



Documento de arquitectura y solución técnica

Creación:	11/10/2016	Autor:	SP
Revisado:		Aprobado:	
Versión:	1.9	Plantilla:	
Código:	AG-SAE-DA-ES-01	Página:	1 de 37

Índice de contenido

1	Introducción.....	3
2	Propósito.....	3
3	Audiencia.....	3
4	Antecedentes.....	3
5	Vistas.....	4
5.1	Vista de Casos de Uso.....	4
5.1.1	Listado de Casos de Uso.....	4
5.2	Vista Física.....	7
5.2.1	Diagrama de Despliegue.....	7
5.3	Vista Lógica.....	10
5.3.1	Diagrama de Secuencia 1: Autenticación.....	10
5.3.2	Diagrama de Secuencia 2: Cambio de empresa.....	13
5.3.3	Diagrama de Secuencia 3: Crear empresa.....	14
5.3.4	Diagrama de Secuencia 4: Modificar empresa.....	14
5.3.5	Diagrama de Secuencia 5: Crear usuario.....	16
5.3.6	Diagrama de Secuencia 6: Crear agenda.....	16
5.3.7	Diagrama de Secuencia 7: Crear recurso.....	17
5.3.8	Diagrama de Secuencia 7: Generar disponibilidad para un día.....	18
5.3.9	Diagrama de Secuencia 8: Copiar disponibilidades.....	19
5.3.10	Diagrama de Secuencia 9: Mostrar llamador.....	20
5.3.11	Diagrama de Secuencia 10: Realizar reserva pública.....	21
5.4	Vista de base de datos.....	21
5.4.1	Esquema global.....	21
5.4.2	Esquema específico de cada tenant.....	22
5.5	Vista de interoperabilidad.....	25
5.5.1	Interoperabilidad con TrámitesUy.....	25
5.5.2	Interoperabilidad con Trazabilidad.....	26
5.6	Vista de Control de acceso.....	30
5.6.1	La agenda accedida no requiere control de acceso.....	31
5.6.2	La agenda accedida requiere control de acceso, es el primer acceso del usuario y el usuario no se ha autenticado ante CDA aún.....	31
5.6.3	La agenda accedida requiere control de acceso, no es el primer acceso del usuario y el usuario ya se ha autenticado ante CDA.....	33
5.6.4	La agenda accedida requiere control de acceso, es el primer acceso del usuario y el usuario ya se ha autenticado ante CDA anteriormente (por solicitud de otra aplicación).....	34
5.6.5	Cierre de sesión.....	35
6	Requerimientos de Hardware.....	35
7	Consideraciones Tecnológicas.....	36
8	Historial de cambios.....	37

1 Introducción

Este documento presenta una descripción de la Arquitectura de la aplicación Sistema de Agenda Electrónica (SAE) para Agesic, la cual es a su vez una adaptación del sistema IMMSAE (Sistema de Agenda Electrónica de la Intendencia de Montevideo), con el objetivo de establecer los elementos que componen el sistema y las interfaces que se utilizan para la comunicación entre los mismos y agentes externos.

2 Propósito

Este documento describe la solución propuesta e implementada desde varios puntos de vista.

3 Audiencia

Este documento está dirigido a los stakeholders del proyecto, arquitectos de software, personal de desarrollo y encargados de la gestión del proyecto.

4 Antecedentes

El Sistema de Agenda Electrónica, en adelante SAE, es una aplicación para gestionar reservas para ciertos trámites y oficinas. Los elementos claves de la aplicación son los siguientes:

- **Agendas:** una agenda es la representación de uno o más trámites que pueden ser realizados por los ciudadanos ante un organismo.
- **Recursos:** un recurso es la representación de una oficina donde los ciudadanos pueden realizar trámites. Una agenda tiene siempre asociados uno o más recursos.
- **Disponibilidad:** una disponibilidad es la representación de un cupo disponible para que un ciudadano pueda realizar un determinado trámite en una oficina específica. La disponibilidad está identificada por el conjunto (trámite, oficina, día, hora, número).
- **Reserva:** una reserva es la asignación a un ciudadano de un número para realizar un cierto trámite, en una cierta oficina, en un día y hora determinados. La reserva es, por tanto, la utilización de una disponibilidad por un ciudadano.

La aplicación está organizada en dos módulos distinguibles: un módulo público y un módulo privado. El módulo público es el utilizado por los ciudadanos para realizar reservas. El módulo privado es el utilizado por los administradores de la instalación de la aplicación para configurar las agendas, recursos y disponibilidades; el módulo privado también ofrece las funcionalidades de reserva para los casos en que se desee ofrecer el servicio de reserva telefónica (los ciudadanos son atendidos por un telefonista que realiza la reserva por ellos a través del módulo privado). Además también se admite la incorporación de acciones personalizadas (a ejecutar cada vez que un ciudadano confirma o cancela una reserva) y validaciones personalizadas (a ejecutar antes de confirmar una reserva para determinar si los datos

ingresados son válidos); estas incorporaciones se hacen en forma de “plugins” que deben ser implementados en lenguaje Java.

Es importante destacar que esta aplicación es una adaptación de una aplicación previa del mismo nombre desarrollada por la Intendencia de Montevideo. A esta aplicación existente se le han añadido (o añadirán) nuevas funcionalidades, y principalmente se le ha agregado la posibilidad de funcionar en forma multi-empresa (multi-tenant), es decir, puede ser utilizada por múltiples usuarios (organismos, empresas, unidades ejecutoras, según el contexto) sin interferencia entre ellos (cada usuario aprecia la instalación como si fuera propia y exclusiva).

5 Vistas

En esta sección se presentan las diferentes vistas de la arquitectura de la aplicación. Para su diseño se tomaron en cuenta los casos de uso relevantes.

5.1 Vista de Casos de Uso

5.1.1 Listado de Casos de Uso

En esta sección se describen los casos de uso relevantes para la arquitectura del sistema. A continuación se muestra una tabla con todos los casos de usos del sistema marcando en color amarillo los relevantes para la arquitectura. Para cada caso de uso se especifica el nombre, los casos de uso que incluye (Include CU) y el/los actores que lo realizan.

Los actores son los siguientes:

- **Superadministrador (SA):** es un usuario que puede realizar todas las acciones posibles sobre la aplicación, sin restricciones, y si tener en cuenta la empresa a la cual corresponde la agenda, el recurso, la disponibilidad o la reserva.
- **Administrador (A):** es un usuario que puede realizar tareas administrativas sobre las agendas, recursos, disponibilidades y reservas correspondientes a una agenda en particular.
- **Planificador (P):** es un usuario que puede realizar tareas de planificación sobre los recursos correspondientes a una agenda de una empresa en particular, esto es, crear las disponibilidades y tal vez (según configuración) los recursos.
- **Funcionario de call center (FC):** es un usuario que puede realizar reservas en nombre de los ciudadanos ante la solicitud de éstos, por ejemplo en forma telefónica.
- **Funcionario de atención (FA):** es un usuario que puede utilizar la aplicación para proceder a realizar las llamadas de los ciudadanos que tienen reserva para un cierto recurso en forma específica. Este usuario es responsable de realizar la llamada del próximo ciudadano, y marcar si dicho ciudadano se presentó (asistencia) o no (ausencia).

- **Llamador (L):** es un usuario abstracto (no representa una persona en particular sino que es solo un artefacto para limitar el acceso) que solo tiene acceso a la pantalla utilizada para desplegar información sobre los últimos llamados, y los siguientes.
- **Usuario anónimo (UA):** es un usuario que tampoco representa a una persona en particular, sino que representa a un ciudadano que está en proceso de reserva y que aún no ha proporcionado ningún dato que permita identificarlo.

A continuación se muestra una tabla con los casos de uso identificados hasta el momento, marcando en amarillo aquellos que se consideran más relevantes para definir la arquitectura.

Id		SA	A	P	FC	FA	UL	UA
1	Módulo privado: Iniciar sesión	*	*	*	*	*	*	
2	Módulo privado: Seleccionar idioma							
3	Módulo privado: Seleccionar empresa	*	*	*	*	*	*	
4	Módulo privado: Crear una empresa	*						
5	Módulo privado: Consultar empresas	*						
6	Módulo privado: Modificar una empresa	*						
7	Módulo privado: Eliminar una empresa	*						
8	Módulo privado: Crear un usuario en una empresa	*	*					
9	Módulo privado: Consultar usuarios de una empresa	*	*					
10	Módulo privado: Modificar usuario en una empresa	*	*					
11	Módulo privado: Eliminar usuario de una empresa	*	*					
12	Módulo privado: Gestionar tokens de una empresa							
13	Módulo privado: Crear agenda en una empresa	*	*					
14	Módulo privado: Consultar agendas de una empresa	*	*					
15	Módulo privado: Modificar agenda de una empresa	*	*					
16	Módulo privado: Eliminar agenda de una empresa	*	*					
17	Módulo privado: Modificar textos de una agenda de una empresa	*	*					
18	Módulo privado: Crear recurso en una agenda	*	*					
19	Módulo privado: Consultar recursos de una agenda	*	*					
20	Módulo privado: Modificar recurso de una agenda	*	*					
21	Módulo privado: Eliminar recurso de una agenda	*	*					
22	Módulo privado: Modificar textos de un recurso de una agenda	*	*					
23	Módulo privado: Copiar recurso en una agenda	*	*					
24	Módulo privado: Exportar recurso de una agenda	*	*					
25	Módulo privado: Importar recurso en una agenda	*	*					

Id		SA	A	P	FC	FA	UL	UA
26	Módulo privado: Crear agrupación de datos a solicitar en un recurso	*	*					
27	Módulo privado: Consultar agrupaciones de datos a solicitar de un recurso	*	*					
28	Módulo privado: Modificar agrupación de datos a solicitar de un recurso	*	*					
29	Módulo privado: Eliminar agrupación de datos a solicitar de un recurso	*	*					
30	Módulo privado: Crear dato a solicitar en un recurso	*	*					
31	Módulo privado: Consultar datos a solicitar de un recurso	*	*					
32	Módulo privado: Modificar dato a solicitar de un recurso	*	*					
33	Módulo privado: Eliminar dato a solicitar de un recurso	*	*					
34	Módulo privado: Definir una validación personalizada en una empresa	*	*					
35	Módulo privado: Asociar validación a un recurso	*	*					
36	Módulo privado: Definir una acción personalizada en una empresa	*	*					
37	Módulo privado: Asociar una acción personalizada a un recurso	*	*					
38	Módulo privado: Generar disponibilidades para un día en un recurso	*		*				
39	Módulo privado: Copiar disponibilidades de un día en un recurso	*		*				
40	Módulo privado: Consultar disponibilidades de un recurso	*		*				
41	Módulo privado: Modificar disponibilidades de un recurso	*		*				
42	Módulo privado: Eliminar disponibilidades de un recurso	*		*				
43	Módulo privado: Realizar una reserva en una agenda	*			*			
44	Módulo privado: Cancelar una reserva en un recurso	*			*			
45	Módulo privado: Configurar el llamador de un recurso	*	*				*	
46	Módulo privado: Mostrar el llamador de una agenda	*	*				*	
47	Módulo privado: Llamar a un ciudadano en un recurso	*	*			*		

Id		SA	A	P	FC	FA	UL	UA
48	Módulo privado: Registrar asistencia/no asistencia de una reserva	*	*			*		
49	Módulo privado: Consultar reserva por identificador en un recurso	*	*					
50	Módulo privado: Consultar reserva por fecha, hora y número en un recurso	*	*					
51	Módulo privado: Consultar reserva por datos en un recurso	*	*					
52	Módulo privado: Obtener reporte de reservas para un período en un recurso, agenda o empresa	*	*					
53	Módulo privado: Obtener reporte de atenciones en un período en un recurso, agenda o empresa	*	*					
54	Módulo privado: Obtener reporte de atenciones por funcionario en un recurso, agenda o empresa	*	*					
55	Módulo privado: Obtener reporte de tiempo de atención por funcionario en un recurso, agenda o empresa	*	*					
56	Módulo público: Realizar una reserva en una agenda							*
57	Módulo público: Cancelar reserva en una agenda							*
58	Módulo privado: Eliminar disponibilidades en un período	*	*	*				
59	Módulo privado: Atención presencial	*			*			
60	Módulo privado: Cancelar reservas en un período	*	*					
61	Módulo privado: Obtener reporte de atenciones presenciales para un período en un recurso, agenda o empresa	*	*					

5.2 Vista Física

5.2.1 Diagrama de Despliegue

La siguiente imagen ilustra gráficamente los elementos que componen el despliegue de la aplicación.

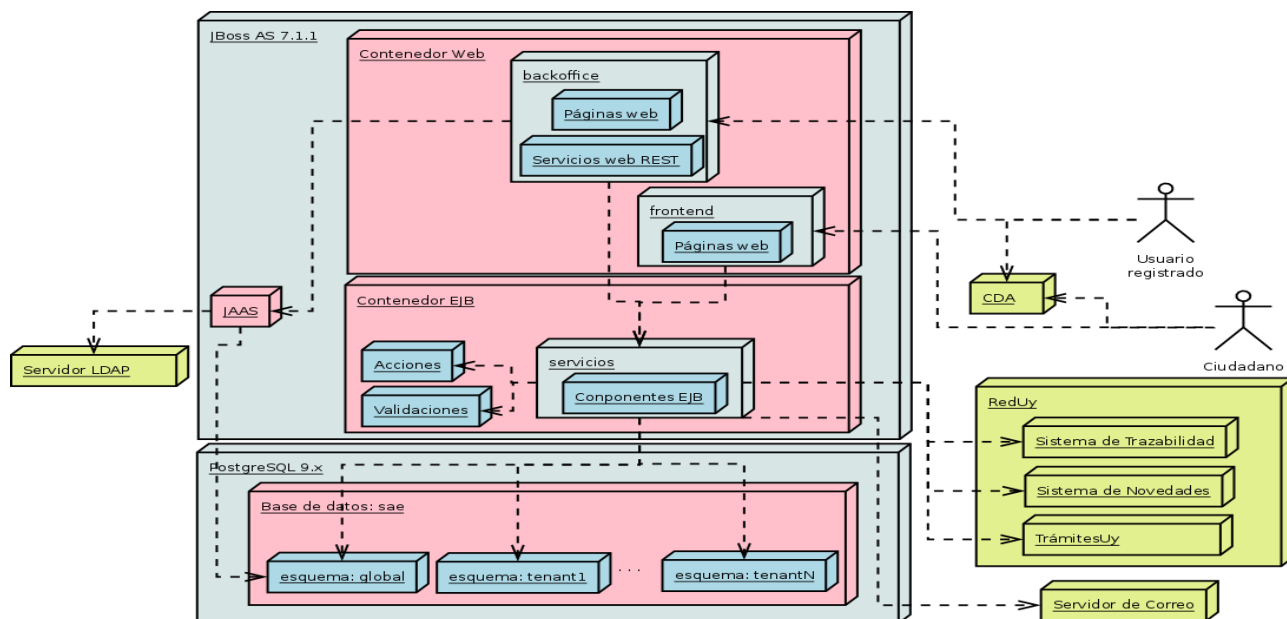


Figura 5.1: Diagrama de despliegue

A continuación se describen los principales elementos que se visualizan en la imagen.

- **Actores.** Básicamente se distinguen dos tipos de actores: los usuarios que deben estar registrados en el sistema y que pueden acceder al módulo privado de la aplicación, para lo cual deben proporcionar su código de usuario y contraseña ("**usuarios registrados**"), y aquellos que acceden de forma anónima (o autenticados según los requerimientos puntuales) a la aplicación y solo tienen acceso al módulo público de la misma ("**ciudadanos**").
- **Servidor de aplicaciones (JBoss AS 7.1.1).** El servidor de aplicaciones es el que da cabida a la aplicación en sí misma. En este componente se instalan tres elementos constituyentes de la aplicación y se configuran los parámetros básicos para el funcionamiento del servidor de aplicaciones:
 - **Contenedor web:** en este componente propio del servidor de aplicaciones se instala la parte encargada de la presentación de la aplicación, es decir, la que hace de interfaz con los usuarios. La presentación está compuesta por dos interfaces: la interfaz pública ("**frontend**"), a la que pueden acceder todos los ciudadanos conociendo una URL apropiada, y la interfaz privada ("**backoffice**") a la que solo pueden acceder los usuarios registrados proporcionando un código de usuario y una contraseña previamente registrados. El componente frontend está compuesto por páginas web dinámicas (para lo cual se utilizan las tecnologías JSF y facelets) mientras que el componente backoffice también está compuesto por páginas web dinámicas (con las mismas tecnologías) y además expone servicios web REST.
 - **Contenedor EJB:** en este componente propio del servidor de aplicaciones se instala la otra parte de la aplicación, encargada de ejecutar la lógica de negocio e interactuar con sistemas externos tales como el servidor de bases de datos, el

servidor de correo electrónico y los proveedores externos de información (EISs).

- **JAAS.** Este es un componente propio del servidor de aplicaciones que se encarga de controlar y limitar el acceso a los recursos. Este componente tiene como finalidad autenticar a los usuarios (validar las credenciales de los mismos) y verificar que cada usuario tiene acceso a los recursos que solicita.
- **Servidor de bases de datos (Postgres 9.x).** En el servidor de bases de datos es donde se almacena toda la información gestionada por la aplicación. Para esto, se define una única base de datos ("sae"), la cual está a su vez compuesta por un conjunto de esquemas, uno de los cuales es común a todos los tenants ("**global**"), y el resto específico de cada tenant ("**tenant1**", ..., "**tenantN**"). En el esquema global se almacenan los datos necesarios para la operativa de la aplicación (definiciones de tenants, datos de usuarios, etc), mientras que en el resto de los esquemas se almacena información relativa a la funcionalidad propia de la aplicación, exclusiva de cada tenant (agendas, recursos, disponibilidades y reservas).
- **CDA.** Este componente representa al Sistema de Control de Acceso de AGESIC. Puede ser utilizado para validar a los usuarios tanto del módulo privado de administración como a los ciudadanos en los casos en que sea requeridos que éstos se autenticuen para poder realizar reservas. Es un componente opcional.
- **Servidor LDAP.** Este componente es opcional ya que es solo utilizado cuando se requiere que la aplicación valide a los usuarios del módulo privado contra un servidor LDAP.
- **Servidor de correo.** Este componente permite el envío de notificaciones mediante correo electrónico.
- **RedUy.** Este componente engloba a varios servicios externos que pueden ser accedidos (opcionalmente, según la configuración aplicada) por la aplicación, en particular el Sistema de Trazabilidad de AGESIC, el Sistema de Novedades de AGESIC, y TrámitesUy.
- **Acciones y Validaciones.** Estos componentes representan a las acciones y validaciones personalizadas que pueden ser incorporadas a la aplicación. La solución básica no incluye ninguno de estos pero pueden ser incorporados en forma relativamente fácil por los organismos que emplean la aplicación.

La siguiente diagrama muestra el despliegue de componentes a escala física, tanto para el caso en que se utilice un balanceador de carga (Apache HTTPd Server) o no:

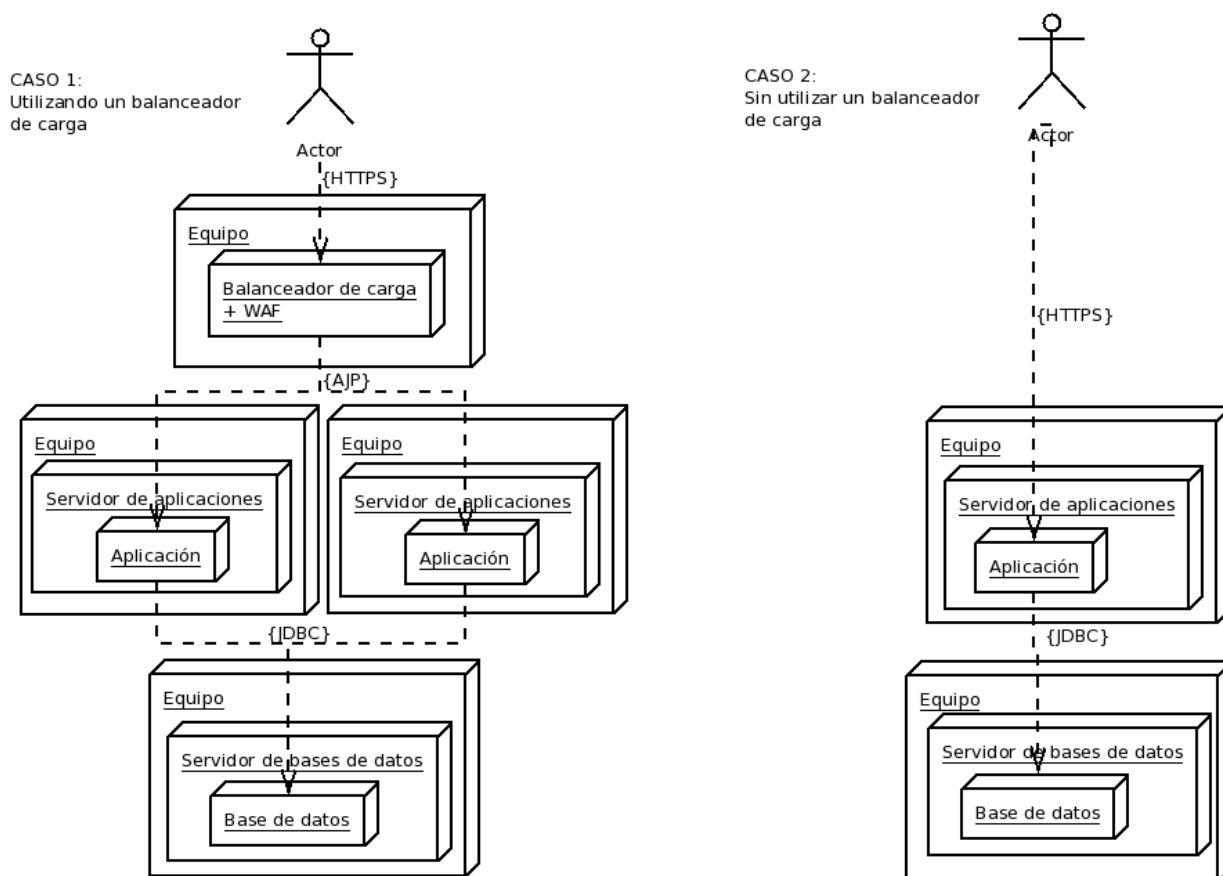


Figura 5.2: Diagrama de despliegue físico (dos casos)

En esta figura se incorpora un elemento adicional, etiquetado “Balanceador de carga + WAF”. Notar que en el caso de utilizar este elemento adicional la comunicación entre dicho elemento y los servidores de aplicaciones es mediante el protocolo AJP. Si bien puede utilizarse el protocolo HTTP no es lo más recomendable en cuanto a rendimiento. También notar que en ambos casos la interacción entre los actores y la aplicación es bajo HTTPS.

5.3 Vista Lógica

En esta vista se representa la funcionalidad que el sistema proporciona a los usuarios. Es decir, se representa lo que el sistema hace y las funciones y servicios que ofrece.

5.3.1 Diagrama de Secuencia 1: Autenticación

Para acceder a un recurso protegido de la aplicación (solo en el backoffice) el usuario debe proporcionar sus credenciales (normalmente código de usuario y contraseña). La verificación de dichas credenciales está a cargo de JAAS. También corresponde a JAAS verificar que una vez que las credenciales del usuario han sido validadas (son correctas) determinar el perfil que posee el usuario entre los definidos en la sección 5.1.1 (puede tener más de uno a la vez). Sin importar el recurso solicitado por el usuario que motivó el inicio de sesión, una vez cumplido el

proceso de solicitud, ingreso y validación de credenciales el usuario es redirigido a una página principal, común. El usuario también es registrado automáticamente en el primero de los tenants en los cuales está habilitado. Para la autenticación hay tres escenarios: autenticación contra la base de datos local de la aplicación, autenticación contra un servidor LDAP, o autenticación contra CDA.

Nota importante: los servicios web REST expuestos por el componente backoffice normalmente estarán excluidos de cualquier autenticación, y especialmente de CDA. Esto es porque el sistema CDA está diseñado solo para permitir la autenticación a clientes humanos, pero los clientes esperados de los servicios web REST serán siempre otras aplicaciones que no podrán especificar en forma interactiva las credenciales requeridas ni sabrán realizar todas las redirecciones solicitadas por CDA. En el caso de requerir que los servicios web REST también estén protegidos mediante CDA será responsabilidad del desarrollador de la aplicación cliente (la que consume los servicios web REST) de saber interpretar los datos enviados y requeridos por CDA y de manejar apropiadamente las redirecciones. Este procedimiento no está explicado en este documento ni en ningún otro perteneciente a SAE ya que está fuera de alcance; por más información se debe consultar al equipo de Seguridad Informática de AGESIC a través de la Mesa de Ayuda de dicho organismo.

Autenticación ante la base de datos local de la aplicación

El primer escenario corresponde al proceso general de control de acceso realizado por la aplicación contra el servidor de bases de datos. El siguiente diagrama describe este escenario:

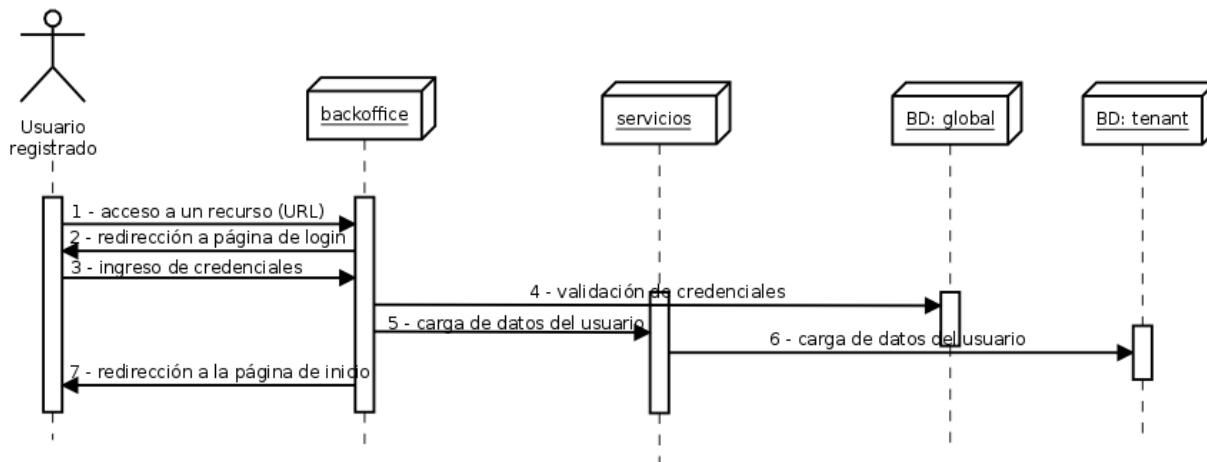


Figura 5.3: Diagrama de secuencia: autenticación (base de datos)

Autenticación ante un servidor LDAP externo

El segundo escenario corresponde al proceso general de control de acceso realizado por la aplicación contra un servidor LDAP externo. El siguiente diagrama describe este escenario:

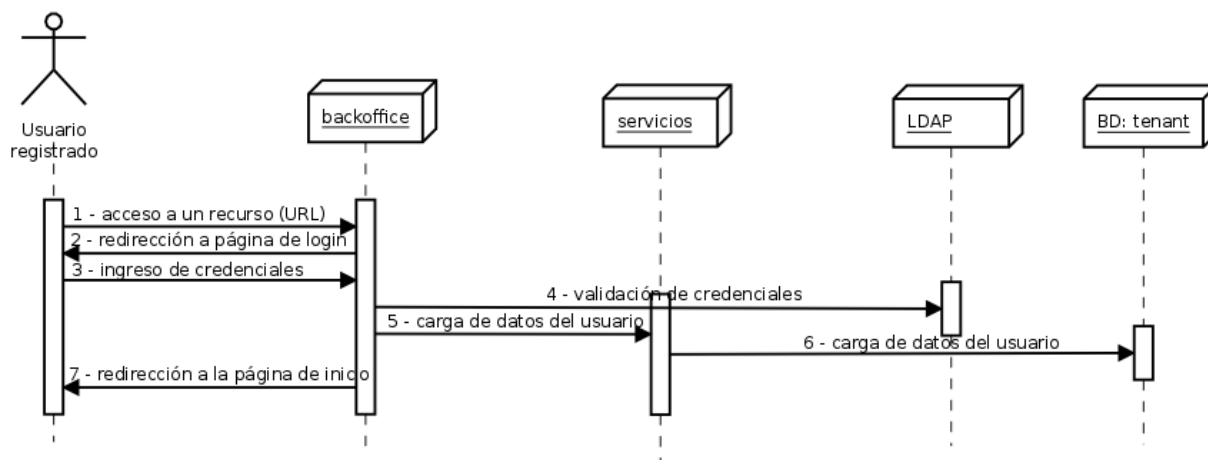


Figura 5.4: Diagrama de secuencia: autenticación (LDAP)

Autenticación ante el servicio de Control de Acceso

El tercer escenario corresponde al proceso general de control de acceso realizado por la aplicación cuando está integrada al Sistema de Control de Acceso. Este, a su vez, tiene dos sub-escenarios: por un lado cuando el usuario que se autentica no está ya autenticado en CDA y el segundo cuando el usuario no está autenticado en la aplicación pero sí en CDA. El siguiente diagrama describe el primer sub-escenario:

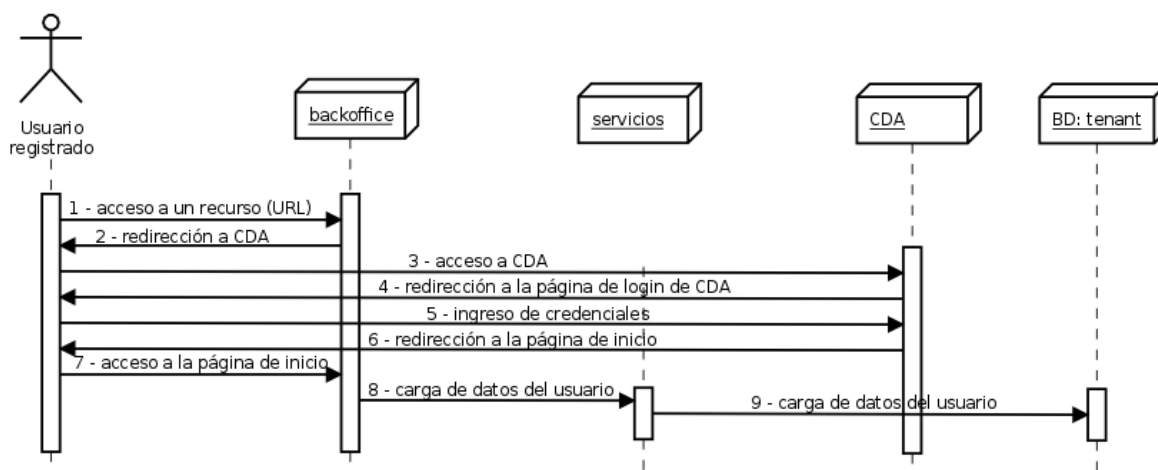


Figura 5.5: Diagrama de secuencia: autenticación (CDA - autenticación inicial)

El siguiente diagrama describe el segundo sub-escenario:

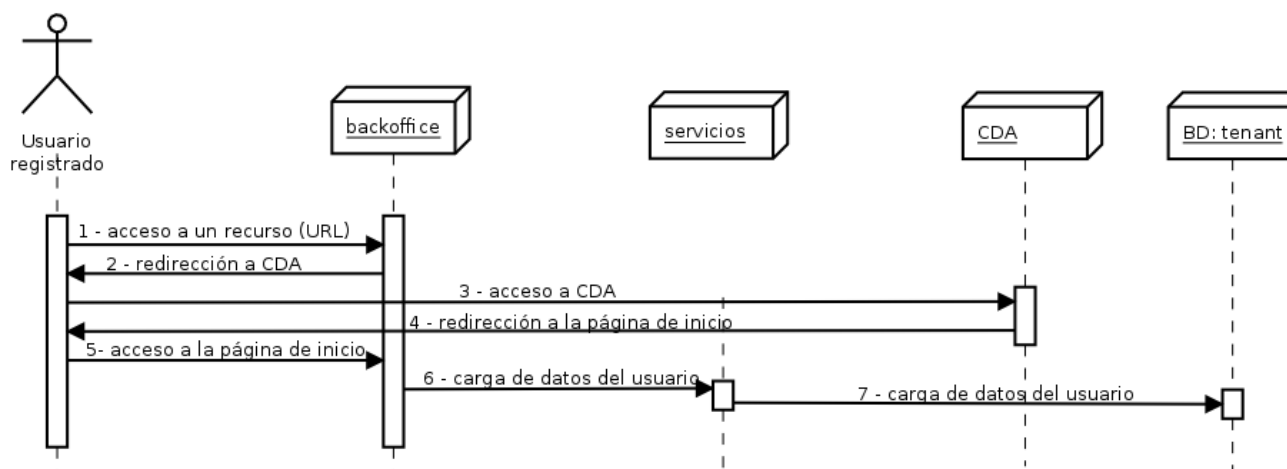


Figura 5.6: Diagrama de secuencia: autenticación (CDA - reautenticación)

5.3.2 Diagrama de Secuencia 2: Cambio de empresa

Este diagrama muestra las interacciones que ocurren cuando el usuario decide cambiar de empresa:

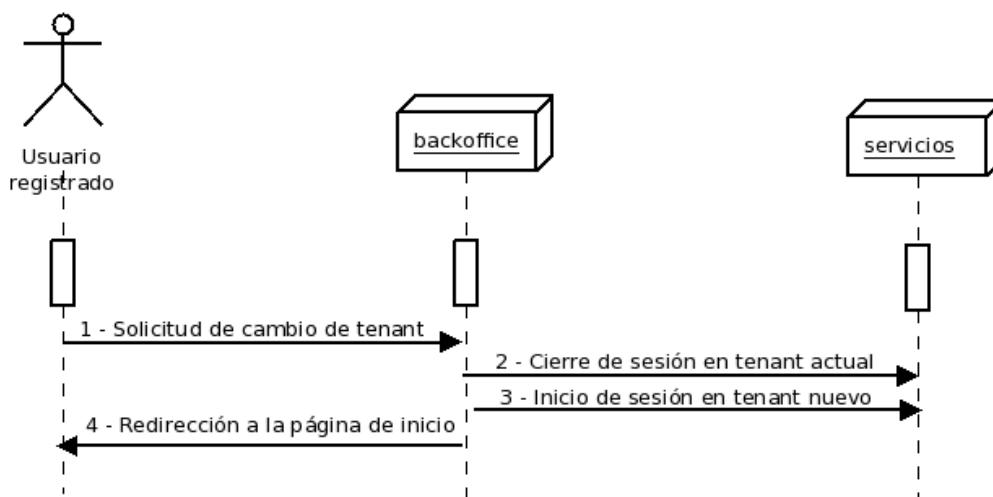


Figura 5.7: Diagrama de secuencia: cambio de empresa

Cuando el usuario solicita el cambio de empresa, la aplicación primero cierra la sesión de dicho usuario asociada a la empresa actual, y le inicia una nueva sesión asociada a la nueva empresa. No es necesario volver a solicitar las credenciales ya que la aplicación ya las obtuvo en el primer inicio de sesión. Nuevamente, el usuario es redirigido a la página de inicio sin importar dónde se encontraba antes de solicitar el cambio.

5.3.3 Diagrama de Secuencia 3: Crear empresa

Este diagrama de secuencia muestra la interacción entre el usuario, los componentes propios de la aplicación y los sistemas externos para registrar una nueva empresa en el sistema.

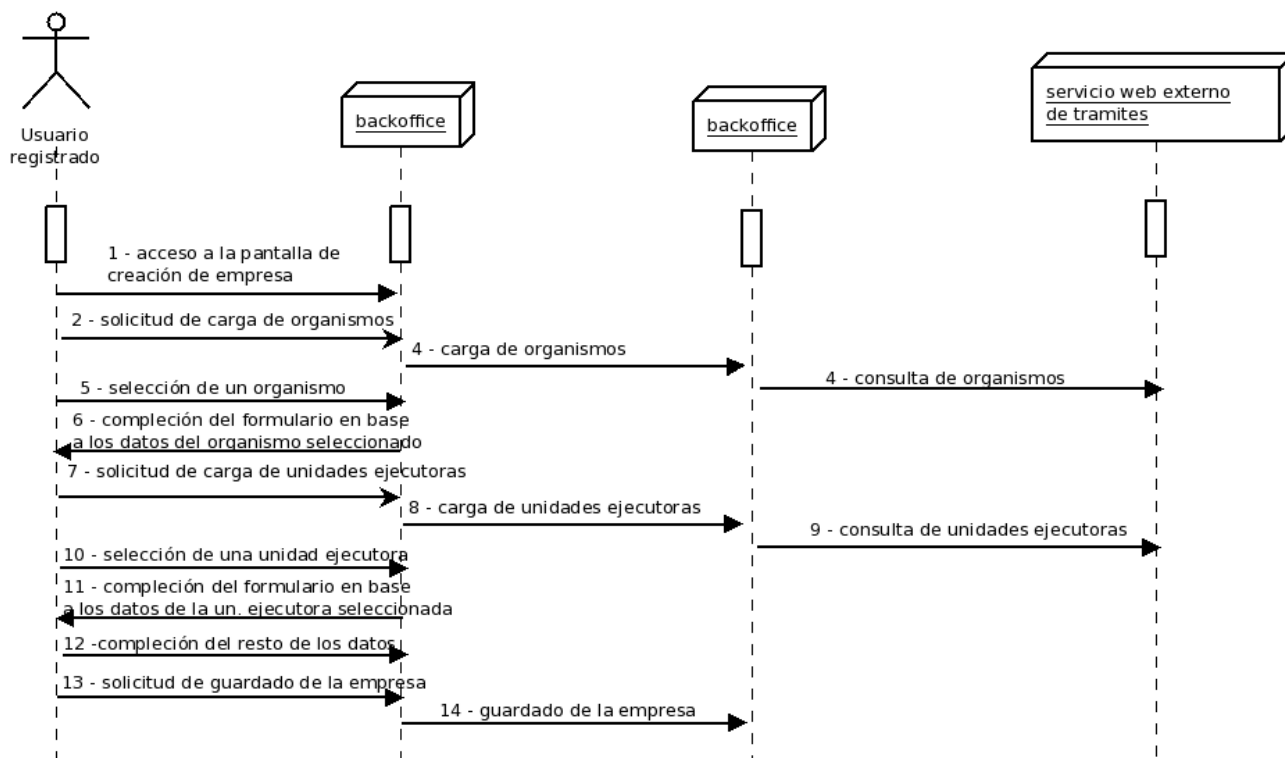


Figura 5.8: Diagrama de secuencia: crear empresa

Cuando el usuario debe crear una empresa la aplicación le proporciona la opción de seleccionar un organismo y una ejecutora a partir de sendas listas. Inicialmente dichas listas estarán vacías pudiendo el usuario solicitar el cargado de las mismas. La aplicación cargará las listas con información obtenida a partir de un servicio web provisto por una fuente externa. Cuando el usuario selecciona un organismo o una unidad ejecutora, los datos correspondientes a ellos serán automáticamente cargados en el formulario. El usuario es libre también de ingresar los datos manualmente sin seleccionar ningún organismo ni unidad ejecutora, aunque debe ser consciente de que la empresa no quedará asociada a ningún organismo y/o unidad ejecutora, y por tanto no podrá cargar los trámites en forma automática cuando desee crear agendas para la empresa.

5.3.4 Diagrama de Secuencia 4: Modificar empresa

Este diagrama de secuencia muestra la interacción entre el usuario, los componentes propios de la aplicación y los sistemas externos para modificar una empresa que ya está ingresada en el sistema.

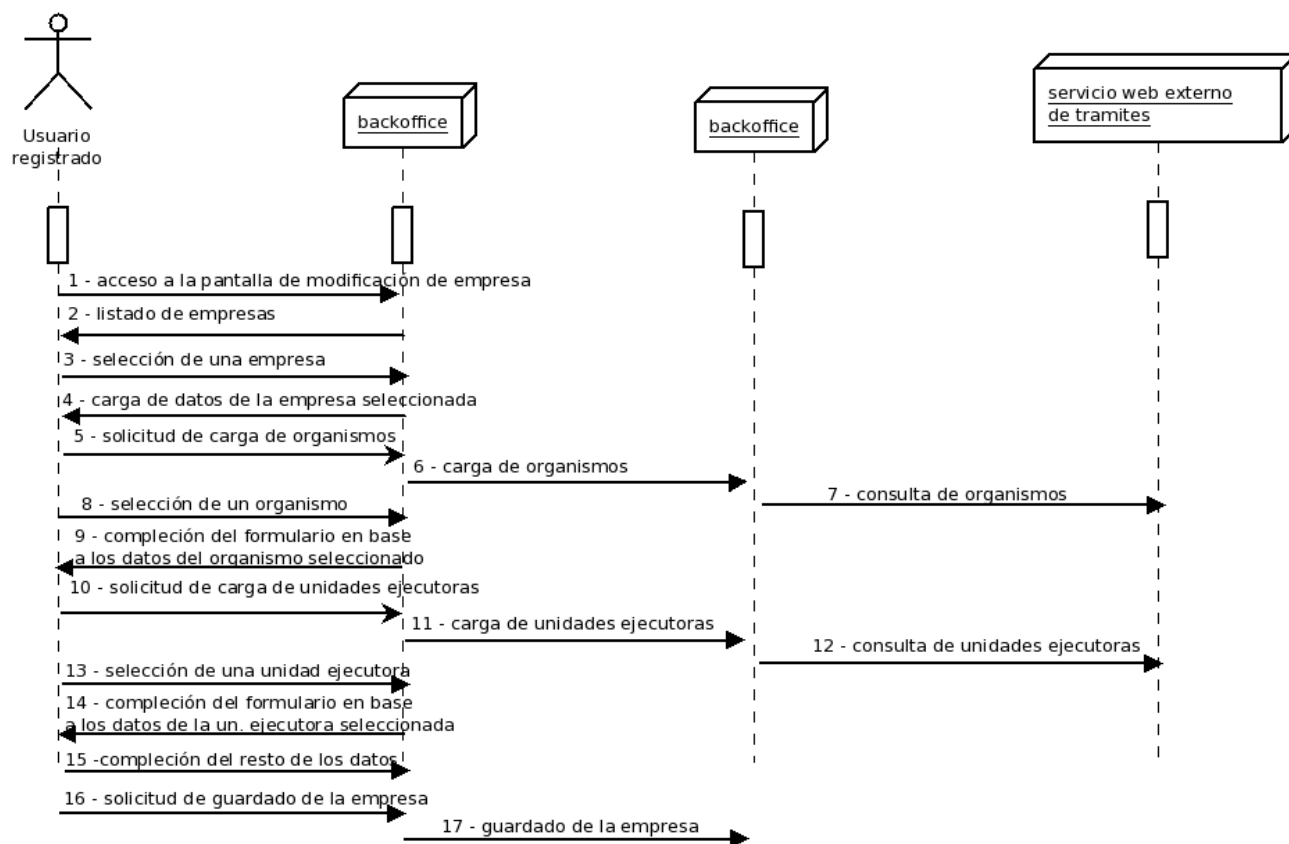


Figura 5.9: Diagrama de secuencia: modificar empresa

Este caso de uso es muy similar a la creación de la empresa, salvo que al comenzar el usuario debe seleccionar una de las empresas ya registradas en el sistema antes de proceder a editar los datos de la misma. El usuario, además, es libre de modificar los datos que considere necesarios dejando los restantes en sus valores originales.

5.3.5 Diagrama de Secuencia 5: Crear usuario

Este diagrama de secuencia muestra la interacción entre el usuario y los componentes propios de la aplicación para registrar un nuevo usuario en el sistema.

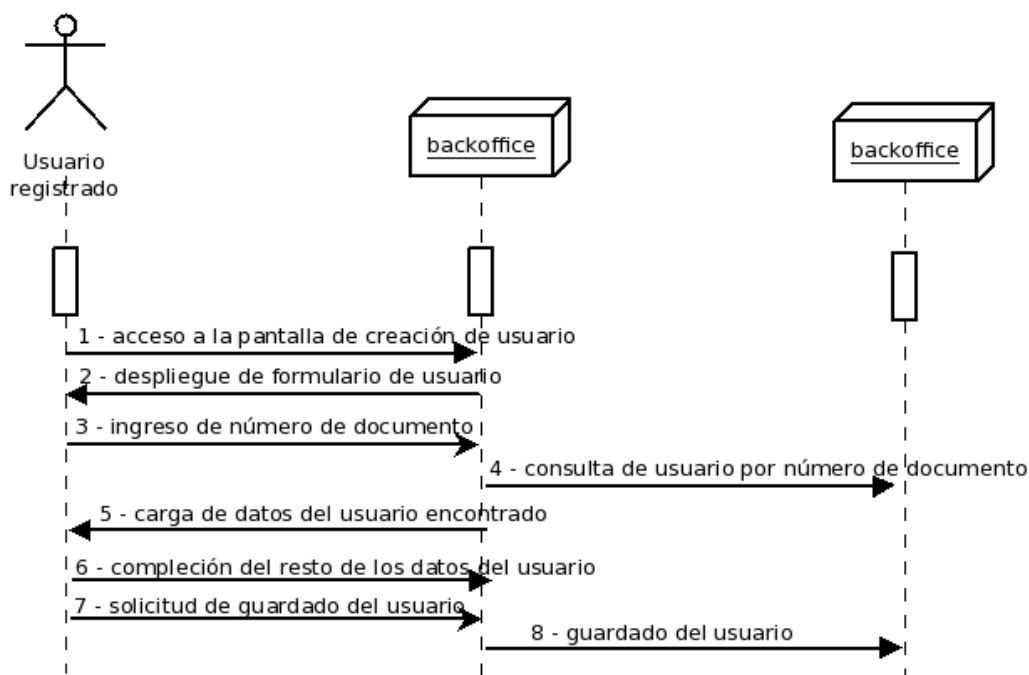


Figura 5.10: Diagrama de secuencia: crear usuario

Este caso de uso contempla dos situaciones: cuando se accede a la funcionalidad para crear un usuario, el primer campo a completar es el número de documento del mismo. Al hacer esto es posible que ya exista un usuario registrado con dicho número de documento, en cuyo caso los datos conocidos son cargados en forma automática en el sistema. Si además el usuario encontrado estaba ya registrado en la empresa actual también se cargan los roles que posee en esta empresa.

5.3.6 Diagrama de Secuencia 6: Crear agenda

Este diagrama de secuencia muestra la interacción entre el usuario, los componentes propios de la aplicación y los sistemas externos para registrar una nueva agenda asociada a la empresa actual.

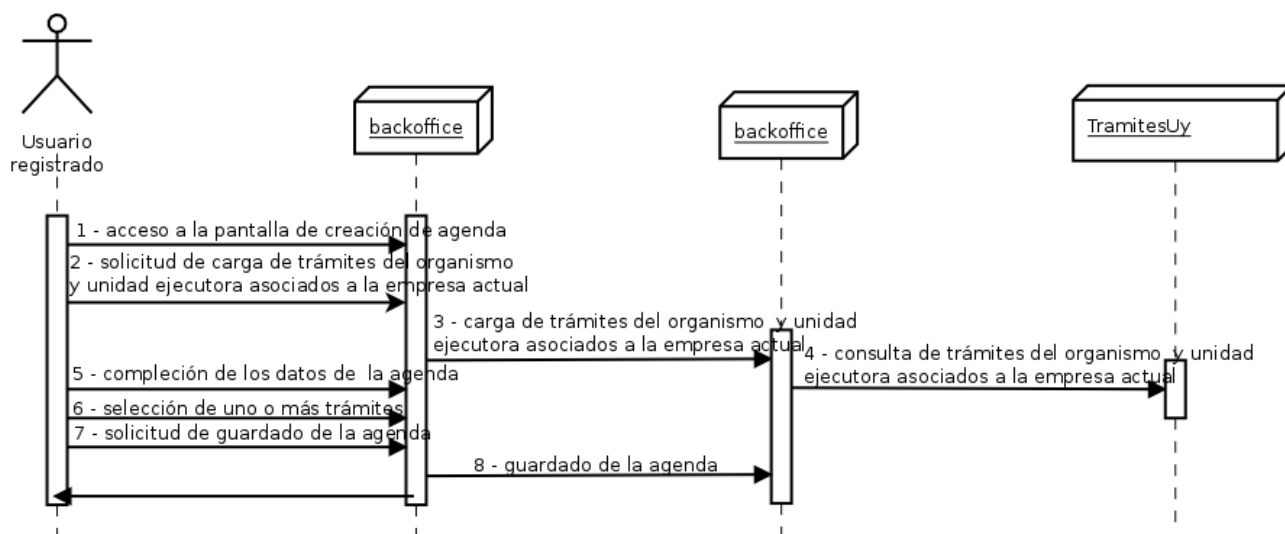


Figura 5.11: Diagrama de secuencia: crear agenda

En el sistema SAE, una agenda corresponde a un trámite que puede ser realizado en un organismo/unidad ejecutora. Por tanto, el sistema permite importar los datos de la agenda seleccionando uno de los trámites disponibles, los cuales son obtenidos en base al organismo y unidad ejecutora asociados a la empresa actual. Entonces, para crear una agenda basta con que el usuario seleccione un trámite del listado de trámites que se muestra en el formulario. En aquellos casos en los cuales la empresa actual no está vinculada a ningún organismo o unidad ejecutora (lo cual es posible, ver el diagrama de secuencia 3) esta lista estará vacía y el usuario debe completar manualmente todos los campos. Siempre es posible registrar agendas que no estén asociadas a ningún trámite en particular, incluso si la empresa está asociada a un organismo y unidad ejecutora válidos.

5.3.7 Diagrama de Secuencia 7: Crear recurso

Este diagrama de secuencia muestra la interacción entre el usuario, los componentes propios de la aplicación y los sistemas externos para registrar un nuevo recurso perteneciente a una agenda particular.

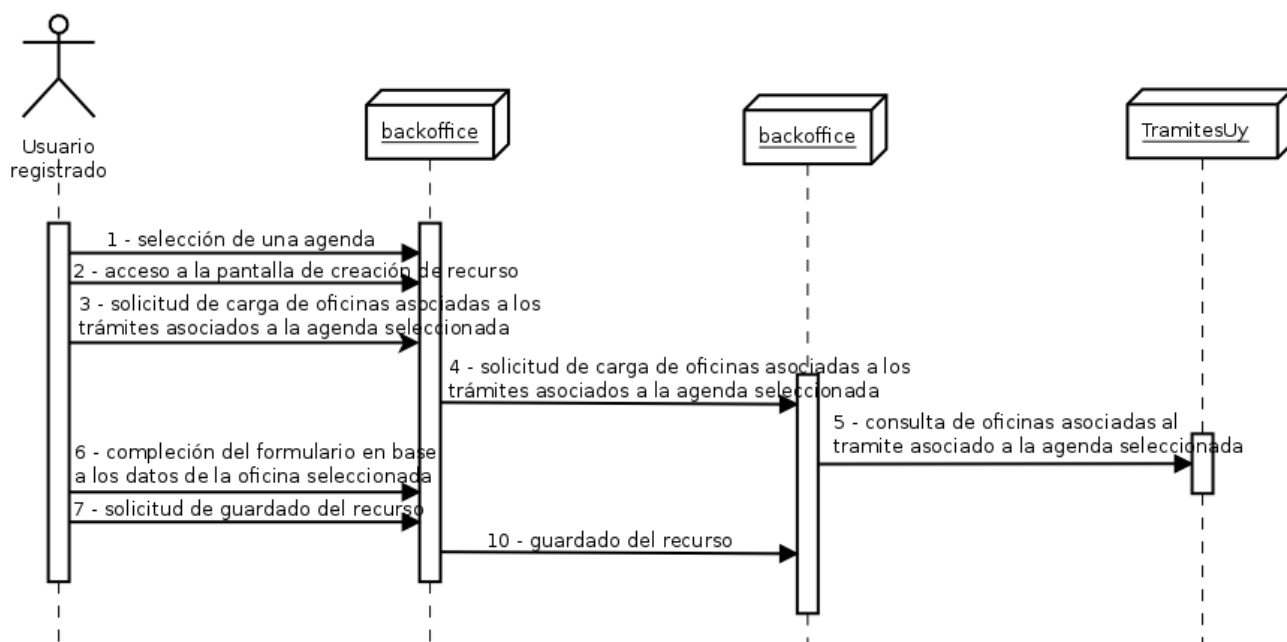


Figura 5.12: Diagrama de secuencia: crear recurso

Para este caso de uso es necesario que el usuario seleccione una agenda antes de proceder a la creación del recurso. Esto es así debido a que los recursos se corresponden con oficinas donde se pueden realizar trámites, y por tanto es necesario seleccionar una agenda (trámite) antes de poder registrar un recurso (oficina). Es importante destacar que si la agenda no está asociada a ningún trámite en particular (bien porque la empresa no está asociada a un organismo o unidad ejecutora, o mismo el usuario decidió no vincular la agenda a un trámite) entonces el usuario no podrá seleccionar la oficina de la lista y deberá ingresar todos los datos en forma manual.

5.3.8 Diagrama de Secuencia 7: Generar disponibilidad para un día

Este diagrama de secuencia muestra la interacción entre el usuario y los componentes propios de la aplicación para generar disponibilidades para un día, para un recurso específico de una agenda particular ("turnos disponibles").

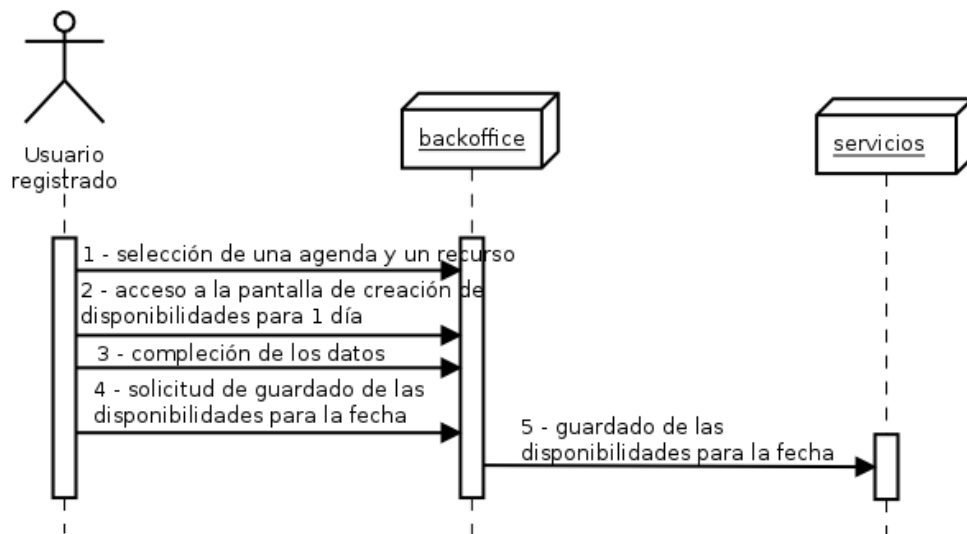


Figura 5.13: Diagrama de secuencia: crear disponibilidades

Este caso de uso permite generar disponibilidades para un día particular, estando trabajando en una agenda y recurso específicos.

5.3.9 Diagrama de Secuencia 8: Copiar disponibilidades

Este diagrama de secuencia muestra la interacción entre el usuario y los componentes propios de la aplicación para generar disponibilidades para un período dado, para un recurso específico de una agenda particular ("turnos disponibles"), copiando las disponibilidades generadas para un día específico, llamado "modelo". La aplicación replicará las disponibilidades del día modelo para todos y cada uno de los días comprendidos en el período especificado.

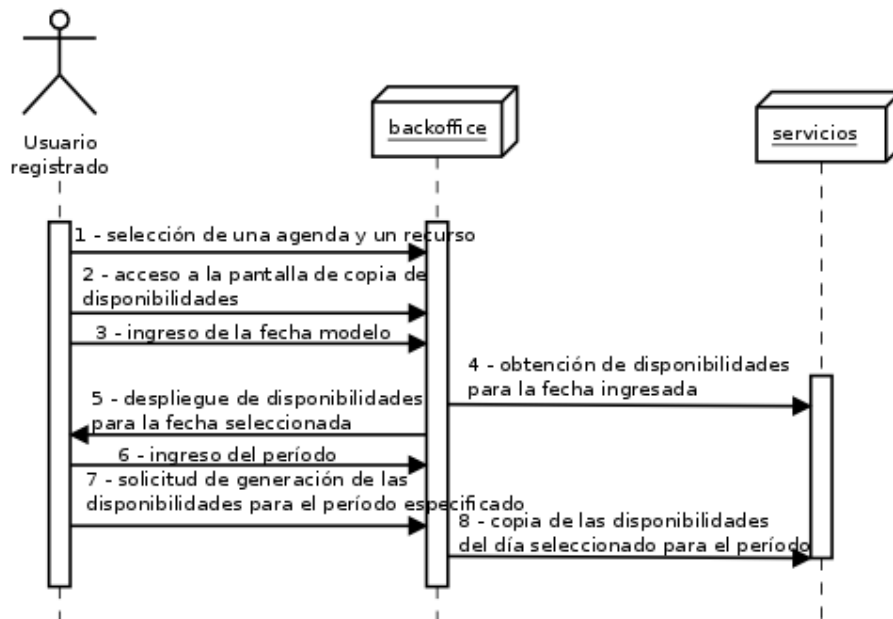


Figura 5.14: Diagrama de secuencia: copiar disponibilidades

5.3.10 Diagrama de Secuencia 9: Mostrar llamador

Este caso de uso permite mostrar una pantalla en la que se despliegan los números que son llamados para ser atendidos.

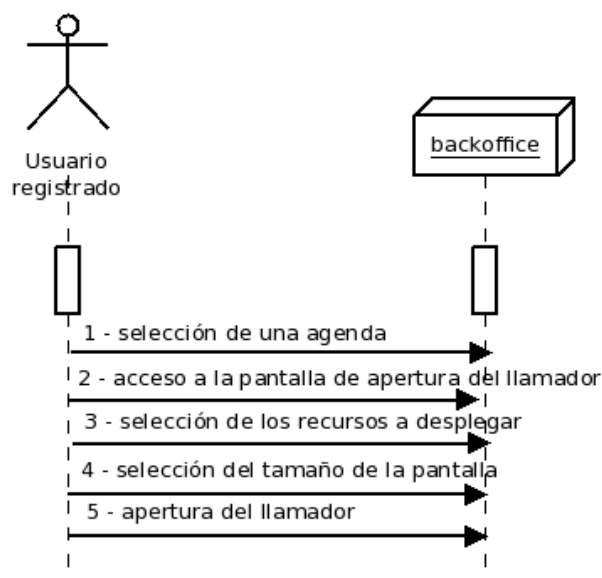


Figura 5.15: Diagrama de secuencia: abrir llamador

5.3.11 Diagrama de Secuencia 10: Realizar reserva pública

Este caso de uso comprende la funcionalidad completa de realización de reserva por parte de un ciudadano.

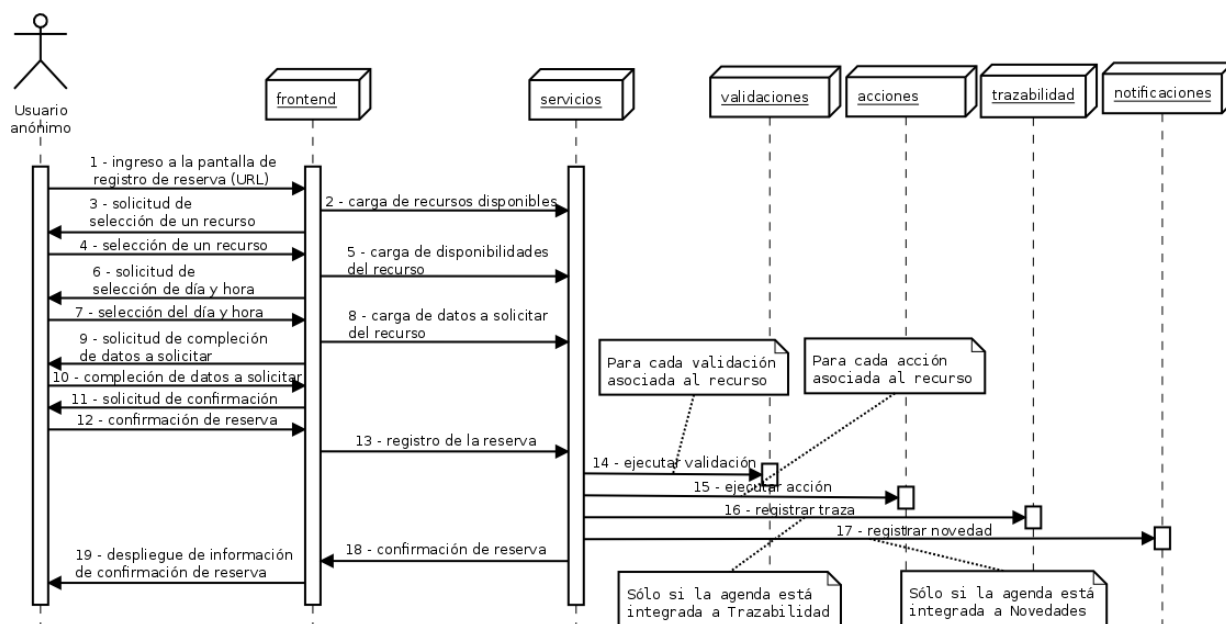


Figura 5.16: Diagrama de secuencia: reserva pública

Este caso de uso no requiere autenticación por parte de los usuarios. Sin embargo, para poder proceder a la reserva de un número es necesario conocer una URL válida que contenga al identificador de la empresa, al identificador de la agenda y opcionalmente al identificador del recurso para el cual se va a realizar la reserva. Luego, y unque el recurso ya haya sido especificado en la URL, la aplicación brinda dar la oportunidad de cambiar de recurso (siempre dentro de la misma agenda). Al momento de la elección del día y la hora la aplicación no permite seleccionar turnos que no tengan cupos disponibles. Finalmente, la confirmación de la reserva contiene un texto legal y debe solicitar al usuario que acepte las condiciones de uso del sistema.

5.4 Vista de base de datos

En esta sección se muestran los diagramas de bases de datos, incluyendo las tablas con todos sus atributos y las relaciones entre ellas.

5.4.1 Esquema global

El esquema global es el que comparten todos los tenants. El siguiente diagrama muestra el modelo de datos de este esquema:

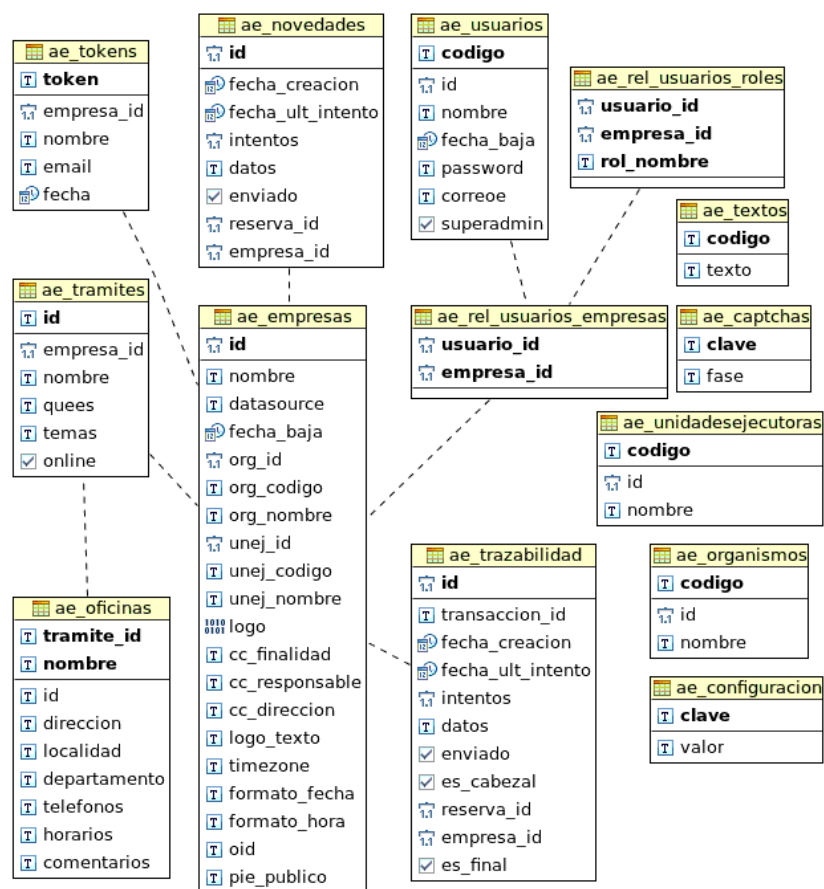


Figura 5.17: Modelo de datos (esquema global)

5.4.2 Esquema específico de cada tenant

A cada tenant le corresponde un esquema propio. Los esquemas de cada tenant son idénticos, a excepción de los datos que contienen, y no tienen ninguna relación directa entre sí ni con el esquema global, aunque sí tienen referencias (no relaciones) a tablas del esquema global.



5.5 Vista de interoperabilidad

En esta sección se describe la interacción de la aplicación SAE con otros sistemas externos mediante servicios web; en particular, se describe la interoperabilidad con TrámitesUy y Trazabilidad. La interacción con el sistema de control de acceso (CDA) se trata en la sección siguiente.

5.5.1 Interoperabilidad con TrámitesUy

Los servicios web de TrámitesUy permiten obtener información sobre los organismos y las unidades ejecutoras del Estado, y sobre los trámites que pueden realizarse en cada uno de ellos, así como las oficinas en las cuales pueden hacerse.

La aplicación SAE utiliza dos servicios web de TrámitesUy:

- **WSTramite:** permite obtener información sobre los organismos y unidades ejecutoras del Estado. Particularmente se utilizan dos operaciones:
 - **obtenerOrganismos**
 - **obtenerUnidadesEjecutoras**
- **GuiaTramites:** permite obtener información sobre los trámites que pueden ser realizados en cada organismo/unidad ejecutora. Particularmente se utilizan las siguientes operaciones:
 - **obtTramitesPorOrgEnOrdenAlfabetico**
 - **obtTramitePorId**

Al momento de registrar una empresa debe indicarse el organismo y la unidad ejecutora con los cuales está vinculada. Esto puede hacerse seleccionando ambas cosas en sendas listas de selección (que son cargadas utilizando las operaciones mencionadas del servicio web WSTramite) o ingresando manualmente el código y nombre del organismo y/o unidad ejecutora. En cualquier caso junto a dichas listas de selección se ofrecen enlaces para forzar el recargado de ellas, invocando nuevamente a la operación correspondiente del servicio web. Si la invocación falla, se mantiene la lista anterior. Inicialmente (cuando se está registrando la primera empresa) estas listas estarán vacías por lo que la primera vez será necesario utilizar estos enlaces.

Una vez creada la empresa, cada vez que se cree una agenda dentro de ella se tiene la posibilidad de asociarla a un trámite; esto se hace seleccionando el trámite de una lista de selección, cuyas opciones son obtenidas consultando al servicio web GuiaTramites, utilizando como parámetros los códigos del organismo y la unidad ejecutora asociados a la empresa a la cual pertenece la agenda. Al igual que en el caso de organismos y unidades ejecutoras, junto a la lista de selección se muestra un enlace que permite recargar la lista de trámites consultando nuevamente al servicio web con los mismos parámetros. De igual manera, si la invocación falla se mantiene el listado anterior.

Al momento de crear recursos para una agenda, la aplicación ofrece la posibilidad de seleccionar una oficina vinculada al trámite elegido (si se eligió asociar la agenda a un trámite). Esta lista también es cargada a partir del servicio web GuiaTramites, pero no existe una operación específica para ello; en su lugar, se invoca la operación obtTramitePorId, la cual incluye, en la respuesta, la lista de oficinas en las cuales se puede realizar el trámite (solo se consideran las oficinas marcadas como nacionales).

5.5.2 Interoperabilidad con Trazabilidad

El sistema de Trazabilidad de AGESIC permite realizar un seguimiento de los procesos que realiza un usuario en múltiples organismos. Este sistema está compuesto por dos operaciones:

- **Cabecal:** permite registrar el comienzo de un proceso. Cada proceso consta exactamente de un cabecal, el cual se registra antes de cualquier línea. Al registrar un cabecal se obtiene un código de trazabilidad único, que debe ser comunicado al ciudadano para que pueda realizar el seguimiento posterior.
- **Línea:** permite registrar cada uno de los pasos ejecutados en el transcurso del proceso. Cada línea siempre está relacionada a exactamente un cabecal, y debe registrarse luego del mismo; a su vez, las líneas deberían estar numeradas en forma consecutiva a medida que se van registrando.

La invocación de estos servicios se realiza en tres ocasiones:

- Cuando un ciudadano confirma una reserva. En este caso se genera el cabecal y una línea. Se indica que el trámite está comenzado. El cabecal se intenta generar en el mismo momento que se confirma la reserva para poder obtener el código de trazabilidad y entregárselo al ciudadano; de no ser posible se almacena para ser enviada posteriormente en forma asincrónica. La línea siempre se almacena para ser enviada posteriormente en forma asincrónica.
- Cuando un ciudadano cancela una reserva. En este caso se genera una línea asociado al cabecal correspondiente a la reserva, indicando que el trámite está finalizado. Esta línea siempre se almacena para ser enviada posteriormente en forma asincrónica.
- Cuando un funcionario de atención al público marca que el ciudadano asistió cuando fue llamado, o cuando marca que no se presentó; en este caso también se marca el trámite como finalizado. Esta línea siempre se almacena para ser enviada posteriormente en forma asincrónica.

Se debe tener en cuenta que es posible que al intentar registrar un cabecal a los efectos de obtener un código de trazabilidad el servicio puede no estar disponible. En este caso la invocación al servicio de trazabilidad es reintentado en forma periódica hasta lograr tener éxito, en cuyo caso se envía mediante correo electrónico (si es posible) al ciudadano el código de trazabilidad.

A continuación se muestra gráficamente la secuencia de invocaciones involucrada:

Confirmación de reserva - con conectividad

En este caso se invoca al servicio de trazabilidad, se obtiene el código y se le devuelve dicho código al ciudadano. La línea se almacena en la base de datos para ser enviada posteriormente en forma asíncrona. El estado registrado del proceso es "2 - en ejecución".

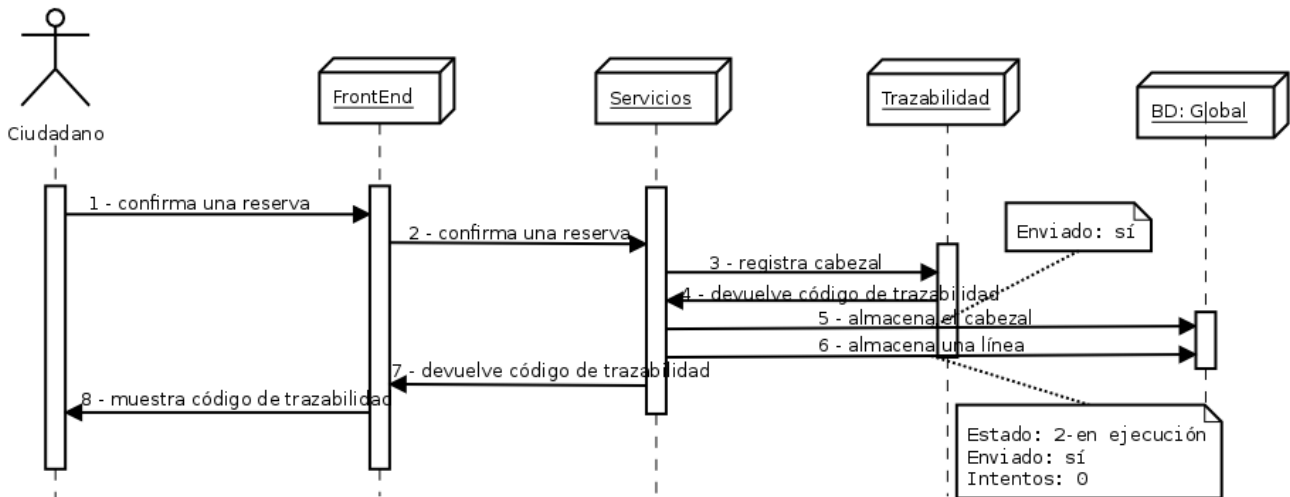


Figura 5.19: Diagrama de secuencia - confirmación de reserva con conexión al Sistema de Trazabilidad

Confirmación de reserva - sin conectividad

En este caso, la aplicación intenta invocar al servicio de trazabilidad para generar el cabezal, pero al no haber conectividad no puede obtener el código de trazabilidad. La aplicación almacena los datos relativos a la invocación y le devuelve al ciudadano todos los datos relativos a la reserva, sin el código de trazabilidad. Tanto el cabezal como la línea son almacenados para ser enviados más tarde en forma asíncrona. El estado registrado del proceso es "2 - en ejecución".

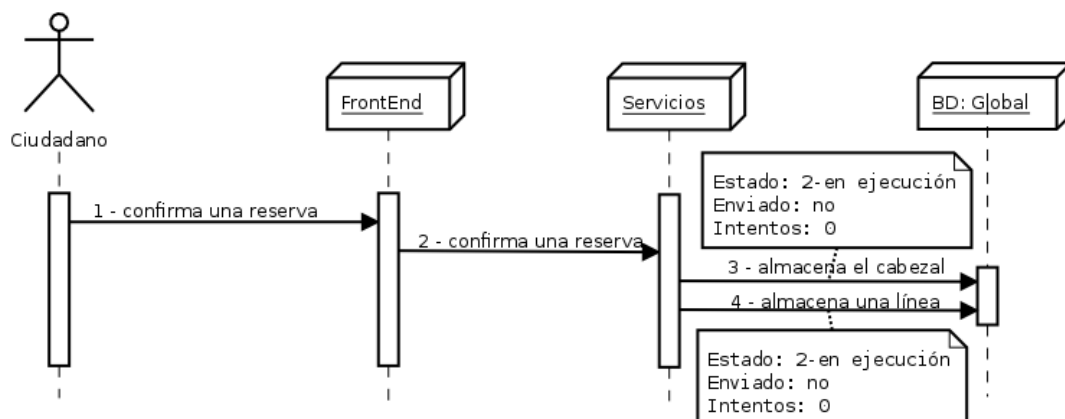


Figura 5.20: Diagrama de secuencia: confirmación de reserva (offline)

Cancelación de reserva

En este caso no se genera un cabecal ya que el mismo fue generado cuando el ciudadano hizo la reserva. Solo se añade una línea al mismo cabecal generado entonces, el cual es reconocido por el identificador de transacción. Solo se registra la línea en la base de datos para ser enviada posteriormente en forma asíncrona. El estado registrado del proceso es “4 - finalizado”.

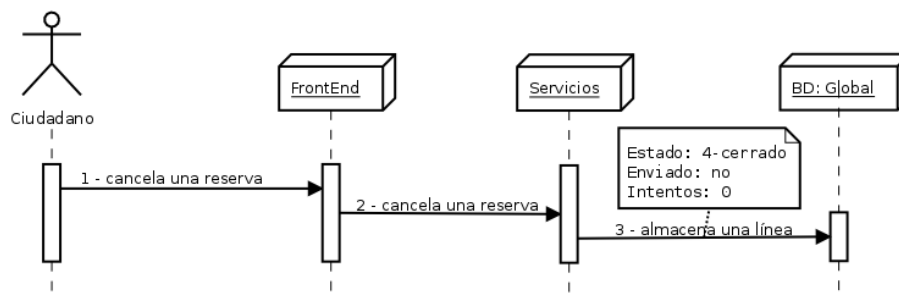


Figura 5.21: Diagrama de secuencia: cancelación de reserva

Marca de asistencia / no asistencia

En este caso tampoco se genera un cabecal ya que el mismo fue generado cuando el ciudadano hizo la reserva. Solo se añade una línea al mismo cabecal generado entonces, el cual es reconocido por el identificador de transacción. Solo se registra la línea en la base de datos para ser enviada posteriormente en forma asíncrona.

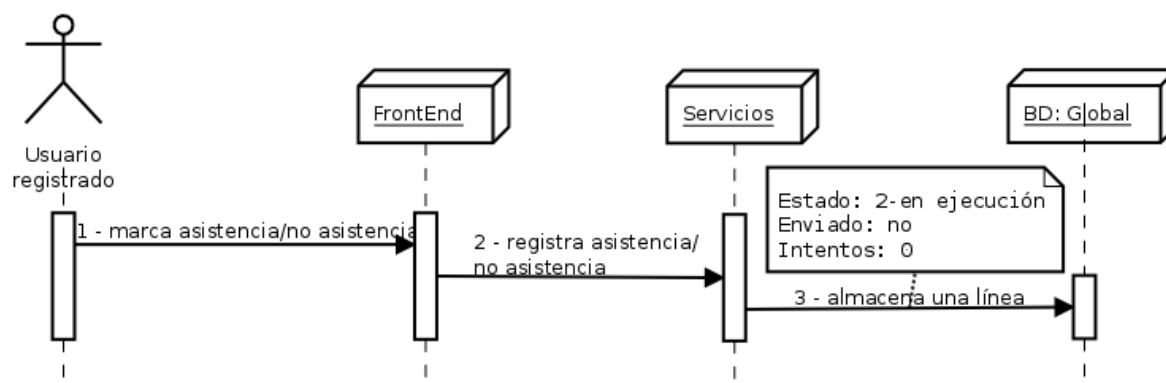


Figura 5.22: Diagrama de secuencia: marca de asistencia/no asistencia

Procedimiento automático de envío de cabezales y líneas

Cada cierto tiempo se ejecuta un proceso que determina los cabezales y líneas pendientes de envío y cuyo contador de intento de envíos no sea superior a un cierto límite configurable para intentar enviarlas. El siguiente diagrama de secuencia muestra este proceso:

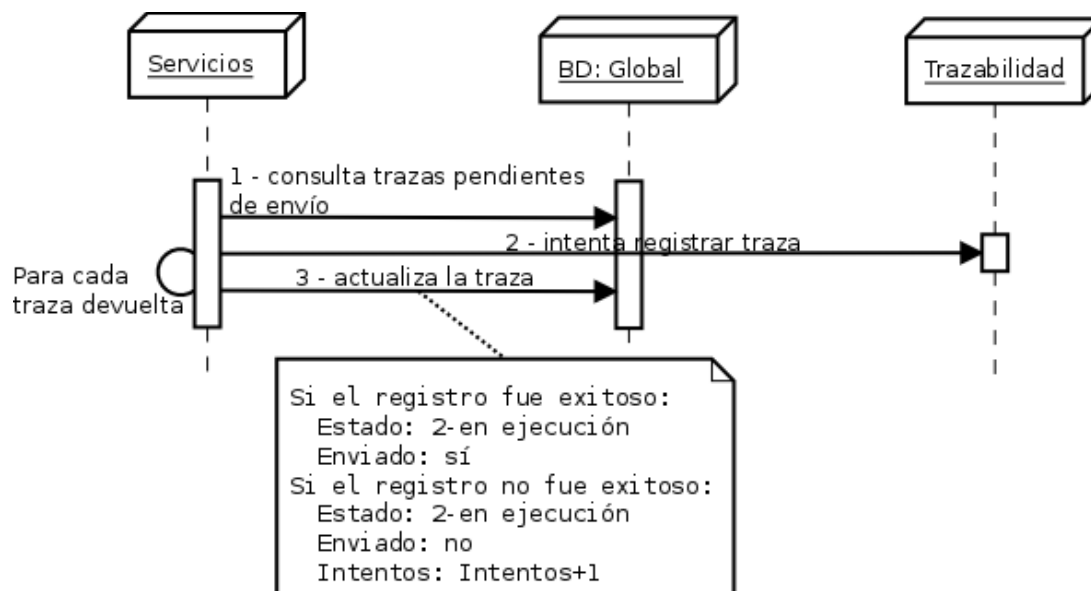


Figura 5.23: Proceso periódico de reintento de trazas

Proceso automático de cierre de procesos

Ninguno de los procesos anteriores marca una traza como finalizada. El caso más cercano es la cancelación, pero este proceso marca la traza como “Cerrada”, lo que implica que el proceso terminó de forma no normal. Para realizar el cierre de las trazas existe otro proceso que se ejecuta a la 1 de la mañana de cada día que se encarga de cerrar todas las trazas del día anterior que pudieron ser enviadas. El siguiente diagrama de secuencia muestra este proceso:

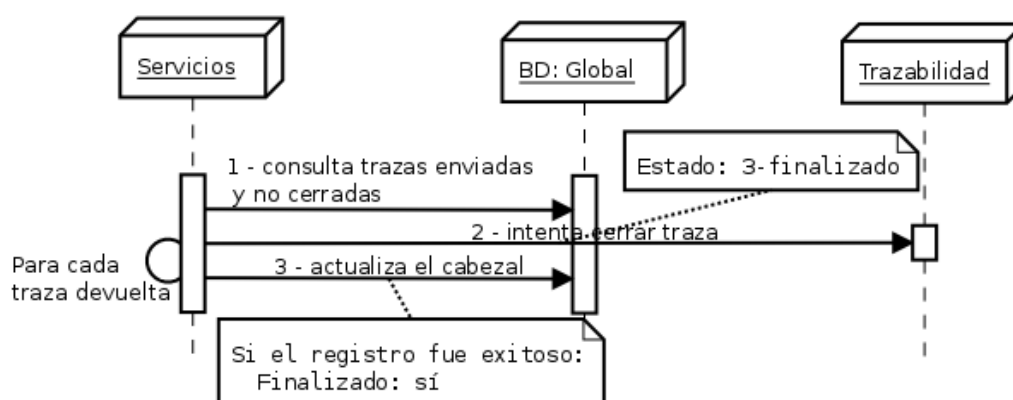


Figura 5.24: Diagrama de secuencia: cierre de trazas

5.6 Vista de Control de acceso

La aplicación puede ser configurada para que la interfaz pública (la que utilizan los ciudadanos para realizar y cancelar reservas) utilice control de acceso, autenticando ante el sistema propio de Agesic (en adelante, “CDA”), utilizando los mecanismos que estén disponibles. Para esto, la solución impone un “filtro” a todas las solicitudes que llegan de los clientes a la aplicación, el cual verifica si el objeto al que están intentando acceder es protegido por CDA entonces redirige al propio cliente al sitio de autenticación de CDA. La solución está implementada mediante una válvula del servidor de aplicaciones JBoss AS. El siguiente diagrama muestra la relación entre los elementos involucrados:

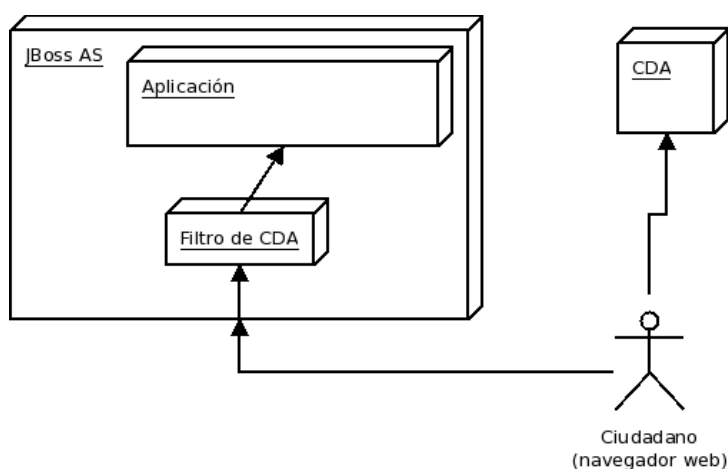


Figura 5.25: Vista de control de acceso

Como se observa en el diagrama, nunca hay interacción directa entre la aplicación y el CDA; la comunicación es siempre a través del navegador web del usuario utilizando “redirecciones automáticas” y mensajes HTTP POST.

La configuración del requerimiento de control de acceso se realiza a nivel de Agenda, es decir, cada Agenda permite especificar si requiere control de acceso o no; si una Agenda requiere control de acceso, todos sus recursos también lo hacen, mientras que si una Agenda no requiere control de acceso ninguno de sus recursos lo hace.

A continuación se muestran varios diagramas de interacción detallando cada una de las alternativas posibles:

1. La agenda accedida no requiere control de acceso.
2. La agenda accedida requiere control de acceso, es el primer acceso del usuario y el usuario no se ha autenticado ante CDA aún.
3. La agenda accedida requiere control de acceso, no es el primer acceso del usuario y el usuario ya se ha autenticado ante CDA.
4. La agenda accedida requiere control de acceso, es el primer acceso del usuario y el

usuario ya se ha autenticado ante CDA anteriormente (por solicitud de otra aplicación).

5.6.1 La agenda accedida no requiere control de acceso

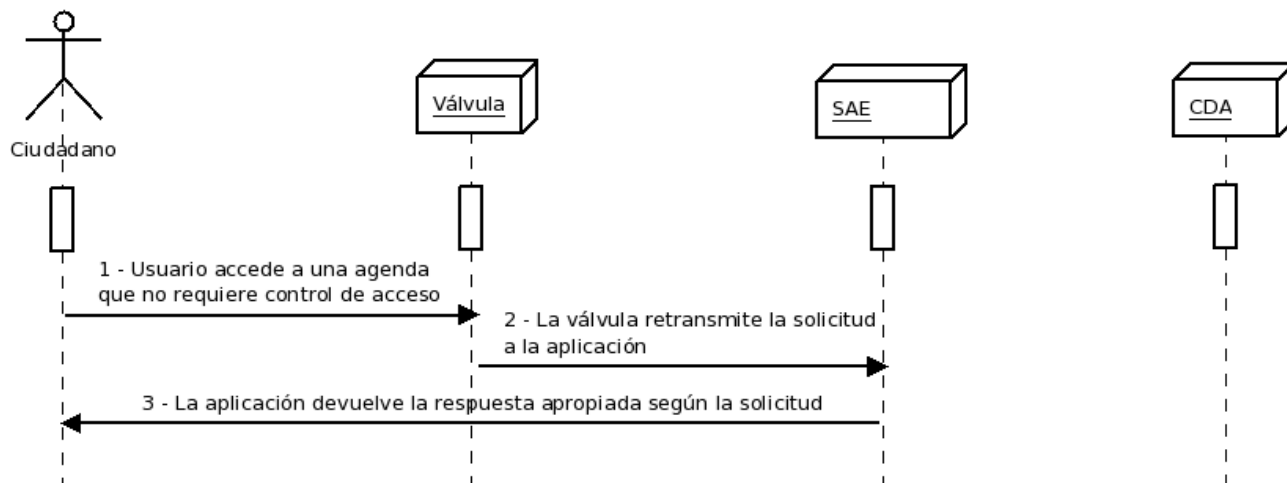


Figura 5.26: Acceso a una agenda que no requiere control de acceso

Dado que la agenda no está configurada para requerir control de acceso, la válvula permitirá que continúen hasta la aplicación normalmente.

5.6.2 La agenda accedida requiere control de acceso, es el primer acceso del usuario y el usuario no se ha autenticado ante CDA aún.

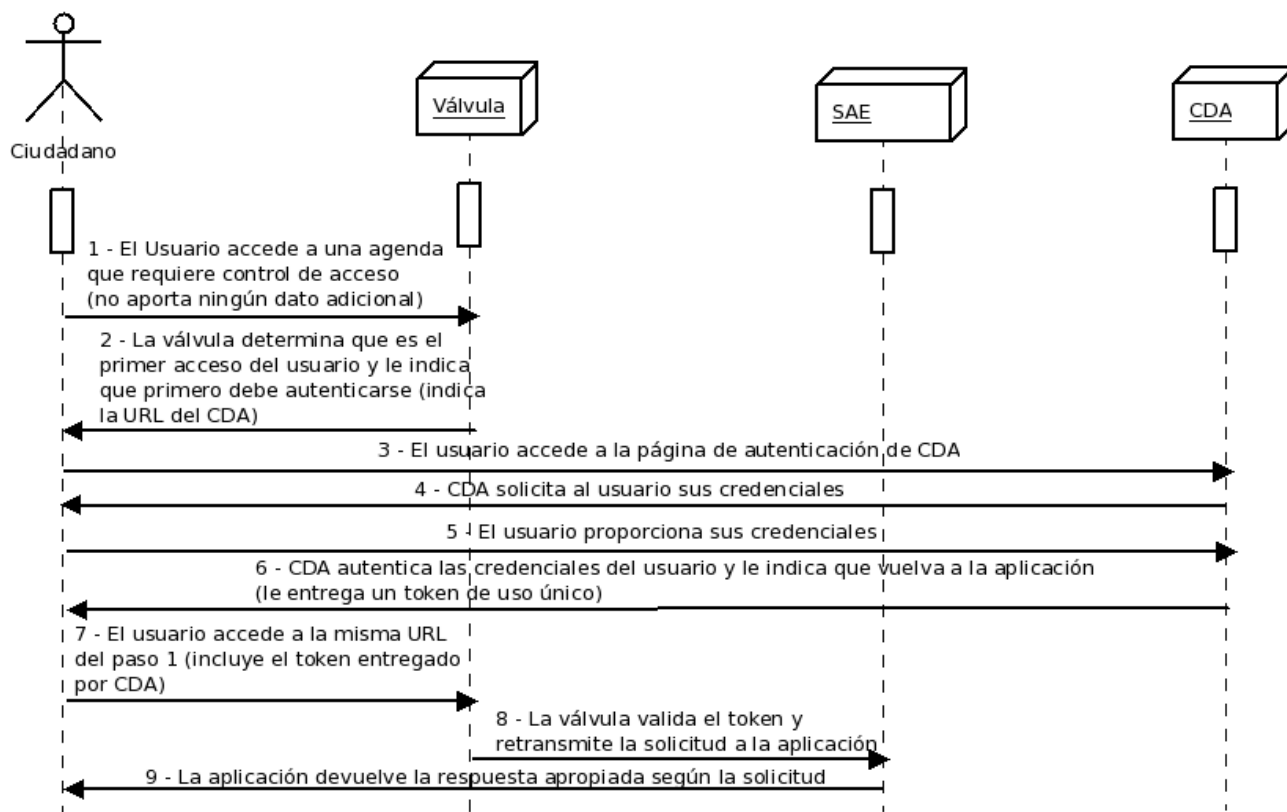


Figura 5.27: Acceso a una agenda que requiere control de acceso y es el primer acceso del usuario a la aplicación y a CDA

Dado que la agenda está configurada para requerir control de acceso la válvula no retransmite la solicitud a la aplicación. Además, determina que es la primera vez que el usuario accede a la agenda. Entonces le devuelve al usuario, como respuesta, una página HTML que contiene un formulario con un campo oculto (hidden) cuyo valor es un token SAML (firmado digitalmente por la válvula); este formulario contiene además un código JavaScript que produce que cuando el navegador del usuario cargue la página el formulario sea automáticamente enviado, mediante POST, a la URL correspondiente a la página de inicio de sesión de CDA (esto debería ejecutarse de forma tan rápida que el usuario no note la existencia de esta página intermedia). CDA determina que el usuario no se ha autenticado aún (por ninguna otra aplicación) y le solicita al usuario sus credenciales. Luego de que el usuario ingresa sus credenciales (según los métodos habilitados por CDA), el procedimiento es inverso: CDA es quien devuelve otra página HTML con un formulario, el cual también contiene un token SAML (firmado por CDA) y un código JavaScript el cual produce que automáticamente se haga un POST hacia la aplicación, el cual nuevamente será filtrado por la válvula. Ahora, dado que la solicitud (enviada por el usuario, pero construida por CDA) contiene un token SAML generado por CDA la válvula reconoce que el usuario se ha autenticado ante CDA y por lo tanto le permite el acceso, redirigiendo la solicitud a la aplicación.

5.6.3 La agenda accedida requiere control de acceso, no es el primer acceso del usuario y el usuario ya se ha autenticado ante CDA

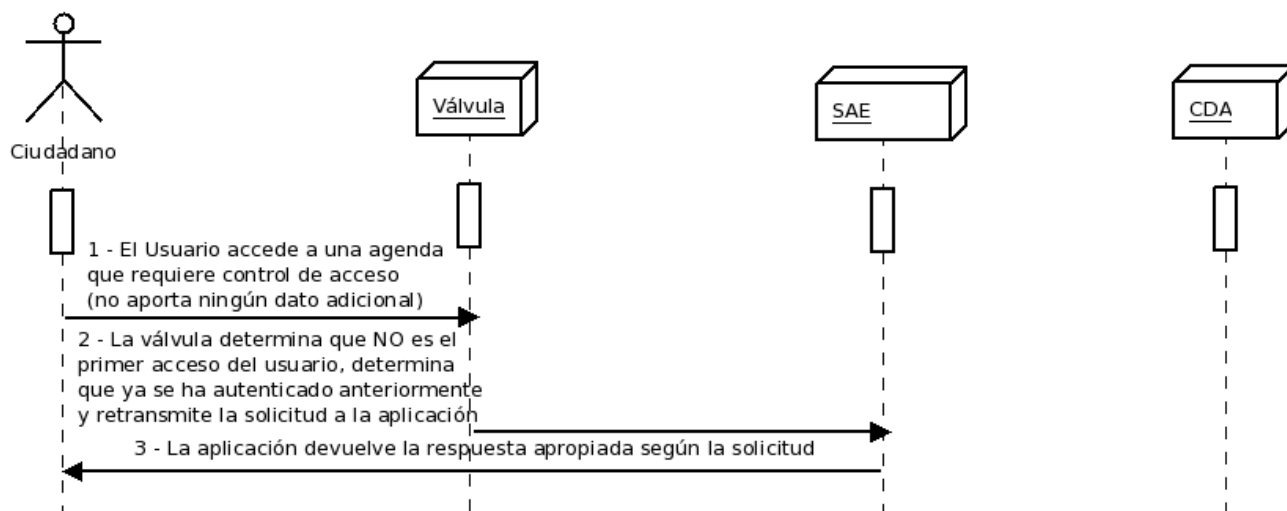


Figura 5.28: Acceso a una agenda que requiere control de acceso y no es el primer acceso del usuario

Dado que la agenda está configurada para requerir control de acceso, la válvula no retransmite la solicitud a la aplicación; en cambio, la válvula reconoce que no es la primera vez que el usuario accede a la agenda (dentro de un tiempo breve, es decir, mientras dure la “sesión”) y por lo tanto considera que el usuario ya se ha autenticado previamente ante CDA (no necesita hacerlo de nuevo) y le permite el acceso, redirigiendo la solicitud a la aplicación. Los mecanismos por los cuales la válvula reconoce que no es el primer acceso de un usuario no están especificados, pero incluyen el uso de tokens de sesión, cookies en el navegador, etc.

5.6.4 La agenda accedida requiere control de acceso, es el primer acceso del usuario y el usuario ya se ha autenticado ante CDA anteriormente (por solicitud de otra aplicación)

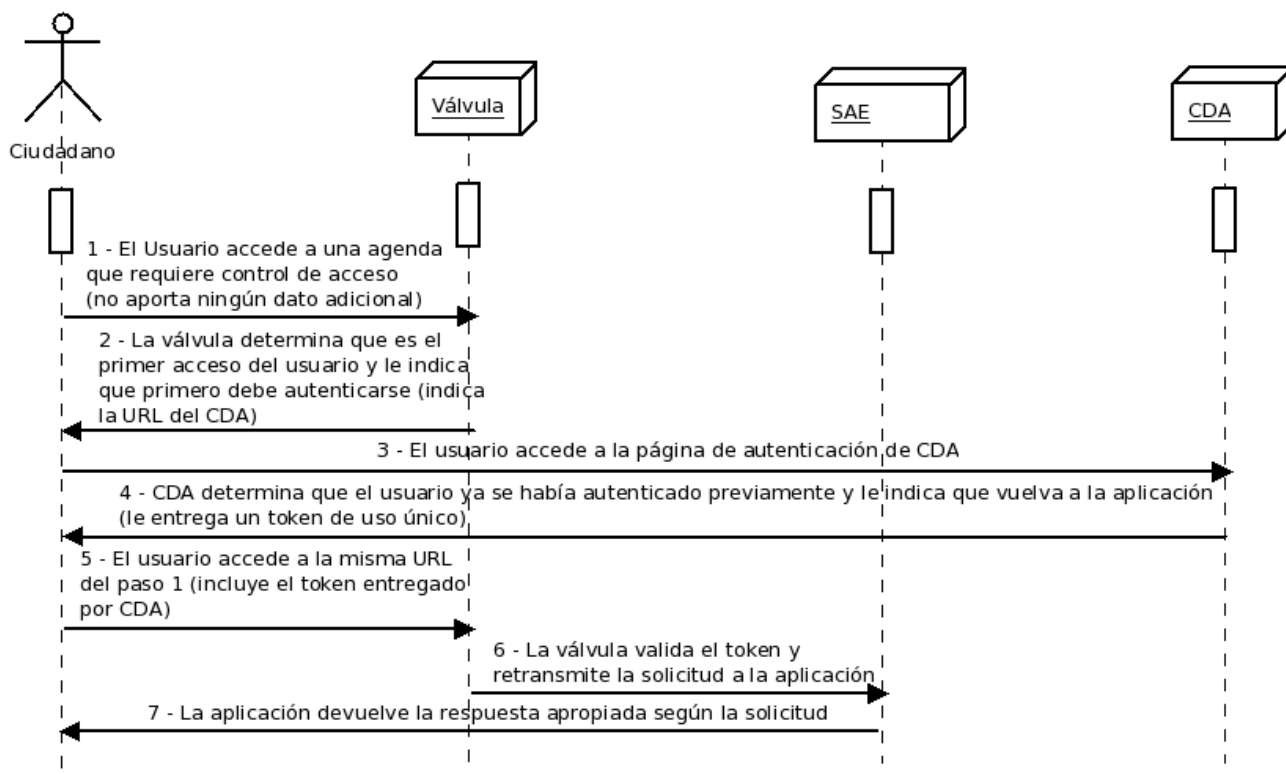


Figura 5.29: Acceso a una agenda que requiere control de acceso, es el primer acceso del usuario y el usuario ya está autenticado en CDA

Dado que la agenda está configurada para requerir control de acceso la válvula no retransmite la solicitud a la aplicación. Además, determina que es la primera vez que el usuario accede a la agenda. Entonces le devuelve al usuario, como respuesta, una página HTML que contiene un formulario con un campo oculto (hidden) cuyo valor es un token SAML (firmado digitalmente por la válvula); este formulario contiene además un código JavaScript que produce que cuando el navegador del usuario cargue la página el formulario sea automáticamente enviado, mediante POST, a la URL correspondiente a la página de inicio de sesión de CDA (esto debería ejecutarse de forma tan rápida que el usuario no note la existencia de esta página intermedia). CDA determina que el usuario ya se ha autenticado anteriormente (por esta u otra aplicación) por lo que no le solicita nuevamente sus credenciales, devolviéndole entonces otra página HTML con un formulario, el cual también contiene un token SAML (firmado por CDA) y un código JavaScript el cual produce que automáticamente se haga un POST hacia la aplicación, el cual nuevamente será filtrado por la válvula. Ahora, dado que la solicitud (enviada por el usuario, pero construida por CDA) contiene un token SAML generado por CDA la válvula reconoce que el usuario se ha autenticado ante CDA y por lo tanto le permite el acceso, redirigiendo la solicitud a la aplicación.

5.6.5 Cierre de sesión

En cualquier caso que requiera control de acceso el usuario tiene la opción de hacer un cierre de sesión (logout o signout). En estos casos, la válvula solo se encarga de comunicar a la aplicación cuál es la URL de cierre de sesión de CDA, siendo responsabilidad de la aplicación redirigir a los usuarios a dicha URL. Una vez que un usuario utiliza esta funcionalidad y el cierre de sesión es exitoso, se estaría en el caso 2 descrito anteriormente.

6 Requerimientos de Hardware

Esta sección tiene como objetivo establecer requerimientos de hardware para el despliegue de la aplicación en un ambiente de producción.

La instalación básica debe estar compuesta por dos equipos iguales, cada uno de los cuales contendrá una instalación de la aplicación (JBoss 7.1.1 + aplicación), más un servidor de bases de datos (PostgreSQL 9.x) y un dispositivo de control de acceso y balanceo de carga (no especificado).

Para los equipos en los cuales se instalará el servidor de aplicaciones se recomienda la siguiente configuración mínima

- Memoria RAM: 12 GB
 - 8 GB JBoss AS
 - 4 GB Sistema Operativo
- Espacio en disco: 80 Gigabytes.
- Conexión red: 1Gbit/s

Para el equipo donde se instalará el servidor de bases de datos se recomienda la siguiente configuración mínima:

- Memoria RAM: 8 GB
 - 4 GB PostgreSQL
 - 4 GB Sistema Operativo
- Espacio en disco: 250 Gigabytes.
- Conexión red: 1Gbit/s

7 Consideraciones Tecnológicas

A continuación se detalla el software a ejecutar en dicho sistema.

- a) Sistema Operativo: Centos, la última versión disponible al momento de realizar la instalación
- b) Java: versión 1.7
- c) Servidor de aplicaciones: JBoss 7.1.1
- d) Servidor de Base de Datos: PostgreSQL 9.x

8 Historial de cambios

Versión		Fecha	Autor	Detalle
1.0	1.0	30/11/2015	SP	Primera versión del documento
1.1	1.0	20/01/2016	SP	Se añadieron los diagramas de bases de datos
1.2	1.1	15/03/2016	SP	Añadidas las secciones de interoperabilidad y control de acceso.
1.3	1.1	18/05/2016	SP	Ajustes por cambios en los mecanismos de seguridad y control de acceso.
1.4	1.1	27/05/2016	SP	Ajustes luego de la instalación.
1.5	1.4	07/10/2016	SP	Ajustes por los cambios ocurridos en las versiones 1.2, 1.3 y 1.4.
1.6	1.4	10/10/2016	SP	Correcciones menores.
1.7	1.5	11/10/2016	RP/SP	Cambio en el título del documento.
1.8	2.0	02/02/2017	SP	Cambios por la versión 2.0. Añadida la correspondencia entre versiones del documento y la versión de la aplicación.
1.9	2.0	04/07/2017	SP	Cambio en el diagrama de despliegue y comentarios sobre la integración de los servicios web REST y CDA.