

Gestión de expedientes para ayuntamientos.

# **Gestión de expedientes para ayuntamientos.**

Aplicación del marco de trabajo GRAILS.

**Rafael Romero Cacho**  
ETIS

**Albert Grau Perise**

**25 de Junio de 2008**

Gestión de expedientes para ayuntamientos.

Dedicado a mi hija Elisa y mi mujer M<sup>a</sup> Jesús  
porque sin su apoyo no lo habría podido conseguir.

Gestión de expedientes para ayuntamientos.

## **Contenido**

En este documento se detalla el análisis, diseño e implementación de una solución para la gestión de expedientes administrativos con sus respectivas solicitudes.

No es un objetivo de este proyecto dar una solución definitiva (entendiéndose como definitiva operativa) para la gestión real de los expedientes de un ayuntamiento pero a tenor de los resultados obtenidos bien podría servir como una buena base para un proyecto más ambicioso.

Para la realización del proyecto se han utilizado diversas tecnologías muy frecuentes en J2EE, pero integradas en un sólo marco de trabajo, GRAILS, el cual aplica el patrón MVC.

La primera parte se dedica al análisis y diseño de la aplicación. A continuación se explica las herramientas que se van a utilizar para su desarrollo. Por último se aportan imágenes de la aplicación desarrollada.

## Índice de contenido

Gestión de expedientes para ayuntamientos.....	1
Objetivo del TFC.....	7
Enfoque y método seguido.....	7
Planificación.....	7
Producto obtenido.....	8
Contenido específico de la memoria.....	8
Especificación del proyecto y análisis.....	9
Descripción.....	9
Software.....	9
Funcionalidades.....	11
Gestión de solicitudes .....	11
Alta de solicitud.....	11
Modificar solicitud.....	11
Retirar solicitud .....	11
Listar solicitudes.....	11
Buscar solicitud.....	11
Gestión de expedientes.....	12
Alta de expediente .....	12
Versionar expediente .....	12
Cerrar expediente .....	12
Listar expedientes.....	12
Buscar expediente.....	12
Gestión de usuarios.....	13
Alta de usuario.....	13
Modificar usuario.....	13
Dar de baja usuario.....	13
Listar usuarios.....	13
Buscar usuario.....	13
Mantenimientos.....	13
Gestión de solicitudes .....	14
Alta de solicitud.....	14
Modificar solicitud.....	14
Retirar solicitud .....	14
Listar solicitudes.....	14
Buscar solicitud.....	14
Gestión de expedientes.....	15
Alta de expediente .....	15
Versionar expediente .....	15
Cerrar expediente .....	15
Listar expedientes.....	15
Buscar expediente.....	15
Gestión de usuarios.....	16
Alta de usuario.....	16
Modificar usuario.....	16
Dar de baja usuario.....	16
Listar usuarios.....	16
Buscar usuario.....	16
Mantenimientos.....	16

## Gestión de expedientes para ayuntamientos.

Lista de paquetes.....	17
Lista de Casos de Uso.....	17
Lista de Classificadores.....	17
Diagramas.....	19
Diagrama de actividad.....	19
Definición.....	20
Lista de elementos.....	20
Partición.....	20
Diagrama de clases.....	26
Casos de uso.....	28
Diagrama de estados.....	30
Definición.....	30
Diagrama de secuencias - Administrador.....	32
Diagrama de secuencias - Alcalde.....	34
Diagrama de secuencias - Secretario.....	36
Diagrama de secuencias - Solicitante.....	38
Diagrama de actividad - Casos de uso.....	40
abrirExpediente .....	40
aceptarEnTramite .....	40
aceptarFinal .....	40
altaSolicitud .....	40
bajaUsuario .....	40
cerrarExpediente .....	42
consultarExpedientes .....	42
consultarSolicitudes .....	42
consultarUsuarios .....	42
crearUsuario .....	42
modificarExpediente .....	42
modificarSolicitud .....	44
modificarUsuario .....	44
rechazarSolicitud .....	44
requerirMasDocumentacion .....	44
retirarSolicitud .....	44
Clases.....	46
Administrador [Actor].....	46
Alcalde [Actor].....	46
DAO .....	46
Expediente .....	48
Fichero .....	50
GestorExpedientes .....	50
GestorSolicitudes .....	50
GestorUsuarios .....	52
PantallaError .....	52
PantallaExpedientes .....	52
PantallaOk .....	54
PantallaSolicitudes .....	54
PantallaUsuarios .....	54
Rol .....	56
Secretario [Actor].....	56
Solicitante [Actor].....	58
Solicitud .....	58

Gestión de expedientes para ayuntamientos.

Usuario Autenticado [Actor].....	62
UsuarioAutenticado .....	62
Conclusiones.....	64
Bibliografía.....	66
Anexo 1. Instalación de GRAILS.....	68
Anexo 2. Creación, compilación y despliegue del proyecto.....	70

Gestión de expedientes para ayuntamientos.

## Objetivo del TFC

El objetivo principal del TFC es el desarrollo de una solución sobre la arquitectura J2EE que cumpla con los requisitos recogidos. Para ello se utilizará el marco de trabajo GRAILS. Este marco de trabajo reúne diversas tecnologías del mundo JAVA en una sola solución facilitando su integración y uso. Algunas de estas tecnologías son HIBERNATE, SPRIN y JPA. También con GRAILS se puede utilizar un lenguaje derivado de JAVA y a medio caballo entre este y RUBY. Dicho lenguaje se llama GROOVY.

Por tanto, como objetivo añadido de este proyecto es el estudiar la aplicación del marco de trabajo GRAILS al desarrollo de aplicaciones empresariales. GRAILS es un nuevo marco de trabajo que ya ha alcanzado la madurez suficiente para ser usado en proyectos de cierta envergadura.

## Enfoque y método seguido

El enfoque seguido ha sido el de dividir todo el proyecto en diversas fases. Las fases seguidas han sido las habituales para estos desarrollos: Recogida de requisitos, análisis de estos requisitos, diseño, desarrollo, pruebas y despliegue.

No obstante, y debido al desconocimiento del marco de trabajo, ha habido momentos donde se ha tenido que volver a replantear algunas de las decisiones tomadas, pero por suerte han sido pocas (caso de las claves asignadas automáticamente que ya comentaré).

## Planificación

La planificación seguida ha sido la siguiente

- Especificación y análisis. 15 días
- Funcionalidades y casos de usos. 15 días
- Diseño. 14 días
- Implementación. 60 días
- Revisión y presentación final. 10 días

El trabajo ha tenido desajustes en la parte de desarrollo debido a los problemas de no disponer de un entorno debidamente configurado para acelerar la productividad, aunque este también era uno de los factores de estudio objetivo del proyecto.

Gestión de expedientes para ayuntamientos.

## Producto obtenido

El producto obtenido es una aplicación web que puede ser desplegada en cualquier contenedor de aplicaciones JAVA, sea o no "FULL-COMPLIANT" J2EE. Muestra como con muy poco código se consigue una aplicación que tiene gestión de usuarios, permisos, expedientes, etcétera. Incluso se ha incluido la funcionalidad de almacenar/recuperar ficheros en la base de datos.

Gracias al marco de trabajo escogido que al incluir la tecnología SPRING permite tener funcionalidades que de otra forma sólo están disponibles en entornos J2EE puros (inyección de BEANS, transacciones,...).

Otro punto interesante es que la aplicación no depende de un motor de base de datos en concreto. De hecho el producto resultante se distribuye con su propia motor independiente, sin necesidad de tener que instalar ningún servidor de base de datos. Lógicamente esto no es lo mismo que tener un servidor independiente dedicado, además del consumo excesivo de recursos que requiere la base de datos integrada. Para la gestión de los accesos a datos se utiliza la librería HIBERNATE la cual garantiza un acceso homogéneo a una gran mayoría de bases de datos sin tener que escribir código dedicado.

## Contenido específico de la memoria

En el resto de capítulos de la memoria se comentarán las fases de análisis, diseño e implementación por las que ha transcurrido el desarrollo de este TFC.



# Especificación del proyecto y análisis

## Descripción

El objetivo de esta herramienta es la gestión informatizada de todas las solicitudes realizadas a un ayuntamiento y su posterior tramitación.

Para este caso se ha escogido el flujo de trabajo más simple, que es el de realizar una solicitud al ayuntamiento para después pasar esta por las fases de revisión y conformidad de las formas y documentación aportada, y si es necesario varias veces, hasta que llegue a ser aceptada finalmente o bien rechazada. También se contempla la posibilidad de que el solicitante pueda retirar su solicitud.

Como se quiere que la herramienta esté lo más accesible posible para los solicitantes se ha considerado la necesidad de que esta tiene que estar con la máxima disponibilidad. Teniendo en cuenta este requisito se ha optado por una aplicación a la cual se pueda acceder vía web.

También uno de los requisitos es que los solicitantes puedan acceder a las demandas que realizan e incluso poder modificarlas si se lo quieren (caso de aportar más documentación bajo requerimiento) . Además las funcionalidades de los usuarios de la herramienta varían dependiendo del rol que tengan que desempeñar en la tramitación de los expedientes. Ante esto es necesario contar con un sistema de validación para los usuarios y su gestión.

Además también se prevee la necesidad de una gestión de expedientes puramente administrativa y totalmente transparente a los solicitantes ya que sólo será visible para el personal del ayuntamiento. El objetivo de estos expedientes es el archivar las solicitudes junto con todas las documentaciones aportadas por el usuario así como los documentos generados en la tramitación de la solicitud.

Por tanto la herramienta se compone de los siguientes subsistemas:

- Gestión de solicitudes
- Gestión de ficheros
- Gestión de expedientes
- Gestión de usuarios
- Mantenimientos

## Software

La aplicación para desarrollar será de tipo J2EE.

La tecnología a utilizar es el framework GRAILS. Esta tecnología es relativamente nueva, comparada con otras, pero en su corta etapa de vida ha conseguido una estabilidad y robustez envidiables, junto con una gran facilidad de cara a los desarrolladores, demostrando ser una opción más que adecuada para las tareas necesarias en este caso. Concretamente se ha utilizado la

Gestión de expedientes para ayuntamientos.

versión con instalador de GRAILS para Windows , la cuál se puede descargar desde la siguiente URL:

<http://grails.org/Download>

Como IDE se ha optado por ECLIPSE con el los plugins necesarios para desarrollar con GRAILS. Concretamente se han seguido los pasos relatados en la siguiente URL:

<http://www.grails.org/Eclipse+IDE+Integration>

No obstante, hay que comentar que este paso no ha sido tan trivial como el de la instalación del FRAMEWORK, dando contínuos problemas.

Gestión de expedientes para ayuntamientos.

## **Funcionalidades**

En general, las funcionalidades de la aplicación son de varios subsistemas de tipo CRUD (Create, Read, Update, Delete) unidos, pero con algunas características añadidas.

### **Gestión de solicitudes**

#### *Alta de solicitud*

Inicia todo el proceso de tramitación administrativa. Sólo pueden dar de alta solicitudes aquellos usuarios que hayan sido validados.

#### *Modificar solicitud*

Un usuario puede realizar modificaciones a una solicitud realizada siempre y cuando así sea requerido previamente por la administración.

#### *Retirar solicitud*

Un usuario puede retirar una solicitud, si así lo cree oportuno en el momento en que le solicitan más documentación a aportar.

#### *Listar solicitudes*

Realización de listados por pantalla de las solicitudes del usuario. No se contemplan, a priori ningún tipo de filtrado.

#### *Buscar solicitud*

Herramienta complementaria de la anterior.

Gestión de expedientes para ayuntamientos.

## **Gestión de expedientes**

### *Alta de expediente*

Acción para crear el expediente que contendrá la solicitud y su documentación, tanto la aportada por el solicitante como la generada por la administración en la tramitación de la solicitud.

### *Versionar expediente*

Cada vez que el solicitante realiza modificaciones a la solicitud se crea una nueva versión para tener una copia histórica de la evolución de la documentación aportada.

### *Cerrar expediente*

Cerrar un expediente implica que a partir de ese momento ya no se puede realizar ninguna operación sobre él. A todos los efectos es como si ya no existiera quedando como historial de lo acontecido.

### *Listar expedientes*

Realización de listados por pantalla de los expedientes. Sólo se listan las versiones finales de los expedientes.

### *Buscar expediente*

Herramienta de búsqueda.

Gestión de expedientes para ayuntamientos.

## **Gestión de usuarios**

La gestión de usuarios merece una atención especial. Es la piedra angular del sistema. Por rol se entiende una agrupación de permisos específicos que otorgan diferentes capacidades a los usuarios de la herramienta. En este caso se ha detectado principalmente tres tipos de roles, pero se deja abierta la posibilidad para que los usuarios puedan tener más de un rol a la vez.

### *Alta de usuario*

Cuando es dado de alta un usuario es necesario asignarle al menos un rol válido. Los roles ya vienen definidos en la herramienta siendo por ahora tres.

### *Modificar usuario*

Modificación de los datos del usuario así como de sus atributos (roles).

### *Dar de baja usuario*

Dar de baja un usuario del sistema. La baja es lógica y no se puede reactivar.

### *Listar usuarios*

Realización de listados por pantalla de los usuarios del sistema. Se incluye un resumen de las solicitudes en los diferentes estados.

### *Buscar usuario*

Búsqueda de usuario

### *Mantenimientos*

Por determinar

Gestión de expedientes para ayuntamientos.

## **Gestión de solicitudes**

### *Alta de solicitud*

Inicia todo el proceso de tramitación administrativa. Sólo pueden dar de alta solicitudes aquellos usuarios que hayan sido validados.

### *Modificar solicitud*

Un usuario puede realizar modificaciones a una solicitud realizada siempre y cuando así sea requerido previamente por la administración.

### *Retirar solicitud*

Un usuario puede retirar una solicitud, si así lo cree oportuno en el momento en que le solicitan más documentación a aportar.

### *Listar solicitudes*

Realización de listados por pantalla de las solicitudes del usuario. No se contemplan, a priori ningún tipo de filtrado.

### *Buscar solicitud*

Herramienta complementaria de la anterior.

Gestión de expedientes para ayuntamientos.

## **Gestión de expedientes**

### *Alta de expediente*

Acción para crear el expediente que contendrá la solicitud y su documentación, tanto la aportada por el solicitante como la generada por la administración en la tramitación de la solicitud.

### *Versionar expediente*

Cada vez que el solicitante realiza modificaciones a la solicitud se crea una nueva versión para tener una copia histórica de la evolución de la documentación aportada.

### *Cerrar expediente*

Cerrar un expediente implica que a partir de ese momento ya no se puede realizar ninguna operación sobre él. A todos los efectos es como si ya no existiera quedando como historial de lo acontecido.

### *Listar expedientes*

Realización de listados por pantalla de los expedientes. Sólo se listan las versiones finales de los expedientes.

### *Buscar expediente*

Herramienta de búsqueda.

Gestión de expedientes para ayuntamientos.

## **Gestión de usuarios**

La gestión de usuarios merece una atención especial. Es la piedra angular del sistema. Por rol se entiende una agrupación de permisos específicos que otorgan diferentes capacidades a los usuarios de la herramienta. En este caso se ha detectado principalmente tres tipos de roles, pero se deja abierta la posibilidad para que los usuarios puedan tener más de un rol a la vez.

### *Alta de usuario*

Cuando es dado de alta un usuario es necesario asignarle al menos un rol válido. Los roles ya vienen definidos en la herramienta siendo por ahora tres.

### *Modificar usuario*

Modificación de los datos del usuario así como de sus atributos (roles).

### *Dar de baja usuario*

Dar de baja un usuario del sistema. La baja es lógica y no se puede reactivar.

### *Listar usuarios*

Realización de listados por pantalla de los usuarios del sistema. Se incluye un resumen de las solicitudes en los diferentes estados.

### *Buscar usuario*

Búsqueda de usuario

### *Mantenimientos*

Por determinar



Gestión de expedientes para ayuntamientos.

## Lista de paquetes

Nombre completo	Descripción
Solicitud::edu::uoc::tfc	Paquete que contiene las clases del proyecto.

## Lista de Casos de Uso

Nombre	Descripción
abrirExpediente	Acción que realiza la apertura de un expediente.
aceptarEnTramite	Aceptar en trámite significa que la solicitud ha sido presentada con toda la documentación necesaria y que cumple los requisitos administrativos.
aceptarFinal	Implica la aceptación definitiva de la solicitud.
altaSolicitud	Inicia el trámite de la solicitud.
bajaUsuario	Realiza la baja lógica de un usuario en el sistema. No se contemplan bajas físicas.
cerrarExpediente	Cerrar el expediente implica que no se puede modificar en ningún aspecto (no se puede editar ni aportar más documentación).
consultarExpedientes	Acción para consultar los expedientes
consultarSolicitudes	Acción para consultar las solicitudes
consultarUsuarios	Acción para consultar los usuarios
crearUsuario	Crea un usuario en el sistema.
modificarExpediente	Modifica un expediente. Normalmente implica modificar la documentación de la solicitud.
modificarSolicitud	Permite modificar los atributos de la solicitud así como la documentación aportada al solicitante de esta.
modificarUsuario	Modifica los atributos de un usuario
rechazarSolicitud	Rechaza la solicitud por diversos motivos. Es necesario especificar el motivo del rechazo.
requerirMasDocumentacion	Requiere que el solicitante modifique la solicitud, bien corrigiendo los datos de esta, bien aportando más documentación
retirarSolicitud	Permite al solicitante retirar una solicitud realizada por él.

## Lista de Classificadores

Nombre	Tipo	Descripción
Administrador	Actor	Usuario autenticado encargado del mantenimiento de los usuarios así como de sus roles.
Alcalde	Actor	Usuario autenticado asignado a realizar las tareas de aprobación o rechazo de las solicitudes realizadas por los usuarios.
DAO	Class	Clase que permite acceder a las funcionalidades de persistencia del servidor.
Expediente	Class	Un expediente contiene la solicitud que lo inicia, su documentación, así como los atributos necesarios para saber en que estado se encuentra el trámite así como las firmas y notas necesarias.
Fichero	Class	Clase para el almacenamiento de los ficheros de la

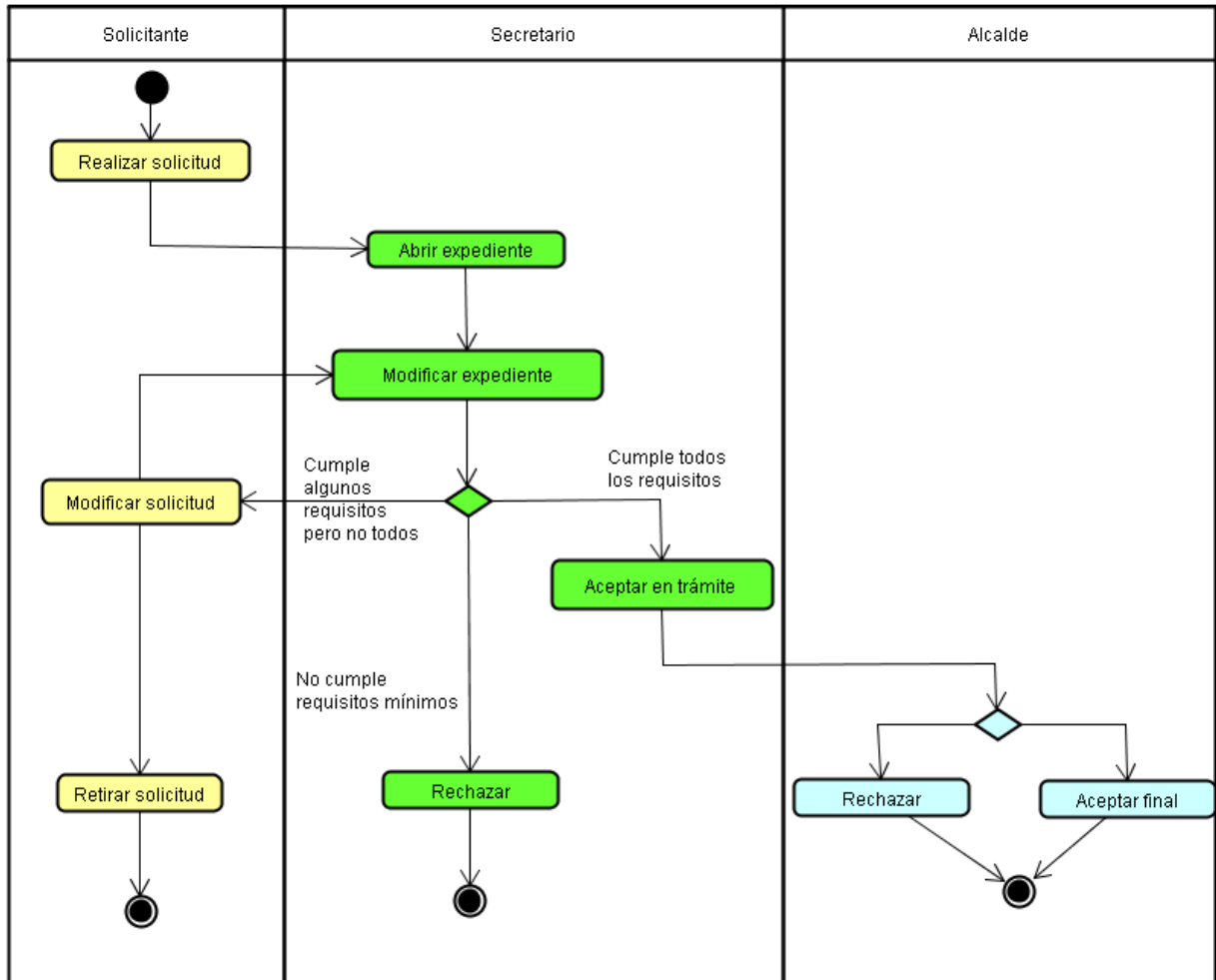
Gestión de expedientes para ayuntamientos.

		solicitud
GestorExpedientes	Class	Gestor de negocio de los expedientes.
GestorSolicitudes	Class	Gestor de negocio de las solicitudes
GestorUsuarios	Class	Gestor de negocio de los usuarios
PantallaError	Class	Pantalla de mensaje de error
PantallaExpedientes	Class	Pantalla de gestión de expedientes
PantallaOk	Class	Pantalla de mensaje de que todo ha ido bien
PantallaSolicitudes	Class	Pantalla de gestión de solicitudes
PantallaUsuarios	Class	Pantalla de gestión de los usuarios
Rol	Class	Rol de un usuario
Secretario	Actor	Usuario autenticado asignado a realizar las tareas que conllevan la gestión de las solicitudes realizadas por los usuarios.
Solicitante	Actor	<p>Usuario que inicia el trámite de una solicitud. Puede ser autenticado o externo.</p> <p>Se entiende por usuario externo aquel que no necesita autenticarse para realizar el trámite. No obstante, y por necesidad de la aplicación, a cada solicitud se le vincula con el NIF/CIF del solicitante, y con un identificador único alfanumérico de la solicitud.</p>
Solicitud	Class	Solicitud realizada por un usuario
Usuario Autenticado	Actor	Usuario que ha sido autenticado en el sistema. Un usuario autenticado tiene acceso a los recursos dentro de la intranet dentro de los roles que le han sido concedidos.
UsuarioAutenticado	Class	Usuario de la red interna del ayuntamiento

Gestión de expedientes para ayuntamientos.

## Diagramas

*Diagrama de actividad*



## Definición

Diagrama de actividad de la solicitud

## Lista de elementos

Nombre	Tipo	Descripción
Alcalde	Partición	Usuario autenticado asignado a realizar las tareas de aprobación o rechazo de las solicitudes realizadas por los usuarios.
Secretario	Partición	Usuario autenticado asignado a realizar las tareas que conllevan la gestión de las solicitudes realizadas por los usuarios.
Solicitante	Partición	Usuario que inicia el trámite de una solicitud. Puede ser autenticado o externo.  Se entiende por usuario externo aquel que no necesita autenticarse para realizar el trámite. No obstante, y por necesidad de la aplicación, a cada solicitud se le vincula con el NIF/CIF del solicitante, y con un identificador único alfanumérico de la solicitud.
Abrir expediente	Action	Acción que realiza la apertura de un expediente. Un expediente contiene la solicitud que lo inicia, su documentación, así como los atributos necesarios para saber en que estado se encuentra el trámite así como las firmas y notas necesarias.
Aceptar en trámite	Action	Establecer un expediente como aceptado en trámite
Aceptar final	Action	
Modificar expediente	Action	Modifica un expediente. Normalmente implica modificar la documentación de la solicitud.
Modificar solicitud	Action	Permite modificar los atributos de la solicitud así como la documentación aportada al solicitante de esta.
Realizar solicitud	Action	Inicia el trámite de la solicitud.
Rechazar	Action	Rechazar un expediente. Equivale a rechazar la solicitud.
Rechazar	Action	Rechazar un expediente. Equivale a rechazar la solicitud.
Retirar solicitud	Action	Permite al solicitante retirar una solicitud realizada por él.

## Partición

▮ Alcalde [Partición]

### Definición

Usuario autenticado asignado a realizar las tareas de aprobación o rechazo de las solicitudes realizadas por los usuarios.

▮ Secretario [Partición]

### Definición

Usuario autenticado asignado a realizar las tareas que conllevan la gestión de las solicitudes realizadas por los usuarios.



## ¶ Solicitante [Partición]

### Definición

Usuario que inicia el trámite de una solicitud. Puede ser autenticado o externo.

Se entiende por usuario externo aquel que no necesita autenticarse para realizar el trámite. No obstante, y por necesidad de la aplicación, a cada solicitud se le vincula con el NIF/CIF del solicitante, y con un identificador único alfanumérico de la solicitud.

### ActionState

#### Abrir expediente [Action]

##### Definición

Acción que realiza la apertura de un expediente. Un expediente contiene la solicitud que lo inicia, su documentación, así como los atributos necesarios para saber en que estado se encuentra el trámite así como las firmas y notas necesarias.

#### Aceptar en trámite [Action]

##### Definición

Establecer un expediente como aceptado en trámite.

#### Aceptar final [Action]

##### Definición

Establecer un expediente como definitivamente aceptado.

#### Modificar expediente [Action]

##### Definición

Modifica un expediente. Normalmente implica modificar la documentación de la solicitud.

#### Modificar solicitud [Action]

##### Definición

Permite modificar los atributos de la solicitud así como la documentación aportada al solicitante de esta.

#### Realizar solicitud [Action]

##### Definición

Inicia el trámite de la solicitud.

#### Rechazar [Action]

##### Definición



Rechazar un expediente. Equivale a rechazar la solicitud.

Rechazar [Action]

#### Definición

Rechazar un expediente. Equivale a rechazar la solicitud.

Retirar solicitud [Action]

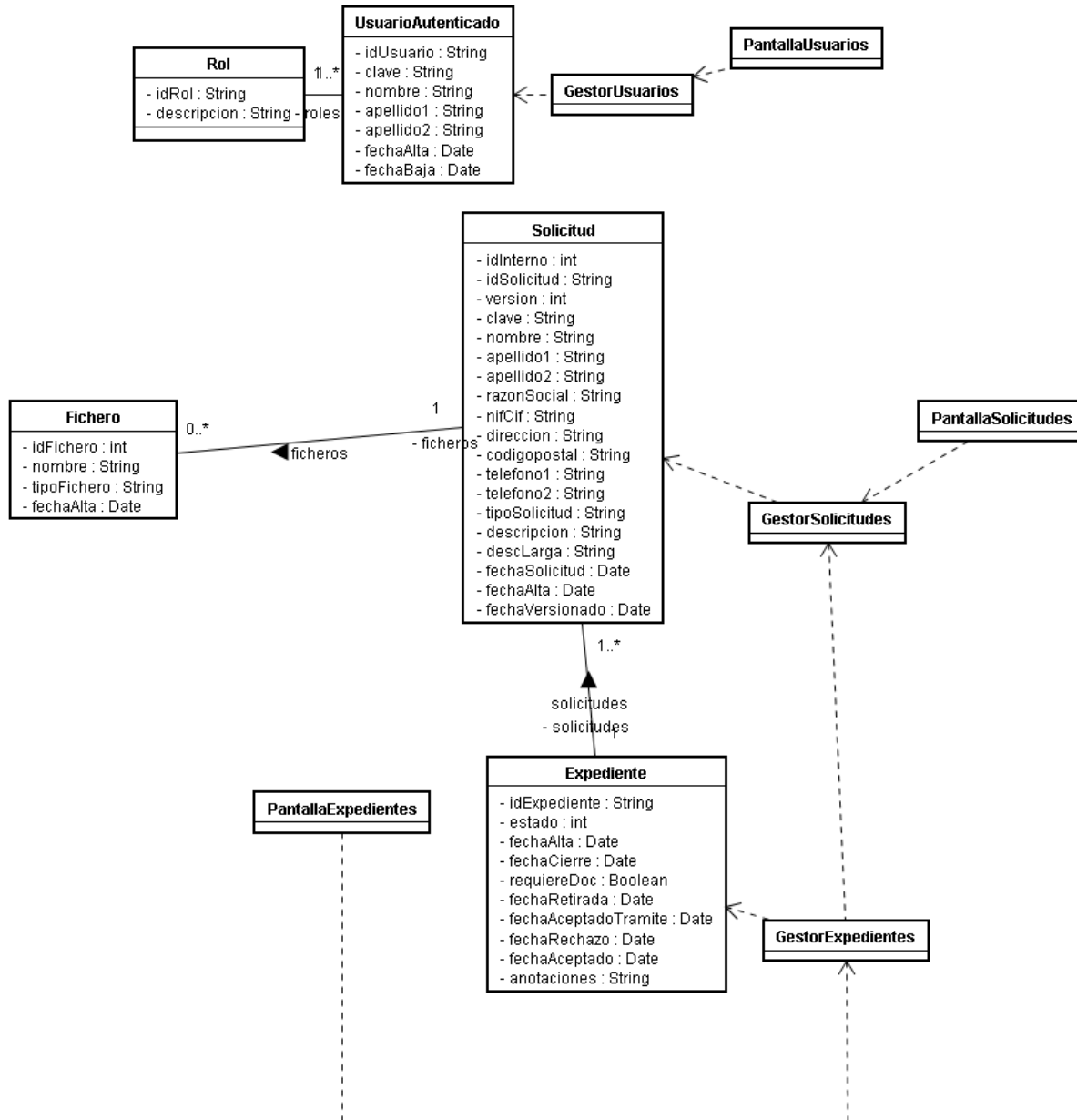
#### Definición

Permite al solicitante retirar una solicitud realizada por él.



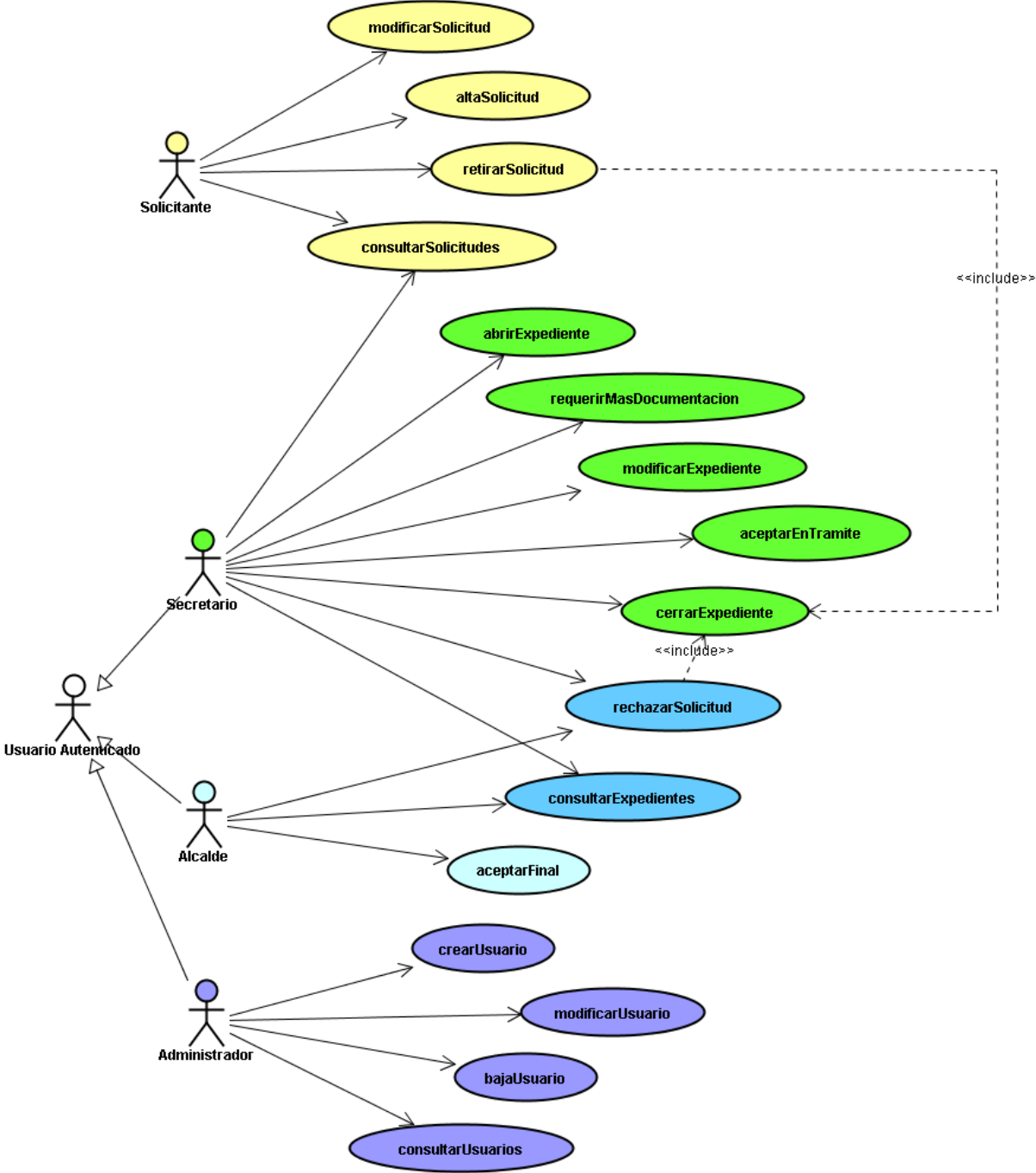


## Diagrama de clases



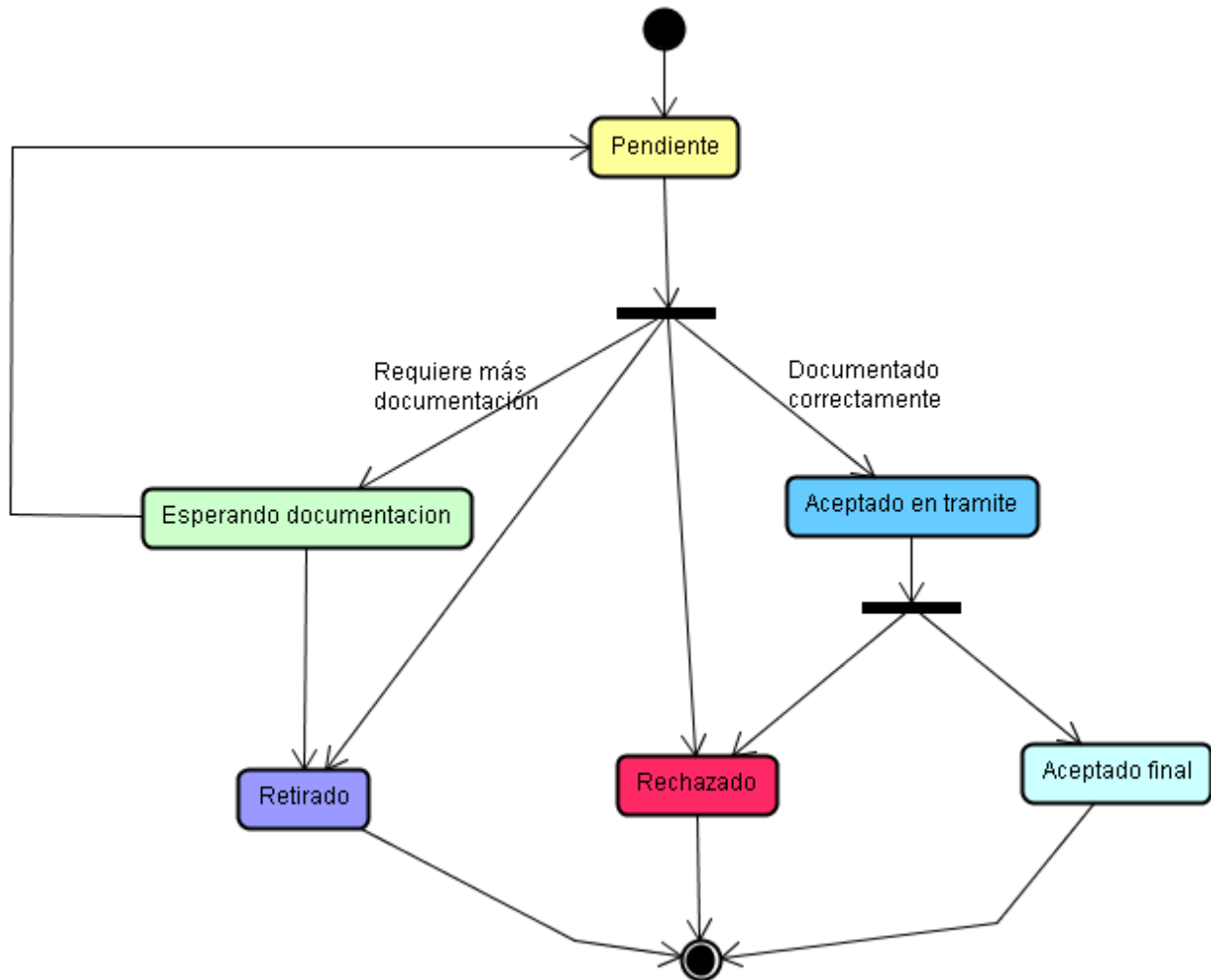


Casos de uso





## Diagrama de estados

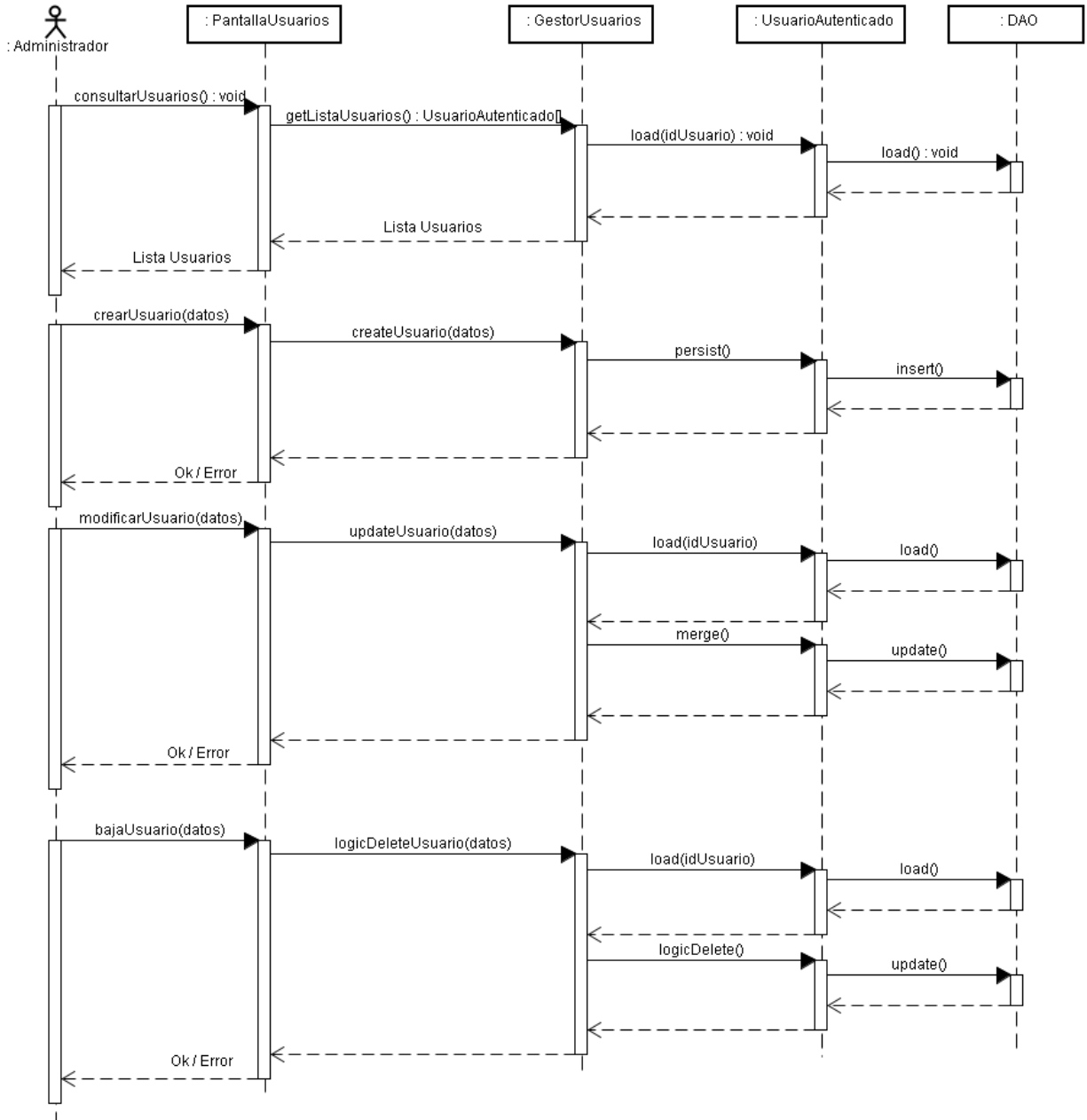


### **Definición**

En este diagrama se visualizan los diferentes estados que puede alcanzar una solicitud.



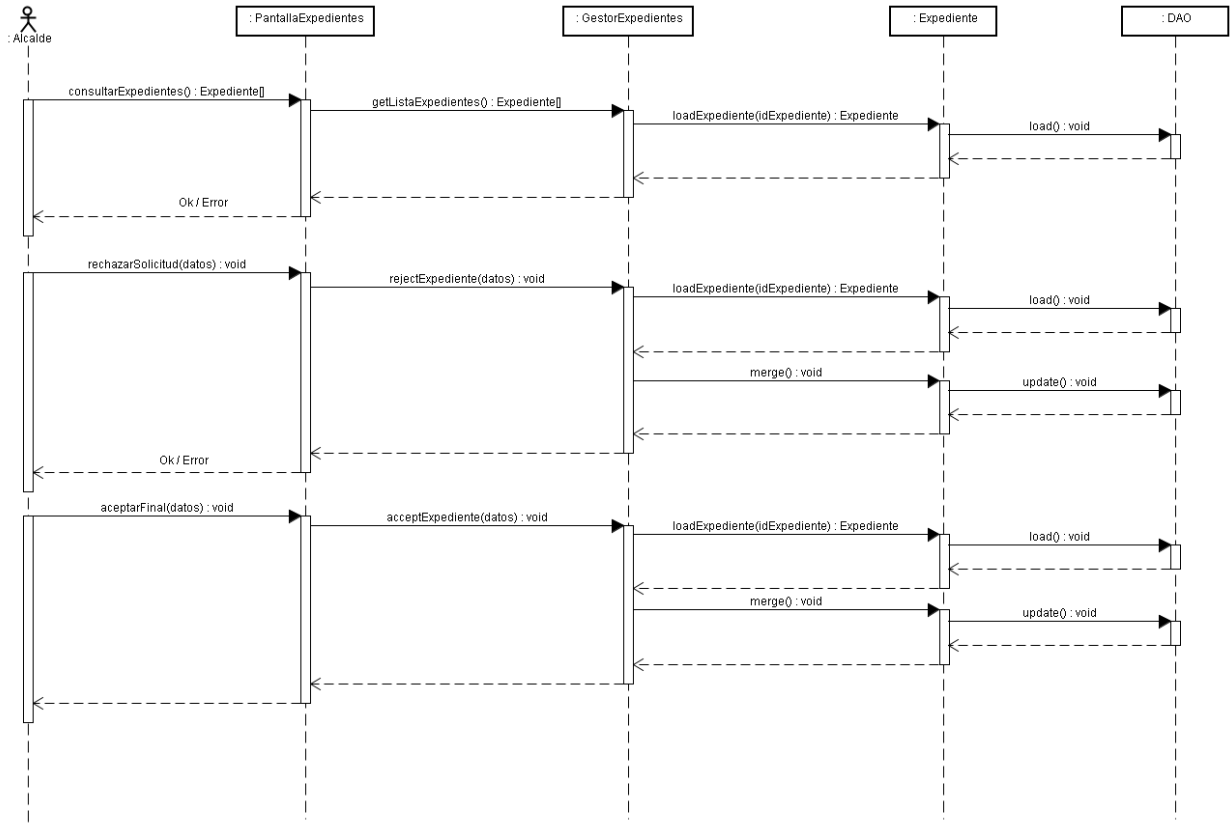
## Diagrama de secuencias - Administrador





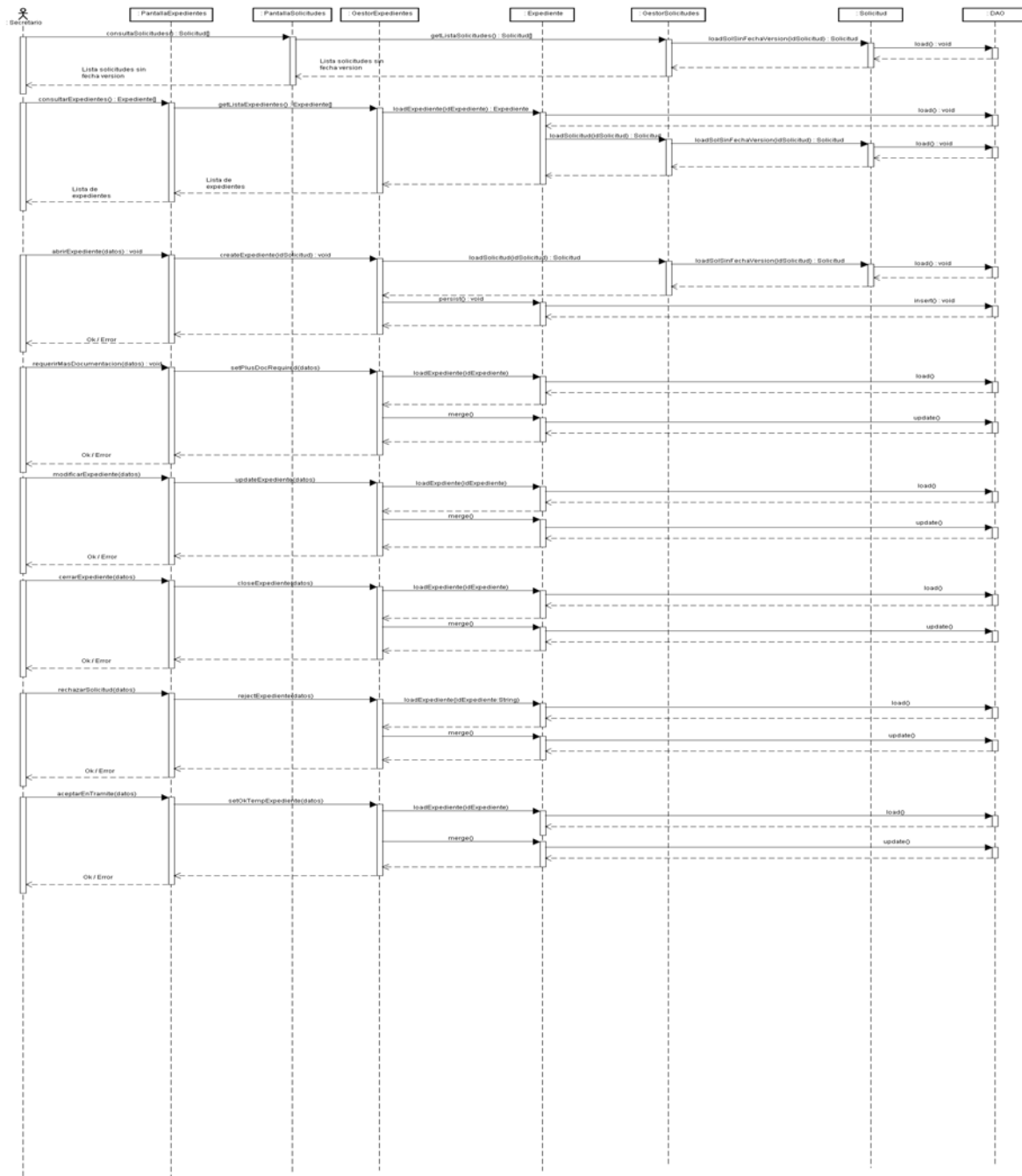


## Diagrama de secuencias - Alcalde



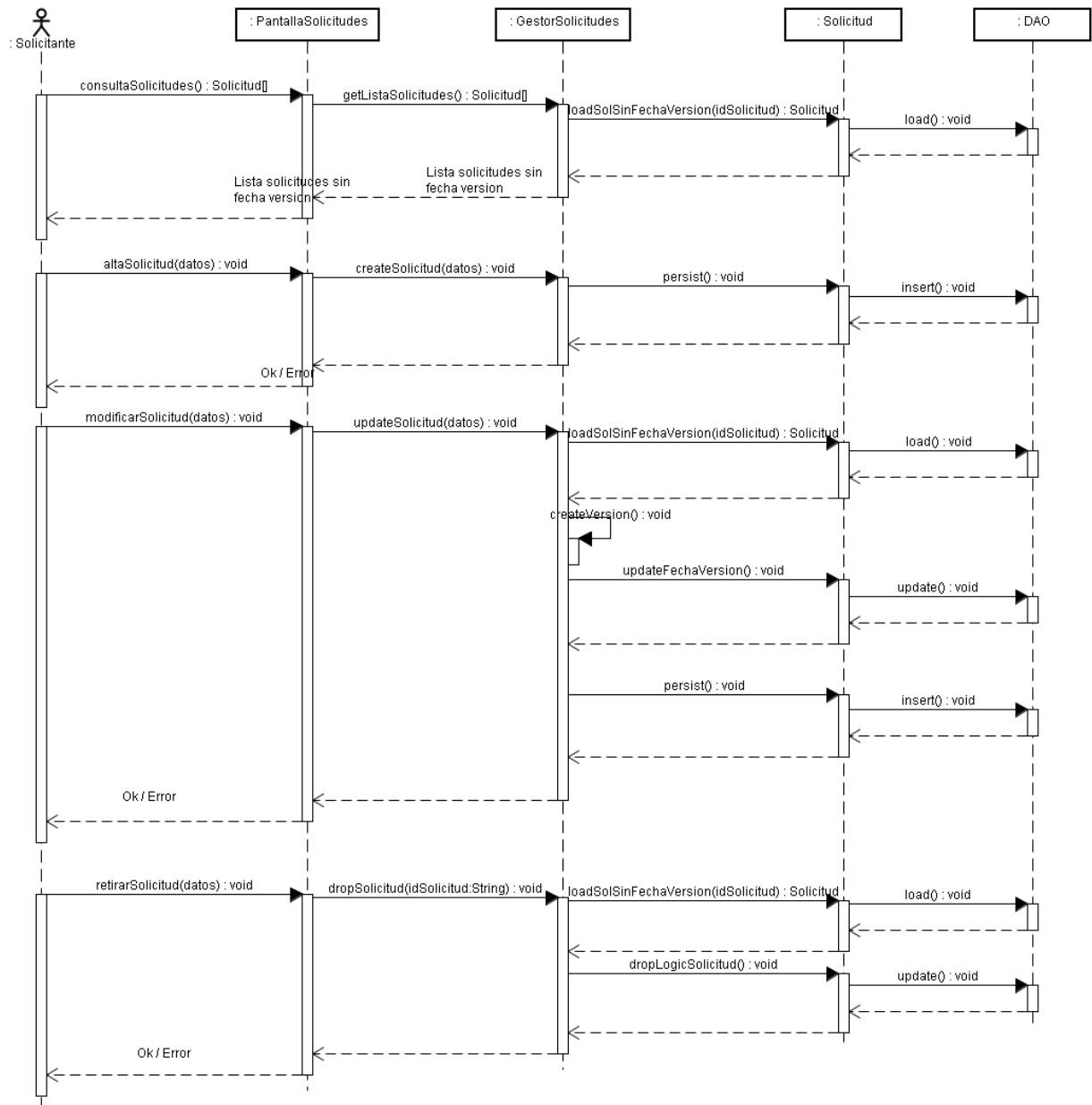


# Diagrama de secuencias - Secretario





## Diagrama de secuencias - Solicitante





## Diagrama de actividad - Casos de uso

### **abrirExpediente**

Caso de uso	abrirExpediente
Descripción	Acción que realiza la apertura de un expediente.
Actor	Secretario
Precondition	Tiene que existir la solicitud
Postcondition	Solicitud vinculada a un expediente

### **aceptarEnTramite**

Caso de uso	aceptarEnTramite
Descripción	Aceptar en trámite significa que la solicitud ha sido presentada con toda la documentación necesaria y que cumple los requisitos administrativos.
Actor	Secretario
Precondition	El expediente está pendiente y si fuera necesario con la documentación correcta
Postcondition	El expediente queda en aceptado en trámite

### **aceptarFinal**

Caso de uso	aceptarFinal
Descripción	Implica la aceptación definitiva de la solicitud.
Actor	Alcalde
Precondition	El expediente está en el estado aceptado en trámite
Postcondition	El expediente queda cerrado con estado aceptado final.

### **altaSolicitud**

Caso de uso	altaSolicitud
Descripción	Inicia el trámite de la solicitud
Actor	Solicitante
Precondition	
Postcondition	

### **bajaUsuario**

Caso de uso	bajaUsuario
Descripción	Reliza la baja lógica de un usuario en el sistema. No se contemplan bajas físicas.
Actor	Administrador
Precondition	El usuario existe
Postcondition	El usuario queda de baja
Nota	No se contempla la reactivación de usuarios.





## **cerrarExpediente**

### **Relacionados (Desde destino a origen)**

Nombre	Tipo	Origen	Descripción
	Include	rechazarSolicitud	
	Include	retirarSolicitud	

Caso de uso	cerrarExpediente
Descripción	Cierra el expediente. Cerrar el expediente implica que no se puede modificar en ningún aspecto (no se puede editar ni aportar más documentación).
Actor	Secretario
Precondition	El expediente está abierto
Postcondition	El expediente está cerrado

## **consultarExpedientes**

Caso de uso	consultarExpedientes
Descripción	Acción para consultar los expedientes
Actor	Secretario Alcalde

## **consultarSolicitudes**

Caso de uso	consultarSolicitudes
Descripción	Acción para consultar las solicitudes.
Actor	Solicitante Secretario

## **consultarUsuarios**

Caso de uso	consultarUsuarios
Descripción	Acción para consultar los usuarios
Actor	Administrador

## **crearUsuario**

Caso de uso	crearUsuario
Descripción	Acción para crear un usuario autenticado.
Actor	Administrador
Precondition	El usuario no existe
Postcondition	El usuario es registrado correctamente en el sistema
Note	No se contempla la reactivación de usuarios

## **modificarExpediente**

Caso de uso	modificarExpediente
Descripción	Modifica un expediente.
Actor	Secretario
Precondition	El expediente existe y está abierto
Postcondition	El expediente tiene nuevos valores para sus atributos.



### **modificarSolicitud**

Caso de uso	modificarSolicitud
Descripción	Permite modificar los atributos de la solicitud así como la documentación aportada al solicitante de esta.
Actor	Solicitante
Precondition	La solicitud existe
Postcondition	La solicitud es modificada

### **modificarUsuario**

Caso de uso	modificarUsuario
Descripción	Modifica los atributos de un usuario autenticado
Actor	Administrador
Precondition	El usuario existe
Postcondition	El usuario es modificado

### **rechazarSolicitud**

Caso de uso	rechazarSolicitud
Descripción	Rechaza la solicitud
Actor	Secretario Alcalde
Precondition	La solicitud está pendiente de tramitar
Postcondition	La solicitud queda rechazada. Su expediente es cerrado.
Sub caso de uso	cerrarExpediente

### **requerirMasDocumentacion**

Caso de uso	requerirMasDocumentacion
Descripción	Requiere que el solicitante modifique la solicitud, bien corrigiendo los datos de esta, bien aportando más documentación
Actor	Secretario
Precondition	La solicitud existe y está pendiente
Postcondition	La solicitud queda como pendiente de aportar más documentación

### **retirarSolicitud**

Caso de uso	retirarSolicitud
Descripción	Permite al solicitante retirar una solicitud que esté pendiente o en espera de más documentación.
Actor	Solicitante
Precondition	La solicitud existe y está pendiente o en espera de aportar más documentación
Postcondition	La solicitud queda retirada
Sub Caso de uso	cerrarExpediente



## Clases

### **Administrador [Actor]**

#### **Declaración**

```
public class Administrador
extends Usuario Autenticado
```

#### **Estereotipo**

```
<<actor>>
```

#### **Definición**

Usuario autenticado encargado del mantenimiento de los usuarios así como de sus roles.

#### **Relaciones**

Nombre	Tipo	Destino	Descripción
	Asociación	crearUsuario	
	Asociación	modificarUsuario	
	Asociación	bajaUsuario	
	Asociación	consultarUsuarios	
	Generalización	Usuario Autenticado	

### **Alcalde [Actor]**

#### **Declaración**

```
public class Alcalde
extends Usuario Autenticado
```

#### **Estereotipo**

```
<<actor>>
```

#### **Definición**

Usuario autenticado asignado a realizar las tareas de aprobación o rechazo de las solicitudes realizadas por los usuarios.

#### **Relaciones**

Nombre	Tipo	Destino	Descripción
	Asociación	rechazarSolicitud	
	Asociación	aceptarFinal	
	Asociación	consultarExpedientes	
	Generalización	Usuario Autenticado	

### **DAO**

#### **Declaración**

```
public class DAO
```



## Definición

Clase que permite acceder a las funcionalidades de persistencia del servidor.

## Operation

Nombre	Descripción
load	Carga un objeto desde un gestor de persistencia
update	Actualiza la representación del objeto en un gestor de persistencia con los atributos actuales
insert	Da de alta un nuevo objeto en el gestor de persistencia

## Expediente

### Declaración

```
public class Expediente
```

### Definición

Un expediente contiene la solicitud que lo inicia, su documentación, así como los atributos necesarios para saber en que estado se encuentra el trámite así como las firmas y notas necesarias.

### Attribute

Nombre	Tipo	Descripción
idExpediente	private String	Identificador del expediente
estado	private int	Identificador de estado
fechaAlta	private Date	Fecha de apertura del expediente
fechaCierre	private Date	Fecha de cierre del expediente
requiereDoc	private Boolean	Identificador de requerimiento de documentación
fechaRetirada	private Date	Fecha de retirada de la solicitud
fechaAceptadoTramite	private Date	Fecha de aceptación en trámite de la solicitud.
fechaRechazo	private Date	Fecha de rechazo de la solicitud. Se informa al cerrar el expediente.
fechaAceptado	private Date	Fecha de aceptación final de la solicitud.
anotaciones	private String	Texto libre

### Operation

Nombre	Descripción
loadExpediente	Método para cargar un expediente
merge	Método para actualizar los datos del expediente en el gestor
persist	Método para dar de alta un expediente en el gestor

### Relation (From Source To Destino)

Nombre	Tipo	Destino	Descripción
solicitudes	Asociación	Solicitud [1..*]	Array de solicitudes del expediente





## **Fichero**

### **Declaración**

```
public class Fichero
```

### **Definición**

Clase para el almacenamiento de los ficheros de la solicitud

### **Attribute**

Nombre	Tipo	Descripción
idFichero	private int	Identificador de fichero
nombre	private String	Nombre del fichero
tipoFichero	private String	Tipo de fichero
fechaAlta	private Date	Fecha de alta

## **GestorExpedientes**

### **Declaración**

```
public class GestorExpedientes
```

### **Definición**

Gestor de negocio de los expedientes.

### **Operation**

Nombre	Descripción
getListaExpedientes	Método para recuperar una lista de expedientes
rejectExpediente	Método para rechazar un expediente
acceptExpediente	Método para aceptar finalmente un expediente
createExpediente	Método para crear un expediente
setPlusDocRequired	Método para marcar un expediente que requiere documentación
updateExpediente	Método para actualizar los datos de un expediente
closeExpediente	Método para cerrar un expediente
setOkTempExpediente	Método para aceptar en trámite un expediente

### **Relaciones**

Nombre	Tipo	Destino	Descripción
	Dependencia	Expediente	
	Dependencia	GestorSolicitudes	

## **GestorSolicitudes**

### **Declaración**

```
public class GestorSolicitudes
```

### **Definición**

Gestor de negocio de las solicitudes



## Operation

Nombre	Descripción
getListaSolicitudes	Recupera una lista de solicitudes
loadSolicitud	Carga una solicitud
createSolicitud	Crea una solicitud
updateSolicitud	Actualiza un solicitud en el gestor
createVersion	Crea una nueva versión de la solicitud. A la versión anterior le establece la fecha de versión del momento
dropSolicitud	Retira una solicitud

## Relaciones

Nombre	Tipo	Destino	Descripción
	Dependencia	Solicitud	

## **GestorUsuarios**

### Declaración

```
public class GestorUsuarios
```

### Definición

Gestor de negocio de los usuarios

## Operation

Nombre	Descripción
getListaUsuarios	Recupera la lista de usuarios
createUsuario	Crea un usuario con autenticación
updateUsuario	Actualiza un usuario con autenticación
logicDeleteUsuario	Establece como inactivo un usuario.No lo borra físicamente

## Relacionados

Nombre	Tipo	Destino	Descripción
	Dependencia	UsuarioAutenticado	

## **PantallaError**

### Declaración

```
public class PantallaError
```

### Definición

Pantalla de mensaje de error

## **PantallaExpedientes**

### Declaración

```
public class PantallaExpedientes
```

### Definición

Pantalla de gestión de expedientes



## Operation

Nombre	Descripción
requerirMasDocumentacion	Marca al expediente como que requiere más información
consultarExpedientes	Recupera la lista de expedientes
rechazarSolicitud	Rechaza el expediente. Implica el rechazo de la solicitud
aceptarFinal	Acepta definitivamente el expediente. Implica la aceptación de la última versión de la solicitud.
abrirExpediente	Crea un expediente a partir de una solicitud
modificarExpediente	Modifica un expediente
cerrarExpediente	Cierra el expediente. Implica que no se pueden realizar modificaciones de ningún tipo ni al expediente ni a la solicitud
aceptarEnTramite	Acepta en trámite el expediente

## Relacionados

Nombre	Tipo	Destino	Descripción
	Dependencia	GestorExpedientes	

## **PantallaOk**

### Declaración

```
public class PantallaOk
```

### Definición

Pantalla de mensaje de que todo ha ido bien

## **PantallaSolicitudes**

### Declaración

```
public class PantallaSolicitudes
```

### Definición

Pantalla de gestión de solicitudes

## Operation

Nombre	Descripción
consultaSolicitudes	Recupera la lista de solicitudes
altaSolicitud	Crea una solicitud
modificarSolicitud	Modifica una solicitud
retirarSolicitud	Retira una solicitud

## Relacionados

Nombre	Tipo	Destino	Descripción
	Dependencia	GestorSolicitudes	

## **PantallaUsuarios**

### Declaración

```
public class PantallaUsuarios
```



## Definición

Pantalla de gestión de los usuarios

## Operation

Nombre	Descripción
consultarUsuarios	Recupera la lista de usuarios
crearUsuario	Crea un usuario
modificarUsuario	Modifica un usuario
bajaUsuario	Baja lógica de un usuario

## Relacionados

Nombre	Tipo	Destino	Descripción
	Dependencia	GestorUsuarios	

## Rol

### Declaración

```
public class Rol
```

### Definición

Rol de un usuario

### Attribute

Nombre	Descripción
idRol	Identificador de rol
descripcion	Descripción

## **Secretario [Actor]**

### Declaración

```
public class Secretario  
extends Usuario Autenticado
```

### Estereotipo

```
<<actor>>
```

### Definición

Usuario autenticado asignado a realizar las tareas que conllevan la gestión de las solicitudes realizadas por los usuarios.

### Relation (From Source To Destino)

Nombre	Tipo	Destino	Descripción
	Asociación	aceptarEnTramite	
	Asociación	rechazarSolicitud	
	Asociación	requerirMasDocumentacion	
	Asociación	abrirExpediente	





	Asociación	modificarExpediente	
	Asociación	cerrarExpediente	
	Asociación	consultarExpedientes	
	Asociación	consultarSolicitudes	
	Generalización	Usuario Autenticado	

## **Solicitante [Actor]**

### **Declaración**

public class Solicitante

### **Estereotipo**

<<actor>>

### **Definición**

Usuario que inicia el trámite de una solicitud. Puede ser autenticado o externo.

Se entiende por usuario externo aquel que no necesita autenticarse para realizar el trámite. No obstante, y por necesidad de la aplicación, a cada solicitud se le vincula con el NIF/CIF del solicitante, y con un identificador único alfanumérico de la solicitud.

### **Relation (From Source To Destino)**

Nombre	Tipo	Destino	Descripción
	Asociación	altaSolicitud	
	Asociación	retirarSolicitud	
	Asociación	modificarSolicitud	
	Asociación	consultarSolicitudes	

## **Solicitud**

### **Declaración**

public class Solicitud

### **Definición**

Solicitud realizada por un usuario

### **Attribute**

Nombre	Descripción
idInterno	Identificador interno de una solicitud
idSolicitud	Identificador público de una solicitud
version	Versión
clave	Clave asignada a la solicitud
nombre	Nombre del solicitante
apellido1	Primer apellido del solicitante
apellido2	Segundo apellido del solicitante
razonSocial	Razón social
nifCif	Nif o Cif
direccion	Dirección del solicitante
codigopostal	Código postal



telefono1	Teléfono 1
telefono2	Teléfono 2
tipoSolicitud	Tipo de solicitud
descripcion	Descripción corta
descLarga	Descripción larga
fechaSolicitud	Fecha de la solicitud
fechaAlta	Fecha de alta en el sistema
fechaVersionado	Fecha de versionado

### Operation

Nombre	Descripción
loadSolSinFechaVersion	Carga la solicitud de identificador idSolicitud con fecha versionado no establecida
persist	Da de alta la solicitud
updateFechaVersion	Establece la fecha versionado
dropLogicSolicitud	Da de baja la solicitud

### Relacionados

Nombre	Tipo	Destino	Descripción
ficheros	Asociación	Fichero [0..*]	



## **Usuario Autenticado [Actor]**

### **Declaración**

public class Usuario Autenticado

### **Estereotipo**

<<actor>>

### **Definición**

Usuario que ha sido autenticado en el sistema. Un usuario autenticado tiene acceso a los recursos dentro de la intranet dentro de los roles que le han sido concedidos.

## **UsuarioAutenticado**

### **Declaración**

public class UsuarioAutenticado

### **Definición**

Usuario de la red interna del ayuntamiento

### **Attribute**

Nombre	Descripción
idUsuario	Identificador del usuario
clave	Clave del usuario.
nombre	Nombre del usuario
apellido1	Primer apellido del usuario
apellido2	Segundo apellido del usuario
fechaAlta	Fecha de alta del usuario
fechaBaja	Fecha de baja del usuario

### **Operation**

Nombre	Descripción
load	Carga un usuario
persist	Da de alta un usuario
merge	Actualiza un usuario
logicDelete	Da de baja un usuario

### **Relacionados**

Nombre	Tipo	Destino	Descripción
	Asociación	Rol [1..*]	



## Conclusiones

Con la realización de este proyecto lo que se buscaba era además del desarrollo de una aplicación J2EE estudiar el uso de GRAILS para su desarrollo. En concreto la versión 1.0.2

En resumen el enfoque empleado en este caso ha demostrado no ser el más acertado para desarrollar con este FRAMEWORK.

GRAILS gracias a la técnica de SCAFFOLDING simplifica al máximo la creación de interfaces de cara al usuario. Por si esto no fuera suficiente además permite generar las vistas y formularios de nuestra aplicación WEB a partir de nuestro modelo de datos.

Gracias al soporte que incluye para JPA, HIBERNATE, SPRING y GROOVY la definición e implementación de las entidades del modelo de datos y su persistencia queda reducida al mínimo. Si optamos por la combinación JPA sólo tendremos que definir nuestra clase y añadir las anotaciones oportunas. Si optamos por GROOVY no necesitaremos apoyarnos en anotaciones y además tendremos la ventaja de poder realizar modificaciones incluso con la aplicación arrancada. Tanto al usar JPA como GROOVY el acceso a base de datos es realizado a través de HIBERNATE usando SPRING como gestor de transacciones.

Teniendo en cuenta todo esto está claro que el proyecto necesitaría ser reenfocado obviando o minimizando las partes de acceso a datos ya que GRAILS lo hace por nosotros. También sería necesario replantear el ciclo de desarrollo para la interfaz gráfica, ya que con GRAILS es trivial generar una primera interfaz en la que basar nuestros diseños.

En el lado negativo tenemos un pobre soporte por parte de los IDEs más comunes a GRAILS. Los plugins disponibles para Eclipse y Netbeans no son muy estables y sí terriblemente lentos en equipos normales. Por tanto para su utilización es necesario un equipo con abundante memoria y lo más rápido posible.

Es preocupante es la gestión de las sesiones que realiza GRAILS. Prácticamente te obliga a utilizar alguno de los plugins disponibles para el FRAMEWORK, lo cual lo más seguro implicaría volver a replantear algunos aspectos de la aplicación.

Por último, la versión que se ha utilizado ha demostrado tener serios problemas con algunas funciones importantes que en versiones anteriores funcionaban perfectamente (ejemplo: gestión de sesiones).

En resumen, GRAILS es un FRAMEWORK apto para realizar desarrollos rápidos y que no requieran interfaces muy específicas. Es especialmente recomendable para aquellas aplicaciones donde haya muchos mantenimientos (operaciones CRUD) ya que nos ahorrará una considerable cantidad de trabajo. Creo que en cuanto haya pulido algunos errores y tenga una mejor integración con IDEs estaremos ante una gran alternativa para desarrollar.





## **Bibliografía**

The Definitive Guide To Grails  
ISBN-13 (pbk): 978-1-59059-758-3  
Graeme Keith Rocher

Getting Started With Grails  
Free-Online Edition  
Jason Rudolph (forward by Venkat Subramaniam)  
<http://infoq.com/minibooks/grails>

Grails - Online Tutorials  
<http://www.grails.org/Tutorials>



## Anexo 1. Instalación de GRAILS

Todo lo que se cite en este punto es referente a la versión de WINDOWS, pero siendo en su mayoría aplicable al resto de versiones.

Una vez descargado e instalado, tendremos acceso a la consola de GRAILS mediante el siguiente icono



Si lo pulsamos se abra un intérprete de línea de comandos.

Escribimos

```
grails help
```

Si aparece una lista de comandos es que está bien instalado. A continuación sólo tenemos que seguir los pasos del tutorial que hay en

<http://www.grails.org/Quick+Start>

para tener nuestra primera aplicación.



## **Anexo 2. Creación, compilación y despliegue del proyecto.**

Arrancamos el entorno de GRAILS y creamos la aplicación gestio. Una vez creada sustituimos el contenido del directorio por el de nuestro archivo. Después nos movemos al subdirectorio "gestio" y ejecutamos

```
grails run-app
```

Al cabo de unos instantes nos informará de que la aplicación ha arrancado y está disponible en

<http://localhost:8080/gestio>

Después de comprobar el funcionamiento interrumpimos la ejecución con CTRL+C y escribimos

```
grails war
```

y nos generará el fichero "gestio-<versio>.war" el cual podremos desplegar en un servidor de aplicaciones.

Nota: En el fichero WAR se incluye una distribución de EHCache, una librería que acelera los accesos del ClassLoader de JAVA pero da problemas con algunos servidores.