



Eusko Jaurlaritzaren Informatika Elkartea
Sociedad Informática del Gobierno Vasco

Aplicación para el Órgano Administrativo de Recursos Contractuales

Especificación de diseño del sistema

Fecha: 26/12/2012

Referencia: 047/2012

EJIE S.A.
Mediterráneo, 14
01010 Vitoria-Gasteiz
Posta-kutxatila / Apartado: 809
01080 Vitoria-Gasteiz
Tel. 945 01 73 00*
Fax. 945 01 73 01
www.ejie.es

Este documento es propiedad de EJIE, S.A. y su contenido es confidencial. Este documento no puede ser reproducido, en su totalidad o parcialmente, ni mostrado a otros, ni utilizado para otros propósitos que los que han originado su entrega, sin el previo permiso escrito de EJIE, S.A.. En el caso de ser entregado en virtud de un contrato, su utilización estará limitada a lo expresamente autorizado en dicho contrato. EJIE, S.A. no podrá ser considerada responsable de eventuales errores u omisiones en la edición del documento.

Control de documentación

Histórico de versiones

Versión:	Fecha:	Resumen de cambios:
1.0	26/12/2012	Versión inicial

Cambios producidos desde la última versión

Primera versión.

Registro

Autor:

Firma:

Fecha: 26/12/2012

Aprobado por:

Firma:

Fecha:

Distribución:

Contenido

	Capítulo/sección	Página
1	Descripción del sistema	6
1.1	Objetivos del sistema	6
1.2	Alcance del sistema	6
1.3	Entorno tecnológico del sistema	9
2	Glosario	10
3	Diagrama de arquitectura del sistema	11
3.1	Descripción de los niveles de arquitectura	11
3.2	Diagrama de arquitectura de soporte	12
3.3	Descripción de los mecanismos genéricos	12
4	Requisitos de los procesos de soporte del sistema	14
5	Normas y estándares aplicables en diseño	15
6	Agrupaciones de casos de uso	16
6.1	Agrupación Solicitud de recurso (001)	17
6.2	Agrupación Generación de expediente de recurso (002)	17
6.3	Agrupación Expediente de contratación (003)	18
6.4	Agrupación Subsanación (004)	19
6.5	Agrupación Traslado y alegaciones (005)	20
6.6	Agrupación Medidas Provisionales (006)	20
6.7	Agrupación Práctica de prueba (007)	21
6.8	Agrupación Resolución del recurso (008)	21
6.9	Agrupación histórico (009)	22
6.10	Agrupación Acceso a expedientes (010)	22
6.11	Agrupación Consulta pública de resoluciones (011)	22
6.12	Agrupación Consulta pública de resoluciones (012)	23
6.13	Caso de uso Presentación del recurso (UC-001)	23
6.14	Caso de uso Envío expediente de contratación (UC-002)	26
6.15	Caso de uso Generación de expediente de recurso (UC-003)	28
6.16	Caso de uso Notificación de envío de expediente de contratación (UC-004)	29
6.17	Caso de uso Mantenimiento de expediente de contratación (UC-005)	30
6.18	Caso de uso Notificación de subsanación (UC-006)	31

1 Descripción del sistema

1.1 Objetivos del sistema

El objetivo consiste en el desarrollo de un sistema que sirva para cubrir todas las necesidades del Órgano Administrativo de Recursos Contractuales de la CAPV (OARC) para atender a la totalidad de los recursos especiales y a las cuestiones de nulidad de la Ley 30/2007 que se interpongan contra los actos de los poderes adjudicadores vascos. Sus resoluciones implican a todos los poderes adjudicadores integrados en los sectores públicos de la Administración General y de las Administraciones Locales de la Comunidad Autónoma de Euskadi. Esto posibilitará que existan criterios únicos y homogéneos en materia de recursos en la contratación pública en las Administraciones Vascas.

1.2 Alcance del sistema

El sistema final dará satisfacción a los requisitos del OARC para el desarrollo del Sistema de Información que atienda los recursos especiales y las cuestiones de nulidad que se interpongan contra los actos de los poderes adjudicadores vascos.

Actualmente se carece de una solución telemática que cubra los procedimientos de:

- Recurso especial en materia de contratación (Ley 30/2007)
- Cuestión de nulidad (Ley 30/2007)

La solución propuesta busca la integración de toda la información manejada por el OARC en el desempeño de su función de forma que **al usuario le resulte fácil el acceso a la información a la que necesite acceder en cada momento**. Se desarrollarán dos aplicaciones una en entorno portal y otra en entorno intranet:

- **Aplicación Intranet OARC:** aplicación web de intranet a la que tendrán acceso los técnicos del OARC encargados de atender los recursos que lleguen tanto desde la aplicación Internet OARC del portal, como desde otros órganos que admitan la presentación de recursos (alta manual). La forma de organizar el trabajo en esta aplicación es a través del concepto de expediente, o conjunto de información asociada a un procedimiento administrativo. La aplicación gestionará los expedientes de recurso contra expedientes de contratación. Para manejar los expedientes de recurso, la aplicación se organiza en bandejas:
 - o **Bandeja de entrada:** en esta bandeja tendremos los eventos de entrada asociados a los expedientes de recurso. Los eventos de entrada agrupan las notificaciones producidas por acciones en la aplicación Internet (por ejemplo, notificación de que un interesado ha leído el recurso).
 - o **Bandeja de salida:** en esta bandeja estarán los eventos de salida generados por los técnicos del OARC en relación con los expedientes tratados. Entendemos por evento de salida a las notificaciones enviadas desde la aplicación, relacionadas con los expedientes de recurso.

- o **Bandeja de tareas del OARC:** en esta bandeja se muestran las tareas a realizar por el OARC colocadas en un calendario.

Aplicación Internet OARC (portal): aplicación web a desplegar en el portal de contratación y a la que tendrán acceso los ciudadanos, las empresas y los órganos de contratación. La funcionalidad aportada por esta aplicación consistirá en facilitar el acceso a la información de las resoluciones (información pública) y funcionalidad propia de la presentación de un recurso (formulario de solicitud de recurso, consulta de los datos del recurso,...).

1.3 Entorno tecnológico del sistema

Los principales componentes del Entorno Tecnológico en el que deberá ejecutarse el Sistema y que estarán sujetos a los estándares de EJIE, son los siguientes:

- Entorno Tecnológico J2EE :
 - o **Framework J2EE UDA**
- Entorno Web - Plataforma Java:
 - o **Servidor Web Apache Web Server 2 sobre Linux Red Hat Enterprise 3**
 - o **Servidor de Aplicaciones BEA Weblogic Server 11g sobre Linux Red Hat Enterprise 3**
 - o **Seguridad XLNET's y PKI Izenpe**
 - o **Gestión de Contenidos Interwoven TeamSite 6.7, Open Deploy 6.1, Autonomy**
 - o **Servicios horizontales del Gobierno Vasco. PLATEA.**
- Base de Datos:
 - o **Oracle 10g sobre HP-UX**

El sistema se desarrollará con la siguiente compatibilidad de navegadores para operar correctamente:

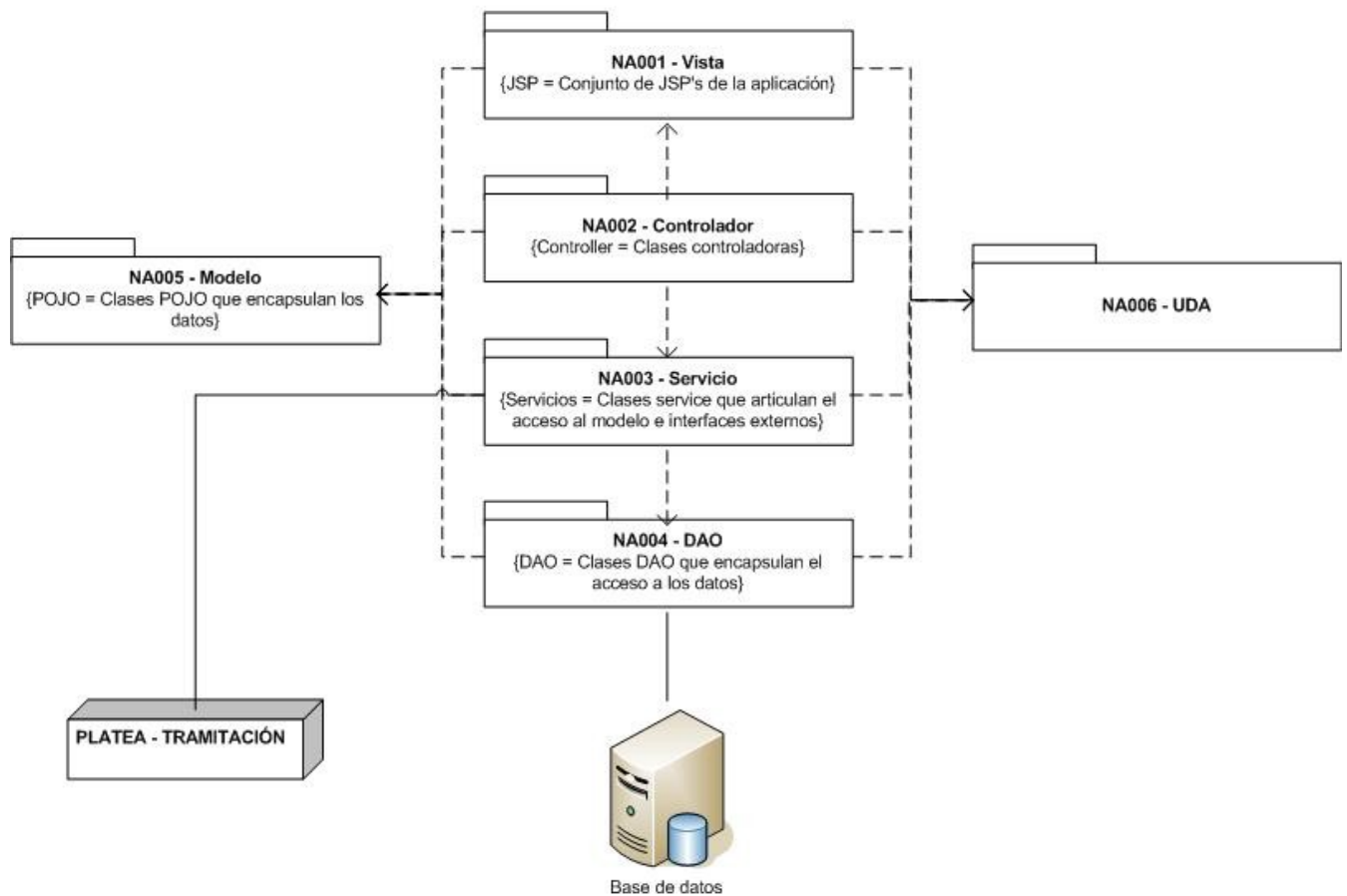
- Internet Explorer 8 o superior, Firefox 3+ o Google chrome 9.
- Resolución mínima de 1024x768, permitiendo mediante el ajuste automático de la pantalla activa, la visualización de resoluciones superiores.

2 Glosario

Término	Definición

3 Diagrama de arquitectura del sistema

Arquitectura del Sistema

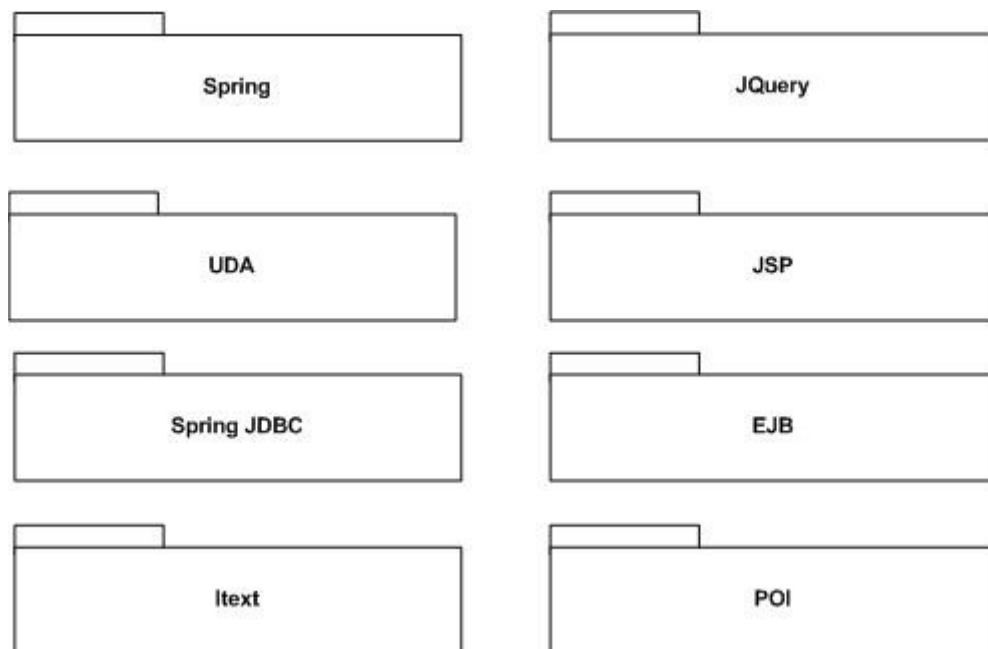


3.1 Descripción de los niveles de arquitectura

Id	Descripción
NA-001	Capa de presentación de la aplicación
NA-002	Capa controladora de toma de decisiones
NA-003	Capa de servicio que articula el acceso al modelo y a los interfaces externos como PLATEA-Tramitación

Id	Descripción
NA-004	Capa de acceso a datos. Encapsula los accesos a la base de datos
NA-005	Capa de modelo desarrollada con POJO's. Esta capa es transversal y permite al resto de capas horizontales transmitir los datos basándose en los mismos beans.
NA-006	Utilidades de Desarrollo de Aplicaciones de EJIE.

3.2 Diagrama de arquitectura de soporte



3.3 Descripción de los mecanismos genéricos

Tipo: S: servicio, M: módulo, C: clase

Nombre	Descripción	Tipo
Spring	Módulo con las clases genéricas de Spring	M
JQuery	Módulo con los scripts de JQuery	M
NA-006-UDA	Módulo con las clases de utilidades de EJIE	M
JSP	Módulo con las clases genéricas de JSP	M

Nombre	Descripción	Tipo
Spring JDBC	Módulo con las clases genéricas de JDBC (acceso a base de datos)	M
EJB	Módulo con la cola JMS para la recepción de eventos de PLATEA-Teletramitación	C
IText	Módulo con las clases genéricas de IText (generación Word y pdf)	M
POI	Módulo con las clases genéricas de POI (generación de excel)	M

4 Requisitos de los procesos de soporte del sistema

Requisitos de diseño y construcción:

Id	Descripción
RQ-DC-001	Se utilizarán las utilidades de desarrollo de aplicaciones de EJIE UDA versión 2.1.0
RQ-DC-002	El servidor de aplicaciones será Weblogic Server 11g (10.3.5)
RQ-DC-003	Se usarán los siguientes navegadores para acceder a la aplicación: Internet Explorer 8 o superior, Firefox 3+ o Google Chrome 9
RQ-DC-004	La base de datos será Oracle 10g

Requisitos de operación y seguridad:

Id	Descripción
RQ-OS-001	Comprobar cada cierto tiempo que el sistema no esté caído.
RQ-OS-002	El acceso a través de XLNets, en la parte Internet protegida, se hará a través de certificado digital
RQ-OS-003	Conectividad con los sistemas de PLATEA-Tramitación

Requisitos de documentación de usuario:

Id	Descripción
RQ-DU-001	Entrega del manual de usuario

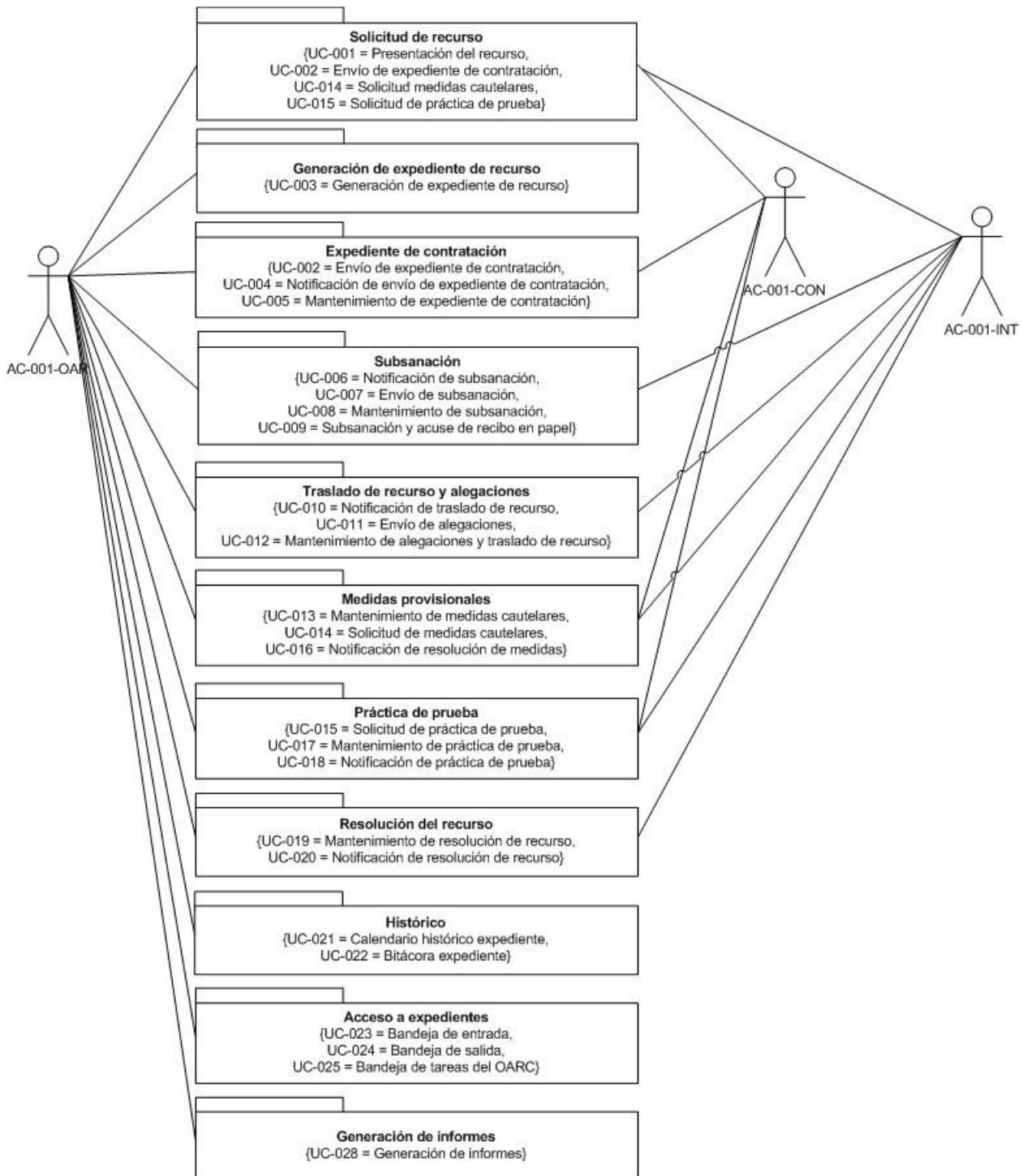
Requisitos de implantación:

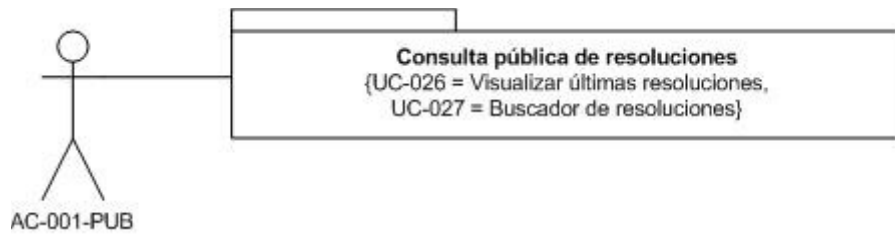
Id	Descripción
RQ-IS-001	Formación de 2 días a los usuarios finales

5 Normas y estándares aplicables en diseño

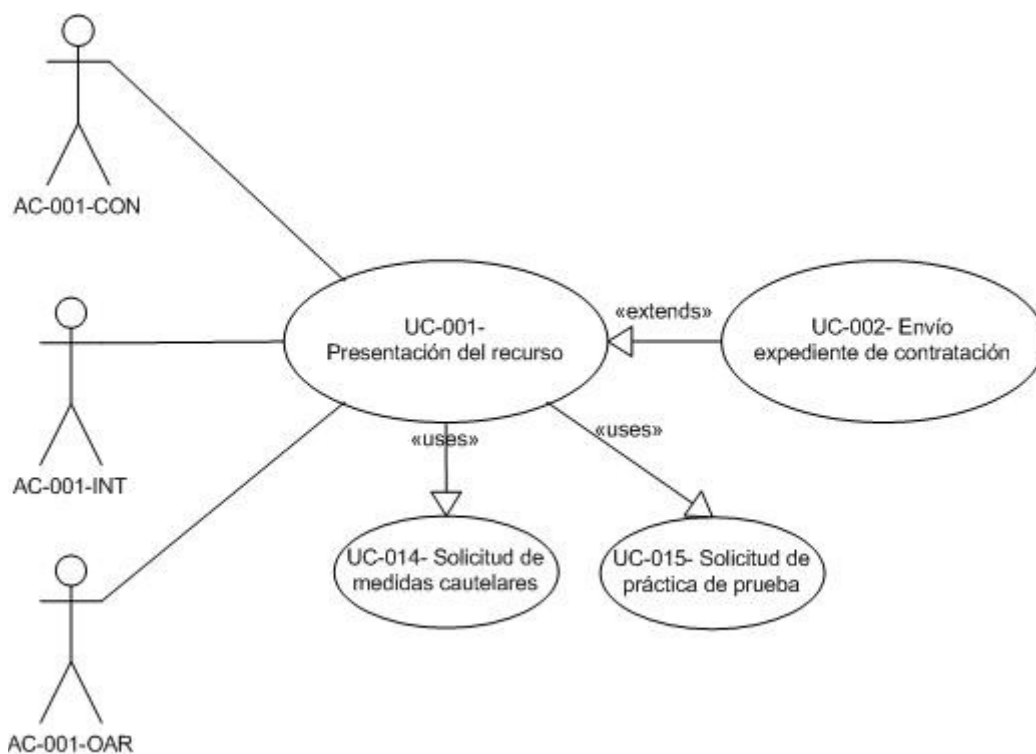
Nombre	Descripción
Estándar W3C	Normativa de accesibilidad y usabilidad Web. Desarrollo que cumpla las normativas de accesibilidad y usabilidad de las aplicaciones Web del Gobierno Vasco.
XLNetS	Modelo de Seguridad para las Comunicaciones de la Red Corporativa del Gobierno Vasco (XL-Nets).
Manual UDA	Guías de uso de UDA.
Libro de estilo del Gobierno Vasco	Libro de estilo de las aplicaciones de intranet e Internet del Gobierno Vasco.
ARINBIDE	ARINBIDE basada en la metodología de desarrollo de Sistemas de Información Métrica v3.
Albergue Weblogic	Normas de funcionamiento y albergue Intranet e Internet del Gobierno Vasco. Nomenclatura de los paquetes y clases.
Albergue Oracle	Normas de creación de esquemas de Bases de Datos
PROBAMET	Normativa de pruebas adoptada por EJIE
SQA	Modelo de aseguramiento de la calidad de EJIE

6 Agrupaciones de casos de uso

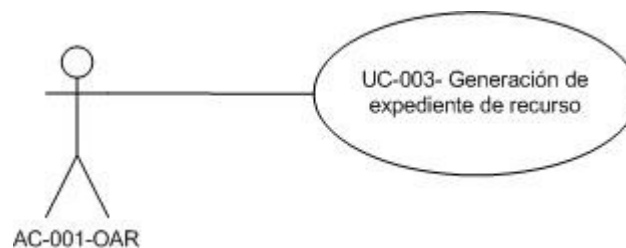




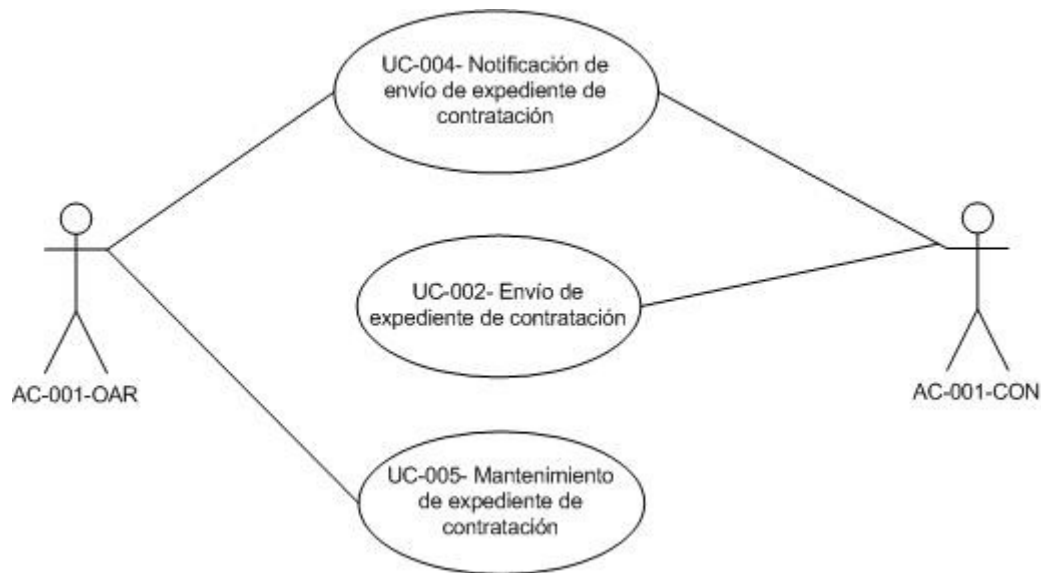
6.1 Agrupación Solicitud de recurso (001)



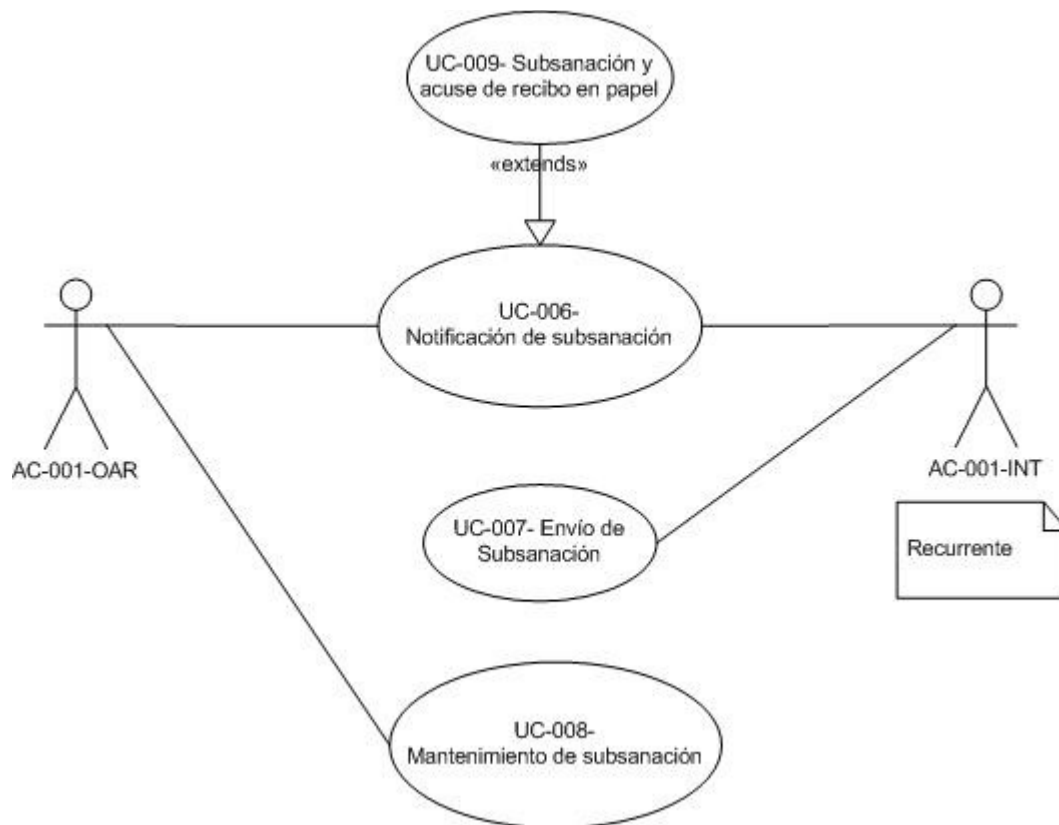
6.2 Agrupación Generación de expediente de recurso (002)



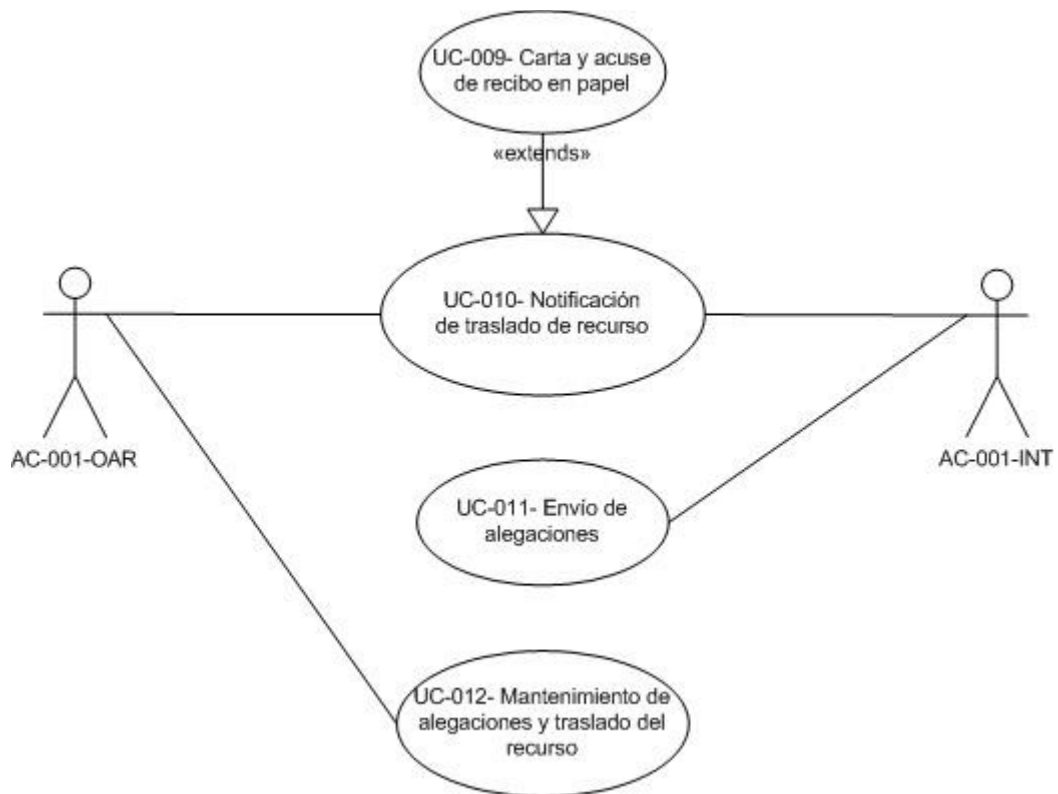
6.3 Agrupación Expediente de contratación (003)



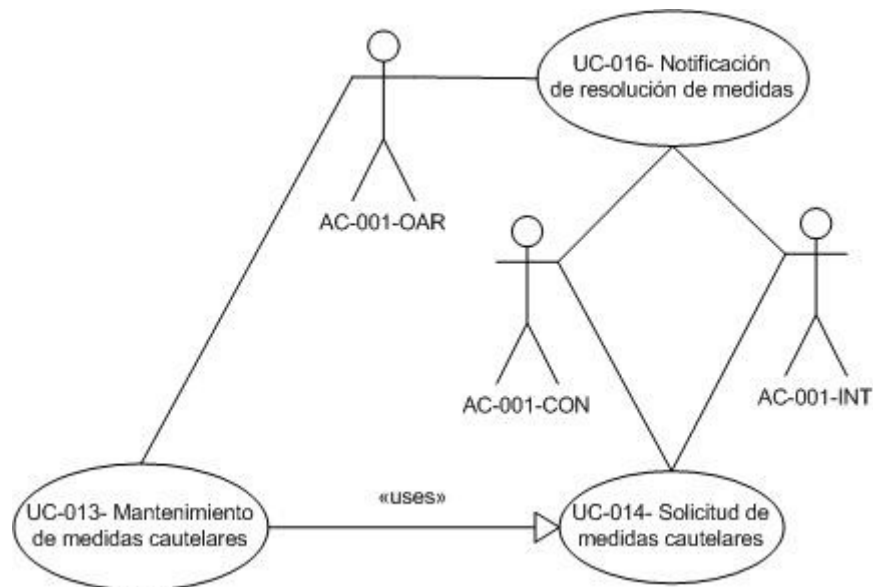
6.4 Agrupación Subsanación (004)



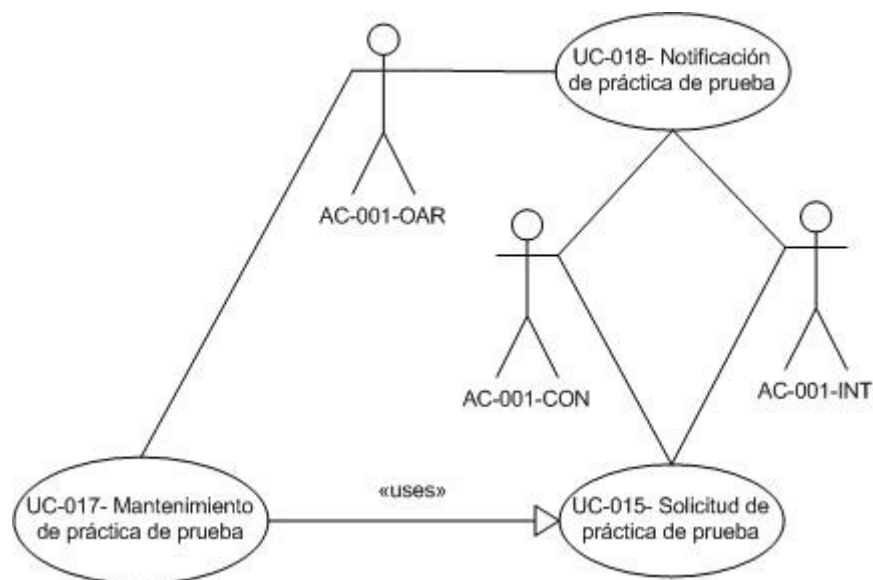
6.5 Agrupación Traslado y alegaciones (005)



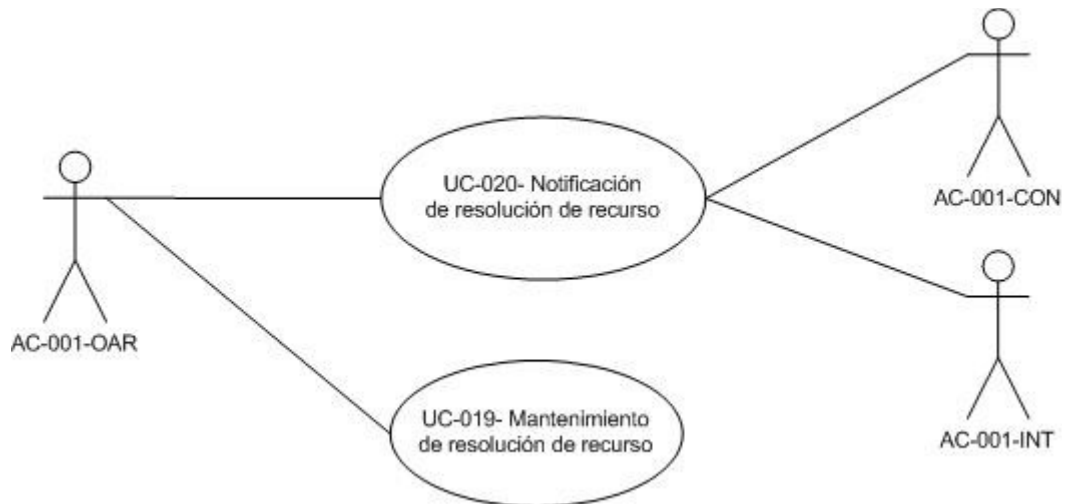
6.6 Agrupación Medidas Provisionales (006)



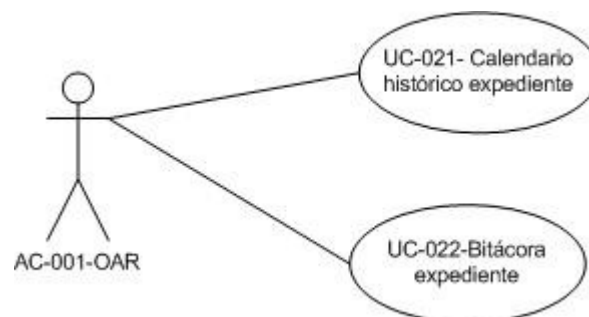
6.7 Agrupación Práctica de prueba (007)



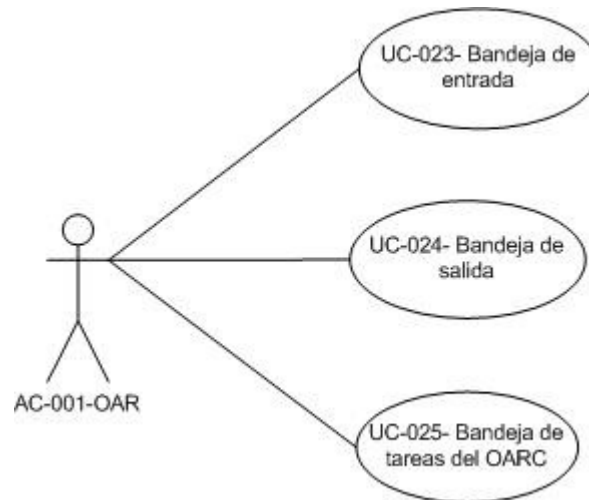
6.8 Agrupación Resolución del recurso (008)



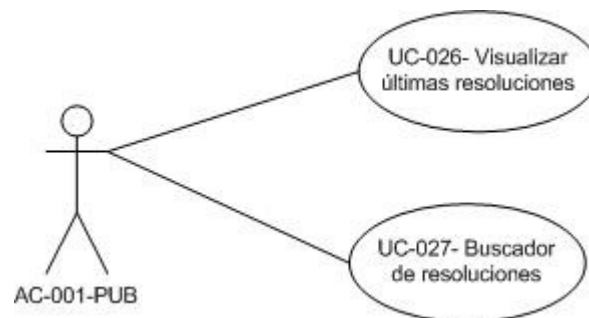
6.9 Agrupación histórico (009)



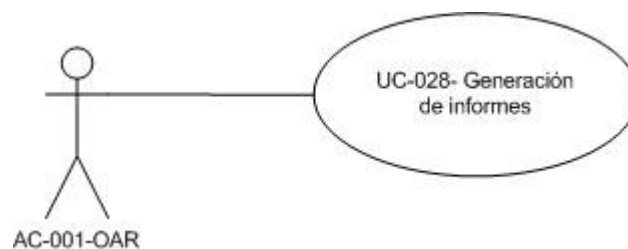
6.10 Agrupación Acceso a expedientes (010)



6.11 Agrupación Consulta pública de resoluciones (011)



6.12 Agrupación Consulta pública de resoluciones (012)



6.13 Caso de uso Presentación del recurso (UC-001)

Denominación:

Representa las acciones que pueden tomar los diferentes actores (Recurrente, Órgano de contratación) para presentar un recurso ante el OARC. La presentación del recurso también incluye la solicitud de medidas cautelares y la solicitud de práctica de prueba si proceden. Hay diferentes tipos de presentación según el actor que añada el recurso al sistema de información:

- Recurrente: presentación telemática de una solicitud de recurso o en papel.
- Órgano de contratación: presentación telemática de un recurso y envío del expediente de contratación asociado.
- OARC: Alta de solicitudes de recurso que le llegan al OARC de forma no telemática.

Actor:

AC-001-OAR, AC-001-INT (Recurrente), AC-001-CON

Precondición:**Postcondición:**

Solicitud de recurso incluida en el sistema de información.

Escenario 1: Presentación de solicitud telemática de recurso por el recurrente

- 1- El recurrente accede al formulario del Portal de contratación para presentación de recurso mediante certificado digital.
- 2- El recurrente rellena la información de solicitud del recurso y envía la información.
- 3- El sistema devuelve un justificante de registro y una fecha de solicitud de registro al recurrente.
- 4- El sistema almacena la solicitud de recurso.

Escenario 2: Presentación de solicitud de recurso por el Órgano adjudicador

- 1- El Órgano adjudicador accede al formulario del Portal de contratación para remisión de expediente por poder adjudicador mediante certificado digital.
- 2- El Órgano adjudicador rellena la información de solicitud del recurso, junto con la del expediente de contratación y envía la información.
- 3- El sistema devuelve un justificante de registro y una fecha de solicitud de registro al Órgano adjudicador.
- 4- El sistema almacena la solicitud de recurso.

Escenario 3: Presentación de solicitud de recurso en papel ante el OARC

- 1- El OARC recibe la solicitud de recurso de forma no telemática.
- 2- El OARC realiza un alta de solicitud de recurso en el formulario de la intranet.
- 4- El sistema almacena la solicitud de recurso.

Diagrama de secuencia:

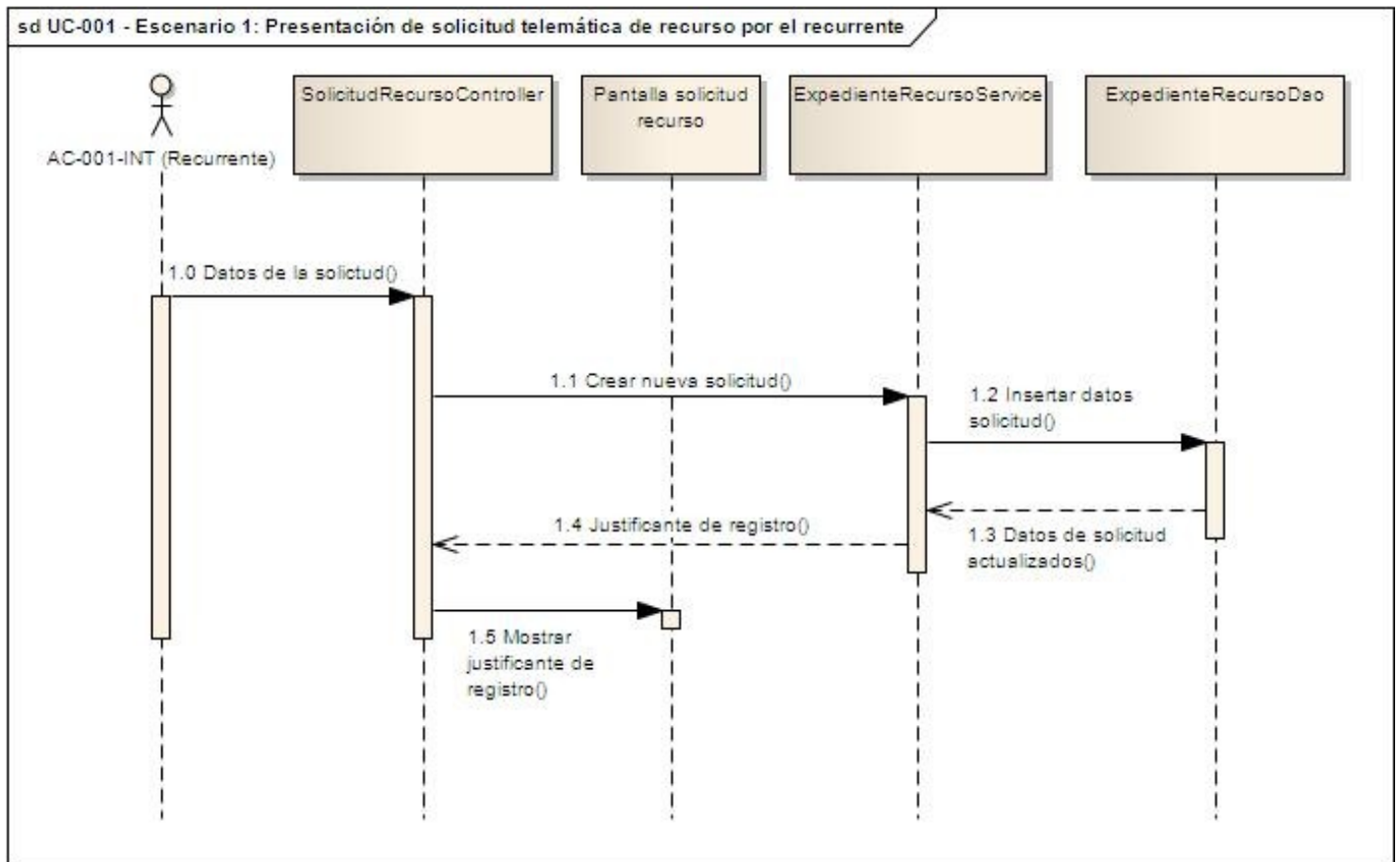


Diagrama de secuencia:

sd UC-001 - Escenario 2: Presentación de solicitud de recurso por el Órgano adjudicador

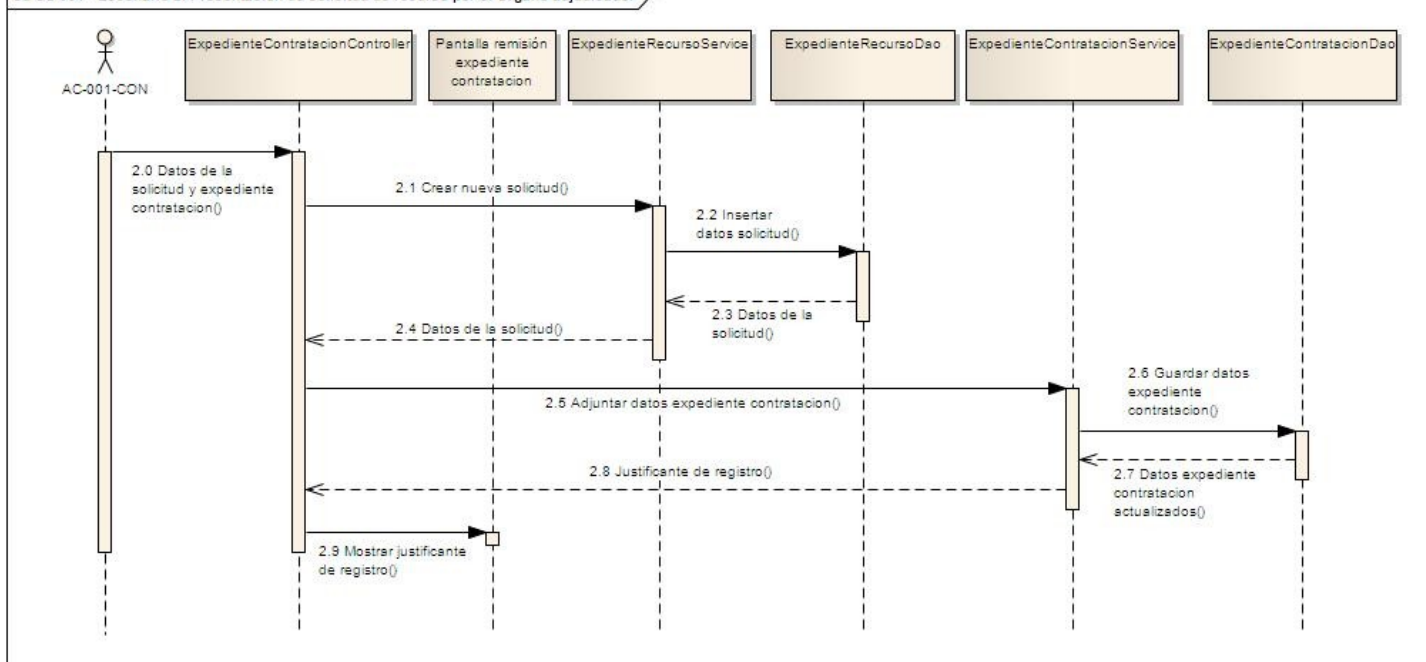
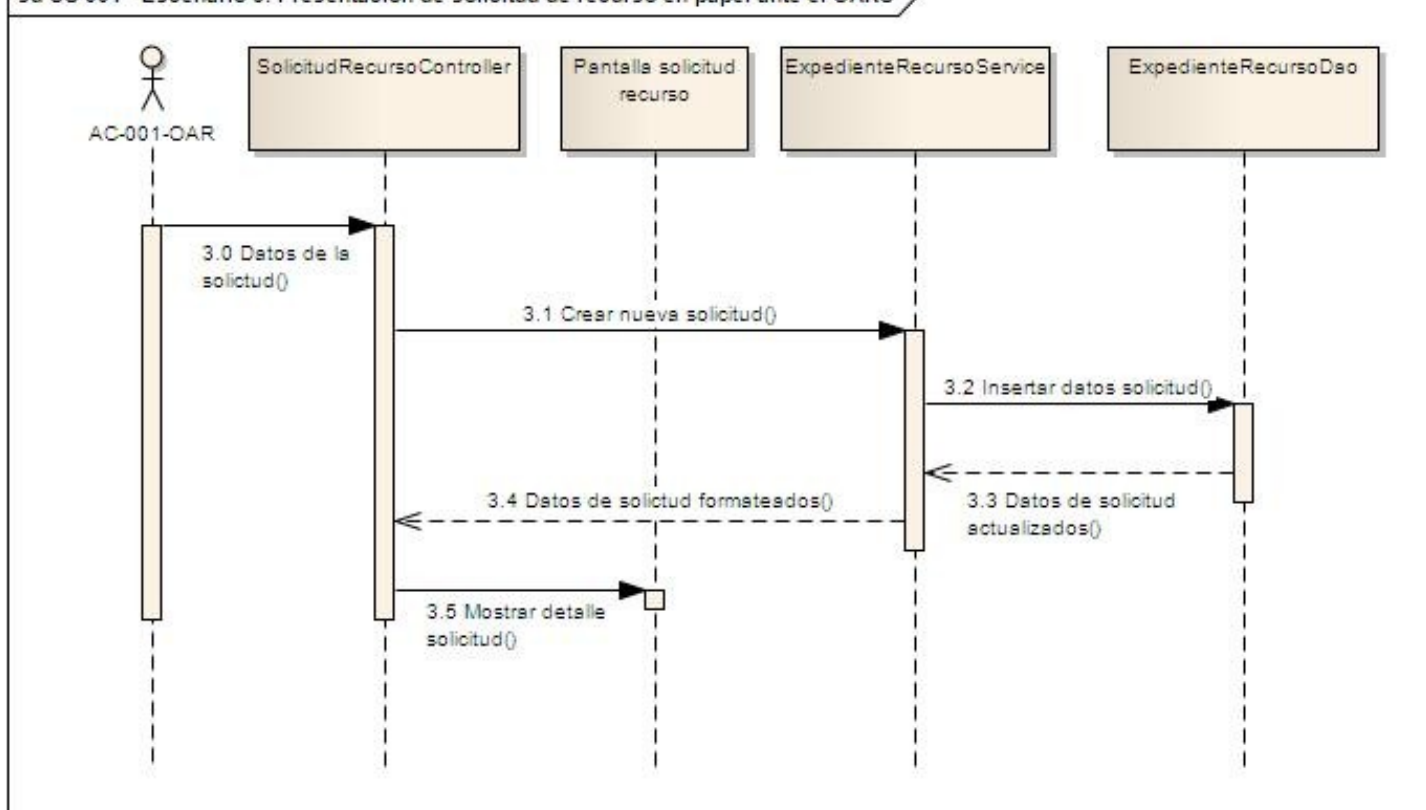


Diagrama de secuencia:

sd UC-001 - Escenario 3: Presentación de solicitud de recurso en papel ante el OARC



6.14 Caso de uso Envío expediente de contratación (UC-002)

Denominación:

El envío del expediente de contratación por el poder adjudicador/Órgano de contratación, se puede realizar en diferentes fases. En el caso de que el recurrente presente el recurso ante el Órgano de contratación, éste último envía el expediente de contratación, junto con la solicitud de recurso al OARC. Para el caso en que la solicitud de recurso se presente ante el OARC, éste último solicitará al Órgano de contratación enviar el expediente de contratación únicamente, ya que la solicitud de recurso ya ha sido presentada directamente al OARC.

Actor:

AC-001-CON

Precondición:

En el escenario 1 la solicitud de recurso se ha presentado ante el Órgano de contratación.

En el escenario 2 la precondición implica que el recurso se haya presentado ante el OARC.

Postcondición:

Expediente de contratación almacenado en el sistema.

Escenario 1: Envío del expediente de contratación junto con la solicitud del recurso

- 1- El Órgano adjudicador accede al formulario del Portal de contratación para remisión de expediente por poder adjudicador mediante certificado digital
- 2- El Órgano adjudicador rellena la información de solicitud del recurso, junto con la del expediente de contratación y envía la información.
- 3- El sistema devuelve un justificante de registro y una fecha de solicitud de registro al Órgano adjudicador.
- 4- El sistema almacena la solicitud de recurso.

Escenario 2: Envío del expediente de contratación tras solicitud del OARC

- 1- El Órgano adjudicador accede al formulario del Portal de contratación para remisión de expediente de contratación.
- 2- El Órgano adjudicador rellena la información del expediente de contratación y envía la información.
- 3- El sistema devuelve una fecha y un justificante de registro al Órgano adjudicador.
- 4- El sistema almacena la información del expediente de contratación.

Diagrama de secuencia:

sd UC-002 - Escenario 1: Envío del expediente de contratación junto con la solicitud del recurso

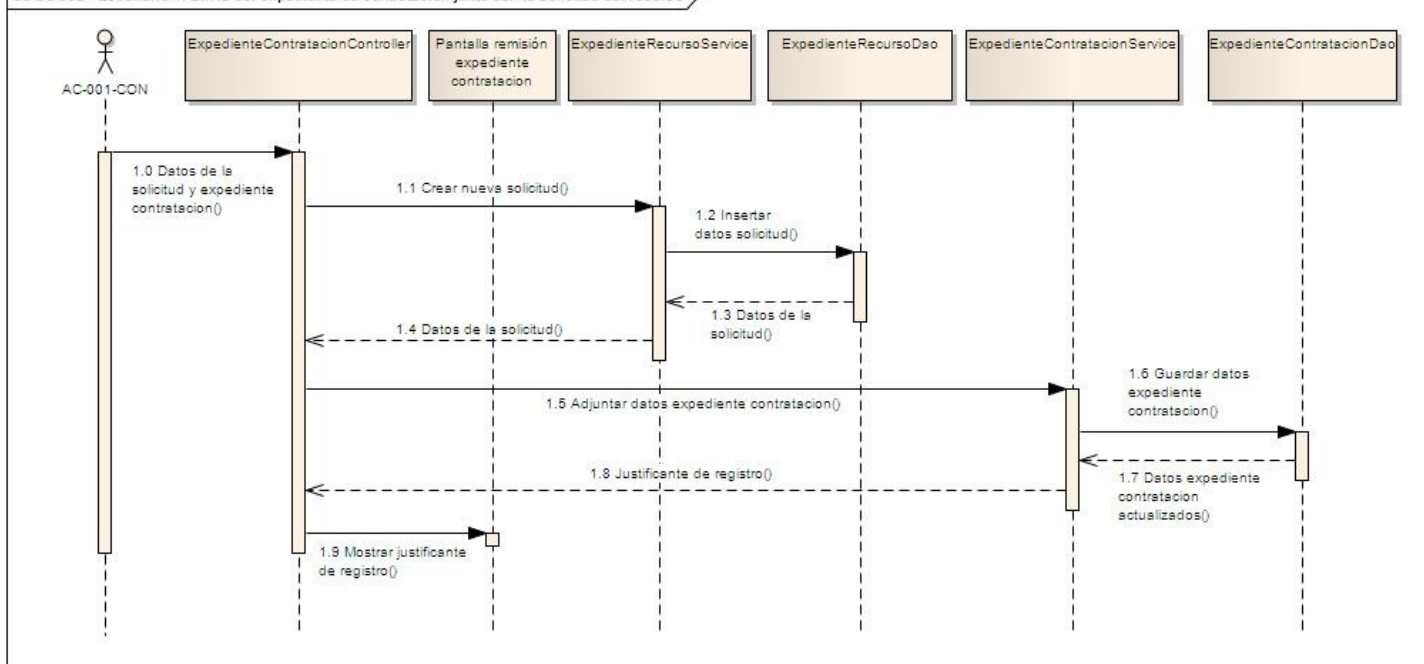
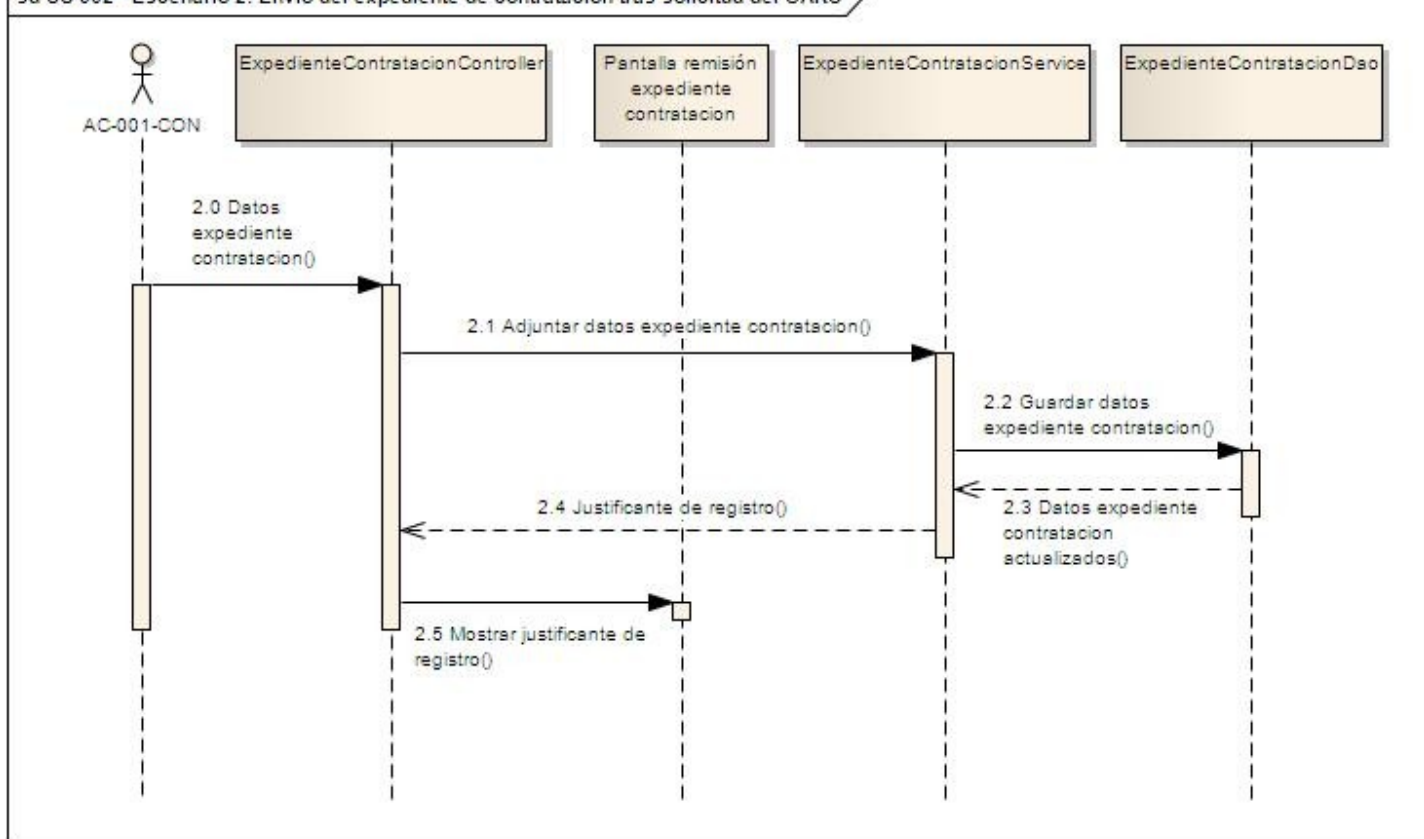


Diagrama de secuencia:

sd UC-002 - Escenario 2: Envío del expediente de contratación tras solicitud del OARC



6.15 Caso de uso Generación de expediente de recurso (UC-003)

Denominación:

Este caso de uso permite al usuario del OARC generar un expediente a partir de una solicitud de recurso.

Actor:

AC-001-OAR

Precondición:

Existe una solicitud de recurso. No se ha generado el expediente de recurso para esa solicitud.

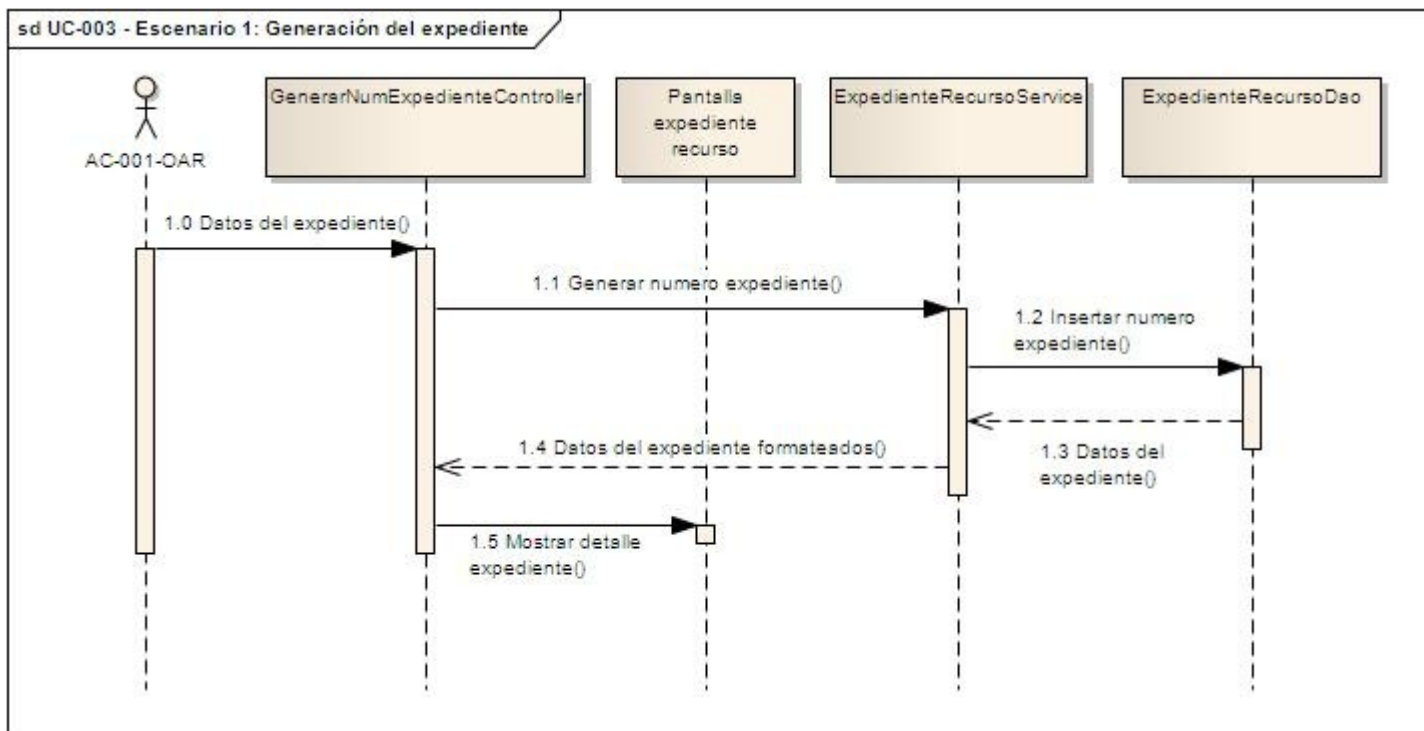
Postcondición:

Expediente generado con su código correspondiente.

Escenario 1: Generación del expediente

- 1- El usuario accede al detalle de una solicitud de recurso.
- 2- El usuario pulsa el botón de alta de expediente.
- 3- El usuario rellena el nombre del expediente de recurso y pulsa guardar.
- 4- El sistema genera el código de expediente asociado, almacena los datos del expediente y muestra el código al usuario.

Diagrama de secuencia:



6.16 Caso de uso Notificación de envío de expediente de contratación (UC-004)

Denominación:

Este caso de uso engloba las acciones que realiza el sistema de información al generar y enviar el OARC la notificación de solicitud del expediente de contratación al Órgano de contratación. Esta notificación informa al poder adjudicador de que debe enviar al OARC el expediente de contratación que se está recurriendo.

Actor:

AC-001-OAR, AC-001-CON

Precondición:

Existe el expediente del recurso.

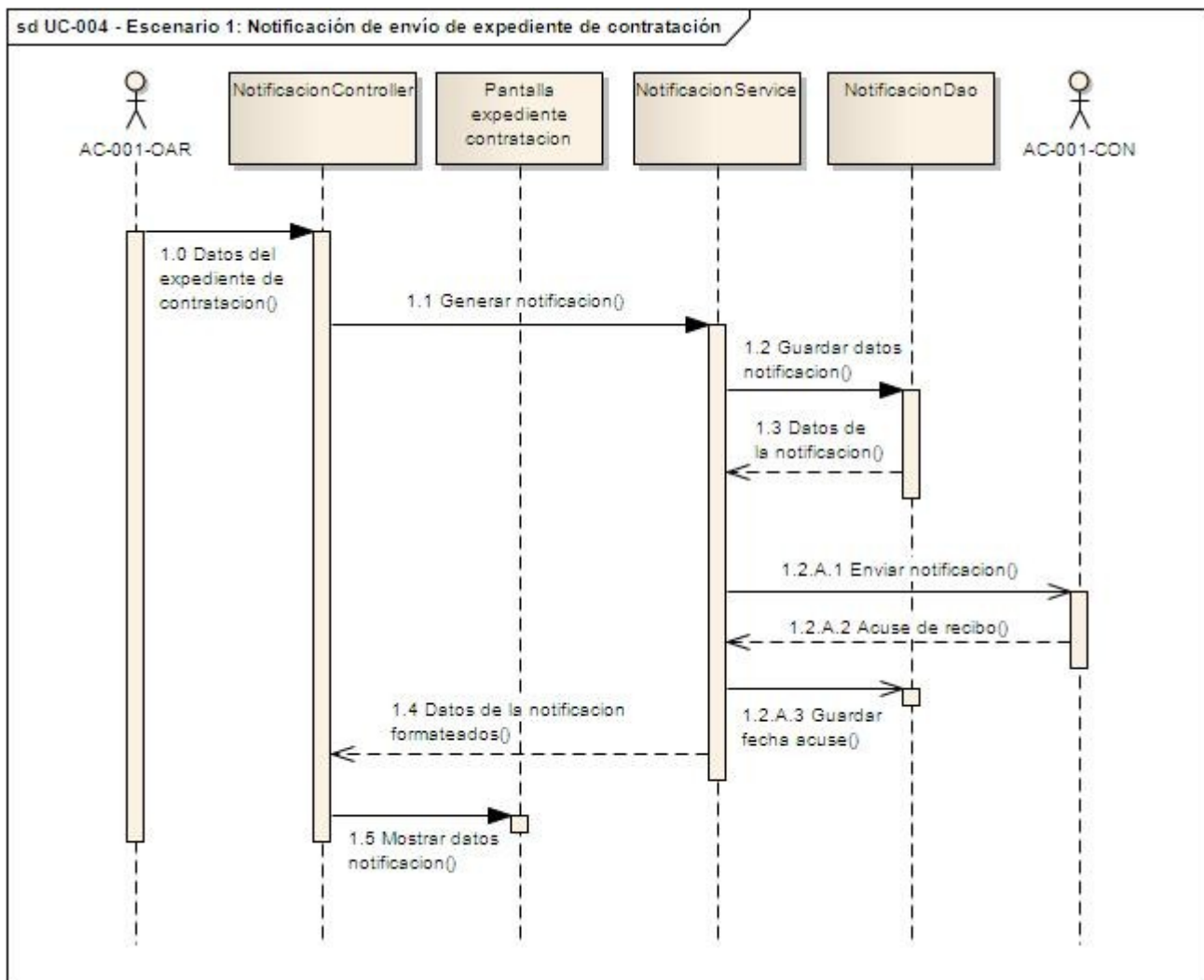
Postcondición:

Notificación enviada al órgano de contratación.

Escenario 1: Notificación de envío de expediente de contratación

- 1- El usuario accede al detalle de expediente de contratación.
- 2- El usuario pulsa el botón de notificación de envío de expediente de contratación.
- 3- El sistema genera y envía la notificación y guarda los datos asociados.

Diagrama de secuencia:



6.17 Caso de uso Mantenimiento de expediente de contratación (UC-005)

Denominación:

Este caso de uso engloba las acciones de mantenimiento del detalle de expediente de contratación. Permite gestionar la lista de interesados asociados al recurso y los documentos adjuntos del expediente. En la lista de interesados se marca el tipo de comunicación que se mantendrá con cada interesado. También podrá adjuntar documento escaneados en caso de que los reciba en papel.

Actor:

AC-001-OAR

Precondición:

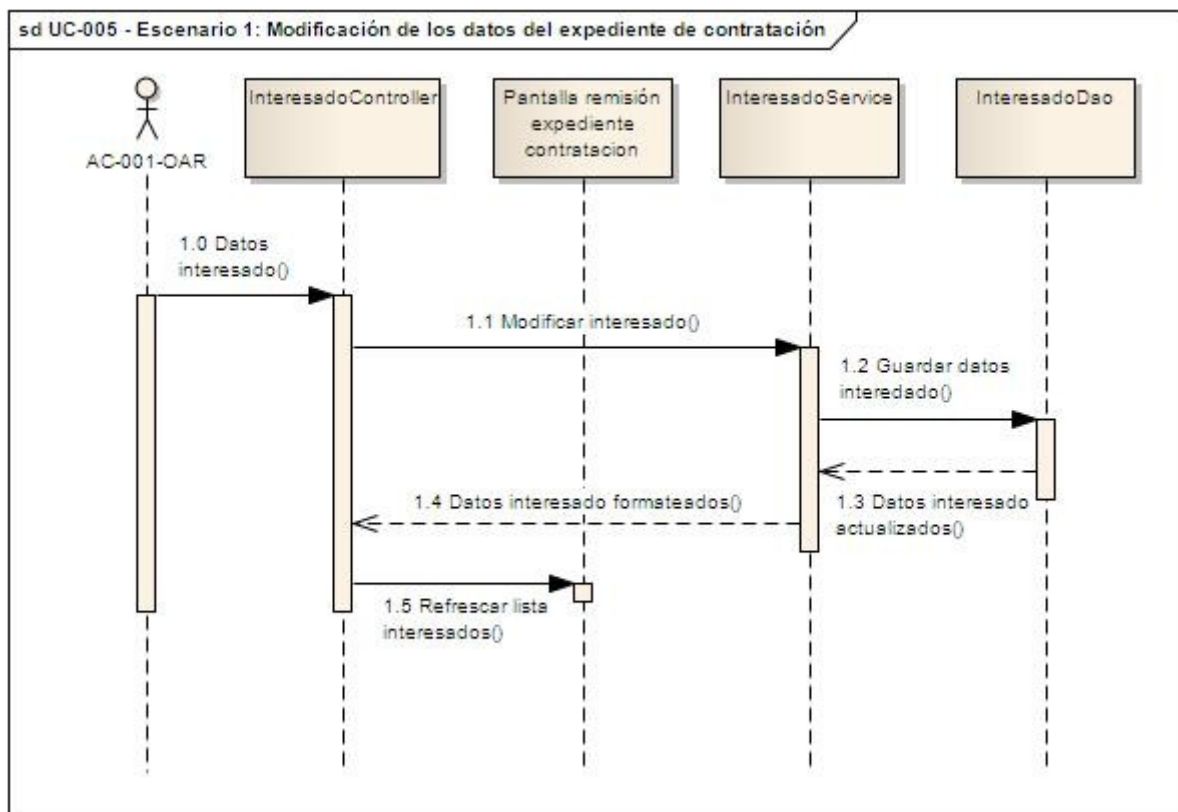
Existe el expediente del recurso.

Postcondición:

Escenario 1: Modificación de los datos del expediente de contratación

- 1- El usuario accede al detalle de expediente de contratación.
- 2- El usuario añade o edita datos de un interesado.
- 3- El sistema guarda los datos asociados.

Diagrama de secuencia:



6.18 Caso de uso Notificación de subsanación (UC-006)

Denominación:

Este caso de uso engloba las acciones que realiza el sistema de información al generar y enviar el OARC la notificación de subsanación al recurrente. Esta notificación informa al recurrente de que debe enviar cierta documentación para subsanar la información enviada en la solicitud de recurso.

Actor:

AC-001-OAR, AC-001-INT

Precondición:

Existe el expediente del recurso.

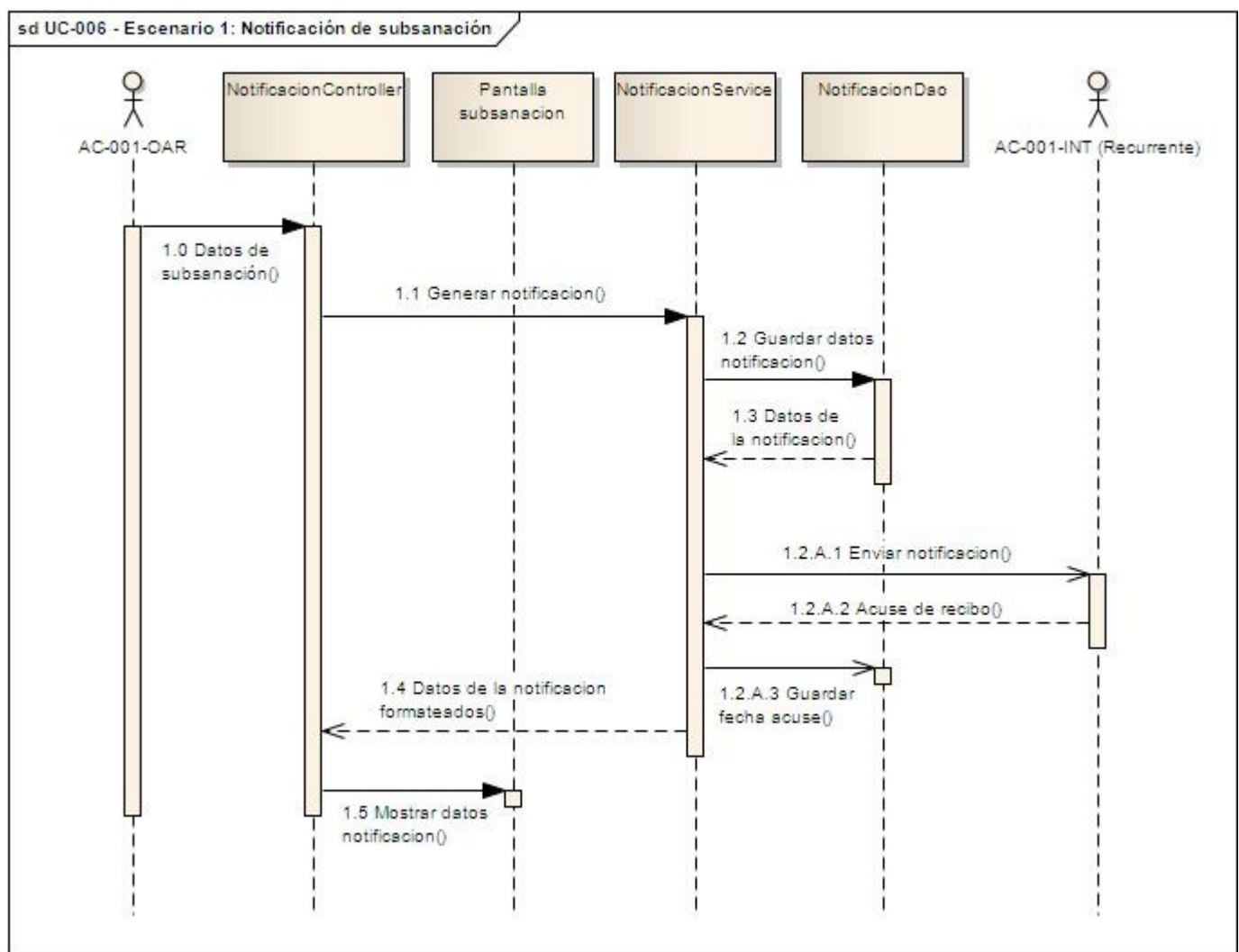
Postcondición:

Notificación enviada al órgano de contratación.

Escenario 1: Notificación de subsanación

- 1- El usuario accede al detalle de subsanación.
- 2- El usuario pulsa el botón de notificación de subsanación.
- 3- El sistema genera y envía la notificación y guarda los datos asociados.

Diagrama de secuencia:



6.19 Caso de uso Envío de subsanación (UC-007)

Denominación:

El recurrente envía al OARC la documentación de subsanación.

Actor:

AC-001-INT

Precondición:

Existe el expediente del recurso. El OARC ha enviado al recurrente la notificación de subsanación.

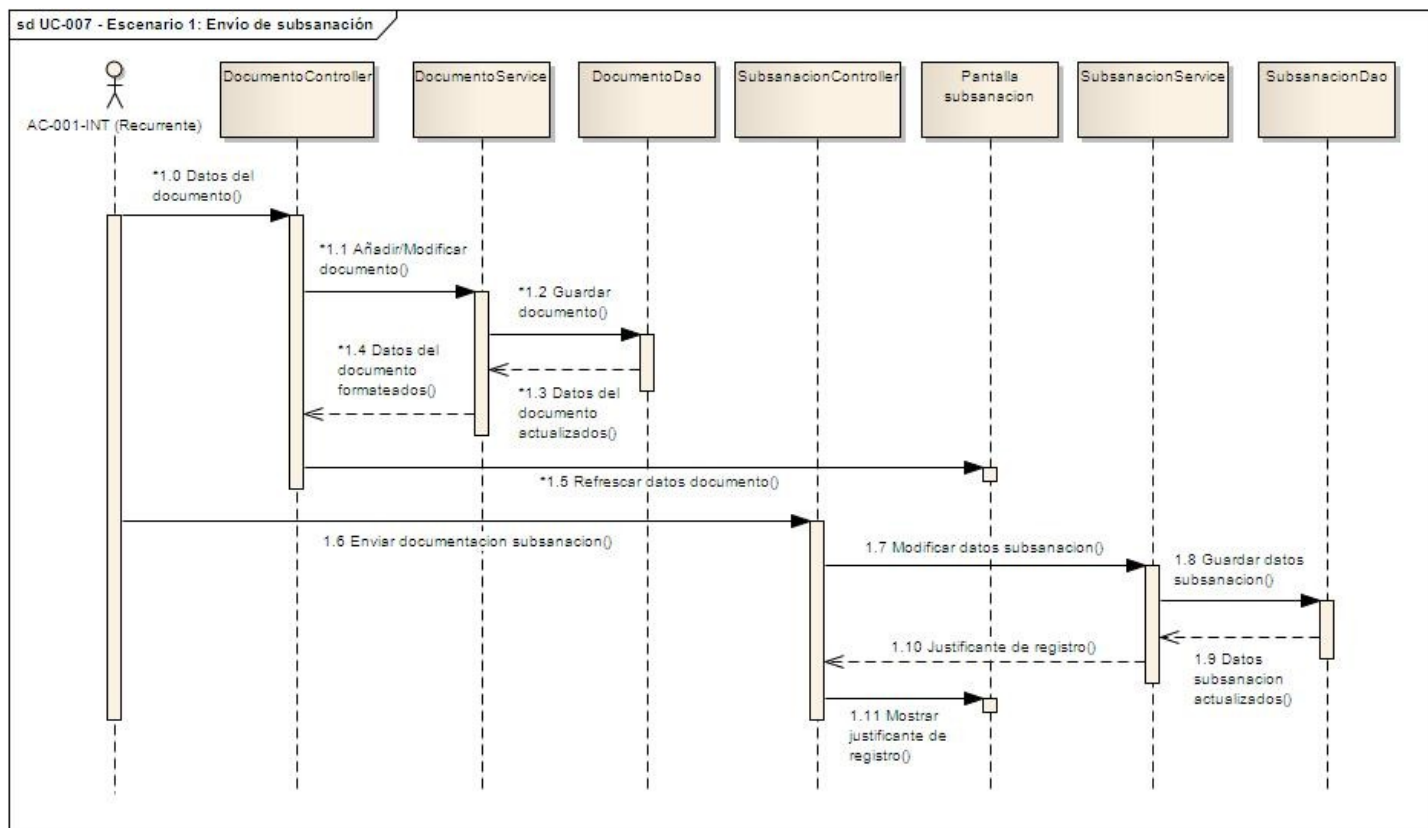
Postcondición:

Documentación de subsanación enviada al OARC.

Escenario 1: Envío de subsanación

- 1- El recurrente accede al detalle de subsanación en el Portal de contratación.
- 2- El recurrente adjunta y envía la documentación de subsanación.
- 3- El sistema guarda los datos asociados y muestra un justificante de registro.

Diagrama de secuencia:



6.20 Caso de uso Mantenimiento de subsanación (UC-008)

Denominación:

El OARC mantiene la información de subsanación actualizándola en caso de recibir en papel la documentación o pudiendo modificar los datos de subsanación.

Actor:

AC-001-OAR

Precondición:

Existe el expediente del recurso.

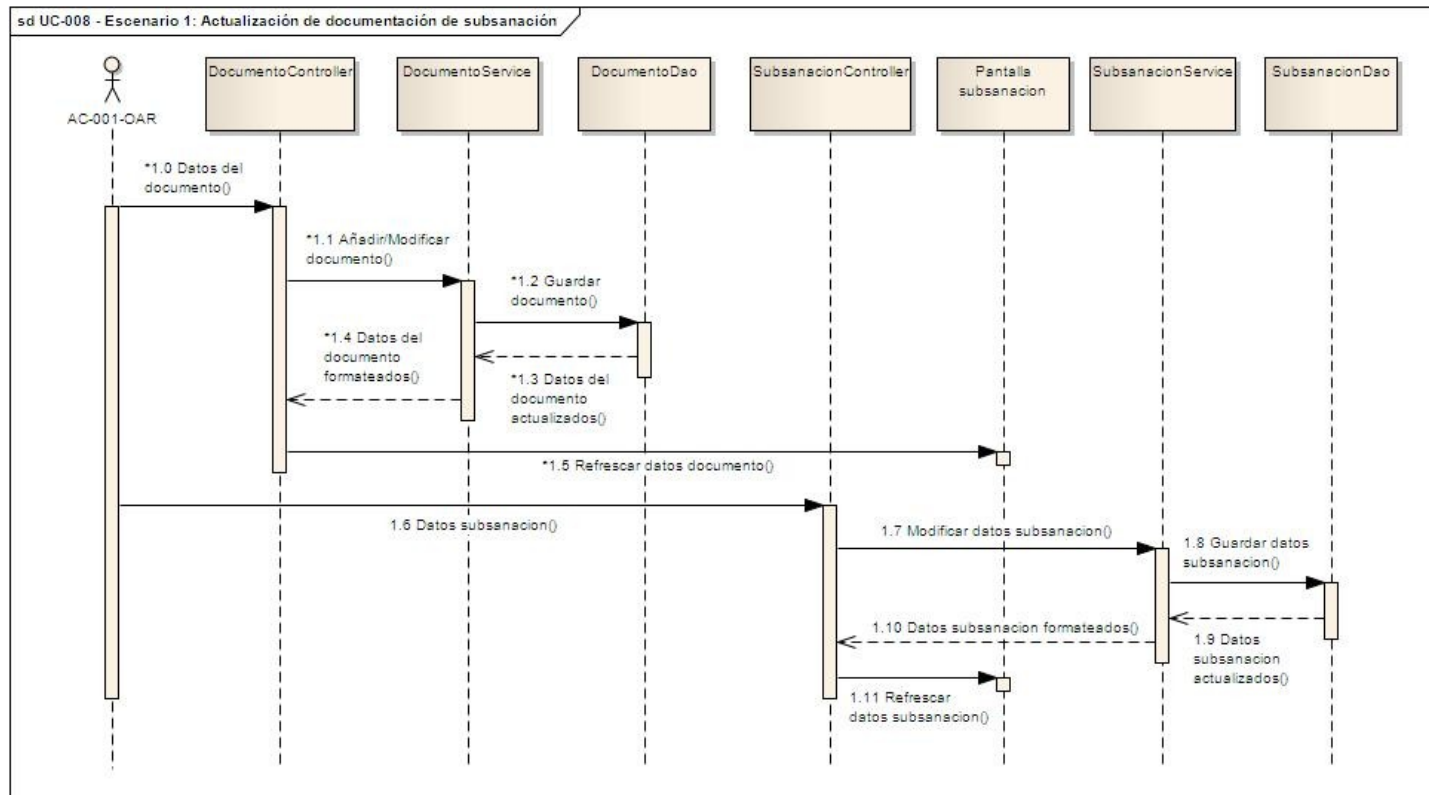
Postcondición:

Datos de subsanación actualizados en el sistema.

Escenario 1: Actualización de documentación de subsanación

- 1- El OARC accede al detalle de subsanación.
- 2- El OARC adjunta documentación de subsanación.
- 3- El sistema guarda los datos asociados.

Diagrama de secuencia:



6.21 Caso de uso Carta y acuse de recibo en papel (UC-009)

Denominación:

El OARC puede generar la notificación, junto con el acuse de recibo en papel.

Actor:

AC-001-OAR

Precondición:

Existe el expediente del recurso.

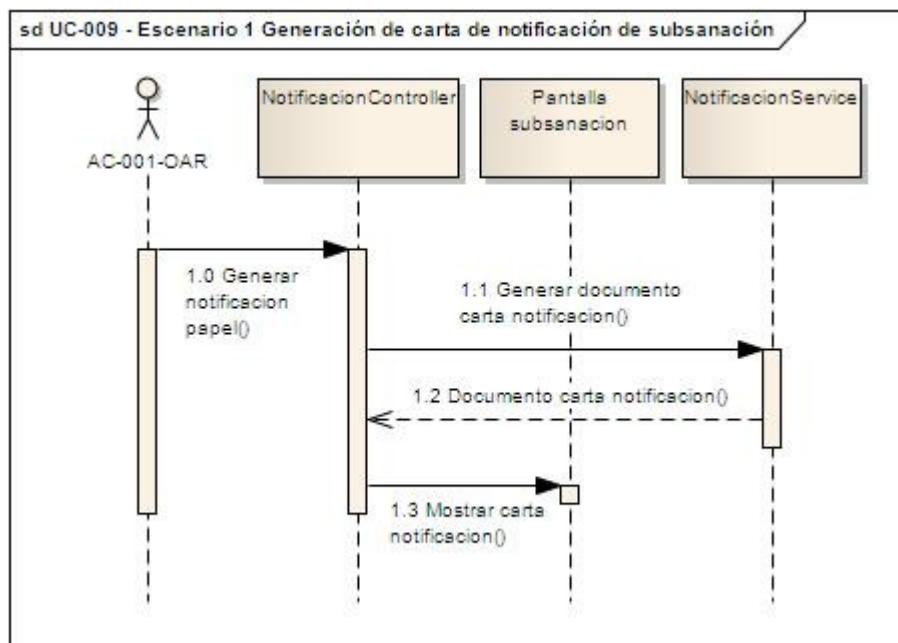
Postcondición:

Documento de notificación de subsanación y acuse de recibo generados.

Escenario 1 Generación de carta de notificación de subsanación

- 1- El usuario accede al detalle de la subsanación.
- 2- El usuario selecciona los documentos a subsanar y pulsa generar notificación en papel.
- 3- El sistema genera el documento de notificación de subsanación.

Diagrama de secuencia:



6.22 Caso de uso Notificación de traslado de recurso (UC-010)

Denominación:

Este caso de uso engloba las acciones que realiza el sistema de información al generar y enviar el OARC la notificación de traslado de recurso a los interesados. Esta notificación informa a los interesados de que existe un recurso que afecta a una licitación en la que están envueltos.

Actor:

AC-001-OAR, AC-001-INT

Precondición:

Existe el expediente del recurso. Los datos del recurso y de los interesados están completos y correctos

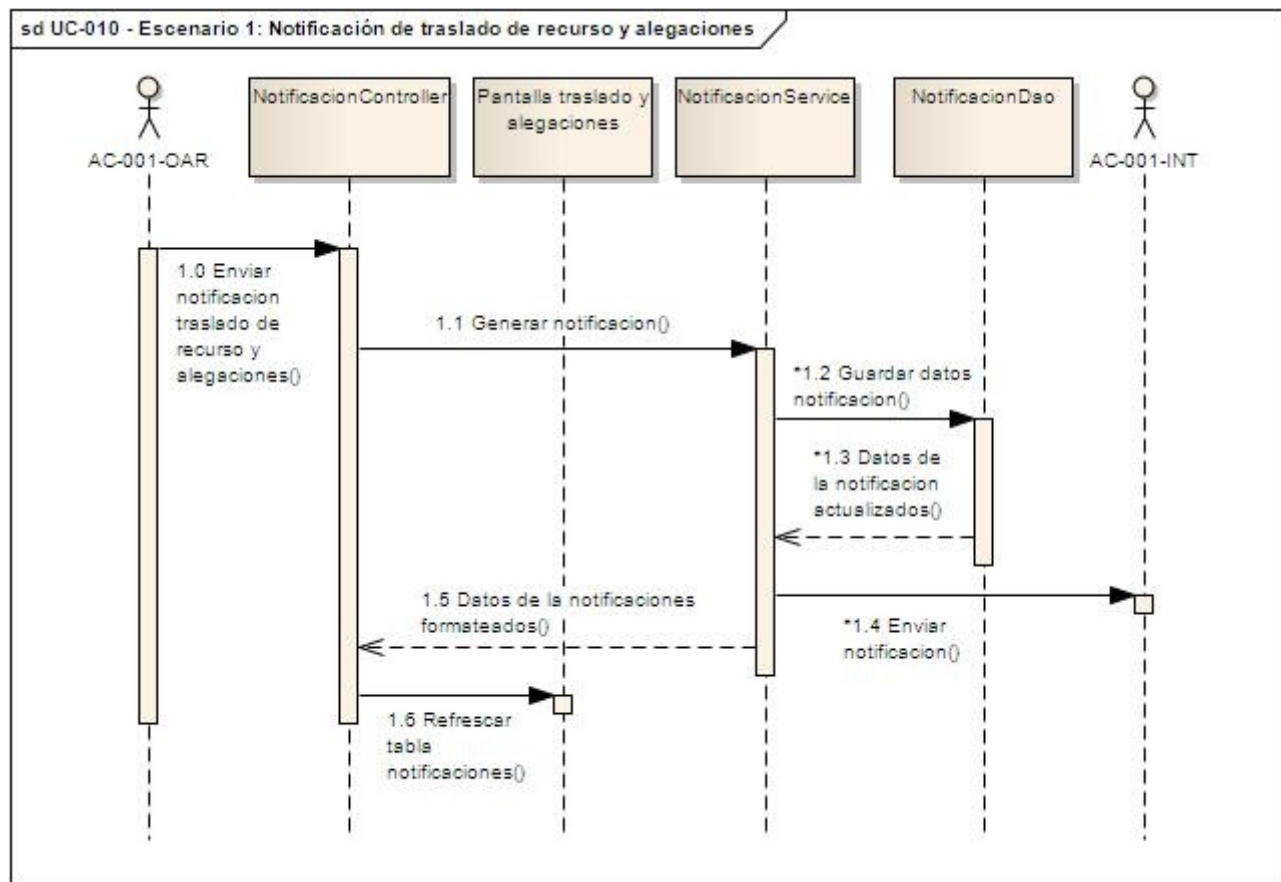
Postcondición:

Notificación enviada a los interesados.

Escenario 1: Notificación de traslado de recurso y alegaciones

- 1- El OARC accede al detalle de traslado de recurso.
- 2- El OARC pulsa el botón de notificación de traslado de recurso.
- 3- El sistema genera, envía la notificación y guarda los datos asociados.

Diagrama de secuencia:



6.23 Caso de uso Envío de alegaciones (UC-011)

Denominación:

Los interesados pueden enviar alegaciones al OARC si lo consideran oportuno. Cada interesado puede enviar o no sus alegaciones al OARC. El período de alegaciones se inicia una vez el interesado lee la notificación de traslado de recurso.

Actor:

AC-001-INT

Precondición:

Existe el expediente del recurso. El OARC ha enviado al recurrente la notificación de traslado de recurso.

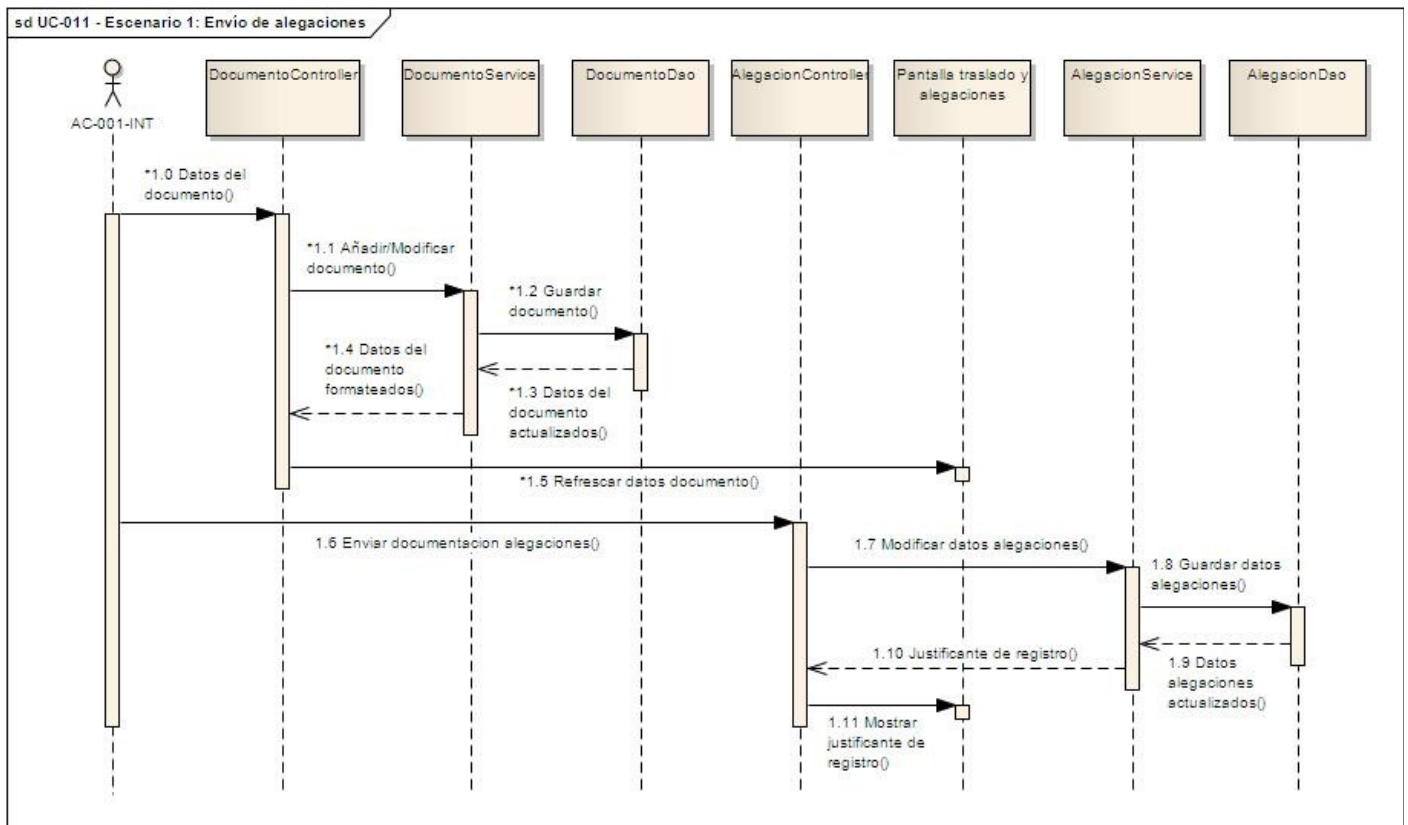
Postcondición:

Escenario 1: Envío de alegaciones

- 1- El interesado accede al detalle del recurso.
- 2- El recurrente adjunta y envía la documentación de alegaciones.

3- El sistema guarda los datos asociados y muestra un justificante de registro.

Diagrama de secuencia:



6.24 Caso de uso Mantenimiento de alegaciones y traslado de recurso (UC-012)

Denominación:

El OARC mantiene la información de traslado de recurso y alegaciones actualizándola en caso de recibir en papel la documentación o pudiendo modificar los datos existentes.

Actor:

AC-001-OAR

Precondición:

Existe el expediente del recurso.

Postcondición:

Datos de alegaciones y traslado de recurso actualizados en el sistema.

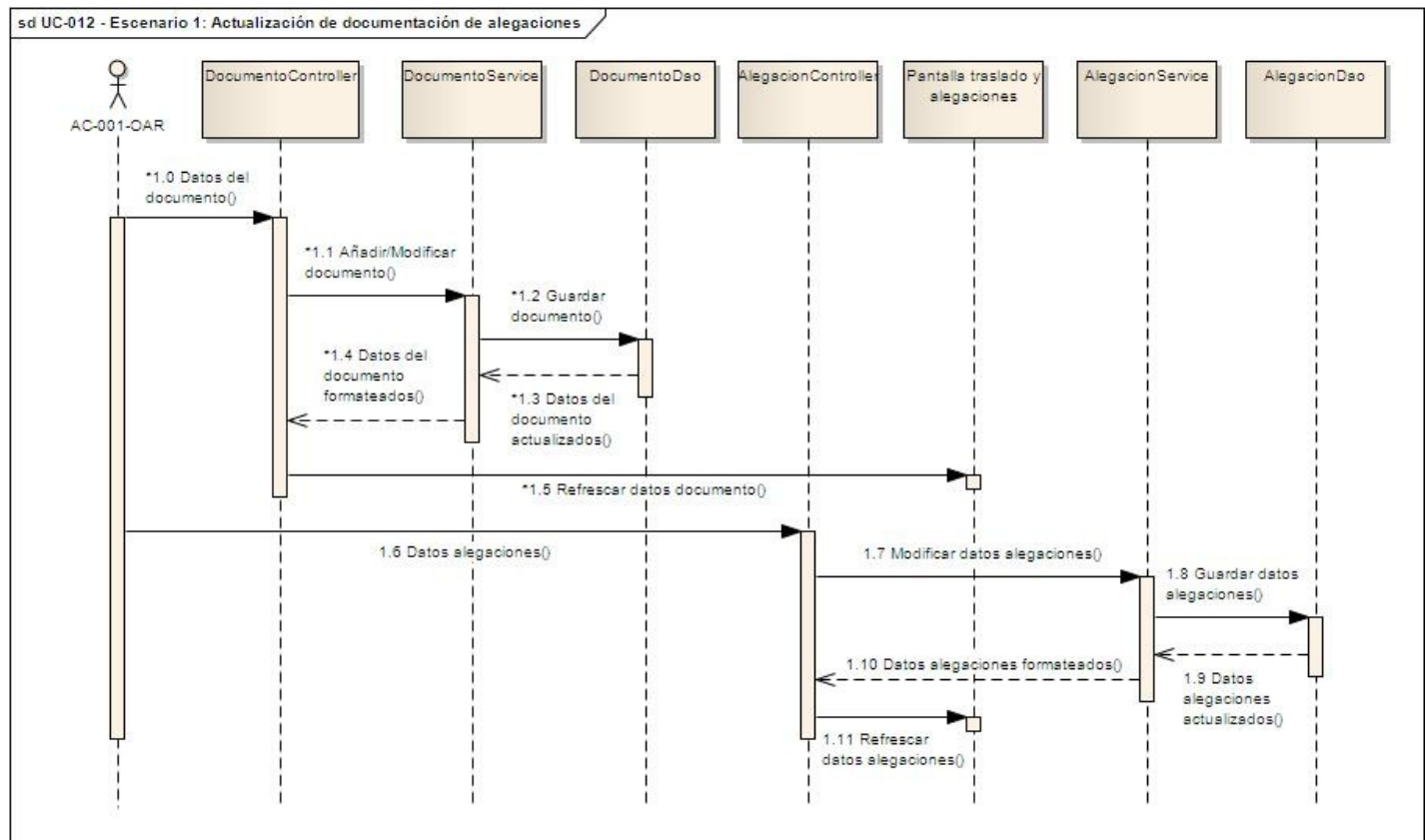
Escenario 1: Actualización de documentación de alegaciones

1- El OARC accede al detalle de traslado de recurso y alegaciones.

2- El OARC adjunta documentación de alegaciones.

3- El sistema guarda los datos asociados.

Diagrama de secuencia:



6.25 Caso de uso Mantenimiento de medidas cautelares (UC-013)

Denominación:

El OARC mantiene la información de medidas cautelares actualizándola en caso de recibir en papel la documentación o pudiendo modificar los datos existentes.

Actor:

AC-001-OAR

Precondición:

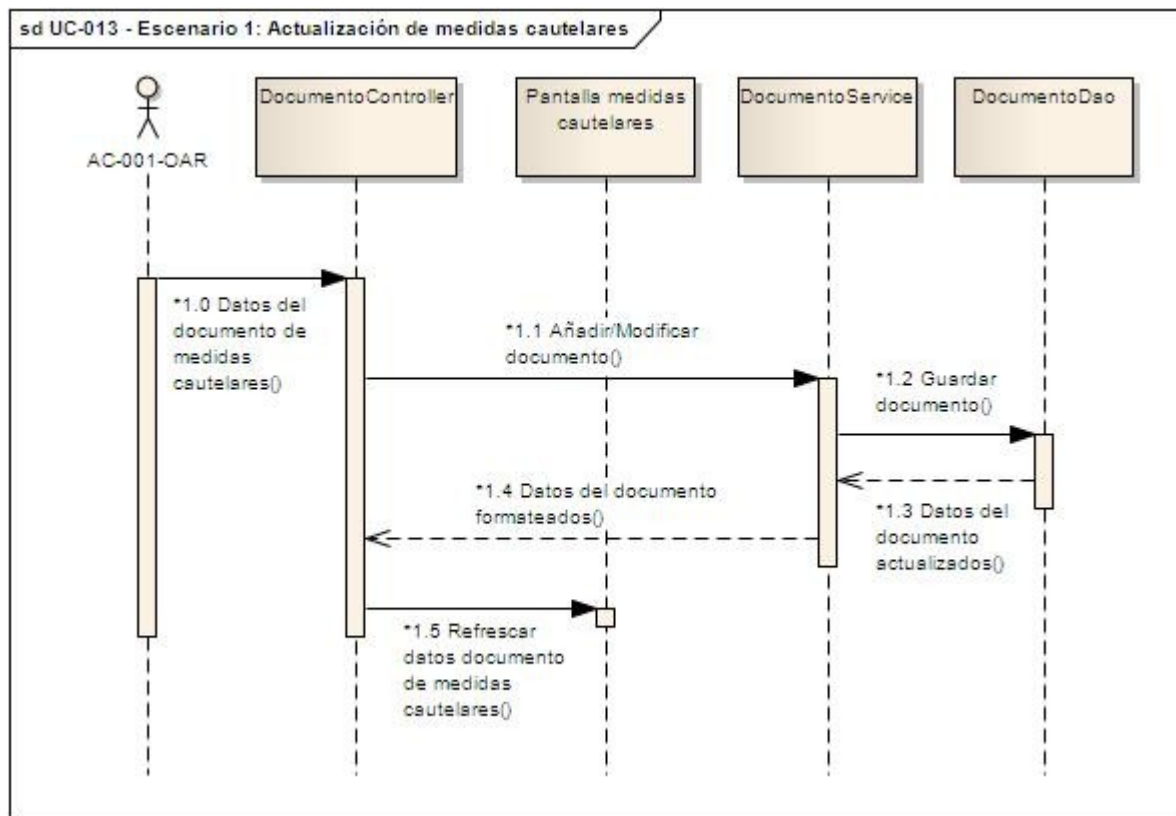
Postcondición:

Datos de medidas cautelares actualizados en el sistema.

Escenario 1: Actualización de medidas cautelares

- 1- El OARC accede al detalle de medidas cautelares.
- 2- El OARC actualiza la información de las medidas cautelares.
- 3- El sistema guarda los datos asociados.

Diagrama de secuencia:



6.26 Caso de uso Solicitud de medidas cautelares (UC-014)

Denominación:

El Órgano de contratación o el recurrente pueden solicitar medidas cautelares al OARC.

Actor:

AC-001-INT, AC-001-CON, AC-001-OAR

Precondición:

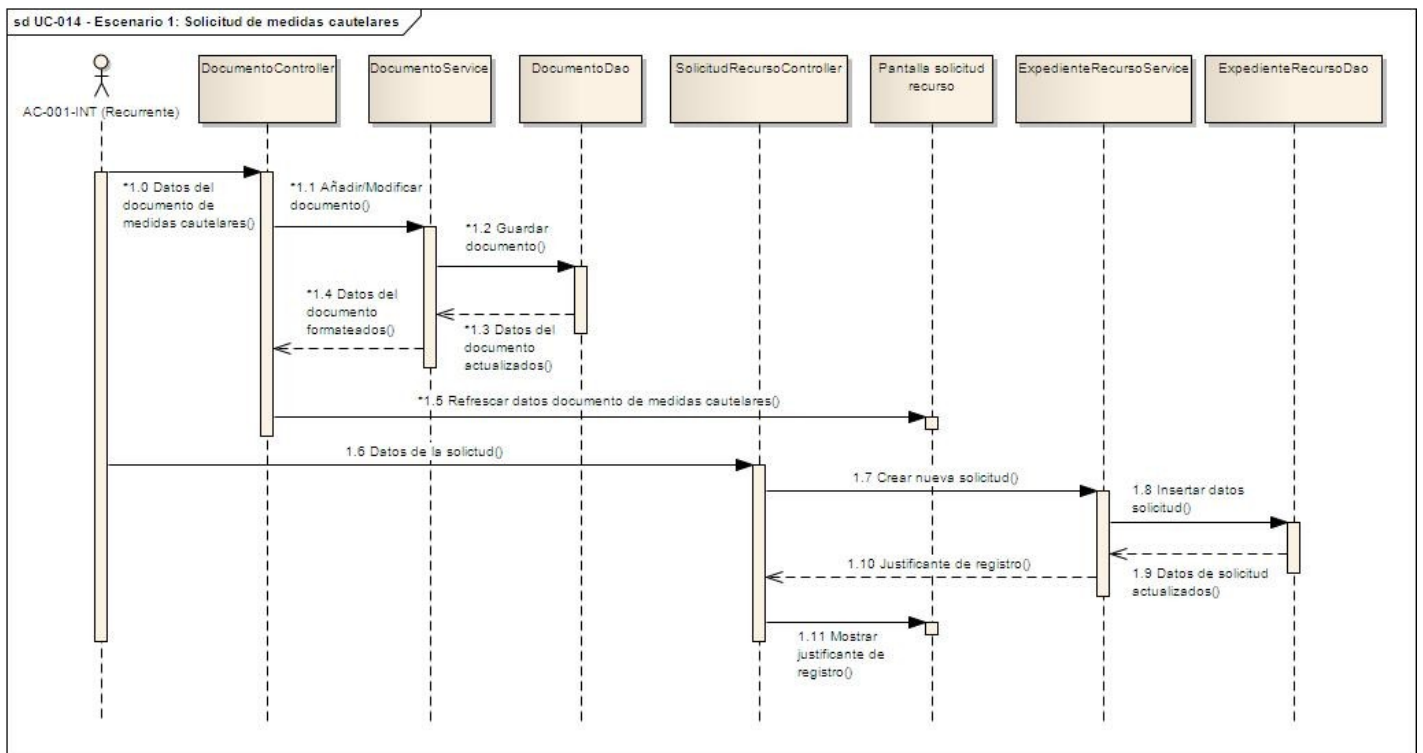
Postcondición:

Solicitud de medidas cautelares actualizadas en el sistema.

Escenario 1: Solicitud de medidas cautelares

- 1- El usuario accede a la solicitud de recurso.
- 2- El usuario rellena la información de las medidas cautelares y la envía.
- 3- El sistema guarda los datos asociados y muestra un justificante de registro.

Diagrama de secuencia:



6.27 Caso de uso Solicitud de práctica de prueba (UC-015)

Denominación:

El Órgano de contratación o el recurrente pueden solicitar la práctica de prueba al OARC.

Actor:

AC-001-INT, AC-001-CON, AC-001-OAR

Precondición:

Postcondición:

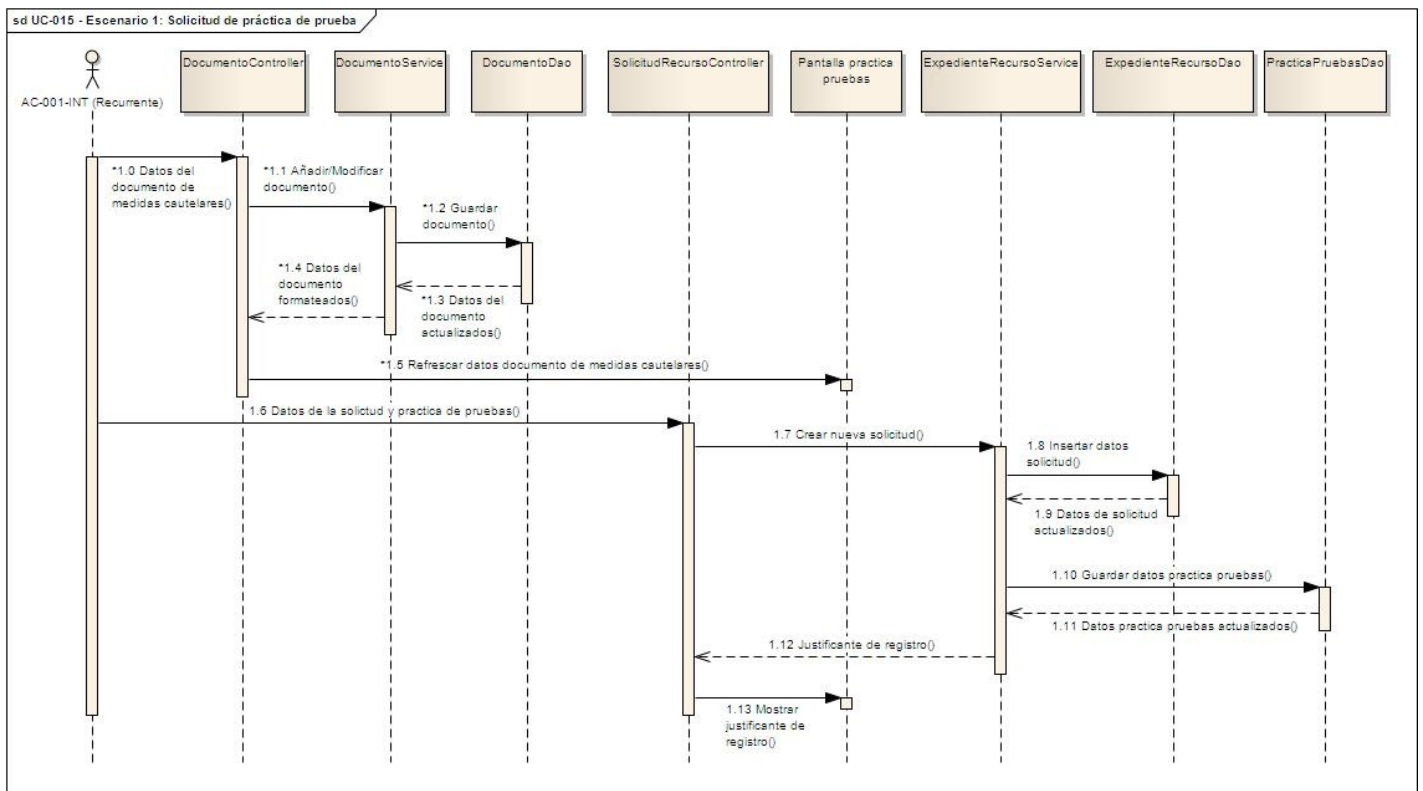
Solicitud de práctica de prueba actualizada en el sistema.

Escenario 1: Solicitud de práctica de prueba

- 1- El usuario accede a la solicitud de recurso.

- 2- El usuario rellena la información de práctica de prueba y la envía.
- 3- El sistema guarda los datos asociados y muestra un justificante de registro.

Diagrama de secuencia:



6.28 Caso de uso Notificación de resolución de medidas (UC-016)

Denominación:

Este caso de uso engloba las acciones que realiza el sistema de información al generar y enviar el OARC la notificación de resolución de medidas a los interesados y al Órgano de contratación. Esta notificación informa a los interesados y al Órgano de contratación de que se ha dictado una resolución para las medidas.

Actor:

AC-001-OAR, AC-001-INT, AC-001-CON

Precondición:

Existen medidas cautelares. Se ha emitido una resolución para las medidas.

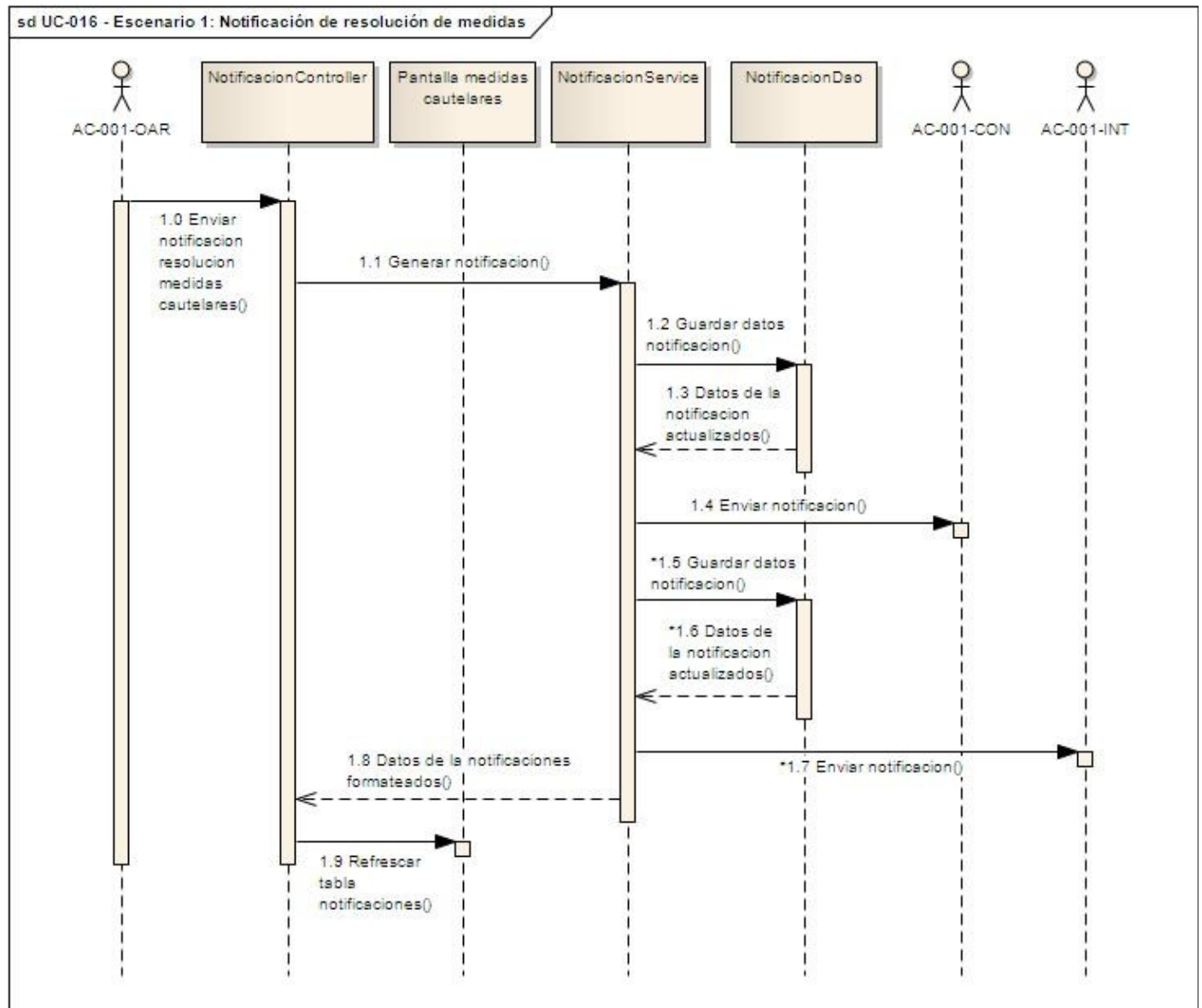
Postcondición:

Notificación enviada a los interesados y al Órgano de contratación.

Escenario 1: Notificación de resolución de medidas

- 1- El OARC accede al detalle de medidas provisionales.
- 2- El OARC pulsa el botón de notificación de resolución de medidas.
- 3- El sistema genera, envía la notificación y guarda los datos asociados.

Diagrama de secuencia:



6.29 Caso de uso Mantenimiento de práctica de prueba (UC-017)

Denominación:

El OARC mantiene la información de la práctica de prueba actualizándola en caso de recibir en papel la documentación o pudiendo modificar los datos existentes.

Actor:

AC-001-OAR

Precondición:

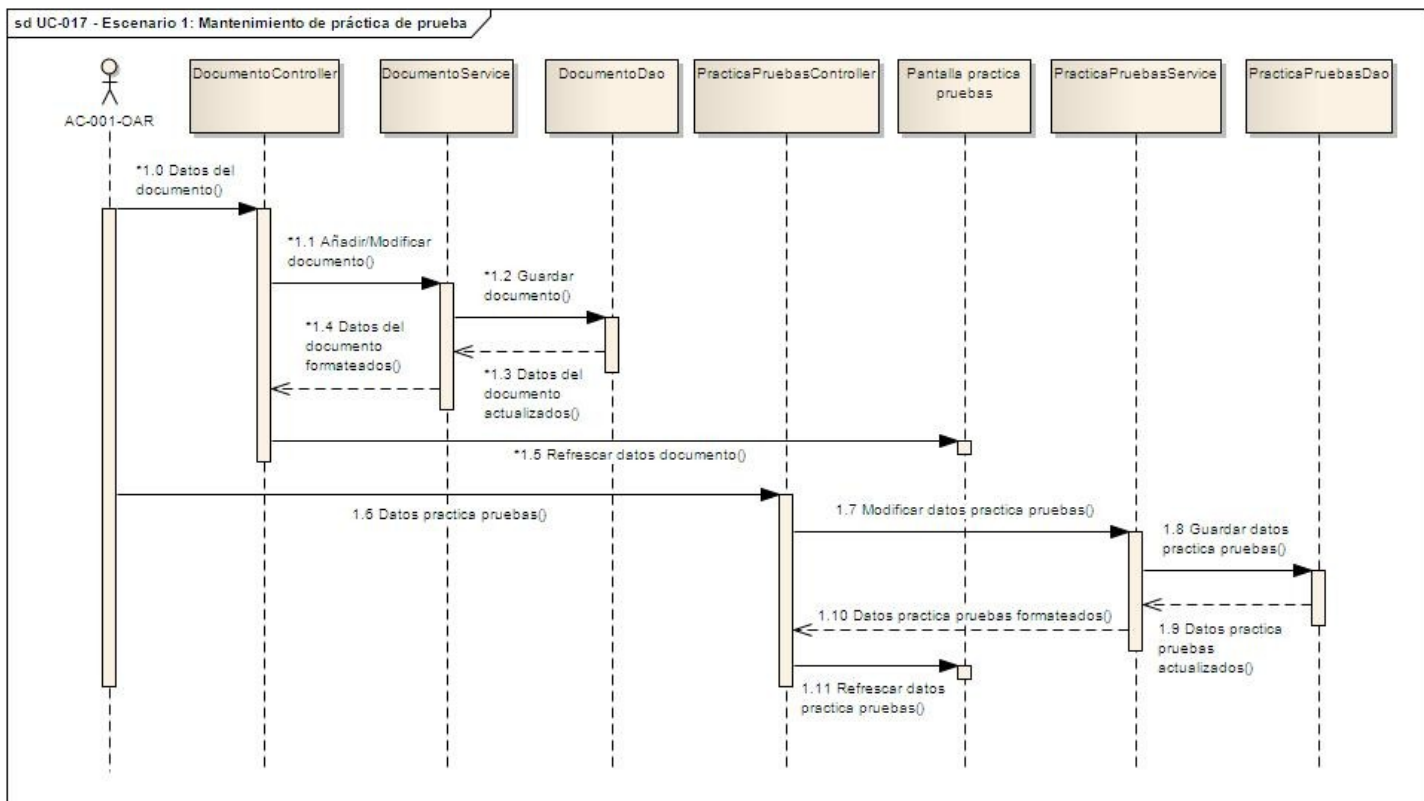
Postcondición:

Datos de práctica de prueba actualizados en el sistema.

Escenario 1: Mantenimiento de práctica de prueba

- 1- El OARC accede al detalle de práctica de prueba.
- 2- El OARC actualiza la información de la práctica de prueba y pulsa guardar.
- 3- El sistema guarda los datos asociados.

Diagrama de secuencia:



6.30 Caso de uso Notificación de práctica de prueba (UC-018)

Denominación:

Este caso de uso engloba las acciones que realiza el sistema de información al generar y enviar el OARC la notificación práctica de prueba a los interesados y al Órgano de contratación. Esta notificación informa a los interesados y al Órgano de contratación de que se ha dictado una resolución para la práctica de prueba.

Actor:

AC-001-OAR, AC-001-INT, AC-001-CON

Precondición:

Existe práctica de prueba. Se ha emitido una resolución para la práctica de prueba.

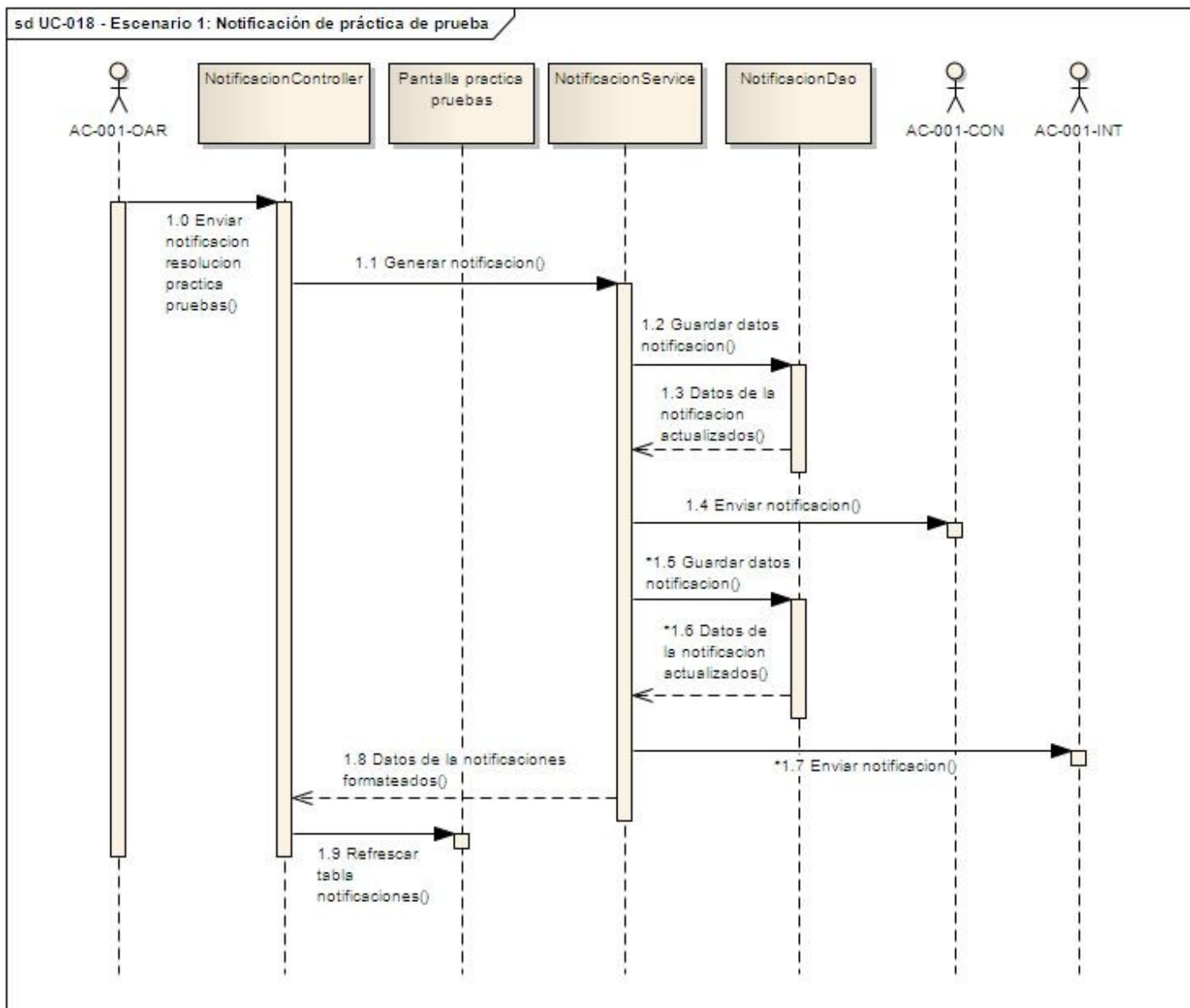
Postcondición:

Notificación enviada a los interesados y al Órgano de contratación.

Escenario 1: Notificación de práctica de prueba

- 1- El OARC accede al detalle de práctica de prueba.
- 2- El OARC pulsa el botón de notificación de resolución de práctica de prueba.
- 3- El sistema genera, envía la notificación y guarda los datos asociados.

Diagrama de secuencia:



6.31 Caso de uso Mantenimiento de resolución de recurso (UC-019)

Denominación:

El OARC mantiene la información de la resolución del recurso, adjuntando la resolución al expediente y su información asociada, junto con los links de la documentación legislativa de referencia en que se ha basado la resolución.

Actor:

AC-001-OAR

Precondición:

Expediente del recurso existente.

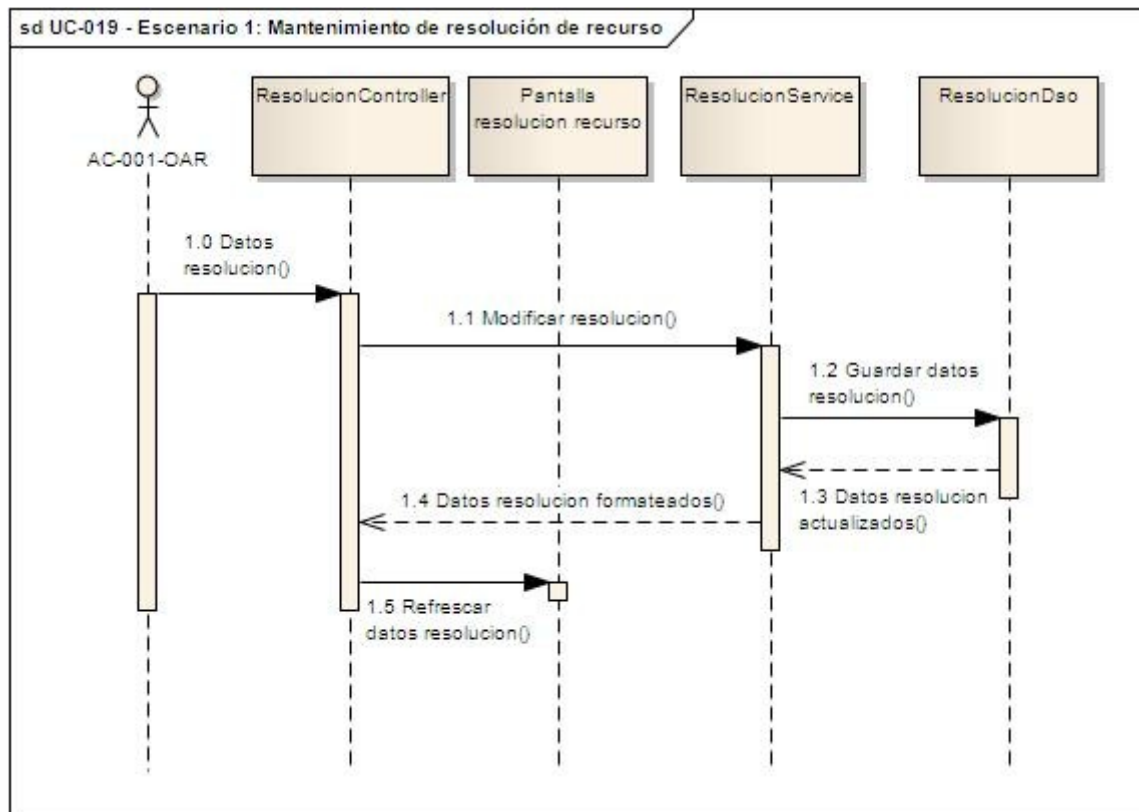
Postcondición:

Datos de la resolución del recurso actualizados en el sistema.

Escenario 1: Mantenimiento de resolución de recurso

- 1- El OARC accede al detalle de la resolución de recurso.
- 2- El OARC actualiza la información de la resolución de recurso y pulsa guardar.
- 3- El sistema guarda los datos asociados.

Diagrama de secuencia:



6.32 Caso de uso Notificación de resolución de recurso (UC-020)

Denominación:

Este caso de uso engloba las acciones que realiza el sistema de información al generar y enviar el OARC la notificación de resolución de recurso a los interesados. Esta notificación informa a los interesados de que se ha dictado la resolución del recurso.

Actor:

AC-001-OAR, AC-001-INT

Precondición:

Existe expediente de recurso y se ha adjuntado la resolución al expediente.

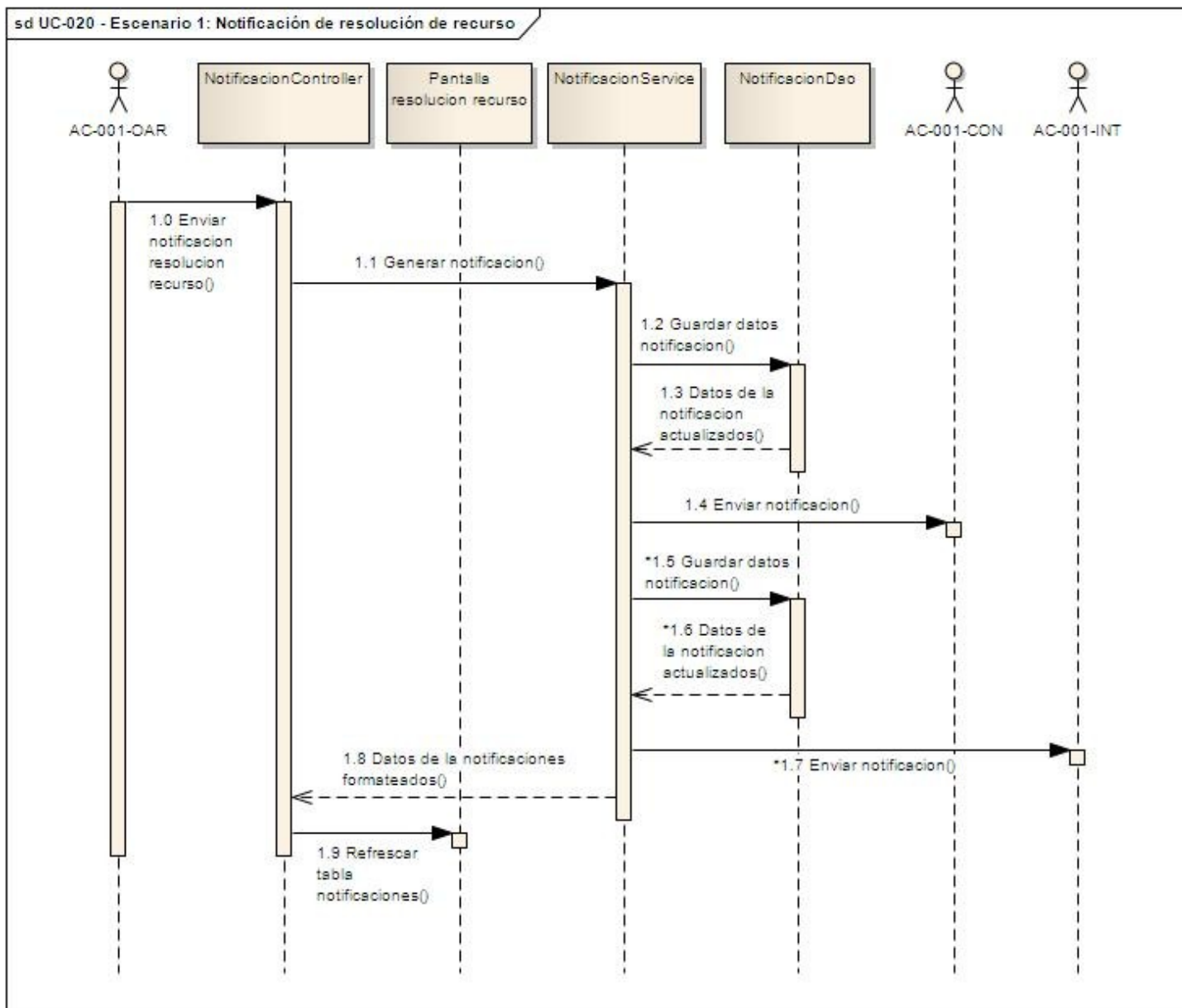
Postcondición:

Notificación enviada a los interesados.

Escenario 1: Notificación de resolución de recurso

- 1- El OARC accede al detalle de resolución de recurso.
- 2- El OARC pulsa el botón de notificación de resolución.
- 3- El sistema genera, envía la notificación y guarda los datos asociados.

Diagrama de secuencia:



6.33 Caso de uso Calendario histórico de expediente (UC-021)

Denominación:

Este caso de uso engloba las acciones que realiza el sistema de información para generar la vista calendario de un expediente, mostrando los eventos que han ocurrido en la vida del expediente de recurso.

Actor:

AC-001-OAR

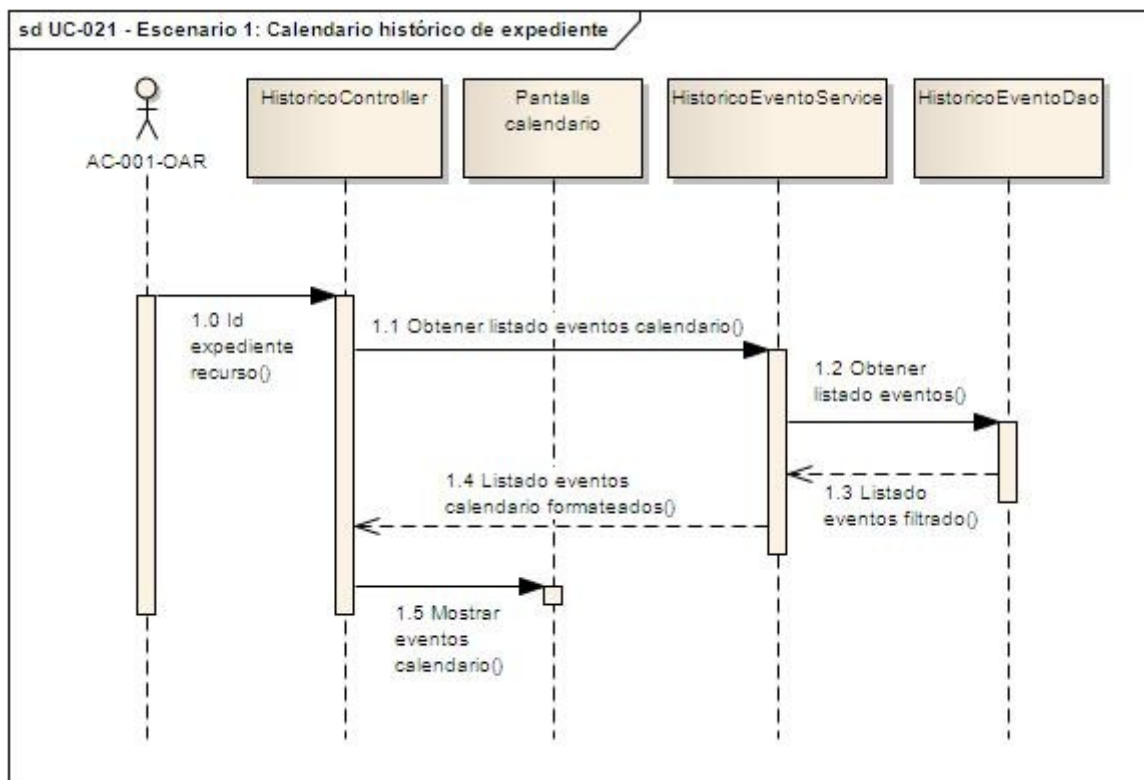
Precondición:

Postcondición:

Visualización del histórico de eventos de un expediente en un formato de calendario.

Escenario 1: Calendario histórico de expediente

1- El OARC accede a la vista de calendario dentro de un expediente.

Diagrama de secuencia:**6.34 Caso de uso Bitácora de expediente (UC-022)****Denominación:**

Este caso de uso engloba las acciones que realiza el sistema de información para generar la vista de bitácora en forma de listado de eventos de un expediente, mostrando los eventos que han ocurrido en la vida del expediente de recurso.

Actor:

AC-001-OAR

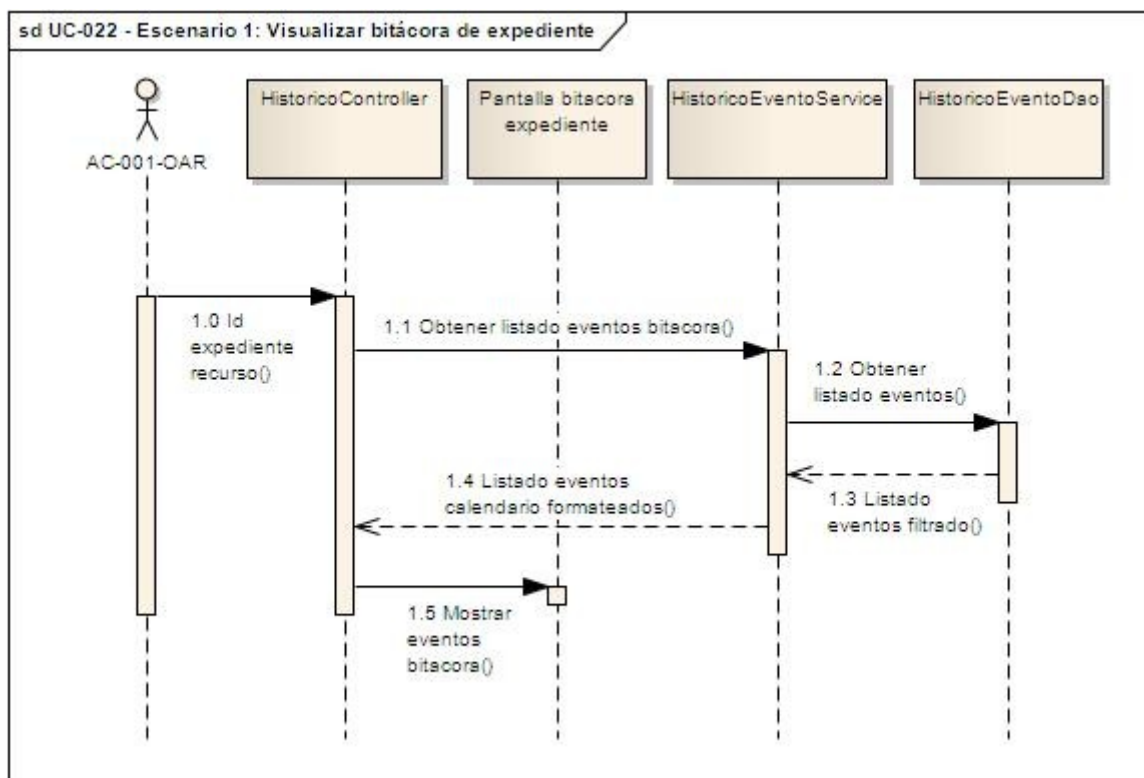
Precondición:**Postcondición:**

Visualización de la bitácora de eventos de un expediente en forma de listado.

Escenario 1: Visualizar bitácora de expediente

1- El OARC accede a la vista de bitácora dentro de un expediente.

Diagrama de secuencia:



6.35 Caso de uso Bandeja de entrada (UC-023)

Denominación:

Este caso de uso engloba las acciones que realiza el sistema de información para mostrar los eventos de entrada que se producen sobre un expediente. Permite el filtrado de información por una serie de criterios y el archivo de ciertas entradas que se considere que no hace falta que aparezcan en los resultados de la búsqueda. Permite acceder al detalle de expediente.

Actor:

AC-001-OAR

Precondición:

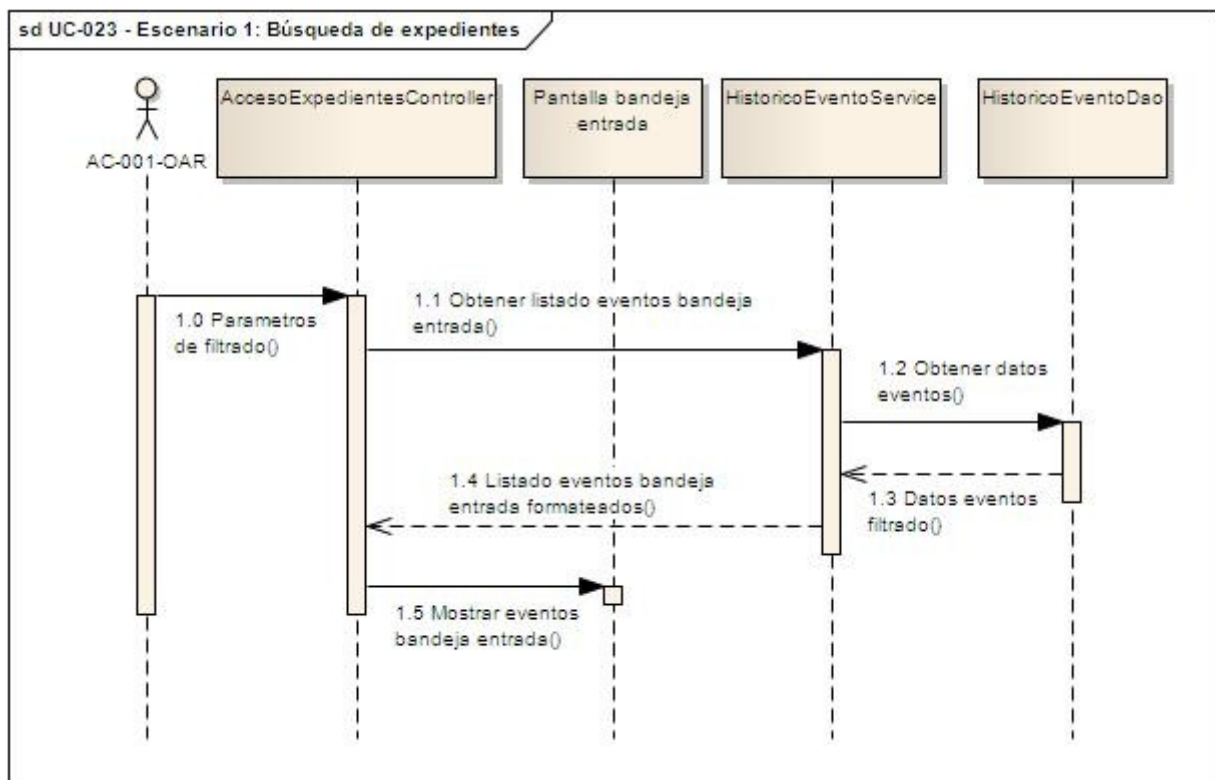
Postcondición:

Visualización de eventos de entrada que cumplan los filtros.

Escenario 1: Búsqueda de expedientes

- 1- El OARC accede al formulario de bandeja de entrada.
- 2- El OARC rellena los filtros oportunos y pulsa buscar.
- 3- El sistema devuelve el listado de eventos de entrada que cumple los filtros seleccionados.

Diagrama de secuencia:



6.36 Caso de uso Bandeja de salida (UC-024)

Denominación:

Este caso de uso engloba las acciones que realiza el sistema de información para mostrar los eventos de salida que se producen sobre un expediente. Permite el filtrado de información por una serie de criterios y el archivo de ciertos eventos de salida que se considere que no hace falta que aparezcan en los resultados de la búsqueda. Permite acceder al detalle de expediente.

Actor:

AC-001-OAR

Precondición:

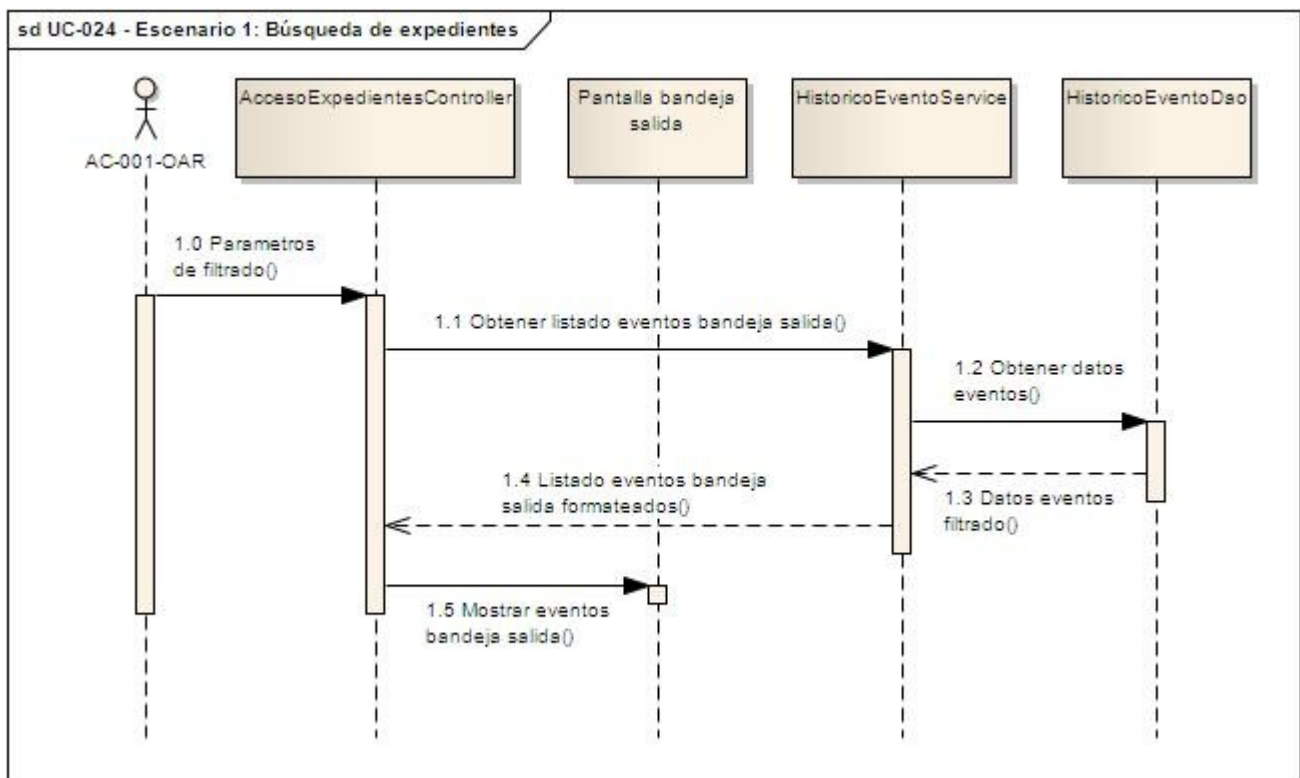
Postcondición:

Visualización de los eventos de salida que cumplan los filtros.

Escenario 1: Búsqueda de expedientes

- 1- El OARC accede al formulario de bandeja de salida.
- 2- El OARC rellena los filtros oportunos y pulsa buscar.
- 3- El sistema devuelve el listado de eventos de salida que cumplan los filtros seleccionados.

Diagrama de secuencia:



6.37 Caso de uso Bandeja de tareas del OARC (UC-025)

Denominación:

Este caso de uso engloba las acciones que realiza el sistema de información para mostrar los plazos de los trámites llevados a cabo en los expedientes en formato calendario. Permite acceder al detalle de expediente.

Actor:

AC-001-OAR

Precondición:

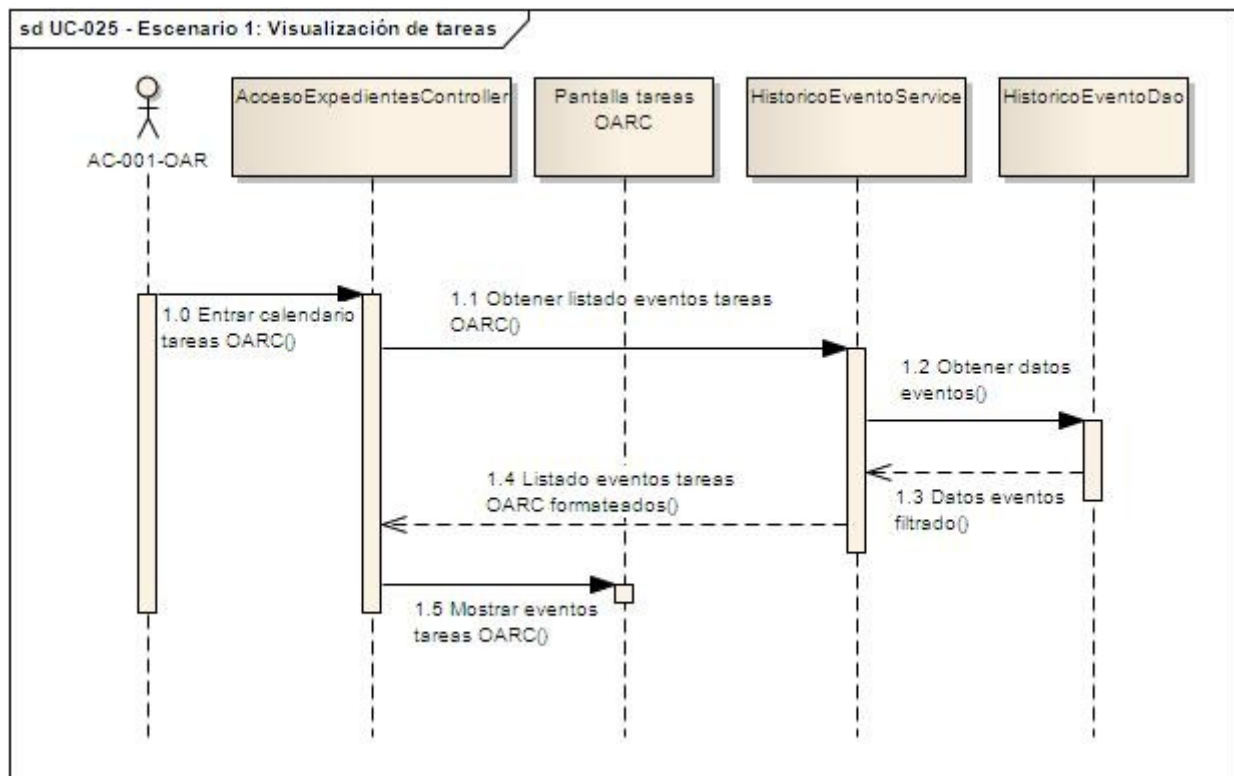
Postcondición:

Visualización de plazos en un calendario para la realización de trámites.

Escenario 1: Visualización de tareas

1- El OARC accede a la bandeja de tareas del OARC.

Diagrama de secuencia:



6.38 Caso de uso Visualizar últimas resoluciones (UC-026)

Denominación:

Este caso de uso engloba las acciones que realiza el sistema de información para mostrar las 10 últimas resoluciones en el portal de contratación de Euskadi. El acceso a esta información es público y no hace falta identificación del sujeto que accede.

Actor:

AC-001-PUB

Precondición:

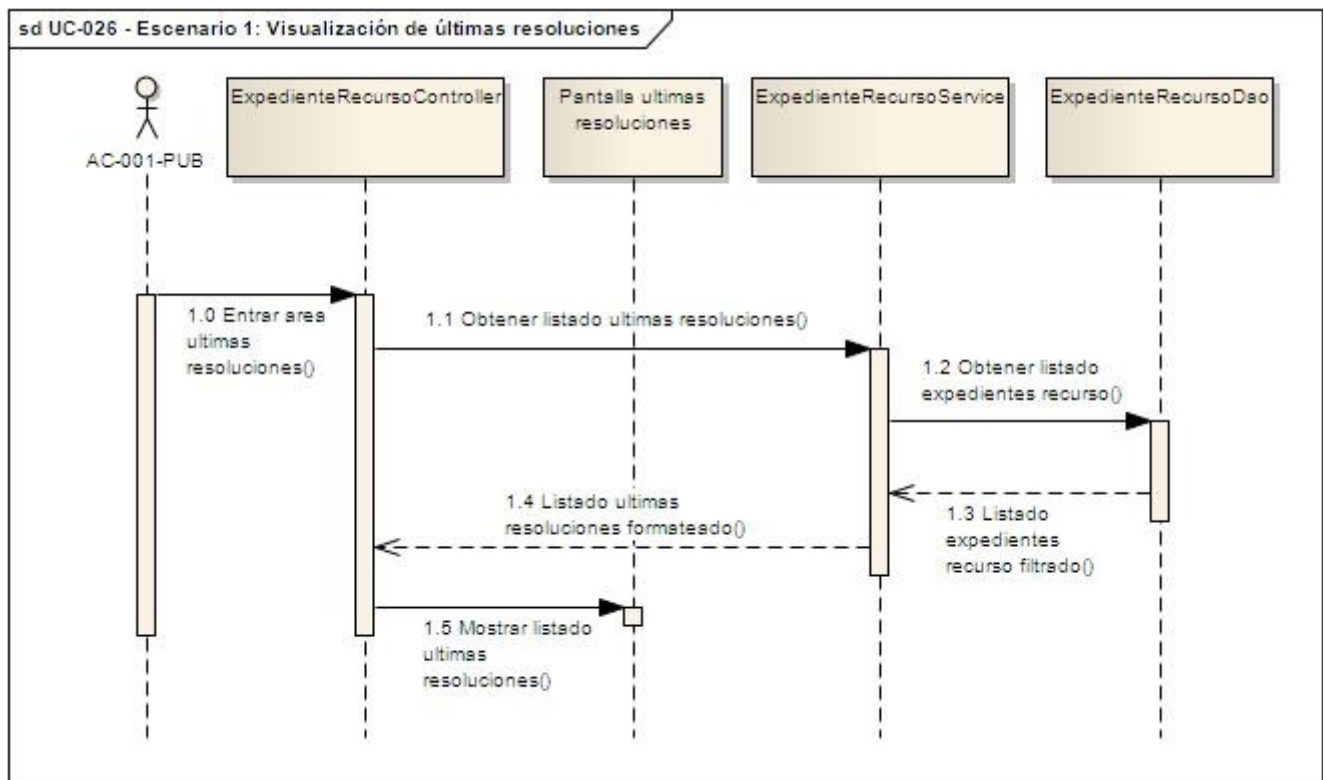
Postcondición:

Visualización de las últimas resoluciones.

Escenario 1: Visualización de últimas resoluciones

1- El usuario público accede al área de últimas resoluciones del Portal.

Diagrama de secuencia:



6.39 Caso de uso Buscador de resoluciones (UC-027)

Denominación:

Dentro del portal de contratación de Euskadi se proporcionará un buscador de resoluciones de recursos con una serie de filtros para acotar los resultados. El acceso a esta información es público y no hace falta identificación del sujeto que accede. Se puede acceder a las resoluciones emitidas por el OARC.

Actor:

AC-001-PUB

Precondición:

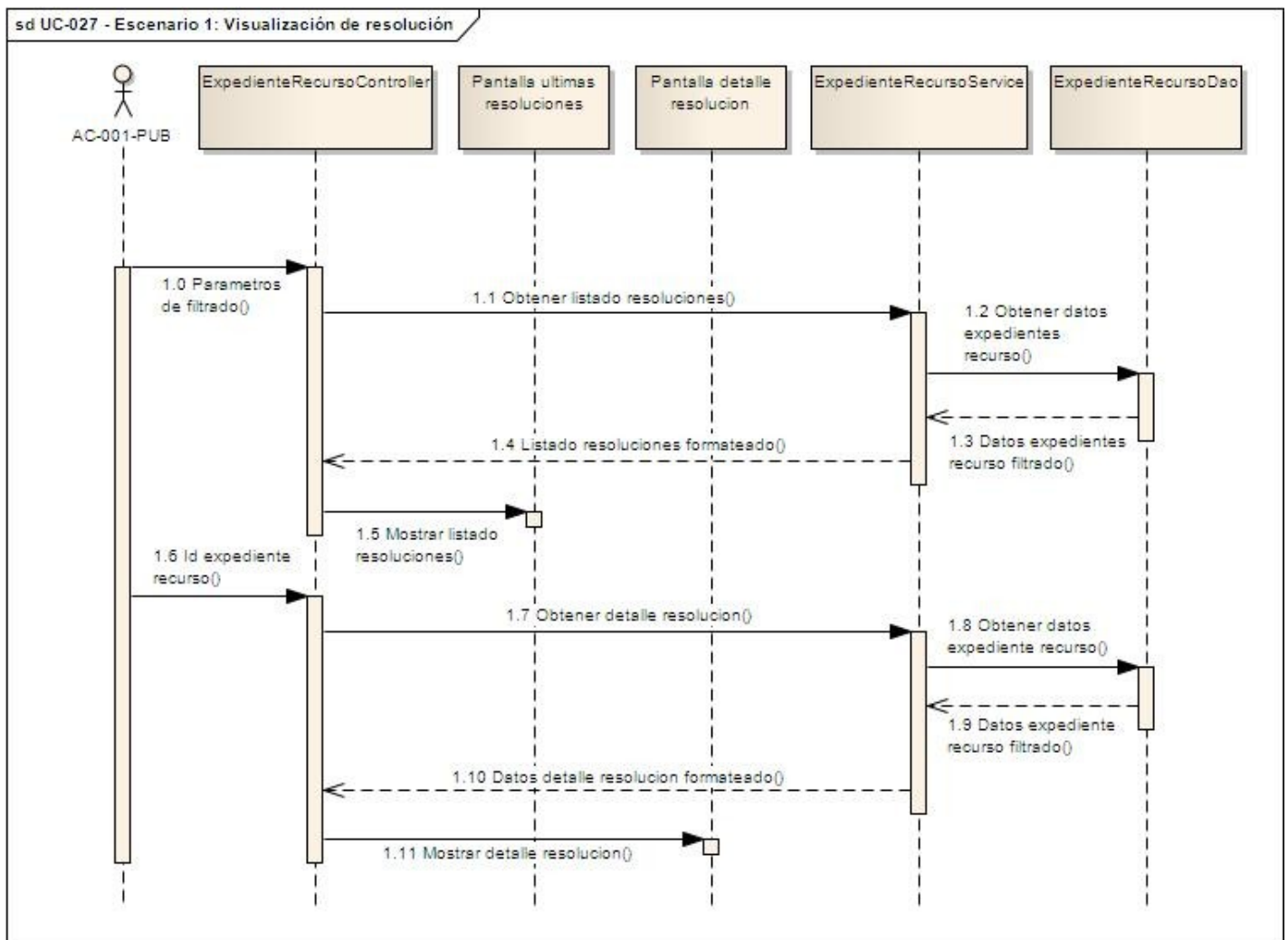
Postcondición:

Visualización de resoluciones.

Escenario 1: Visualización de resolución

- 1- El usuario público accede al buscador de resoluciones de recursos del Portal y pulsa buscar.
- 2- El sistema devuelve las resoluciones que cumplen los filtros
- 3- El usuario accede al detalle de la resolución.

Diagrama de secuencia:



6.40 Caso de uso Generación de informes (UC-028)

Denominación:

El usuario del OARC podrá generar una serie de informes. Estos informes tendrán un formulario con unos filtros para acotar los resultados y se generarán en formato Excel.

Actor:

AC-001-OAR

Precondición:

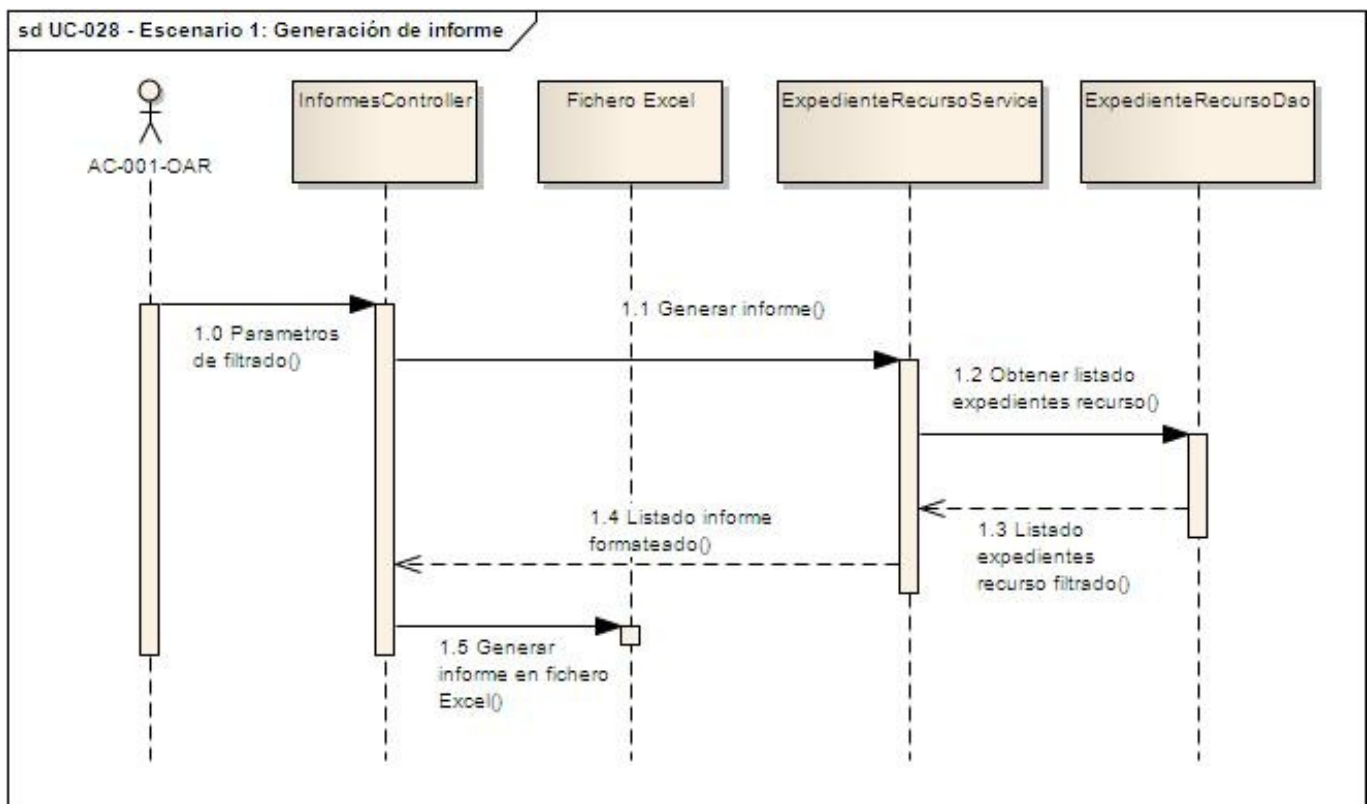
Postcondición:

Generación de informe en formato Excel.

Escenario 1: Generación de informe

- 1- El usuario accede a los filtros de un informe los selecciona y pulsa generar documento.
- 2- El sistema devuelve un documento Excel con los resultados.

Diagrama de secuencia:



7 Modelo de clases de diseño

7.1 Subsistema 1 Solicitud de recurso

class Subsistema 001 - Solicitud de recurso

ExpedienteRecurso
<ul style="list-style-type: none"> actoRecurrido: Documento anuncioPrevioExp: Documento escontoRecurso: Documento idAmbito: int idEntidadImpulsora: long idExpedienteRecurso: long idPoderAdjudicador: long idTipoActo: int idTipoProcedimiento: int interesado: Interesado medidasCautelares: boolean numExpedienteContratacion: String objetoContrato: String otrosDocumentos: List poderPresCompartiente: Documento practicaPruebas: boolean
<ul style="list-style-type: none"> «property gets» + getActoRecurrido(): Documento + getAnuncioPrevioExp(): int + getEscontoRecurso(): Documento + getIdAmbito(): int + getIdEntidadImpulsora(): long + getIdExpedienteRecurso(): long + getIdPoderAdjudicador(): long + getIdTipoActo(): int + getIdTipoProcedimiento(): int + getInteresado(): Interesado + getNumExpedienteContratacion(): String + getObjetoContrato(): String + getOtrosDocumentos(): List + getIdPresCompartiente(): Documento + getIdMedidasCautelares(): boolean + getIdPracticaPruebas(): boolean «property sets» + setActoRecurrido(Documento): void + setAnuncioPrevioExp(int): void + setEscontoRecurso(Documento): void + setIdAmbito(int): void + setIdEntidadImpulsora(long): void + setIdExpedienteRecurso(long): void + setIdPoderAdjudicador(long): void + setIdTipoActo(int): void + setIdTipoProcedimiento(int): void + setInteresado(Interesado): void + setIdMedidasCautelares(boolean): void + setNumExpedienteContratacion(String): void + setObjetoContrato(String): void + setIdPresCompartiente(Documento): void + setIdPracticaPruebas(boolean): void

SolicitudRecursoController
<ul style="list-style-type: none"> expedienteRecursoService: ExpedienteRecursoService historicoEventoService: HistoricoEventoService
<ul style="list-style-type: none"> + add(ExpedienteRecurso): ExpedienteRecurso + edit(ExpedienteRecurso): ExpedienteRecurso + getById(Long): ExpedienteRecurso + getCreateForm(Model): String «property gets» + getExpedienteRecursoService(): ExpedienteRecursoService + getHistoricoEventoService(): HistoricoEventoService «property sets» + setExpedienteRecursoService(ExpedienteRecursoService): void + setHistoricoEventoService(HistoricoEventoService): void

ExpedienteRecursoService
<ul style="list-style-type: none"> expedienteRecursoDao: ExpedienteRecursoDao
<ul style="list-style-type: none"> + add(ExpedienteRecurso): ExpedienteRecurso + find(ExpedienteRecurso): ExpedienteRecurso + findAll(Pagination, ExpedienteRecurso): List + findAllCount(ExpedienteRecurso): Long + findAllLike(boolean, Pagination, ExpedienteRecurso): List + findAllLikeCount(boolean, ExpedienteRecurso): Long + remove(ExpedienteRecurso): void + removeMultiple(List): void + update(ExpedienteRecurso): ExpedienteRecurso «property gets» + getExpedienteRecursoDao(): ExpedienteRecursoDao «property sets» + setExpedienteRecursoDao(ExpedienteRecursoDao): void

ExpedienteRecursoDao
<ul style="list-style-type: none"> + add(ExpedienteRecurso): ExpedienteRecurso + find(ExpedienteRecurso): ExpedienteRecurso + findAll(Pagination, ExpedienteRecurso): List + findAllCount(ExpedienteRecurso): Long + findAllLike(boolean, Pagination, ExpedienteRecurso): List + findAllLikeCount(boolean, ExpedienteRecurso): Long + generarNumeroExpediente(ExpedienteRecurso): ExpedienteRecurso + remove(ExpedienteRecurso): void + update(ExpedienteRecurso): ExpedienteRecurso

Interesado
<ul style="list-style-type: none"> adjudicador: boolean cif: String cpRepresentante: int denominacionSocial: String direccionRepresentante: String dniRepresentante: String idExpedienteRecurso: long idMunicipioRepresentante: long idProvinciaRepresentante: long mailRepresentante: String medioPostal: boolean nombreRepresentante: String recurrente: boolean telefonoRepresentante: String
<ul style="list-style-type: none"> «property gets» + getAdjudicador(): boolean + getCif(): String + getCpRepresentante(): int + getDenominacionSocial(): String + getDireccionRepresentante(): String + getDniRepresentante(): String + getIdExpedienteRecurso(): long + getIdMunicipioRepresentante(): long + getIdProvinciaRepresentante(): long + getMailRepresentante(): String + getNombreRepresentante(): String + getTelefonoRepresentante(): String + getIsadjudicador(): boolean + getIsmedioPostal(): boolean + getIsrecurrente(): boolean «property sets» + setAdjudicador(boolean): void + setCif(String): void + setCpRepresentante(int): void + setDenominacionSocial(String): void + setDireccionRepresentante(String): void + setDniRepresentante(String): void + setIdExpedienteRecurso(long): void + setIdMunicipioRepresentante(long): void + setIdProvinciaRepresentante(long): void + setMailRepresentante(String): void + setMedioPostal(boolean): void + setNombreRepresentante(String): void + setRecurrente(boolean): void + setTelefonoRepresentante(String): void

InteresadoController
<ul style="list-style-type: none"> interesadoService: InteresadoService
<ul style="list-style-type: none"> + add(Interesado): Interesado + edit(Interesado): Interesado + getAll(Pagination, Interesado): List + getById(Long): Documento + remove(Long): Interesado + removeMultiple(List): List «property gets» + getInteresadoService(): InteresadoService «property sets» + setInteresadoService(InteresadoService): void

InteresadoService
<ul style="list-style-type: none"> interesadoDao: InteresadoDao
<ul style="list-style-type: none"> + add(Interesado): Interesado + find(Interesado): Interesado + findAll(Pagination, Interesado): List + findAllCount(Interesado): Long + remove(Interesado): void + removeMultiple(List): void + update(Interesado): Documento «property gets» + getInteresadoDao(): InteresadoDao «property sets» + setInteresadoDao(InteresadoDao): void

InteresadoDao
<ul style="list-style-type: none"> + add(Interesado): Interesado + find(Interesado): Interesado + findAll(Pagination, Interesado): List + findAllCount(Interesado): Long + remove(Interesado): void + removeMultiple(List): void + update(Interesado): Interesado

HistoricoEvento
<ul style="list-style-type: none"> colorCalendario: String colorPlazo: String descripcionEventoEs: String descripcionEventoEu: String descripcionPlazoEs: String descripcionPlazoEu: String flagCalendario: boolean flagEntrada: boolean flagSalida: boolean idEvento: int idExpedienteRecurso: long idHistoricoEvento: long idInteresado: long idPlazo: int
<ul style="list-style-type: none"> «property gets» + getColorCalendario(): String + getColorPlazo(): String + getDescripcionEventoEs(): String + getDescripcionEventoEu(): String + getDescripcionPlazoEs(): String + getDescripcionPlazoEu(): String + getIdEvento(): long + getIdExpedienteRecurso(): long + getIdHistoricoEvento(): long + getIdInteresado(): long + getIdPlazo(): int «property sets» + setColorCalendario(String): void + setColorPlazo(String): void + setDescripcionEventoEs(String): void + setDescripcionEventoEu(String): void + setDescripcionPlazoEs(String): void + setDescripcionPlazoEu(String): void + setFlagCalendario(boolean): void + setFlagEntrada(boolean): void + setFlagSalida(boolean): void + setIdEvento(long): void + setIdExpedienteRecurso(long): void + setIdHistoricoEvento(long): void + setIdInteresado(long): void + setIdPlazo(int): void

HistoricoEventoService
<ul style="list-style-type: none"> historicoEventoDao: HistoricoEventoDao
<ul style="list-style-type: none"> + add(HistoricoEvento): HistoricoEvento + find(HistoricoEvento): HistoricoEvento + findAll(Pagination, HistoricoEvento): List + findAllCount(HistoricoEvento): Long + findAllLike(boolean, Pagination, HistoricoEvento): List + findAllLikeCount(boolean, HistoricoEvento): Long + remove(HistoricoEvento): void + removeMultiple(List): void + update(HistoricoEvento): HistoricoEvento «property gets» + getHistoricoEventoDao(): HistoricoEventoDao «property sets» + setHistoricoEventoDao(HistoricoEventoDao): void

HistoricoEventoDao
<ul style="list-style-type: none"> + add(HistoricoEvento): HistoricoEvento + find(HistoricoEvento): HistoricoEvento + findAll(Pagination, HistoricoEvento): List + findAllCount(HistoricoEvento): Long + findAllLike(boolean, Pagination, HistoricoEvento): List + findAllLikeCount(boolean, HistoricoEvento): Long + generarNumeroExpediente(HistoricoEvento): HistoricoEvento + remove(HistoricoEvento): void + update(HistoricoEvento): HistoricoEvento

Documento
<ul style="list-style-type: none"> data: byte[] deleteType: String fecha: Date idDocumento: long idExpedienteRecurso: long idTipoDoc: int mime: String nombre: String size: long uri: String uriDelete: String
<ul style="list-style-type: none"> «property gets» + getData(): byte[] + getDeleteType(): String + getFecha(): Date + getIdDocumento(): long + getIdExpedienteRecurso(): long + getIdTipoDoc(): int + getMimeType(): String + getNombre(): String + getSize(): long + getUri(): String + getUriDelete(): String «property sets» + setData(byte[]): void + setDeleteType(String): void + setFecha(Date): void + setIdDocumento(long): void + setIdExpedienteRecurso(long): void + setIdTipoDoc(int): void + setMimeType(String): void + setNombre(String): void + setSize(long): void + setUri(String): void + setUriDelete(String): void

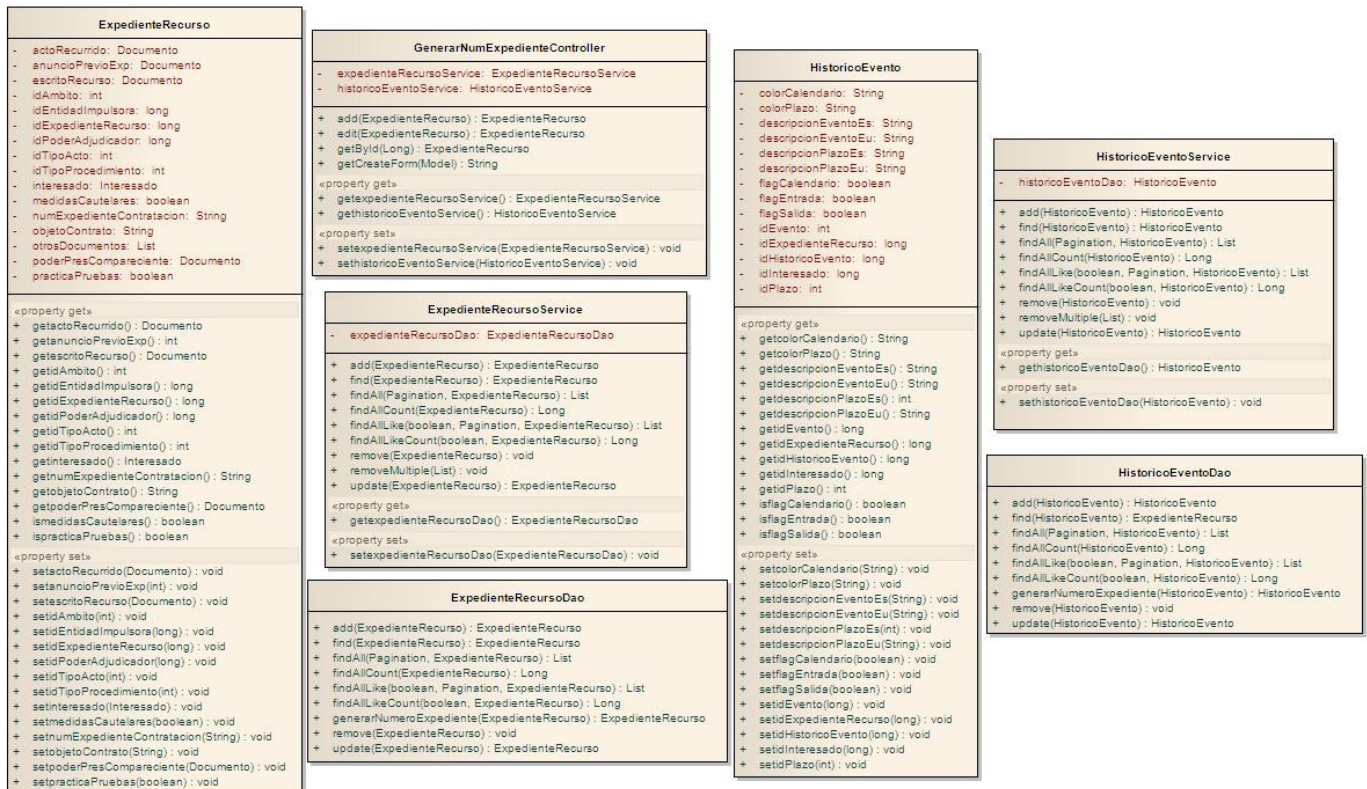
DocumentoController
<ul style="list-style-type: none"> documentoService: DocumentoService
<ul style="list-style-type: none"> + add(Documento): Documento + edit(Documento): Documento + getAll(Pagination, Documento): List + getById(Long): Documento + remove(Long): Documento + removeMultiple(List): List «property gets» + getDocumentoService(): DocumentoService «property sets» + setDocumentoService(DocumentoService): void

DocumentoService
<ul style="list-style-type: none"> documentoDao: DocumentoDao
<ul style="list-style-type: none"> + add(Documento): Documento + find(Documento): Documento + findAll(Pagination, Documento): List + findAllCount(Documento): Long + remove(Documento): void + removeMultiple(List): void + update(Documento): Documento «property gets» + getDocumentoDao(): DocumentoDao «property sets» + setDocumentoDao(DocumentoDao): void

DocumentoDao
<ul style="list-style-type: none"> + add(Documento): Documento + find(Documento): Documento + findAll(Pagination, Documento): List + findAllCount(Documento): Long + remove(Documento): void + removeMultiple(List): void + update(Documento): Documento

7.2 Subsistema 2 Generación de expediente de recurso

class Subsistema 002 - Generación de expediente de recurso



7.3 Subsistema 3 Expediente de contratación

class Subsistema 003 - Expediente de contratación

ExpedienteContratacion
<ul style="list-style-type: none"> - fechaEnvioComoSolicitud: Date - idExpedienteRecurso: long - numExpedienteContratacion: String - objetoContrato: String
<ul style="list-style-type: none"> «property gets» + getFechaEnvioComoSolicitud(): Date + getIdExpedienteRecurso(): long + getNumExpedienteContratacion(): String + getObjetoContrato(): String
<ul style="list-style-type: none"> «property sets» + setFechaEnvioComoSolicitud(Date): void + setIdExpedienteRecurso(long): void + setNumExpedienteContratacion(String): void + setObjetoContrato(String): void

ExpedienteContratacionController
<ul style="list-style-type: none"> - expedienteContratacionService: ExpedienteContratacionService - expedienteRecursoService: ExpedienteRecursoService - historicoEventoService: HistoricoEventoService
<ul style="list-style-type: none"> + add(ExpedienteContratacion): ExpedienteContratacion + edit(ExpedienteContratacion): ExpedienteContratacion + getById(Long): ExpedienteContratacion + getCreateForm(Model): String
<ul style="list-style-type: none"> «property gets» + getExpedienteContratacionService(): ExpedienteContratacionService + getExpedienteRecursoService(): ExpedienteRecursoService + getHistoricoEventoService(): HistoricoEventoService
<ul style="list-style-type: none"> «property sets» + setExpedienteContratacionService(ExpedienteContratacionService): void + setExpedienteRecursoService(ExpedienteRecursoService): void + setHistoricoEventoService(HistoricoEventoService): void

ExpedienteContratacionDao
<ul style="list-style-type: none"> + add(ExpedienteContratacion): ExpedienteContratacion + find(ExpedienteContratacion): ExpedienteContratacion + update(ExpedienteContratacion): ExpedienteContratacion

ExpedienteContratacionService
<ul style="list-style-type: none"> - expedienteContratacionDao: ExpedienteContratacionDao
<ul style="list-style-type: none"> + add(ExpedienteContratacion): ExpedienteContratacion + find(ExpedienteContratacion): ExpedienteContratacion + update(ExpedienteContratacion): ExpedienteContratacion
<ul style="list-style-type: none"> «property gets» + getExpedienteContratacionDao(): ExpedienteContratacionDao
<ul style="list-style-type: none"> «property sets» + setExpedienteContratacionDao(ExpedienteContratacionDao): void

HistoricoEvento
<ul style="list-style-type: none"> - colorCalendario: String - colorPlazo: String - descripcionEventoEs: String - descripcionEventoEu: String - descripcionPlazoEs: String - descripcionPlazoEu: String - flagCalendario: boolean - flagEntrada: boolean - flagSalida: boolean - idEvento: int - idExpedienteRecurso: long - idHistoricoEvento: long - idInteresado: long - idPlazo: int
<ul style="list-style-type: none"> «property gets» + getColorCalendario(): String + getColorPlazo(): String + getDescripcionEventoEs(): String + getDescripcionEventoEu(): String + getDescripcionPlazoEs(): String + getDescripcionPlazoEu(): String + getIdEvento(): long + getIdExpedienteRecurso(): long + getIdHistoricoEvento(): long + getIdInteresado(): long + getIdPlazo(): int + isFlagCalendario(): boolean + isFlagEntrada(): boolean + isFlagSalida(): boolean
<ul style="list-style-type: none"> «property sets» + setColorCalendario(String): void + setColorPlazo(String): void + setDescripcionEventoEs(String): void + setDescripcionEventoEu(String): void + setDescripcionPlazoEs(String): void + setDescripcionPlazoEu(String): void + setFlagCalendario(boolean): void + setFlagEntrada(boolean): void + setFlagSalida(boolean): void + setIdEvento(long): void + setIdExpedienteRecurso(long): void + setIdHistoricoEvento(long): void + setIdInteresado(long): void + setIdPlazo(int): void

HistoricoEventoService
<ul style="list-style-type: none"> - historicoEventoDao: HistoricoEvento
<ul style="list-style-type: none"> + add(HistoricoEvento): HistoricoEvento + find(HistoricoEvento): HistoricoEvento + findAll(Pagination, HistoricoEvento): List + findAllCount(HistoricoEvento): Long + findAllLike(boolean, Pagination, HistoricoEvento): List + findAllLikeCount(boolean, HistoricoEvento): Long + remove(HistoricoEvento): void + removeMultiple(List): void + update(HistoricoEvento): HistoricoEvento
<ul style="list-style-type: none"> «property gets» + getHistoricoEventoDao(): HistoricoEvento
<ul style="list-style-type: none"> «property sets» + setHistoricoEventoDao(HistoricoEvento): void

HistoricoEventoDao
<ul style="list-style-type: none"> + add(HistoricoEvento): HistoricoEvento + find(HistoricoEvento): ExpedienteRecurso + findAll(Pagination, HistoricoEvento): List + findAllCount(HistoricoEvento): Long + findAllLike(boolean, Pagination, HistoricoEvento): List + findAllLikeCount(boolean, HistoricoEvento): Long + generalNumExpediente(HistoricoEvento): HistoricoEvento + remove(HistoricoEvento): void + update(HistoricoEvento): HistoricoEvento

Documento
<ul style="list-style-type: none"> - data: byte[] - deleteType: String - fecha: Date - idDocumento: long - idExpedienteRecurso: long - idTipoDoc: int - mimeType: String - nombre: String - size: long - uri: String - uriDelete: String
<ul style="list-style-type: none"> «property gets» + getDate(): byte[] + getDeleteType(): String + getFecha(): Date + getIdDocumento(): long + getIdExpedienteRecurso(): long + getIdTipoDoc(): int + getMimeType(): String + getNombre(): String + getSize(): long + getUri(): String + getUriDelete(): String
<ul style="list-style-type: none"> «property sets» + setDate(byte[]): void + setDeleteType(String): void + setFecha(Date): void + setIdDocumento(long): void + setIdExpedienteRecurso(long): void + setIdTipoDoc(int): void + setMimeType(String): void + setNombre(String): void + setSize(long): void + setUri(String): void + setUriDelete(String): void

DocumentoController
<ul style="list-style-type: none"> - documentoService: DocumentoService
<ul style="list-style-type: none"> + add(Documento): Documento + edit(Documento): Documento + getAll(Documento): List + getAllPagination(Documento): JQGridJSONModel + getById(Long): Documento + remove(Multiple(List): List + removeMultiple(List): List
<ul style="list-style-type: none"> «property gets» + getDocumentoService(): DocumentoService
<ul style="list-style-type: none"> «property sets» + setDocumentoService(DocumentoService): void

DocumentoService
<ul style="list-style-type: none"> - documentoDao: DocumentoService
<ul style="list-style-type: none"> + add(Documento): Documento + find(Documento): Documento + findAll(Pagination, Documento): List + findAllCount(Documento): Long + remove(Documento): void + removeMultiple(List): void + update(Documento): Documento
<ul style="list-style-type: none"> «property gets» + getDocumentoDao(): int
<ul style="list-style-type: none"> «property sets» + setDocumentoDao(int): void

DocumentoDao
<ul style="list-style-type: none"> + add(Documento): Documento + find(Documento): Documento + findAll(Pagination, Documento): List + findAllCount(Documento): Long + remove(Documento): void + remove(Documento): void + update(Documento): Documento

Interesado
<ul style="list-style-type: none"> - adjudicador: boolean - cf: String - cpRepresentante: int - denominacionSocial: String - direccionRepresentante: String - dnRepresentante: String - idExpedienteRecurso: long - idMunicipioRepresentante: long - idProvinciaRepresentante: long - maRepresentante: String - medioPostal: boolean - nombreRepresentante: String - recuento: boolean - telefonoRepresentante: String
<ul style="list-style-type: none"> «property gets» + getAdjudicador(): boolean + getCf(): String + getCpRepresentante(): int + getDenominacionSocial(): String + getDireccionRepresentante(): String + getIdExpedienteRecurso(): long + getIdMunicipioRepresentante(): long + getIdProvinciaRepresentante(): long + getMaRepresentante(): String + getMedioPostal(): boolean + getNombreRepresentante(): String + getRecuento(): boolean + getTelefonoRepresentante(): String
<ul style="list-style-type: none"> «property sets» + setAdjudicador(boolean): void + setCf(String): void + setCpRepresentante(int): void + setDenominacionSocial(String): void + setDireccionRepresentante(String): void + setIdExpedienteRecurso(long): void + setIdMunicipioRepresentante(long): void + setIdProvinciaRepresentante(long): void + setMaRepresentante(String): void + setMedioPostal(boolean): void + setNombreRepresentante(String): void + setRecuento(boolean): void + setTelefonoRepresentante(String): void

InteresadoController
<ul style="list-style-type: none"> - interesadoService: InteresadoService
<ul style="list-style-type: none"> + add(Interesado): Interesado + edit(Interesado): Interesado + getAll(Interesado): List + getAllPagination(Interesado): JQGridJSONModel + getById(Long): Documento + remove(Long): Interesado + removeMultiple(List): List
<ul style="list-style-type: none"> «property gets» + getInteresadoService(): InteresadoService
<ul style="list-style-type: none"> «property sets» + setInteresadoService(InteresadoService): void

InteresadoService
<ul style="list-style-type: none"> - interesadoDao: InteresadoDao
<ul style="list-style-type: none"> + add(Interesado): Interesado + find(Interesado): Interesado + findAll(Pagination, Interesado): List + findAllCount(Interesado): Long + remove(Interesado): void + removeMultiple(List): void + update(Interesado): Documento
<ul style="list-style-type: none"> «property gets» + getInteresadoDao(): InteresadoDao
<ul style="list-style-type: none"> «property sets» + setInteresadoDao(InteresadoDao): void

InteresadoDao
<ul style="list-style-type: none"> + add(Interesado): Interesado + find(Interesado): Interesado + findAll(Pagination, Interesado): List + findAllCount(Interesado): Long + remove(Interesado): void + remove(Interesado): void + update(Interesado): Interesado

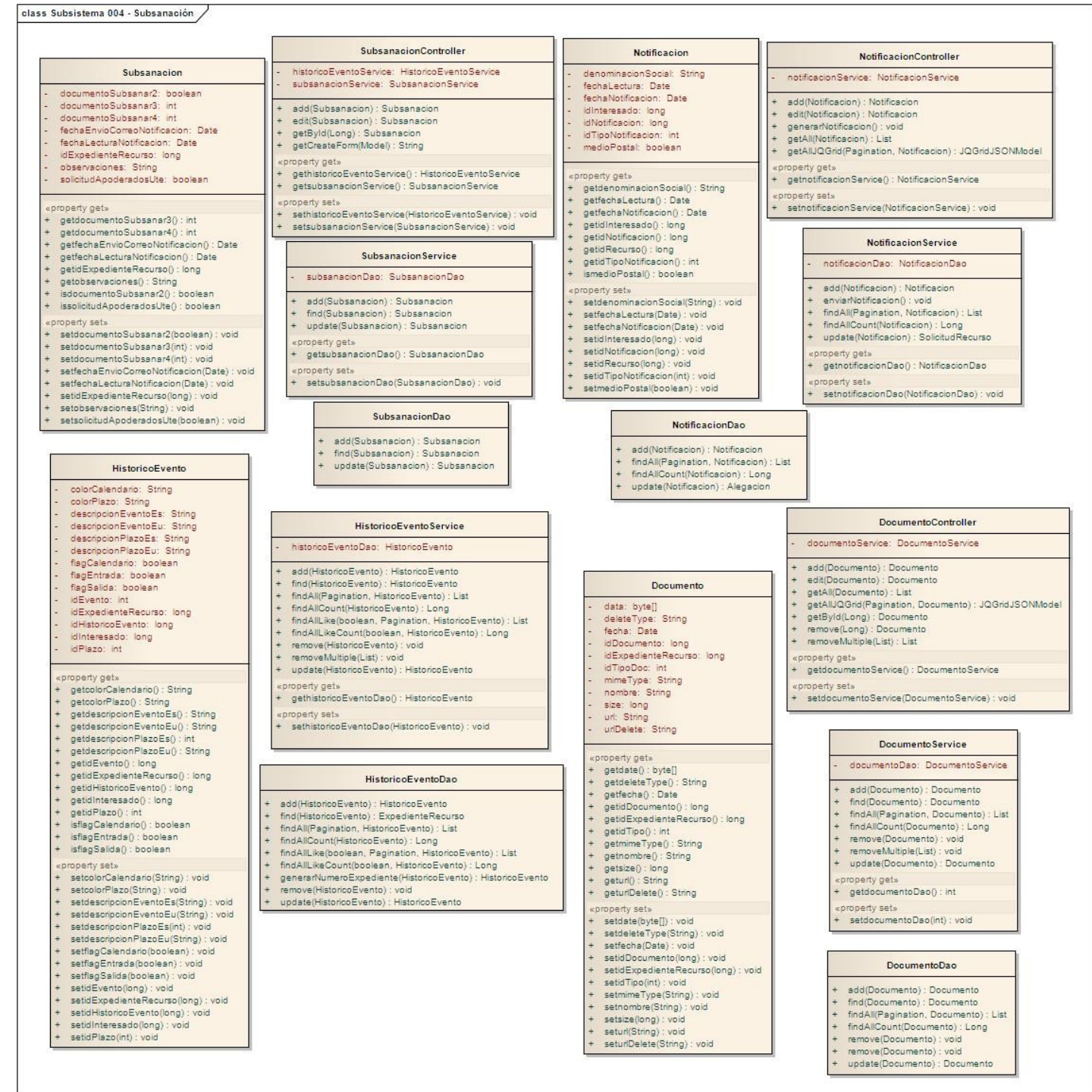
Notificacion
<ul style="list-style-type: none"> - denominacionSocial: String - fechaLectura: Date - fechaNotificacion: Date - idInteresado: long - idNotificacion: long - idTipoNotificacion: int - medioPostal: boolean
<ul style="list-style-type: none"> «property gets» + getDenominacionSocial(): String + getFechaLectura(): Date + getFechaNotificacion(): Date + getIdInteresado(): long + getIdNotificacion(): long + getIdTipoNotificacion(): int + getMedioPostal(): boolean
<ul style="list-style-type: none"> «property sets» + setDenominacionSocial(String): void + setFechaLectura(Date): void + setFechaNotificacion(Date): void + setIdInteresado(long): void + setIdNotificacion(long): void + setIdTipoNotificacion(int): void + setMedioPostal(boolean): void

NotificacionDao
<ul style="list-style-type: none"> + add(Notificacion): Notificacion + find(Notificacion): Notificacion + findAllCount(Notificacion): Long + update(Notificacion): Alegacion

NotificacionController
<ul style="list-style-type: none"> - notificacionService: NotificacionService
<ul style="list-style-type: none"> + add(Notificacion): Notificacion + edit(Notificacion): Notificacion + getAll(Notificacion): void + getAllPagination(Notificacion): JQGridJSONModel
<ul style="list-style-type: none"> «property gets» + getNotificacionService(): NotificacionService
<ul style="list-style-type: none"> «property sets» + setNotificacionService(NotificacionService): void

NotificacionService
<ul style="list-style-type: none"> - notificacionDao: NotificacionDao
<ul style="list-style-type: none"> + add(Notificacion): Notificacion + edit(Notificacion): void + findAll(Pagination, Notificacion): List + findAllCount(Notificacion): Long + update(Notificacion): SolicitudRecurso
<ul style="list-style-type: none"> «property gets» + getNotificacionDao(): NotificacionDao
<ul style="list-style-type: none"> «property sets» + setNotificacionDao(NotificacionDao): void

7.4 Subsistema 4 Subsanación



7.5 Subsistema 5 Traslado y alegaciones

class Subsistema 005 - Traslado y alegaciones

Alegacion
<ul style="list-style-type: none"> - denominacionSocial: String - fechaLimite: Date - fechaRecepcion: Date - idExpedienteRecurso: long - idRecurso: long - observaciones: String
<pre>«property get» + getdenominacionSocial(): String + getfechaLimite(): Date + getfechaRecepcion(): Date + getidExpedienteRecurso(): long + getidRecurso(): long + getobservaciones(): String «property set» + setdenominacionSocial(String): void + setfechaLimite(Date): void + setfechaRecepcion(Date): void + setidExpedienteRecurso(long): void + setidRecurso(long): void + setobservaciones(String): void</pre>

AlegacionDao
<ul style="list-style-type: none"> + add(Alegacion): Alegacion + find(Alegacion): Alegacion + findAll(Pagination, Alegacion): List + findAllCount(Alegacion): Long + remove(Alegacion): void + remove(Alegacion): void + update(Alegacion): Alegacion

AlegacionController
<ul style="list-style-type: none"> - alegacionService: AlegacionService - historicoEventoService: HistoricoEventoService
<pre>+ add(Alegacion): Alegacion + edit(Alegacion): Alegacion + getAll(Alegacion): List + getAliuQGrid(Pagination, Alegacion): JQueryJSONModel + getById(Long): Alegacion + getCreateForm(Model): String + remove(Long): Alegacion + removeMultiple(List): List «property get» + getalegacionService(): AlegacionService + gethistoricoEventoService(): HistoricoEventoService «property set» + setalegacionService(AlegacionService): void + sethistoricoEventoService(HistoricoEventoService): void</pre>

AlegacionService
<ul style="list-style-type: none"> - alegacionDao: AlegacionDao
<pre>+ add(Alegacion): Alegacion + find(Alegacion): Alegacion + findAll(Pagination, Alegacion): List + findAllCount(Alegacion): Long + remove(Alegacion): void + removeMultiple(List): void + update(Alegacion): Alegacion «property get» + getalegacionDao(): AlegacionDao «property set» + setalegacionDao(AlegacionDao): void</pre>

Notificacion
<ul style="list-style-type: none"> - denominacionSocial: String - fechaLectura: Date - fechaNotificacion: Date - idInteresado: long - idNotificacion: long - idTipoNotificacion: int - medioPostal: boolean
<pre>«property get» + getdenominacionSocial(): String + getfechaLectura(): Date + getfechaNotificacion(): Date + getidInteresado(): long + getidNotificacion(): long + getidRecurso(): long + getidTipoNotificacion(): int + ismedioPostal(): boolean «property set» + setdenominacionSocial(String): void + setfechaLectura(Date): void + setfechaNotificacion(Date): void + setidInteresado(long): void + setidNotificacion(long): void + setidRecurso(long): void + setidTipoNotificacion(int): void + setmedioPostal(boolean): void</pre>

NotificacionDao
<ul style="list-style-type: none"> + add(Notificacion): Notificacion + findAll(Pagination, Notificacion): List + findAllCount(Notificacion): Long + update(Notificacion): Alegacion

NotificacionController
<ul style="list-style-type: none"> - notificacionService: NotificacionService
<pre>+ add(Notificacion): Notificacion + edit(Notificacion): Notificacion + generarNotificacion(): void + getAll(Notificacion): List + getAliuQGrid(Pagination, Notificacion): JQueryJSONModel «property get» + getnotificacionService(): NotificacionService «property set» + setnotificacionService(NotificacionService): void</pre>

NotificacionService
<ul style="list-style-type: none"> - notificacionDao: NotificacionDao
<pre>+ add(Notificacion): Notificacion + enviarNotificacion(): void + findAll(Pagination, Notificacion): List + findAllCount(Notificacion): Long + update(Notificacion): SolicitudRecurso «property get» + getnotificacionDao(): NotificacionDao «property set» + setnotificacionDao(NotificacionDao): void</pre>

HistoricoEvento
<ul style="list-style-type: none"> - colorCalendario: String - colorPlazo: String - descripcionEventoEs: String - descripcionEventoEu: String - descripcionPlazoEs: String - descripcionPlazoEu: String - flagCalendario: boolean - flagEntrada: boolean - flagSalida: boolean - idEvento: int - idExpedienteRecurso: long - idHistoricoEvento: long - idInteresado: long - idPlazo: int
<pre>«property get» + getcolorCalendario(): String + getcolorPlazo(): String + getdescripcionEventoEs(): String + getdescripcionEventoEu(): String + getdescripcionPlazoEs(): int + getdescripcionPlazoEu(): String + getidEvento(): long + getidExpedienteRecurso(): long + getidHistoricoEvento(): long + getidInteresado(): long + getidPlazo(): int + isflagCalendario(): boolean + isflagEntrada(): boolean + isflagSalida(): boolean «property set» + setcolorCalendario(String): void + setcolorPlazo(String): void + setdescripcionEventoEs(String): void + setdescripcionEventoEu(String): void + setdescripcionPlazoEs(int): void + setdescripcionPlazoEu(String): void + setflagCalendario(boolean): void + setflagEntrada(boolean): void + setflagSalida(boolean): void + setidEvento(long): void + setidExpedienteRecurso(long): void + setidHistoricoEvento(long): void + setidInteresado(long): void + setidPlazo(int): void</pre>

HistoricoEventoService
<ul style="list-style-type: none"> - historicoEventoDao: HistoricoEvento
<pre>+ add(HistoricoEvento): HistoricoEvento + find(HistoricoEvento): HistoricoEvento + findAll(Pagination, HistoricoEvento): List + findAllCount(HistoricoEvento): Long + findAllLike(boolean, Pagination, HistoricoEvento): List + findAllLikeCount(boolean, Pagination, HistoricoEvento): Long + remove(HistoricoEvento): void + removeMultiple(List): void + update(HistoricoEvento): HistoricoEvento «property get» + gethistoricoEventoDao(): HistoricoEvento «property set» + sethistoricoEventoDao(HistoricoEvento): void</pre>

HistoricoEventoDao
<pre>+ add(HistoricoEvento): HistoricoEvento + find(HistoricoEvento): ExpedienteRecurso + findAll(Pagination, HistoricoEvento): List + findAllCount(HistoricoEvento): Long + findAllLike(boolean, Pagination, HistoricoEvento): List + findAllLikeCount(boolean, Pagination, HistoricoEvento): Long + generarNumeroExpediente(HistoricoEvento): HistoricoEvento + remove(HistoricoEvento): void + update(HistoricoEvento): HistoricoEvento</pre>

Documento
<ul style="list-style-type: none"> - data: byte[] - deleteType: String - fecha: Date - idDocumento: long - idExpedienteRecurso: long - idTipoDoc: int - mimeType: String - nombre: String - size: long - uri: String - uriDelete: String
<pre>«property get» + getDate(): byte[] + getdeleteType(): String + getfecha(): Date + getidDocumento(): long + getidExpedienteRecurso(): long + getidTipo(): int + getmimeType(): String + getnombre(): String + getsize(): long + geturi(): String + geturiDelete(): String «property set» + setDate(byte[]): void + setdeleteType(String): void + setfecha(Date): void + setidDocumento(long): void + setidExpedienteRecurso(long): void + setidTipo(int): void + setmimeType(String): void + setnombre(String): void + setsize(long): void + seturi(String): void + seturiDelete(String): void</pre>

DocumentoController
<ul style="list-style-type: none"> - documentoService: DocumentoService
<pre>+ add(Documento): Documento + edit(Documento): Documento + getAll(Documento): List + getAliuQGrid(Pagination, Documento): JQueryJSONModel + getById(Long): Documento + remove(Long): Documento + removeMultiple(List): List «property get» + getdocumentoService(): DocumentoService «property set» + setdocumentoService(DocumentoService): void</pre>

DocumentoService
<ul style="list-style-type: none"> - documentoDao: DocumentoService
<pre>+ add(Documento): Documento + find(Documento): Documento + findAll(Pagination, Documento): List + findAllCount(Documento): Long + remove(Documento): void + removeMultiple(List): void + update(Documento): Documento «property get» + getdocumentoDao(): int «property set» + setdocumentoDao(int): void</pre>

DocumentoDao
<pre>+ add(Documento): Documento + find(Documento): Documento + findAll(Pagination, Documento): List + findAllCount(Documento): Long + remove(Documento): void + remove(Documento): void + update(Documento): Documento</pre>

7.6 Subsistema 6 Medidas Provisionales

class Subsistema 006 - Medidas Provisionales

Notificacion
<ul style="list-style-type: none"> - denominacionSocial: String - fechaLectura: Date - fechaNotificacion: Date - idInteresado: long - idNotificacion: long - idTipoNotificacion: int - medioPostal: boolean
<ul style="list-style-type: none"> «property gets» + getdenominacionSocial(): String + getfechaLectura(): Date + getfechaNotificacion(): Date + getidInteresado(): long + getidNotificacion(): long + getidRecurso(): long + getidTipoNotificacion(): int + ismedioPostal(): boolean «property sets» + setdenominacionSocial(String): void + setfechaLectura(Date): void + setfechaNotificacion(Date): void + setidInteresado(long): void + setidNotificacion(long): void + setidRecurso(long): void + setidTipoNotificacion(int): void + setmedioPostal(boolean): void

NotificacionDao
<ul style="list-style-type: none"> + add(Notification): Notification + findAll(Pagination, Notification): List + findAllCount(Notification): Long + update(Notification): Alegacion

NotificacionController
<ul style="list-style-type: none"> - notificacionService: NotificacionService + add(Notification): Notification + edit(Notification): Notification + generarNotificacion(): void + getAll(Notification): List + getAllJQGrid(Pagination, Notification): JQGridJSONModel «property gets» + getnotificacionService(): NotificacionService «property sets» + setnotificacionService(NotificacionService): void

NotificacionService
<ul style="list-style-type: none"> - notificacionDao: NotificacionDao + add(Notification): Notification + enviarNotificacion(): void + findAll(Pagination, Notification): List + findAllCount(Notification): Long + update(Notification): SolicitudRecurso «property gets» + getnotificacionDao(): NotificacionDao «property sets» + setnotificacionDao(NotificacionDao): void

MedidasProvisionalesController
<ul style="list-style-type: none"> - historicoEventoService: HistoricoEventoService + getCreateForm(Model): String «property gets» + gethistoricoEventoService(): HistoricoEventoService «property sets» + sethistoricoEventoService(HistoricoEventoService): void

HistoricoEvento
<ul style="list-style-type: none"> - colorCalendario: String - colorPlazo: String - descripcionEventoEs: String - descripcionEventoEu: String - descripcionPlazoEs: String - descripcionPlazoEu: String - flagCalendario: boolean - flagEntrada: boolean - flagSalida: boolean - idEvento: int - idExpedienteRecurso: long - idHistoricoEvento: long - idInteresado: long - idPlazo: int
<ul style="list-style-type: none"> «property gets» + getcolorCalendario(): String + getcolorPlazo(): String + getdescripcionEventoEs(): String + getdescripcionEventoEu(): String + getdescripcionPlazoEs(): int + getdescripcionPlazoEu(): String + getidEvento(): long + getidExpedienteRecurso(): long + getidHistoricoEvento(): long + getidInteresado(): long + getidPlazo(): int + isflagCalendario(): boolean + isflagEntrada(): boolean + isflagSalida(): boolean «property sets» + setcolorCalendario(String): void + setcolorPlazo(String): void + setdescripcionEventoEs(String): void + setdescripcionEventoEu(String): void + setdescripcionPlazoEs(int): void + setdescripcionPlazoEu(String): void + setflagCalendario(boolean): void + setflagEntrada(boolean): void + setflagSalida(boolean): void + setidEvento(long): void + setidExpedienteRecurso(long): void + setidHistoricoEvento(long): void + setidInteresado(long): void + setidPlazo(int): void

HistoricoEventoService
<ul style="list-style-type: none"> - historicoEventoDao: HistoricoEvento + add(HistoricoEvento): HistoricoEvento + find(HistoricoEvento): HistoricoEvento + findAll(Pagination, HistoricoEvento): List + findAllCount(HistoricoEvento): Long + findAllLike(boolean, Pagination, HistoricoEvento): List + findAllLikeCount(boolean, HistoricoEvento): Long + remove(HistoricoEvento): void + removeMultiple(List): void + update(HistoricoEvento): HistoricoEvento «property gets» + gethistoricoEventoDao(): HistoricoEvento «property sets» + sethistoricoEventoDao(HistoricoEvento): void

HistoricoEventoDao
<ul style="list-style-type: none"> + add(HistoricoEvento): HistoricoEvento + find(HistoricoEvento): ExpedienteRecurso + findAll(Pagination, HistoricoEvento): List + findAllCount(HistoricoEvento): Long + findAllLikeCount(boolean, HistoricoEvento): Long + generarNumeroExpediente(HistoricoEvento): HistoricoEvento + remove(HistoricoEvento): void + update(HistoricoEvento): HistoricoEvento

Documento
<ul style="list-style-type: none"> - data: byte[] - deleteType: String - fecha: Date - idDocumento: long - idExpedienteRecurso: long - idTipoDoc: int - mimeType: String - nombre: String - size: long - uri: String - uriDelete: String
<ul style="list-style-type: none"> «property gets» + getdata(): byte[] + getdeleteType(): String + getfecha(): Date + getidDocumento(): long + getidExpedienteRecurso(): long + getidTipo(): int + getmimeType(): String + getnombre(): String + getsize(): long + geturi(): String + geturiDelete(): String «property sets» + setdata(byte[]): void + setdeleteType(String): void + setfecha(Date): void + setidDocumento(long): void + setidExpedienteRecurso(long): void + setidTipo(int): void + setmimeType(String): void + setnombre(String): void + setsize(long): void + seturi(String): void + seturiDelete(String): void

DocumentoController
<ul style="list-style-type: none"> - documentoService: DocumentoService + add(Documento): Documento + edit(Documento): Documento + getAll(Documento): List + getAllJQGrid(Pagination, Documento): JQGridJSONModel + getByid(Long): Documento + remove(Long): Documento + removeMultiple(List): List «property gets» + getdocumentoService(): DocumentoService «property sets» + setdocumentoService(DocumentoService): void

DocumentoService
<ul style="list-style-type: none"> - documentoDao: DocumentoService + add(Documento): Documento + find(Documento): Documento + findAll(Pagination, Documento): List + findAllCount(Documento): Long + remove(Documento): void + removeMultiple(List): Documento + update(Documento): Documento «property gets» + getdocumentoDao(): int «property sets» + setdocumentoDao(int): void

DocumentoDao
<ul style="list-style-type: none"> + add(Documento): Documento + find(Documento): Documento + findAll(Pagination, Documento): List + findAllCount(Documento): Long + remove(Documento): void + remove(Documento): void + update(Documento): Documento

7.7 Subsistema 7 Práctica de prueba

class Subsistema 007 - Práctica de prueba



7.8 Subsistema 8 Resolución del recurso

class Subsistema 008 - Resolución del recurso

Resolucion
<ul style="list-style-type: none"> documentacionReferencia: List documentoResolucion: Documento fechaPresentacionRecurso: Date fechaResolucion: Date idExpedienteRecurso: long numeroResolucion: String sentidoResolucion: int
«property gets» + getdocumentacionReferencia(): List + getdocumentoResolucion(): Documento + getfechaPresentacionRecurso(): Date + getfechaResolucion(): Date + getidExpedienteRecurso(): long + getnumeroResolucion(): String + getsentidoResolucion(): int
«property sets» + setdocumentacionReferencia(List): void + setdocumentoResolucion(Documento): void + setfechaPresentacionRecurso(Date): void + setfechaResolucion(Date): void + setidExpedienteRecurso(long): void + setnumeroResolucion(String): void + setsentidoResolucion(int): void

ResolucionController
<ul style="list-style-type: none"> historicoEventoService: HistoricoEventoService resolucionService: ResolucionService
+ add(Resolucion): Resolucion + edit(Resolucion): Resolucion + getByld(Long): Resolucion + getCreateForm(Model): String
«property gets» + gethistoricoEventoService(): HistoricoEventoService + getresolucionService(): ResolucionService
«property sets» + sethistoricoEventoService(HistoricoEventoService): void + setresolucionService(ResolucionService): void

ResolucionService
<ul style="list-style-type: none"> resolucionDao: ResolucionDao
+ add(Resolucion): Resolucion + find(Resolucion): Resolucion + update(Resolucion): Resolucion
«property gets» + getresolucionDao(): ResolucionDao
«property sets» + setresolucionDao(ResolucionDao): void

ResolucionDao
+ add(Resolucion): Resolucion + find(Resolucion): Resolucion + update(Resolucion): Resolucion

Notificacion
<ul style="list-style-type: none"> denominacionSocial: String fechaLectura: Date fechaNotificacion: Date idInteresado: long idNotificacion: long idTipoNotificacion: int medioPostal: boolean
«property gets» + getdenominacionSocial(): String + getfechaLectura(): Date + getfechaNotificacion(): Date + getidInteresado(): long + getidNotificacion(): long + getidRecurso(): long + getidTipoNotificacion(): int + ismedioPostal(): boolean
«property sets» + setdenominacionSocial(String): void + setfechaLectura(Date): void + setfechaNotificacion(Date): void + setidInteresado(long): void + setidNotificacion(long): void + setidRecurso(long): void + setidTipoNotificacion(int): void + setmedioPostal(boolean): void

NotificacionDao
+ add(Notificacion): Notificacion + findAll(Pagination, Notificacion): List + findAllCount(Notificacion): Long + update(Notificacion): Alegacion

NotificacionController
<ul style="list-style-type: none"> notificacionService: NotificacionService
+ add(Notificacion): Notificacion + edit(Notificacion): Notificacion + generarNotificacion(): void + getAll(Notificacion): List + getAllJQGrid(Pagination, Notificacion): JQGridJSONModel
«property gets» + getnotificacionService(): NotificacionService
«property sets» + setnotificacionService(NotificacionService): void

NotificacionService
<ul style="list-style-type: none"> notificacionDao: NotificacionDao
+ add(Notificacion): Notificacion + enviarNotificacion(): void + findAll(Pagination, Notificacion): List + findAllCount(Notificacion): Long + update(Notificacion): SolicitudRecurso
«property gets» + getnotificacionDao(): NotificacionDao
«property sets» + setnotificacionDao(NotificacionDao): void

HistoricoEvento
<ul style="list-style-type: none"> colorCalendario: String colorPlazo: String descripcionEventoEs: String descripcionEventoEu: String descripcionPlazoEs: String descripcionPlazoEu: String flagCalendario: boolean flagEntrada: boolean flagSalida: boolean idEvento: int idExpedienteRecurso: long idHistoricoEvento: long idInteresado: long idPlazo: int
«property gets» + getcolorCalendario(): String + getcolorPlazo(): String + getdescripcionEventoEs(): String + getdescripcionEventoEu(): String + getdescripcionPlazoEs(): String + getdescripcionPlazoEu(): String + getidEvento(): long + getidExpedienteRecurso(): long + getidHistoricoEvento(): long + getidInteresado(): long + getidPlazo(): int + isflagCalendario(): boolean + isflagEntrada(): boolean + isflagSalida(): boolean
«property sets» + setcolorCalendario(String): void + setcolorPlazo(String): void + setdescripcionEventoEs(String): void + setdescripcionEventoEu(String): void + setdescripcionPlazoEs(int): void + setdescripcionPlazoEu(String): void + setflagCalendario(boolean): void + setflagEntrada(boolean): void + setflagSalida(boolean): void + setidEvento(long): void + setidExpedienteRecurso(long): void + setidHistoricoEvento(long): void + setidInteresado(long): void + setidPlazo(int): void

HistoricoEventoService
<ul style="list-style-type: none"> historicoEventoDao: HistoricoEvento
+ add(HistoricoEvento): HistoricoEvento + find(HistoricoEvento): HistoricoEvento + findAll(Pagination, HistoricoEvento): List + findAllCount(HistoricoEvento): Long + findAllLike(boolean, Pagination, HistoricoEvento): List + findAllLikeCount(boolean, HistoricoEvento): Long + remove(HistoricoEvento): void + removeMultiple(List): void + update(HistoricoEvento): HistoricoEvento
«property gets» + gethistoricoEventoDao(): HistoricoEvento
«property sets» + sethistoricoEventoDao(HistoricoEvento): void

HistoricoEventoDao
+ add(HistoricoEvento): HistoricoEvento + find(HistoricoEvento): ExpedienteRecurso + findAll(Pagination, HistoricoEvento): List + findAllCount(HistoricoEvento): Long + findAllLike(boolean, Pagination, HistoricoEvento): List + findAllLikeCount(boolean, HistoricoEvento): Long + generarNumeroExpediente(HistoricoEvento): HistoricoEvento + remove(HistoricoEvento): void + update(HistoricoEvento): HistoricoEvento

Documento
<ul style="list-style-type: none"> data: byte[] deleteType: String fecha: Date idDocumento: long idExpedienteRecurso: long idTipoDoc: int mimeType: String nombre: String size: long url: String urlDelete: String
«property gets» + getdata(): byte[] + getdeleteType(): String + getfecha(): Date + getidDocumento(): long + getidExpedienteRecurso(): long + getidTipo(): int + getmimeType(): String + getnombre(): String + getsize(): long + geturl(): String + geturlDelete(): String
«property sets» + setdata(byte[]): void + setdeleteType(String): void + setfecha(Date): void + setidDocumento(long): void + setidExpedienteRecurso(long): void + setidTipo(int): void + setmimeType(String): void + setnombre(String): void + setsize(long): void + seturl(String): void + seturlDelete(String): void

DocumentoController
<ul style="list-style-type: none"> documentoService: DocumentoService
+ add(Documento): Documento + edit(Documento): Documento + getAll(Documento): List + getAllJQGrid(Pagination, Documento): JQGridJSONModel + getByld(Long): Documento + remove(Long): Documento + removeMultiple(List): List
«property gets» + getdocumentoService(): DocumentoService
«property sets» + setdocumentoService(DocumentoService): void

DocumentoService
<ul style="list-style-type: none"> documentoDao: DocumentoService
+ add(Documento): Documento + find(Documento): Documento + findAll(Pagination, Documento): List + findAllCount(Documento): Long + remove(Documento): void + removeMultiple(List): void + update(Documento): Documento
«property gets» + getdocumentoDao(): Documento
«property sets» + setdocumentoDao(int): void

DocumentoDao
+ add(Documento): Documento + find(Documento): Documento + findAll(Pagination, Documento): List + findAllCount(Documento): Long + remove(Documento): void + remove(Documento): void + update(Documento): Documento

7.9 Subsistema 9 Histórico

class Subsistema 009 - Histórico

HistoricoEvento

- colorCalendario: String
- colorPlazo: String
- descripcionEventoEs: String
- descripcionEventoEu: String
- descripcionPlazoEs: String
- descripcionPlazoEu: String
- flagCalendario: boolean
- flagEntrada: boolean
- flagSalida: boolean
- idEvento: int
- idExpedienteRecurso: long
- idHistoricoEvento: long
- idInteresado: long
- idPlazo: int

«property get»

- + getColorCalendario(): String
- + getColorPlazo(): String
- + getdescripcionEventoEs(): String
- + getdescripcionEventoEu(): String
- + getdescripcionPlazoEs(): int
- + getdescripcionPlazoEu(): String
- + getIdEvento(): long
- + getIdExpedienteRecurso(): long
- + getIdHistoricoEvento(): long
- + getIdInteresado(): long
- + getIdPlazo(): int
- + isflagCalendario(): boolean
- + isflagEntrada(): boolean
- + isflagSalida(): boolean

«property set»

- + setColorCalendario(String): void
- + setColorPlazo(String): void
- + setdescripcionEventoEs(String): void
- + setdescripcionEventoEu(String): void
- + setdescripcionPlazoEs(int): void
- + setdescripcionPlazoEu(String): void
- + setflagCalendario(boolean): void
- + setflagEntrada(boolean): void
- + setflagSalida(boolean): void
- + setIdEvento(long): void
- + setIdExpedienteRecurso(long): void
- + setIdHistoricoEvento(long): void
- + setIdInteresado(long): void
- + setIdPlazo(int): void

HistoricoController

- historicoEventoService: HistoricoEventoService

- + getAll(HistoricoEvento): List
- + getAllJQGrid(Pagination, HistoricoEvento): JQGridJSONModel
- + getCreateForm(Model): String

«property get»

- + gethistoricoEventoService(): HistoricoEventoService

«property set»

- + sethistoricoEventoService(HistoricoEventoService): void

HistoricoEventoService

- historicoEventoDao: HistoricoEvento

- + add(HistoricoEvento): HistoricoEvento
- + find(HistoricoEvento): HistoricoEvento
- + findAll(Pagination, HistoricoEvento): List
- + findAllCount(HistoricoEvento): Long
- + findAllLike(boolean, Pagination, HistoricoEvento): List
- + findAllLikeCount(boolean, HistoricoEvento): Long
- + remove(HistoricoEvento): void
- + removeMultiple(List): void
- + update(HistoricoEvento): HistoricoEvento

«property get»

- + gethistoricoEventoDao(): HistoricoEvento

«property set»

- + sethistoricoEventoDao(HistoricoEvento): void

HistoricoEventoDao

- + add(HistoricoEvento): HistoricoEvento
- + find(HistoricoEvento): ExpedienteRecurso
- + findAll(Pagination, HistoricoEvento): List
- + findAllCount(HistoricoEvento): Long
- + findAllLike(boolean, Pagination, HistoricoEvento): List
- + findAllLikeCount(boolean, HistoricoEvento): Long
- + generarNumeroExpediente(HistoricoEvento): HistoricoEvento
- + remove(HistoricoEvento): void
- + update(HistoricoEvento): HistoricoEvento

7.10 Subsistema 10 Acceso a expedientes

class Subsistema 010 - Acceso a expedientes

HistoricoEvento

- colorCalendario: String
- colorPlazo: String
- descripcionEventoEs: String
- descripcionEventoEu: String
- descripcionPlazoEs: String
- descripcionPlazoEu: String
- flagCalendario: boolean
- flagEntrada: boolean
- flagSalida: boolean
- idEvento: int
- idExpedienteRecurso: long
- idHistoricoEvento: long
- idInteresado: long
- idPlazo: int

«property get»

- + getColorCalendario() : String
- + getColorPlazo() : String
- + getdescripcionEventoEs() : String
- + getdescripcionEventoEu() : String
- + getdescripcionPlazoEs() : int
- + getdescripcionPlazoEu() : String
- + getIdEvento() : long
- + getIdExpedienteRecurso() : long
- + getIdHistoricoEvento() : long
- + getIdInteresado() : long
- + getIdPlazo() : int
- + isflagCalendario() : boolean
- + isflagEntrada() : boolean
- + isflagSalida() : boolean

«property set»

- + setColorCalendario(String) : void
- + setColorPlazo(String) : void
- + setdescripcionEventoEs(String) : void
- + setdescripcionEventoEu(String) : void
- + setdescripcionPlazoEs(int) : void
- + setdescripcionPlazoEu(String) : void
- + setflagCalendario(boolean) : void
- + setflagEntrada(boolean) : void
- + setflagSalida(boolean) : void
- + setIdEvento(long) : void
- + setIdExpedienteRecurso(long) : void
- + setIdHistoricoEvento(long) : void
- + setIdInteresado(long) : void
- + setIdPlazo(int) : void

AccesoExpedientesController

- historicoEventoService: HistoricoEventoService

- + archivar(List) : List
- + getAll(HistoricoEvento) : List
- + getAllJQGrid(Pagination, HistoricoEvento) : JQGridJSONModel
- + getCreateForm(Model) : String

«property get»

- + gethistoricoEventoService() : HistoricoEventoService

«property set»

- + sethistoricoEventoService(HistoricoEventoService) : void

HistoricoEventoService

- historicoEventoDao: HistoricoEvento

- + add(HistoricoEvento) : HistoricoEvento
- + find(HistoricoEvento) : HistoricoEvento
- + findAll(Pagination, HistoricoEvento) : List
- + findAllCount(HistoricoEvento) : Long
- + findAllLike(boolean, Pagination, HistoricoEvento) : List
- + findAllLikeCount(boolean, HistoricoEvento) : Long
- + remove(HistoricoEvento) : void
- + removeMultiple(List) : void
- + update(HistoricoEvento) : HistoricoEvento

«property get»

- + gethistoricoEventoDao() : HistoricoEvento

«property set»

- + sethistoricoEventoDao(HistoricoEvento) : void

HistoricoEventoDao

- + add(HistoricoEvento) : HistoricoEvento
- + find(HistoricoEvento) : ExpedienteRecurso
- + findAll(Pagination, HistoricoEvento) : List
- + findAllCount(HistoricoEvento) : Long
- + findAllLike(boolean, Pagination, HistoricoEvento) : List
- + findAllLikeCount(boolean, HistoricoEvento) : Long
- + generarNumeroExpediente(HistoricoEvento) : HistoricoEvento
- + remove(HistoricoEvento) : void
- + update(HistoricoEvento) : HistoricoEvento

7.11 Subsistema 11 Consulta pública de resoluciones

class Subsistema 011 - Consulta pública de resoluciones

ExpedienteRecurso

- actoRecurrido: Documento
- anuncioPrevioExp: Documento
- escritoRecurso: Documento
- idAmbito: int
- idEntidadImpulsora: long
- idExpedienteRecurso: long
- idPoderAdjudicador: long
- idTipoActo: int
- idTipoProcedimiento: int
- interesado: Interesado
- medidasCautelares: boolean
- numExpedienteContratacion: String
- objetoContrato: String
- otrosDocumentos: List
- poderPresCompareciente: Documento
- practicaPruebas: boolean

«property get»

- + getactoRecurrido(): Documento
- + getanuncioPrevioExp(): int
- + getescritoRecurso(): Documento
- + getidAmbito(): int
- + getidEntidadImpulsora(): long
- + getidExpedienteRecurso(): long
- + getidPoderAdjudicador(): long
- + getidTipoActo(): int
- + getidTipoProcedimiento(): int
- + getinteresado(): Interesado
- + getnumExpedienteContratacion(): String
- + getobjetoContrato(): String
- + getpoderPresCompareciente(): Documento
- + ismedidasCautelares(): boolean
- + ispracticaPruebas(): boolean

«property set»

- + setactoRecurrido(Documento): void
- + setanuncioPrevioExp(int): void
- + setescritoRecurso(Documento): void
- + setidAmbito(int): void
- + setidEntidadImpulsora(long): void
- + setidExpedienteRecurso(long): void
- + setidPoderAdjudicador(long): void
- + setidTipoActo(int): void
- + setidTipoProcedimiento(int): void
- + setinteresado(Interesado): void
- + setmedidasCautelares(boolean): void
- + setnumExpedienteContratacion(String): void
- + setobjetoContrato(String): void
- + setpoderPresCompareciente(Documento): void
- + setpracticaPruebas(boolean): void

ExpedienteRecursoController

- expedienteRecursoService: ExpedienteRecursoService

- + add(ExpedienteRecurso): ExpedienteRecurso
- + edit(ExpedienteRecurso): ExpedienteRecurso
- + getAll(ExpedienteRecurso): List
- + getAllJQGrid(Pagination, ExpedienteRecurso): JQGridJSONModel
- + getByid(Long): ExpedienteRecurso
- + getcreateForm(Model): String
- + remove(Long): ExpedienteRecurso
- + removeMultiple(List): List

«property get»

- + getexpedienteRecursoService(): ExpedienteRecursoService

«property set»

- + setexpedienteRecursoService(ExpedienteRecursoService): void

ExpedienteRecursoService

- expedienteRecursoDao: ExpedienteRecursoDao

- + add(ExpedienteRecurso): ExpedienteRecurso
- + find(ExpedienteRecurso): ExpedienteRecurso
- + findAll(Pagination, ExpedienteRecurso): List
- + findAllCount(ExpedienteRecurso): Long
- + findAllLike(boolean, Pagination, ExpedienteRecurso): List
- + findAllLikeCount(boolean, ExpedienteRecurso): Long
- + remove(ExpedienteRecurso): void
- + removeMultiple(List): void
- + update(ExpedienteRecurso): ExpedienteRecurso

«property get»

- + getexpedienteRecursoDao(): ExpedienteRecursoDao

«property set»

- + setexpedienteRecursoDao(ExpedienteRecursoDao): void

ExpedienteRecursoDao

- + add(ExpedienteRecurso): ExpedienteRecurso
- + find(ExpedienteRecurso): ExpedienteRecurso
- + findAll(Pagination, ExpedienteRecurso): List
- + findAllCount(ExpedienteRecurso): Long
- + findAllLike(boolean, Pagination, ExpedienteRecurso): List
- + findAllLikeCount(boolean, ExpedienteRecurso): Long
- + generarNumeroExpediente(ExpedienteRecurso): ExpedienteRecurso
- + remove(ExpedienteRecurso): void
- + update(ExpedienteRecurso): ExpedienteRecurso

7.12 Subsistema 12 Informes

class Subsistema 012 - Generacion de informes

ExpedienteRecurso

- actoRecurrido: Documento
- anuncioPrevioExp: Documento
- escritoRecurso: Documento
- idAmbito: int
- idEntidadImpulsora: long
- idExpedienteRecurso: long
- idPoderAdjudicador: long
- idTipoActo: int
- idTipoProcedimiento: int
- interesado: Interesado
- medidasCautelares: boolean
- numExpedienteContratacion: String
- objetoContrato: String
- otrosDocumentos: List
- poderPresCompareciente: Documento
- practicaPruebas: boolean

«property get»

- + getactoRecurrido(): Documento
- + getanuncioPrevioExp(): int
- + getescritoRecurso(): Documento
- + getidAmbito(): int
- + getidEntidadImpulsora(): long
- + getidExpedienteRecurso(): long
- + getidPoderAdjudicador(): long
- + getidTipoActo(): int
- + getidTipoProcedimiento(): int
- + getinteresado(): Interesado
- + getnumExpedienteContratacion(): String
- + getobjetoContrato(): String
- + getpoderPresCompareciente(): Documento
- + ismedidasCautelares(): boolean
- + ispracticaPruebas(): boolean

«property set»

- + setactoRecurrido(Documento): void
- + setanuncioPrevioExp(int): void
- + setescritoRecurso(Documento): void
- + setidAmbito(int): void
- + setidEntidadImpulsora(long): void
- + setidExpedienteRecurso(long): void
- + setidPoderAdjudicador(long): void
- + setidTipoActo(int): void
- + setidTipoProcedimiento(int): void
- + setinteresado(Interesado): void
- + setmedidasCautelares(boolean): void
- + setnumExpedienteContratacion(String): void
- + setobjetoContrato(String): void
- + setpoderPresCompareciente(Documento): void
- + setpracticaPruebas(boolean): void

InformesController

- expedienteRecursoService: ExpedienteRecursoService

+ generarInforme(ExpedienteRecurso): List

«property get»

+ getExpedienteRecursoService(): ExpedienteRecursoService

«property set»

+ setExpedienteRecursoService(ExpedienteRecursoService): void

ExpedienteRecursoService

- expedienteRecursoDao: ExpedienteRecursoDao

+ add(ExpedienteRecurso): ExpedienteRecurso

+ find(ExpedienteRecurso): ExpedienteRecurso

+ findAll(Pagination, ExpedienteRecurso): List

+ findAllCount(ExpedienteRecurso): Long

+ findAllLike(boolean, Pagination, ExpedienteRecurso): List

+ findAllLikeCount(boolean, ExpedienteRecurso): Long

+ remove(ExpedienteRecurso): void

+ removeMultiple(List): void

+ update(ExpedienteRecurso): ExpedienteRecurso

«property get»

+ getExpedienteRecursoDao(): ExpedienteRecursoDao

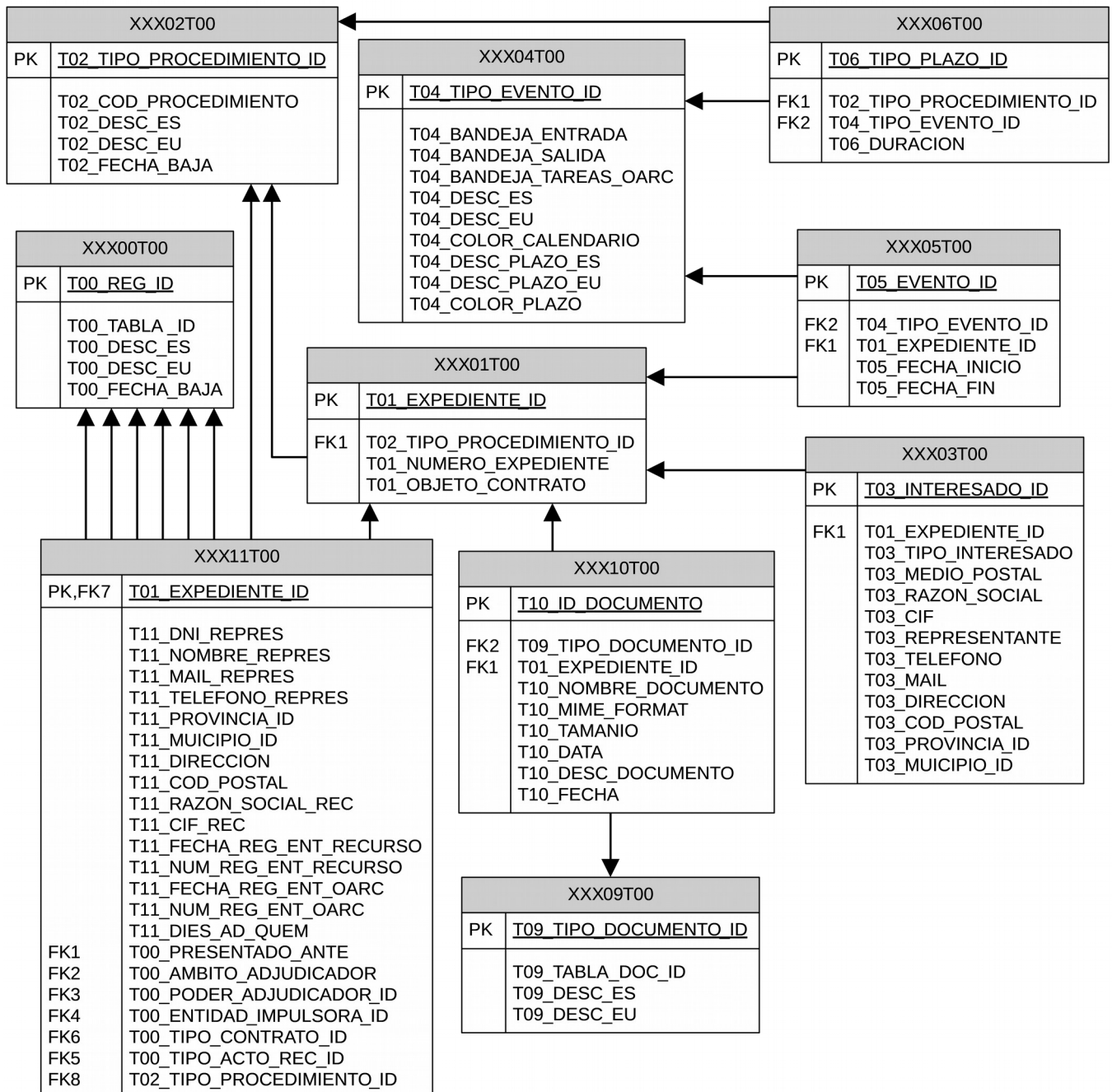
«property set»

+ setExpedienteRecursoDao(ExpedienteRecursoDao): void

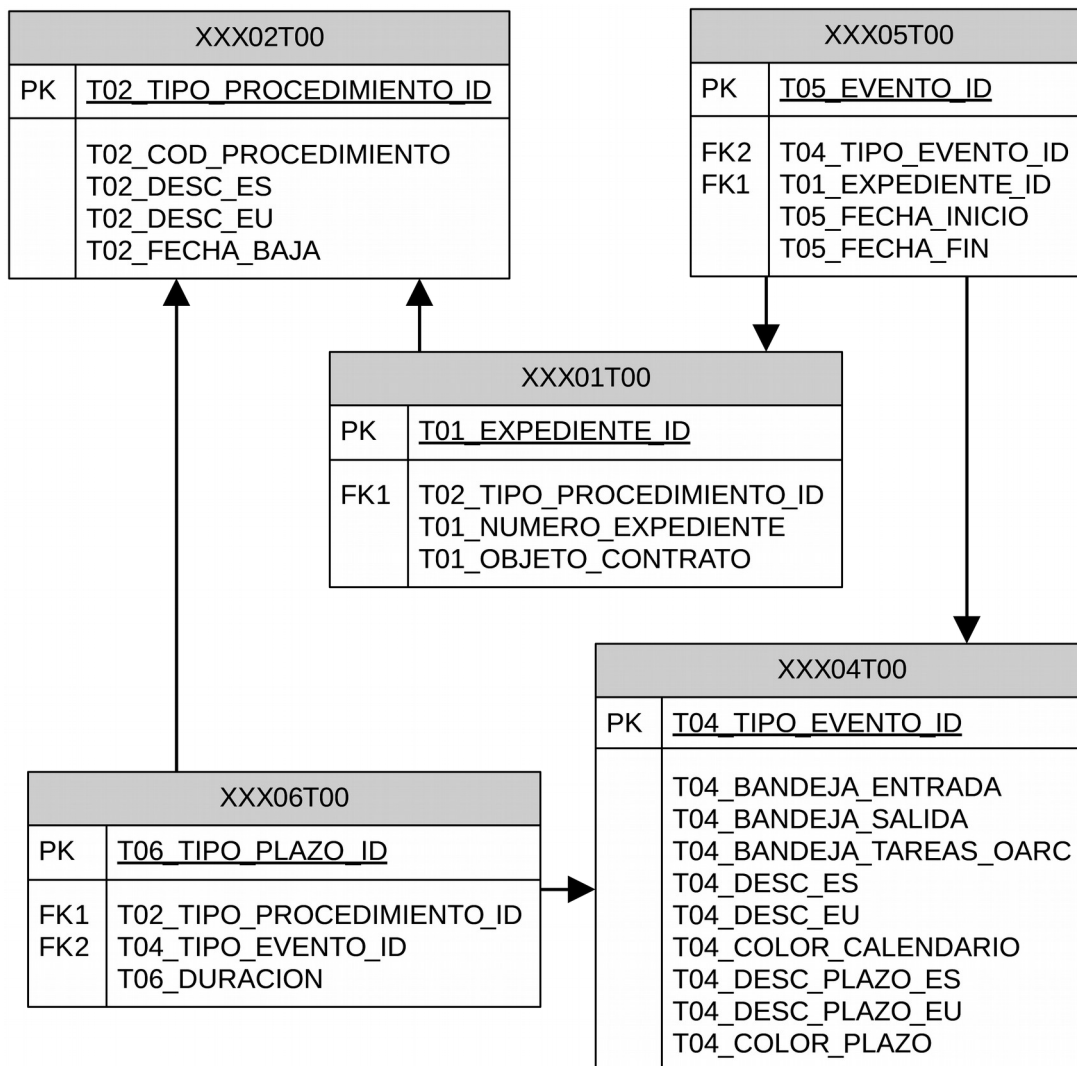
ExpedienteRecursoDao

- + add(ExpedienteRecurso): ExpedienteRecurso
- + find(ExpedienteRecurso): ExpedienteRecurso
- + findAll(Pagination, ExpedienteRecurso): List
- + findAllCount(ExpedienteRecurso): Long
- + findAllLike(boolean, Pagination, ExpedienteRecurso): List
- + findAllLikeCount(boolean, ExpedienteRecurso): Long
- + generarNumeroExpediente(ExpedienteRecurso): ExpedienteRecurso
- + remove(ExpedienteRecurso): void
- + update(ExpedienteRecurso): ExpedienteRecurso

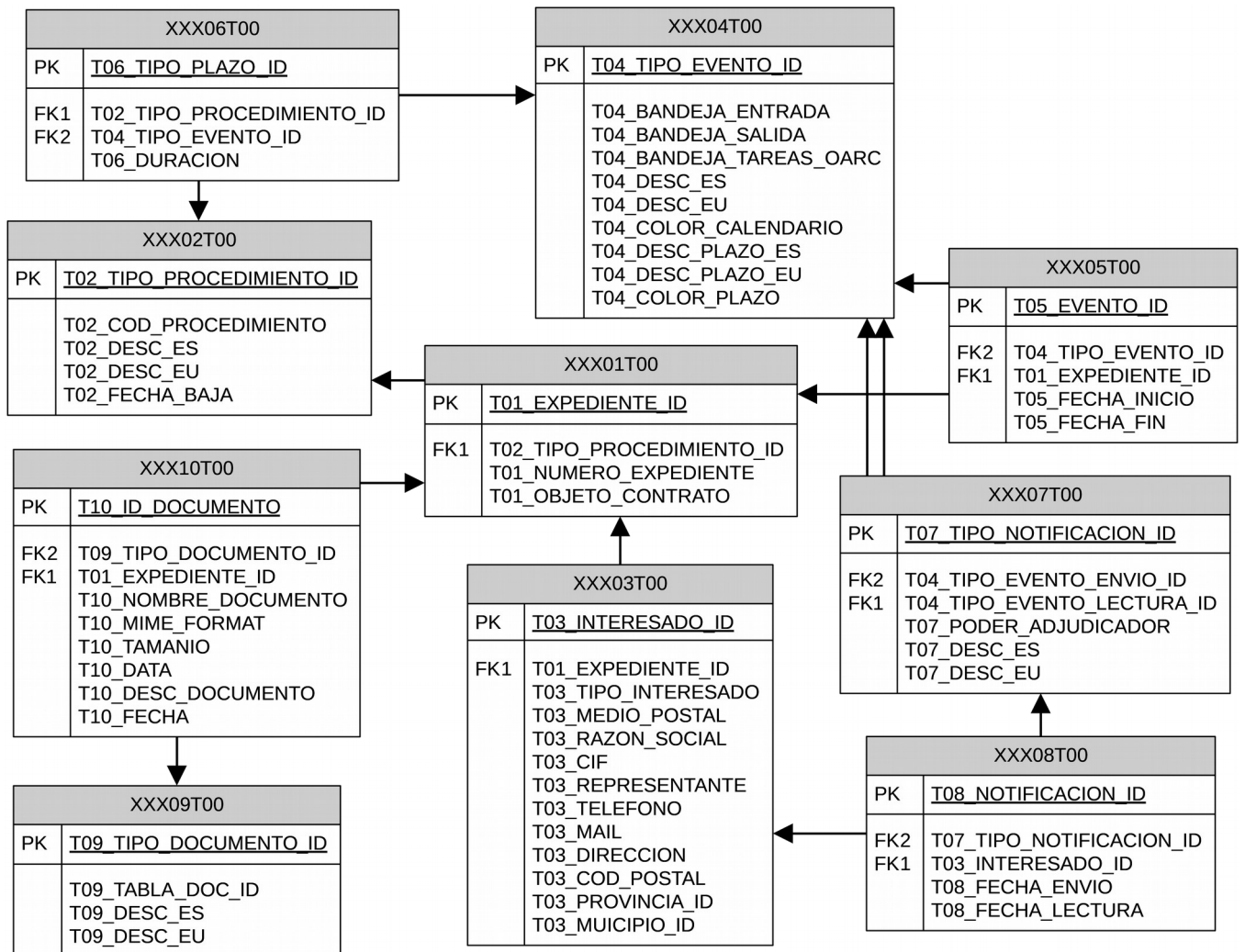
8.1.1. Subsistema 001- Solicitud de recurso



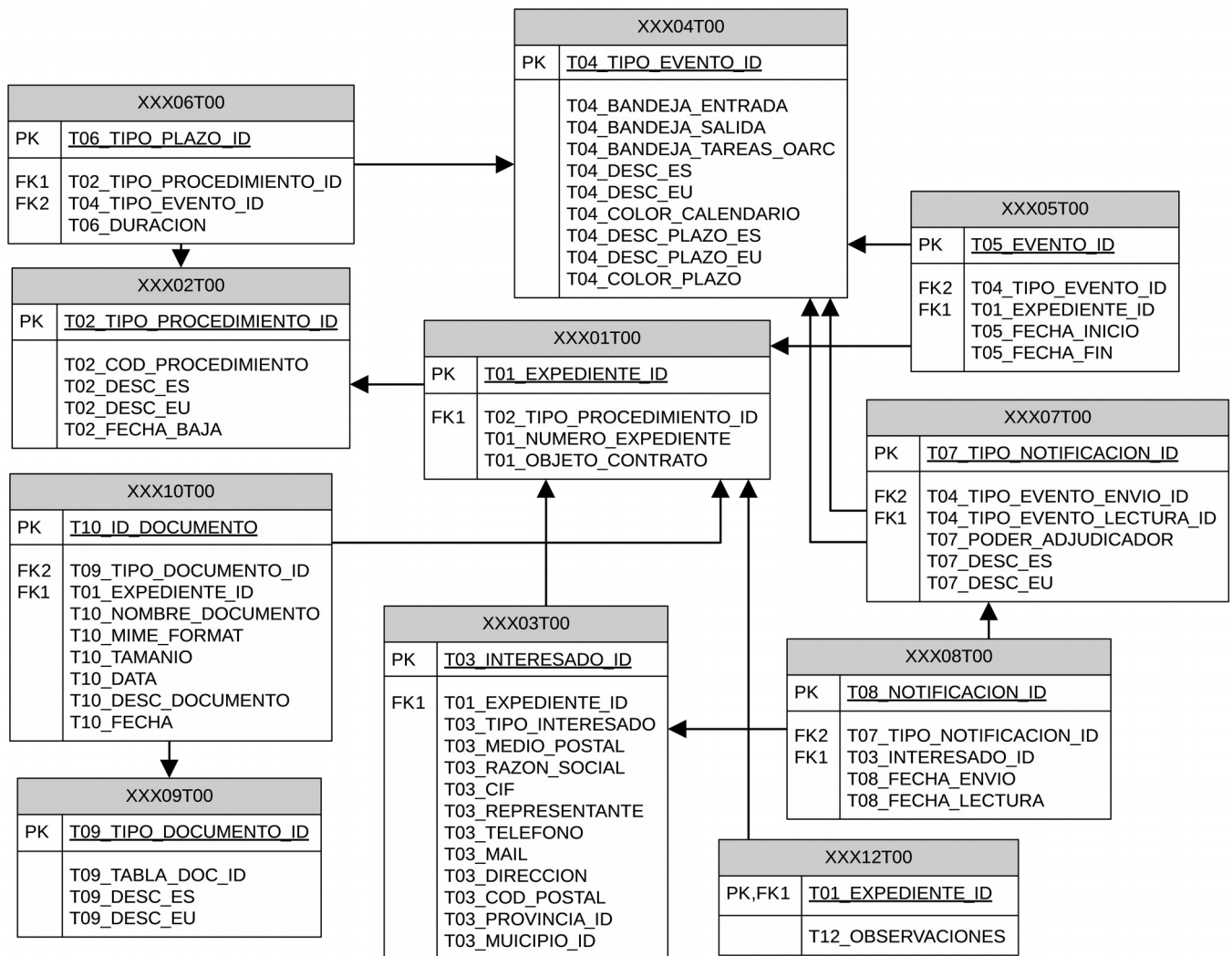
8.1.2. Subsistema 002 - Generación de expediente de recurso



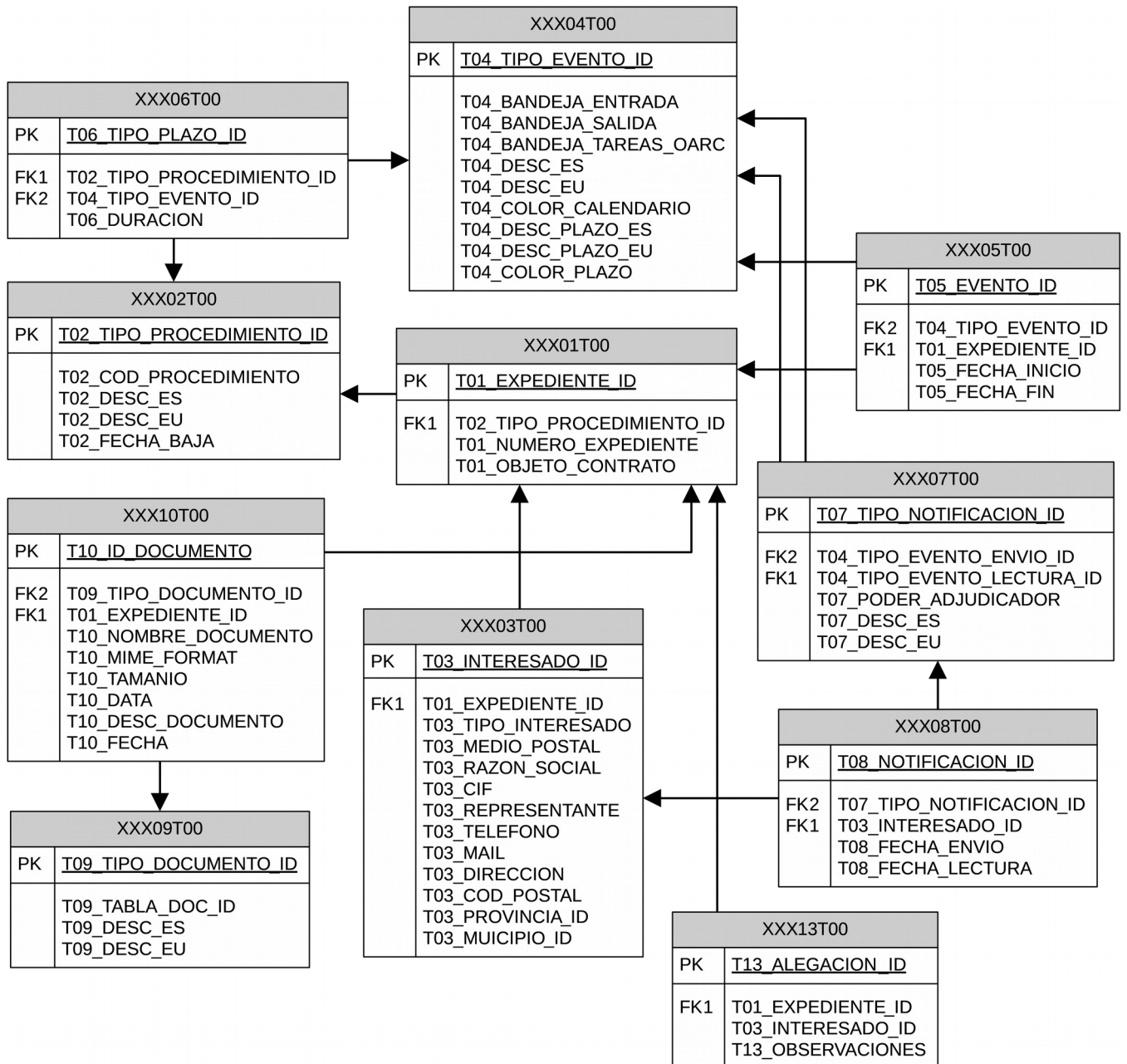
8.1.3. Subsistema 003 - Expediente de contratación



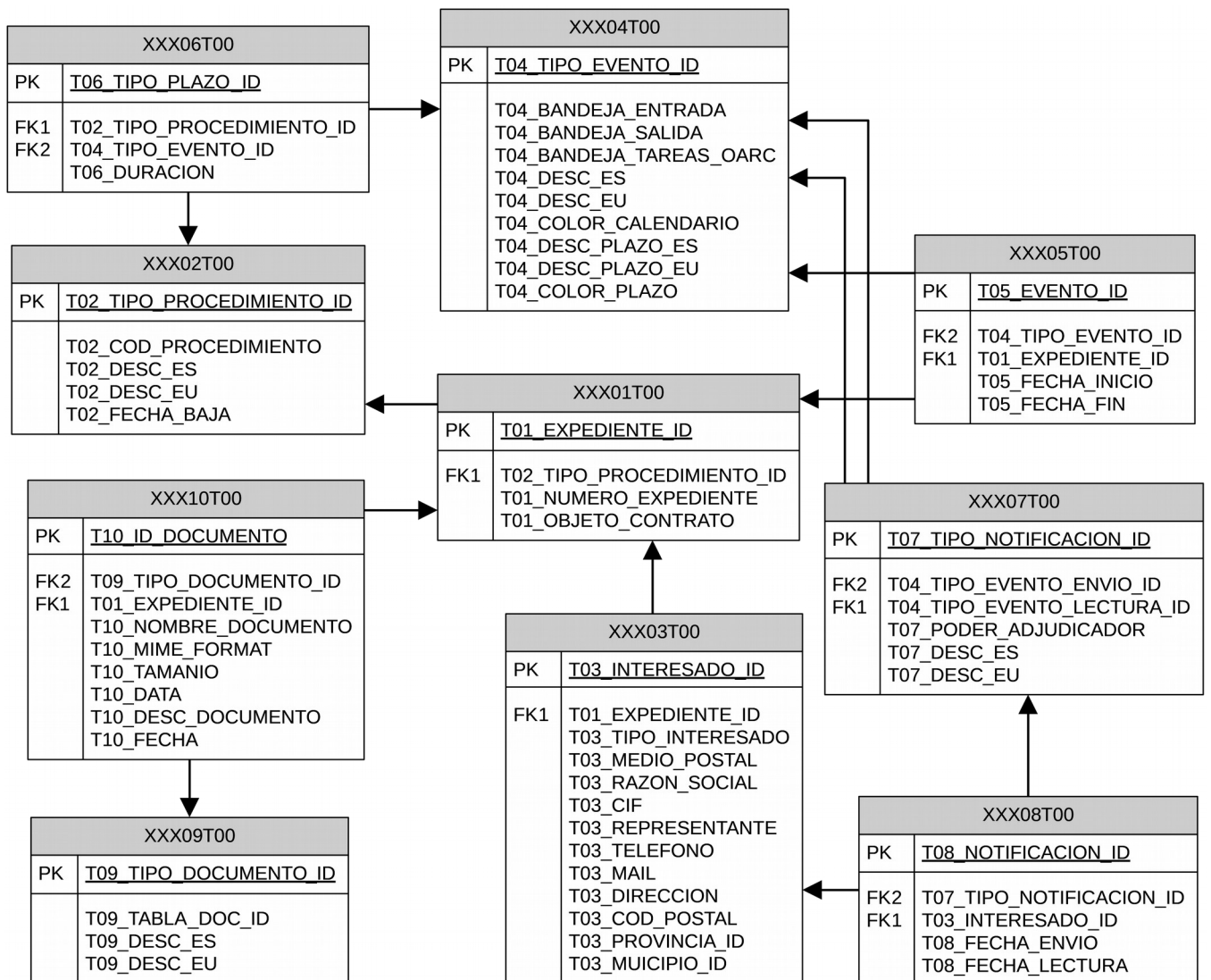
8.1.4. Subsistema 004 – Subsanación



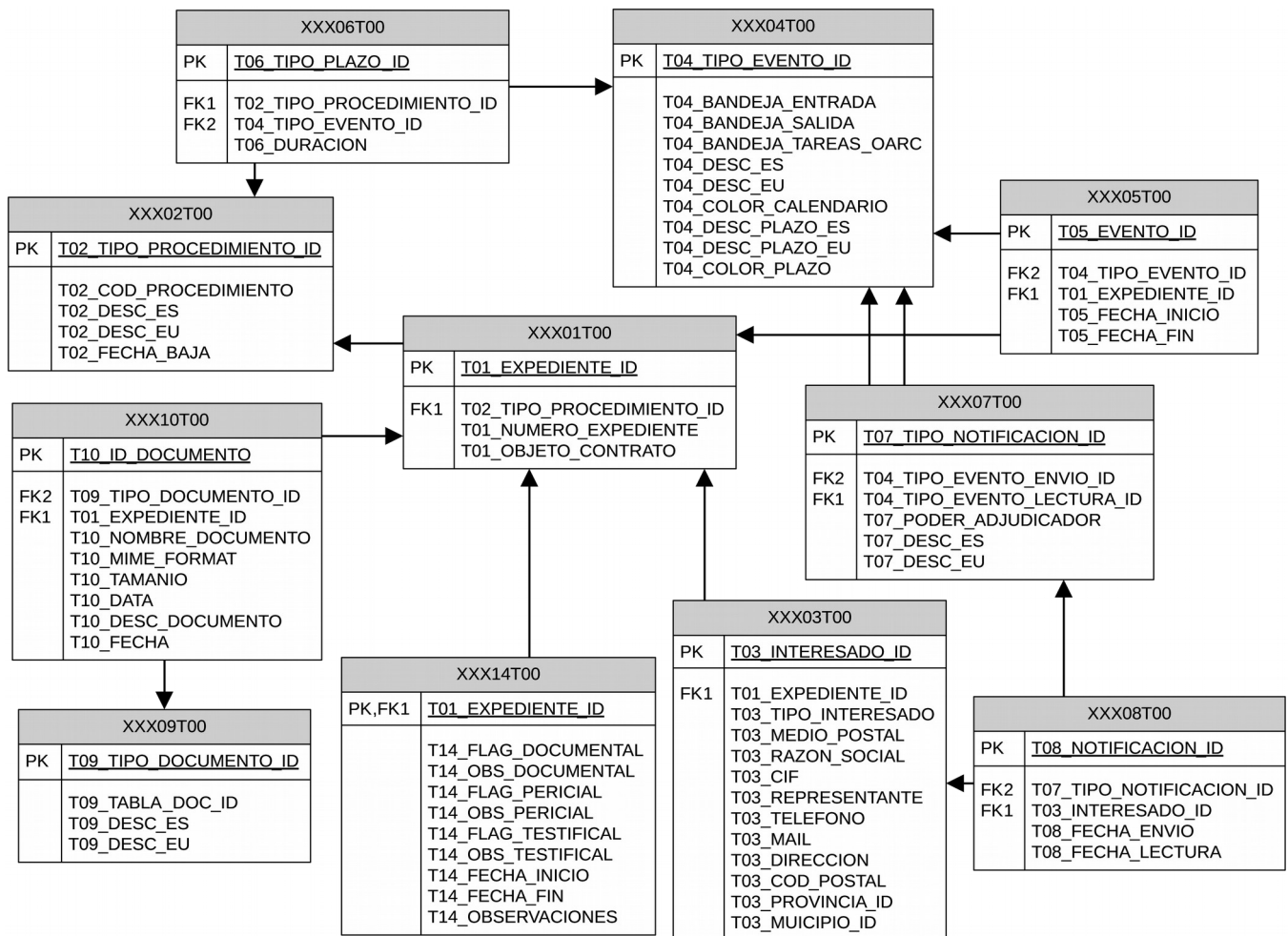
8.1.5. Subsistema 005 - Traslado y alegaciones



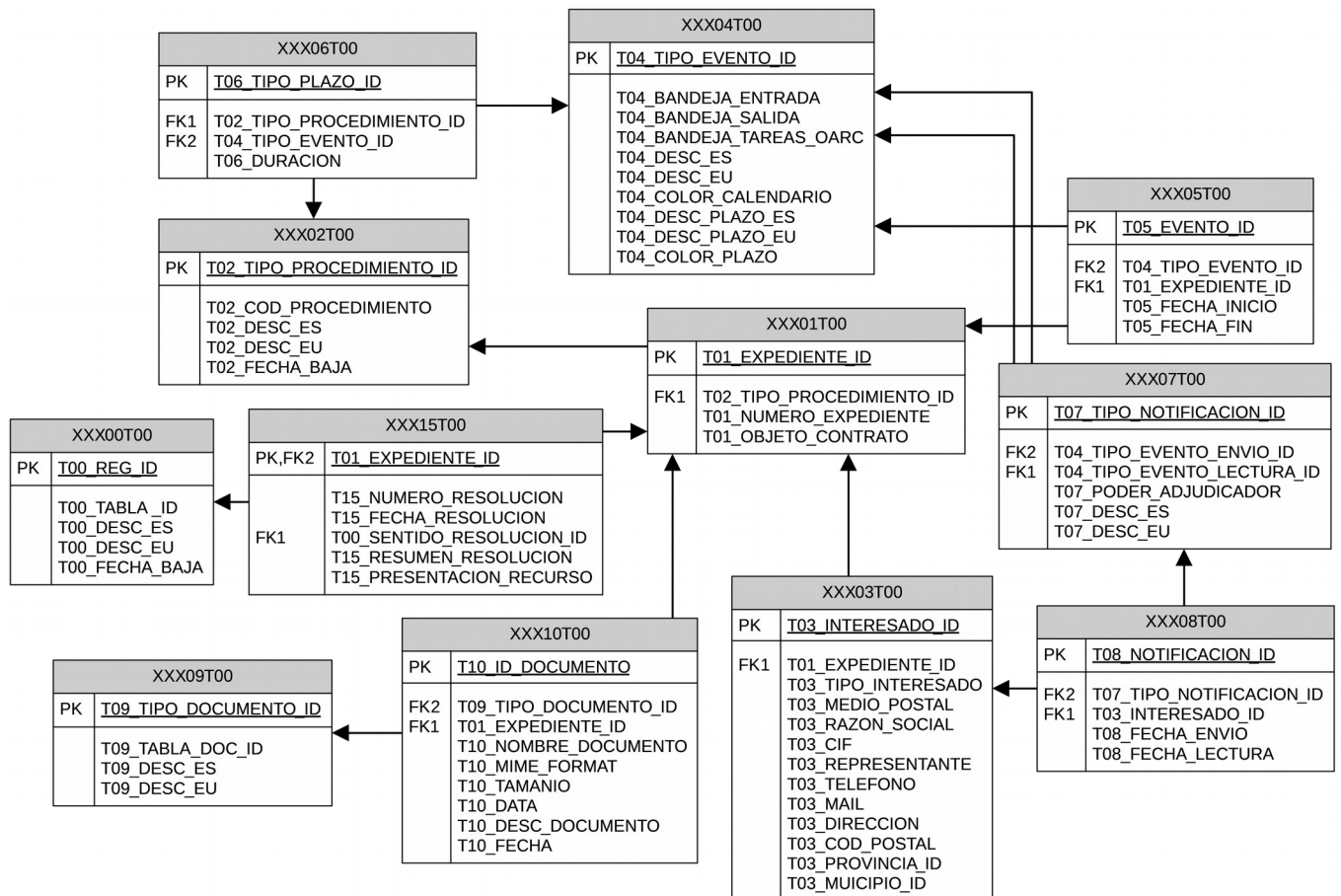
8.1.6. Subsistema 006 - Medidas Provisionales



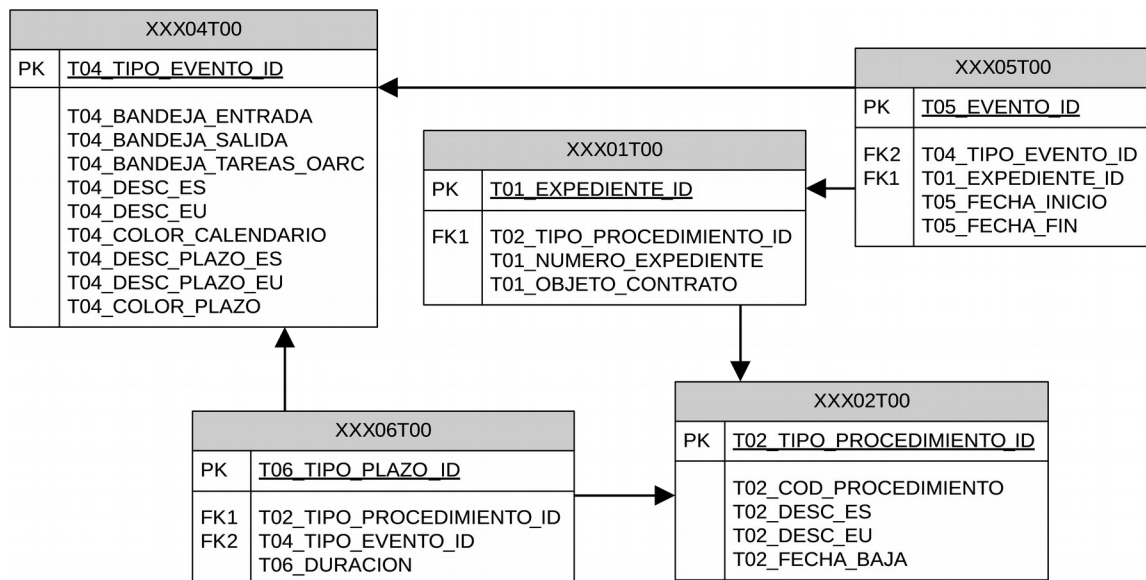
8.1.7. Subsistema 007 - Práctica de prueba



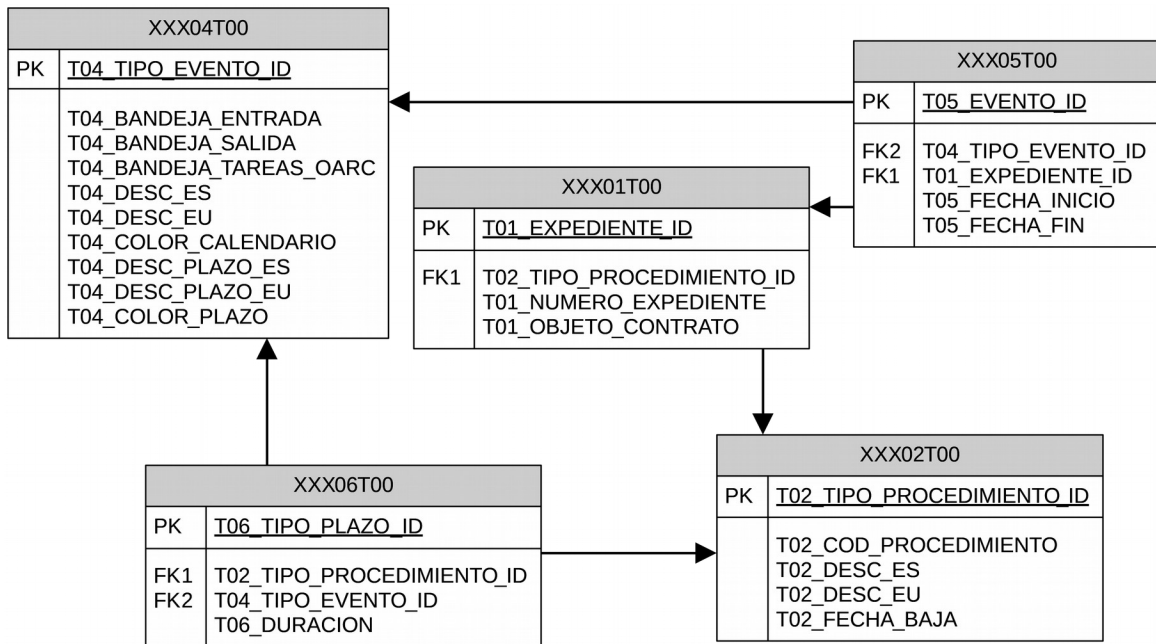
8.1.8. Subsistema 008 - Resolución del recurso



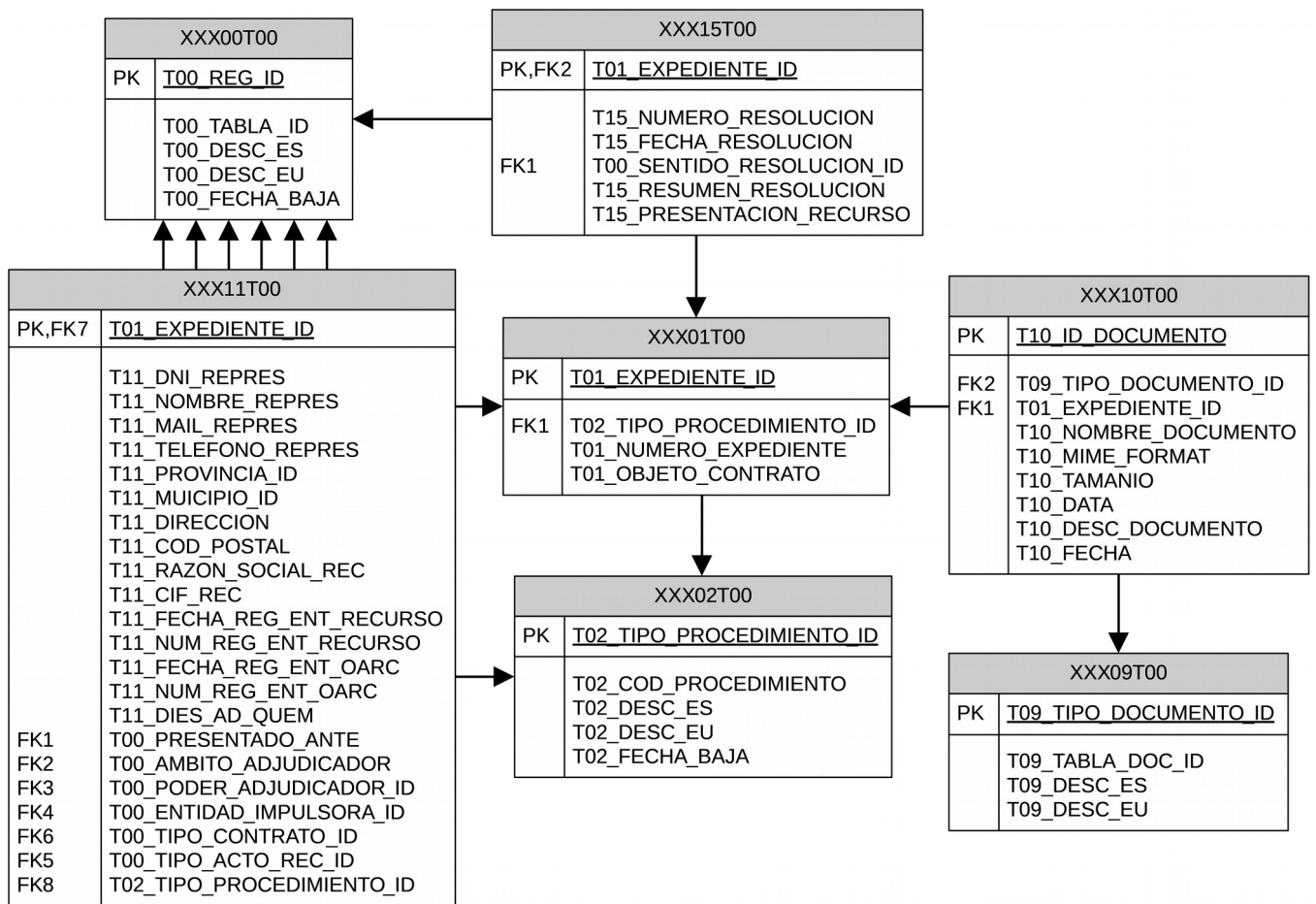
8.1.9. Subsistema 009 – Histórico



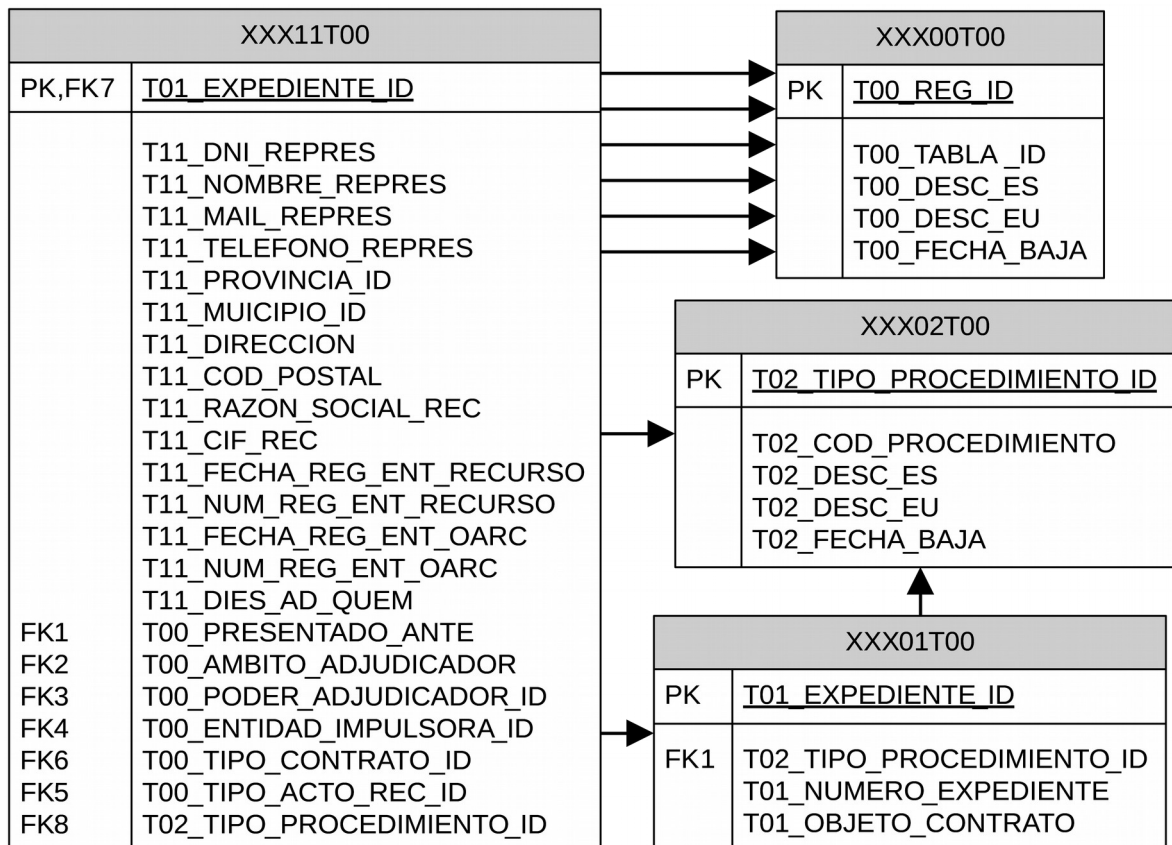
8.1.10. Subsistema 010 - Acceso a expedientes



8.1.11. Subsistema 011 - Consulta pública de resoluciones



8.1.12. Subsistema 012 - Generación de informes



8.2 Descripción de tablas

8.2.1. XXX00T00

DEFINICIÓN DE LA TABLA				
Base de datos	XXXD		Tablespace	XXX01E00
Nombre tabla	XXX00T00		Sinónimo	XXX00S00
Descripción	Maestros Generales			
Tamaño del registro	409		Número estimado de registros	1000
ATRIBUTOS				
Columna	Tipo de datos	Tamaño	Admite nulos	Descripción
T00_TABLA_ID	NUMBER	4	NO	Identificativo de la tabla
T00_REG_ID	NUMBER	10	NO	Identificativo del registro dentro de la tabla
T00_DESC_ES	VARCHAR2	400	YES	Descripcion castellano
T00_DESC_EU	VARCHAR2	400	YES	Descripcion euskera
T00_FECHA_BAJA	DATE	7	YES	Fecha de baja del registro

INDICES			
Nombre	Duplicados	Columnas	Orden
PK_XXX00T00	No	T00_REG_ID	Ascendente

8.2.2. XXX01T00

DEFINICIÓN DE LA TABLA				
Base de datos	XXXD		Tablespace	XXX01E00
Nombre tabla	XXX01T00		Sinónimo	XXX01S00
Descripción	Expedientes			
Tamaño del registro	119		Número estimado de registros	500
ATRIBUTOS				
Columna	Tipo de datos	Tamaño	Admite nulos	Descripción
T01_EXPEDIENTE_ID	NUMBER	10	NO	Identificativo de la tabla
T02_TIPO_PROCEDIMIENTO_ID	NUMBER	4	NO	Identificativo del registro dentro de la tabla
T01_NUMERO_EXPEDIENTE	VARCHAR2	20	YES	Descripcion castellano
T01_OBJETO_CONTRATO	VARCHAR2	200	YES	Descripcion euskera

INDICES			
Nombre	Duplicados	Columnas	Orden
PK_XXX01T00	No	T00_EXPEDIENTE_ID	Ascendente
FK_XXX01T00_XXX02T00_00	Si	T02_TIPO_PROCEDIMIENTO_ID	Ascendente

8.2.3. XXX02T00

DEFINICIÓN DE LA TABLA				
Base de datos	XXXD		Tablespace	XXX01E00
Nombre tabla	XXX02T00		Sinónimo	XXX02S00
Descripción	Tipos de procedimientos			
Tamaño del registro	412		Número estimado de registros	15
ATRIBUTOS				
Columna	Tipo de datos	Tamaño	Admite nulos	Descripción
T02_TIPO_PROCEDIMIENTO_ID	NUMBER	4	NO	Identificativo de la tabla
T02_COD_PROCEDIMIENTO	VARCHAR2	5	YES	Código del procedimiento
T02_DESC_ES	VARCHAR2	400	YES	Descripcion castellano
T02_DESC_EU	VARCHAR2	400	YES	Descripcion euskera
T02_FECHA_BAJA	DATE	7	YES	Fecha de baja del registro

INDICES			
Nombre	Duplicados	Columnas	Orden
PK_XXX01T00	No	T00_EXPEDIENTE_ID	Ascendente
FK_XXX01T00_XXX02T00_00	Si	T02_TIPO_PROCEDIMIENTO_ID	Ascendente

8.2.4. XXX03T00

DEFINICIÓN DE LA TABLA				
Base de datos	XXXD		Tablespace	XXX01E00
Nombre tabla	XXX03T00		Sinónimo	XXX03S00
Descripción	Interesados. Cuando se da de alta una solicitud se crean dos entradas en esta tabla, una para el recurrente y otra para el poder adjudicador			
Tamaño del registro	329		Número estimado de registros	10000
ATRIBUTOS				
Columna	Tipo de datos	Tamaño	Admite nulos	Descripción
T03_INTERESADO_ID	NUMBER	10	NO	Identificativo de la tabla
T01_EXPEDIENTE_ID	NUMBER	10	NO	Identificativo del expediente
T03_TIPO_INTERESADO	NUMBER	4	NO	Tipo interesado. 1-->Normal, 2-->Recurrente, 3-->Poder adjudicador
T03_MEDIO_POSTAL	NUMBER	1	NO	Flag medio postal. 0-->No, 1-->Si
T03_RAZON_SOCIAL	VARCHAR2	200	YES	Razón social
T03_CIF	VARCHAR2	20	YES	CIF
T03_REPRESENTANTE	VARCHAR2	100	YES	Representante
T03_TELEFONO	VARCHAR2	20	YES	Teléfono
T03_MAIL	VARCHAR2	100	YES	Email
T03_DIRECCION	VARCHAR2	150	YES	Dirección
T03_COD_POSTAL	VARCHAR2	10	YES	Código postal
T03_PROVINCIA_ID	NUMBER	10	YES	Identificativo de la Provincia
T03_MUICIPIO_ID	NUMBER	10	YES	Identificativo del Municipio

INDICES			
Nombre	Duplicados	Columnas	Orden
PK_XXX03T00	No	T03_INTERESADO_ID	Ascendente
FK_XXX03T00_XXX01T00_00	Si	T01_EXPEDIENTE_ID	Ascendente

8.2.5. XXX04T00

DEFINICIÓN DE LA TABLA				
Base de datos	XXXD		Tablespace	XXX01E00
Nombre tabla	XXX04T00		Sinónimo	XXX04S00
Descripción	Tipo de Eventos. Recoge los eventos que se producen en un expediente. También guarda los plazos que lanza cada evento			
Tamaño del registro	819		Número estimado de registros	30
ATRIBUTOS				
Columna	Tipo de datos	Tamaño	Admite nulos	Descripción
T04_TIPO_EVENTO_ID	NUMBER	4	NO	Identificativo de la tabla del tipo de evento
T04_BANDEJA_ENTRADA	NUMBER	1	YES	Flag bandeja entrada
T04_BANDEJA_SALIDA	NUMBER	1	YES	Flag bandeja salida
T04_BANDEJA_TAREAS_OARC	NUMBER	1	YES	Flag bandeja tareas OARC
T04_DESC_ES	VARCHAR2	400	YES	Descripcion castellano
T04_DESC_EU	VARCHAR2	400	YES	Descripcion euskera
T04_COLOR_CALENDARIO	VARCHAR2	10	YES	Color del evento en el calendario del expediente

T04_DESC_PLAZO_ES	VARCHAR2	400	YES	Descripción plazo castellano
T04_DESC_PLAZO_EU	VARCHAR2	400	YES	Descripción plazo euskera
T04_COLOR_PLAZO	VARCHAR2	10	YES	Color del plazo en la bandeja de tareas del OARC

INDICES			
Nombre	Duplicados	Columnas	Orden
PK_XXX04T00	No	T04_TIPO_EVENTO_ID	Ascendente

8.2.6. XXX05T00

DEFINICIÓN DE LA TABLA				
Base de datos	XXXD		Tablespace	XXX01E00
Nombre tabla	XXX05T00		Sinónimo	XXX05S00
Descripción	Eventos. Recoge todos los eventos de los expedientes.			
Tamaño del registro	29		Número estimado de registros	21000
ATRIBUTOS				
Columna	Tipo de datos	Tamaño	Admite nulos	Descripción
T05_EVENTO_ID	NUMBER	10	NO	Identificativo de la tabla
T04_TIPO_EVENTO_ID	NUMBER	4	NO	Identificativo del tipo de evento que tipifica el evento.
T01_EXPEDIENTE_ID	NUMBER	10	YES	Identificativo del expediente al que pertenece el evento
T05_FECHA_INICIO	DATE	7	YES	Fecha inicio. Fecha en que se produce el evento
T05_FECHA_FIN	DATE	7	YES	Fecha fin. En el caso de que el evento dispare un plazo, esta fecha recoge el fin de

				dicho plazo.
--	--	--	--	--------------

INDICES			
Nombre	Duplicados	Columnas	Orden
PK_XXX05T00	No	T05_ EVENTO _ID	Ascendente
FK_XXX05T00_XXX04T00_00	Si	T04_TIPO_EVENTO_ID	Ascendente
FK_XXX05T00_XXX01T00_00	Si	T01_EXPEDIENTE_ID	Ascendente

8.2.7. XXX06T00

DEFINICIÓN DE LA TABLA				
Base de datos	XXXD		Tablespace	XXX01E00
Nombre tabla	XXX06T00		Sinónimo	XXX06S00
Descripción	Tipo de Plazos. Reúne los diferentes tipos de plazos que se muestran en la bandeja de tareas del OARC.			
Tamaño del registro	14		Número estimado de registros	60
ATRIBUTOS				
Columna	Tipo de datos	Tamaño	Admite nulos	Descripción
T06_TIPO_PLAZO_ID	NUMBER	8	NO	Identificativo de la tabla
T02_TIPO_PROCEDIMIENTO_ID	NUMBER	4	NO	Identificativo del tipo de procedimiento
T04_TIPO_EVENTO_ID	NUMBER	4	NO	Identificativo del tipo de evento, que contiene la descripción del plazo
T06_DURACION	NUMBER	4	YES	Duración del plazo

INDICES			
Nombre	Duplicados	Columnas	Orden
PK_XXX06T00	No	T06_TIPO_PLAZO_ID	Ascendente
FK_XXX06T00_XXX02T00_00	Si	T02_TIPO_PROCEDIMIENTO_ID	Ascendente
FK_XXX06T00_XXX04T00_00	Si	T04_TIPO_EVENTO_ID	Ascendente

8.2.8. XXX07T00

DEFINICIÓN DE LA TABLA				
Base de datos	XXXD		Tablespace	XXX01E00
Nombre tabla	XXX07T00		Sinónimo	XXX07S00
Descripción	Tipo de Notificaciones			
Tamaño del registro	417		Número estimado de registros	10
ATRIBUTOS				
Columna	Tipo de datos	Tamaño	Admite nulos	Descripción
T07_TIPO_NOTIFICACION_ID	NUMBER	4	NO	Identificativo de la tabla
T04_TIPO_EVENTO_ENVIO_ID	NUMBER	10	YES	Identificativo del tipo de evento que se dispara con el envío de la notificación. Relación usada para la automatización de notificaciones
T04_TIPO_EVENTO_LECTURA_ID	NUMBER	10	YES	Identificativo del tipo de evento que se dispara con la lectura de la notificación. Relación usada para la automatización de notificaciones
T07_PODER_ADJUDICA	NUMBER	1	YES	Flag que indica si la notificación se envía al

DOR				poder adjudicador
T07_DESC_ES	VARCHAR2	400	YES	Descripcion castellano
T07_DESC_EU	VARCHAR2	400	YES	Descripcion euskera

INDICES			
Nombre	Duplicados	Columnas	Orden
PK_XXX07T00	No	T07_TIPO_NOTIFICACION_ID	Ascendente
FK_XXX07T00_XXX04T00_00	Si	T04_TIPO_EVENTO_ENVIO_ID	Ascendente
FK_XXX07T00_XXX04T00_01	Si	T04_TIPO_EVENTO_LECTURA_ID	Ascendente

8.2.9. XXX08T00

DEFINICIÓN DE LA TABLA				
Base de datos	XXXD		Tablespace	XXX01E00
Nombre tabla	XXX08T00		Sinónimo	XXX08S00
Descripción	Notificaciones			
Tamaño del registro	29		Número estimado de registros	8000
ATRIBUTOS				
Columna	Tipo de datos	Tamaño	Admite nulos	Descripción
T08_NOTIFICACION_ID	NUMBER	10	NO	Identificativo de la tabla
T07_TIPO_NOTIFICACION_ID	NUMBER	4	NO	Identificativo del tipo de notificación

T03_INTERESADO_ID	NUMBER	10	NO	Identificativo del interesado
T08_FECHA_ENVIO	DATE	7	YES	Fecha de envío de la notificación
T08_FECHA_LECTURA	DATE	7	YES	Fecha de lectura de la notificación

INDICES			
Nombre	Duplicados	Columnas	Orden
PK_XXX08T00	No	T08_NOTIFICACION_ID	Ascendente
FK_XXX08T00_XXX07T00_00	Si	T07_TIPO_NOTIFICACION_ID	Ascendente
FK_XXX08T00_XXX03T00_00	Si	T03_INTERESADO_ID	Ascendente

8.2.10. XXX09T00

DEFINICIÓN DE LA TABLA				
Base de datos	XXXD		Tablespace	XXX01E00
Nombre tabla	XXX09T00		Sinónimo	XXX09S00
Descripción	Tipos de documentos			
Tamaño del registro	406		Número estimado de registros	25
ATRIBUTOS				
Columna	Tipo de datos	Tamaño	Admite nulos	Descripción
T09_TIPO_DOCUMENTO_ID	NUMBER	4	NO	Identificativo de la tabla
T09_TABLA_DOC_ID	NUMBER	4	YES	Identificativo de la tabla de documentos en la que aparece el documento
T09_DESC_ES	VARCHAR2	400	YES	Descripcion castellano
T09_DESC_EU	VARCHAR2	400	YES	Descripcion euskera

--	--	--	--	--

INDICES			
Nombre	Duplicados	Columnas	Orden
PK_XXX09T00	No	T09_TIPO_DOCUMENTO_ID	Ascendente

8.2.11. XXX10T00

DEFINICIÓN DE LA TABLA				
Base de datos	XXXD		Tablespace	XXX01E00
Nombre tabla	XXX10T00		Sinónimo	XXX10S00
Descripción	Documentos. Alberga la información de cada documento albergado en el sistema			
Tamaño del registro	428		Número estimado de registros	
ATRIBUTOS				
Columna	Tipo de datos	Tamaño	Admite nulos	Descripción
T10_ID_DOCUMENTO	NUMBER	10	NO	Identificativo de la tabla
T09_TIPO_DOCUMENTO_ID	NUMBER	4	NO	Identificativo
T01_EXPEDIENTE_ID	NUMBER	10	YES	Año de la convocatoria
T10_NOMBRE_DOCUMENTO	VARCHAR2	200	NO	Nombre del documento
T10_MIME_FORMAT	VARCHAR2	100	NO	Formato mime del documento
T10_TAMANIO	NUMBER	10	NO	Tamaño del documento
T10_DATA	BLOB	4000	YES	Datos binarios del documento

T10_DESC_DOCUMENTO	VARCHAR2	500	YES	Descripción del documento
T10_FECHA	DATE	7	YES	Fecha de subida del documento

INDICES			
Nombre	Duplicados	Columnas	Orden
PK_XXX10T00	No	T10_ID_DOCUMENTO	Ascendente
FK_XXX10T00_XXX09T00_00	Si	T09_TIPO_DOCUMENTO_ID	Ascendente
FK_XXX10T00_XXX01T00_00	Si	T01_EXPEDIENTE_ID	Ascendente

8.2.12. XXX11T00

DEFINICIÓN DE LA TABLA				
Base de datos	XXXD		Tablespace	XXX01E00
Nombre tabla	XXX11T00		Sinónimo	XXX11S00
Descripción	Datos de la Solictud. Alberga la información de la solicitud de expediente, esta tabla se ha generado para descargar a la tabla de expedientes de un buen número de campos agilizando el resto de consultas. Por ello tienen una relación de 1 a 1.			
Tamaño del registro	442		Número estimado de registros	500
ATRIBUTOS				
Columna	Tipo de datos	Tamaño	Admite nulos	Descripción
T01_EXPEDIENTE_ID	NUMBER	10	NO	Identificativo del expediente

T11_DNI_REPRES	VARCHAR2	20	YES	DNI del representante
T11_NOMBRE_REPRES	VARCHAR2	200	YES	Nombre del representante
T11_MAIL_REPRES	VARCHAR2	100	YES	Email del representante
T11_TELEFONO_REPRES	VARCHAR2	20	YES	Teléfono del representante
T11_PROVINCIA_ID	NUMBER	10	YES	Identificativo de la Provincia del representante
T11_MUICIPIO_ID	NUMBER	10	YES	Identificativo del Municipio del representante
T11_DIRECCION	VARCHAR2	150	YES	Dirección del representante
T11_COD_POSTAL	VARCHAR2	10	YES	Código postal del representante
T11_RAZON_SOCIAL_REC	VARCHAR2	200	YES	Razón social del recurrente
T11_CIF_REC	VARCHAR2	20	YES	CIF del recurrente
T11_FECHA_REG_ENT_RECURSO	DATE	7	YES	Fecha de registro de entrada del recurso
T11_NUM_REG_ENT_RECURSO	NUMBER	20	YES	Número de registro de entrada del recurso
T11_FECHA_REG_ENT_OARC	DATE	7	YES	Fecha de registro de entrada de la solicitud en el OARC
T11_NUM_REG_ENT_OARC	NUMBER	20	YES	Número de registro de entrada de la solicitud en el OARC
T11_DIES_AD_QUEM	NUMBER	4	YES	Dies ad quem, día hasta el que (fin de un plazo)
T00_PRESENTADO_ANTE	NUMBER	1	YES	Recurso presentado ante: 0-->OARC, 1-->Poder Adjudicador
T00_AMBITO_ADJUDICADOR	NUMBER	1	YES	Ámbito del Poder Adjudicador: 0-->CAPV, 1-->Araba/Álava, 0-->Gipuzkoa, 1-->Bizkaia
T00_PODER_ADJUDICADOR_ID	NUMBER	10	YES	Identificativo del Poder Adjudicador
T00_ENTIDAD_IMPULSORA_ID	NUMBER	10	YES	Identificativo de la entidad impulsora
T00_TIPO_CONTRATO_I	NUMBER	4	YES	Identificativo del tipo de contrato

D				
T00_TIPO_ACTO_REC_ID	NUMBER	4	YES	Identificativo del tipo de acto recurrido
T02_TIPO_PROCEDIMIENTO_ID	NUMBER	4	YES	Identificativo del tipo de procedimiento

INDICES			
Nombre	Duplicados	Columnas	Orden
PK_XXX11T00	No	T01_EXPEDIENTE_ID	Ascendente
FK_XXX11T00_XXX01T00_00	Si	T01_EXPEDIENTE_ID	Ascendente
FK_XXX11T00_XXX00T00_00	Si	T00_PRESENTADO_ANTE	Ascendente
FK_XXX11T00_XXX00T00_01	Si	T00_AMBITO_ADJUDICADOR	Ascendente
FK_XXX11T00_XXX00T00_02	Si	T00_PODER_ADJUDICADOR_ID	Ascendente
FK_XXX11T00_XXX00T00_03	Si	T00_TIPO_CONTRATO_ID	Ascendente
FK_XXX11T00_XXX00T00_04	Si	T00_TIPO_ACTO_REC_ID	Ascendente
FK_XXX11T00_XXX02T00_00	Si	T02_TIPO_PROCEDIMIENTO_ID	Ascendente

8.2.13. XXX12T00

DEFINICIÓN DE LA TABLA				
Base de datos	XXXD		Tablespace	XXX01E00
Nombre tabla	XXX12T00		Sinónimo	XXX12S00
Descripción	Datos de Subsanación. Relación 1 a 1 con la tabla de expedientes. Es una tabla que crecerá en número de campos a lo largo de la vida de la aplicación.			
Tamaño del registro	2006		Número estimado de registros	500
ATRIBUTOS				
Columna	Tipo de datos	Tamaño	Admite nulos	Descripción
T01_EXPEDIENTE_ID	NUMBER	10	NO	Identificativo del expediente
T12_OBSERVACIONES	VARCHAR2	4000	YES	Observaciones de la subsanación

INDICES			
Nombre	Duplicados	Columnas	Orden
PK_XXX12T00	No	T01_EXPEDIENTE_ID	Ascendente
FK_XXX12T00_XXX01T00_00	Si	T01_EXPEDIENTE_ID	Ascendente

8.2.14. XXX13T00

DEFINICIÓN DE LA TABLA				
Base de datos	XXXD		Tablespace	XXX01E00
Nombre tabla	XXX13T00		Sinónimo	XXX13S00
Descripción	Alegaciones. Relación de alegaciones presentadas a un recurso			
Tamaño del registro	2018		Número estimado de registros	3000
ATRIBUTOS				
Columna	Tipo de datos	Tamaño	Admite nulos	Descripción
T13_ALEGACION_ID	NUMBER	10	NO	Identificativo de la tabla
T01_EXPEDIENTE_ID	NUMBER	10	NO	Identificativo del expediente
T03_INTERESADO_ID	NUMBER	10	NO	Identificativo del interesado que realiza la alegación
T13_OBSERVACIONES	VARCHAR2	4000	YES	Observaciones de la alegación

INDICES			
Nombre	Duplicados	Columnas	Orden
PK_XXX13T00	No	T13_ALEGACION_ID	Ascendente
FK_XXX13T00_XXX01T00_00	Si	T01_EXPEDIENTE_ID	Ascendente
FK_XXX13T00_XXX03T00_00	Si	T03_INTERESADO_ID	Ascendente

8.2.15. XXX14T00

DEFINICIÓN DE LA TABLA				
Base de datos	XXXD		Tablespace	XXX01E00
Nombre tabla	XXX14T00		Sinónimo	XXX14S00
Descripción	Práctica de prueba. Relacionada 1 a 1 con la tabla de expedientes para aligerarla de campos y mejorar su rendimiento en consultas. Representa los campos del módulo de práctica de prueba que sólo se piden en dicho módulo.			
Tamaño del registro	8026		Número estimado de registros	500
ATRIBUTOS				
Columna	Tipo de datos	Tamaño	Admite nulos	Descripción
T01_EXPEDIENTE_ID	NUMBER	10	NO	Identificativo de la tabla
T14_FLAG_DOCUMENTAL	NUMBER	1	YES	Flag que indica si la práctica de pruebas tendrá datos de tipo documental
T14_OBS_DOCUMENTAL	VARCHAR2	4000	YES	Observaciones de tipo documental de la práctica de prueba
T14_FLAG_PERICIAL	NUMBER	1	YES	Flag que indica si la práctica de pruebas tendrá datos de tipo documental
T14_OBS_PERICIAL	VARCHAR2	4000	YES	Observaciones de tipo documental de la práctica de prueba
T14_FLAG_TESTIFICAL	NUMBER	1	YES	Flag que indica si la práctica de pruebas tendrá datos de tipo documental
T14_OBS_TESTIFICAL	VARCHAR2	4000	YES	Observaciones de tipo documental de la práctica de prueba
T14_FECHA_INICIO	DATE	7	YES	Fecha inicio de la práctica de prueba
T14_FECHA_FIN	DATE	7	YES	Fecha fin de la práctica de prueba
T14_OBSERVACIONES	VARCHAR2	4000	YES	Observaciones de la práctica de prueba

INDICES			
Nombre	Duplicados	Columnas	Orden
PK_XXX14T00	No	T01_EXPEDIENTE_ID	Ascendente
FK_XXX14T00_XXX01T00_00	Si	T01_EXPEDIENTE_ID	Ascendente

8.2.16. XXX15T00

DEFINICIÓN DE LA TABLA				
Base de datos	XXXD		Tablespace	XXX01E00
Nombre tabla	XXX15T00		Sinónimo	XXX15S00
Descripción	Resoluciones. Relacionada 1 a 1 con la tabla de expedientes para aligerarla de campos y mejorar su rendimiento en consultas. Representa los campos del módulo de resolución de un expediente de recurso.			
Tamaño del registro	2036		Número estimado de registros	500
ATRIBUTOS				
Columna	Tipo de datos	Tamaño	Admite nulos	Descripción
T01_EXPEDIENTE_ID	NUMBER	10	NO	Identificativo del expediente
T15_NUMERO_RESOLUCION	NUMBER	10	YES	Número de la resolución
T15_FECHA_RESOLUCION	DATE	7	YES	Fecha de la resolución
T00_SENTIDO_RESOLUCION_ID	NUMBER	10	YES	Sentido de la resolución
T15_RESUMEN_RESOLUCION	VARCHAR2	4000	YES	Resumen de la resolución
T15_PRESENTACION_RECURSO	DATE	7	YES	Fecha de presentación del recurso contra la resolución

INDICES			
Nombre	Duplicados	Columnas	Orden
PK_XXX15T00	No	T01_EXPEDIENTE_ID	Ascendente
FK_XXX15T00_XXX01T00_00	Si	T01_EXPEDIENTE_ID	Ascendente
FK_XXX15T00_XXX00T00_00	Si	T00_SENTIDO_RESOLUCION_ID	Ascendente

8.2.17. XXX16T00

DEFINICIÓN DE LA TABLA				
Base de datos	XXXD		Tablespace	XXX01E00
Nombre tabla	XXX16T00		Sinónimo	XXX16S00
Descripción	Links con la Documentación de Referencia en la que se ha basado la resolución.			
Tamaño del registro	112		Número estimado de registros	3000
ATRIBUTOS				
Columna	Tipo de datos	Tamaño	Admite nulos	Descripción
T16_LINK_ID	NUMBER	10	NO	Identificativo de la tabla
T01_EXPEDIENTE_ID	NUMBER	10	NO	Identificativo del expediente
T16_NOMBRE_LINK	VARCHAR2	200	YES	Nombre del link
T16_DESCRIPCION_LINK	VARCHAR2	200	YES	Descripción del link

INDICES			
Nombre	Duplicados	Columnas	Orden
PK_XXX16T00	No	T16_LINK_ID	Ascendente

FK_XXX16T00_XXX01T00_00	Si	T01_EXPEDIENTE_ID	Ascendente
-------------------------	----	-------------------	------------

9.4 Subsistema 004 - Subsanción

[illegible]

9.5 Subsistema 005 - Traslado y alegaciones

[illegible]

9.6 Subsistema 006 - Medidas Provisionales

[illegible]

9.7 Subsistema 007 - Práctica de prueba

[illegible]

9.8 Subsistema 008 - Resolución del recurso

[illegible]

9.9 Subsistema 009 - Histórico

[illegible]

