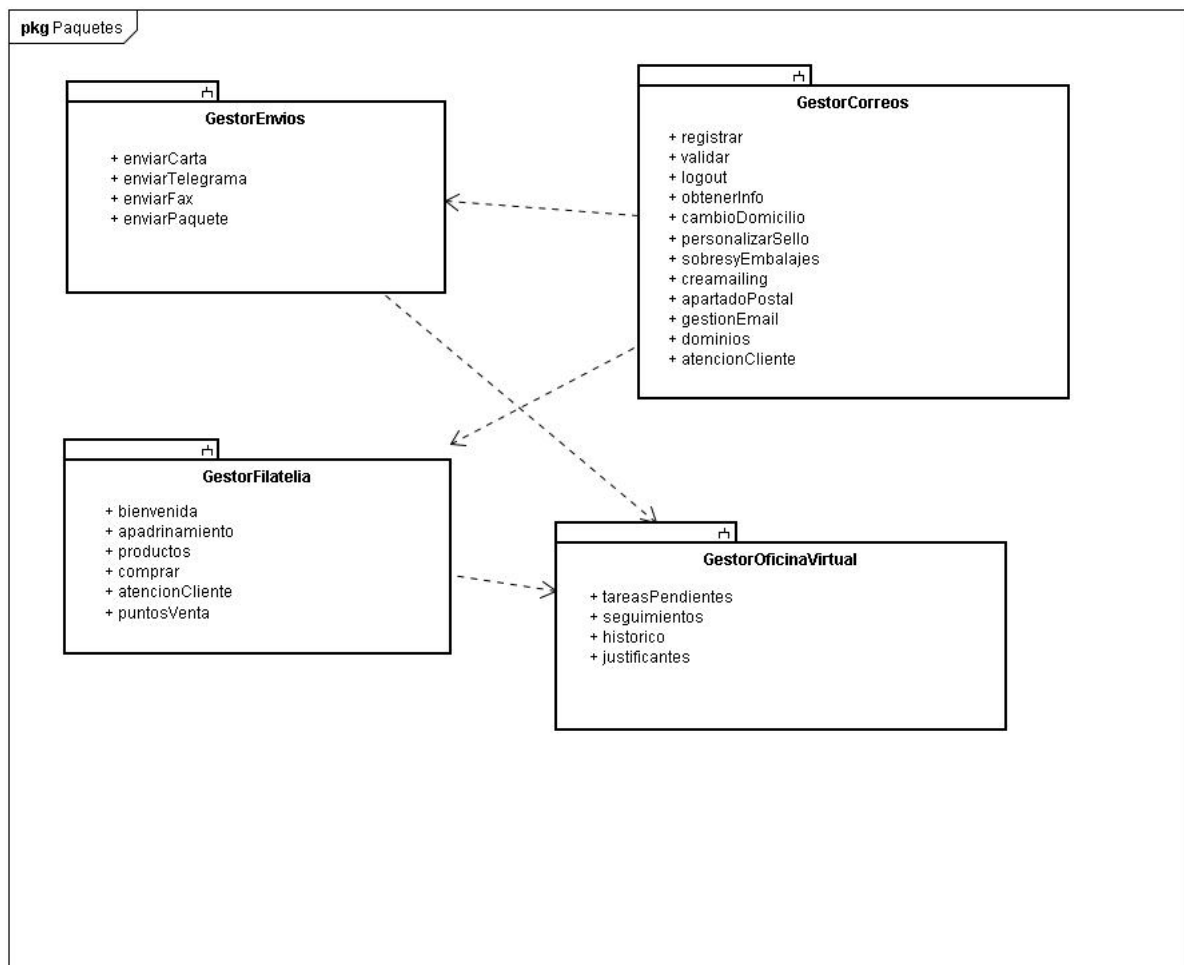


Figura 2: Diagrama de paquetes.

En la figura anterior se puede ver la comunicación que tiene cada uno de los subsistemas especificados, es decir, las relaciones de cada uno de estos. Además, en el caso de ser necesario, cada uno tendrá acceso a la base de datos para registrar la información. En la figura del punto 1.1 de este documento se puede ver esa comunicación de estos subsistemas con el sistema de gestión de base de datos.

## 1.6. DSI 1.6: Especificación del Entorno Tecnológico

Las características que debe tener el *Servidor de Base de Datos* son:

### ■ Hardware:

- Sistema de Almacenamiento: disco duro con una capacidad inicial de dos terabytes (2 TB).
- Memoria RAM: inicialmente sería suficiente con dos gigabytes (2 GB).
- Tarjeta de Red: estándar RJ45 para la comunicación con los demás servidores.
- Unidad DVD de grabación o puerto USB: para copias de seguridad periódicas.
- Procesadores: inicialmente con un procesador debería ser suficiente (estos inicialmente serán de la marca Intel).
- Teclado, ratón y demás componentes necesarios para el correcto encendido de esta computadora.



**■ Software:**

- Sistema Operativo: Windows NT/200 Advanced Server.
  - Base de Datos: Oracle (Lenguaje de consulta estructurado SQL).
  - Software para la realización de las copias de seguridad.
  - J2EE 1.4 para el correcto funcionamiento de las aplicaciones.
  - En caso de que sea necesario: software para la comunicación con los demás servidores.
- No deberá permitir la conexión de otros servidores. Y tampoco debería permitir que los servidores, apropiados para la conexión, puedan situarse en el mismo registro de una misma tabla con el fin de evitar inconsistencia en la información.

Las características de los *Servidores* son:

**■ Hardware:**

- Sistema de Almacenamiento: disco duro con una capacidad inicial de quinientos gigabytes (500 GB). Dentro de este sistema habrá una pequeña parte para una caché en la que se guardarán los datos de las consultas, su tamaño podría ser de dos gigabytes (2 GB).
  - Memoria RAM: inicialmente sería suficiente con cuatro gigabytes (4 GB).
  - Tarjeta de Red: estándar RJ45 para la comunicación con el servidor de base de datos.
  - Procesadores: inicialmente se utilizará un Intel Dual Core.
  - Teclado, ratón y demás componentes necesarios para el correcto encendido de estas computadoras.
- **Software:**
- Sistema Operativo: Windows NT/200 Advanced Server.
  - Navegador: para la comprobación del correcto funcionamiento del sistema.
  - En caso de que sea necesario: software para la comunicación con el servidor de base de datos.
  - J2EE 1.4 para el correcto funcionamiento de las aplicaciones.
  - Entorno de desarrollo: puede ser un procesador de textos normal a través del cual realizar todo lo referente a la implementación.
- Los usuarios accederán a éstos a través de Internet (TCP/IP) por lo que la aplicación deberá alojarse en un dominio para su posible utilización.

Las características que deben tener las computadores de los *Usuarios* son:

**■ Hardware:**

- Tarjeta de Red o tarjeta de red inalámbrica: para la conexión a internet.
  - Teclado, ratón y demás componentes necesarios para el correcto encendido de estas computadoras.
- **Software:**
- Sistema Operativo: Windows, Linux o Macintosh.
  - Navegador Web: para el acceso a la aplicación. (Internet Explorer, Mozilla Firefox,...) Estos navegadores web deben tener activado el permiso de ejecución de JavaScript.
- Requisito fundamental en estas computadoras es su conexión a Internet para tener comunicación con los servidores.

### 1.7. DSI 1.7: Especificación de Requisitos de Operación y Seguridad

La aplicación cuenta con los siguiente mecanismos de seguridad:

- Los servidores de base de datos realizarán *copias de seguridad* (Backup) de forma periódica (este tiempo esta por decidir según la cantidad de información almacenada). Además para la concurrencia este servidor utilizará un mecanismo que no permitirá a más de un servidor acceder al mismo registro de una tabla.
- El *accesos al servidor de base de datos* será sólo desde la red local, por lo tanto no estará conectada a Internet. Además sólo los servidores autorizados tendrán acceso a la información almacenada en éste. Para ésto se utilizara una nombre de usuario y una contraseña (cifrada) para entrar en servidor de BBDD.
- Los demás servidores tendrán un *grupo de usuarios* con sus correspondientes contraseñas (cifradas) que permitirán la entrada a estas computadoras. Éstas estarán conectadas a Internet pero no se permitirá la conexión remota a menos que alguno de los administradores de estos servidores lo permite en un momento determinado.
- Los servidores que contendrán la *lógica de negocio*, a parte de disponer un usuario y clave para entrar en ellos, también se necesitará permisos de administrador para la modificación de aspectos relacionados con la aplicación.
- Los *datos de los usuarios registrados* estarán protegidos debido a la Ley Orgánica de Protección de Datos de Carácter Personal.
- En el caso de producirse una pérdida total de los datos se recuperarán estos a partir de última copia de seguridad hecha.
- Durante la franja horaria de mayor número de usuarios se tendrá especial cuidado con las transacciones realizadas por cada uno de ellos para evitar problemas en los datos y la inclusión de usuarios mal intencionados. Por lo que en este período de tiempo los servidores estarán funcionando a pleno rendimiento para el cumplimiento correcto de seguridad.

## 2. DSI 2: Diseño de la Arquitectura de Soporte

### 2.1. DSI 2.1: Diseño de Subsistemas de Soporte

Los subsistemas de soporte serán exactamente los mismos que los mencionados en el punto 1.5 de éste documento, es decir, serán los mismos que los subsistemas de diseño.

### 2.2. DSI 2.2: Identificación de Mecanismos Genéricos de Diseño

Algunos de los mecanismos genéricos de diseño que podemos encontrar en la aplicación son:

- La *recogida de datos* insertados en formularios por un usuario registrado con el fin de realizar alguna transacción. Por ejemplo: realizar un pedido, el envío de un documento,... En el caso de observar algún error en los datos introducidos se avisará al usuario con la correspondiente información de cómo rellenar el campo o los campos que han fallado.
- La *inserción, borrado o actualización* de información en la *BD* será de la misma forma a diferencia de la tabla en cuestión a utilizar.
  - *Inserción*: primero se establece la conexión con el servidor de la BD, se mira que no exista ya esa información en ella, se inserta y se cierra la conexión.
  - *Borrado*: establecimiento de la conexión con el servidor de la BD, ver si existe, borrado y cierre de conexión.
  - *Actualización*: establecer la conexión con el servidor de la BD, ver si existe, actualizar y cerrar la conexión.
- Se establecerán los *mecanismos de concurrencia* necesarios para el acceso a recursos comunes y críticos.
- La *visualización de la información* por al usuario será realizado de la misma manera siempre y el tipo de letra a utilizar será la misma (ya en la implementación deberá especificarse con mayor detalle este punto).
- El *inicio de sesión* para los usuario registrados será comprobado de la misma forma: conexión con el servidor de BD, en la tabla usuarios buscar el nombre de usuario insertado y comprobar que los datos coincidan, redireccionar a la pagina principal si sus datos son correctos y cierre de la conexión. En el caso de datos incorrectos se mostrará un mensaje de error correspondiente.
- El *cierre de sesión* se lleva a cabo de la misma forma comprobando que el usuario esté previamente con una sesión activa.
- La *muestra de información de error* tanto en el inicio de sesión como en la inserción de datos en formularios se realizará de la misma manera.

### 3. DSI 3: Diseño de Casos de Uso Reales

#### 3.1. DSI 3.1: Identificación de Clases Asociadas a un Caso de Uso

Las clases asociadas a los casos de uso son las que se muestran a continuación:

- Clases referentes a los **Gestores**:
  - Clase *Gestor de Correos*: lleva el control de las acciones que puede hacer un usuario registrado o no. Tales acciones son:
    - Para usuarios no registrados: registrarse, obtener información, atención al cliente y solicitar un dominio.
    - Para usuarios registrados: puede hacer las acciones anteriores y además todo lo relacionado con la aplicación, algunas de éstas son: validar o login, logout, envíos, cambio de domicilio, personalización de sellos, creamailing, ....
  - Clase *Gestor de Envíos*: lleva el control de los envíos que puede realizar un cliente registrado. Tales envíos pueden ser: cartas, telegramas, documentos, paquetes, fax y también el seguimiento de los envíos.
  - Clase *Gestor de Filatelia*: controla todo lo relacionado con la Filatelia, como por ejemplo: buscar una sucursal o una asociación dentro de una provincia, buscar información a partir de unos datos dados, realizar un pedido.
  - Clase *Gestor de la Oficina Virtual*: maneja las acciones relacionadas con la oficina virtual de correos, algunas de éstas son: seguimiento de los envíos, ver los últimos envíos, lista de tareas pendientes, solicitudes de justificantes o facturas.
- Clases referentes a las **Interfaces**:
  - Clase *IUPrincipal*: interfaz de usuario a través de la cual se visualiza la página principal de correos.
  - Clase *IUEnvíos*: interfaz de usuario para mostrar todo lo referente a la interfaz de envíos de documentos y paquetes del portal.
  - Clase *IUOficinaVirtual*: interfaz de usuario para mostrar todo lo referente a la oficina virtual de correos.
  - Clase *IUFilatelia*: interfaz de usuario encargada del portal filatélico.
  - Clase *IUCorreos*: interfaz de usuario que nos muestra el contenido referente a la validación (login) y el registro de usuarios.
- Clases referentes a los **Envíos**:
  - Clase *Envío*: clase abstracta de la cual heredan las dos siguientes clases. Se encarga de los diferentes envíos que puede realizar un usuario registrado.
  - Clase *EnvíoDocumento*: clase que hereda de la de *Envío* y que indica que el tipo de envío es un documento. El tipo de documento dependiendo de su valor será una carta, un telegrama o un fax.
  - Clase *EnvíoPaquete*: clase que hereda de la de *Envío* y que indica que el tipo de envío es un paquete y la prioridad o urgencia con la que debe ser tratado dicho paquete.
- Clase *Usuario*: clase en la que se guardan los datos referentes a los usuarios.
- Clase *Información*: clase donde se almacena el tipo y contenido de cierta información de la aplicación.
- Clase *Colecciones*: clase que guardará los datos correspondientes a las colecciones filatélicas.
- Clase *Productos*: clase con la que se hace referencias a los productos del Portal de Correos y los del Portal Filatélico.

- Clase *Pedidos*: clase que referencia a todos los pedidos que los usuarios registrados realizan.
- Clase *Asociación*: clase encargada de almacenar los datos de cada asociación filatélica disponible.
- Clase *Sucursal*: clase en la cual se almacena la información de una sucursal concreta.
- Clase *CentrosCorreos*: clase encargada de obtener información general de la base de datos, como asociaciones filatélicas o sucursales.

### 3.2. DSI 3.2: Diseño de la Realización de los Casos de Uso

Los diagramas de secuencia que nos informan de como se llevan a cabo los distintos casos de uso son los que se muestran a continuación:

- **Login:**

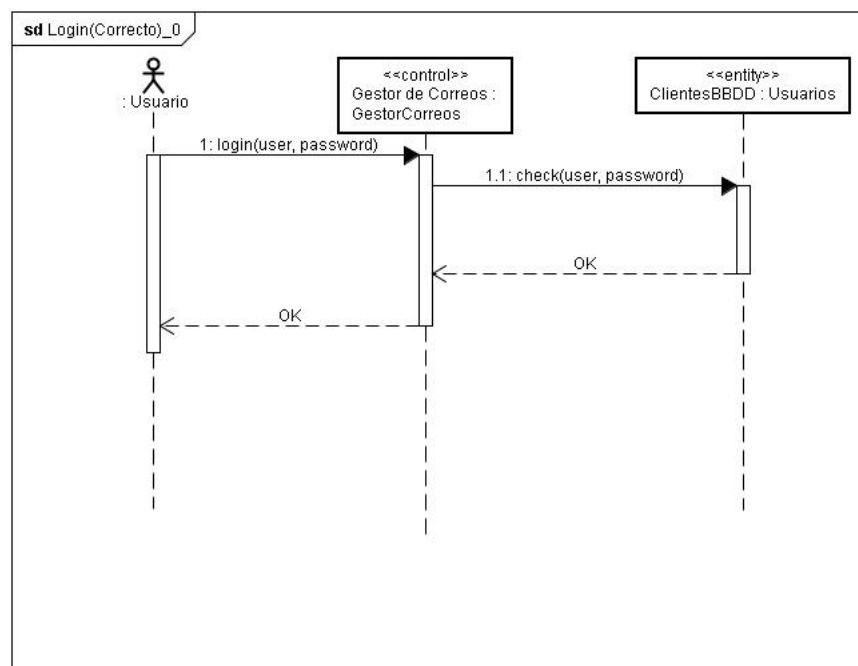


Figura 3: Diagrama de secuencia de login (correcto).

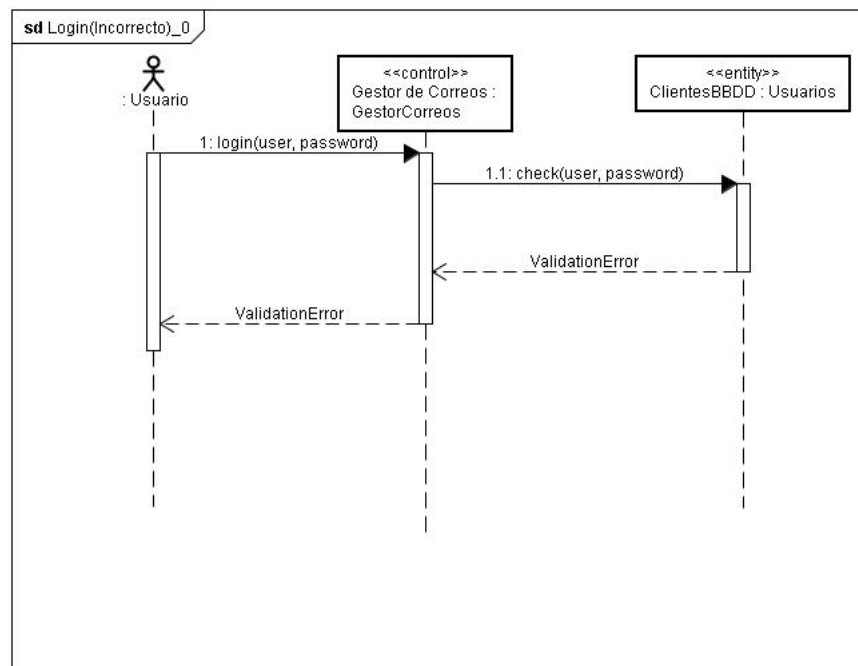


Figura 4: Diagrama de secuencia de login (incorrecto).

Un usuario, previamente registrado, introduce sus datos en la aplicación y pulsa el botón “Entrar”, entonces el Gestor de Correos comprueba los datos mediante una consulta a la base de datos y hay dos posibilidades:

- Datos Correctos: la BD y el gestor de correos devuelven “OK”. Si todo va bien la aplicación redirige al usuario a la pantalla principal de usuarios registrados.
- Datos Incorrectos: la BD y el gestor de correos devuelven “ValidationError”. Por lo que en este caso se le mostrará al usuario un mensaje semejante a: su nombre o contraseña no son correctos.

■ **Logout:**

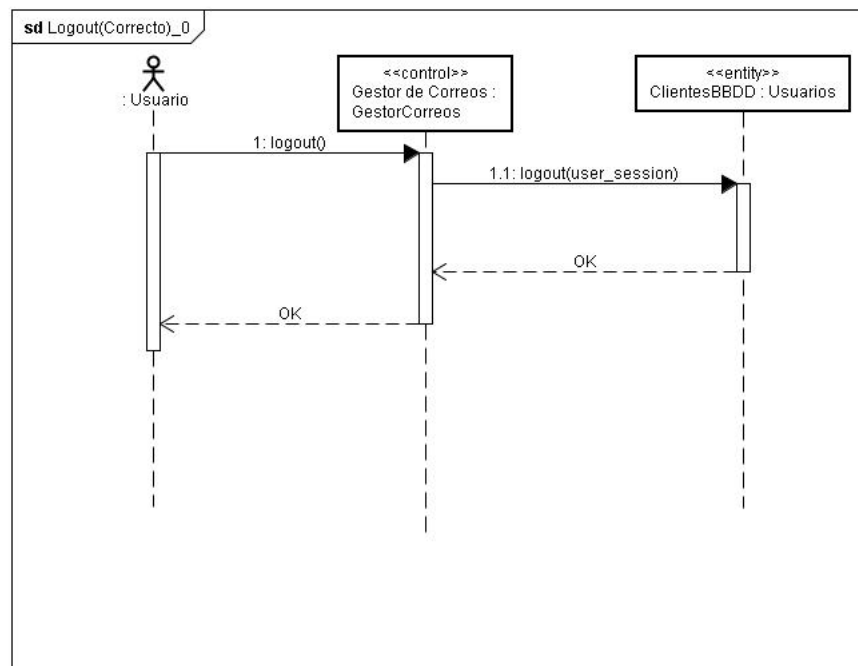


Figura 5: Diagrama de secuencia de logout (correcto).

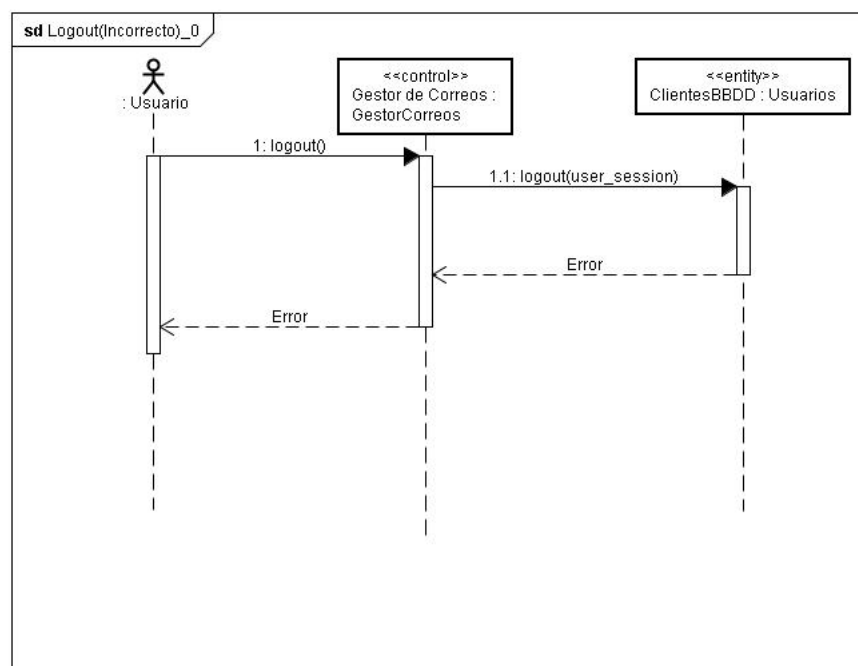


Figura 6: Diagrama de secuencia de logout (incorrecto).

Cuando el usuario considere oportuno podrá cerrar la sesión pulsando en sobre el enlace correspondiente (ej: “Cerrar Sesión”, entonces el gestor de correos recoge la información y mediante una consulta a la base de datos podrán ocurrir las siguientes situaciones:

- El usuario tiene una sesión activa y desea cerrarla: la BD y el gestor de correos devuelven “OK”, por lo que redirige al usuario a la pagina principal de la oficina virtual, donde podrá volver a entrar.
  - El usuario no tiene una sesión activa y desea cerrarla: la BD y el gestor de correos devuelven “Error”. No se muestra un mensaje ya que se redirige al usuario a la pagina principal de acceso al cliente.
- **Registro:**

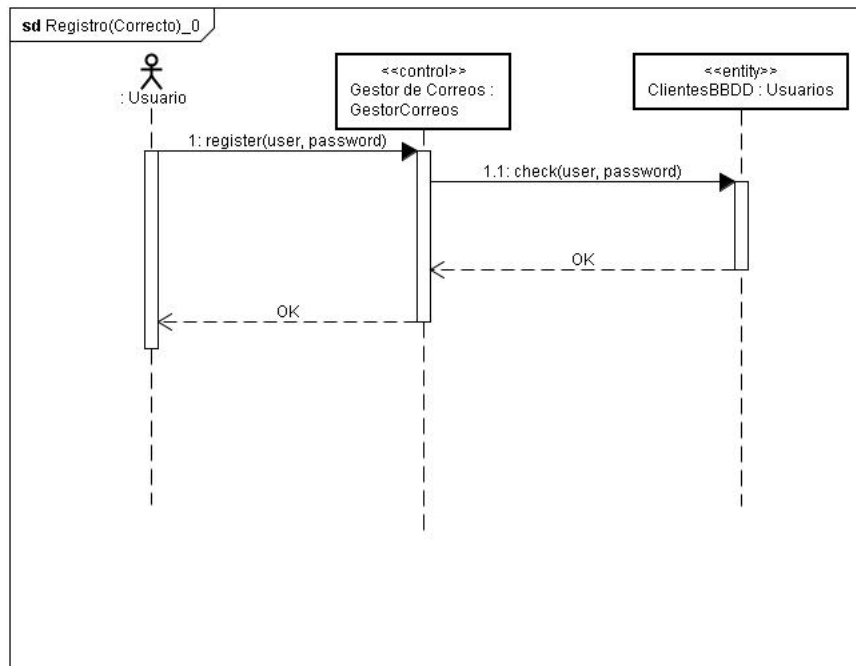


Figura 7: Diagrama de secuencia de registro (correcto).



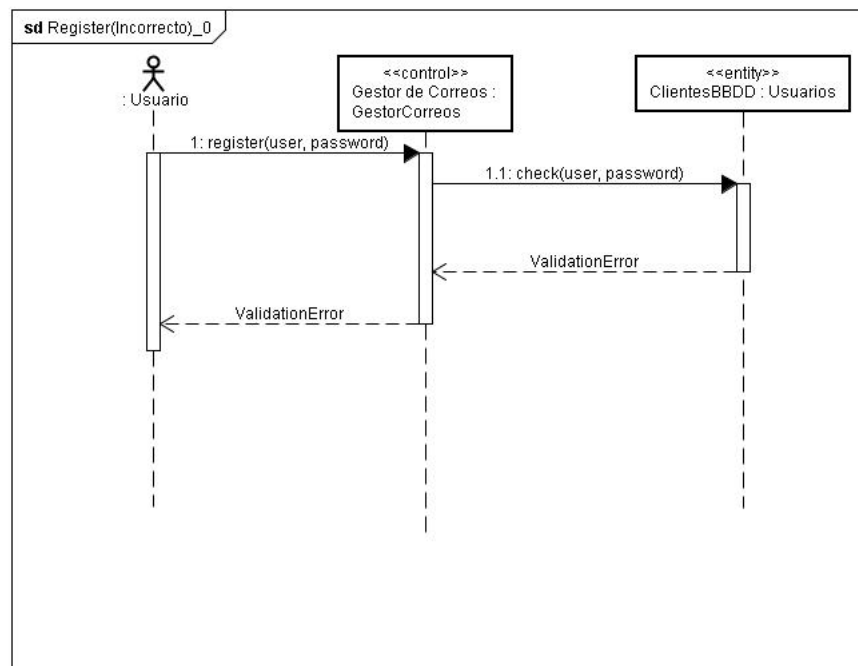


Figura 8: Diagrama de secuencia de registro (incorrecto).

El usuario sin registrar deberá rellenar el formulario con sus datos, cuando pulsa el botón “Aceptar” el gestor de correos recoge dicha información y a través de una consulta a la base de datos podrá ocurrir lo siguiente:

- Si la base de datos devuelve “OK” entonces se inserta el nuevo usuario en la BD.
- Si la base de datos devuelve “ValidationError” entonces no se inserta el nuevo usuario en la BD, debido a que el nombre de usuario ya existe. Por lo que se le muestra un mensaje de error.

■ **Información:**

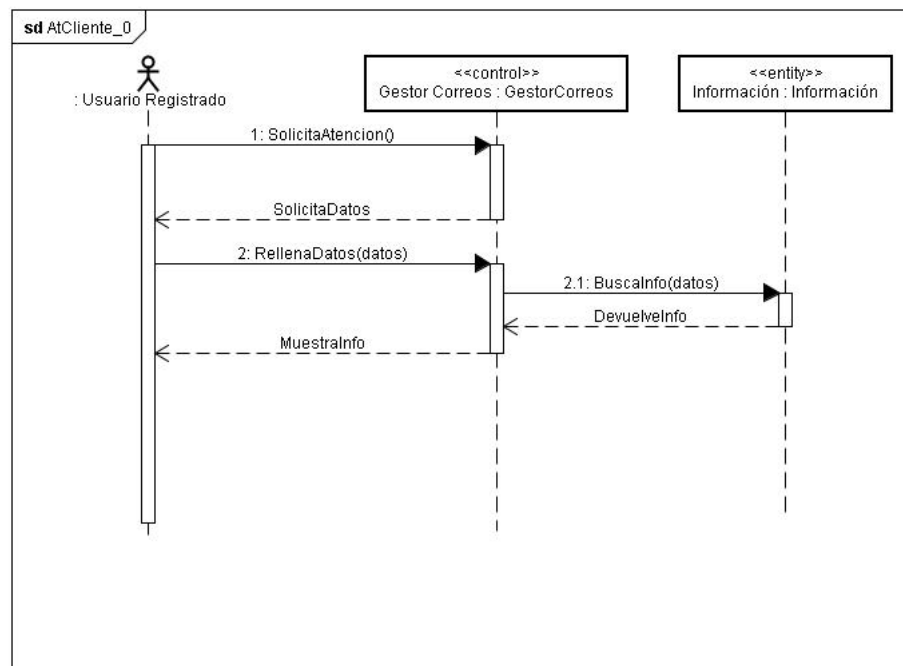


Figura 9: Diagrama de secuencia de atención al cliente genérica.

Un usuario registrado pide atención a la aplicación, mediante el gestor de correos se le piden una serie de datos, después de rellenar el formulario el gestor busca, a partir de la información que recoge del formulario, en la base de datos y ésta devuelve la información correspondiente que posteriormente le será mostrada.

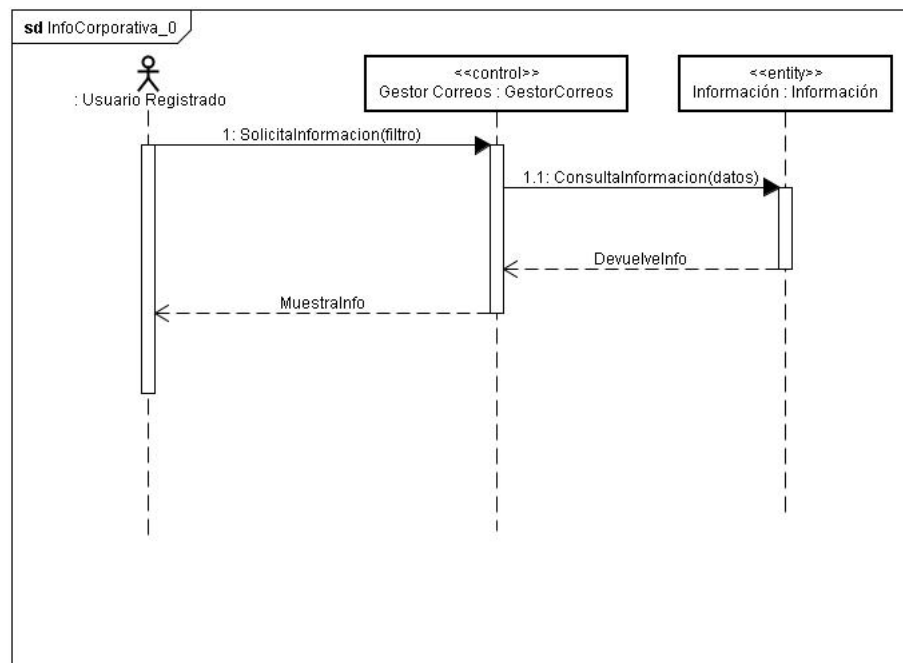


Figura 10: Diagrama de secuencia información corporativa.

Un usuario registrado solicita información a la aplicación, el gestor de correos mediante una consulta a la base de datos le devuelve la información correspondiente a la pedida.

■ **Oficina Virtual:**

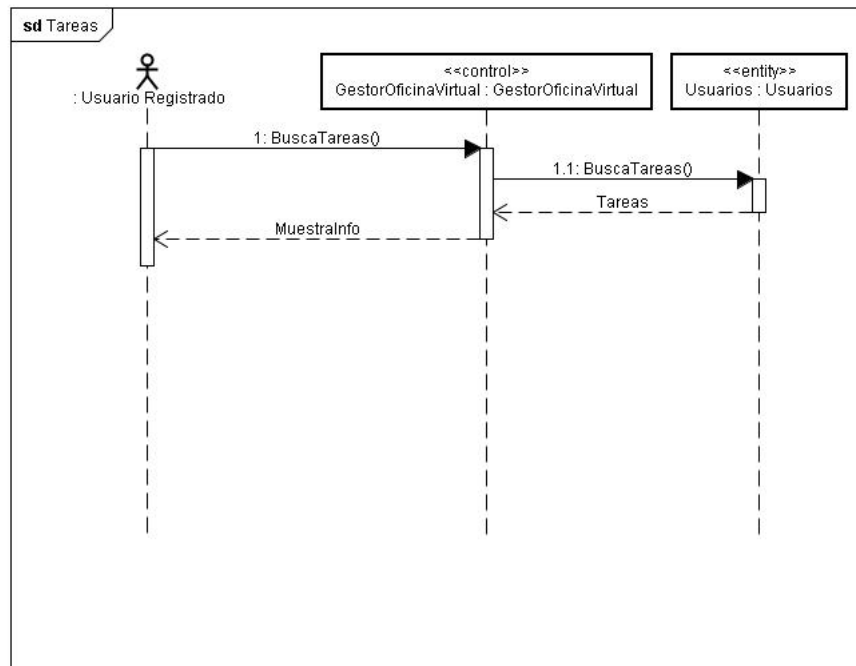


Figura 11: Diagrama de secuencia de tareas pendientes.

El usuario que previamente se ha registrado pincha en el enlace de tareas, el gestor de la oficina virtual consulta a la base de datos sobre las tareas de ese usuario y ésta devuelve sus tareas y el gestor le muestra la información.

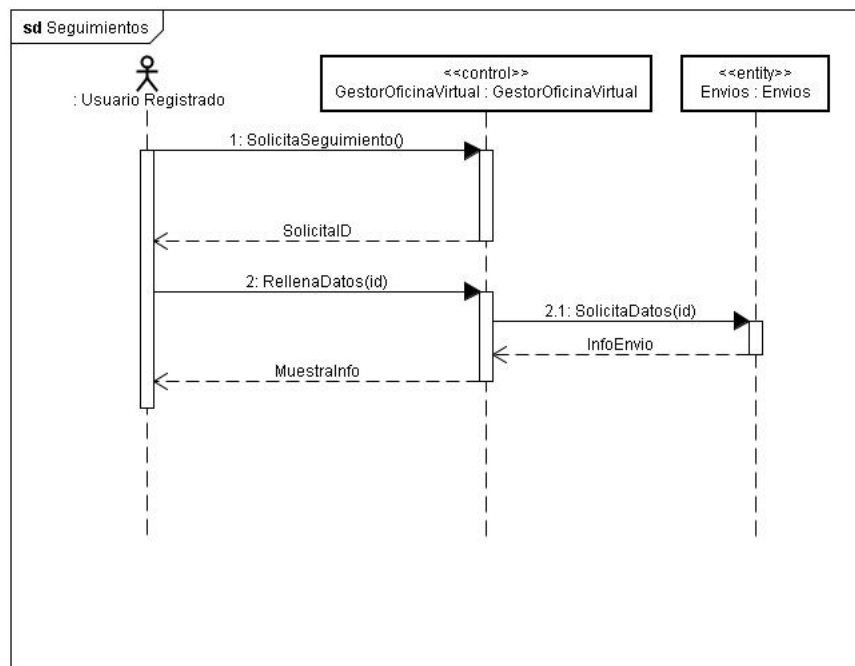


Figura 12: Diagrama de secuencia de seguimiento de envíos.

Cuando un usuario registrado solicita el seguimiento de un envío, el gestor de la oficina virtual le pide el identificador y una vez que se lo proporciona el gestor solicita la información a la base de datos, la cual devolverá los datos correspondientes a ese envío y el gestor le mostrará al usuario la información.

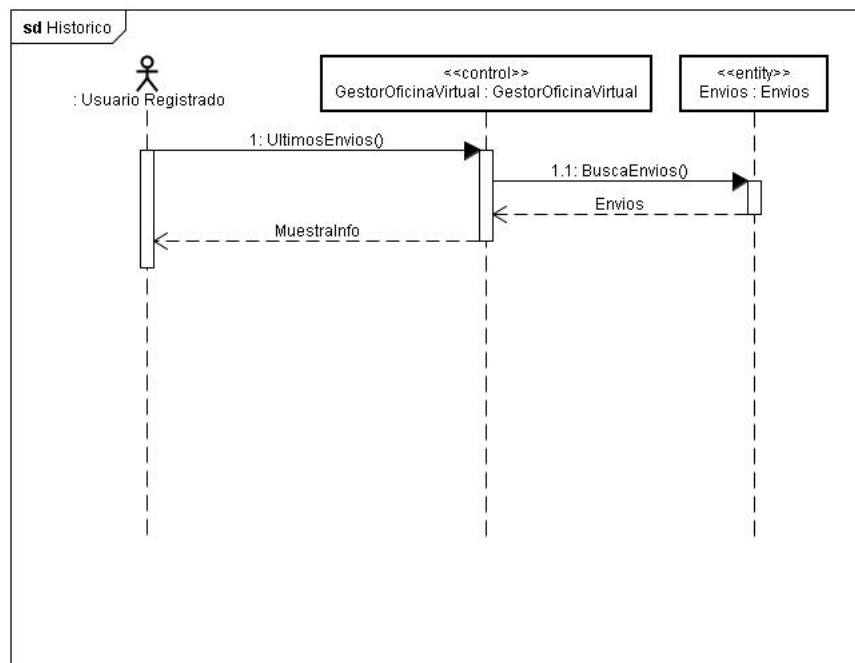


Figura 13: Diagrama de secuencia de historico de envíos.

El cliente (usuario registrado) pide la información de los últimos envíos que realizó, el gestor de la oficina virtual consulta en la base de datos sus últimos envíos, la BD le devuelve esos envíos y el gestor le muestra la información al usuario.

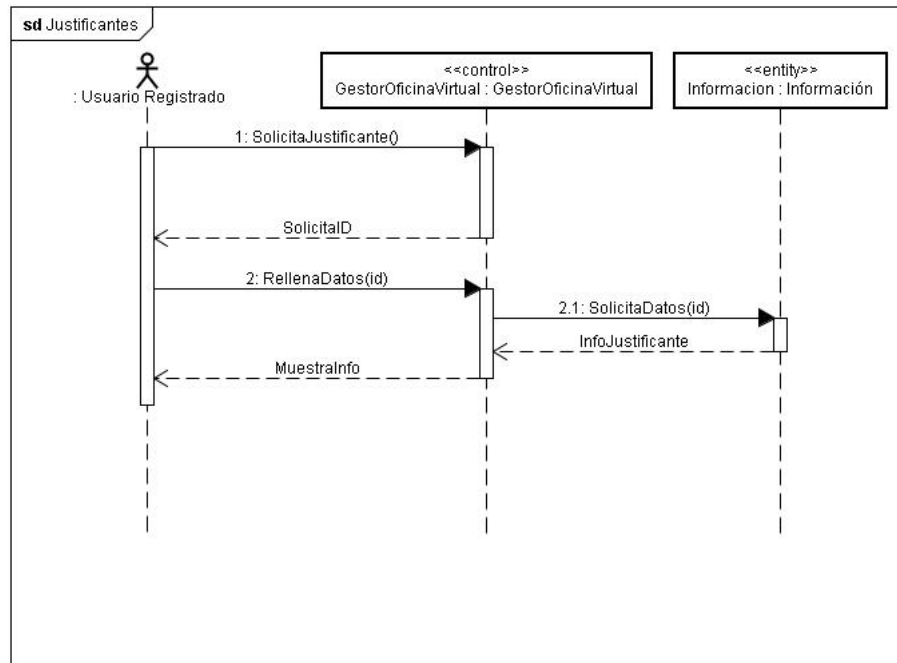


Figura 14: Diagrama de secuencia de obtención de justificantes.

Cuando un usuario registrado necesita el justificante o factura de un envío o pedido pinchará en su enlace correspondiente, el gestor de la oficina virtual le pedirá que introduzca el identificador, una vez rellenado el gestor recoge esos datos para hacer una consulta en la base de datos y ésta devolverá la información del justificantes y el gestor le mostrará al usuario dicha información.

■ **Filatelía:**

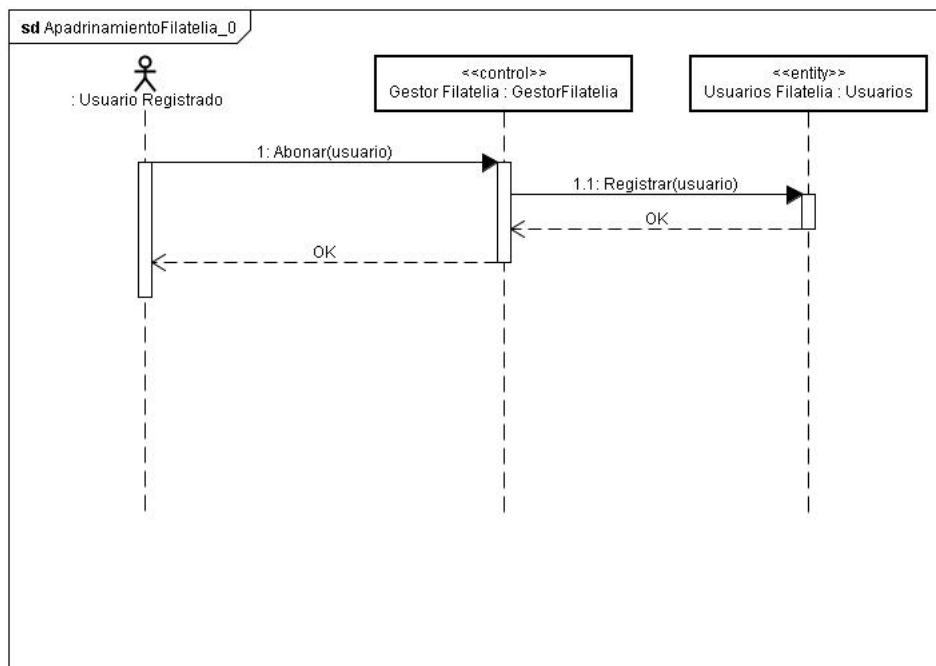


Figura 15: Diagrama de secuencia de apadrinamiento.

El usuario registrado quiere realizar la acción de apadrinar debe abonar una cantidad, el gestor de filatelia registra al usuario en la filatelia y la base de datos devuelve “OK”, por lo que el gestor le muestra “OK” al usuario.

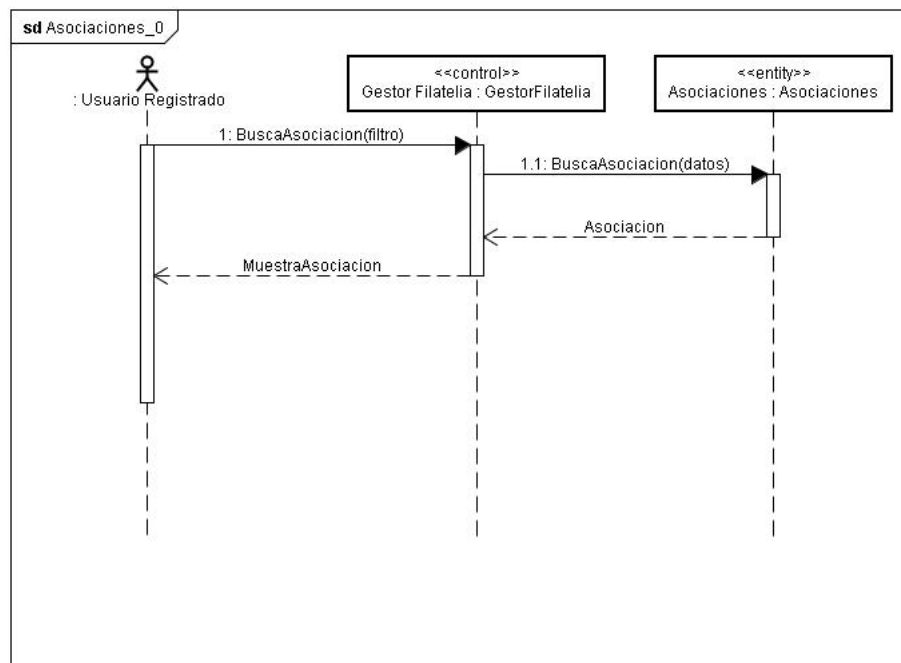


Figura 16: Diagrama de secuencia de asociaciones.

El usuario registrado busca una asociación filatélica, el gestor de la filatelia busca dicha asociación en la base de datos y ésta devuelve esa asociación que será mostrada al usuario en pantalla mediante el previo tratamiento de la información por el gestor.

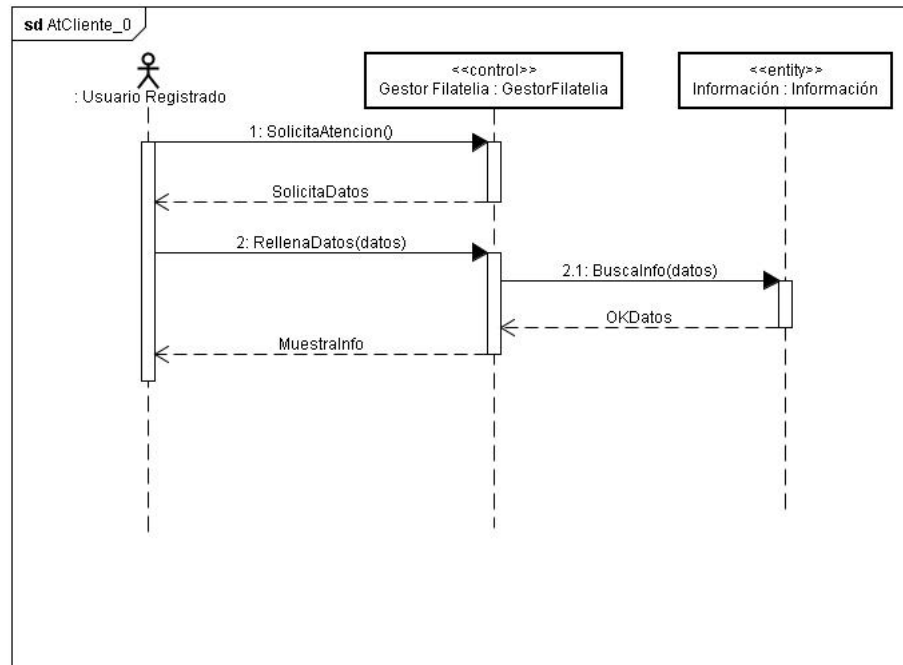


Figura 17: Diagrama de secuencia de atención al cliente.

El usuario registrado solicita la atención a la aplicación, el gestor de la filatelia le pide que ingrese unos datos, una vez rellenados, el gestor busca dichos datos en la base de datos y ésta devuelve “OK” con los datos y el gestor muestra la información en pantalla.

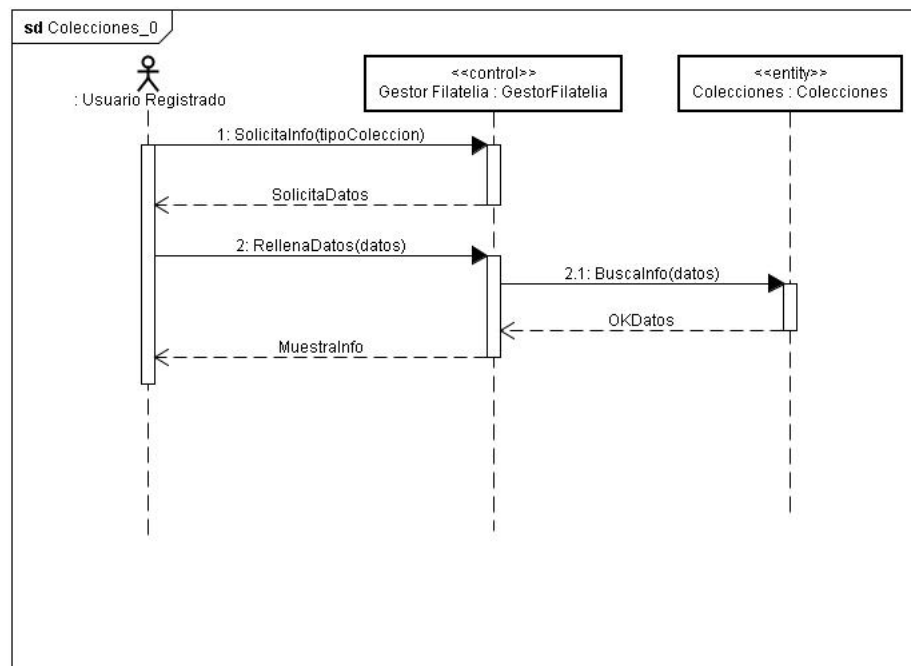


Figura 18: Diagrama de secuencia de colecciones filatélicas.

Un usuario registrado solicita la información de un tipo de colección filatélica, el gestor de filatelia pide una serie de datos al usuario, que una vez rellenos busca en la base de datos la información, ésta le devuelve “OK” con los datos y el gestor le muestra al usuario la información por pantalla.



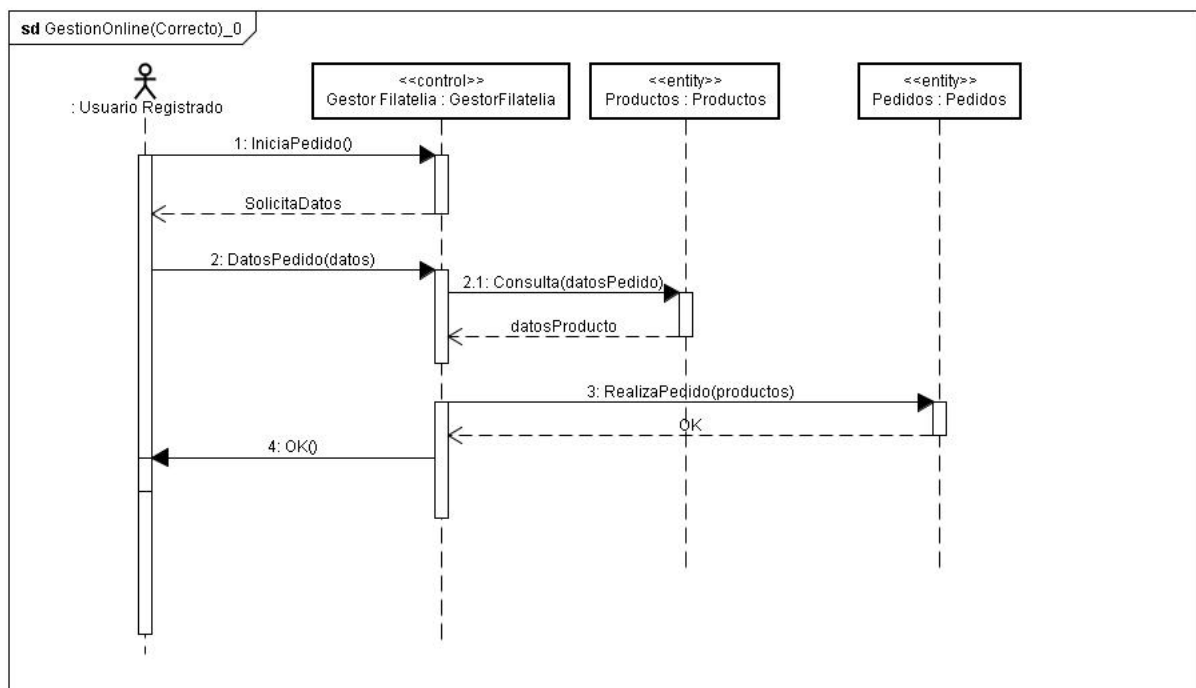


Figura 19: Diagrama de secuencia de gestión online (correcto).

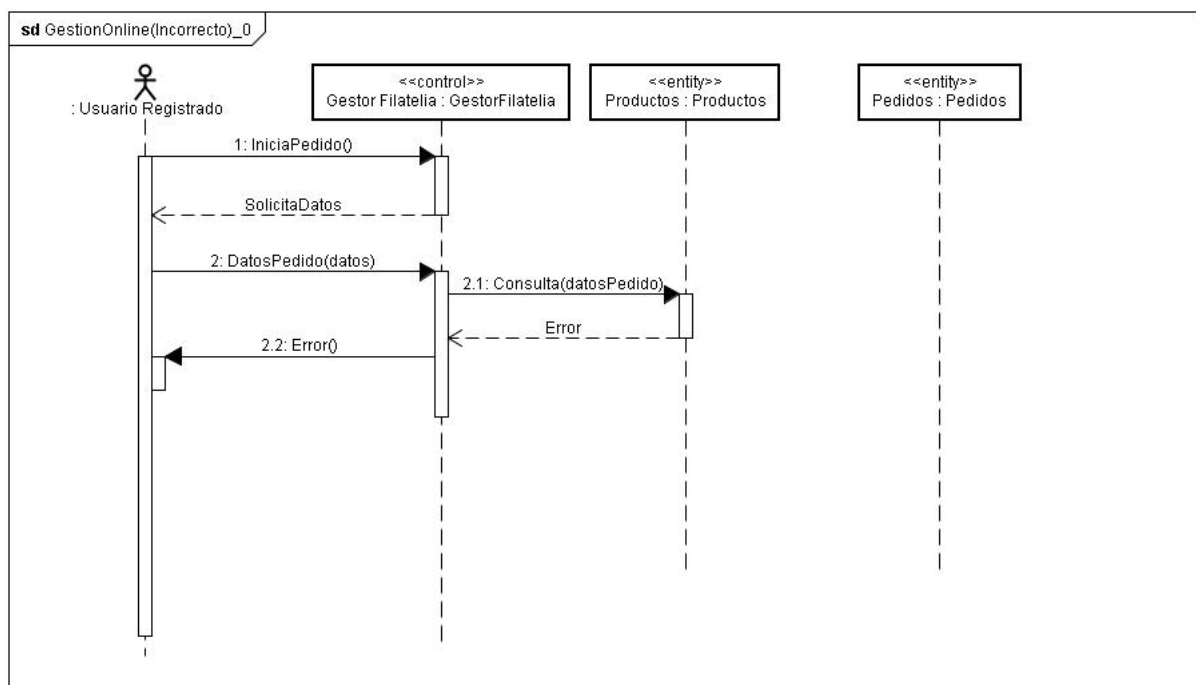


Figura 20: Diagrama de secuencia de gestión online (incorrecto).

El usuario registrado desea realizar un pedido sobre un producto, el gestor de filatelia le solicita el ingreso de unos datos, cuando el usuario ingresa esos datos, el gestor comprueba con la base de datos que ese producto existe y devuelve la información del producto, y a partir de dicha información se realiza el pedido, ingresando en la base de datos la información que corresponde. En el caso de que el producto no exista no se podrá hacer efectivo el pedido de dicho producto.

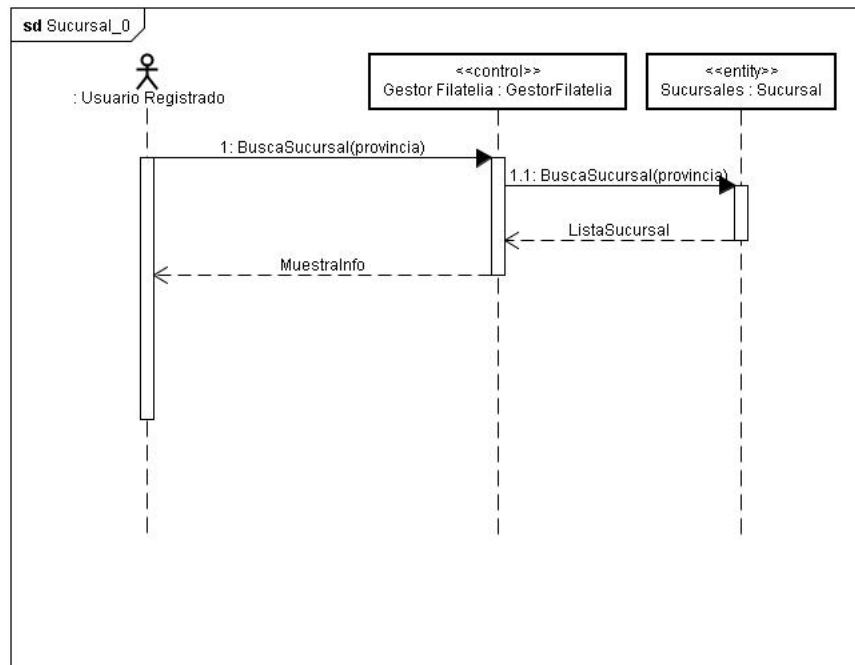


Figura 21: Diagrama de secuencia de búsqueda de sucursales.

El usuario registrado busca sucursales de una provincia, el gestor de la filatelia busca en la base de datos la lista de sucursales de esa provincia y una vez devuelto el resultado se le muestra la información por pantalla al usuario.

#### ■ Envíos

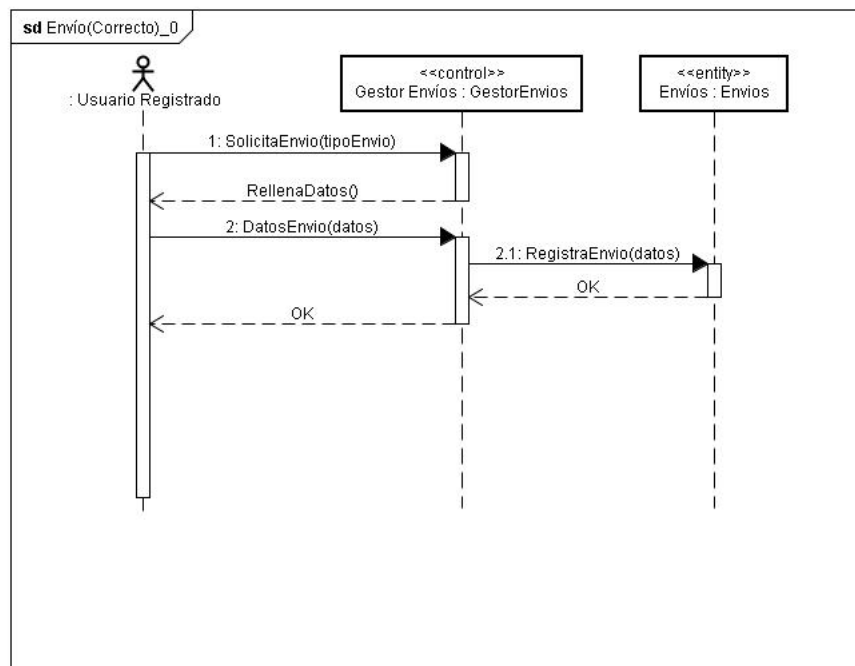


Figura 22: Diagrama de secuencia de envío (correcto).

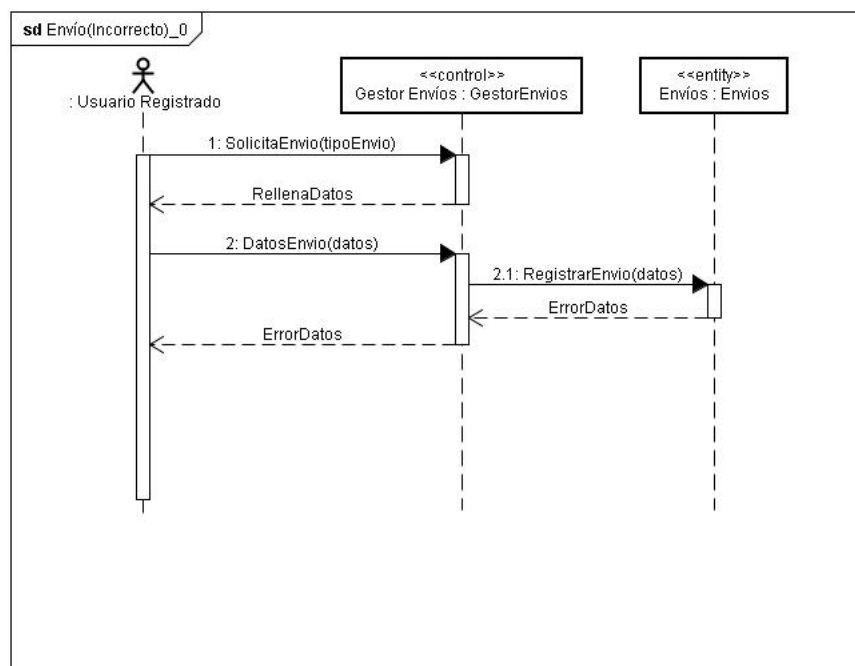


Figura 23: Diagrama de secuencia de envío (incorrecto).

El usuario registrado desea realizar un envío, el gestor de envíos le solicita la información del envío, el cliente ingresa los datos correspondientes y pueden suceder dos situaciones:

- El gestor intenta registrar los datos del envío en la base de datos y ésta le responde con “OK”,

por lo que el gestor devuelve un “OK” al usuario.

- El gestor intenta registrar los datos del envío en la base de datos y ésta le responde con “ErrorDatos”, por lo que el gestor devuelve un “ErrorDatos” al usuario con el motivo de que se corrijan esos datos incorrectos.

### 3.3. DSI 3.3: Revisión de la Interfaz de Usuario

La interfaz de usuario está estructuradas en los tres perfiles de usuarios existentes en el sistema, que son:

- Interfaz para el perfil de Cliente Particular:

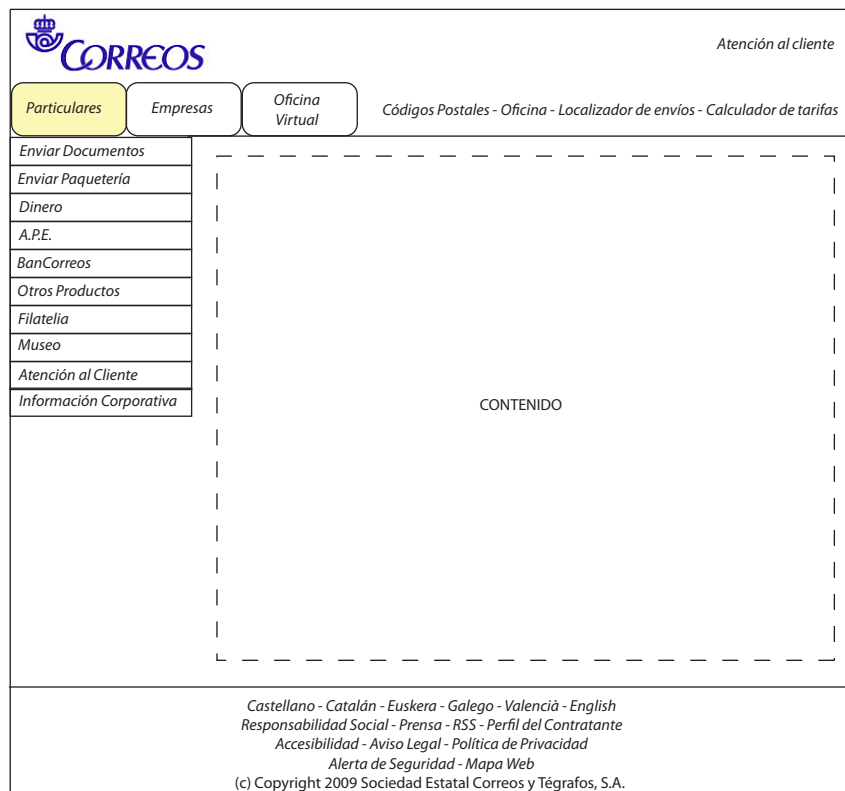


Figura 24: Captura de la interfaz principal de particulares.

Como podemos ver en la captura anterior, la estructura de la interfaz para particulares sigue el esquema descrito. El menú a la izquierda permite al usuario acceder a todas las opciones disponibles con un sólo click, y se encuentra ordenado por la frecuencia de uso de cada una de las opciones. Cada evento vendrá representado por un click con el ratón en el menú elegido, que llevará a la modificación del contenido de la interfaz. Este contenido dependerá de la opción elegida, por lo que no se puede tratar de manera estática.

- Interfaz para el perfil de Cliente Empresa:

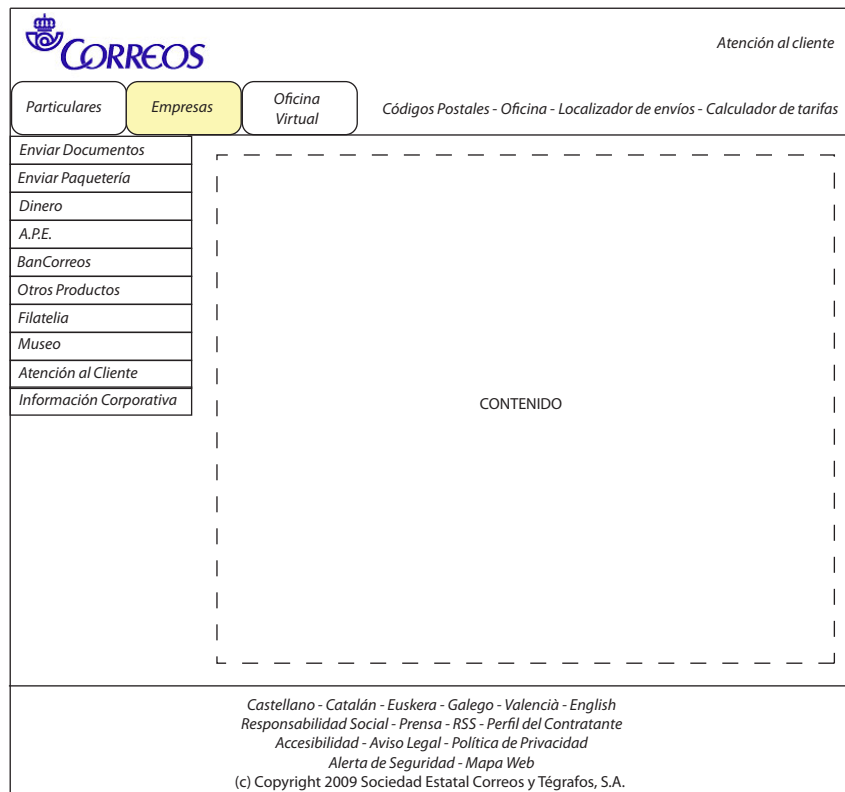


Figura 25: Captura de la interfaz principal de empresas.

En este caso, las opciones disponibles son las de empresas, pero la estructura global de la pantalla es la misma. Los eventos se gestionan de la misma forma, al recibir clicks de ratón.

- Intrefaz para la Oficina Virtual, en la que el usuario deberá estar registrado para acceder a ella:

Figura 26: Captura de la interfaz principal de oficina virtual.

Para la oficina virtual la estructura es la misma pero con las opciones propias de la oficina. Gracias a esto, el portal mantiene un aspecto homogéneo que no intimida al usuario ni lo desconcierta.

Los *formatos de impresión* serán los mismos para todos los documentos, con el logo de la empresa en la parte superior, junto con el título del documento y el número de página. Los documentos disponibles para la impresión son los siguientes:

- **Justificantes:** Los justificantes de las facturas pagadas incluirán al comienzo el número y fecha de facturación, y tras esto, la información referente a la factura.
- **Seguimiento de envíos:** Los seguimientos de envíos figurarán en forma de tabla, dividida por las siguientes columnas: fecha, estado. De esta forma el usuario identifica de forma clara los estados por lo que ha ido pasando el envío.
- **Producto filatélico:** La impresión de información de un producto filatélico se dispondrá con una imagen del mismo que ocupará el cuarto superior izquierdo del documento, junto con la información del documento debajo del mismo, ocupando el resto del mismo.
- **Facturas:** Las facturas serán todas similares en el aspecto. Contendrán el número de factura a la izquierda y a la derecha el nombre del titular de la factura. A continuación aparecerá una tabla dividida por las columnas de *descripción* y *precio*. El contenido de esa tabla será la descripción de cada compra y el precio asociado a la misma, a razón de una por línea.

### 3.4. DSI 3.4: Revisión de Subsistemas de Diseño e Interfaces

Las diferentes clases, actores y mensajes que intervienen en cada uno de los casos de uso son:

- Casos de Uso **Login, Logout, Registro:**

- *Clases* que intervienen en los casos de uso:
  - GestorCorreos.
  - IUCorreos.
  - Usuario.
- *Actores* que intervienen en el caso de uso:
  - Usuario Registrado.
  - Usuario No Registrado.
- *Mensajes* entre el usuario y el sistema:
  - Validación Correcta: Si el usuario introduce los datos correctos el sistema le devolverá "OK".
  - Validación Incorrecta: Si el usuario introduce los datos que no se encuentran en la base de datos el sistema le devolverá "ValidationError".
  - Cierre Sesión Correcto: el usuario tiene una sesión activa y la cierra, entonces el sistema le devuelve "OK".
  - Cierre Sesión Incorrecto: el usuario tiene una sesión inactiva y quiere cerrarla, entonces el sistema le devuelve "Error".
  - Registro Correcto: el usuario inserta los datos en el formulario correctamente y el sistema le devuelve "OK".
  - Registro Incorrecto: el usuario inserta los datos en el formulario de forma errónea y el sistema le devuelve "ValidationError". El motivo de esta situación será la existencia de ese usuario ya en la base de datos.
- Caso de Uso **Oficina Virtual**:
  - *Clases* que intervienen en el caso de uso:
    - GestorOficinaVirtual.
    - IUOficinaVirtual.
  - *Actor* que interviene en el caso de uso:
    - Usuario Registrado.
  - *Mensajes* entre el usuario y el sistema:
    - Tareas Pendientes: el usuario solicita las tareas que tiene pendientes y el gestor de la oficina virtual realiza una consulta a la base de datos, la cual le devuelve un mensaje con la información de estas tareas que tiene sin realizar.
    - Seguimiento Envíos: el usuario solicita el seguimiento de un envío para lo que inserta un identificador, con el cual el gestor de la oficina virtual busca en la base de datos y le devuelve un mensaje con la información del envío que pedía.
    - Histórico: el usuario solicita la información de los últimos envíos, el gestor de la oficina virtual hace una consulta a la base de datos y ésta le devuelve un mensaje con todos los envíos que este usuario ha realizado.
    - Justificantes: el usuario solicita la información de un justificante para lo cual proporciona un identificador, y con éste el gestor de la oficina virtual busca en la base de datos y le devuelve un mensaje con la información del justificante que buscaba.
- Caso de Uso **Envíos**:
  - *Clases* que intervienen en el caso de uso:
    - GestorEnvíos.
    - IUEnvíos.
    - Envíos.
    - EnvíoDocumento o EnvíoPaquete: dependiendo de si lo que quiere enviar un usuario es un documento (carta, telegrama o fax) o si quiere enviar un paquete con una cierta urgencia.

- *Actor* que interviene en el caso de uso:
  - Usuario Registrado.
- *Mensajes* entre el usuario y el sistema:
  - Envío Correcto: el usuario rellena los datos de forma correcta por lo que el sistema le devuelve “OK”.
  - Envío Incorrecto: el usuario introduce algún error en el formulario por lo que el sistema le devuelve “ErrorDatos”.
- Caso de Uso **Filatelía**:
  - *Clases* que intervienen en el caso de uso:
    - GestorFilatelía.
    - IUFilatelía.
    - CentrosCorreos.
    - Sucursal.
    - Asociación.
    - Colecciones.
    - Pedidos.
    - Productos.
    - Información.
  - *Actores* que intervienen en el caso de uso:
    - Usuario Registrado.
    - Usuario No Registrado.
  - *Mensajes* entre el usuario y el sistema:
    - El usuario no registrado sólo podrá realizar un registro en el sistema para poder acceder al Portal Filatélico. En el caso de registro correcto el sistema le devuelve “OK” sino le devolverá “ValidationError”.
    - Apadrinamiento: el usuario registrado abonará una cantidad de dinero y el gestor de filatelía registrará al usuario en los usuarios Filatelía de la base de datos, la cual le devolverá un mensaje de “OK”.
    - Colecciones: el cliente solicita la información de las colecciones filatélicas y el gestor de filatelía hace una consulta sobre la tabla de colecciones y le devuelve un mensaje con los datos solicitados.
    - Asociaciones: el usuario registrado pide información de una asociación filatélica y el gestor de filatelía busca en la tabla de asociaciones y devuelve un mensaje con la información solicitada.
    - Atención al Cliente: el usuario necesita atención y el gestor de filatelía le pide unos datos y busca en la tabla de información según esos datos y le devuelve un mensaje con dicha información.
    - Búsqueda Sucursal: el cliente solicita la información de una sucursal de una provincia y el gestor de filatelía busca en la base de datos esa sucursal y le devuelve un mensaje con dicha información.
    - Realizar Pedido: el usuario quiere realizar el pedido de un producto, inserta los datos, el gestor de la filatelía busca en la tabla de productos y si encuentra el producto realiza el pedido y devuelve un mensaje de “OK”, sino devuelve un mensaje de “Error”.
- Caso de Uso **Gestión**:
  - *Clases* que intervienen en el caso de uso:
    - GestorOficinaVirtual.
    - GestorCorreos.



- IUPrincipal.
- Información.
- *Actores* que intervienen en el caso de uso:
  - Usuario Registrado.
  - Usuario No Registrado.
- *Mensajes* entre el usuario y el sistema:
  - El usuario no registrado solo puede solicitar información y consulta de dominios a la aplicación. La información a la cual podrá acceder es Museos, Recibos, Recarga de Móviles, Locutorios y Tiendas Telecom. El sistema (gestor de correos) dependiendo de la consulta que quiera realizar el usuario así buscará en la base de datos dicha información y se la enviará al usuario para mostrársela por pantalla.
  - Cambio de Domicilio: el usuario solicita cambiar su domicilio, por lo que el gestor de la oficina virtual le solicita los nuevos datos y actualiza la tabla de usuarios de la base de datos y el sistema le devuelve un mensaje de "OK".
  - Personalización de Sellos: el usuario desea personalizar un sello, el gestor de la oficina virtual le pide los datos y la imagen del sello y una vez introducidos dichos datos, el sistema le envía un mensaje de "OK".
  - Sobres y embalajes: el usuario solicita el servicio y el gestor de la oficina virtual le responde con el mensaje de "OK".
  - Creamailing: el usuario solicita este servicio y el gestor de la oficina virtual le responde con el mensaje de "OK".
  - Apartado Postal: el usuario solicita este servicio y el gestor de la oficina virtual le responde con el mensaje de "OK".
  - Gestión E-mail: el usuario solicita este servicio y el gestor de la oficina virtual le responde con el mensaje de "OK".
- Caso de Uso **Información**:
  - *Clases* que intervienen en el caso de uso:
    - GestorCorreos.
    - IUPrincipal.
    - Información.
  - *Actores* que intervienen en el caso de uso:
    - Usuario Registrado.
  - *Mensajes* entre el usuario y el sistema:
    - El usuario solicita información el gestor de correos la busca en la base de datos y compone un mensaje de información para mostrársela al usuario por pantalla.

## 4. DSI 4: Diseño de Clases

### 4.1. DSI 4.1: Identificación de Clases Adicionales

Las clases de control, de identidad y de interfaz son las siguientes:

- **Clases de Control:** GestorCorreos, GestorEnvios, GestorFilatelia y GestorOficinaVirtual.
- **Clases de Entidad:** Envío (clase abstracta), EnvíoDocumento, EnvíoPaquete, Usuario, Información, Asociación, Sucursal, CentrosCorreos, Colecciones, Productos y Pedidos.
- **Clases de Interfaz:** IUPrincipal, IUEnvíos, IUOficinaVirtual, IUFilatelia e IUCorreos.

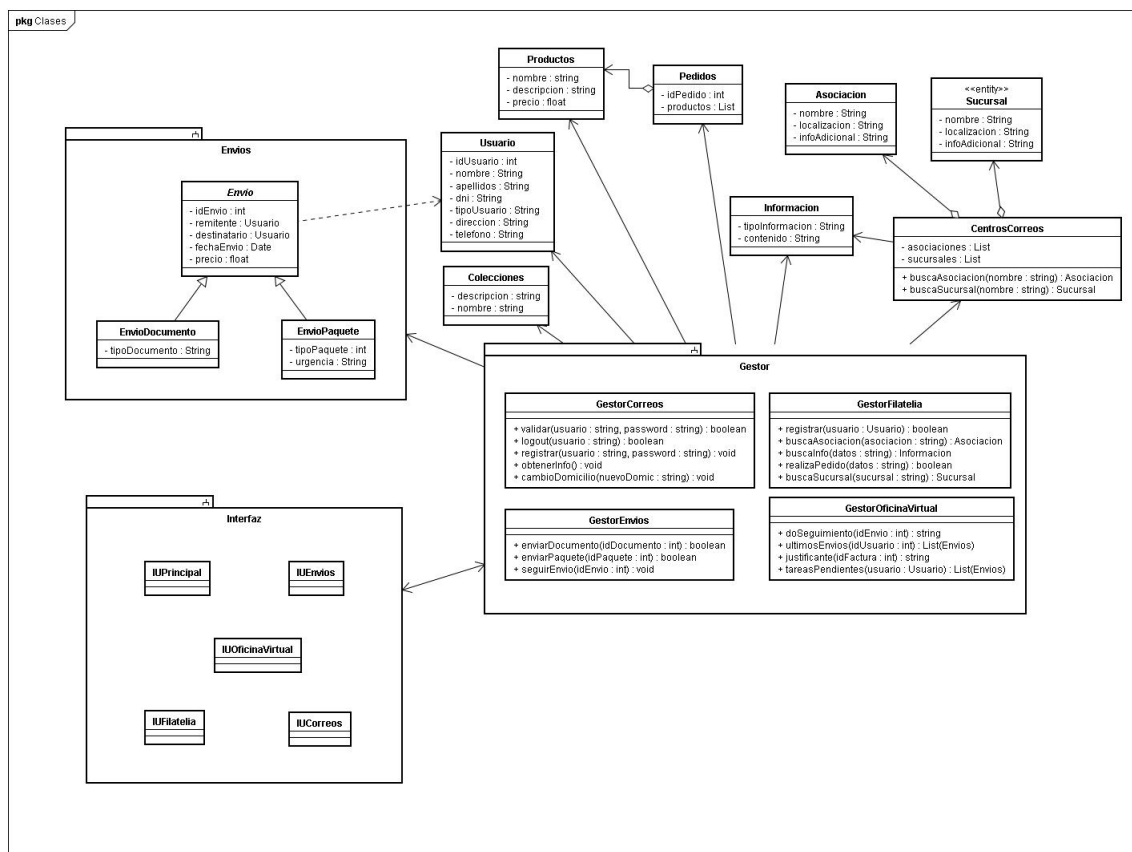


Figura 27: Diagrama de clases del sistema.

### 4.2. DSI 4.2: Diseño de Asociaciones y Agregaciones

Mediante el seguimiento del diagrama de clases anterior podemos identificar las siguientes relaciones:

- **Agregación:**
  - *Asociación, Sucursal y CentrosCorreos*: la relación que existe entre estas clases es de agregación ya que la clase de CentrosCorreos está formada por dos listas diferentes de asociaciones y sucursales. No es una relación de composición debido a que la clase CentrosCorreos sigue existiendo con funciones propias en caso de que no existieran asociaciones ni sucursales.
  - *Productos y Pedidos*: la relación que establecen estas clases es de agregación debido a que la clase Pedidos está formada por un identificador y por una lista de productos que se solicitaron cuando se realizó el pedido. La relación no es de composición porque la clase Pedidos existe aunque en haya ningún producto.

**■ Otras relaciones:**

- *Gestor - Interfaz*: las clases del paquete *Interfaz* tan sólo se relacionan con las del paquete *Gestor* debido a que son estas últimas las encargadas de gestionar la información que va a mostrar la interfaz. Por lo tanto, la interfaz se va a relacionar con los gestores para obtener la información que deben mostrar por pantalla.
- *Gestor - Envíos*: las clases del paquete *Gestor* se relacionan con las del paquete *Envíos* para obtener y procesar información de los envíos registrados, así como para registrar nuevos envíos. Por esto es una relación unidireccional, ya que tan sólo los gestores accederán a la información de los envíos.
- *Gestor - Usuario*: los gestores se relacionan con la clase de usuario para obtener información de cualquier usuario así como para crear y/o eliminar usuarios. Es una relación unidireccional.
- *Gestor - Información*: los gestores se relacionan con la clase *Información* para obtener información de forma que las interfaces puedan entenderla. Es una relación unidireccional.
- *Gestor - CentrosCorreos*: los gestores se relacionan con la clase *CentrosCorreos* ya que esta última es la encargada de buscar sucursales o asociaciones filatélicas, así como información de las mismas. Se trata de una relación unidireccional, ya que *CentrosCorreos* no necesita acceder para nada a los gestores.
- *Gestor - Colecciones*: los gestores se relacionan con la clase *Colecciones* cuando éstos necesitan obtener información de dicha clase. Se trata de una relación unidireccional.
- *Gestor - Pedidos*: los gestores se relacionan con la clase *Pedidos* cuando necesitan obtener información acerca de los pedidos que ha realizado cierto usuario. Se trata de una relación unidireccional.
- *Gestor - Productos*: los gestores se relacionan con la clase *Productos* cuando necesitan obtener una información sobre un conjunto de producto o sobre un producto concreto. Se trata de una relación unidireccional.
- *Envíos - Usuario*: Los envíos deben relacionarse con la clase *Usuario* ya que almacenan dos usuarios en cada envío: el remitente y el destinatario. Es una relación unidireccional.

**4.3. DSI 4.3: Identificación de Atributos de las Clases**

Los atributos de cada una de las clase son:

**■ Clase Envíos:**

- *idEnvío*: tipo Entero.
- *remitente*: tipo de la clase Usuario.
- *destinatario*: tipo de la clase Usuario.
- *fechaEnvío*: tipo Date.
- *precio*: tipo Float.

**■ Clase EnvíoDocumento:**

- *tipoDocumento*: tipo String. Podrá ser una carta, un telegrama o un fax.

**■ Clase EnvíoPaquete:**

- *tipoPaquete*: tipo String.
- *urgencia*: tipo String. Podrá ser alta, media o baja.

**■ Clase Usuario:**

- *idUsuario*: tipo Entero.
- *nombre*: tipo String.

- *apellidos*: tipo String.
- *dni*: tipo String.
- *tipoUsuario*: tipo String. Podrá ser Particular o Empresa.
- *dirección*: tipo String.
- *teléfono*: tipo String.
- Clase **Colecciones**:
  - *nombre*: tipo String.
  - *descripción*: tipo String.
- Clase **Productos**:
  - *nombre*: tipo String.
  - *descripción*: tipo String.
  - *precio*: tipo Float.
- Clase **Pedidos**:
  - *idPedido*: tipo Entero.
  - *productos*: tipo Lista de la clase Productos.
- Clase **Asociación**:
  - *nombre*: tipo String.
  - *localización*: tipo String.
  - *infoAdicional*: tipo String.
- Clase **Sucursal**:
  - *nombre*: tipo String.
  - *localización*: tipo String.
  - *infoAdicional*: tipo String.
- Clase **CentrosCorreos**:
  - *asociaciones*: tipo Lista de la clase Asociación.
  - *sucursales*: tipo Lista de la clase Sucursal.
- Clase **Información**:
  - *tipoInformación*: tipo String.
  - *contenido*: tipo String.

Las clases que no aparecen en esta lista es porque no tienen ningún atributo en su interior.

#### 4.4. DSI 4.4: Identificación de Operaciones de las Clases

Las operaciones que incluye cada una de las clases son:

- Clase **GestorCorreos**:
  - *validar*: valida a un usuario en el sistema si el nombre de usuario y la contraseña que introduce éste son datos correctos. En caso contrario devuelve un código de error.
  - *logout*: si el nombre de usuario recibido ha sido validado, cierra la sesión de ese usuario en el sistema y en caso contrario devuelve un error.

- *registrar*: registra a un nuevo usuario en el sistema si no existe ya.
- *obtenerInfo*: obtiene información concreta de un usuario.
- *cambioDomicilio*: permite el cambio de domicilio actual del usuario.
- Clase **GestorEnvíos**:
  - *enviarDocumento*: se encarga de registrar el envío del documento identificado con el id recibido como parámetro en la base de datos.
  - *enviarPaquete*: se encarga de registrar el envío del paquete identificado con el id recibido como parámetro en la base de datos.
  - *seguirEnvío*: se encarga de obtener información del estado del envío identificado con el id recibido como parámetro de la base de datos.
- Clase **GestorFilatelia**:
  - *registrar*: registra a un nuevo usuario en la base de datos.
  - *buscaAsociación*: obtiene la asociación identificada con el nombre recibido como parámetro.
  - *buscaInfo*: obtiene la información correspondiente a los datos recibidos como parámetro.
  - *realizaPedido*: registra un nuevo pedido en la base de datos.
  - *buscaSucursal*: obtiene la sucursal identificada con el nombre recibido como parámetro.
- Clase **GestorOficinaVirtual**:
  - *doSeguimiento*: obtiene información del estado del envío identificado con el id recibido como parámetro.
  - *ultimosEnvíos*: obtiene información de los últimos envíos realizados por el usuario identificado por el id recibido como parámetro.
  - *justificante*: obtiene los datos de un justificante de la factura recibida como parámetro.
  - *tareasPendientes*: obtiene un listado con los envíos pendientes del usuario recibido como parámetro.
- Clase **CentrosCorreos**:
  - *buscaAsociación*: obtiene de la lista de asociaciones la asociación cuyo nombre coincida con el recibido como parámetro.
  - *buscaSucursal*: obtiene de la lista de sucursales la sucursal cuyo nombre coincida con el recibido como parámetro.

Las clases que no aparecen en esta lista es porque no tienen ninguna operación en su interior.

#### 4.5. DSI 4.5: Diseño de la Jerarquía

Después de la realización de los puntos anteriores a este se ha llegado a la conclusión que el diagrama de clases mostrado en el punto 4.1 ya es viable para el entorno de desarrollo especificado.

En nuestra jerarquía de clases simplemente va a existir una clase abstracta (*Envíos*) de la cual van a heredar la clase *EnvíoDocumento* y la clase *EnvíoPaquete*.

#### 4.6. DSI 4.7: Especificación de Necesidades de Migración y Carga Inicial de Datos

Será necesario una migración de datos debido a que el sistema anterior tiene datos importantes que no pueden ser eliminados como pueden ser los siguientes:

- Los usuarios ya registrados en la aplicación y todo lo referente a éstos (envíos, pedidos, tareas, historiales,...),
- Los productos del Portal de Correos y del Portal Filatélico,
- Información referente a la empresa,
- ...

Y después de haber realizado la migración de esos datos, será necesaria la inclusión de nuevos datos en el sistema para todo lo que se refiere a la nueva aplicación. La carga de éstos datos no debería entrar en conflicto con los anteriores debido a que se han tratado por separado y además han sido pensados para que ésto no ocurra.

No se utilizara ningún hardware complementario para la realización de esta migración. Y el software a utilizar es el propio de la base de datos nueva (Oracle).

## 5. DSI 6: Diseño Físico de Datos

### 5.1. DSI 6.1: Diseño del Modelo Físico de Datos

El software de base de datos a utilizar en el servidor correspondiente será Oracle, el cual utiliza un lenguaje de consulta estructurado (SQL). A continuación se muestra el contenido físico de la base de datos, es decir, las tablas con cada atributo y el tipo y rango de éste:

■ **Tabla Usuario:**

- *IDUsuario*: tipo CHAR(10), Primary Key.
- *Nickname*: tipo CHAR(50). NOT NULL.
- *Password*: tipo CHAR(20). NOT NULL.
- *Nombre*: tipo CHAR(50). NOT NULL.
- *Provincia*: tipo CHAR(100). NOT NULL.
- *Municipio*: tipo CHAR(100). NOT NULL.
- *CPostal*: tipo SHORT. NOT NULL.
- *Calle*: tipo CHAR(200). NOT NULL.
- *Número*: tipo CHAR(3). NOT NULL.
- *Piso*: tipo CHAR(2).
- *Escalera*: tipo CHAR(2).
- *Teléfono*: tipo CHAR(10).

■ **Tabla Particular:**

- *IDUsuario*: tipo CHAR(10). Primary Key y Foreign Key a la tabla Usuario.
- *Apellido1*: tipo CHAR(20). NOT NULL.
- *Apellido2*: tipo CHAR(20).
- *DNI*: tipo CHAR(9). NOT NULL.
- *EsColeccionista*: tipo BIT. NOT NULL.

■ **Tabla Empresa:**

- *IDUsuario*: tipo CHAR(10). Primary Key y Foreign Key a la tabla Usuario.
- *CIF*: tipo CHAR(10). NOT NULL.
- *EsAsocFilatélica*: tipo BIT. NOT NULL.
- *Descripción*: tipo LONGTEXT.
- *Fax*: tipo SHORT.

■ **Tabla Producto:**

- *Código*: tipo CHAR(10). Primary Key.
- *Nombre*: tipo CHAR(50). NOT NULL.
- *Descripción*: tipo CHAR(400).
- *Precio*: tipo CHAR(10). NOT NULL.

■ **Tabla ProdFilatelico:**

- *CódigoProd*: tipo CHAR(10). Primary Key y Foreign Key a la tabla Producto.
- *CódigoColec*: tipo CHAR(10). Unique y Foreign Key a la tabla Colección.

■ **Tabla Colección:**

- *Código*: tipo CHAR(10). Primary Key.
  - *Nombre*: tipo CHAR(20). NOT NULL
  - *Descripción*: tipo CHAR(200).
- **Tabla Envío:**
- *IDEnvío*: tipo CHAR(10). Primary Key.
  - *IDUsuario*: tipo CHAR(10). Unique y Foreign Key a la tabla Usuario. NOT NULL.
  - *CódigoProd*: tipo CHAR(10). Unique y Foreign Key a la tabla Producto. NOT NULL.
  - *Remitente*: tipo TEXT(100).
  - *DirRemitente*: tipo TEXT(100).
  - *Destinatario*: tipo TEXT(100). NOT NULL.
  - *DirDestinatario*: tipo TEXT(100). NOT NULL.
  - *FechaIni*: tipo DATETIME. NOT NULL.
  - *FechaFin*: tipo DATETIME.
  - *Estado*: tipo CHAR(15). NOT NULL.
- **Tabla Pedido:**
- *IDPedido*: tipo CHAR(10). Primary Key.
  - *IDUsuario*: tipo CHAR(10). Unique y Foreign Key a la tabla Usuario. NOT NULL.
  - *CódigoProd*: tipo CHAR(10). Unique y Foreign Key a la tabla Producto. NOT NULL.
  - *FechaIni*: tipo DATETIME. NOT NULL.
  - *FechaFin*: tipo DATETIME.
  - *Estado*: tipo CHAR(15). NOT NULL.

La migración de los datos del actual Portal de Correos al nuevo Portal serán:

1. La información de la empresa.
2. Productos Filatélicos,
3. Productos de Venta (tipos de documentos, paquetes, etc.),
4. Usuarios del Sistema,
5. Tareas e historiales de cada usuario.

Después de la migración completa de los datos, se insertarán los nuevos elementos que no pertenecían al portal anterior. Para la correcta migración de dicha información no se utilizará ningún hardware específico.

Siempre que un servidor considere oportuno realizar una consulta en la base de datos se deberá hacer mediante una serie de índices que incorporarán aquellas tablas de mayor acceso por parte del usuario, ya que de esta forma se agiliza mucho la devolución de los datos, y a partir de hay irá obteniendo la información complementaria que se desee.

El sistema de la base de datos será implementado de tal manera que jamás se permitirá el acceso a más de un servidor a un mismo registro de la misma tabla con el fin de evitar los problemas de consistencia de datos que se podrían dar como consecuencia de la concurrencia entre servidores.



## 5.2. DSI 6.2: Especificación de los Caminos de Acceso a los Datos

Los servidores tendrán acceso a los datos mediante una conexión de red local (LAN) al servidor de base de datos.

Cómo se ha indicado en el punto anterior el sistema ya implementa los mecanismos adecuados para resolver los problemas de concurrencia que podrían surgir si más de un servidor desea obtener exactamente la misma información del servidor de base de datos.

Debido a que un usuario tendrá acciones muy comunes con la aplicación se crearán una serie de índices en las tablas con mayor acceso con el fin de mejorar la obtención de dichos datos. Algunos índices que a priori se crearán son:

- En la tabla ENVÍO habrá un índice compuesto por los campos *IDUsuario* y *FechaIni* para conseguir un acceso más rápido a estos datos.
- En la tabla PARTICULAR habrá un índice compuesto por *Apellido1* y *Apellido2* para buscar de forma más rápida un usuario.
- ...

Simplemente hemos mencionado alguno de los índices que serán creados, pero seguramente se incorporarán más en la base de datos cuando ya se realice la implementación de ésta.

Respecto a la seguridad del acceso a estos datos se requerirá la identificación correcta de los servidores desde los que se puede acceder a dicha información. Es decir, el servidor de BBDD mantendrá una tabla con las MAC de los servidores que tendrán acceso a éste, y además cada servidor tendrá el nombre de usuario y contraseña necesario para realizar las consultas. Cualquier otro servidor que quiera realizar una consulta a esta base de datos no podrá hacerla debido a que éste no está registrado en dicha tabla del servidor de BBDD.

## 5.3. DSI 6.3: Optimización del Modelo Físico de Datos

Con el fin de mejorar los tiempos de respuesta de estos servidores las consultas se realizarán sobre datos muy concretos (si es posible) para evitar ciertos retrasos innecesarios al realizar dichas consultas sobre un conjunto muy amplio de datos.

Cada servidor irá almacenando esas pequeñas consultas, mientras los datos no se vean modificados, para evitar tener que hacer dos consultas seguidas sobre la misma información y así poder eliminar cuellos de botella al acceder a la vez a la base de datos varios servidores. Por lo que estos servidores tendrán una pequeña caché en la que almacenarán esos datos. El sistema empleado para esta caché será de reemplazo de los datos más antiguos, es decir, cuando la caché esté llena el sistema de reemplazamiento a utiliza será FIFO (First In First Out).

El servidor de base de datos llevará un registro de qué información tiene cada servidor para que en el caso de modificación de esos datos avisar a dichos servidores de que borren de su caché esos datos y en la próxima solicitud de esa información deberá hacer una nueva consulta a la base de datos.

## 5.4. DSI 6.4: Especificación de la Distribución de Datos

Toda la información estará situada en el servidor de base de datos. Pero parcialmente cada servidor tendrá una pequeña parte de esta información en sus cachés, que como se ha mencionado anteriormente, sirven para la mejora del tiempo de respuesta y para evitar cuellos de botella que se provocarían cuando muchos servidores desean realizar consultas sobre una misma base de datos. Esto debe realizarse de esta forma ya que el número de visitas media por mes alcanza las 400.000 (cuatrocientas mil).

## 6. DSI 8: Generación de Especificaciones de Construcción

### 6.1. DSI 8.1: Especificación del Entorno de Construcción

Las computadoras encargadas del desarrollo de este sistema deberán tener:

- Requisitos **Hardware**:
  - Sistema de Almacenamiento: disco duro con un tamaño aproximado de quinientos gigabytes (200 GB).
  - Módulos de memoria RAM con un tamaño aproximado de cuatro gigabytes (4 GB).
  - Tarjeta de Red: RJ45 para el acceso a Internet en caso de que el desarrollador necesite realizar alguna consulta.
  - Teclado, Ratón, Monitor y otros dispositivos que son necesarios para el correcto funcionamiento de la máquina.
- Requisitos **Software**:
  - Sistema Operativo: Windows.
  - Navegador Web: para ir observando el desarrollo de la aplicación.
  - Base de datos local: Oracle, para realizar pruebas de forma local.
  - J2EE 1.4: para el desarrollo de aplicaciones en las que se utilice java.
  - Procesador de Textos sencillo para el código de las páginas web, por ejemplo: Notepad++. Aunque puede ser utilizado el entorno de desarrollo Eclipse, Visual Basic, DevC++ u otros (se especificará más adelante cual utilizar).
  - Se utilizarán los compiladores correspondientes para la compilación de código desarrollado en Visual Basic, C++ o .Net
- Cada desarrollador tendrá un ordenador y un nombre de usuario y contraseña para acceder a éste.
- Los permisos de éstos usuarios (desarrolladores) serán de administrador para evitar tener problemas de éste tipo.

Todo código deberá ir acompañado de su correspondiente descripción de lo que hace dicha parte. Es decir, cada método tendrá en su parte superior un pequeño comentario mencionando el objetivo y los parámetros que se le pasan al método y lo que devuelve dicho método. Como opción del desarrollador podrá incluir dentro de cada método pequeños comentarios de lo que se realiza en cada una de las partes, aunque no es obligatorio pero sí muy recomendable.

### 6.2. DSI 8.2: Definición de Componentes y Subsistemas de Construcción

El diagrama de componentes en el que se muestran los componentes y el subsistema de construcción es el siguiente:

Según el diagrama anterior los componentes y subsistemas de construcción son los siguientes:

- Subsistema *Cliente*: su componente es el propio usuario de la aplicación.
- Subsistema *Servidor*, los componentes que contiene son:
  - *GestorCorreos*.
  - *GestorFilatelia*.
  - *GestorEnvíos*.
  - *GestorOficinaVirtual*.
- Subsistema *Base de Datos*, los componentes que contiene son:
  - Tabla *Usuarios*.

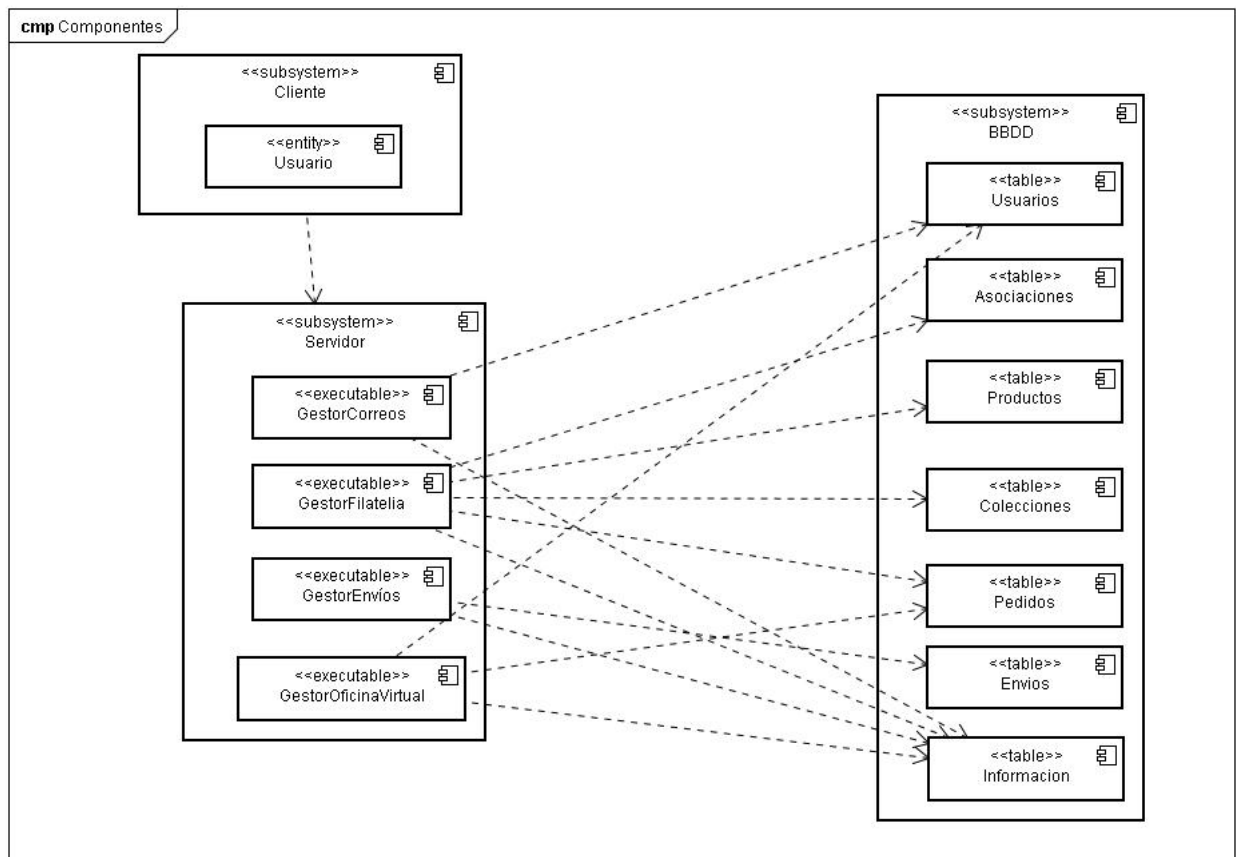


Figura 28: Diagrama de componentes.

- Tabla *Asociaciones*.
- Tabla *Productos*.
- Tabla *Pedidos*.
- Tabla *Envíos*.
- Tabla *Información*.
- Tabla *Colecciones*.

La compilación de cada uno de los componentes mencionados se podrá hacer por separado ya que se pretende que cada uno sea lo más independiente posible del resto de la aplicación. En caso de que se necesite una especificación mayor de la compilación de éstos se informará con mayor detalle en el documento referente a la construcción de la aplicación.

### 6.3. DSI 8.3: Elaboración de Especificaciones de Construcción

Los componentes de construcción son:

- *GestorCorreos*: será un componente con el que se dará soporte al registro, la validación (login), el cierre de sesión (logout), la obtención de información, el cambio de domicilio del cliente, la personalización de los sellos que desee el cliente, sobres y embalajes, creamailing, el apartado postal, la gestión del correo electrónico, los dominios y la atención al cliente.

- *GestorFilatelila*: componente a través del cual se dará soporte a las siguientes acciones: bienvenida a la filatelia, apadrinamiento, información de los productos de Filatelia, comprar productos, atención al cliente y mostrar información sobre los puntos de venta.
- *GestorEnvíos*: será un componente a través del cual deberá dar soporte a: enviar una Carta, un Telegrama, un Fax o un paquete, este último se le deberá de indicar la urgencia de envío que tiene para su tratamiento.
- *GestorOficinaVirtual*: será un componente con el que se dará soporte a: listar las tareas que tiene pendiente el usuario, obtener el seguimiento sobre un determinado envío para lo cual será necesario especificar el identificador que se le proporcionó al cliente en el momento de realizarlo, el histórico de todos los envíos que ha llevado a cabo el cliente y la solicitud de un justificante para lo cual el cliente deberá proporcionar el identificador del pedido o envío realizado.
- Tabla *Usuarios*: tendrá los campos correspondientes para guardar toda la información necesaria de los clientes.
- Tabla *Asociaciones*: tendrá los campos correspondientes para guarda toda la información necesaria de las asociaciones filatélicas.
- Tabla *Productos*: tendrá los campos correspondientes para guarda toda la información necesaria de los productos del Portal de Correos y el portal Filatélico.
- Tabla *Pedidos*: tendrá los campos correspondientes para guarda toda la información necesaria de los pedidos que realizan los usuarios registrados (clientes).
- Tabla *Envíos*: tendrá los campos correspondientes para guarda toda la información necesaria de los envíos que solicitan los usuarios registrados (clientes).
- Tabla *Información*: tendrá los campos correspondientes para guarda toda la información necesaria del resto del sistema.
- Tabla *Colecciones*: tendrá los campos correspondientes para guarda toda la información necesaria de las colecciones filatélicas.

#### 6.4. DSI 8.4: Elaboración de Especificaciones del Modelo Físico de Datos

Para la implementación del modelo físico de datos especificado en el apartado 6.1 de este documento, será necesario ejecutar el siguiente código SQL en la base de datos de Oracle para la creación de las tablas:

```
CREATE TABLE Usuario (
IDUsuario VARCHAR2(10) CONSTRAINT pk_usuario PRIMARY KEY,
Nickname VARCHAR2(50) NOT NULL,
Password VARCHAR2(20) NOT NULL,
Nombre VARCHAR2(50) NOT NULL,
Provincia VARCHAR2(100) NOT NULL,
Municipio VARCHAR2(100) NOT NULL,
CPPostal NUMBER NOT NULL,
Calle VARCHAR2(200) NOT NULL,
Número VARCHAR2(3) NOT NULL,
Piso VARCHAR2(2),
Escalera VARCHAR2(2),
Teléfono VARCHAR2(10)
);

CREATE TABLE Particular (
IDUsuario VARCHAR2(10) CONSTRAINT pk_usuario PRIMARY KEY
CONSTRAINT fk_usuario REFERENCES Usuario(IDUsuario),
```

```
Apellido1 VARCHAR2(20) NOT NULL,  
Apellido2 VARCHAR2(20),  
DNI VARCHAR2(9) NOT NULL,  
EsColeccionista VARCHAR(1) NOT NULL  
);
```

```
CREATE TABLE Empresa (  
IDUsuario VARCHAR2(10) CONSTRAINT pk_usuario PRIMARY KEY  
CONSTRAINT fk_usuario REFERENCES Usuario(IDUsuario),  
CIF VARCHAR2(10) NOT NULL,  
EsAsocFilatelica VARCHAR2(1) NOT NULL,  
Descripción VARCHAR2(500),  
Fax NUMBER  
);
```

```
CREATE TABLE Producto (  
Codigo VARCHAR2(10) CONSTRAINT pk_producto PRIMARY KEY,  
Nombre VARCHAR2(50) NOT NULL,  
Descripcion VARCHAR2(400),  
Precio VARCHAR2(10) NOT NULL  
);
```

```
CREATE TABLE Coleccion (  
Codigo VARCHAR2(10) CONSTRAINT pk_coleccion PRIMARY KEY,  
Nombre VARCHAR2(20) NOT NULL,  
Descripcion VARCHAR2(200)  
);
```

```
CREATE TABLE ProdFilatelico (  
CodigoProd VARCHAR2(10) CONSTRAINT pk_producto PRIMARY KEY  
CONSTRAINT fk_producto REFERENCES Producto(Codigo),  
CodigoColec VARCHAR2(10) UNIQUE  
CONSTRAINT fk_coleccion REFERENCES Coleccion(Codigo)  
);
```

```
CREATE TABLE Envio (  
IDEnvio VARCHAR2(10) CONSTRAINT pk_envio PRIMARY KEY,  
IDUsuario VARCHAR2(10) UNIQUE  
CONSTRAINT fk_usuario REFERENCES Usuario(IDUsuario),  
CodigoProd VARCHAR2(10) UNIQUE  
CONSTRAINT fk_producto REFERENCES Producto(Codigo),  
Remitente VARCHAR2(100),  
DirRemitente VARCHAR2(100),  
Destinatario VARCHAR2(100) NOT NULL,  
DirDestinatario VARCHAR2(100) NOT NULL,  
FechaIni DATE NOT NULL,  
FechaFin DATE,  
Estado VARCHAR2(15) NOT NULL  
);
```

```
CREATE TABLE Pedido (  
IDPedido VARCHAR2(10) CONSTRAINT pk_pedido PRIMARY KEY,  
IDUsuario VARCHAR2(10) UNIQUE  
CONSTRAINT fk_usuario REFERENCES Usuario(IDUsuario),  
CodigoProd VARCHAR2(10) UNIQUE
```

```
CONSTRAINT fk_producto REFERENCES Producto(Codigo),  
FechaIni DATE NOT NULL,  
FechaFin DATE,  
Estado VARCHAR(15) NOT NULL  
);
```

## 7. DSI 9: Diseño de la Migración y Carga Inicial de Datos

### 7.1. DSI 9.1: Especificación del Entorno de Migración

Debido a que no será necesario utilizar hardware complementario para la correcta realización del plan de migración propuesto, este entorno de migración simplemente necesitará una computadora completa en la que tenga el software utilizado para la base de datos, es decir, en esta computadora deberá estar instalado Oracle.

Por lo tanto, los requisitos de ese ordenador serán los mismos que los especificados en el punto 8.1 de este documento sobre la especificación del entorno de construcción. Aunque puede suceder que algunos de esos requisitos sean de menor importancia.

### 7.2. DSI 9.2: Diseño de Procedimientos de Migración y Carga Inicial

Para llevar a cabo la migración y carga inicial de datos de forma correcta y sin tener problemas de seguridad se creará una base de datos desde cero. Los pasos a seguir son:

1. Crear una base de datos vacía.
2. Asignar los usuarios que tienen capacidad para modificar esta base de datos. Estos usuarios son totalmente diferentes a los usuarios del Portal de Correos.
3. Crear las tablas y relaciones entre éstas.
4. Extracción de la base de datos anterior todo lo necesario para incluirlo en la nueva.
5. Insertar en las tablas los datos referentes al sistema antiguo.
6. Insertar en las tablas los datos referentes al sistema nuevo.
7. Llevar a cabo una serie de pruebas para comprobar el correcto funcionamiento de la base de datos.
8. Exportar esta base de datos al servidor de base de datos.
9. Crear en el servidor de base de datos una tabla con las MAC de los servidores que van a acceder a este servidor. Para ofrecer una seguridad referente a los datos.
10. Establecer la comunicación de los servidores con el servidor de base de datos.
11. Realizar pruebas desde los servidores para acceder a los datos.

Debe tenerse especial cuidado a la hora de exportar los datos de la aplicación anterior con el fin de no borrar registros necesarios para el nuevo Portal de Correos. Además cuando se realiza la inserción de esos datos y de los nuevos tendrá que realizarse en el orden indicado para no provocar conflictos en los datos.

Por motivos de seguridad se irán haciendo copias de seguridad de la base de datos aún sin terminar su total migración, con el fin de evitar la pérdida de información.

### 7.3. DSI 9.3: Diseño Detallado de Componentes de Migración y Carga Inicial

Los principales componentes de la migración de los datos son las tablas que deben ser creadas en la nueva base de datos. Estas tablas deben ser las siguientes:

- Usuarios,
- Pedidos,
- Productos,
- Asociaciones,

- Colecciones,
- Información,
- Envíos.

Los campos de cada una de estas tablas fueron especificados en el punto 8.4 de este documento.

Referente a la carga de los datos antiguos y nuevos se deberá ir haciendo con sumo cuidado para evitar los conflictos entre estos, por lo que el mejor plan será ingresar los datos antiguos antes de insertar los nuevos.

Después de haber realizado todo esto se llevarán a cabo una serie de pruebas para la comprobación del correcto funcionamiento de la base de datos. Estas pruebas deberían determinar si se corrigen ciertos aspectos o se continua con el proceso. A continuación se creará la tabla con las MAC de los servidores que van a tener accesos a este servidor para cumplir con las condiciones de seguridad. Se establecerá la conexión de red local entre los servidores y el servidor de base de datos, tras esto se realizarán pruebas desde los servidores para comprobar el correcto funcionamiento de las consultas de éstos a la base de datos.

#### **7.4. DSI 9.4: Revisión de la Planificación de la Migración**

Una vez alcanzado el punto final del proceso de migración de los datos y haber realizado las pruebas suficientes se continuará con el proceso de integrar parte de la aplicación en estos servidores con el motivo de seguir haciendo pruebas con la base de datos pero ya desde la propia aplicación.



## 8. DSI 10: Especificación Técnica del Plan de Pruebas

### 8.1. DSI 10.1: Especificación del Entorno de Pruebas

El entorno de pruebas será el siguiente:

■ **Hardware:**

- Sistema de Almacenamiento: disco duro de cien gigabytes(100 GB).
- Unidad de lectora y grabadora de DVD.
- Puertos USB.
- Dos gigabytes de memoria RAM (2 GB).
- Tarjeta de RED: RJ45, pero no se dispondrá de acceso a Internet, solo a la aplicación.
- Teclado, Ratón y demás dispositivos para el correcto funcionamiento de un ordenador.

■ **Software:**

- Sistema Operativo instalado: Windows XP o Seven, Ubuntu, Mac OS X.
- Navegador Web: Internet Explorer, Mozilla Firefox y Safari con las versiones más recientes.
- Este entorno de desarrollo de pruebas tendrá unos 50 ordenadores individuales, y no tendrán comunicación entre ellos.
- Se dispondrá de una herramienta para medir el tráfico web, para asegurar que el tráfico en una operación en el portal conlleva una carga capaz de llevarse a cabo tanto en conexiones rápidas como en las lentas.
- En cada puesto habrá un test indicando las características de la aplicación que deben ser probadas. En el test se incluirá un espacio adicional en el cual el usuario podrá indicar otras características, dentro de las indicas, que ha probado y no vienen especificadas en el test.

La elección de los usuarios para realizar las pruebas será llevada a cabo mediante una pequeña entrevista de trabajo muy simple para comprobar la capacidad de éstos con el fin de que las pruebas sean realizadas correctamente. El perfil de éstos usuarios no será demasiado complejo debido a que se buscan personas con un simple sentido común y con unos conocimientos mínimos sobre ordenadores.

### 8.2. DSI 10.2: Especificación Técnica de Niveles de Prueba

Antes de que los usuarios encargados de llevar a cabo un conjunto de pruebas, los desarrolladores han ido realizando *pruebas unitarias* con cada componente del sistema y una vez pasadas éstas habrán realizado otras referentes a las *pruebas de integración* con las que se verifica el correcto funcionamiento tras el ensamblaje de todos los componentes. Por supuesto las pruebas de integración no serán realizadas de golpe sino que se van integrando varios componentes y se llevan a cabo ciertas pruebas, cuando éstas consiguen sus objetivos se siguen incluyen más componentes a las pruebas hasta se logra completar la integración de todos los componentes y las pruebas alcanzan los objetivos deseados.

Después de todo esto, un grupo seleccionado de usuarios llevarán a cabo las *pruebas del sistema*. Dichos usuarios dispondrán de varias hojas de test que deberán ir siguiendo para la correcta realización de la prueba, además irán señalando las ventajas e inconvenientes de cada una de las características que están probando. Por supuesto el usuario dentro de las características marcadas a probar podrá hacer todo tipo de pruebas tanto buenas como malas, ya que lo que se pretende es comprobar todo tipo de situaciones que puedan llegar a suceder. Y al finalizar la prueba deberán entregar una hoja con los resultados a la persona indicada con el fin de volver a modificar (si es necesario) la aplicación para satisfacer las necesidades de los usuarios. Los criterios de verificación y aceptación de cada prueba vendrán determinados en los test con los que el usuario ve guiado su trabajo.

Las pruebas tendrán un período de duración de unos tres meses, durante los cuales se realizarán pruebas rotativas entre los diferentes usuarios durante jornadas de 8 horas, de forma que cada uno llevará a cabo pruebas durante el período de una hora, se cambiará la hora siguiente, volverá en la siguiente, así hasta completar la jornada. En cada hora de trabajo el usuario realizará pruebas sobre un elemento diferente, entre el Portal Principal, el Portal Filatélico y la Oficina Virtual.

A cada semana de pruebas le seguirá otra de modificaciones y verificaciones con el propósito de mejorar paulatinamente el sistema y adaptarlo a las necesidades que vayan surgiendo. Así estos usuarios comprobarán las mejoras añadidas.

Al final del período de pruebas, se realizarán las *pruebas de aceptación* a través de las cuales se validará que el sistema cumple con todos los requisitos de funcionamiento esperado con el fin de conseguir la aceptación final del sistema por parte del usuario.

### 8.3. DSI 10.3: Revisión de la Planificación de Pruebas

La aceptación de seguir adelante con el proceso depende en gran factor del resultado obtenido de estas pruebas. En el caso de que las pruebas sean satisfactorias se continuará adelante, sin embargo, si éstas no corresponden correctamente a los requisitos especificados será necesario una nueva actualización de la aplicación para cumplir con los requisitos afectados.

Por lo tanto hasta la no validación completa de estas pruebas no seguirá adelante el proyecto ya que podrían surgir errores muy graves más adelante y su corrección sería más costosa.

## 9. DSI 11: Establecimiento de Requisitos de Implantación

### 9.1. DSI 11.1: Especificación de Requisitos de Documentación de Usuario

Se realizará un manual de usuario de la aplicación el cual será entregado a los usuarios con el fin de que puedan manejar correctamente la aplicación. Además este documento les servirá para ayudar a los clientes a realizar sus gestiones cuando éstos llamen para solicitar información o ayuda, por lo que también les valdrá como una pequeña formación. El manual será entregado tanto en papel como en soporte informático.

Se irán realizando las modificaciones oportunas del documento según vaya cambiando el sistema, provocado por las modificaciones de éste durante el período de pruebas. Estas sucesivas actualizaciones serán controladas mediante su versión.

### 9.2. DSI 11.2: Especificación de Requisitos de Implantación

Algunos requisitos de implantación son especificados a continuación:

- Los desarrolladores deberán tener los conocimientos apropiados sobre los lenguajes de programación que el sistema utiliza y otros aspectos referentes a la implementación. Por lo tanto, la elección de esto se deberá hacer cuidadosamente para no cometer errores.
- Se realizarán grupos de trabajo para elaborar cada una de las características del sistema y que luego serán unificadas. Por lo que deben trabajar de forma paralela y los módulos, que cada grupo desarrolla, deben estar débilmente acoplados y fuertemente cohesionados. El número de estos grupos esta por decidir pero no deberían ser grupos de más de 5 personas.
- Antes de realizar cualquier incorporación nueva a la aplicación, es decir, la implementación de aspectos de la aplicación que todavía no se han insertado, se deberá tener más o menos un consenso de cómo realizar dicha implementación. Con esto se pretende que el grupo piense bien en lo que debe hacer para proponer una solución eficiente y así poder evitar que el grupo se ponga de forma inmediata a implementar una solución, de esta manera se ahorrará mucho trabajo y tiempo.
- Se requiere que los ordenadores tengan los requisitos mínimos especificados en el punto 8.1 de este documento.