

# eShareSpace

eSharespace

Oscar Doral Collazo

Trabajo fin de carrera

Consultor: Juan José Cuadrado Gallego

**Propósito:**

En este documento encontrará la memoria del trabajo final de carrera de Oscar Doral Collazo basado en el análisis y especificación del producto denominado eSharespace.

**Audiencia:**

*El documento no tiene una audiencia limitada.*

**Fecha de emisión:**

6 de Junio de 2012

**Histórico de revisiones:**

Versión	Fecha	Cambios realizados	Autor
1.0 (Borrador 1)	06-06-2012	Original	Oscar Doral Collazo

**Referencias:**

1. Este documento es el TFC de Oscar Doral Collazo para la UOC.
2. Este documento ha sido enviado a:  
Juan José Cuadrado Gallego

**Obtención de copias de este documento:**

Este documento estará disponible en la web de la UOC.

**Control de cambios:**

Si desea modificar este documento por favor envíe la petición a Oscar Doral Collazo a través de la UOC.

**Dedicaciones y agradecimientos:**

**A Mariola ya que sin ella nada de esto sería posible.**

**A Mari, tu ejemplo siempre nos guiará.**

## Índice

<b>1</b>	<b>INTRODUCCIÓN .....</b>	<b>5</b>
<b>2</b>	<b>METODOLOGÍA .....</b>	<b>5</b>
2.1	PLAN DE TRABAJO.....	5
2.2	ESPECIFICACIÓN Y ANÁLISIS DE REQUISITOS.....	5
2.3	DISEÑO DETALLADO.....	6
2.4	MEMORIA Y PRESENTACIÓN.....	6
2.5	DEFENSA DEL PROYECTO REALIZADO. ....	6
<b>3</b>	<b>ESPECIFICACIÓN Y ANÁLISIS.....</b>	<b>6</b>
3.1	METAS Y OBJETIVOS .....	6
3.2	DECLARACIÓN DE ALCANCE .....	7
3.3	CONTEXTO DEL SOFTWARE.....	7
3.4	PRINCIPALES LIMITACIONES.....	8
<b>4</b>	<b>ESCENARIOS DE USO .....</b>	<b>8</b>
4.1	PERFILES DE USUARIO .....	8
4.1.1	<i>Usuario no identificado.....</i>	<i>9</i>
4.1.2	<i>Usuario del sistema.....</i>	<i>9</i>
4.1.3	<i>Administrador del sistema.....</i>	<i>9</i>
4.1.4	<i>Superadministrador .....</i>	<i>9</i>
4.2	CASOS DE USO.....	9
4.2.1	<i>Casos de uso de un usuario no identificado .....</i>	<i>10</i>
4.2.2	<i>Usuario del sistema.....</i>	<i>15</i>
4.2.3	<i>Administrador del sistema.....</i>	<i>21</i>
4.2.4	<i>Superadministrador .....</i>	<i>36</i>
4.3	DESCRIPCIÓN Y MODELO DE DATOS.....	39
4.3.1	<i>Descripción de datos.....</i>	<i>39</i>
4.3.2	<i>Modelo de datos.....</i>	<i>41</i>
4.3.3	<i>Relaciones.....</i>	<i>43</i>
<b>5</b>	<b>DISEÑO DETALLADO .....</b>	<b>43</b>
5.1	INTRODUCCIÓN .....	43
5.2	MODELO DE DATOS COMÚN .....	45
5.3	NAVEGABILIDAD ENTRE PANTALLAS.....	46
5.4	EXCEPCIONES .....	47
5.5	MODELO DE GESTIÓN DE UNA ENTIDAD .....	47
5.5.1	<i>Creación de una entidad.....</i>	<i>49</i>
5.5.2	<i>Modificación de una entidad .....</i>	<i>51</i>
5.5.3	<i>Búsqueda de una entidad.....</i>	<i>53</i>
5.5.4	<i>Búsqueda de entidades.....</i>	<i>55</i>
5.5.5	<i>Eliminación de una entidad.....</i>	<i>57</i>
5.6	SUBSISTEMA DE GESTIÓN DE USUARIOS (SGU).....	59
5.6.1	<i>Modelo de datos.....</i>	<i>59</i>
5.6.2	<i>Diagramas de secuencia .....</i>	<i>61</i>
5.6.3	<i>Fichas CRC.....</i>	<i>66</i>
5.7	SUBSISTEMA DE GESTIÓN DE PROGRAMACIONES (SGP).....	69
5.7.1	<i>Modelo de datos.....</i>	<i>69</i>
5.7.2	<i>Negocio.....</i>	<i>70</i>
5.7.3	<i>Diagramas de secuencia .....</i>	<i>72</i>
5.7.4	<i>Fichas CRC.....</i>	<i>73</i>
5.8	SUBSISTEMA DE GESTIÓN DE ELEMENTOS-TIPO (SGET) .....	75
5.8.1	<i>Modelo de datos.....</i>	<i>75</i>

---

5.8.2	<i>Negocio</i> .....	75
5.8.3	<i>Diagramas de secuencia</i> .....	77
5.8.4	<i>Fichas CRC</i> .....	79
5.9	SUBSISTEMA DE GESTIÓN DE RESERVAS (SGR).....	81
5.9.1	<i>Modelo de datos</i> .....	81
5.9.2	<i>Negocio</i> .....	81
5.9.3	<i>Diagramas de secuencia</i> .....	83
5.9.4	<i>Fichas CRC</i> .....	86
5.10	SUBSISTEMA DE GESTIÓN DE RESERVAS HISTÓRICAS (SGRH).....	87
5.10.1	<i>Modelo de datos</i> .....	87
5.10.2	<i>Negocio</i> .....	87
5.10.3	<i>Diagramas de secuencia</i> .....	89
5.10.4	<i>Fichas CRC</i> .....	90
5.11	DISEÑO DE BASE DE DATOS.....	91
5.11.1	<i>Modelo de tablas de base de datos</i> .....	91
<b>6</b>	<b>COMPORTAMIENTO DEL SISTEMA</b> .....	<b>94</b>
6.1	USABILIDAD.....	94
<b>7</b>	<b>GLOSARIO</b> .....	<b>94</b>
<b>8</b>	<b>BIBLIOGRAFÍA</b> .....	<b>95</b>
<b>9</b>	<b>ANEXOS</b> .....	<b>95</b>
9.1	SOFTWARE.....	95
9.2	PLANIFICACIÓN DEL TFC.....	96

## 1 Introducción

En una comunidad de propietarios se hace uso de una pista polideportiva y dos de pádel mediante una reserva previa. Actualmente la reserva se hace por medio del personal de mantenimiento de las pistas y mediante solicitud física lo cual hace que no todo el mundo tenga las mismas posibilidades de reserva de pistas y quita tiempo efectivo del mantenimiento de la instalación.

El propósito de este documento por tanto es describir la especificación de requisitos de un sistema que provea a una comunidad de usuarios de un mecanismo de gestión de elementos compartidos de manera online. Este sistema, y hasta que no se decida un nombre final, recibirá el nombre de "**eSharespace**".

Aunque el análisis inicial se hará sobre una comunidad de propietarios y unas pistas deportivas como elementos a compartir, la aplicación se generalizará ara permitir la gestión de reservas de cualquier tipo de elemento (pistas deportivas, salas, bicicletas, raquetas...) de tal manera que podrá establecerse en cualquier situación en la que se requiera compartir elementos entre un grupo de usuarios.

En conclusión, eSharespace permitirá, a los usuarios acreditados en el sistema, reservar elementos sin la necesidad de personarse físicamente. De esta manera, y dado el impacto de internet en la sociedad, se estima que todos los usuarios tendrán las mismas posibilidades de acceso a la reserva.

## 2 Metodología

La metodología que se seguirá para consecución de este proyecto es la del ciclo de vida clásico de desarrollo del software o modelo en cascada. En este tendremos varias etapas ordenadas y cuyo inicio vendrá determinado por la consecución de la anterior.

Las etapas por las que pasaremos a lo largo de este trabajo serán:

- Plan de trabajo.
- Especificación y análisis de requisitos.
- Diseño detallado.
- Memoria y presentación.
- Defensa del proyecto realizado.

### 2.1 Plan de trabajo

Durante esta fase se detallará el objetivo del trabajo y la definición del resto de etapas con sus objetivos así como una planificación del proyecto a alto nivel.

### 2.2 Especificación y análisis de requisitos.

Durante esta etapa se trabajará sobre los siguientes objetivos:

- Toma de requisitos.
- Especificación de requisitos por medio de:
  - Identificación de actores.

- Identificación de guiones.
- Análisis de casos de uso.
- Definición del diagrama E/R de las principales entidades.
- Requisitos funcionales de la interface.

### **2.3 Diseño detallado.**

Durante esta etapa se trabajará sobre los siguientes objetivos:

- Diagramas de casos de uso.
- Diagramas de actividad.
- Diagramas de estados.
- Diagramas de clases.
- Diagramas de colaboración.
- Diagramas de secuencia.
- Diseño de la arquitectura.
- Diseño de la interface.

### **2.4 Memoria y presentación.**

En esta etapa se realizará el documento presente y una presentación que sintetiza el trabajo realizado.

### **2.5 Defensa del proyecto realizado.**

Durante esta fase se contestará a todas aquellas dudas planteadas por el Tribunal Virtual.

## **3 Especificación y Análisis**

### **3.1 Metas y objetivos**

Las principales metas y objetivos de este software son:

- 2.1 Permitir el acceso a todos los escenarios planteados en este documento desde múltiples tipos de dispositivo (PC, teléfonos móviles,...).
- 2.2 Trabajará siempre en línea dando siempre información sincronizada del estado del sistema y de las reservas de sus elementos.
- 2.3 Habrá perfiles de usuarios que podrán gestionar las reservas de terceros de tal manera que no se perderá la posibilidad de realizar reservas como se viene haciendo hasta ahora.

- 2.4 Avisar al usuario mediante un correo electrónico o un mensaje SMS de los cambios de estado en las operaciones en las que se vea involucrado:
  - Alta y baja en el sistema.
  - Realización y eliminación de la reserva.
  - Inicio del tiempo de la reserva.
  - Finalización del tiempo reservado.
- 2.5 Implementará una seguridad robusta.
- 2.6 Proveerá de personalizaciones según el dispositivo desde el que se acceda, véase un navegador web o una aplicación para móvil.
- 2.7 Será multi-idioma comenzando inicialmente en castellano, inglés y francés pero agregando a posteriori el resto de idiomas.

### 3.2 Declaración de alcance

**eSharespace** es un sistema que permitirá el acceso tanto desde navegadores web como desde aplicaciones específicas de plataformas móviles. Inicialmente el alcance se prevé muy ambicioso por tanto se plantea acortarlo de manera que se pueda ofrecer toda la funcionalidad al mayor conjunto de dispositivos. Dado que los dispositivos móviles poseen navegadores web y podrán por tanto acceder aunque la interface de usuario no sea la más adecuada, éste documento cubrirá el acceso al sistema mediante una aplicación web. No obstante, se planteará que en el alcance inicial el servidor ofrezca un API con el que se podrá acceder a todas las operaciones que plantea este documento para posteriormente desarrollar clientes más adecuados para plataformas específicas.

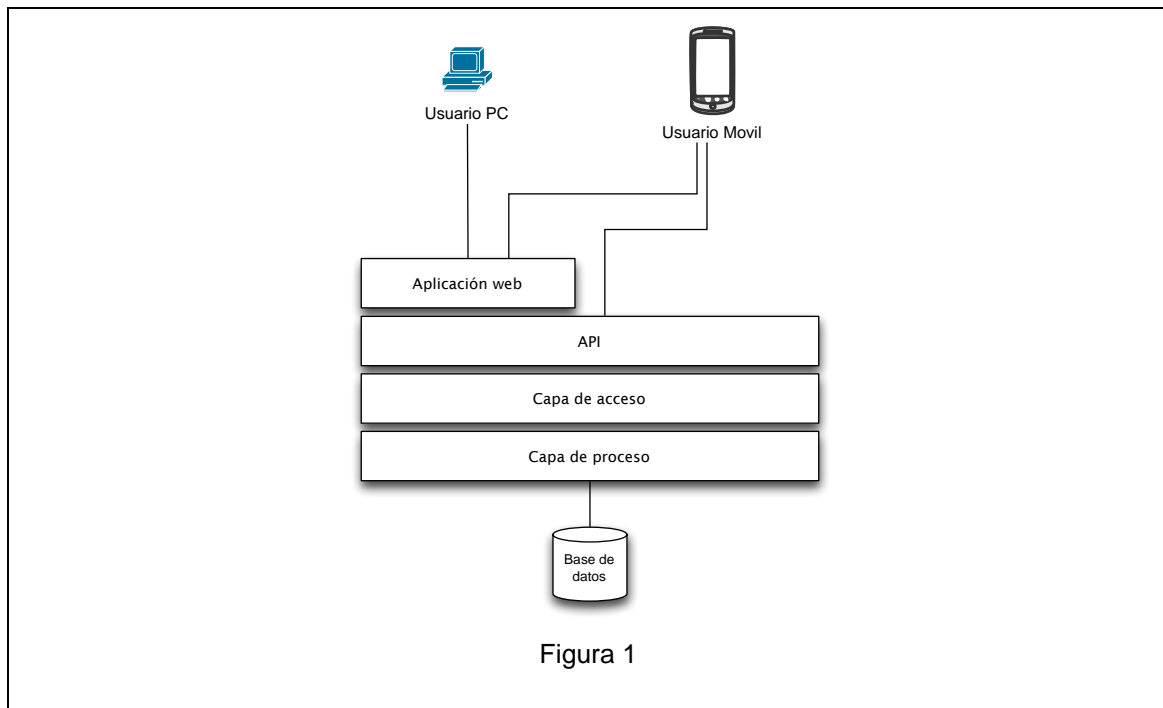
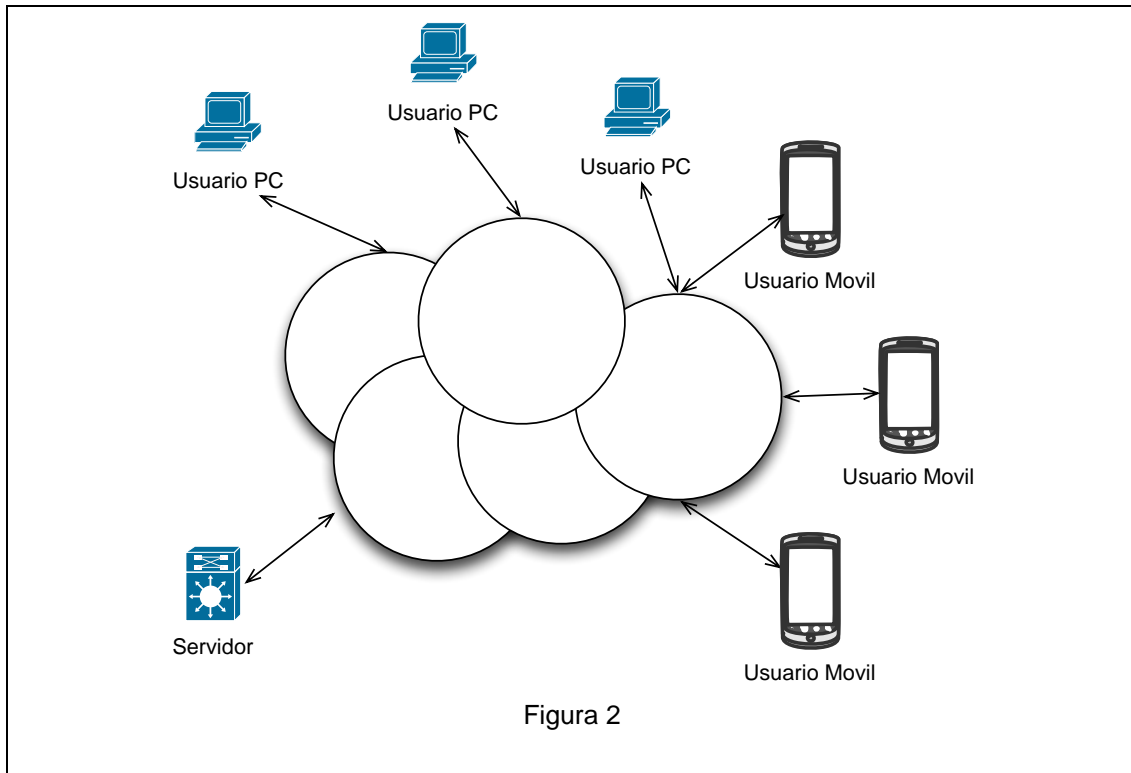


Figura 1

### 3.3 Contexto del software



La aplicación por tanto se dividirá en un servidor donde se ubicará el acceso a base de datos, la capa de lógica de negocio y seguridad y en múltiples clientes. Los clientes conectarán con el servidor a través de internet y por medio de un navegador web o de una aplicación específica de la plataforma.



### 3.4 Principales limitaciones

eSharespace tendrá las siguientes limitaciones:

No se permitirá la definición de programaciones de reserva cuya hora de inicio en conjunto con la duración de la reserva y la repetición abarque más de un día.

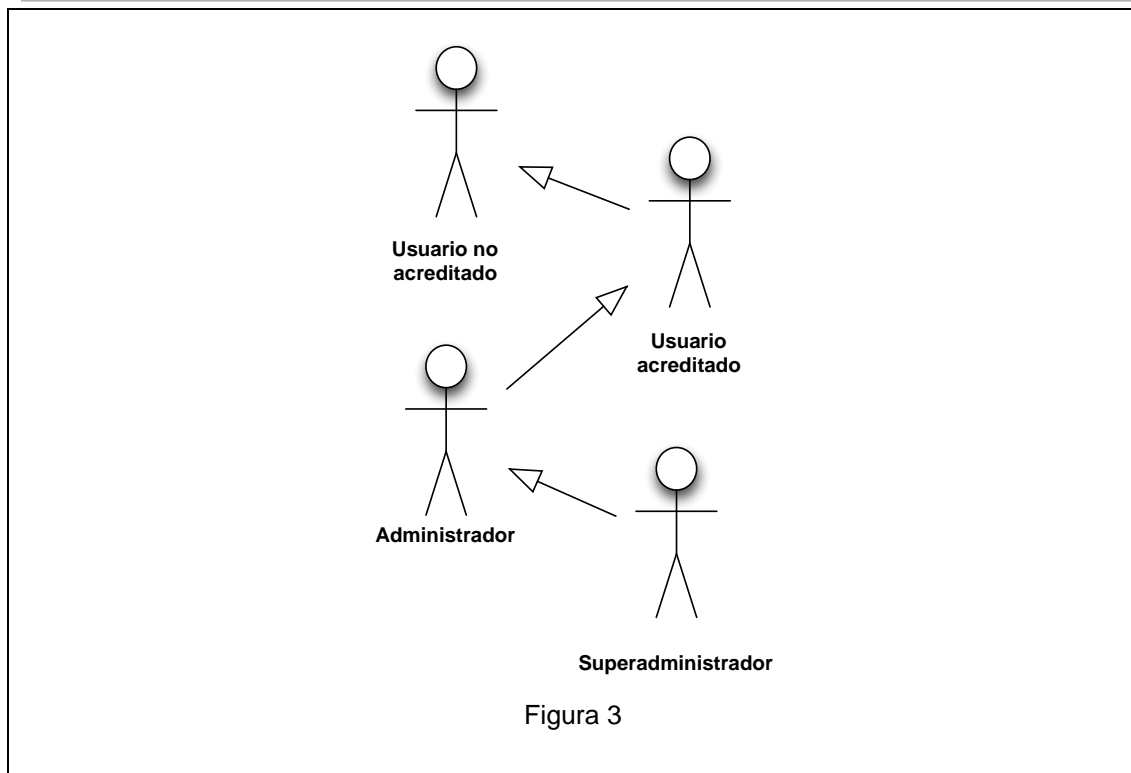
Actualmente no se definirán limitaciones a la hora de realizar reservas. Esto es que un usuario podrá realizar todas las reservas que necesite sin ningún límite.

## 4 Escenarios de uso

En esta sección del documento se describirán los tipos de usuarios del sistema (roles) definidos según la manera en que lo usan.

### 4.1 Perfiles de usuario

Los perfiles de usuario serán los siguientes:



#### 4.1.1 Usuario no identificado

Será todo aquel usuario que no tenga acreditación de acceso al sistema.

#### 4.1.2 Usuario del sistema

Será todo aquel usuario que tenga una acreditación valida de acceso al sistema.

#### 4.1.3 Administrador del sistema

Será todo aquel usuario que tenga una acreditación valida de acceso al sistema y además posea el rol de administrador.

#### 4.1.4 Superadministrador

Será todo aquel usuario que tenga una acreditación valida de acceso al sistema y además posea el rol de superadministrador.

### 4.2 Casos de uso.

Esta sección describe las maneras en las que los usuarios anteriormente mencionados utilizan este software. Los casos de uso se han codificado de la siguiente manera:

**CU-(Identificador subsistema).(Secuencia caso de uso).(Subcaso)**

De tal manera que se agruparán por subsistemas:

- Subsistema de conexión y usuarios (SGU) → CU-1.x.x

- Subsistema de Elementos Tipo (SGET) → CU-2.x.x
- Subsistema de Reservas (SGR) → CU-3.x.x
- Subsistema de Reservas Históricas (SGRH) → CU-4.x.x
- Subsistema de Programaciones (SGP) → CU-5.x.x

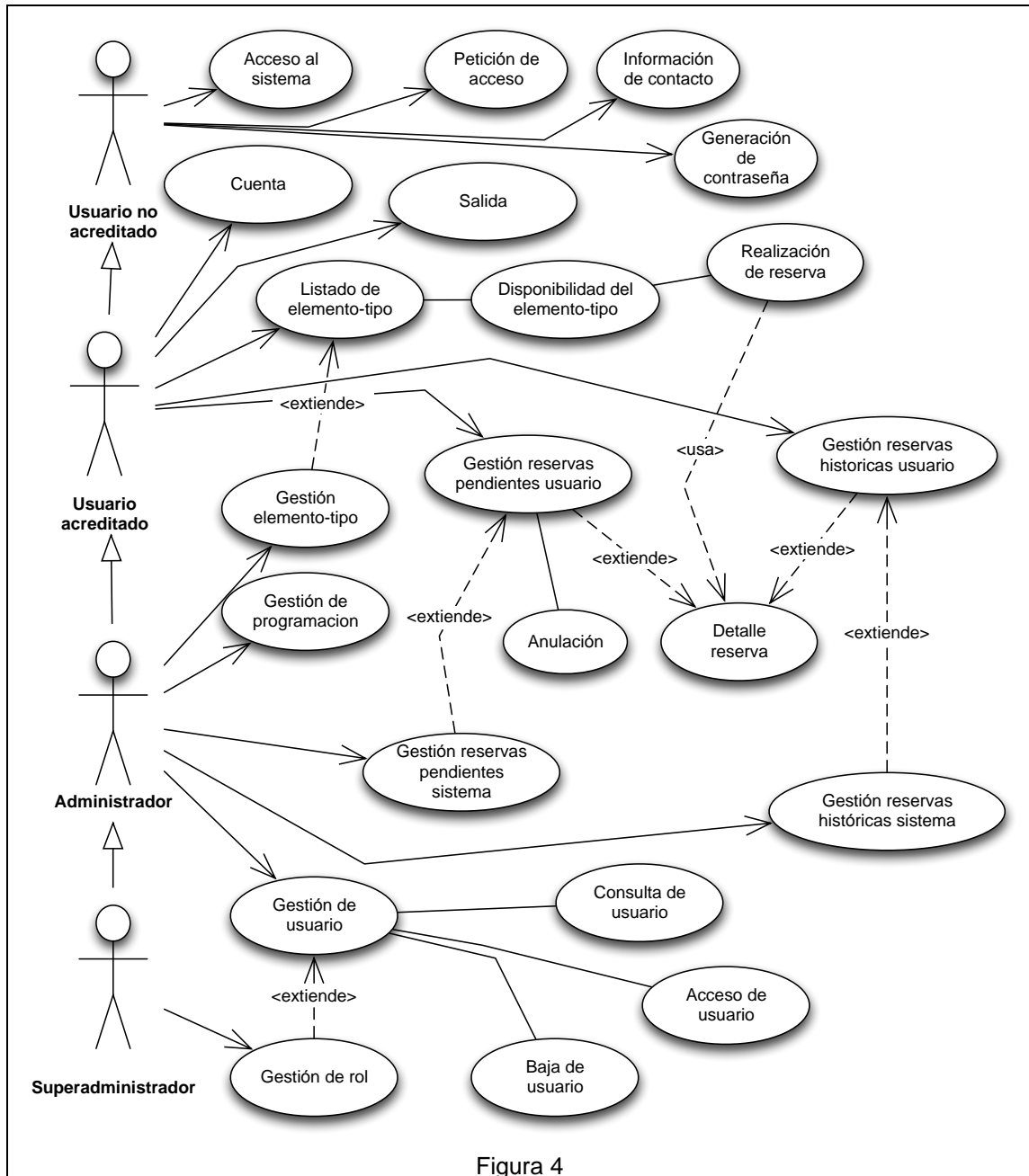


Figura 4

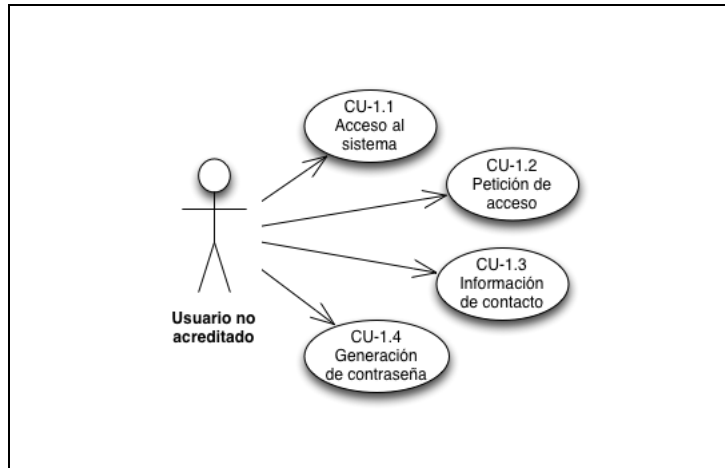
#### 4.2.1 Casos de uso de un usuario no identificado

Esta sección describe todos los posibles casos de uso de un usuario no identificado. Una vez instalado el sistema y configurado un superadministrador deberán darse de alta el resto de usuarios. El alta de usuarios se realizará previa solicitud de los mismos por lo que el sistema debe permitir acceder a usuarios no acreditados con el fin de permitirles hacer la solicitud de alta en el sistema. De la misma manera, se

deberá ofrecer acceso a la información de contacto de la administración de la aplicación.

Desde la pantalla de bienvenida estas serán las posibles operaciones:

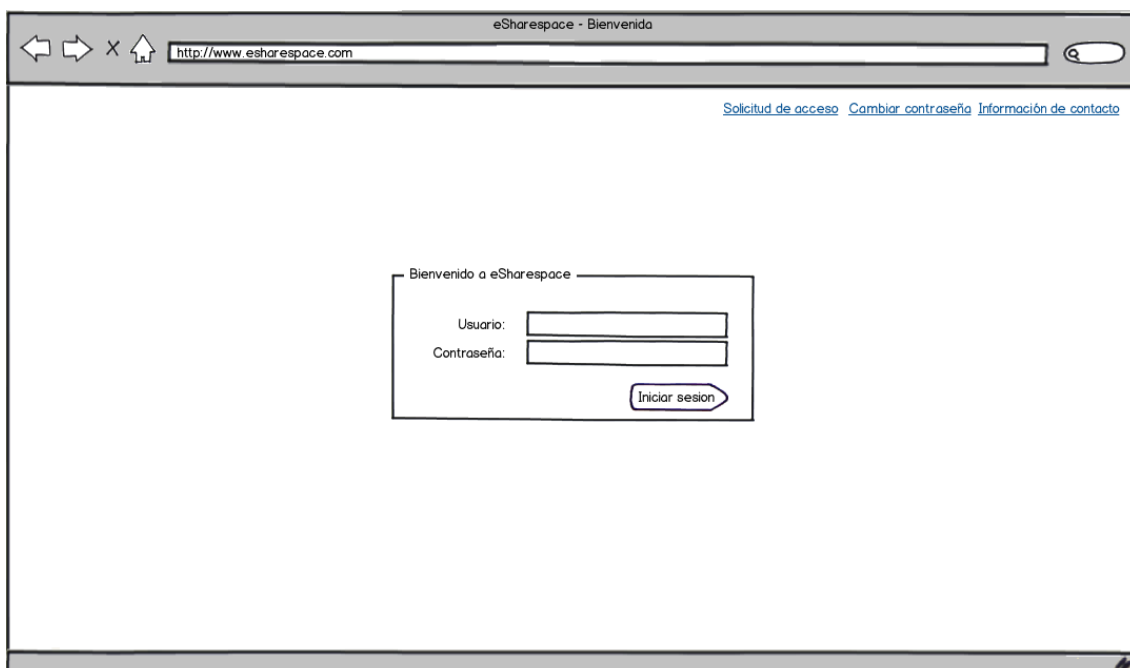
- CU-1.1 Acceso al sistema.
- CU-1.2 Petición de acceso.
- CU-1.3 Información de contacto.
- CU-1.4 Solicitud de cambio de contraseña.



#### 4.2.1.1 CU-1.1 - Acceso al sistema (pantalla de bienvenida)

Se permitirá el acceso al sistema a cualquier usuario presentando una pantalla sobre la que podrán acreditarse, registrarse y visualizar la información de contacto.

Este es un prototipo de la pantalla en cuestión:



#### 4.2.1.2 CU-1.2- Petición de acceso

Desde la pantalla de bienvenida se ofrecerá el acceso a la vista de petición de acceso donde se mostrará un formulario a rellenar por parte del usuario con los siguientes datos:

<b>Campo</b>	<b>Tipo</b>	<b>Significado</b>	<b>Obligatorio</b>
<b>Nombre</b>	Texto	Nombre del usuario	SI
<b>Apellidos</b>	Texto	Apellidos del usuario	SI
<b>Dirección</b>	Texto	Dirección del usuario	SI
<b>Población</b>	Texto	Población del usuario	SI
<b>Código postal</b>	Numérico	Código postal del usuario	SI
<b>Provincia</b>	Texto	Provincia del usuario	SI
<b>Email</b>	Texto	Dirección de correo electrónico del usuario (servirá de acceso a la aplicación)	SI
<b>Teléfono fijo</b>	Numérico	Número de teléfono del usuario	NO
<b>Teléfono móvil</b>	Numérico	Número de teléfono móvil del usuario	NO
<b>Contraseña</b>	Texto	Contraseña del usuario	SI
<b>Repetición de la contraseña</b>	Texto	Validación de la contraseña del usuario	SI

Cuando el usuario envíe el formulario de acceso se validará que los datos introducidos sean correctos y se informará de los errores detectados para su corrección.

Este es un prototipo de la pantalla en cuestión:

The screenshot shows a web browser window with the address bar containing 'http://www.esharespace.com/signup'. The page title is 'eSharespace - Solicitud de acceso'. The main content is a registration form titled 'Formulario de acceso' with the following fields: Nombre, Apellidos, Dirección, Población, Código Postal, email, Repita email, Teléfono fijo, Teléfono móvil, Contraseña, and Repita contraseña. A button labeled 'Enviar solicitud' is located at the bottom right of the form.

#### 4.2.1.3 CU-1.3 - Información de contacto

Desde la pantalla de bienvenida se podrá acceder a vista de información de contacto de la administración de la aplicación. En esta vista se mostrará la dirección, horario, teléfono e email de la administración de la aplicación.

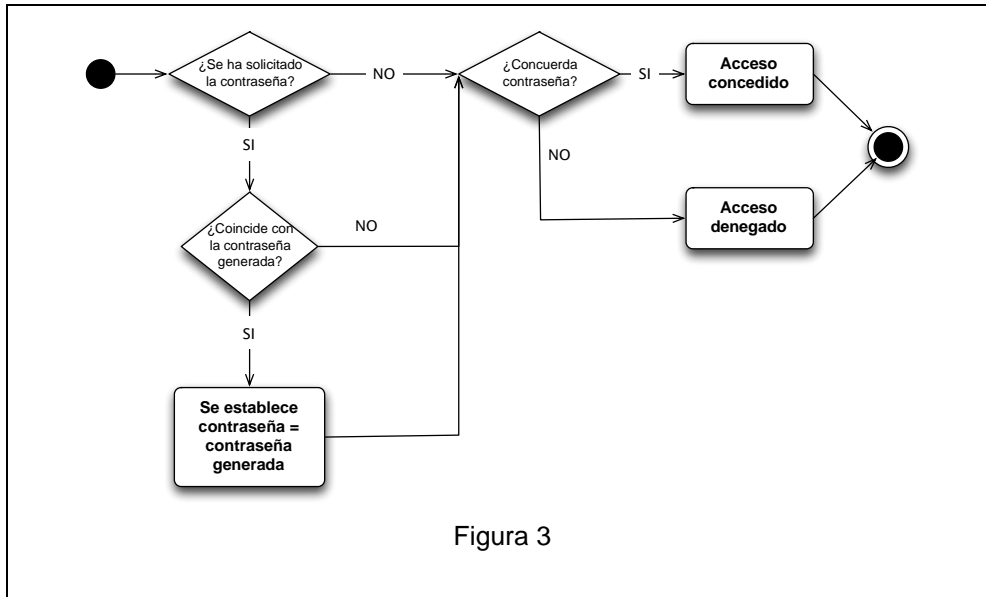
Este es un prototipo de la pantalla en cuestión:

The screenshot shows a web browser window with the address bar containing 'http://www.esharespace.com/contact'. The page title is 'eSharespace - Información de contacto'. The main content is a contact information box titled 'Información de contacto' with the following text: Dirección: C\ Blah Blah Blah 3, Ciudad (CP), Provincia; Teléfono: 999 99 99 99; FAX: 888 88 88 88; email: contacto@esharespace.com. A button labeled 'Volver a la página de bienvenida' is located at the bottom right of the box.

#### 4.2.1.4 CU-1.4 - Solicitud de cambio de contraseña

Desde la pantalla de bienvenida se ofrecerá el acceso a la vista de petición de cambio de contraseña. Esta vista permite que en caso que hayamos olvidado nuestra contraseña se envíe al usuario una contraseña generada con el fin de que pueda acceder a la aplicación. La vista solicitará la dirección de correo con la que

se dio el alta en el sistema para realizar la operación y la validará mediante un captcha. Una vez superado el captcha y se haya validado que existe un usuario con esa dirección de correo el sistema procederá a la generación de una contraseña aleatoria que se enviará por email al usuario. En este momento el usuario podrá acceder tanto con su antigua contraseña como con esta generada. En el momento que acceda al sistema con alguna de las dos contraseñas se establecerá como contraseña de usuario y se le presentará el menú principal. Este será el diagrama de actividad para esta validación:



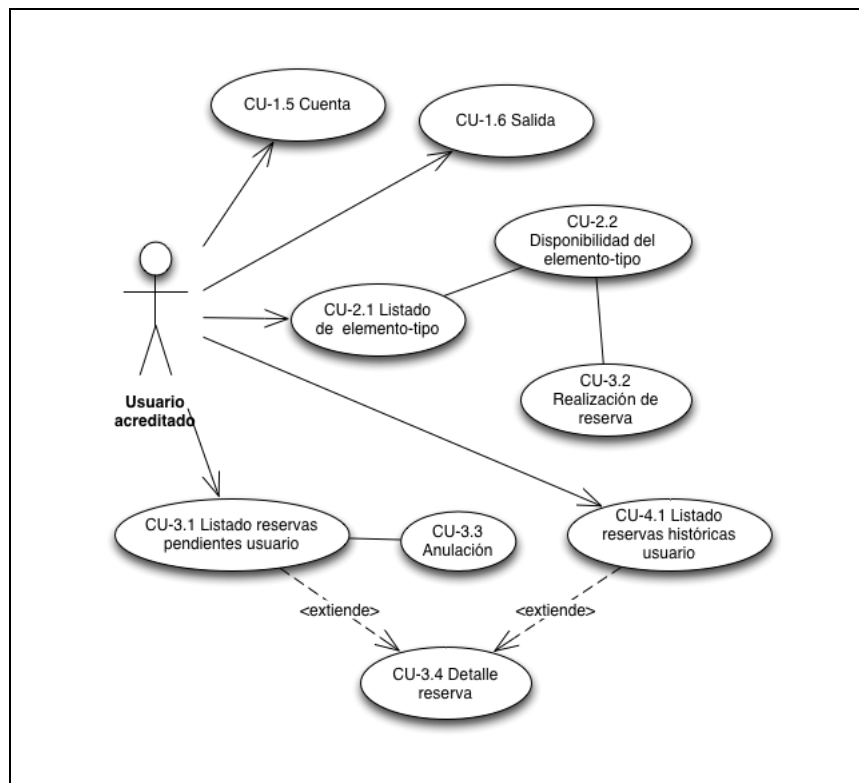
Este es un prototipo de la pantalla en cuestión:



### 4.2.2 Usuario del sistema

Esta sección describe todos los posibles casos de uso de un usuario con acreditación de acceso al sistema. Una vez hecha la petición de acceso, y validada por el administrador, el usuario podrá acceder desde la pantalla de bienvenida a la aplicación mediante su acreditación. Al introducir una acreditación correcta se iniciará la sesión del usuario y se le mostrará la pantalla de acceso que contiene un menú principal desde el que podrá acceder a las siguientes vistas y operaciones:

- CU-2.1 - Listado de elementos-tipo del usuario.
- CU-2.2 - Disponibilidad de un elemento.
- CU-3.1 - Gestión de reservas pendientes del usuario.
- CU-3.2 - Reserva de un elemento.
- CU-3.3 - Anulación de una reserva.
- CU-3.4 - Detalle de una reserva.
- CU-4.1 - Gestión de reservas históricas del usuario.
- CU-1.5 - Cuenta.
- CU-1.6 - Salida del sistema.



#### 4.2.2.1 CU-2.1 - Listado de elementos-tipo del usuario

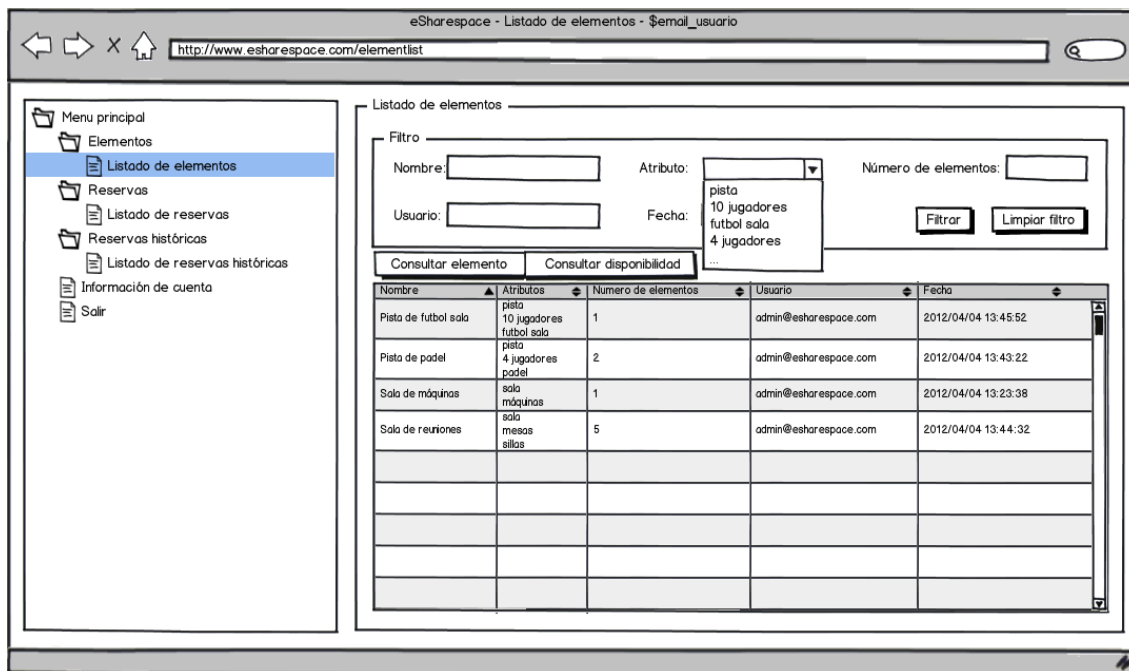
Desde el menú principal el usuario podrá acceder a la vista del listado de elementos-tipo del sistema donde se le mostrará una barra de herramientas y una tabla con todos los elementos-tipo del sistema que dispongan de al menos un elemento definido con una programación vigente. La tabla mostrará los siguientes datos de cada elemento-tipo:

Atributo	Significado
----------	-------------



<b>Nombre</b>	Nombre del elemento-tipo.
<b>Atributos</b>	Atributos del elemento-tipo.
<b>Elementos</b>	Número de elementos definidos bajo este tipo.

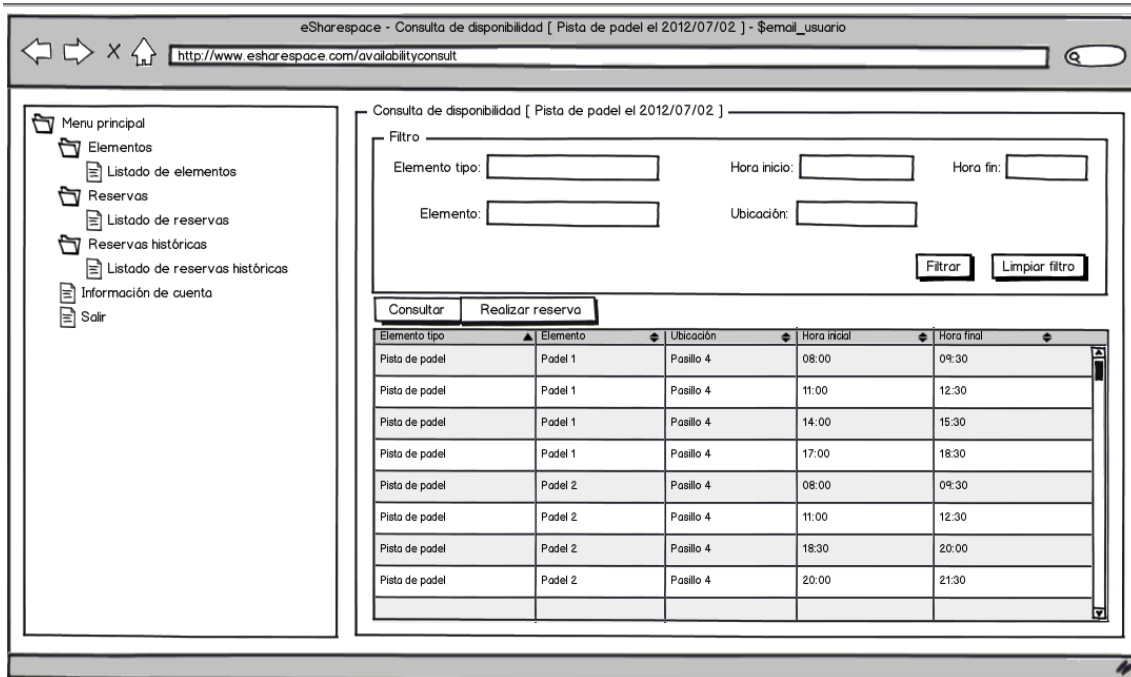
Este es un prototipo de la pantalla en cuestión:



Desde la barra de herramientas, y previa selección de un elemento tipo, el usuario podrá realizar las siguientes operaciones:

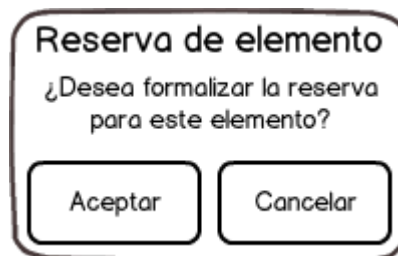
#### 4.2.2.1.1 CU-2.2 - Disponibilidad de un elemento

Desde la barra de herramientas de la vista del listado de elementos-tipo el usuario podrá invocar la disponibilidad de un elemento-tipo para un día concreto mayor o igual a la fecha actual. De esta manera tendrá acceso a una vista donde se mostrará la disponibilidad de todos los elementos de ese tipo para ese día en concreto. Sobre esta vista podrá realizar la reserva de un elemento en una de sus ranuras disponibles.



#### 4.2.2.1.2 CU-3.2 - Reserva de un elemento

Desde la vista de disponibilidad de un elemento-tipo para un día en concreto se permitirá que el usuario realice una reserva sobre un elemento con una ranura disponible. El sistema comprobará que la reserva sea posible y la guardará en el sistema notificando al usuario la reserva por email o por sms en caso que se disponga de su número de teléfono móvil. El sistema presentará el detalle de la reserva al usuario.



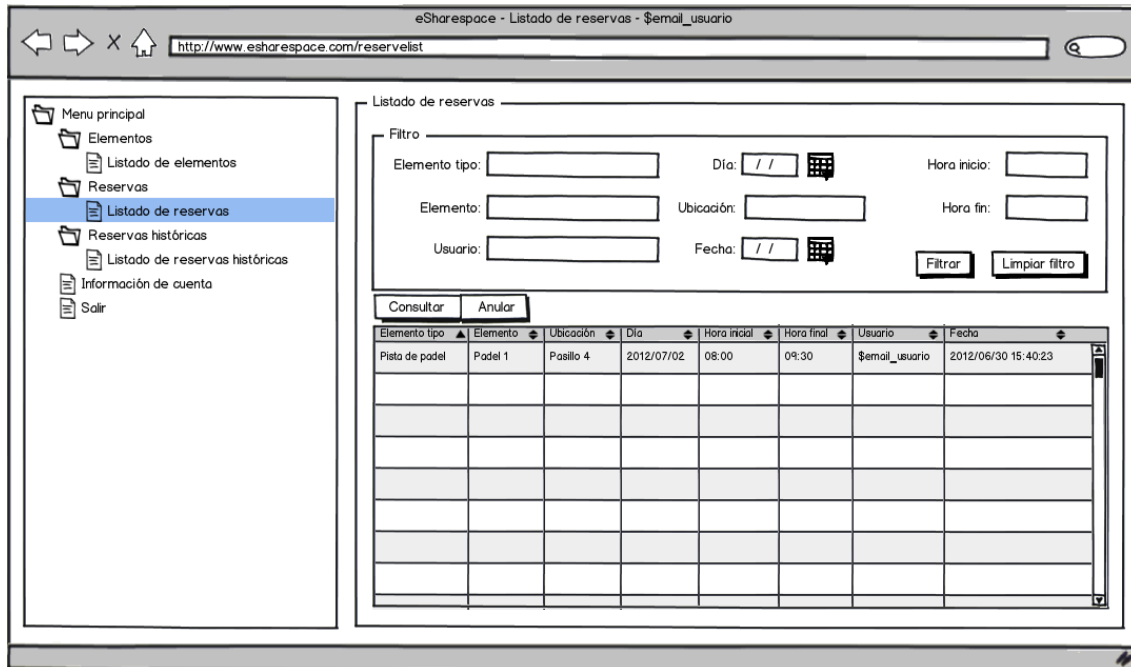
#### 4.2.2.2 CU-3.1 - Gestión de reservas pendientes del usuario

Desde el menú principal el usuario podrá acceder a la vista del listado de reservas pendientes. En este listado en modo de tabla se mostrarán los siguientes datos de la reserva:

Atributo	Significado
<b>Elemento-tipo</b>	Elemento-tipo sobre el que se hizo la reserva.
<b>Elemento</b>	Elemento reservado.
<b>Ubicación</b>	Ubicación del elemento.
<b>Día</b>	Día de la reserva.
<b>Hora inicial</b>	Hora inicial de la reserva
<b>Hora final</b>	Hora final de la reserva

<b>Fecha</b>	Fecha en que se realizó la reserva
--------------	------------------------------------

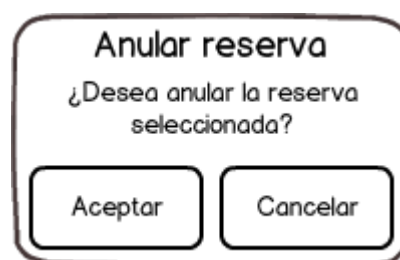
Este es un prototipo de la pantalla en cuestión:



Desde esta vista el sistema permitirá la siguiente operación

#### 4.2.2.2.1 CU-3.3 - Anulación de una reserva

El usuario desde la vista de reservas pendientes podrá, previa selección de una reserva, proceder a la anulación de reservas. Esta operación requiere de la confirmación del usuario. Una vez eliminada la reserva el sistema notificará de la anulación al usuario que la solicitó por email o sms si se dispone del número de teléfono móvil del usuario.



#### 4.2.2.2.2 CU-3.4 – Detalle de una reserva

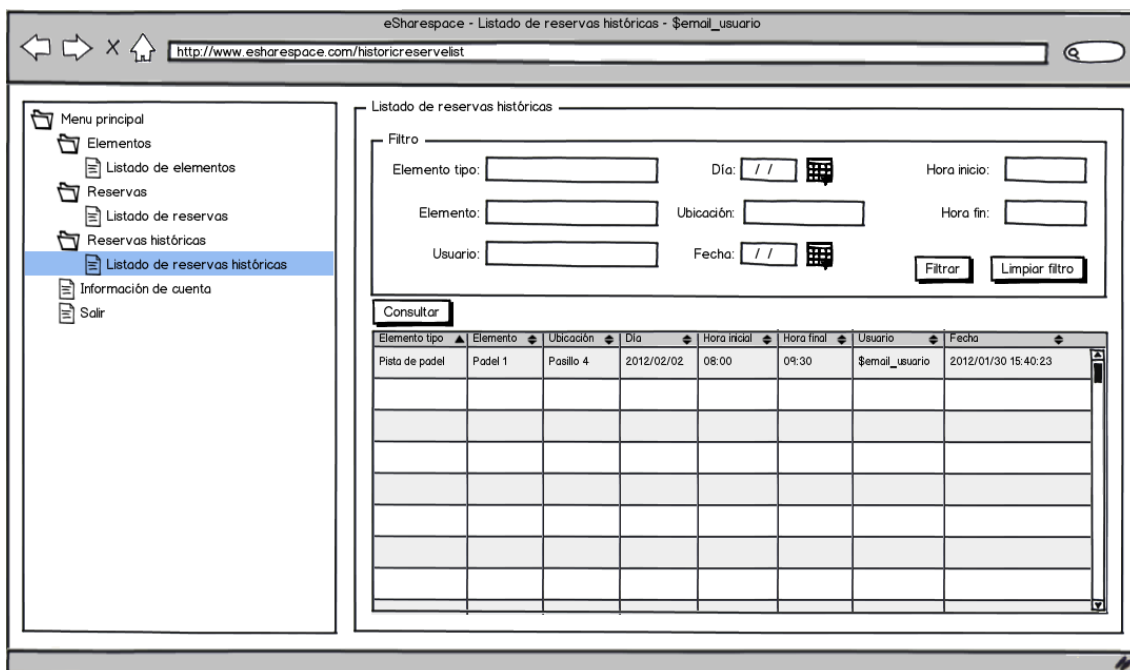
Todos los atributos de la reserva son visibles desde el listado de reservas por lo que no será necesario aportar más información acerca de la reserva en una nueva vista.

#### 4.2.2.3 CU-4.1 - Gestión de reservas históricas del usuario

Desde el menú principal el usuario podrá acceder a la vista del listado de reservas históricas. En este listado en modo de tabla se mostrarán los siguientes datos de la reserva:

Atributo	Significado
<b>Elemento-tipo</b>	Elemento-tipo sobre el que se hizo la reserva.
<b>Elemento</b>	Elemento reservado.
<b>Ubicación</b>	Ubicación del elemento.
<b>Día</b>	Día de la reserva.
<b>Hora inicial</b>	Hora inicial de la reserva
<b>Hora final</b>	Hora final de la reserva
<b>Fecha</b>	Fecha en que se realizó la reserva

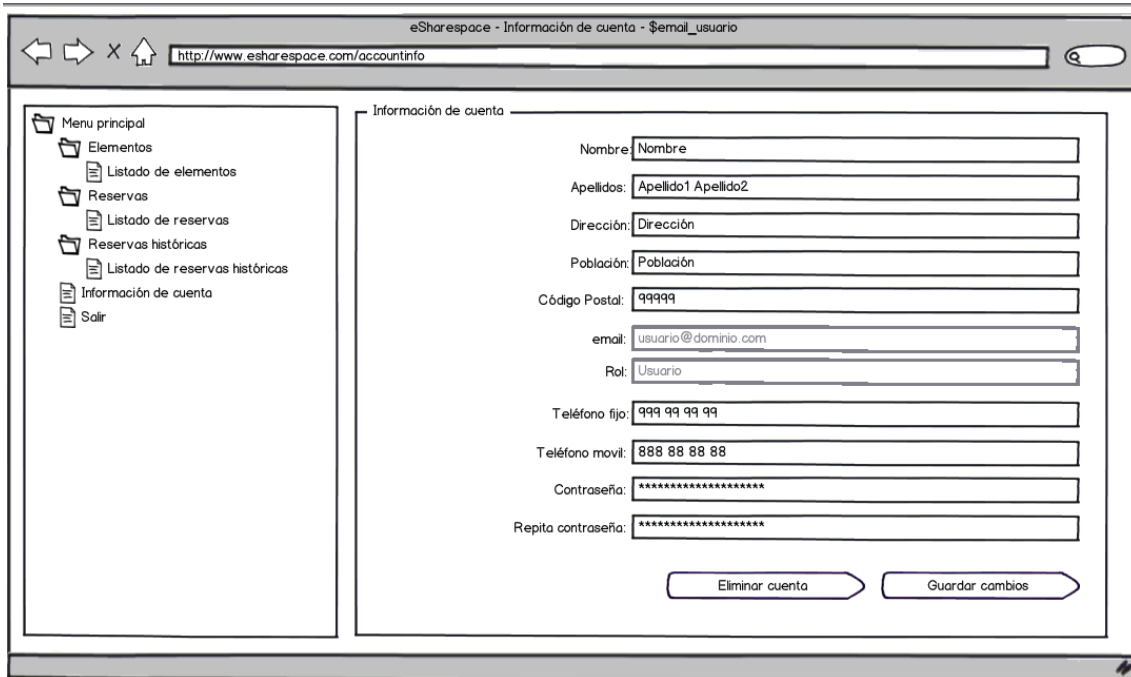
Este es un prototipo de la pantalla en cuestión:



#### 4.2.2.4 CU-1.5 - Cuenta

Desde el menú principal el usuario podrá acceder a esta vista desde la que podrá modificar sus datos de usuario o darse de baja en la aplicación.

Este es un prototipo de la pantalla en cuestión:



The screenshot shows a web browser window with the URL <http://www.esharespace.com/accountinfo>. The page title is 'eSharespace - Información de cuenta - \$email\_usuario'. On the left is a 'Menu principal' with items: Elementos, Listado de elementos, Reservas, Listado de reservas, Reservas históricas, Listado de reservas históricas, Información de cuenta, and Salir. The main content area is titled 'Información de cuenta' and contains the following form fields:

- Nombre:
- Apellidos:
- Dirección:
- Población:
- Código Postal:
- email:
- Rol:
- Teléfono fijo:
- Teléfono móvil:
- Contraseña:
- Repita contraseña:

At the bottom of the form are two buttons: 'Eliminar cuenta' and 'Guardar cambios'.

#### 4.2.2.4.1 CU-1.5.1 - Modificación de los datos de usuario

Desde la vista de cuenta el usuario podrá visualizar todos sus datos de usuario y modificar los siguientes datos:

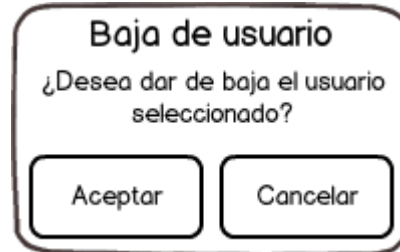
Campo	Tipo	Significado
<b>Nombre</b>	Texto	Nombre del usuario
<b>Apellidos</b>	Texto	Apellidos del usuario
<b>Dirección</b>	Texto	Dirección del usuario
<b>Población</b>	Texto	Población del usuario
<b>Código postal</b>	Numérico	Código postal del usuario
<b>Provincia</b>	Texto	Provincia del usuario
<b>Teléfono fijo</b>	Numérico	Número de teléfono del usuario
<b>Teléfono móvil</b>	Numérico	Número de teléfono móvil del usuario
<b>Contraseña anterior</b>	Texto	Requerido para el cambio de contraseña. Deberá informar la contraseña anterior.
<b>Nueva contraseña</b>	Texto	Requerido para el cambio de contraseña. Deberá informar la nueva contraseña
<b>Repetición de la nueva contraseña</b>	Texto	Requerido para el cambio de contraseña. Validación de la nueva contraseña

El usuario así podrá modificar sus datos o su contraseña de manera independiente.

#### 4.2.2.4.2 CU-1.5.2 - Baja de usuario

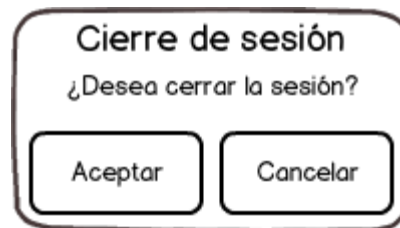
Desde la vista de cuenta se le ofrecerá al usuario la posibilidad de darse de baja en la aplicación. El usuario deberá confirmar esta operación que provocará que se

eliminen todos sus datos de cuenta y se eliminen las reservas pendientes. No se eliminarán sin embargo los datos de reservas históricos que deberán mantenerse para proporcionar datos estadísticos. Una vez realizada la baja del usuario se notificará por correo de tal efecto y de las reservas que se han anulado.



#### 4.2.2.5 CU-1.6 - Salida del sistema

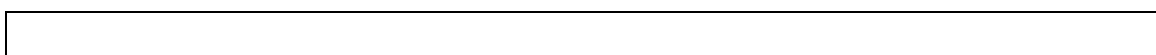
Desde el menú principal el usuario podrá invocar la salida del sistema lo cual provocará, previa confirmación, que se cierre la sesión del usuario. Una vez cerrada la sesión se le presentará la pantalla de bienvenida. Si el usuario deseara volver a operar con el sistema deberá tener que volver a introducir sus credenciales de usuario.

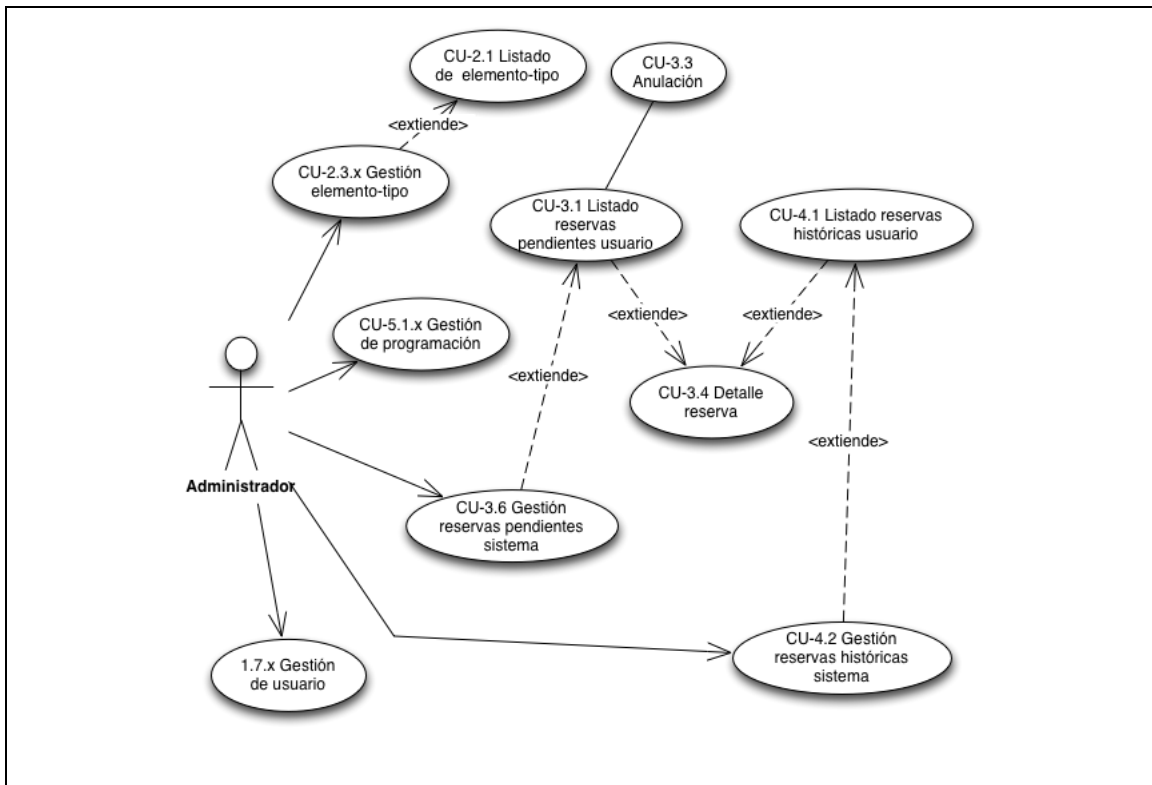


#### 4.2.3 Administrador del sistema

Esta sección describe todos los posibles casos de uso de un usuario con acreditación de acceso al sistema y que a su vez tiene el rol de administrador. El rol de administrador será concedido por parte del superadministrador. Cualquier usuario del sistema podrá tener el rol de administrador del sistema, solo requiere que el superadministrador se lo conceda. Sí es necesario que el usuario este previamente dado de alta en el sistema. Una vez acreditado un usuario con rol de administrador el sistema le presentará un menú que le permitirá acceder a las siguientes vistas y operaciones:

- CU-5.1 - Gestión de programaciones.
- CU-2.3 - Gestión de elementos-tipo del sistema.
- CU-1.7 - Gestión de usuarios.
- CU-3.6 - Gestión de reservas pendientes del sistema.
- CU-4.2 - Gestión de reservas históricas del sistema.
- CU-1.5 - Cuenta.
- CU-1.6 - Salida del sistema.





#### 4.2.3.1 CU-5.1 - Gestión de programaciones

Desde el menú principal el administrador podrá acceder a la vista de gestión de programaciones. Esta vista mostrará una barra de herramientas y una tabla sobre la que aparecerán las programaciones definidas en el sistema.

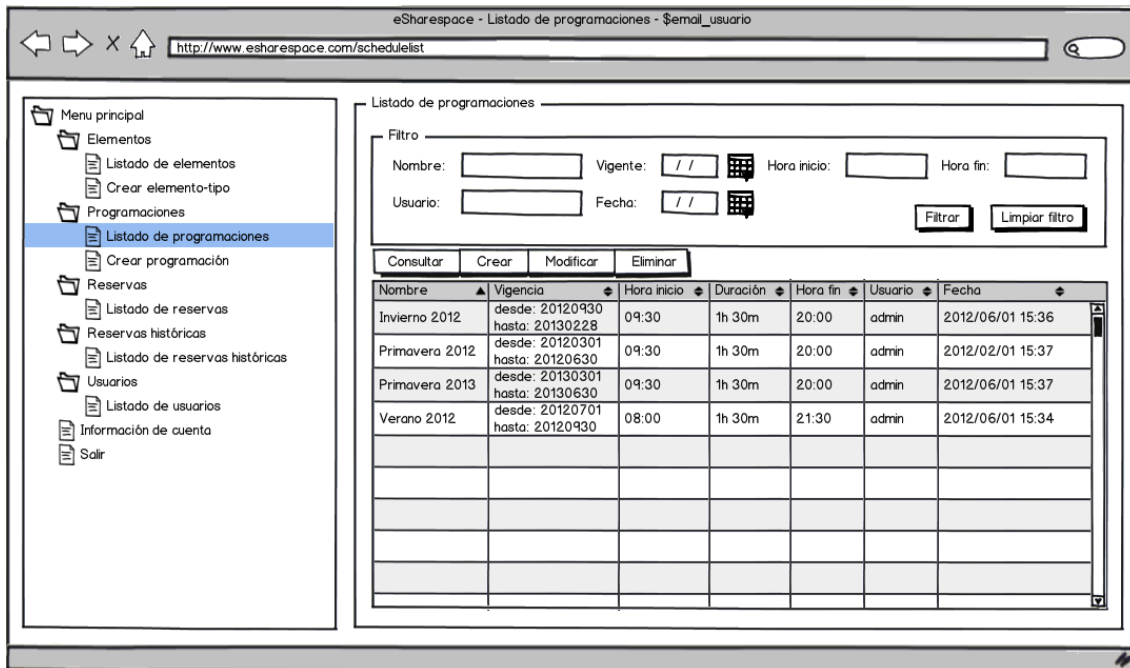
Esta tabla mostrará los siguientes datos de las programaciones del sistema.

Atributo	Significado
<b>Nombre</b>	Nombre de la programación.
<b>Vigencia</b>	Vigencia de la programación.
<b>Hora inicio</b>	Hora de inicio de la programación.
<b>Duración</b>	Duración de la programación.
<b>Hora fin</b>	Hora de finalización de la programación.
<b>Usuario</b>	Usuario de última modificación.
<b>Fecha</b>	Fecha de última modificación.

Desde la barra de herramientas se podrán invocar las siguientes acciones:

- Creación de una programación.
- Modificación de una programación.
- Consulta de una programación.
- Eliminación de una programación.

Este es un prototipo de la pantalla en cuestión:



#### 4.2.3.1.1 CU-5.1.1 - Creación de una programación.

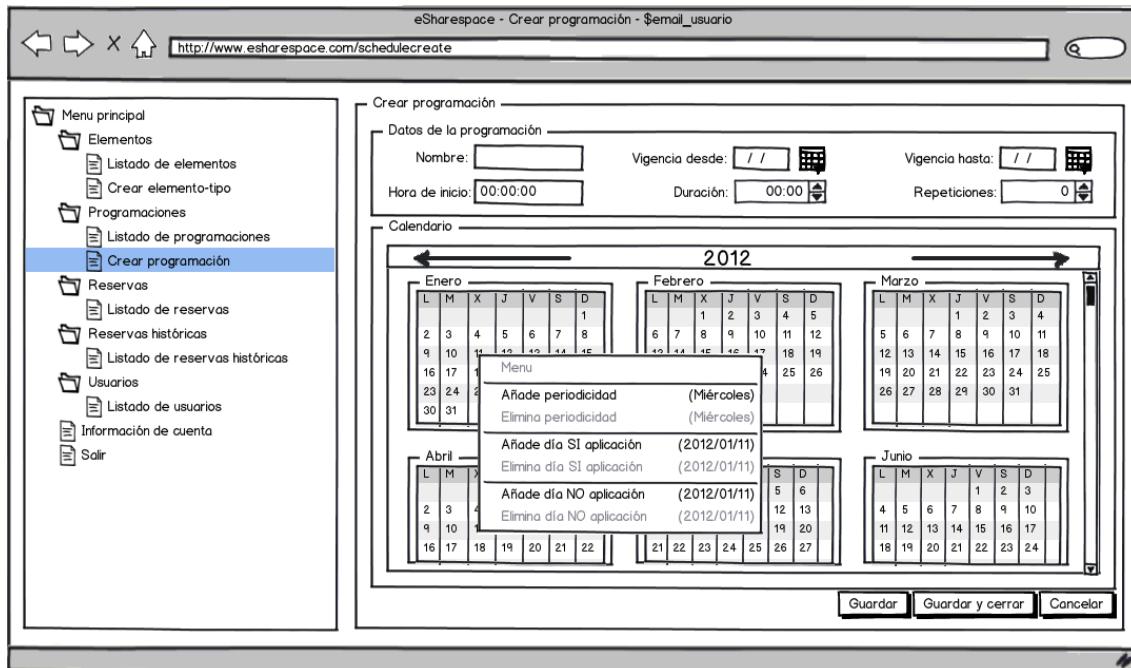
Desde la barra de herramientas de la vista de gestión de programaciones se podrá invocar la creación de una nueva programación. Esta acción permitirá que se acceda a la vista de creación donde se podrá definir una programación. Una programación consta de los siguientes atributos:

Campo	Tipo	Significado	Obligatorio
<b>Nombre</b>	Texto	Nombre de la programación. Deberá ser único para el conjunto de programaciones del sistema.	SI
<b>Calendario</b>	Compuesto	Calendario de la programación. Consta de periodicidad diaria [L, M, X, J, V, S, D] y de días de SI aplicación y días de NO aplicación. Los días de SI y NO aplicación deben encontrarse dentro de la vigencia definida, de lo contrario no tendrán efecto.	SI
<b>Vigencia</b>	Compuesto	Vigencia de la programación. Determina la fecha desde y hasta en que esta programación es efectiva. La fecha desde es obligatoria y debe ser mayor que la fecha actual. La fecha hasta puede dejarse sin definir determinando así que no expira su vigencia.	SI
<b>Hora inicio</b>	Tiempo	Hora de inicio de la programación. Deberá ser positivo y menor que 24:00.	SI
<b>Duración</b>	Numérico	Duración en minutos de la reserva. Deberá	SI



		ser mayor que 0.	
<b>Repetición</b>	Numérico	Indica las veces que se repetirá la reserva a partir de la hora de inicio y contando la duración de la misma.	SI

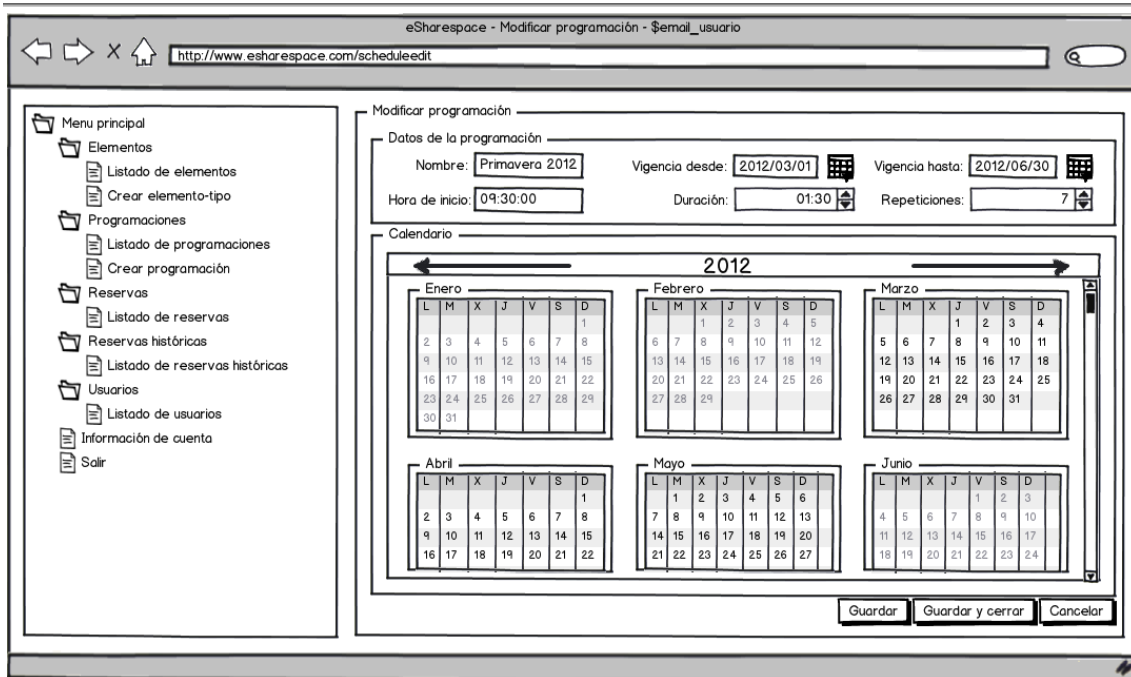
Este es un prototipo de la pantalla en cuestión:



#### 4.2.3.1.2 CU-5.1.2 - Modificación de una programación.

Desde la barra de herramientas de la vista de gestión de programaciones se podrá invocar la modificación de una programación seleccionada en la tabla. Esta acción permitirá que se acceda a la vista de modificación. Esta vista es idéntica a la de creación pero en vez de aparecer los campos vacíos aparecerán rellenos con la programación a editar.

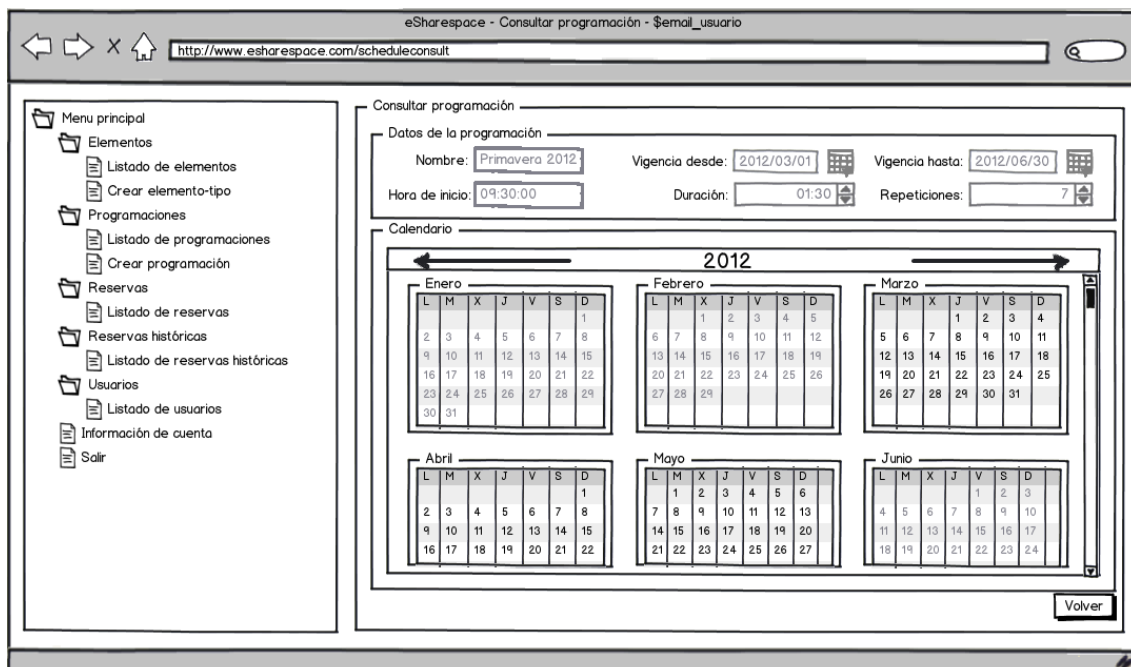
Este es un prototipo de la pantalla en cuestión:



#### 4.2.3.1.3 CU-5.1.3 - Consulta de una programación.

Desde la barra de herramientas de la vista de gestión de programaciones se podrá invocar la consulta de una programación seleccionada en la tabla. Esta acción permitirá que se acceda a la vista de consulta. Esta vista es idéntica a la de creación pero no se permitirá la edición de ningún campo.

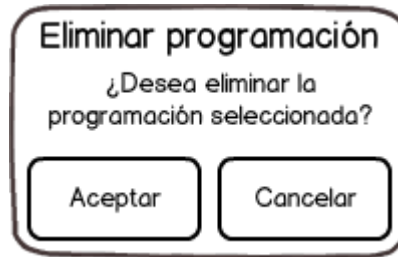
Este es un prototipo de la pantalla en cuestión:



#### 4.2.3.1.4 CU-5.1.4 - Eliminación de una programación.

Desde la barra de herramientas de la vista de gestión de programaciones se podrá invocar la eliminación de una programación seleccionada en la tabla. Esta acción

solicitará confirmación al usuario y no permitirá el borrado de la programación si está en uso por algún Elemento del sistema.



#### 4.2.3.2 CU-2.3 - Gestión de elementos-tipo del sistema

Desde el menú principal el administrador podrá acceder a la vista de gestión de elementos-tipo. Esta vista mostrará una barra de herramientas y una tabla sobre la que aparecerán los elementos-tipo definidos en el sistema.

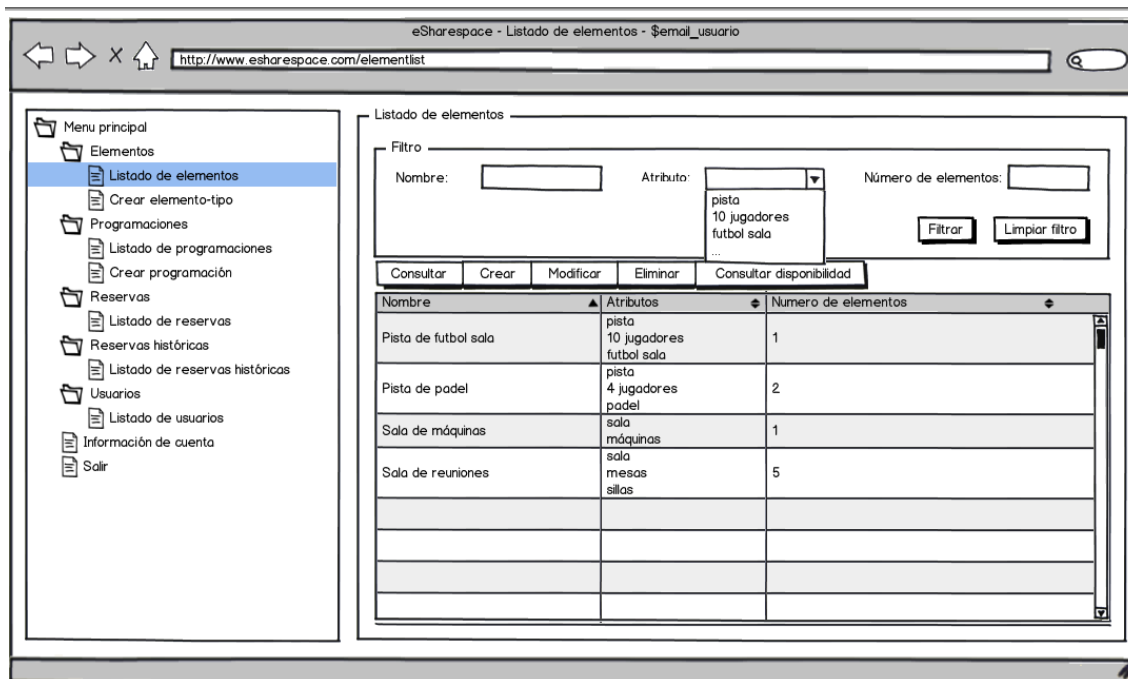
Esta tabla mostrará los siguientes datos de los elementos-tipo del sistema:

Atributo	Significado
<b>Nombre</b>	Nombre del elemento-tipo.
<b>Atributos</b>	Atributos del elemento-tipo.
<b>Elementos</b>	Número de elementos definidos bajo este tipo.
<b>Usuario</b>	Usuario de última modificación.
<b>Fecha</b>	Fecha de última modificación.

Desde la barra de herramientas se podrán invocar las siguientes acciones:

- Creación de un elemento-tipo.
- Modificación de un elemento-tipo.
- Consulta de un elemento-tipo.
- Eliminación de un elemento-tipo.

Este es un prototipo de la pantalla en cuestión:

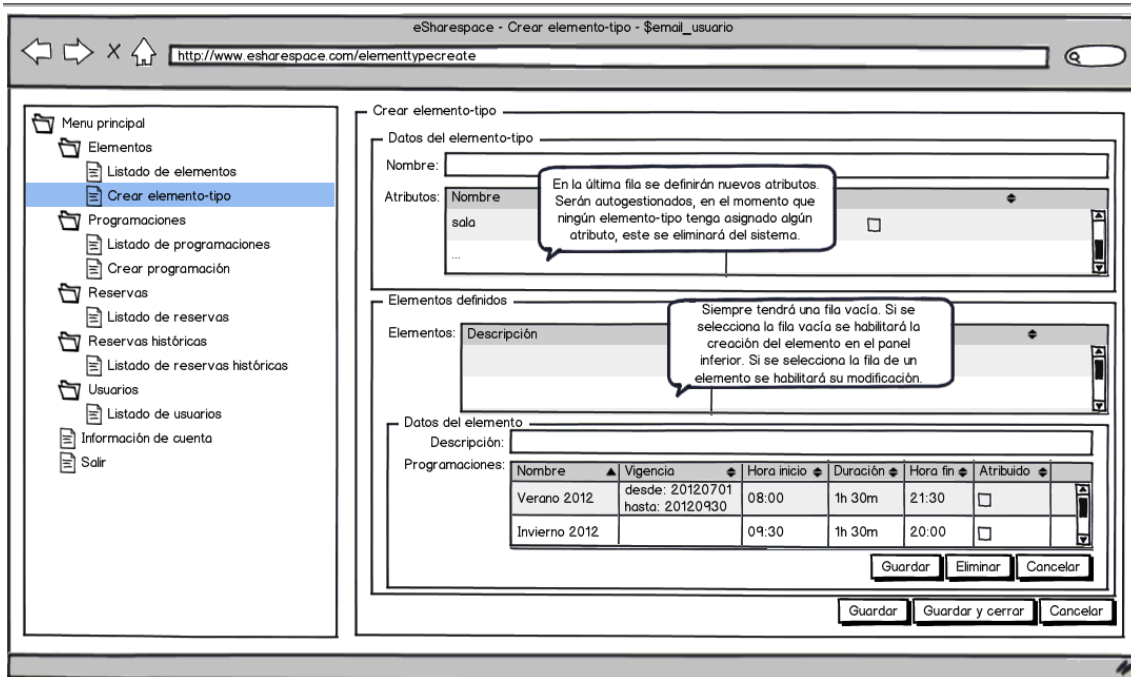


**4.2.3.2.1 CU-2.3.1 - Creación de un elemento-tipo.**

Desde la barra de herramientas de la vista de gestión de elementos-tipo se podrá invocar la creación de un nuevo elemento-tipo. Esta acción permitirá que se acceda a la vista de creación donde se podrá definir un elemento-tipo. Un elemento tipo consta de los siguientes atributos:

Campo	Tipo	Significado	Obligatorio
<b>Nombre</b>	Texto	Nombre del elemento-tipo. Deberá ser único para el conjunto de elementos-tipo del sistema.	SI
<b>Atributos</b>	Compuesto	Atributos del elemento-tipo. Son etiquetas que definen al elemento-tipo.	SI
<b>Elemento</b>	Compuesto	Elemento definido bajo este elemento-tipo. Se trata de la instanciación del elemento-tipo. Deberá definirse una descripción del elemento en forma de texto y las programaciones a las que atenderá. No se podrán definir programaciones que se solapen.	NO

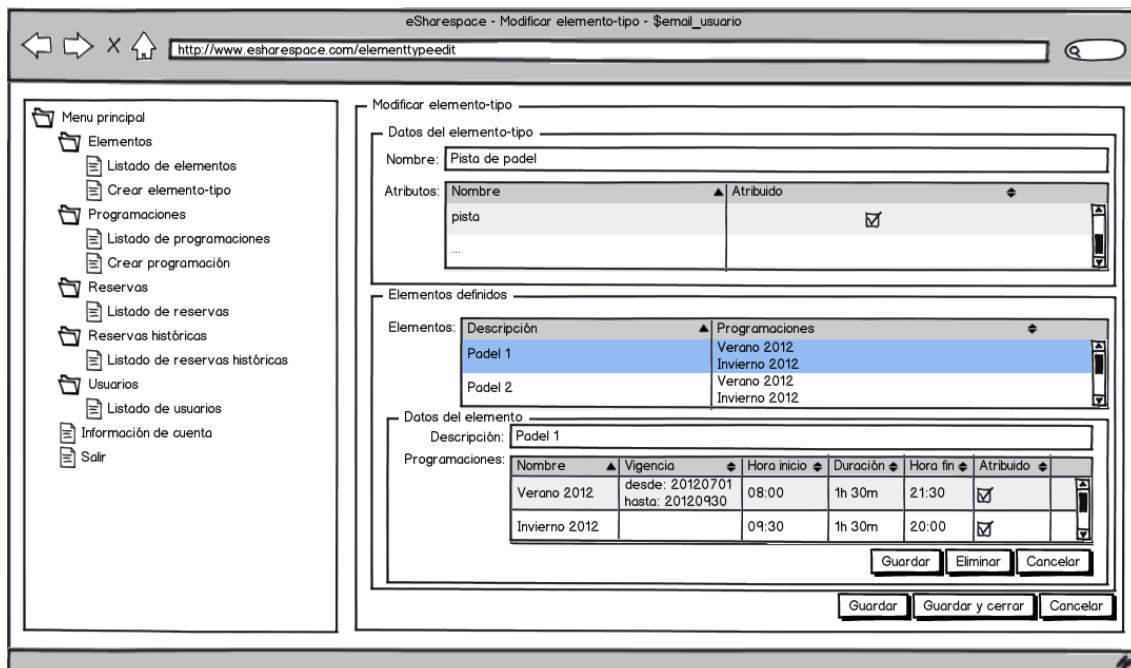
Este es un prototipo de la pantalla en cuestión:



#### 4.2.3.2.2 CU-2.3.2 - Modificación de un elemento-tipo.

Desde la barra de herramientas de la vista de gestión de elementos-tipo se podrá invocar la modificación de un elemento-tipo seleccionado en la tabla. Esta acción permitirá que se acceda a la vista de modificación. Esta vista es idéntica a la de creación pero en vez de aparecer los campos vacíos aparecerán rellenos con el elemento-tipo a editar.

Este es un prototipo de la pantalla en cuestión:

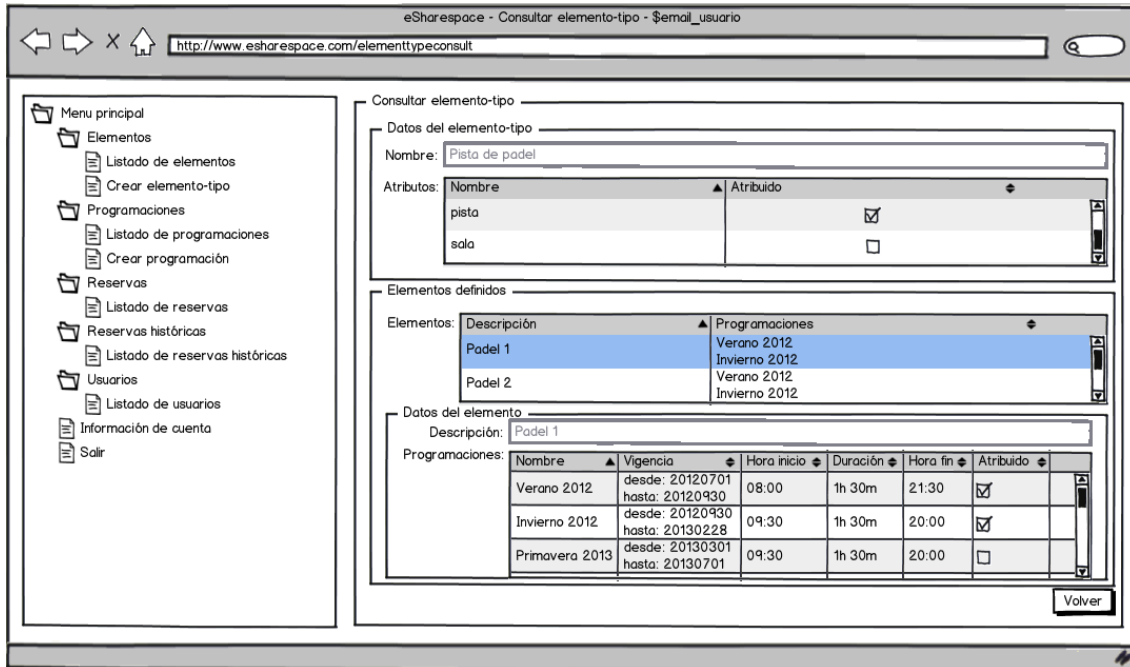


#### 4.2.3.2.3 CU-2.3.3 - Consulta de un elemento-tipo.

Desde la barra de herramientas de la vista de gestión de elementos-tipo se podrá invocar la consulta de un elemento-tipo seleccionado en la tabla. Esta acción

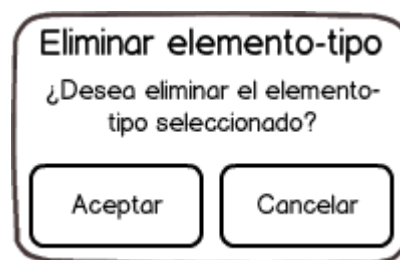
permitirá que se acceda a la vista de consulta. Esta vista es idéntica a la de creación pero no se permitirá la edición de ningún campo.

Este es un prototipo de la pantalla en cuestión:



#### 4.2.3.2.4 CU-2.3.4 - Eliminación de un elemento-tipo.

Desde la barra de herramientas de la vista de gestión de elementos-tipo se podrá invocar la eliminación de un elemento-tipo seleccionado en la tabla. Esta acción solicitará confirmación al usuario. En caso de que alguno de los Elementos del elemento-tipo a eliminar tenga una reserva planificada el sistema informará de tal caso, avisará que se anularán todas las reservas y pedirá confirmación de nuevo para el borrado.



#### 4.2.3.3 CU-1.7 - Gestión de usuarios

Desde el menú principal el administrador podrá acceder a la vista de gestión de usuarios. Esta vista mostrará una barra de herramientas y una tabla sobre la que aparecerán los usuarios definidos en el sistema.

Esta tabla mostrará los siguientes datos de los usuarios del sistema:

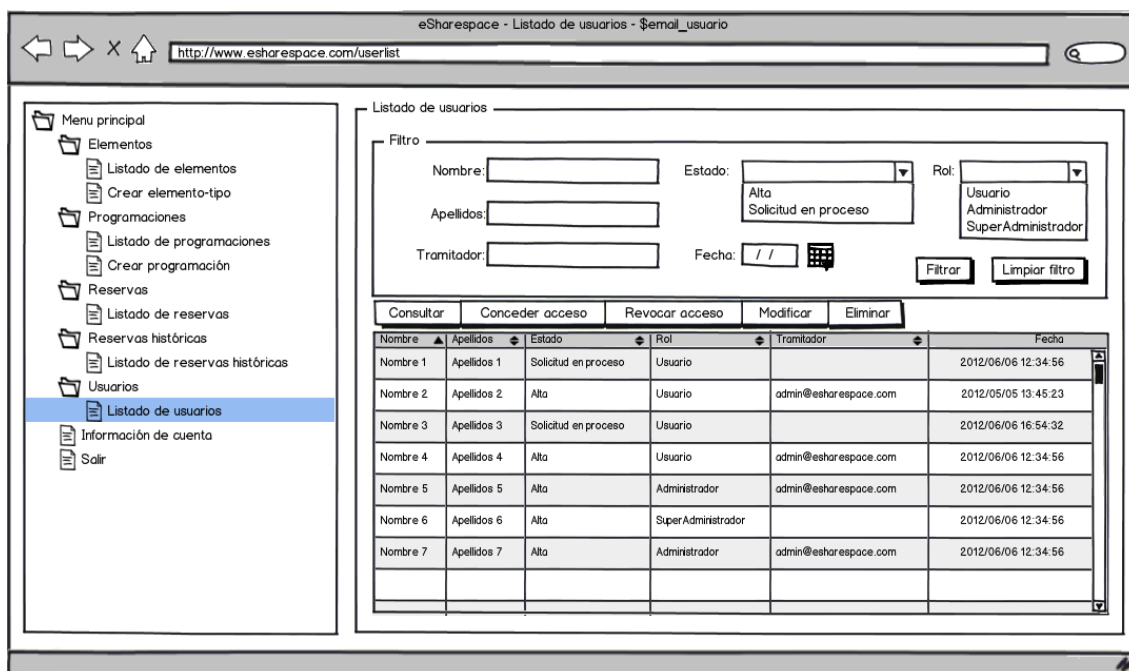
Atributo	Significado
<b>Nombre</b>	Nombre del usuario.
<b>Apellidos</b>	Apellidos del usuario.

<b>Estado</b>	Estado del usuario. El estado será uno de los siguientes: <ul style="list-style-type: none"> <li>• Solicitud en proceso.</li> <li>• Alta.</li> </ul>
<b>Rol</b>	Rol del usuario. Podrá ser uno de los siguientes: <ul style="list-style-type: none"> <li>• Usuario.</li> <li>• Administrador.</li> <li>• Superadministrador.</li> </ul>
<b>Fecha</b>	Fecha de ingreso al sistema.
<b>Tramitador</b>	Usuario que tramitó su solicitud de ingreso.

Desde la barra de herramientas se podrán invocar las siguientes acciones:

- Consulta de un usuario.
- Concesión de acceso.
- Revocación de acceso.
- Baja del usuario.

Este es un prototipo de la pantalla en cuestión:



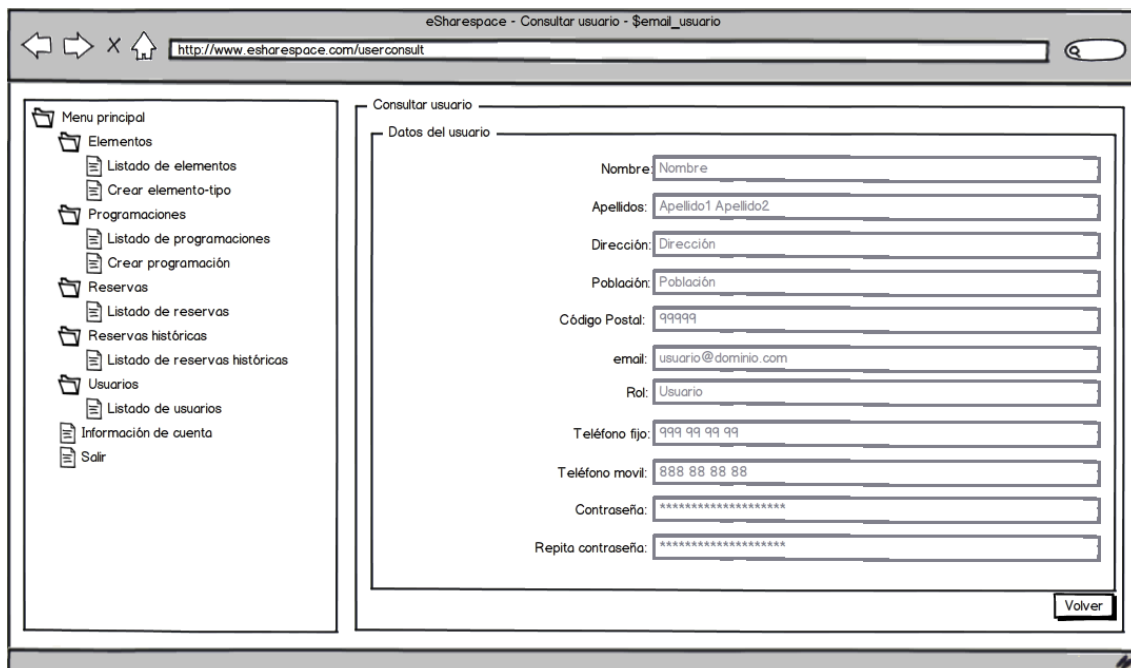
#### 4.2.3.3.1 CU-1.7.1 - Consulta de un usuario.

Desde la barra de herramientas de la vista de gestión de usuarios se podrá invocar la consulta de un usuario seleccionado en la tabla. Esta acción permitirá que se acceda a la vista de consulta del usuario que tendrá una barra de herramientas y se podrá visualizar los siguientes atributos del usuario:

Campo	Significado
Nombre	Nombre del usuario

<b>Apellidos</b>	Apellidos del usuario
<b>Rol</b>	Rol del usuario
<b>Dirección</b>	Dirección del usuario
<b>Población</b>	Población del usuario
<b>Código postal</b>	Código postal del usuario
<b>Provincia</b>	Provincia del usuario
<b>Email</b>	Dirección de correo electrónico del usuario (servirá de acceso a la aplicación)
<b>Teléfono fijo</b>	Número de teléfono del usuario
<b>Teléfono móvil</b>	Número de teléfono móvil del usuario

Este es un prototipo de la pantalla en cuestión:



#### 4.2.3.3.2 CU-1.7.2 - Modificar un usuario.

Desde la barra de herramientas de la vista de gestión de usuarios se podrá invocar la modificación de un usuario seleccionado en la tabla. Esta acción permitirá que se acceda a la vista de modificación del usuario que tendrá una barra de herramientas y se podrá visualizar los siguientes atributos del usuario:

<b>Campo</b>	<b>Significado</b>
<b>Nombre</b>	Nombre del usuario
<b>Apellidos</b>	Apellidos del usuario
<b>Rol</b>	Rol del usuario
<b>Dirección</b>	Dirección del usuario
<b>Población</b>	Población del usuario

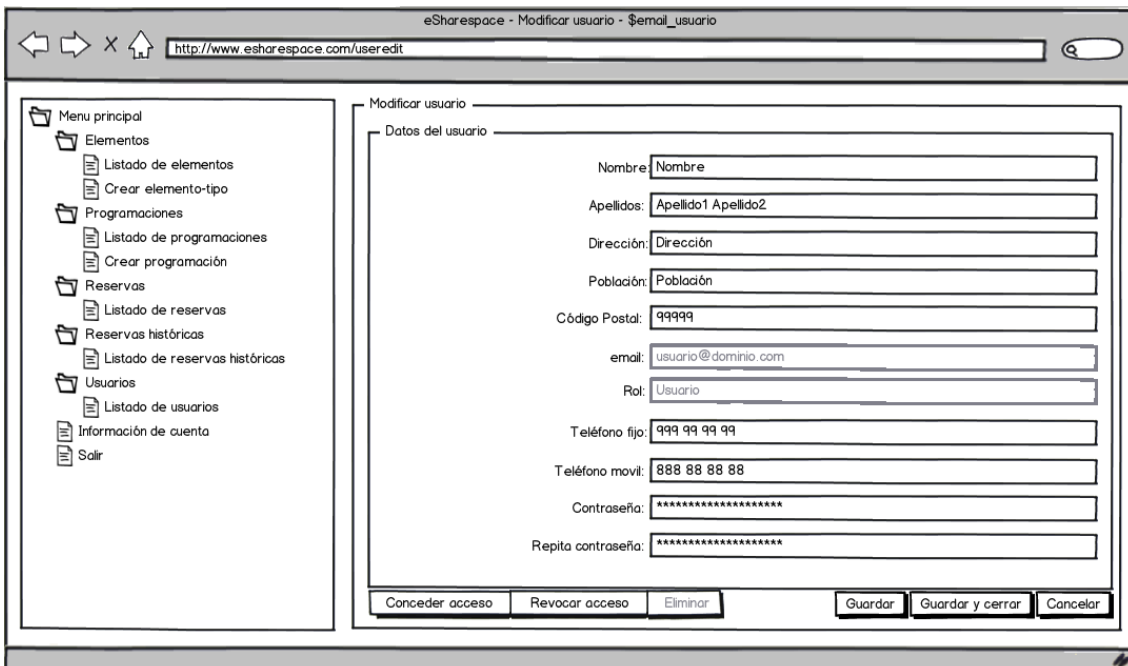


<b>Código postal</b>	Código postal del usuario
<b>Provincia</b>	Provincia del usuario
<b>Email</b>	Dirección de correo electrónico del usuario (servirá de acceso a la aplicación)
<b>Teléfono fijo</b>	Número de teléfono del usuario
<b>Teléfono móvil</b>	Número de teléfono móvil del usuario

Desde la barra de herramientas de esta vista el administrador podrá realizar las siguientes operaciones:

- Concesión de acceso (solo visible si el usuario ha realizado la solicitud de acceso y está pendiente de respuesta).
- Revocación de acceso (solo visible si el usuario ha realizado la solicitud de acceso y está pendiente de respuesta).
- Baja de usuario (solo visible si consta con estado de "Alta" en el sistema).

Este es un prototipo de la pantalla en cuestión:



#### 4.2.3.3.3 CU-1.7.2 - Concesión de acceso a un usuario.

Esta operación se podrá invocar desde dos vistas del IHM:

- Barra de herramientas de la vista de gestión de usuarios.
- Barra de herramientas de la vista de consulta de un usuario.

Serán necesarias las siguientes validaciones:

1. Si nos encontramos en la vista de gestión de usuarios, será necesario que sobre la tabla se encuentre un usuario seleccionado para poder invocar a la operación.
2. El usuario deberá encontrarse en situación de "Solicitud en proceso".

Esta operación, una vez invocada, realizará los siguientes pasos:

1. Requerirá la confirmación por parte del administrador.
2. Cambiará el estado del usuario en el sistema de "Solicitud en proceso" a "Alta".
3. Enviará una notificación al usuario implicado en forma de email o sms si se dispone del número de teléfono móvil.

#### **4.2.3.3.4 CU-1.7.3 - Revocación de acceso a un usuario.**

Esta operación se podrá invocar desde dos vistas del IHM:

- Barra de herramientas de la vista de gestión de usuarios.
- Barra de herramientas de la vista de consulta de un usuario.

Serán necesarias las siguientes validaciones:

1. Si nos encontramos en la vista de gestión de usuarios, será necesario que sobre la tabla se encuentre un usuario seleccionado para poder invocar a la operación.
2. El usuario deberá encontrarse en situación de "Solicitud en proceso".

Esta operación, una vez invocada, realizará los siguientes pasos:

1. Requerirá la confirmación por parte del administrador.
2. Eliminará los datos del usuario del sistema.
3. Enviará una notificación al usuario implicado en forma de email.

#### **4.2.3.3.5 CU-1.7.4 - Baja de un usuario.**

Esta operación se podrá invocar desde dos vistas del IHM:

- Barra de herramientas de la vista de gestión de usuarios.
- Barra de herramientas de la vista de consulta de un usuario.

Serán necesarias las siguientes validaciones:

1. Si nos encontramos en la vista de gestión de usuarios, será necesario que sobre la tabla se encuentre un usuario seleccionado para poder invocar a la operación.
2. El usuario deberá encontrarse en situación de "Alta".

Esta operación, una vez invocada, realizará los siguientes pasos:

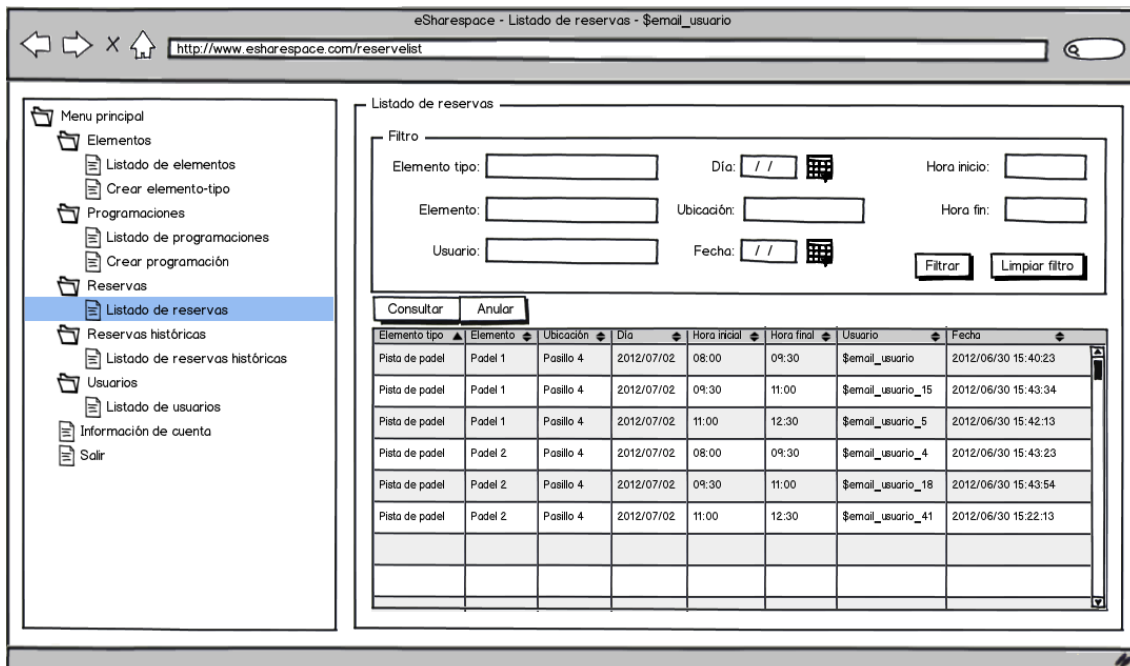
1. Requerirá la confirmación por parte del administrador.
2. Eliminará los datos del usuario del sistema.
3. Enviará una notificación al usuario implicado en forma de email.

#### 4.2.3.4 CU-3.6 - Gestión de reservas pendientes del sistema

Desde el menú principal el administrador podrá acceder a la vista de gestión de reservas pendientes en el sistema. En esta vista se mostrará una barra de herramientas y una tabla que mostrará los siguientes datos de la reserva:

Atributo	Significado
<b>Elemento-tipo</b>	Elemento-tipo sobre el que se hizo la reserva.
<b>Elemento</b>	Elemento reservado.
<b>Ubicación</b>	Ubicación del elemento.
<b>Día</b>	Día de la reserva.
<b>Hora inicial</b>	Hora inicial de la reserva.
<b>Hora final</b>	Hora final de la reserva.
<b>Fecha</b>	Fecha en que se realizó la reserva.
<b>Usuario</b>	Usuario que realizó la reserva.

Este es un prototipo de la pantalla en cuestión:



Desde la barra de herramientas de esta vista el administrador podrá realizar la siguiente operación:

- Anulación de reserva.

#### 4.2.3.4.1 CU-3.3 - Anulación de reserva

Desde la barra de herramientas de la vista de gestión de reservas pendientes, y previa selección de una reserva, se podrá invocar la anulación de la reserva seleccionada. El sistema ofrecerá al administrador la introducción de un motivo para la anulación de la reserva y le pedirá que confirme la acción. Una vez eliminada la reserva el sistema notificará al usuario que la solicitó la anulación, junto con el motivo en caso de ser informado, por email o sms si se dispone del número de teléfono móvil del usuario.

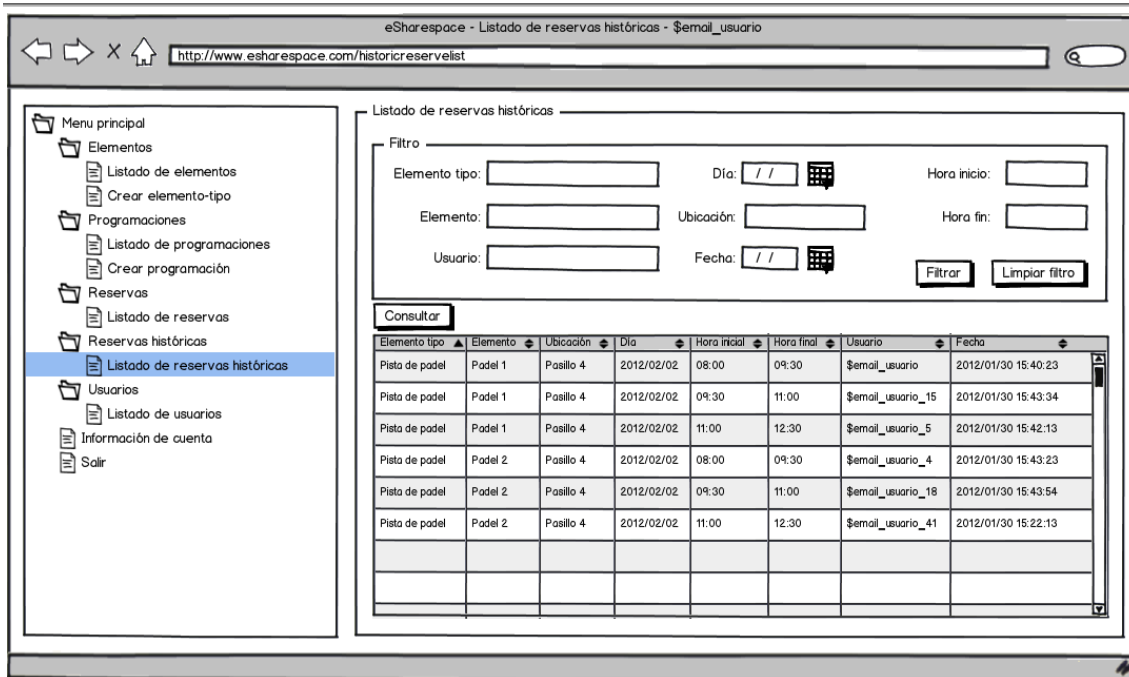


#### 4.2.3.5 CU-4.2 - Gestión de reservas históricas del sistema

Desde el menú principal el administrador podrá acceder a la vista de gestión de reservas históricas del sistema. En esta vista se mostrará una barra de herramientas y una tabla que mostrará los siguientes datos de la reserva:

Atributo	Significado
<b>Elemento-tipo</b>	Elemento-tipo sobre el que se hizo la reserva.
<b>Elemento</b>	Elemento reservado.
<b>Ubicación</b>	Ubicación del elemento.
<b>Día</b>	Día de la reserva.
<b>Hora inicial</b>	Hora inicial de la reserva.
<b>Hora final</b>	Hora final de la reserva.
<b>Fecha</b>	Fecha en que se realizó la reserva.
<b>Usuario</b>	Usuario que realizó la reserva.

Este es un prototipo de la pantalla en cuestión:



Menu principal

- Elementos
  - Listado de elementos
  - Crear elemento-tipo
- Programaciones
  - Listado de programaciones
  - Crear programación
- Reservas
  - Listado de reservas
  - Reservas históricas
  - Listado de reservas históricas
- Usuarios
  - Listado de usuarios
  - Información de cuenta
  - Salir

Listado de reservas históricas

Filtro

Elemento tipo:  Día:  Hora inicio:

Elemento:  Ubicación:  Hora fin:

Usuario:  Fecha:

Elemento tipo	Elemento	Ubicación	Día	Hora inicial	Hora final	Usuario	Fecha
Pista de padel	Padel 1	Pasillo 4	2012/02/02	08:00	09:30	\$email_usuario	2012/01/30 15:40:23
Pista de padel	Padel 1	Pasillo 4	2012/02/02	09:30	11:00	\$email_usuario_15	2012/01/30 15:43:34
Pista de padel	Padel 1	Pasillo 4	2012/02/02	11:00	12:30	\$email_usuario_5	2012/01/30 15:42:13
Pista de padel	Padel 2	Pasillo 4	2012/02/02	08:00	09:30	\$email_usuario_4	2012/01/30 15:43:23
Pista de padel	Padel 2	Pasillo 4	2012/02/02	09:30	11:00	\$email_usuario_18	2012/01/30 15:43:54
Pista de padel	Padel 2	Pasillo 4	2012/02/02	11:00	12:30	\$email_usuario_41	2012/01/30 15:22:13

#### 4.2.3.6 CU-1.5 - Cuenta

Desde el menú principal el administrador podrá acceder a la vista de cuenta que tendrá el mismo formato y operaciones que la descrita para el usuario acreditado en el punto 0.

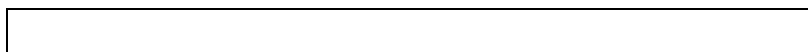
#### 4.2.3.7 CU-1.6 - Salida del sistema

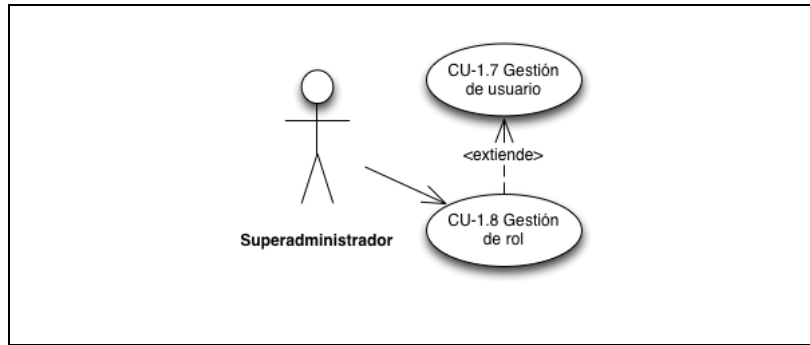
Desde el menú principal el administrador podrá acceder a esta operación. El flujo de esta operación será el mismo que para un usuario acreditado y ya descrito en el punto 4.2.2.5.

#### 4.2.4 Superadministrador

Esta sección describe todos los posibles casos de uso de un usuario con acreditación de acceso al sistema y rol de superadministrador. Este rol se designa por instalación y permitirá a un usuario el alta en el sistema sin necesidad de realizar la solicitud de acceso. El superadministrador una vez acreditado podrá acceder desde la pantalla de bienvenida a la aplicación a la pantalla de acceso desde la que se le mostrarán, a partir del menú principal, las siguientes vistas:

- CU-5.1 - Gestión de programaciones.
- CU-2.3 - Gestión de elementos-tipo del sistema.
- CU-1.7 - Gestión de usuarios.
- CU-1.8 - Modificación del rol de usuario.
- CU-3.6 - Gestión de reservas pendientes del sistema.
- CU-4.2 - Gestión de reservas históricas del sistema.
- CU-1.5 - Cuenta.
- CU-1.6 - Salida del sistema.





#### 4.2.4.1 CU-5.1 - Gestión de programaciones

Desde el menú principal el superadministrador podrá acceder a la vista de gestión de programaciones. Esta vista y las posibles operaciones serán las mismas que las ya descritas para el administrador del sistema en el punto 4.2.3.1.

#### 4.2.4.2 CU-2.3 - Gestión de elementos-tipo del sistema

Desde el menú principal el superadministrador podrá acceder a la vista de gestión de elementos-tipo. Esta vista y las posibles operaciones serán las mismas que las ya descritas para el administrador del sistema en el punto 4.2.3.2.

#### 4.2.4.3 CU-1.7 - Gestión de usuarios

Desde el menú principal el superadministrador podrá acceder a la vista de gestión de usuarios. Esta vista mostrará una barra de herramientas y una tabla sobre la que aparecerán los usuarios definidos en el sistema.

Esta tabla mostrará los siguientes datos de los usuarios del sistema:

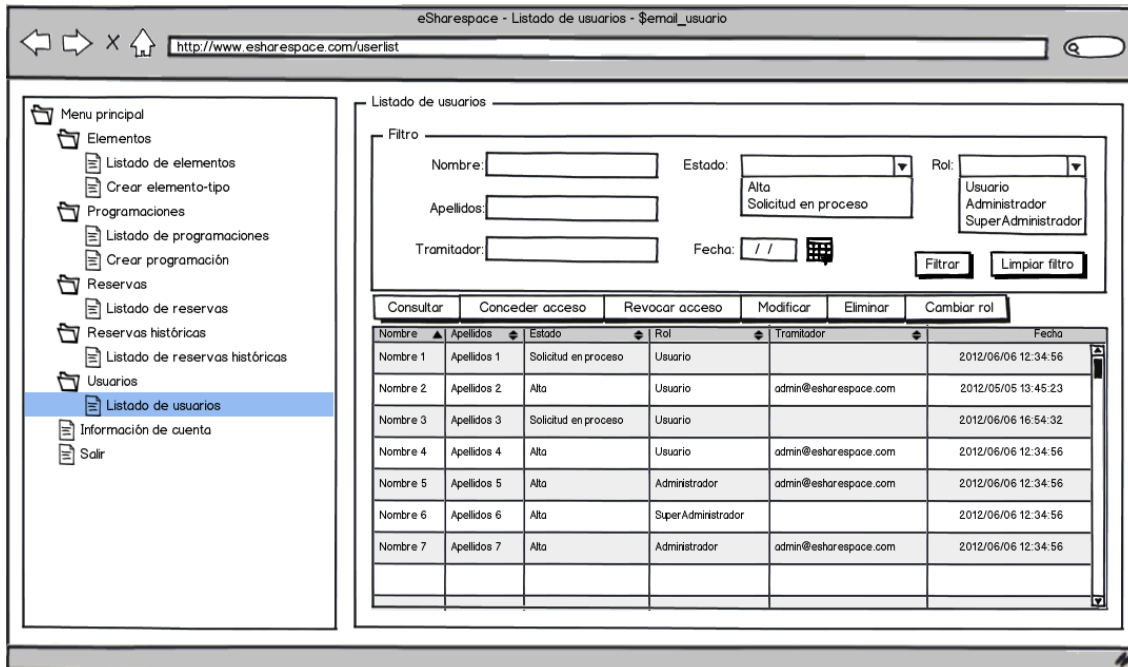
Atributo	Significado
<b>Nombre</b>	Nombre del usuario.
<b>Apellidos</b>	Apellidos del usuario.
<b>Estado</b>	Estado del usuario. El estado será uno de los siguientes: <ul style="list-style-type: none"> <li>• Solicitud en proceso.</li> <li>• Alta.</li> </ul>
<b>Rol</b>	Rol del usuario. Podrá ser uno de los siguientes: <ul style="list-style-type: none"> <li>• Usuario.</li> <li>• Administrador.</li> <li>• Superadministrador.</li> </ul>
<b>Fecha</b>	Fecha de ingreso al sistema.
<b>Tramitador</b>	Usuario que tramitó su solicitud de ingreso.

Desde la barra de herramientas se podrán invocar las siguientes acciones:

- Consulta de un usuario. Descrita en el punto 4.2.3.3.1.
- Concesión de acceso. Descrita en el punto 4.2.3.3.3.
- Revocación de acceso. Descrita en el punto 4.2.3.3.4.

- Baja del usuario. Descrita en el punto 4.2.3.3.5.
- Cambio de rol.

Este es un prototipo de la pantalla en cuestión:



#### 4.2.4.3.1 CU-1.8 - Modificación del rol de usuario

Esta operación se podrá invocar desde dos vistas del IHM para el rol de superadministrador:

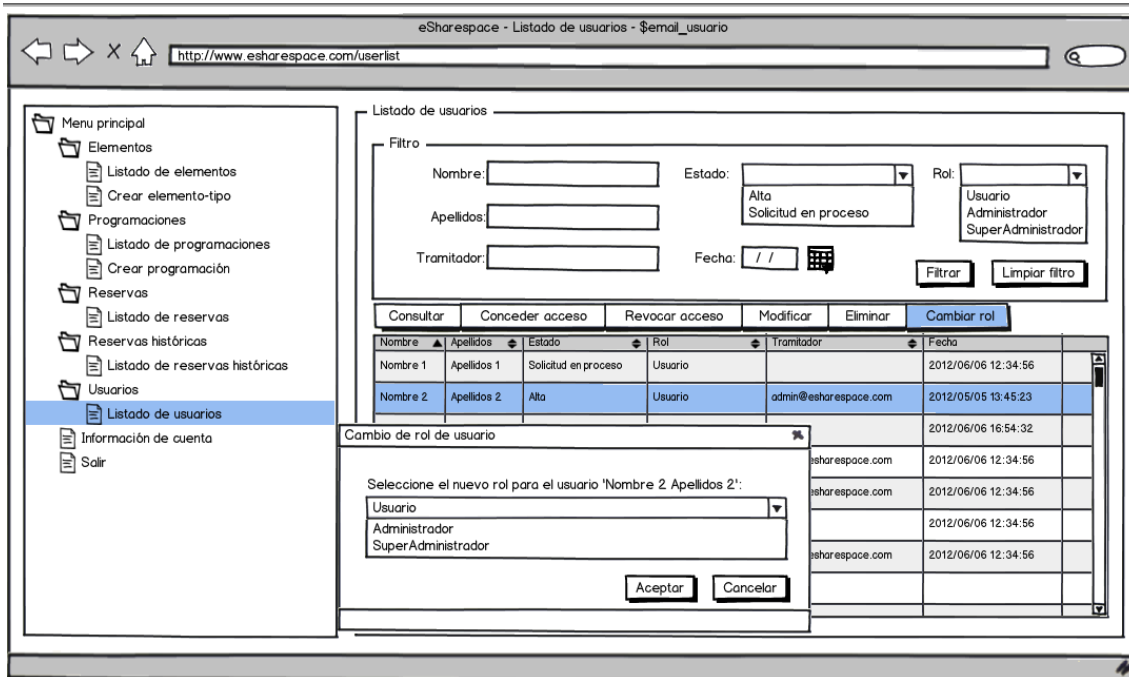
- Barra de herramientas de la vista de gestión de usuarios.
- Barra de herramientas de la vista de consulta de un usuario.

Si nos encontramos en la vista de gestión de usuarios, será necesario que sobre la tabla se encuentre un usuario seleccionado para poder invocar esta operación

Esta operación, una vez invocada, realizará los siguientes pasos:

1. Mostrará un diálogo modal con un combo donde aparecerá seleccionado el rol actual del usuario.
2. En el combo aparecerán los siguientes tipos de roles:
  - a. Usuario.
  - b. Administrador.
  - c. Superadministrador.
3. Si se selecciona un rol diferente al que actualmente tiene el usuario y se acepta el diálogo el sistema cambiará el rol al usuario previa confirmación de la operación.
4. El sistema enviará una notificación al usuario implicado en forma de email o sms si se dispone del número de teléfono móvil indicando su cambio de rol.

En el siguiente prototipo se ve el diálogo que aparecerá al acceder a la operación:



#### 4.2.4.4 CU-3.6 - Gestión de reservas pendientes del sistema

Desde el menú principal el superadministrador podrá acceder a la vista de gestión de reservas pendientes. Esta vista y las posibles operaciones serán las mismas que las ya descritas para el administrador del sistema en el punto 4.2.3.4.

#### 4.2.4.5 CU-4.2 - Gestión de reservas históricas del sistema

Desde el menú principal el superadministrador podrá acceder a la vista de gestión de reservas históricas. Esta vista y las posibles operaciones serán las mismas que las ya descritas para el administrador del sistema en el punto 4.2.3.5.

#### 4.2.4.6 CU-1.5 - Cuenta

Desde el menú principal el superadministrador se podrá acceder a la vista de cuenta que tendrá el mismo formato y operaciones que la descrita para el usuario acreditado en el punto 0.

#### 4.2.4.7 CU-1.6 - Salida del sistema

Desde el menú principal el superadministrador podrá acceder a esta operación. El flujo de esta operación será el mismo que para un usuario acreditado y ya descrito en el punto 4.2.2.5.

### 4.3 Descripción y modelo de datos

Esta sección del documento describe los datos que serán utilizados por el sistema.

#### 4.3.1 Descripción de datos

En esta aplicación se manejarán los siguientes modelos de datos:

##### 4.3.1.1 Usuario

Un usuario será aquella persona que tenga acceso a la aplicación. Distinguimos entre:

- Usuario no acreditado: no tiene credenciales de acceso a la aplicación.



- Usuario acreditado: tiene credenciales de acceso a la aplicación y dentro de estos se encuentran los siguientes roles definidos en la aplicación:
  - Usuario.
  - Administrador.
  - Superadministrador.

Cualquier usuario con acceso a la aplicación podrá solicitar el acceso por medio de un formulario de petición de acceso. Este formulario una vez enviado será valorado por un administrador que determinará la concesión o revocación de acceso al usuario. Una vez concedido el acceso el usuario podrá acreditarse para acceder al menú principal de la aplicación.

#### 4.3.1.2 Rol

Define el tipo de perfil que tiene un usuario acreditado en la aplicación. Como se ha descrito anteriormente, los tipos de rol para los usuarios de la aplicación serán:

- Usuario: usuario final de la aplicación.
- Administrador: gestor de la aplicación.
- Superadministrador: gestor de administradores.

#### 4.3.1.3 Elemento-tipo

Los administradores del sistema podrán gestionar los elementos-tipo definidos. Un elemento-tipo no es más que una plantilla de elemento. Sobre este elemento-tipo se le podrá definir un nombre y unos atributos que servirán para diferenciarlo del resto de plantillas.

#### 4.3.1.4 Atributo

Los atributos son etiquetas de los elementos-tipo que sirven para denominarlos o modelarlos.

#### 4.3.1.5 Elemento

Es la instanciación de un elemento-tipo. A partir de la plantilla (el elemento-tipo) podremos decir cuántos elementos tenemos de ese modelo. Cada elemento tendrá obligatoriamente una descripción que lo hará único (matrícula, ubicación...) y un listado de programaciones de reserva. Como validación, ninguna de las programaciones que tiene asignadas el elemento pueden estar solapadas, esto es que no puede tener dos ranuras de distintas programaciones que coincidan en tiempo.

#### 4.3.1.6 Programación

Determina los tiempos efectivos y duración de la reserva de un elemento. Se compone de un nombre, una vigencia, un calendario, una hora de inicio, una duración y un número de repeticiones. Esto permite que desde una hora de inicio y un número de veces determinado por el número de repeticiones se permita la reserva de un elemento durante la duración establecida.

##### 4.3.1.6.1 Vigencia

Se trata del periodo de tiempo durante el que es vigente una programación. Se determina con una fecha desde y una fecha hasta. Toda vigencia tendrá definida

una fecha desde pero no es obligatorio que defina una fecha hasta. Si la vigencia no tiene definida una fecha hasta se entenderá que es hasta siempre.

#### 4.3.1.6.2 Calendario

Define la periodicidad semanal (Lunes, Martes, Miércoles, Jueves, Viernes, Sábado y Domingo), los días de si aplicación y los días de no aplicación de una programación dentro de una vigencia. No podrá darse el caso que los días de sí o no aplicación estén definidos fuera de la vigencia de la programación.

#### 4.3.1.7 Reserva

Se trata de la asignación a un usuario de una ranura en la programación de un elemento. No se permitirá que una misma ranura de un mismo elemento esté asignada a dos usuarios distintos.

### 4.3.2 Modelo de datos

Este será el modelo de datos para el sistema descrito:

#### 4.3.2.1 Usuario

Campo	Tipo	Significado	Obligatorio
<b>Nombre</b>	Texto	Nombre del usuario	SI
<b>Apellidos</b>	Texto	Apellidos del usuario	SI
<b>Rol</b>	Rol	Rol del usuario	SI
<b>Dirección</b>	Texto	Dirección del usuario	SI
<b>Población</b>	Texto	Población del usuario	SI
<b>Código postal</b>	Numérico	Código postal del usuario	SI
<b>Provincia</b>	Texto	Provincia del usuario	SI
<b>Email</b>	Texto	Dirección de correo electrónico del usuario (servirá de acceso a la aplicación)	SI
<b>Teléfono fijo</b>	Numérico	Número de teléfono del usuario	NO
<b>Teléfono móvil</b>	Numérico	Número de teléfono móvil del usuario	NO
<b>Contraseña</b>	Texto	Contraseña del usuario	SI
<b>Repetición de la contraseña</b>	Texto	Validación de la contraseña del usuario	SI
<b>Contraseña generada</b>	Texto	Contraseña aleatoria generada por el sistema. En caso de no haberse solicitado la contraseña será nula.	NO
<b>Tramitador</b>	Usuario	Usuario que tramitó el alta de este usuario.	SI

#### 4.3.2.2 Rol

Campo	Tipo	Significado	Obligatorio
Rol	Texto	Rol del usuario	SI

#### 4.3.2.3 Elemento-tipo

Campo	Tipo	Significado	Obligatorio
Nombre	Texto	Nombre del elemento-tipo	SI
Atributos	Lista	Lista de atributos del elemento-tipo.	SI

#### 4.3.2.4 Atributo

Campo	Tipo	Significado	Obligatorio
Nombre	Texto	Nombre del atributo	SI

#### 4.3.2.5 Elemento

Campo	Tipo	Significado	Obligatorio
Descripción	Texto	Descripción del elemento.	SI
Programaciones	Lista	Lista de programaciones del elemento.	SI

#### 4.3.2.6 Programación

Campo	Tipo	Significado	Obligatorio
Nombre	Texto	Nombre de la programación	SI
Vigencia	Vigencia	Vigencia de la programación.	SI
Calendario	Calendario	Texto en formato L,M,X,J,V,S,D;DIAS_SI;DIAS_NO.	SI
Hora inicio	Tiempo	Hora de inicio de la programación	SI
Duración	Numérico	Duración en minutos de una ranura de la programación.	SI
Repeticiones	Numérico	Número de ranuras de la programación.	SI

#### 4.3.2.7 Vigencia

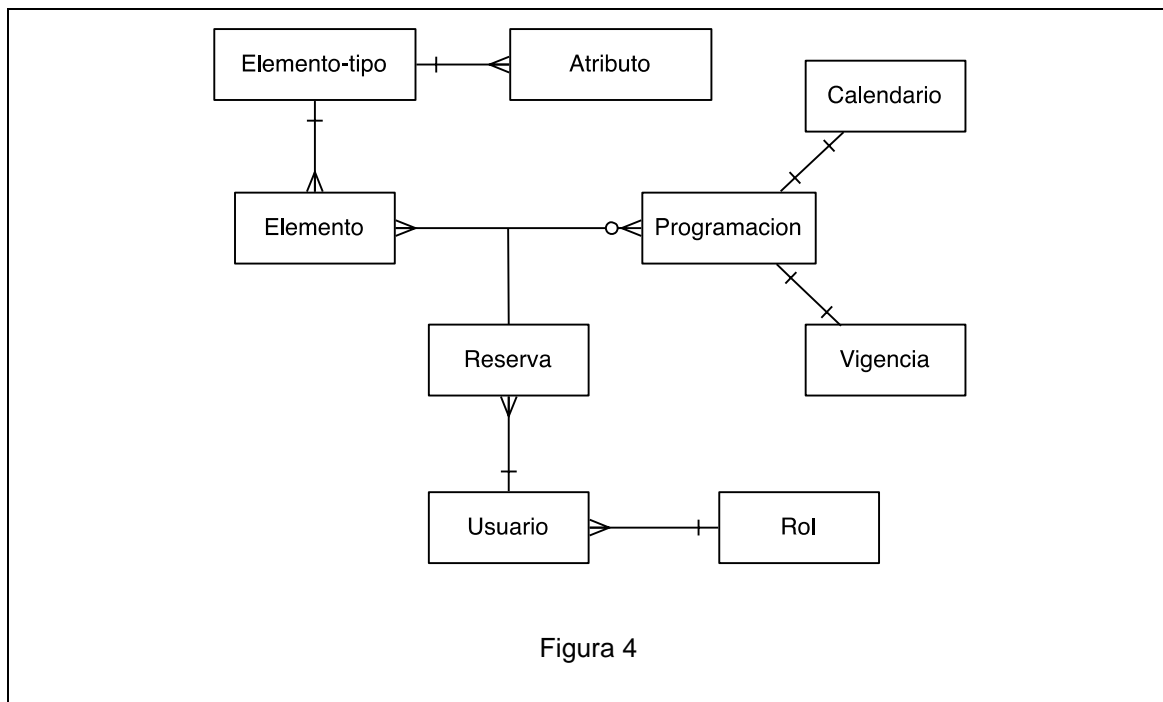
Campo	Tipo	Significado	Obligatorio
Desde	Día	Día de inicio de la vigencia (incluido)	SI
Hasta	Día	Día final de la vigencia (incluido)	NO

#### 4.3.2.8 Reserva

Campo	Tipo	Significado	Obligatorio
<b>Elemento</b>	Elemento	Elemento a reservar.	SI
<b>Día</b>	Día	Día de la reserva.	SI
<b>Hora inicio</b>	Tiempo	Hora de inicio de la reserva	SI
<b>Usuario</b>	Usuario	Usuario que tiene la reserva asignada	SI

#### 4.3.3 Relaciones

Estas son las relaciones entre el modelo de datos mencionado anteriormente:



## 5 Diseño detallado

### 5.1 Introducción

En este documento se especifica el diseño técnico del proyecto “eSharespace” en continuación al documento anterior de especificación y análisis de requisitos y continuando con el ciclo de vida clásico de desarrollo del software.

La arquitectura que se propone encapsula la gestión de entidades en subsistemas de tal manera que se puedan agrupar los casos de uso detectados en el documento de especificación en subsistemas concretos.

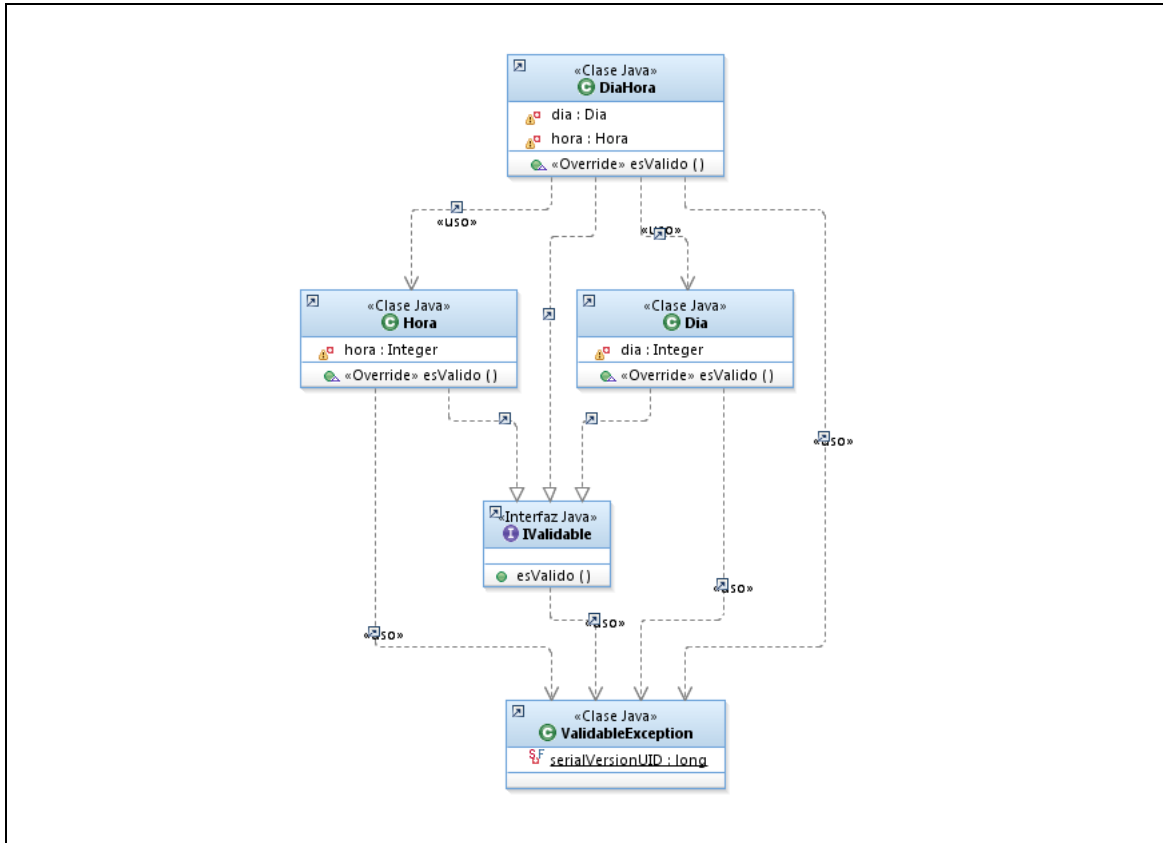
Cada subsistema tendrá los siguientes componentes:

- Una interface donde se declaren las operaciones que expone a los clientes.
- Un modelo de datos de intercambio con el que trabajar con los clientes.

- Un modelo de datos interno con el que realizar la lógica de negocio.
- Una implementación de la interface donde:
  - Realizará la conversión entre el modelo interno y de intercambio.
  - Realizará las comprobaciones de permisos requeridos por las operaciones.
- Un manager de proceso que:
  - Contendrá la lógica de negocio del subsistema.
  - Validará las entidades.
- Un manager de acceso a datos (DAO) que gestionará la persistencia de la entidad.

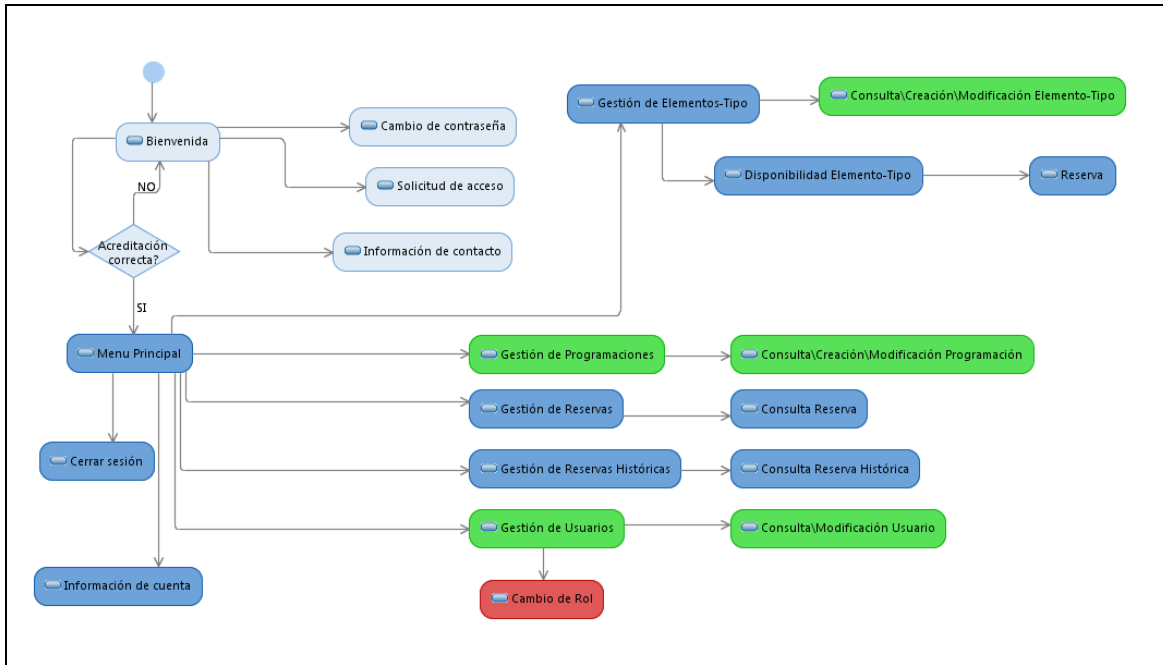
## 5.2 Modelo de datos común

Hay clases e interfaces que son utilizadas tanto por varios subsistemas como por los clientes. Las siguientes clases se definirán como un modelo de datos común con el fin que se puedan reutilizar entre los diversos subsistemas y clientes.



### 5.3 Navegabilidad entre pantallas

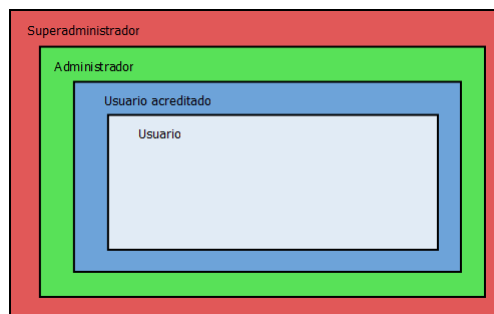
La navegación entre pantallas para poder acceder a cada funcionalidad requerida será la siguiente:



El sistema gestionará el acceso a 4 tipos de usuario (usuarios, usuarios acreditados, administradores y superadministrador) y las pantallas a las que tendrán acceso se encuentran en grupos de colores quedando como sigue:

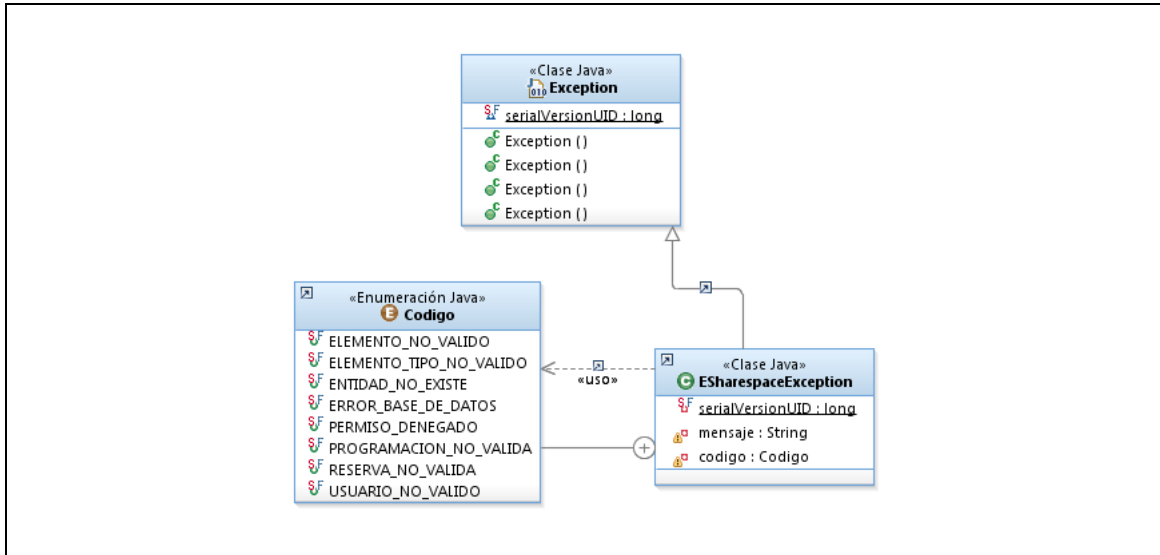
- Usuario: Pantallas en color azul claro
- Usuario acreditado: Pantallas en color azul oscuro.
- Administrador: Pantallas en color verde.
- Superadministrador: Pantallas en color rojo.

Los roles de usuario están jerarquizados de tal manera que el rol superior tiene los mismos permisos o acceso a pantallas que el rol inferior más los específicos del suyo.



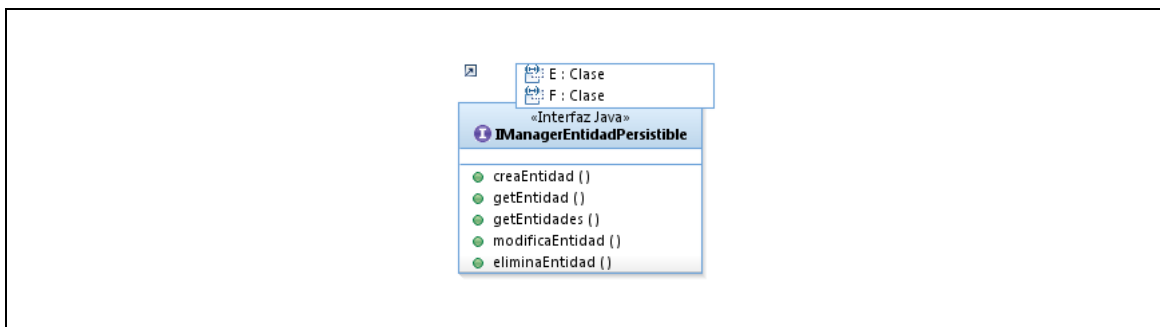
### 5.4 Excepciones

Todas las excepciones de la aplicación heredarán de una excepción general (ESharespaceException) que albergará un código de error y un mensaje descriptivo de lo ocurrido.



### 5.5 Modelo de gestión de una entidad

En todos los subsistemas de la aplicación se realizan operaciones y lógica de gestión de una o varias entidades. Todas las entidades de la aplicación tienen el mismo ciclo de vida (creación, consulta, modificación y eliminación) por lo que para unificar la gestión de todas las entidades se definirá una interface con las operaciones necesarias para gestionar el ciclo de vida de una entidad. De esta manera, todas las implementaciones de interface, los gestores de proceso y de base de datos deberán implementar esta interface.



Esta interface expone las operaciones necesarias para completar todo el ciclo de vida de una entidad. Las operaciones son:

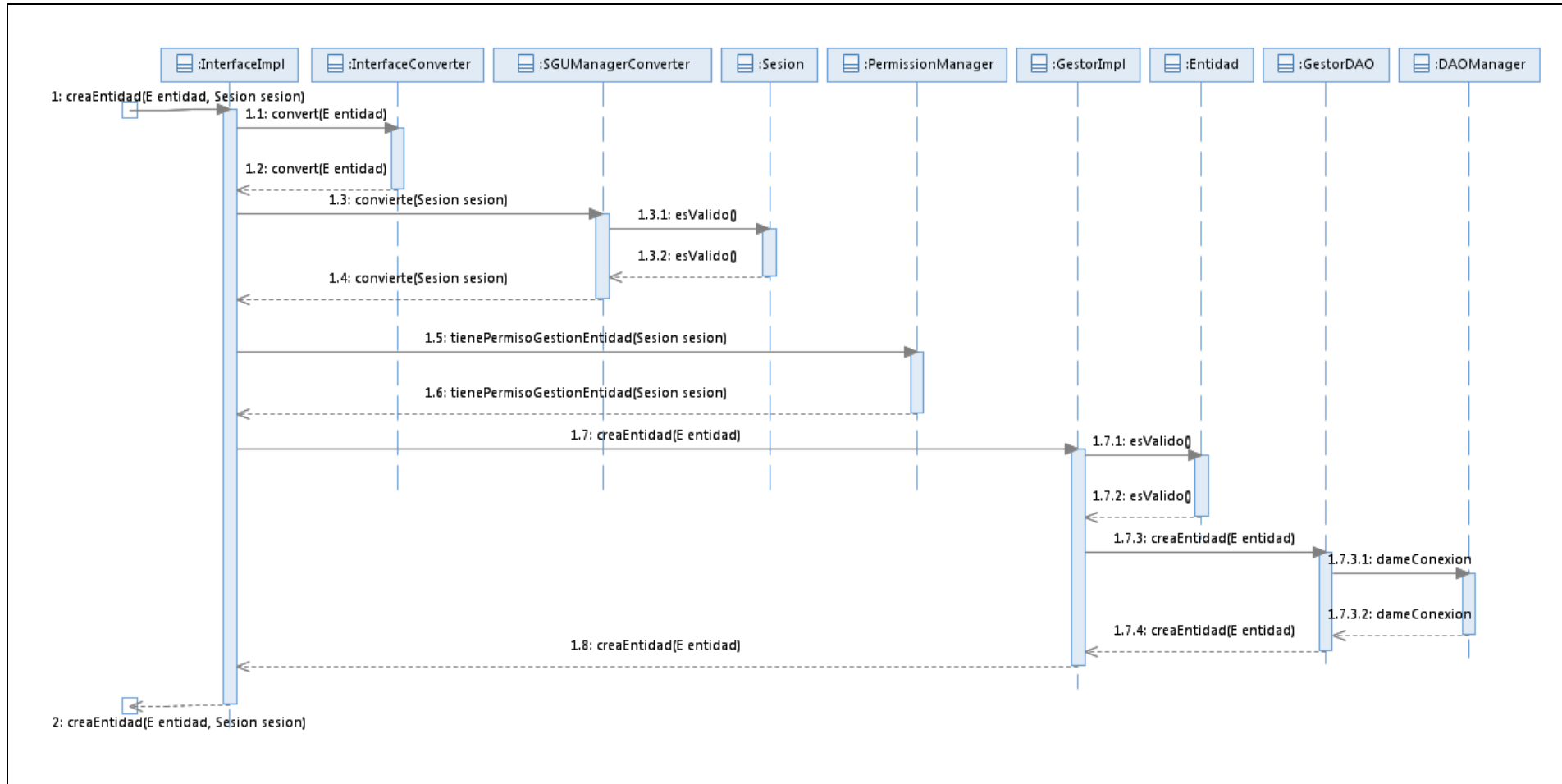
- creaEntidad(E entidad): creará la entidad en el sistema y la asignará un identificador único.
- getEntidad(Object id): busca la entidad con el identificador definido.



- `getEntidades(F filtro)`: busca las entidades que cumplan con el filtro definido.
- `modificaEntidad(E entidad)`: modifica la entidad.
- `eliminaEntidad(Object id)`: elimina la entidad con el identificador definido.

### 5.5.1 Creación de una entidad

Este será el diagrama de secuencia para la creación de una entidad:

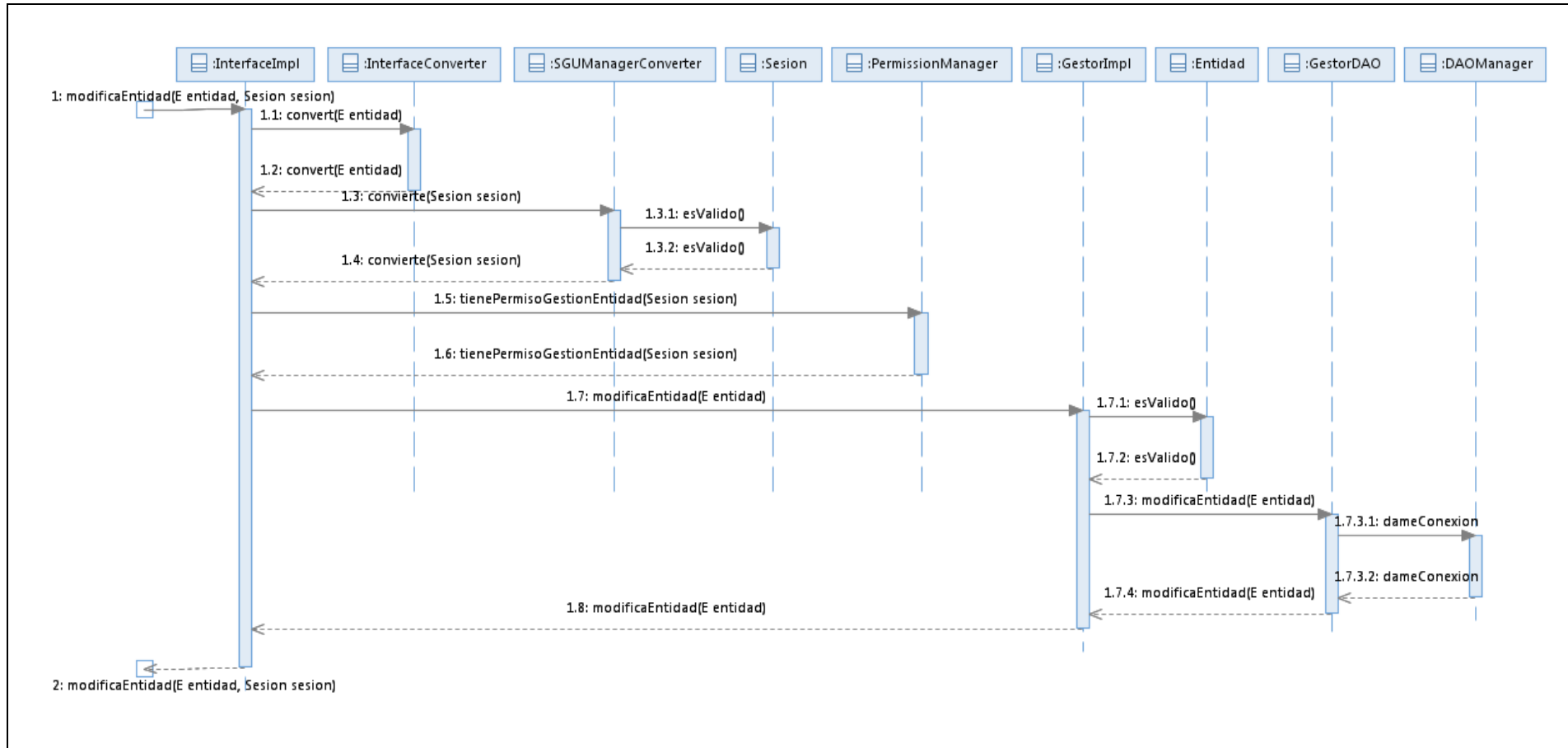


La operativa por tanto será la siguiente:

1. Para invocar la operación será necesario pasar como parámetro la entidad a guardar y la sesión del usuario.
2. En la implementación de la interface se procederá a validar la sesión y se comprobará que la sesión dispone de los permisos suficientes para invocar la operación.
3. Después, la implementación de la interface procederá a invocar la llamada del gestor de proceso.
4. El gestor de proceso validará la entidad y procederá a persistir la entidad mediante el gestor de base de datos.

### 5.5.2 Modificación de una entidad

Este será el diagrama de secuencia para la modificación de una entidad:

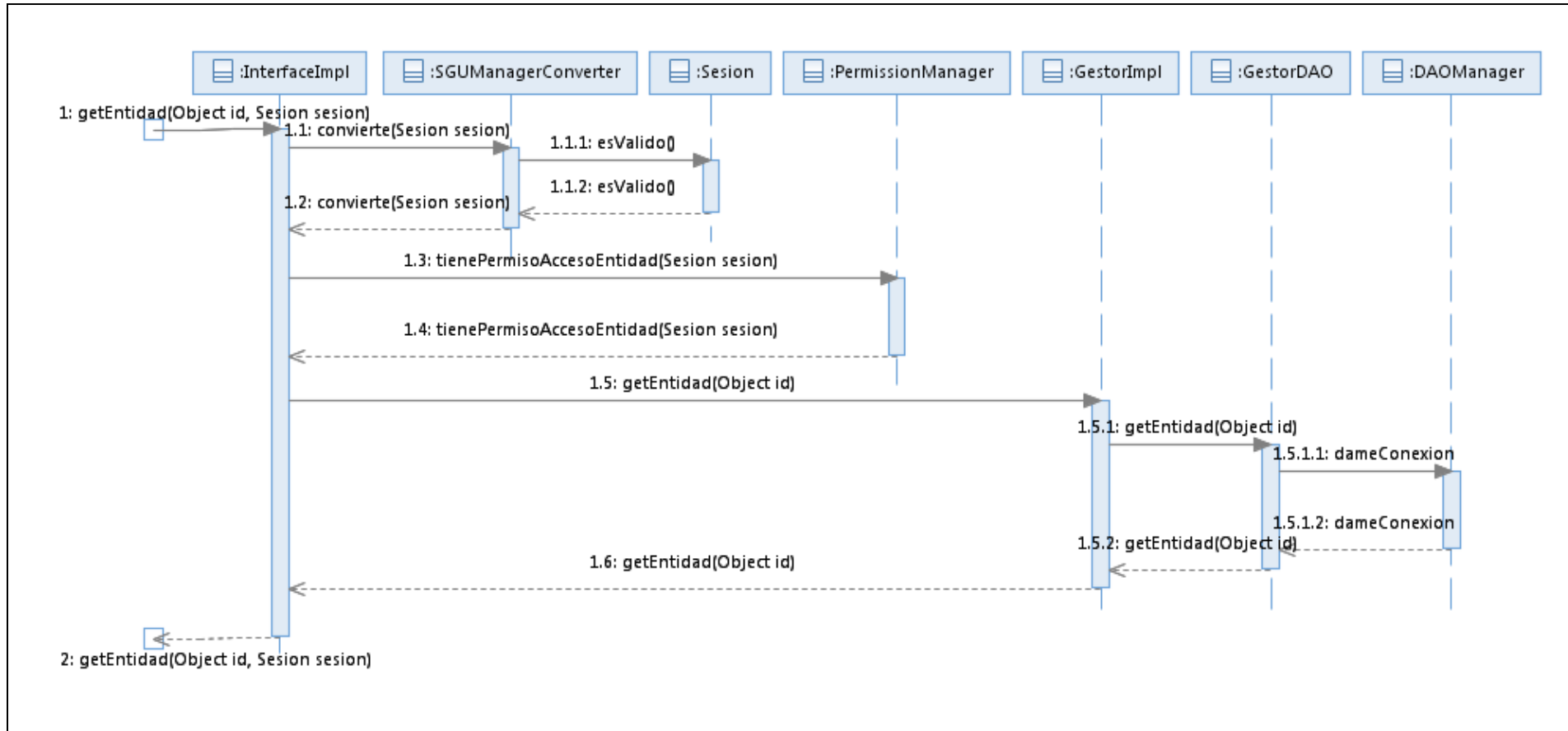


La operativa por tanto será la siguiente:

1. Para invocar la operación será necesario pasar como parámetro la entidad a guardar y la sesión del usuario.
2. En la implementación de la interface se procederá a validar la sesión y se comprobará que la sesión dispone de los permisos suficientes para invocar la operación.
3. Después, la implementación de la interface procederá a invocar la llamada del gestor de proceso.
4. El gestor de proceso validará la entidad y procederá a persistir la entidad mediante el gestor de base de datos.

### 5.5.3 Búsqueda de una entidad

Este será el diagrama de secuencia para la búsqueda de una entidad:

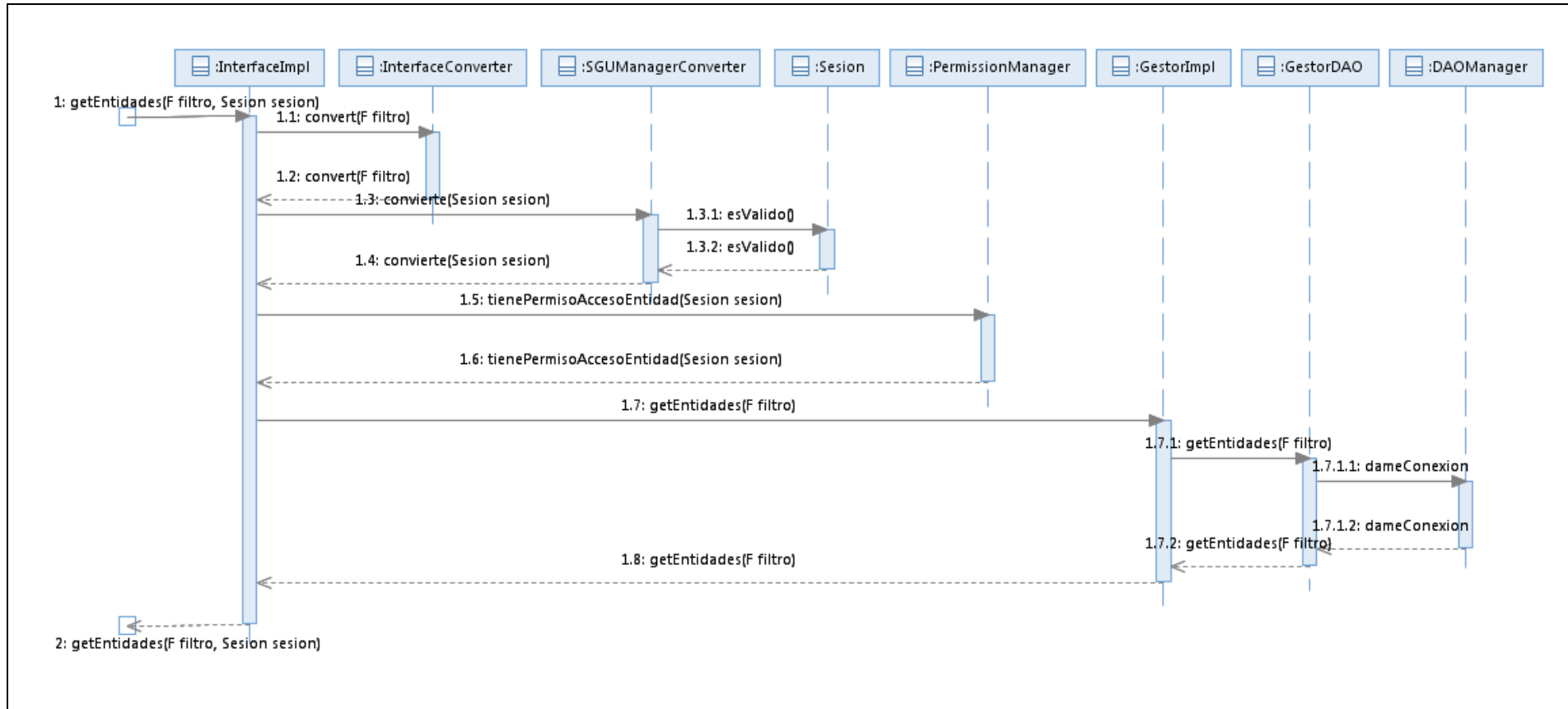


La operativa por tanto será la siguiente:

1. Para invocar la operación será necesario pasar como parámetro el identificador de la entidad a buscar y la sesión del usuario.
2. En la implementación de la interface se procederá a validar la sesión y se comprobará que la sesión dispone de los permisos suficientes para invocar la operación.
3. Después, la implementación de la interface procederá a invocar la llamada del gestor de proceso.
4. El gestor de proceso pedirá al gestor de base de datos la entidad con el identificador informado.

### 5.5.4 Búsqueda de entidades

Este será el diagrama de secuencia para la búsqueda del conjunto de entidades que cumpla un filtro:



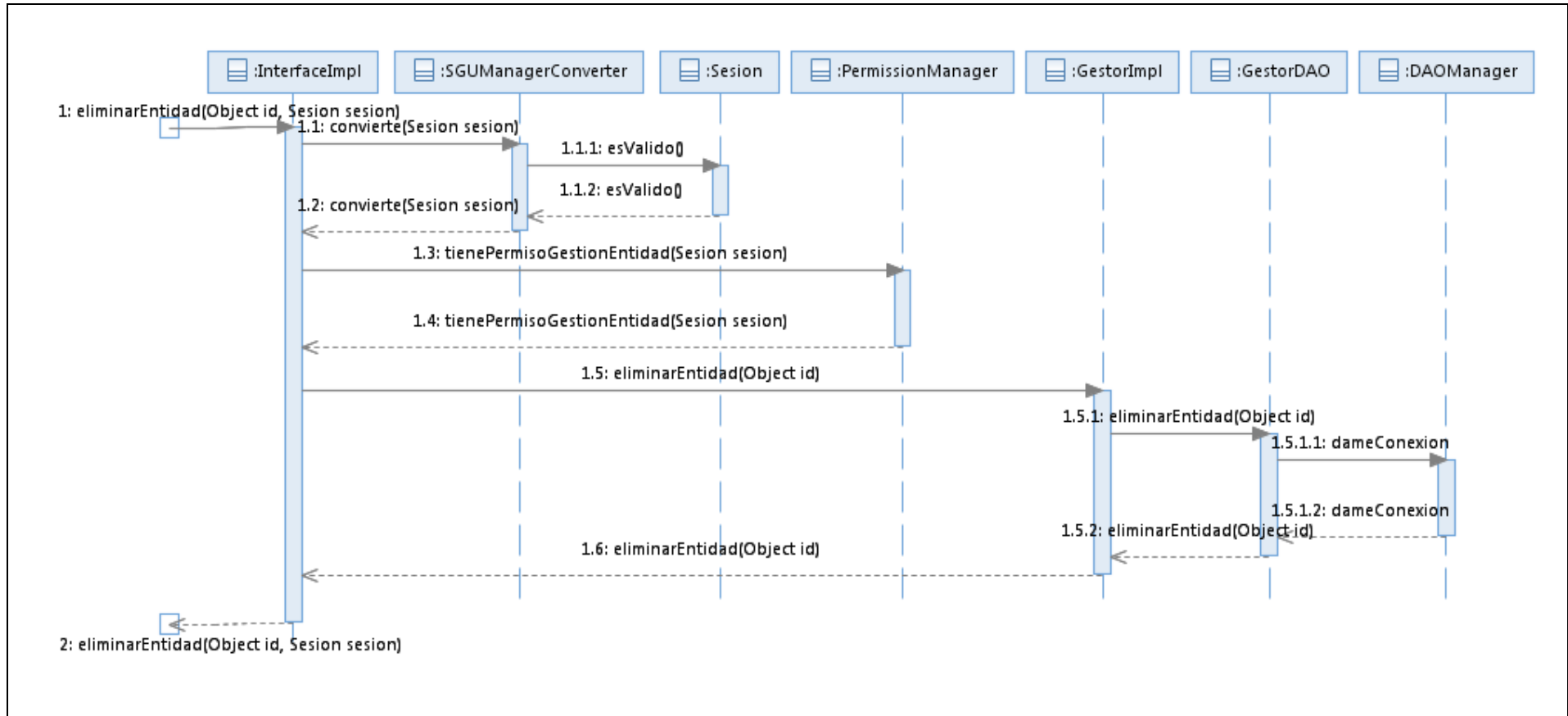
La operativa por tanto será la siguiente:



1. Para invocar la operación será necesario pasar como parámetro el filtro que deben pasar las entidades que se quieren buscar y la sesión del usuario.
2. En la implementación de la interface se procederá a validar la sesión y se comprobará que la sesión dispone de los permisos suficientes para invocar la operación.
3. Después, la implementación de la interface procederá a invocar la llamada del gestor de proceso.
4. El gestor de proceso pedirá al gestor de base de datos las entidades que cumplan el filtro.

### 5.5.5 Eliminación de una entidad

Este será el diagrama de secuencia para la eliminación de una entidad del sistema:



La operativa por tanto será la siguiente:

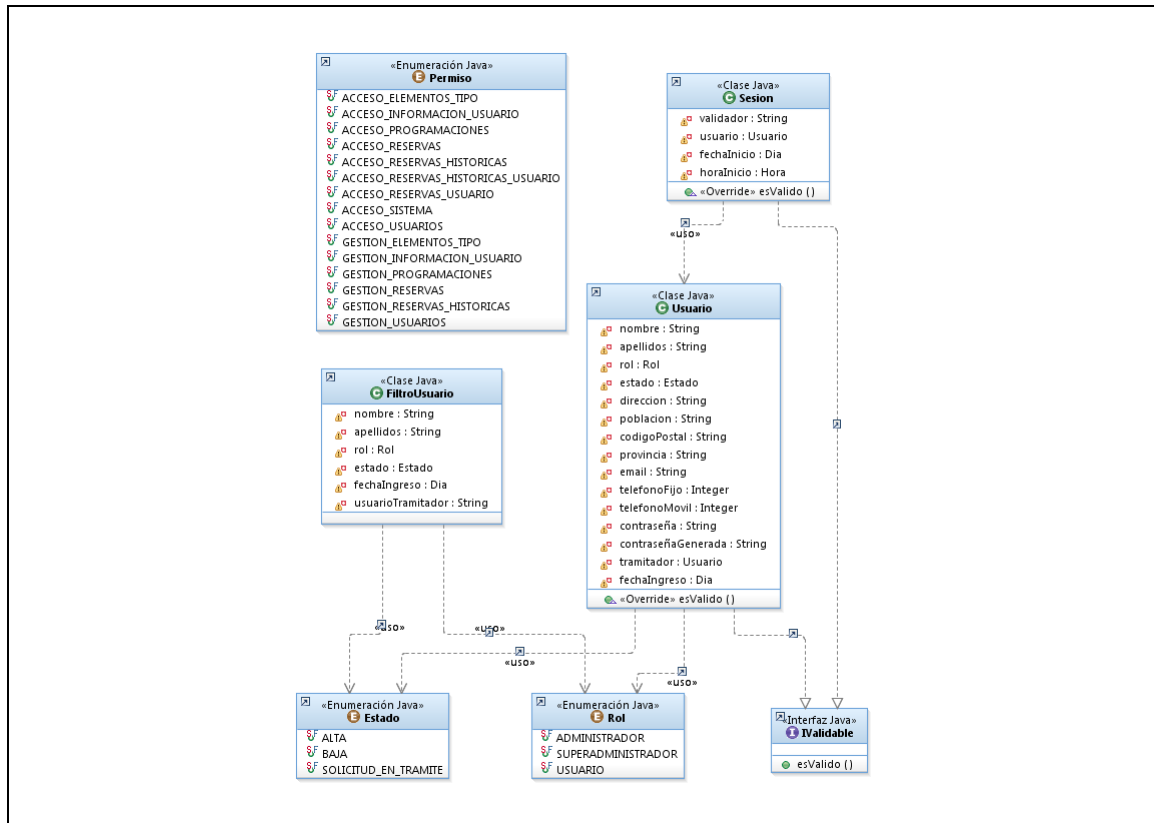
1. Para invocar la operación será necesario pasar como parámetro el identificador de la entidad a eliminar y la sesión del usuario.
2. En la implementación de la interface se procederá a validar la sesión y se comprobará que la sesión dispone de los permisos suficientes para invocar la operación.
3. Después, la implementación de la interface procederá a invocar la llamada del gestor de proceso.
4. El gestor de proceso pedirá al gestor de base de datos que elimine la entidad con el identificador definido.

## 5.6 Subsistema de gestión de usuarios (SGU)

Este subsistema gestionará la creación, modificación y eliminación de usuarios del sistema. También proporcionará operaciones para la comprobación de permisos en operaciones y para proporcionar información acerca de los usuarios.

### 5.6.1 Modelo de datos

El modelo de datos necesario para manejar la información relativa a este subsistema y a todo el conjunto de sus operaciones es el siguiente:

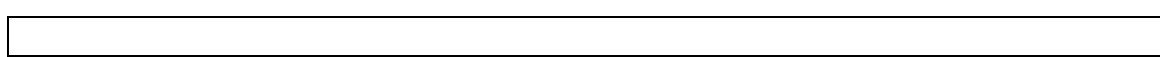


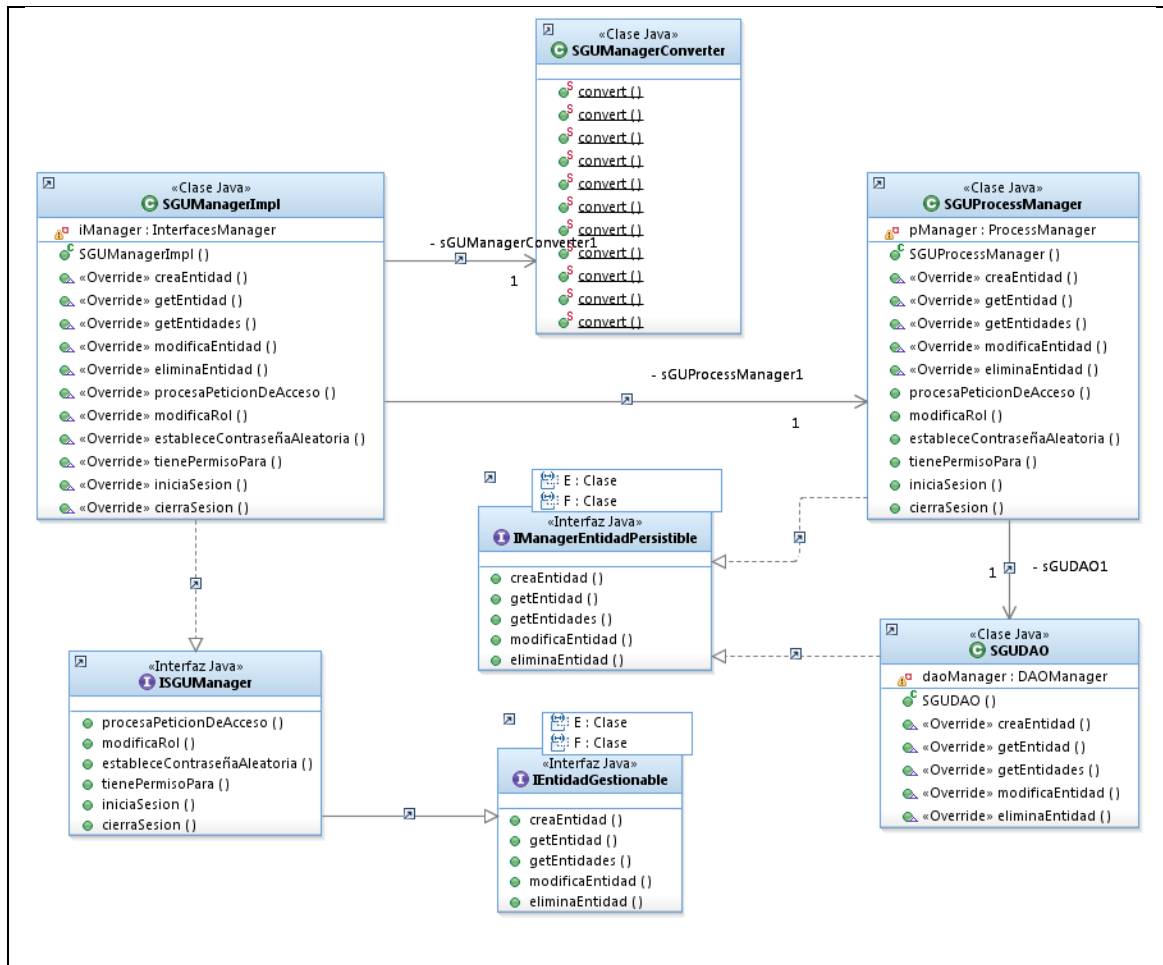
#### 5.6.1.1 Negocio

En este punto se detalla todo lo correspondiente a las operaciones y lógica de negocio del SGU.

##### 5.6.1.1.1 Clases gestoras

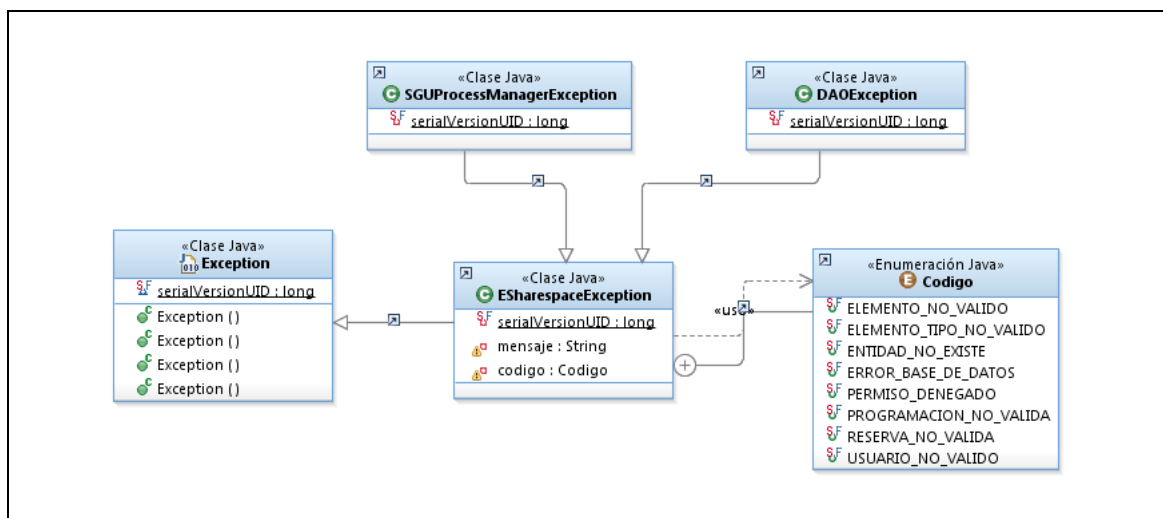
Estas serán las clases gestoras que formarán el subsistema del SGU:





### 5.6.1.1.2 Excepciones

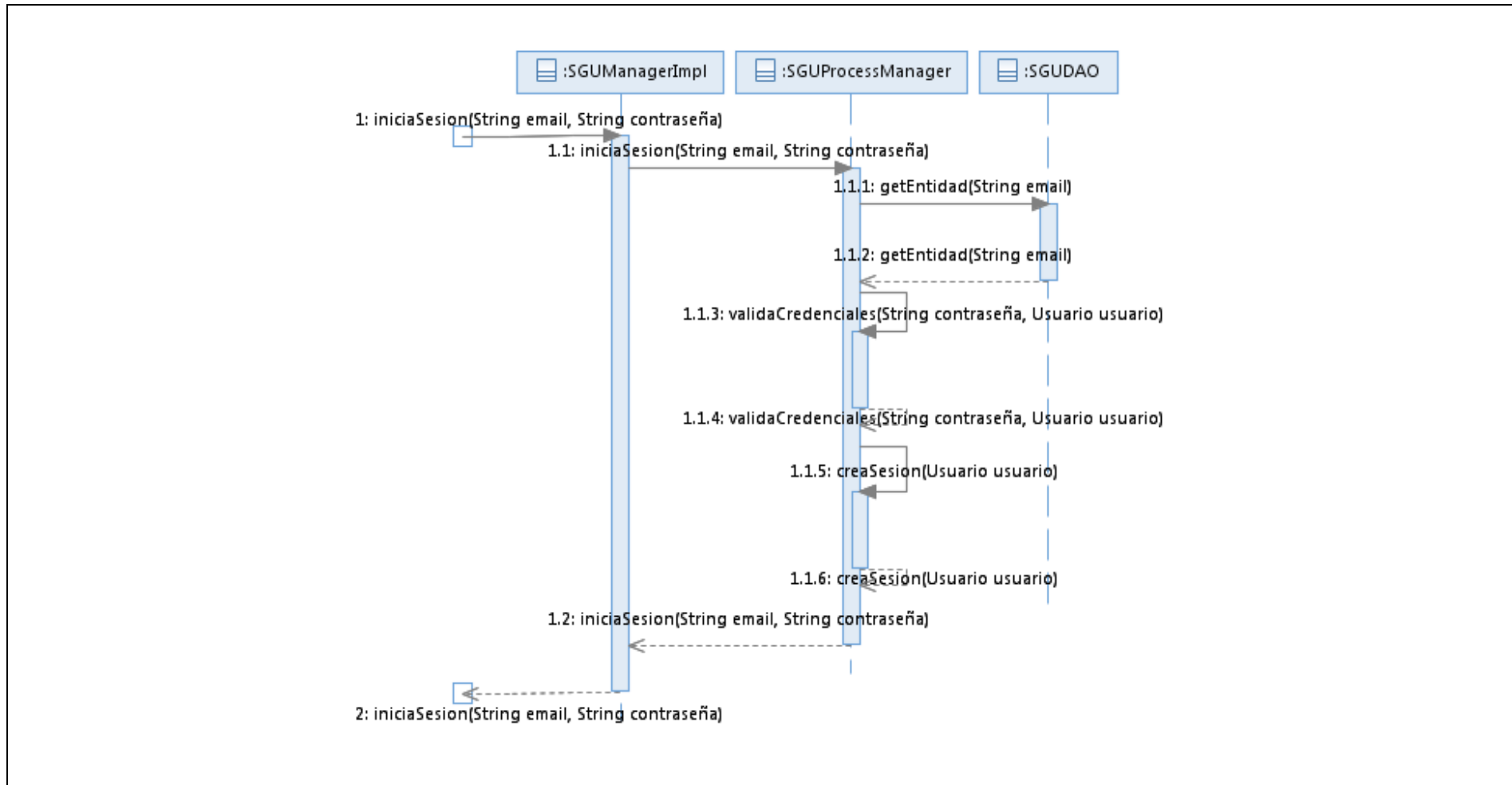
Estas serán las excepciones que manejará este subsistema:



### 5.6.2 Diagramas de secuencia

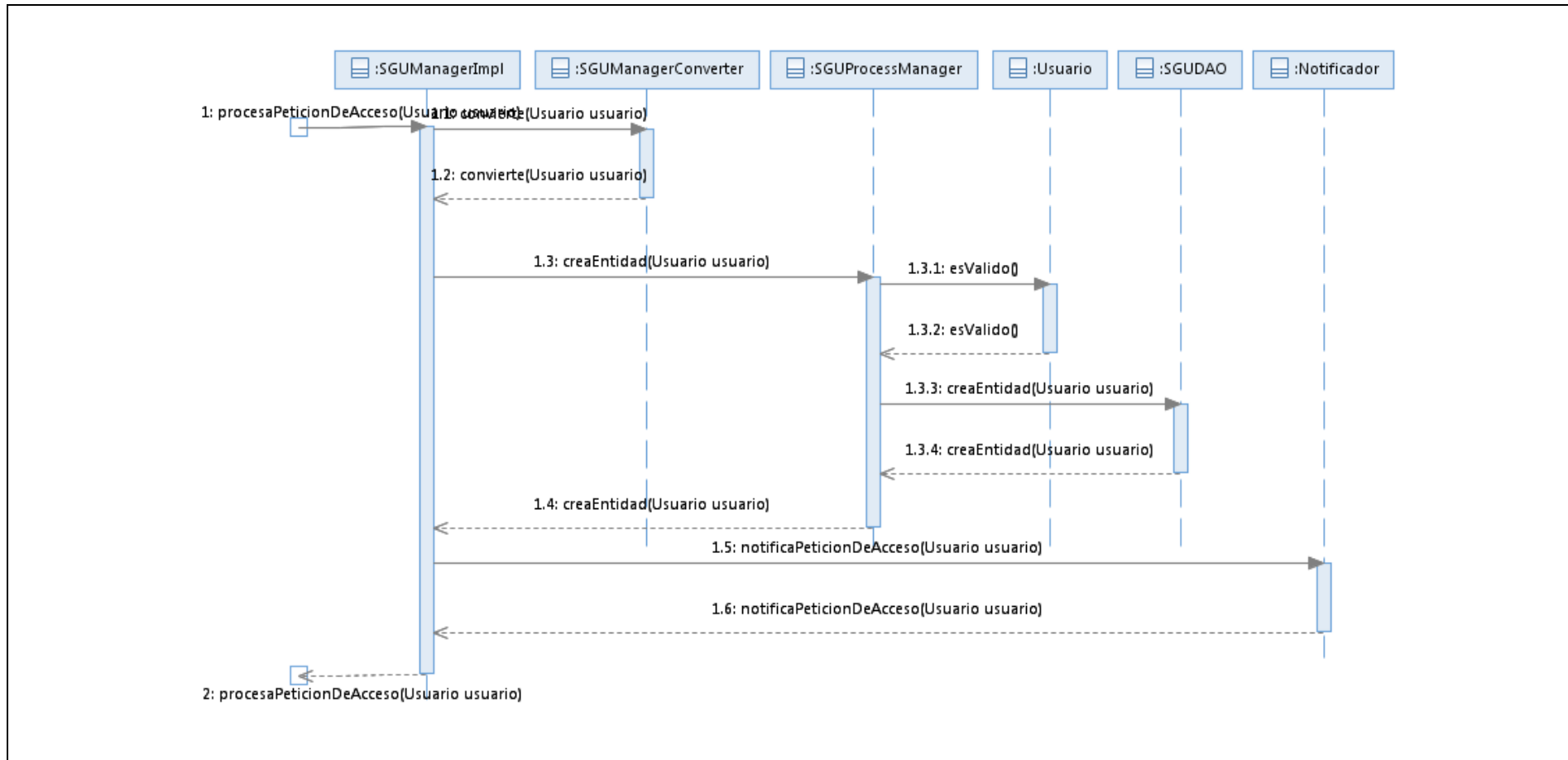
A continuación se listan los diagramas de secuencia en los que intervienen procesos de servidor.

#### 5.6.2.1 Caso de uso – CU-1.1 – Iniciar sesión

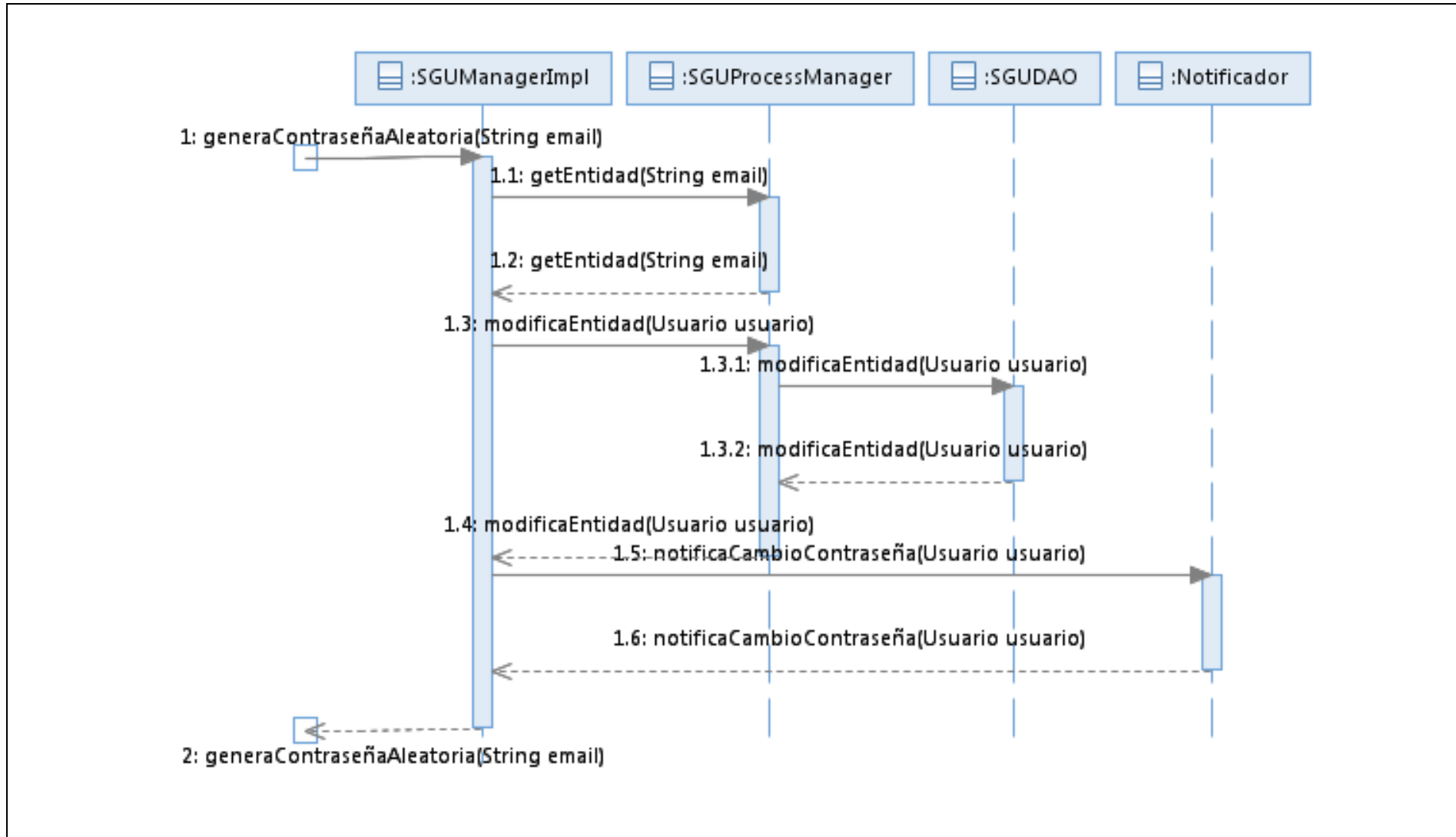


### 5.6.2.2 Caso de uso – CU-1.2 - Procesar petición de acceso

Este caso de uso se modelará como la creación de un usuario utilizando así la operación de gestión básica de una entidad ([Creación de una entidad](#)). Este usuario tendrá Estado “SOLICITUD\_EN\_TRAMITE” mientras que un Administrador no valide los datos.

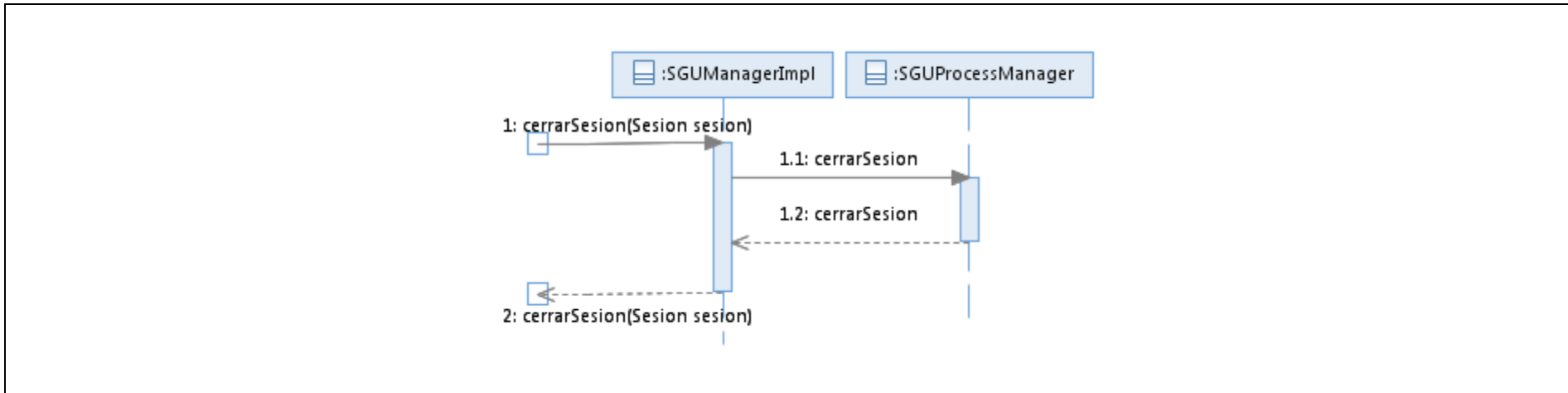


5.6.2.3 Caso de uso – CU-1.4 – Establecer contraseña aleatoria





#### 5.6.2.4 Caso de uso – CU-1.6 – Cerrar sesión



#### 5.6.2.5 Caso de uso – CU-1.7 – Listar usuarios

Este caso de uso está comprendido en la gestión básica de una entidad definida anteriormente ([Búsqueda de entidades](#)).

#### 5.6.2.6 Caso de uso – CU-1.7.1 – Información de usuario

Este caso de uso está comprendido en la gestión básica de una entidad definida anteriormente ([Búsqueda de entidad](#)).

#### 5.6.2.7 Caso de uso – CU-1.7.2 – Modificación de usuario

Este caso de uso está comprendido en la gestión básica de una entidad definida anteriormente ([Modificación de entidad](#)).

#### 5.6.2.8 Caso de uso – CU-1.7.3 – Alta de usuario

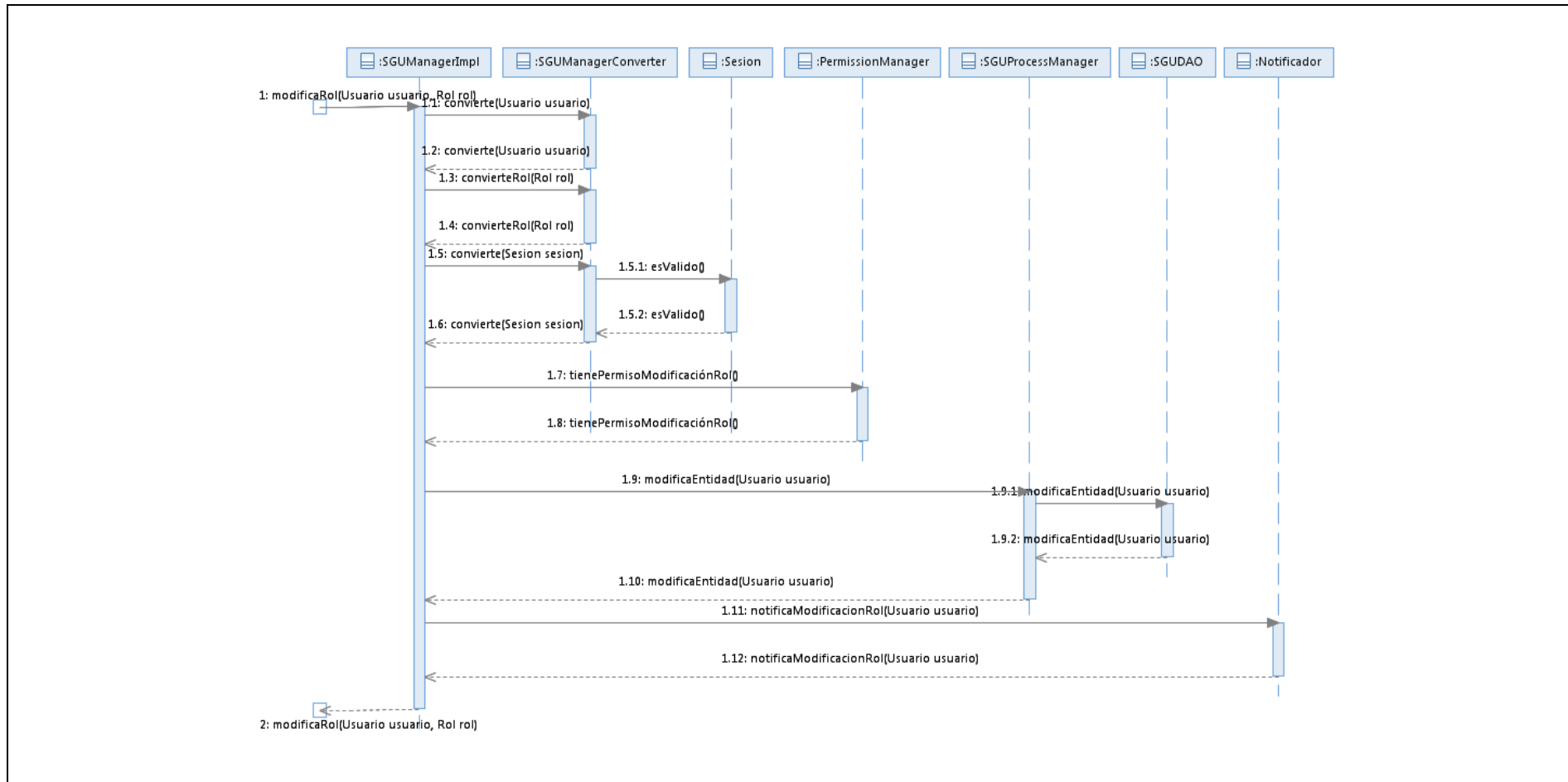
Este caso de uso está comprendido en la gestión básica de una entidad definida anteriormente ([Modificación de entidad](#)) ya que el alta consiste en un cambio de estado del usuario.

#### 5.6.2.9 Caso de uso – CU-1.7.4 – Baja de usuario

Este caso de uso está comprendido en la gestión básica de una entidad definida anteriormente ([Eliminación de una entidad](#)).

### 5.6.2.10 Caso de uso – CU-1.8 – Modificar Rol

Este caso de uso está comprendido en la gestión básica de una entidad definida anteriormente ([Modificación de entidad](#)). La única diferencia con respecto a la secuencia del modelo de modificación de una entidad es que se especificará el Rol junto con el Usuario a modificar.



### 5.6.3 Fichas CRC

#### 5.6.3.1 Usuario

Clase	Usuario
Descripción	Representa la información existente en el sistema de un usuario
Tipo	Principal
Características	Concreta y persistente
Responsabilidades	Proporcionar los datos de un usuario concreto
Colaboraciones	Rol
<b>Constructores:</b>	
+Usuario()	
<b>Atributos:</b>	
-nombre : String	Nombre del usuario
-apellidos : String	Apellidos del usuario
-rol : Rol	Rol del usuario
-estado : Estado	Estado del usuario
-dirección : String	Dirección del usuario
-población : String	Población del usuario
-codigoPostal : String	Código Postal del usuario
-provincia : String	Provincia del usuario
-email : String	Email del usuario, este dato será clave primaria para esta entidad.
-telefonoFijo : String	Teléfono fijo del usuario
-telefonoMovil : String	Teléfono móvil del usuario
-contraseña : String	Contraseña del usuario
-contraseñaGenerada : String	Contraseña generada del usuario
-tramitador : Usuario	Usuario que ha tramitado el alta del usuario
-fechaIngreso : DiaHora	Día y hora en que hizo el ingreso en la aplicación
<b>Métodos:</b>	La clase es un bean por lo que dispondrá de un getter y setter para cada uno de los atributos de los que dispone.

#### 5.6.3.2 Rol

Clase	Rol
Descripción	Representa los roles existentes para usuarios acreditados en el sistema.
Tipo	Enumerado
Características	Concreta y estática
Responsabilidades	Mantiene los roles válidos para un usuario del sistema

Colaboraciones	Usuario
<b>Constructores:</b>	
No dispone	
<b>Atributos:</b>	
+USUARIO	Rol de un usuario del sistema
+ADMINISTRADOR	Rol de un administrador del sistema
+SUPERADMINISTRADOR	Rol de un superadministrador del sistema
<b>Métodos:</b>	Esta clase enumerada no dispondrá de métodos.

### 5.6.3.3 Estado

<b>Clase</b>	<b>Estado</b>
Descripción	Representa los estados por los que podrá transcurrir un usuario.
Tipo	Enumerado
Características	Concreta y estática
Responsabilidades	Mantiene los estados válidos para un usuario del sistema
Colaboraciones	Usuario
<b>Constructores:</b>	
No dispone	
<b>Atributos:</b>	
+SOLICITUD_EN_TRAMITE	Estado en el que el usuario ha hecho la solicitud de acceso y esta a la espera de contestación.
+ALTA	Estado en el que se encuentra el usuario una vez ha sido dado de alta en el sistema.
+BAJA	Estado en el que se encuentra un usuario que ha causado baja en el sistema.
<b>Métodos:</b>	Esta clase enumerada no dispondrá de métodos.

### 5.6.3.4 Permiso

<b>Clase</b>	<b>Permiso</b>
Descripción	Representa los permisos que manejará el sistema
Tipo	Enumerado
Características	Concreta y estática
Responsabilidades	Mantiene el catálogo de permisos que manejará el sistema
Colaboraciones	Usuario, Rol
<b>Constructores:</b>	

No dispone	
<b>Atributos:</b>	
+ACCESO_SISTEMA	Permiso de acceso al sistema
+ACCESO_INFORMACION_USUARIO	Permiso de acceso a la información del usuario
+GESTION_INFORMACION_USUARIO	Permiso de gestión de la información del usuario
+ACCESO_USUARIOS	Permiso de acceso a la información de otros usuarios
+GESTION_USUARIOS	Permiso de gestión de la información de otros usuarios
+ACCESO_PROGRAMACIONES	Permiso de acceso a programaciones
+GESTION_PROGRAMACIONES	Permiso de gestión de programaciones
+ACCESO_ELEMENTOS_TIPO	Permiso de acceso a elementos-tipo
+GESTION_ELEMENTOS_TIPO	Permiso de gestión de elementos-tipo
+ACCESO_RESERVAS_USUARIO	Permiso de acceso a las reservas del usuario
+ACCESO_RESERVAS	Permiso de acceso a reservas
+GESTION_RESERVAS	Permiso de gestión de reservas
+ACCESO_RESERVAS_HISTORICAS_USUARIO	Permiso de acceso a reservas históricas
+ACCESO_RESERVAS_HISTORICAS	Permiso de acceso a reservas históricas
+GESTION_RESERVAS_HISTORICAS	Permiso de gestión de reservas históricas
<b>Métodos:</b>	Esta clase enumerada no dispondrá de métodos.

#### 5.6.3.5 Sesión

Clase	Sesion
Descripción	Representa una sesión activa en el sistema
Tipo	Principal
Características	Concreta y volátil
Responsabilidades	Mantiene la información de una sesión valida del sistema
Colaboraciones	Usuario
<b>Constructores:</b>	
+Sesion()	
<b>Atributos:</b>	
-validador : String	Clave generada que sirve para determinar si la sesión es válida
-usuario : Usuario	Usuario para el que se crea la sesión
-fechaInicio : Dia	Día en el que se inició la sesión

-horaInicio : Hora	Hora en la que se inició la sesión
<b>Métodos:</b>	La clase es un bean por lo que dispondrá de un getter y setter para cada uno de los atributos de los que dispone.

### 5.6.3.6 FiltroUsuario

Clase	FiltroUsuario
Descripción	Representa los criterios de búsqueda de usuarios
Tipo	Principal
Características	Concreta y volátil
Responsabilidades	Mantener los criterios de búsqueda
Colaboraciones	Usuario
<b>Constructores:</b>	
+FiltroUsuario()	
<b>Atributos:</b>	
-email : String	Dirección de email del usuario
-nombre : String	Nombre del usuario
-apellidos : String	Apellidos del usuario
-rol : Rol	Rol del usuario
-estado : Estado	Estado del usuario
-tramitador : String	Usuario que ha tramitado el alta del usuario
-fechaIngreso : Dia	Día en que hizo el ingreso en la aplicación
<b>Métodos:</b>	La clase es un bean por lo que dispondrá de un getter y setter para cada uno de los atributos de los que dispone.

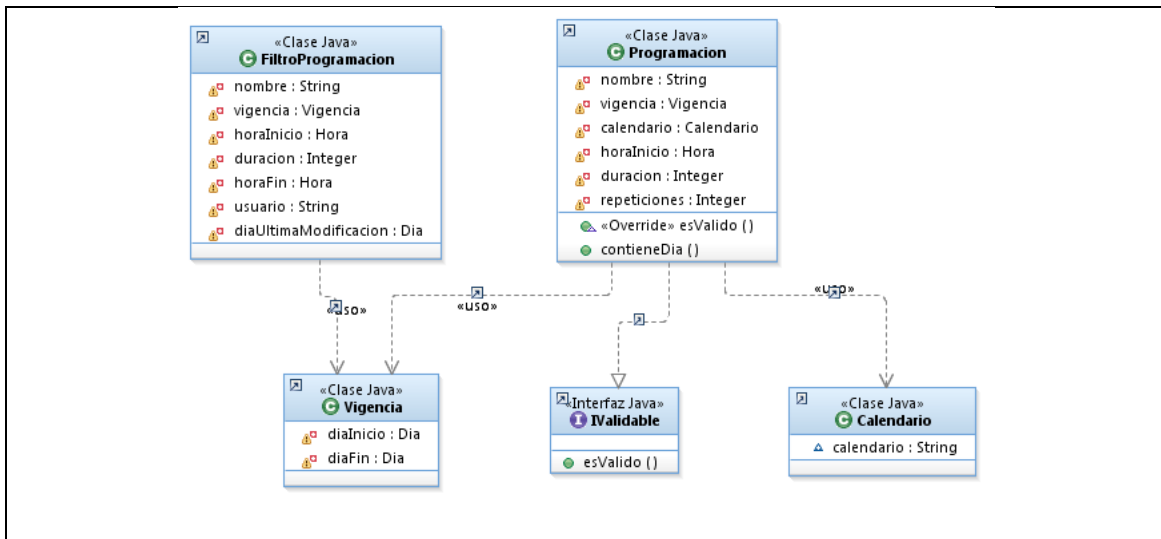
## 5.7 Subsistema de gestión de programaciones (SGP)

Este subsistema gestionará la creación, modificación y eliminación de programaciones del sistema. Ofrecerá también operaciones para la consulta de las entidades que gestiona.

### 5.7.1 Modelo de datos

El modelo de datos necesario para manejar la información relativa a este subsistema y a todo el conjunto de sus operaciones es el siguiente:

--

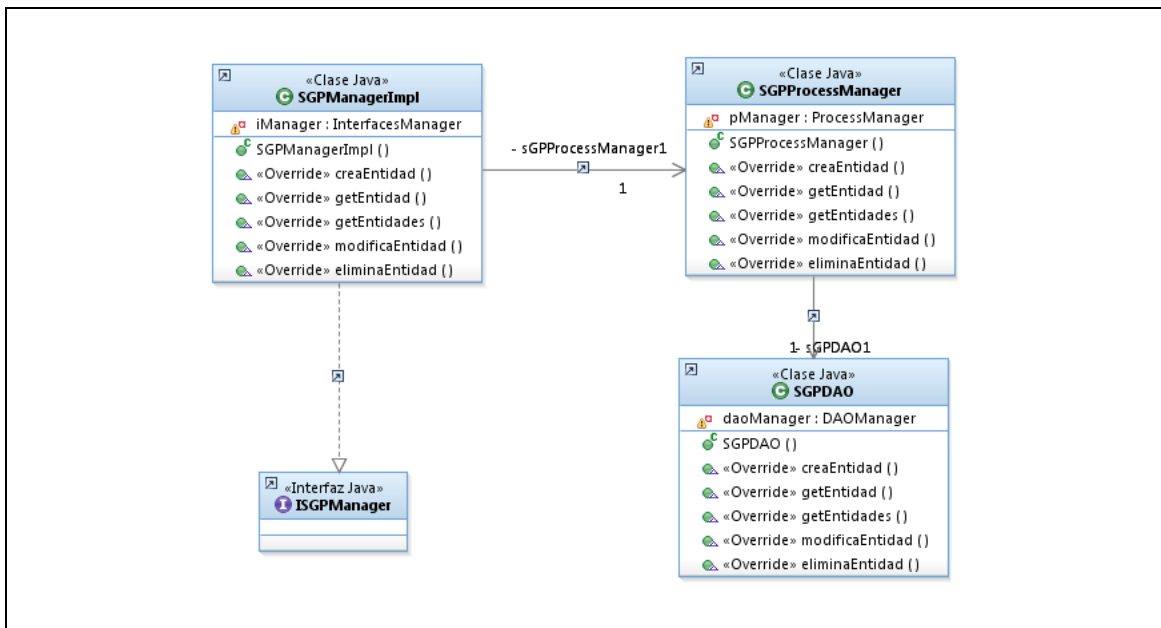


### 5.7.2 Negocio

En este punto se detalla todo lo correspondiente a las operaciones y lógica de negocio del SGP.

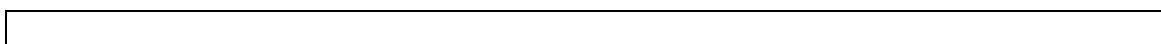
#### 5.7.2.1 Clases gestoras

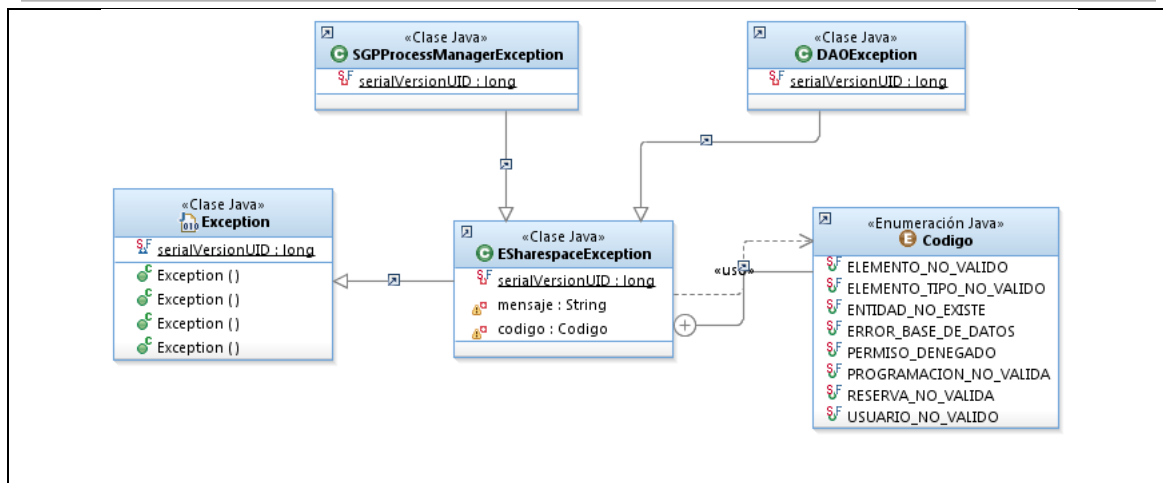
Estas serán las clases gestoras que formarán el subsistema del SGP:



#### 5.7.2.2 Excepciones

Estas serán las excepciones que manejará este subsistema:







### 5.7.3 Diagramas de secuencia

#### 5.7.3.1 Caso de uso – CU-5.1 – Listar programaciones

Este caso de uso está comprendido en la gestión básica de una entidad definida anteriormente ([Búsqueda de entidades](#)).

#### 5.7.3.2 Caso de uso – CU-5.1.1 – Crear programación

Este caso de uso está comprendido en la gestión básica de una entidad definida anteriormente ([Creación de una entidad](#)).

#### 5.7.3.3 Caso de uso – CU-5.1.2 – Modificar programación

Este caso de uso está comprendido en la gestión básica de una entidad definida anteriormente ([Modificación de una entidad](#)).

#### 5.7.3.4 Caso de uso – CU-5.1.3 – Consultar programación

Este caso de uso está comprendido en la gestión básica de una entidad definida anteriormente ([Búsqueda de una entidad](#)).

#### 5.7.3.5 Caso de uso – CU-5.1.4 – Eliminar programación

Este caso de uso está comprendido en la gestión básica de una entidad definida anteriormente ([Eliminación de una entidad](#)).

#### 5.7.4 Fichas CRC

##### 5.7.4.1 Programacion

<b>Clase</b>	<b>Programacion</b>
Descripción	Representa la información existente en el sistema acerca de una programación
Tipo	Principal
Características	Concreta y persistente
Responsabilidades	Proporcionar los datos de una programación concreta
Colaboraciones	Calendario, Vigencia, Elemento
<b>Constructores:</b>	
+Programacion()	
<b>Atributos:</b>	
-nombre : String	Nombre de la programación, este dato será clave primaria para la entidad.
-vigencia : Vigencia	Vigencia de la programación
-calendario : Calendario	Calendario de la programación
-horaInicio : Hora	Hora de inicio de la programación
-duración : Integer	Duración de un evento de la programación
-repeticiones : Integer	Repeticiones del evento
-usuario : String	Email del último usuario que modificó la programación
-ultimaModificacion : DiaHora	Día y hora de última modificación
<b>Métodos:</b>	
La clase es un bean por lo que dispondrá de un getter y setter para cada uno de los atributos de los que dispone.	

##### 5.7.4.2 Calendario

<b>Clase</b>	<b>Calendario</b>
Descripción	Representa la información existente en el sistema acerca del calendario de una programación
Tipo	Principal
Características	Concreta y persistente
Responsabilidades	Proporcionar el calendario de una programación
Colaboraciones	Programación
<b>Constructores:</b>	
+Calendario()	
<b>Atributos:</b>	
-calendario : String	Calendario definido, tendrá el siguiente formato:  PERIODICIDADES;DIAS_ACTIVOS;DIAS_NO_ACTIVOS

	<p>Un ejemplo: 1,3,5;20120501,20120503;20120502,20120504</p> <p>Significado:</p> <ul style="list-style-type: none"> <li>• Periodicidad (1,3,5): Todos los lunes, miércoles y viernes.</li> <li>• Días activos (20120501,20120503): días 1 y 3 de Mayo de 2012.</li> <li>• Días no activos (20120502,20120504): días 2 y 4 de Mayo de 2012.</li> </ul>
<b>Métodos:</b>	La clase es un bean por lo que dispondrá de un getter y setter para cada uno de los atributos de los que dispone.

#### 5.7.4.3 Vigencia

Clase	Vigencia
Descripción	Representa la información existente en el sistema acerca de la vigencia de una programación
Tipo	Principal
Características	Concreta y persistente
Responsabilidades	Proporcionar los datos de vigencia de una programación
Colaboraciones	Programación
<b>Constructores:</b>	
+Vigencia()	
<b>Atributos:</b>	
-diaInicio : Dia	Inicio de la vigencia en formato AAAAMMDD
-diaFin : Dia	Fin de la vigencia en formato AAAAMMDD
<b>Métodos:</b>	La clase es un bean por lo que dispondrá de un getter y setter para cada uno de los atributos de los que dispone.

#### 5.7.4.4 FiltroProgramacion

Clase	FiltroProgramacion
Descripción	Representa los criterios de búsqueda de programaciones
Tipo	Principal
Características	Concreta y volátil
Responsabilidades	Mantener los criterios de búsqueda
Colaboraciones	Programacion
<b>Constructores:</b>	
+FiltroProgramacion()	
<b>Atributos:</b>	
-nombre : String	Nombre de la programación

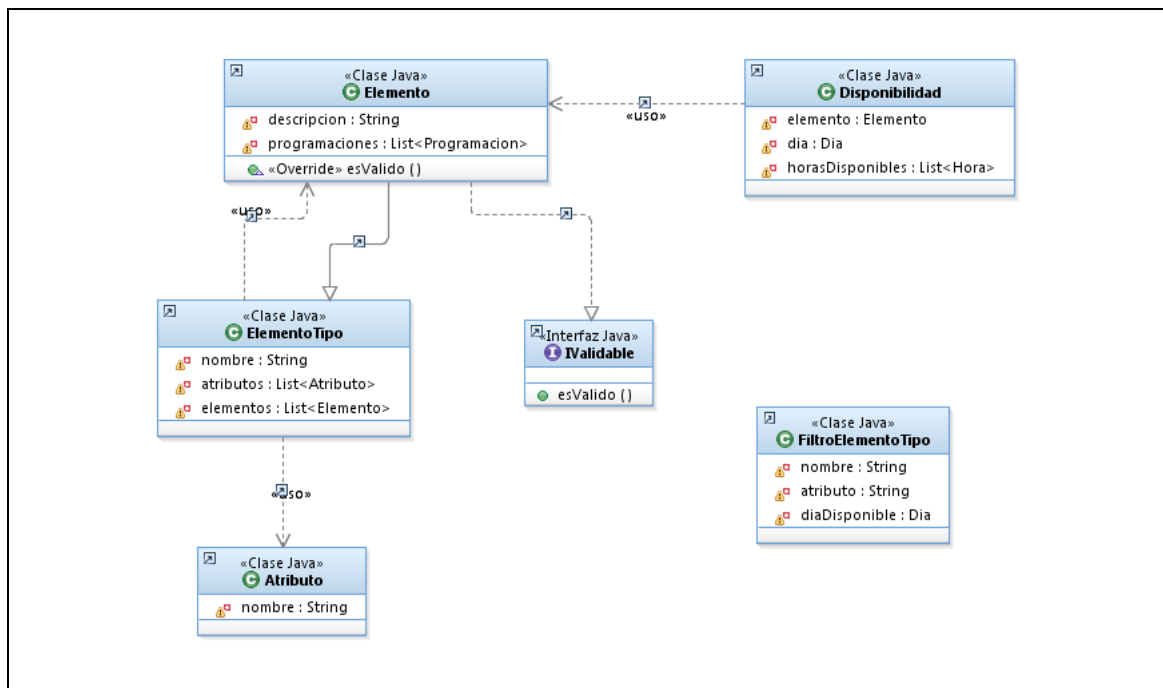
-vigencia : Vigencia	Vigencia en la que esté vigente la programación
-horaInicio : Hora	Hora de inicio de la programación
-duración : Integer	Duración de la programación
-horaFin : Integer	Hora de finalización de la programación
-usuario : String	Email del último usuario que modificó la programación
<b>Métodos:</b>	La clase es un bean por lo que dispondrá de un getter y setter para cada uno de los atributos de los que dispone.

## 5.8 Subsistema de gestión de elementos-tipo (SGET)

Este subsistema gestionará la creación, modificación y eliminación de elementos-tipo y sus correspondientes elementos del sistema. Ofrecerá también operaciones para la consulta de las entidades que gestiona.

### 5.8.1 Modelo de datos

El modelo de datos necesario para manejar la información relativa a este subsistema y a todo el conjunto de sus operaciones es el siguiente:

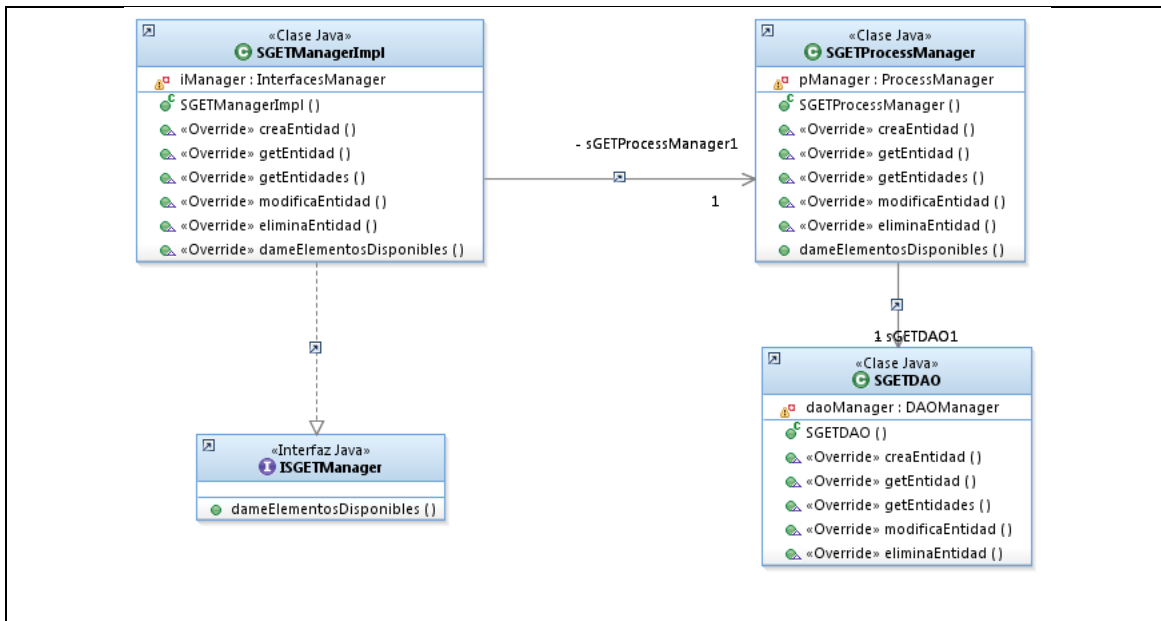


### 5.8.2 Negocio

En este punto se detalla todo lo correspondiente a las operaciones y lógica de negocio del SGET.

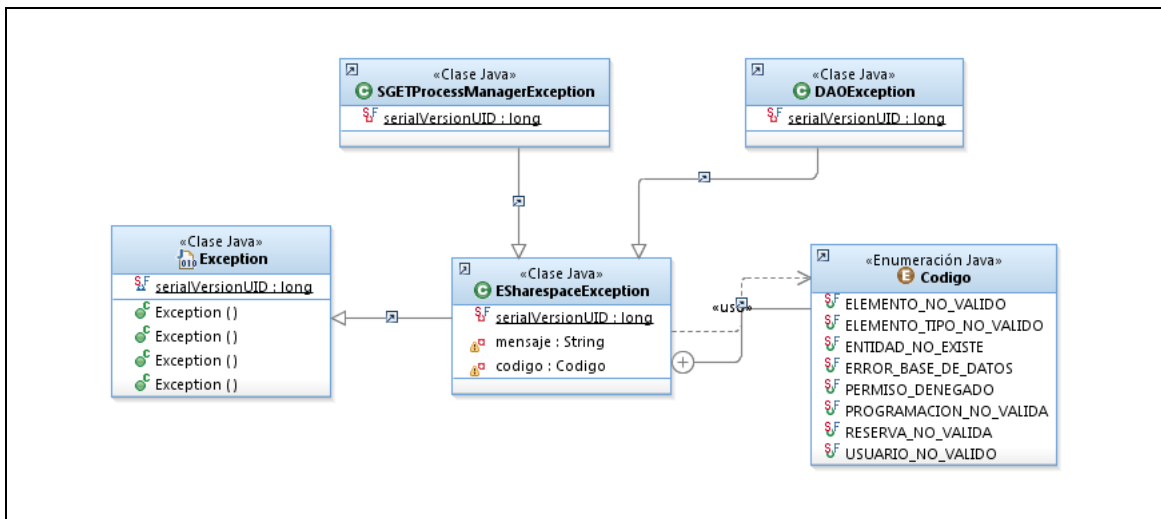
#### 5.8.2.1 Clases gestoras

Estas serán las clases gestoras que formarán el subsistema del SGET:



### 5.8.2.2 Excepciones

Estas serán las excepciones que manejará este subsistema:

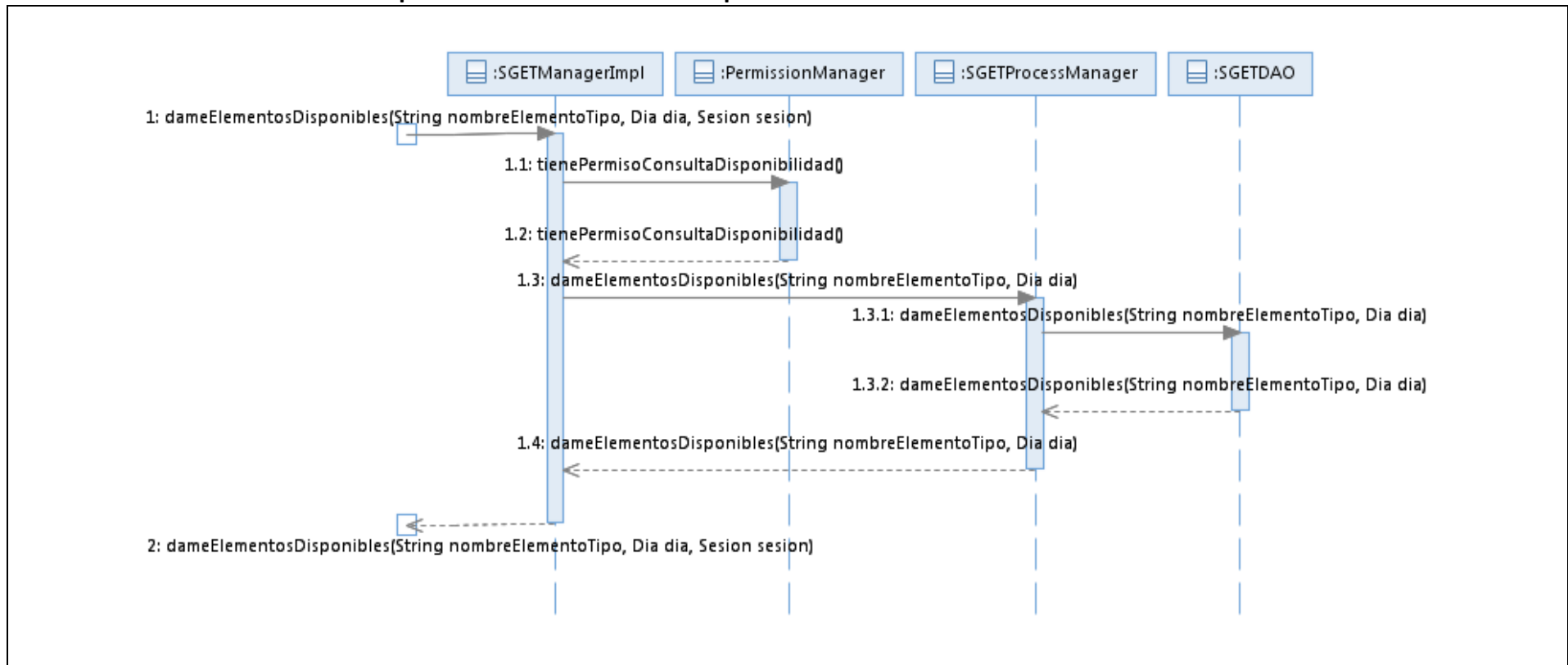


### 5.8.3 Diagramas de secuencia

#### 5.8.3.1 Caso de uso – CU-2.1 y CU-2.3 – Listar elementos-tipo

Este caso de uso está comprendido en la gestión básica de una entidad definida anteriormente ([Búsqueda de entidades](#)).

#### 5.8.3.2 Caso de uso – CU-2.2 – Disponibilidad de un elemento-tipo



#### 5.8.3.3 Caso de uso – CU-2.3.1 – Crear elemento-tipo

Este caso de uso está comprendido en la gestión básica de una entidad definida anteriormente ([Creación de una entidad](#)).

**5.8.3.4 Caso de uso – CU-2.3.2 – Modificar elemento-tipo**

Este caso de uso está comprendido en la gestión básica de una entidad definida anteriormente ([Modificación de una entidad](#)).

**5.8.3.5 Caso de uso – CU-2.3.3 – Consultar elemento-tipo**

Este caso de uso está comprendido en la gestión básica de una entidad definida anteriormente ([Búsqueda de una entidad](#)).

**5.8.3.6 Caso de uso – CU-2.3.4 – Eliminar elemento-tipo**

Este caso de uso está comprendido en la gestión básica de una entidad definida anteriormente ([Eliminación de una entidad](#)).

#### 5.8.4 Fichas CRC

##### 5.8.4.1 ElementoTipo

Clase	ElementoTipo
Descripción	Representa la información existente en el sistema de un elemento-tipo
Tipo	Principal
Características	Concreta y persistente
Responsabilidades	Proporcionar los datos de un elemento-tipo concreto
Colaboraciones	Elemento, Atributo
<b>Constructores:</b>	
+ElementoTipo()	
<b>Atributos:</b>	
-nombre : String	Nombre del elemento-tipo, será la clave primaria de la entidad.
-atributos : Atributo [1..*]	Atributos del elemento tipo.
-elementos : Elemento [0..*]	Elementos definidos bajo este elemento-tipo
-usuario : Usuario	Email del último usuario que ha modificado el elemento-tipo
-ultimaModificación : DiaHora	Día y hora de la última modificación de un elemento-tipo
<b>Métodos:</b>	
La clase es un bean por lo que dispondrá de un getter y setter para cada uno de los atributos de los que dispone.	

##### 5.8.4.2 Atributo

Clase	Atributo
Descripción	Representa un atributo definido en el sistema
Tipo	Principal
Características	Concreta y persistente
Responsabilidades	Proporcionar un catálogo de atributos con los que categorizar los elementos-tipo
Colaboraciones	Elemento-Tipo
<b>Constructores:</b>	
+Atributo()	
<b>Atributos:</b>	
-nombre : String	Nombre y valor del atributo
<b>Métodos:</b>	
La clase es un bean por lo que dispondrá de un getter y setter para cada uno de los atributos de los que dispone.	



#### 5.8.4.3 Elemento

Clase	Elemento
Descripción	Representa la información existente en el sistema de un elemento
Tipo	Principal
Características	Concreta y persistente
Responsabilidades	Proporcionar los datos de un elemento concreto
Colaboraciones	Elemento-Tipo, Programacion, Reserva, Disponibilidad
<b>Constructores:</b>	
+Elemento ()	
<b>Atributos:</b>	
-descripcion : String	Descripción del elemento.
-programaciones : Programacion [1..*]	Programaciones definidas para el elemento.
<b>Métodos:</b>	La clase es un bean por lo que dispondrá de un getter y setter para cada uno de los atributos de los que dispone.

#### 5.8.4.4 Disponibilidad

Clase	Disponibilidad
Descripción	Representa la disponibilidad para un elemento en un día concreto
Tipo	Principal
Características	Concreta y volátil
Responsabilidades	Proporcionar la disponibilidad del elemento para un día
Colaboraciones	Elemento, Dia
<b>Constructores:</b>	
+Disponibilidad()	
<b>Atributos:</b>	
-elemento : Elemento	Elemento para el que se informa su disponibilidad
-dia : Dia	Día para el que solicita la disponibilidad
-horasDisponibles : Hora [0..*]	Lista de horas disponibles para el elemento en cuestión
<b>Métodos:</b>	La clase es un bean por lo que dispondrá de un getter y setter para cada uno de los atributos de los que dispone.

#### 5.8.4.5 FiltroElementoTipo

Clase	FiltroElementoTipo
Descripción	Representa los criterios de búsqueda de elementos-tipo
Tipo	Principal
Características	Concreta y volátil

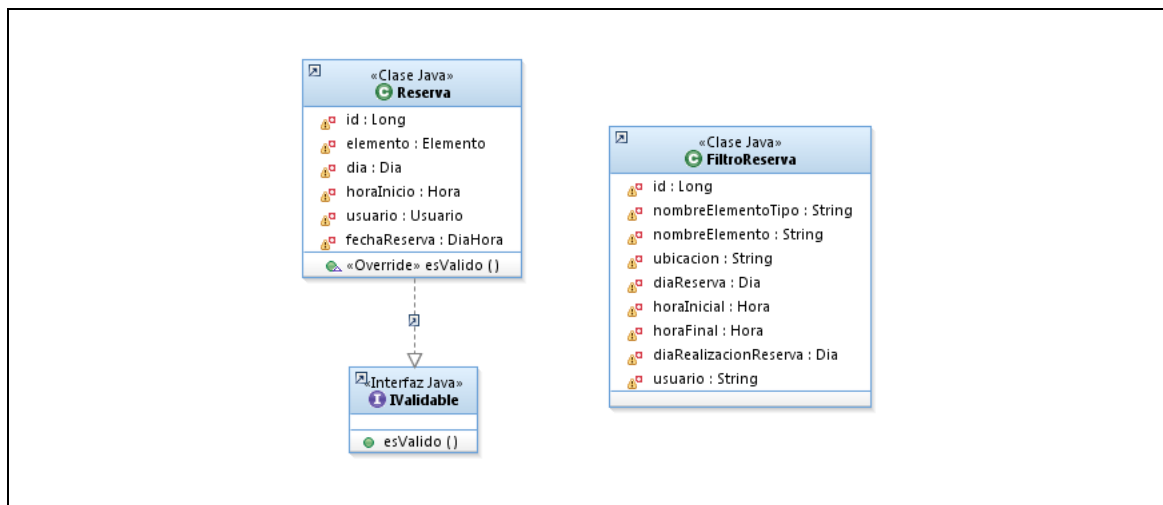
Responsabilidades	Mantener los criterios de búsqueda
Colaboraciones	ElementoTipo
<b>Constructores:</b>	
+FiltroElementoTipo()	
<b>Atributos:</b>	
-nombre : String	Nombre del elemento-tipo
-atributo : String	Nombre del atributo
-diaDisponible : Dia	Día en el que el elemento-tipo tiene al menos un elemento disponible.
<b>Métodos:</b>	
La clase es un bean por lo que dispondrá de un getter y setter para cada uno de los atributos de los que dispone.	

## 5.9 Subsistema de gestión de reservas (SGR)

Este subsistema gestionará la creación y eliminación de reservas del sistema. Ofrecerá también operaciones para la consulta de las entidades que gestiona.

### 5.9.1 Modelo de datos

El modelo de datos necesario para manejar la información relativa a este subsistema y a todo el conjunto de sus operaciones es el siguiente:

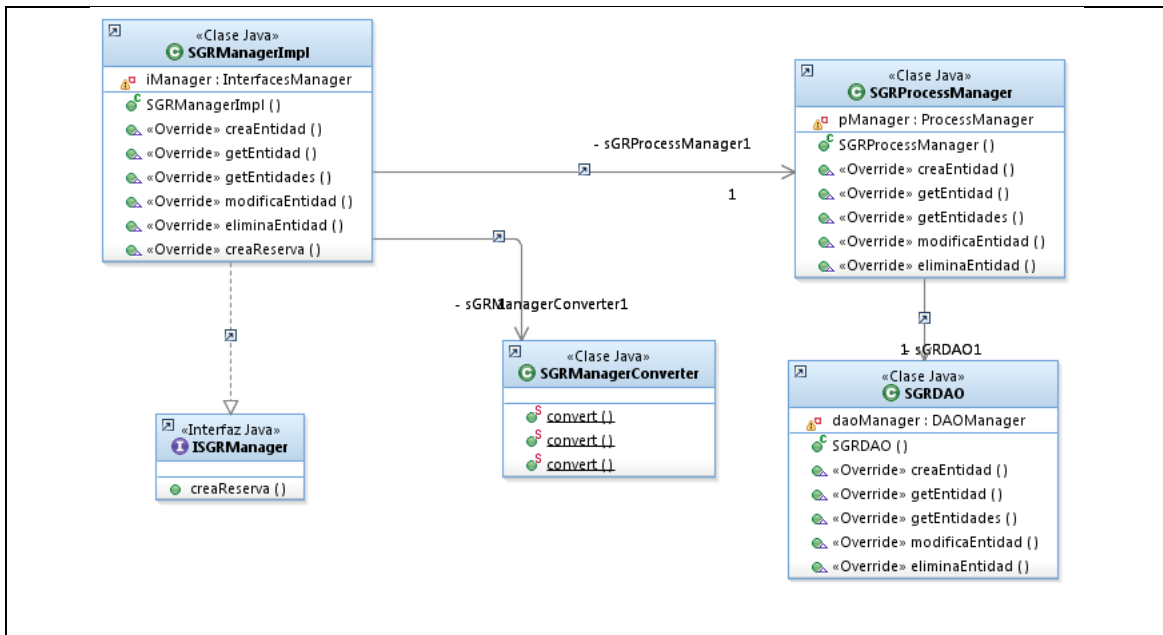


### 5.9.2 Negocio

En este punto se detalla todo lo correspondiente a las operaciones y lógica de negocio del SGR.

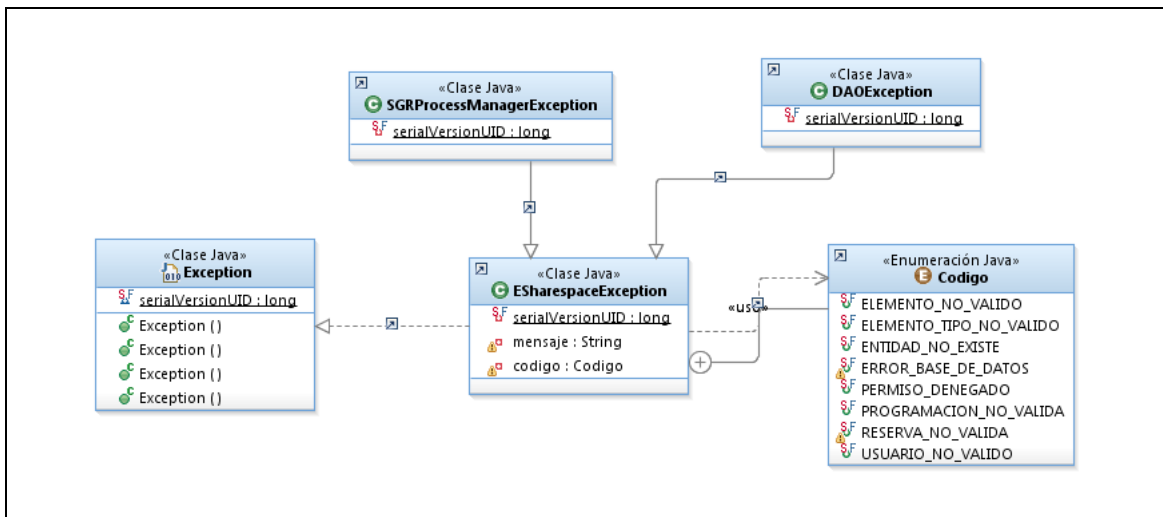
#### 5.9.2.1 Clases gestoras

Estas serán las clases gestoras que formarán el subsistema del SGR:



### 5.9.2.2 Excepciones

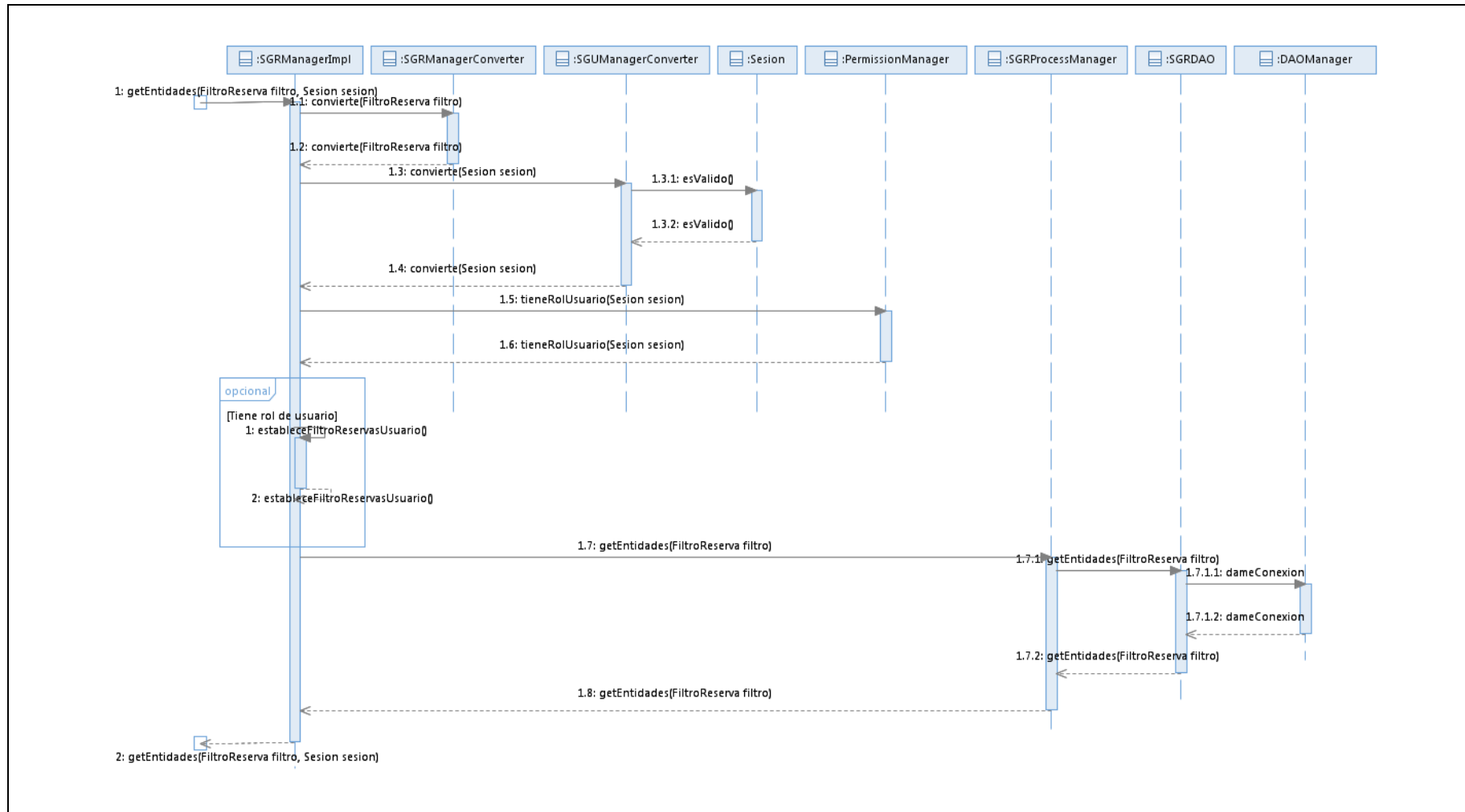
Estas serán las excepciones que manejará este subsistema:



### 5.9.3 Diagramas de secuencia

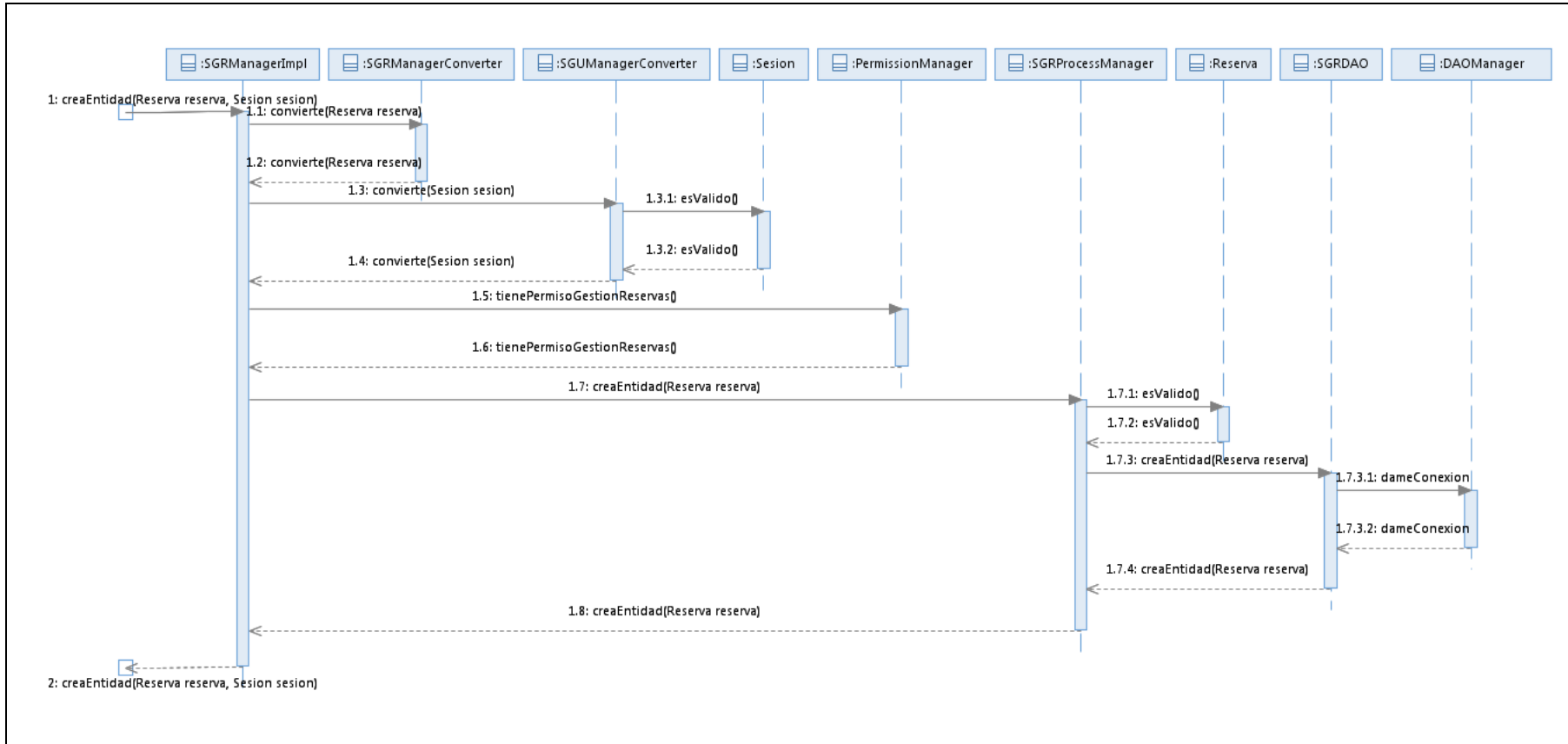
#### 5.9.3.1 Caso de uso – CU-3.1 y CU-3.6 – Listar reservas de usuario

Esta será la secuencia para el caso de uso:



### 5.9.3.2 Caso de uso – CU-3.2 - Crear reserva

Esta será la secuencia para el caso de uso:

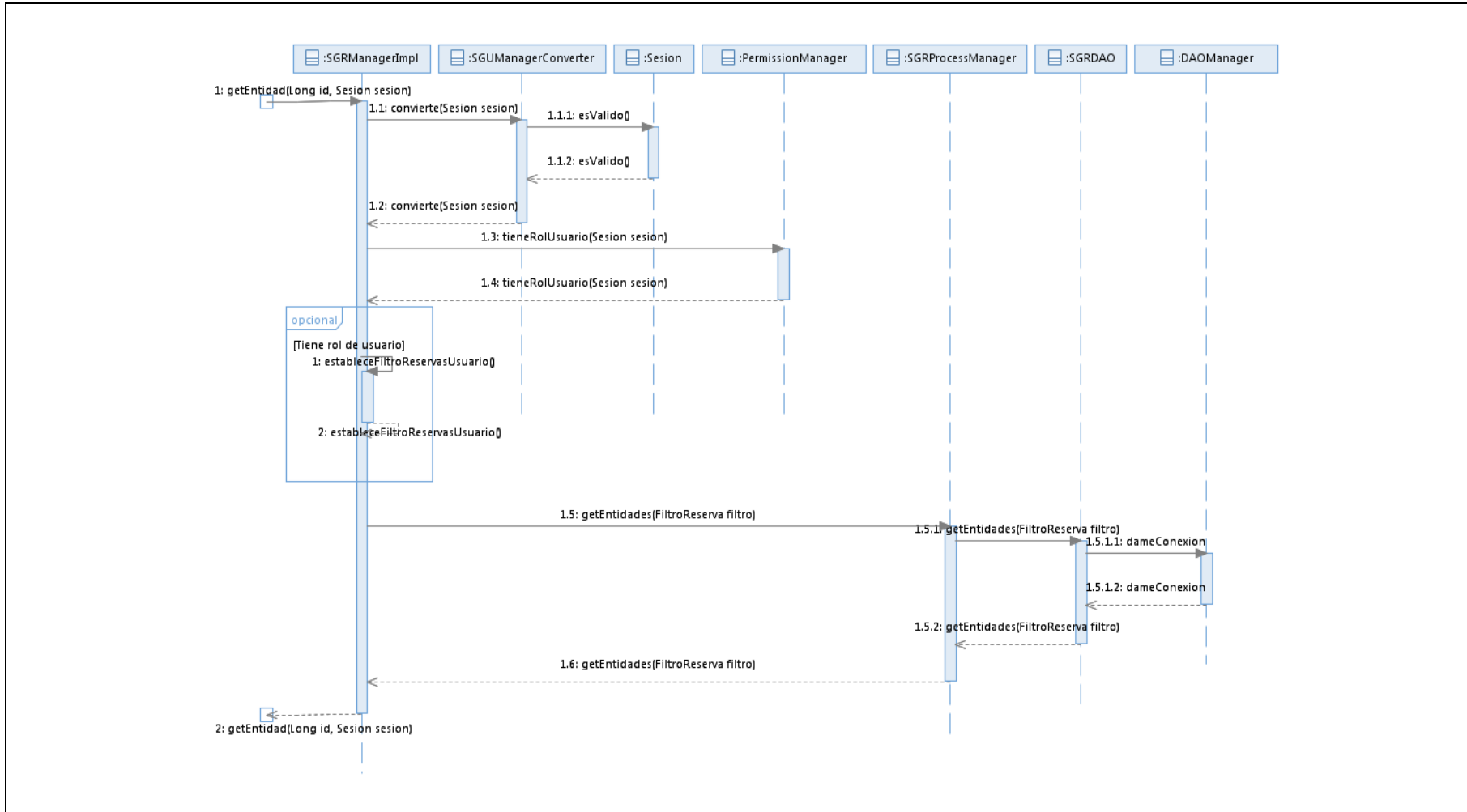


### 5.9.3.3 Caso de uso – CU-3.3 - Anular una reserva

Este caso de uso está comprendido en la gestión básica de una entidad definida anteriormente ([Eliminación de una entidad](#)).

### 5.9.3.4 Caso de uso – CU-3.4 - Consultar reserva

Esta será la secuencia para el caso de uso:



## 5.9.4 Fichas CRC

### 5.9.4.1 Reserva

Clase	Reserva
Descripción	Representa la información de una reserva de un elemento para un día y hora concreto en el sistema
Tipo	Principal
Características	Concreta y persistente
Responsabilidades	Proporcionar los datos de una reserva
Colaboraciones	Usuario, Elemento
<b>Constructores:</b>	
+Reserva()	
<b>Atributos:</b>	
-id : Long	Identificador de la reserva
-elemento : Elemento	Elemento para el que se ha hecho esta reserva
-dia : Dia	Dia para el que se ha realizado la reserva
-horaInicio : Hora	Hora de inicio de la reserva
-usuario : Usuario	Usuario para el que se ha realizado la reserva
-fechaReserva : DiaHora	Momento en el que se realizó la reserva
<b>Métodos:</b>	La clase es un bean por lo que dispondrá de un getter y setter para cada uno de los atributos de los que dispone.

### 5.9.4.2 FiltroReserva

Clase	FiltroReserva
Descripción	Representa los criterios de búsqueda de reservas
Tipo	Principal
Características	Concreta y volátil
Responsabilidades	Mantener los criterios de búsqueda
Colaboraciones	Reserva
<b>Constructores:</b>	
+FiltroReserva()	
<b>Atributos:</b>	
-id : Long	Identificador de la reserva
-nombreElementoTipo : String	Nombre del elemento-tipo para el que se ha realizado la reserva
-nombreElemento : String	Nombre del elemento concreto para el que se ha realizado la reserva
-ubicación : String	Ubicación del elemento reservado
-diaReserva : Dia	Día para el que se ha realizado la reserva del elemento

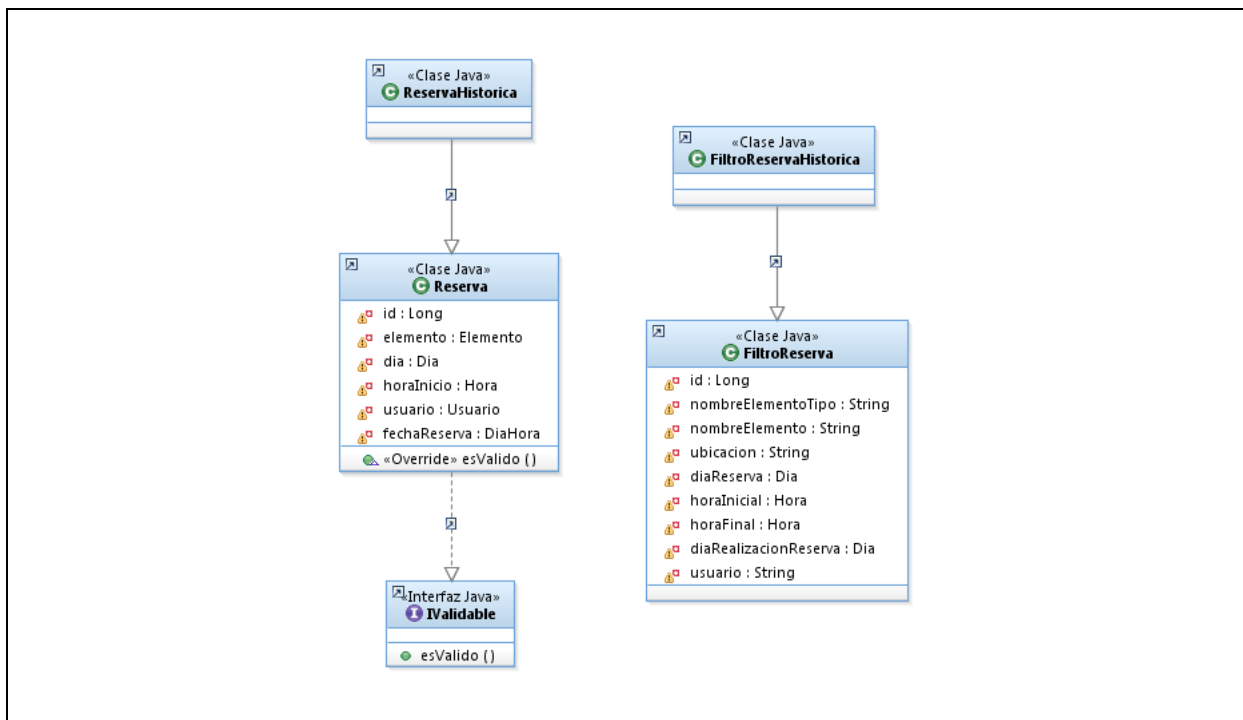
-horaInicial : Hora	Hora de inicio de la reserva
-horaFinal : Hora	Hora de finalización de la reserva
-diaRealizacionReserva : Dia Dia	Día en que se realizó la reserva
-usuario : String	Email del usuario que realizó la reserva
<b>Métodos:</b>	La clase es un bean por lo que dispondrá de un getter y setter para cada uno de los atributos de los que dispone.

### 5.10 Subsistema de gestión de reservas históricas (SGRH)

Este subsistema ofrecerá al resto del sistema la información de las reservas expiradas (históricas). Esta entidad no puede crearse (ya que lo que realmente se crea es la reserva) ni eliminarse para dejar un registro del que poder sacar datos estadísticos en un futuro.

#### 5.10.1 Modelo de datos

El modelo de datos necesario para manejar la información relativa a este subsistema y a todo el conjunto de sus operaciones es el siguiente:



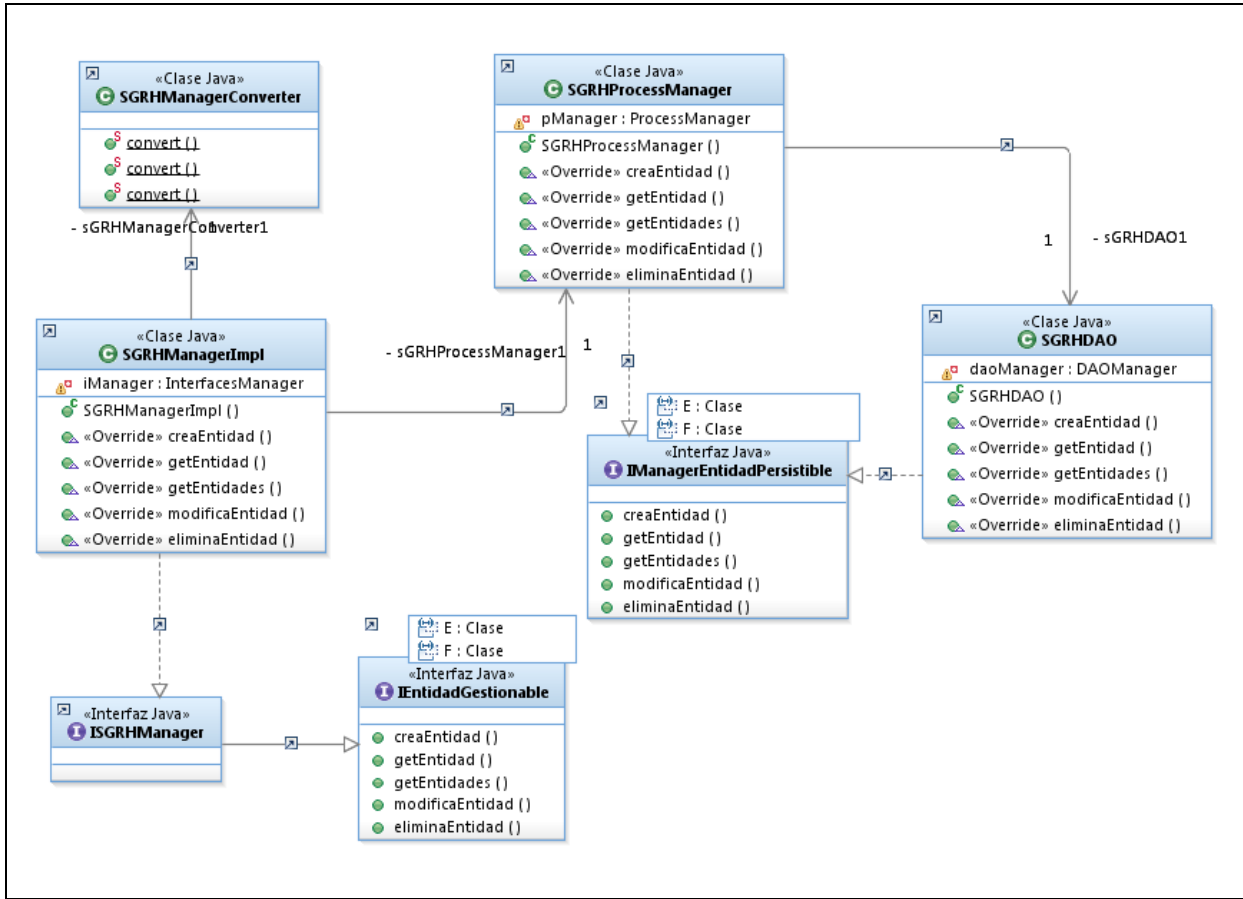
#### 5.10.2 Negocio

En este punto se detalla todo lo correspondiente a las operaciones y lógica de negocio del SGRH.

##### 5.10.2.1 Clases gestoras

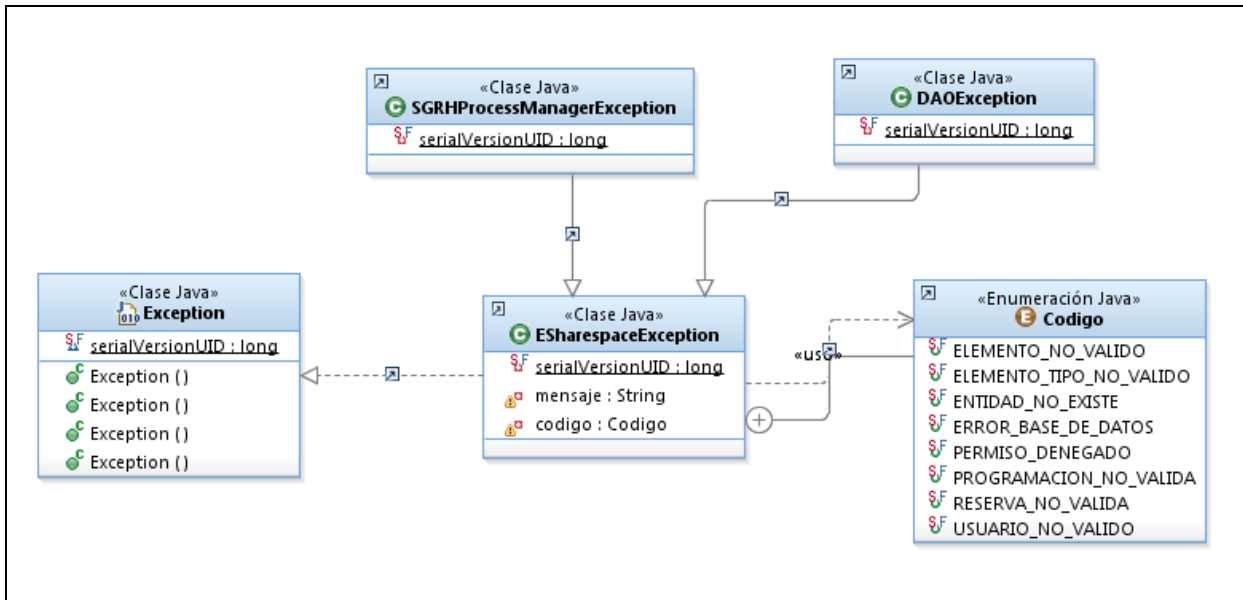
Estas serán las clases gestoras que formarán el subsistema del SGRH:





### 5.10.2.2 Excepciones

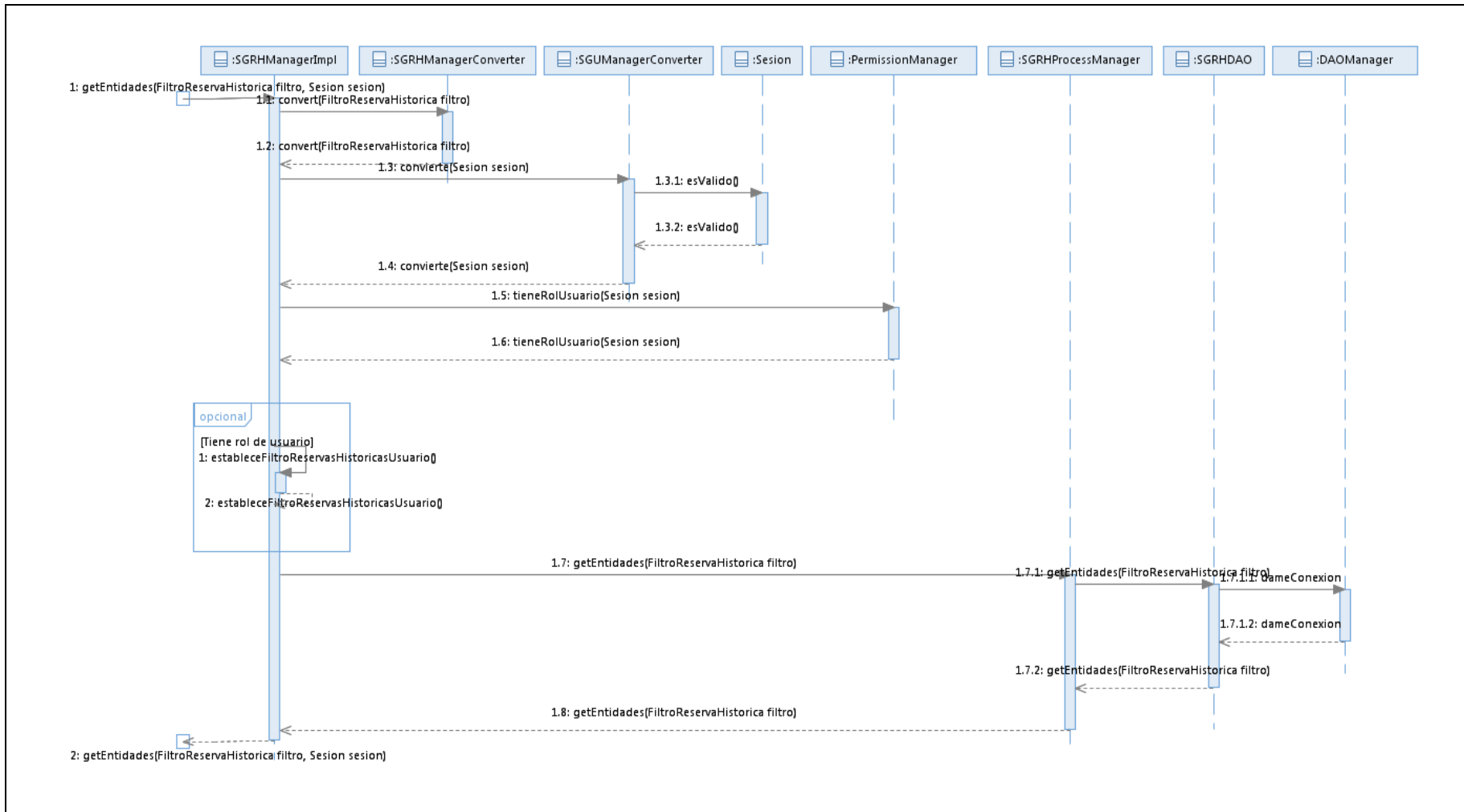
Estas serán las excepciones que manejará este subsistema:



### 5.10.3 Diagramas de secuencia

#### 5.10.3.1 Caso de uso – CU-4.1 y CU-4.2 – Listar reservas históricas

Esta será la secuencia para el caso de uso:



#### 5.10.4 Fichas CRC

##### 5.10.4.1 ReservaHistorica

<b>Clase</b>	<b>ReservaHistorica</b>
Descripción	Representa la información de una reserva histórica de un elemento para un día y hora concreto en el sistema
Tipo	Principal y extiende a Reserva
Características	Concreta y persistente
Responsabilidades	Proporcionar los datos de una reserva histórica
Colaboraciones	Usuario, Elemento
<b>Constructores:</b>	
+ReservaHistorica()	
<b>Atributos:</b>	
-id : Long	Identificador de la reserva histórica
-elemento : Elemento	Elemento para el que se ha hecho esta reserva
-dia : Dia	Dia para el que se ha realizado la reserva
-horaInicio : Hora	Hora de inicio de la reserva
-usuario : Usuario	Usuario para el que se ha realizado la reserva
-fechaReserva : DiaHora	Momento en el que se realizó la reserva
<b>Métodos:</b>	La clase es un bean por lo que dispondrá de un getter y setter para cada uno de los atributos de los que dispone.

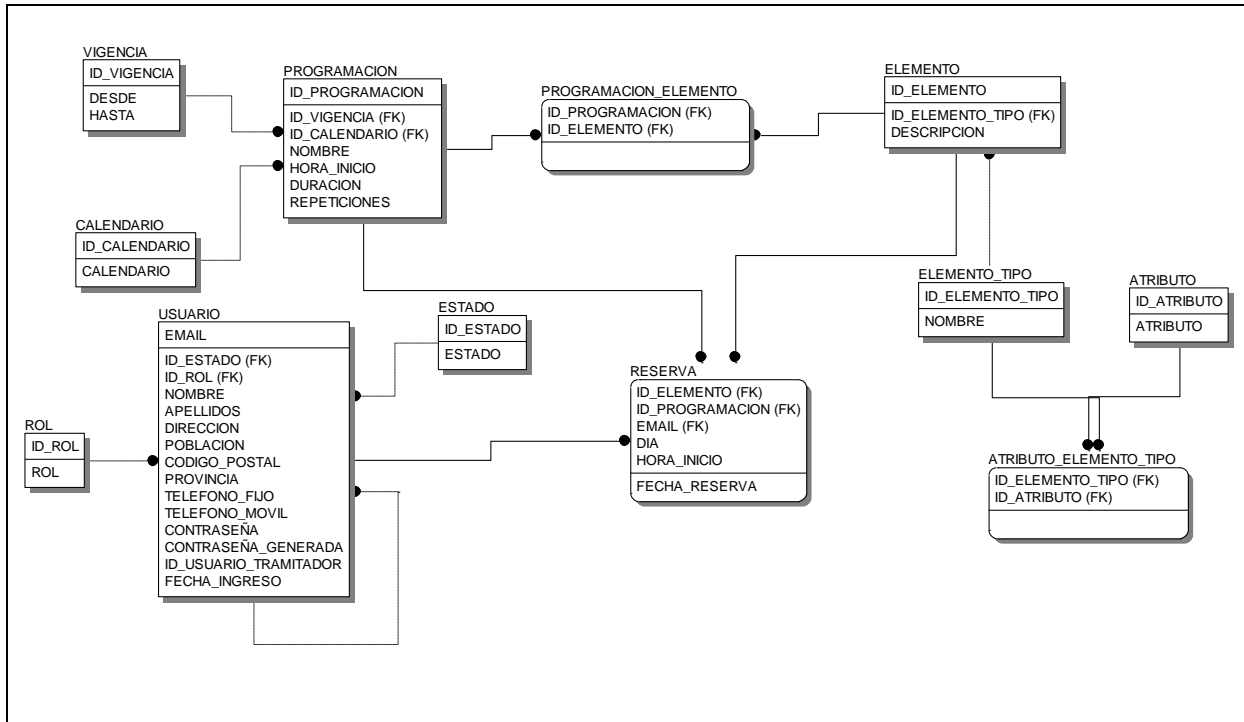
##### 5.10.4.2 FiltroReservaHistorica

<b>Clase</b>	<b>FiltroReservaHistorica</b>
Descripción	Representa los criterios de búsqueda de reservas históricas
Tipo	Principal
Características	Concreta y volátil
Responsabilidades	Mantener los criterios de búsqueda
Colaboraciones	ReservaHistorica
<b>Constructores:</b>	
+FiltroReservaHistorica()	
<b>Atributos:</b>	
-id : Long	Identificador de la reserva histórica
-nombreElementoTipo : String	Nombre del elemento-tipo para el que se ha realizado la reserva
-nombreElemento : String	Nombre del elemento concreto para el que se ha realizado la reserva
-ubicación : String	Ubicación del elemento reservado

-diaReserva : Dia	Día para el que se ha realizado la reserva del elemento
-horaInicial : Hora	Hora de inicio de la reserva
-horaFinal : Hora	Hora de finalización de la reserva
-diaRealizacionReserva : Dia	Día en que se realizó la reserva
-usuario : String	Email del usuario que realizó la reserva
<b>Métodos:</b>	La clase es un bean por lo que dispondrá de un getter y setter para cada uno de los atributos de los que dispone.

### 5.11 Diseño de base de datos

Este será el modelo de base de datos necesario para persistir la información de la aplicación.



#### 5.11.1 Modelo de tablas de base de datos

<b>USUARIO</b>			
Atributo	Tipo	Clave principal	Clave foránea
EMAIL	VARCHAR	SI	
ID_ROL	INTEGER		SI
ID_ESTADO	INTEGER		SI
NOMBRE	VARCHAR		

<b>APELLIDOS</b>	VARCHAR		
<b>DIRECCION</b>	VARCHAR		
<b>POBLACION</b>	VARCHAR		
<b>CODIGO_POSTAL</b>	VARCHAR		
<b>PROVINCIA</b>	VARCHAR		
<b>TELEFONO_FIJO</b>	NUMBER		
<b>TELEFONO_MOVIL</b>	NUMBER		
<b>CONTRASEÑA</b>	VARCHAR		
<b>CONTRASEÑA_GENERADA</b>	VARCHAR		
<b>ID_USUARIO_TRAMITADOR</b>	INTEGER		SI
<b>FECHA_INGRESO</b>	DATE		

### ROL

Atributo	Tipo	Clave principal	Clave foránea
<b>ID_ROL</b>	INTEGER	SI	
<b>ROL</b>	VARCHAR		

### ESTADO

Atributo	Tipo	Clave principal	Clave foránea
<b>ID_ESTADO</b>	INTEGER	SI	
<b>ESTADO</b>	VARCHAR		

### PROGRAMACION

Atributo	Tipo	Clave principal	Clave foránea
<b>ID_PROGRAMACION</b>	INTEGER	SI	
<b>ID_CALENDARIO</b>	INTEGER		SI
<b>ID_VIGENCIA</b>	INTEGER		SI
<b>NOMBRE</b>	VARCHAR		
<b>HORA_INICIO</b>	INTEGER		
<b>DURACION</b>	INTEGER		
<b>REPETICIONES</b>	INTEGER		

### VIGENCIA

Atributo	Tipo	Clave principal	Clave foránea
<b>ID_VIGENCIA</b>	INTEGER	SI	
<b>DESDE</b>	INTEGER		
<b>HASTA</b>	INTEGER		

### CALENDARIO

Atributo	Tipo	Clave principal	Clave foránea
ID_CALEDARIO	INTEGER	SI	
CALENDARIO	VARCHAR		

<b>ELEMENTO_TIPO</b>			
Atributo	Tipo	Clave principal	Clave foránea
ID_ELEMENTO_TIPO	INTEGER	SI	
NOMBRE	VARCHAR		

<b>ATRIBUTO</b>			
Atributo	Tipo	Clave principal	Clave foránea
ID_ATRIBUTO	INTEGER	SI	
ATRIBUTO	VARCHAR		

<b>ATRIBUTO_ELEMENTO_TIPO</b>			
Atributo	Tipo	Clave principal	Clave foránea
ID_ELEMENTO_TIPO	INTEGER	SI	SI
ID_ATRIBUTO	INTEGER	SI	SI

<b>ELEMENTO</b>			
Atributo	Tipo	Clave principal	Clave foránea
ID_ELEMENTO	INTEGER	SI	
ID_ELEMENTO_TIPO	INTEGER		SI
DESCRIPCION	VARCHAR		

<b>PROGRAMACION_ELEMENTO</b>			
Atributo	Tipo	Clave principal	Clave foránea
ID_PROGRAMACION	INTEGER	SI	SI
ID_ELEMENTO	INTEGER	SI	SI

<b>RESERVA</b>			
Atributo	Tipo	Clave principal	Clave foránea
ID_ELEMENTO	INTEGER	SI	SI
ID_PROGRAMACION	INTEGER	SI	SI
EMAIL	VARCHAR	SI	SI
DIA	INTEGER	SI	
HORA_INICIO	INTEGER	SI	
FECHA_RESERVA	DATE		

## 6 Comportamiento del sistema

### 6.1 Usabilidad

- La interface de usuario seguirá el estilo definido por las aplicaciones de Windows.
- El sistema detectará el idioma de preferencia del usuario y para mostrar los textos y recursos del IHM en ese idioma.
- Si no se dispone del idioma de preferencia del usuario se presentarán en inglés.
- Los botones se ordenarán de tal manera que los que definen acciones de confirmación o inicio de una operación se ubicarán a la izquierda. Los que definen cancelación o salida de la vista actual se ubicarán a la derecha.
- Se tendrán las siguientes consideraciones acerca de las tablas:
- Todas las tablas ofrecerán un panel que permitirá de filtrar por cada una de las columnas mostradas.
- Todas las tablas permitirán la ordenación de su contenido mediante un clic en el título de la columna. Un segundo clic realizará un orden inverso.
- Todas las tablas presentarán una paginación de su contenido permitiendo la navegación a la primera y última página además de hacia la siguiente y la anterior. El número de elementos mostrado en una página será también definido mediante un combo con los siguientes valores de 25, 50, 100 y 200.
- En la apertura de una vista que contenga una tabla esta se iniciará de tal manera que mostrará los primeros 25 elementos encontrados al ordenar por la primera columna de la tabla en modo ascendente.
- Se tendrán las siguientes consideraciones con respecto a las vistas:
- Todas las vistas dispondrán de una barra de herramientas sobre la que se ubicarán las acciones correspondientes a la vista en cuestión alineadas a la izquierda de la barra.
- Todas las barras de herramientas dispondrán de un botón alineado a la derecha de acceso a la ayuda contextual. Esta ayuda describirá la vista y todas las posibles operaciones.
- Todas las vistas permitirán la exportación a Excel y PDF de su contenido.
- El sistema guardará un log con el detalle de todas las operaciones y de los usuarios y puestos desde donde se invocaron.

## 7 Glosario

Esta sección contiene los términos y referencias utilizados en este documento.

- **eSharespace:** Nombre de la aplicación.
- **TFC:** Trabajo final de carrera.
- **Elemento-Tipo:** Es la definición abstracta de un elemento. Sirve para definir plantillas o grupos de elementos del mismo tipo.
- **Ranura:** Es el rango horario de un elemento dentro de su programación que no tiene reserva.
- **IHM:** Interface Hombre Máquina.
- **Captcha:** Son las siglas de **Completely Automated Public Turing test to tell Computers and Humans Apart**. Se trata de una prueba para diferenciar a humanos de máquinas.

## 8 Bibliografía

- Xhafa, Fatos, Técnicas de desarrollo de software. Fundació per a la Universitat Oberta de Catalunya, 2003.
- Campderrich Falgueras, Benet, Recerca Informàtica, S.L., Ingeniería del Software. Fundació per a la Universitat Oberta de Catalunya, 2004.
- Sistac Planas, Jaume, Bases de Datos I. Fundació per a la Universitat Oberta de Catalunya, 2005.

## 9 Anexos

### 9.1 Software

Este ha sido el software utilizado para la realización de este documento:

- Procesador de texto:
  - Microsoft Office 2007.
- Modelado UML:
  - IBM Rational Software Architect.
  - Omnigraffle
- Modelado ER:
  - ERwin.
- Prototipos de pantallas
  - Balsamiq Mockups.



## 9.2 Planificación del TFC

Esta ha sido la planificación utilizada para la consecución del TFC.

