Anexos

Apéndice A

Análisis y diseño de la aplicación

En este anexo se presenta una descripción del análisis y diseño que se llevó a cabo antes de realizar la implementación de la aplicación para poder comprender completamente el sistema a desarrollar. Se mostrará el documento de especificación de requisitos, el diagrama de despliegue del sistema, y los diagramas de clases, casos de uso, secuencia y navegación. Por último se explicará la base de datos desarrollada.

A.1. Catálogo de requisitos del sistema

En este apartado se lleva a cabo la definición, análisis y validación del catálogo de requisitos obtenido a partir de la información facilitada. Con dicho catálogo se puede comprobar que la herramienta desarrollada se ajusta a los requisitos del cliente. El sistema a desarrollar tiene como objetivo cumplir una serie de requisitos funcionales y no funcionales que se resumen en la Tabla A.1.

ID.Req	Requisito
RF-1	Permitir que la herramienta pueda acoplar complementos, librerías y productos adicionales orientados al I+D que puedan mejorar las funcionalidades básicas del producto.
RF-2	Permitir el acceso a la herramienta mediante usuario y contraseña por razones de seguridad y control.
RF-3	Unificar los contenidos de los gestores de contenido actuales para facilitar las búsquedas al usuario.
RF-4	Mejorar el uso del tesauro en las búsquedas.

Continuación de la tabla

ID.Req	Requisito
RF-5	Mejorar la estructura del tesauro permitiendo multinivel y relaciones entre conceptos.
RF-6	Vincular las fichas de documentación con los ficheros sonoros con los que se relacionan.
RF-7	Elaborar una pantalla principal de formato sencillo desde la cual se permitirá los accesos a las diferentes acciones disponibles.
RF-8	Disponer de un formulario de búsqueda que permitirá realizar consultas utilizando todos los campos disponibles.
RF-9	Disponer de una pantalla de resultados de búsqueda desde la cual se podrán editar, borrar y crear nuevos documentos.
RF-10	Permitir que desde la pantalla de resultados se puedan seleccionar los elementos encontrados para visualizarlos.
RF-11	Disponer de una pantalla de presentación de documentos desde la cual se podrán editar la información de estos.
RF-12	Disponer de una opción donde poder gestionar de forma cómoda el contenido del tesauro de la organización.
RF-13	Disponer de una configuración desde la cual se podrá gestionar la información almacenada en la base de datos.
RF-14	Permitir la gestión de los usuarios de la aplicación y sus permisos.
RF-15	Poder visualizar las acciones realizadas por los usuarios.
RF-16	Generar de forma automática informes de contenido en formato PDF.
RF-17	Generar de forma automática resúmenes de contenido en formato TXT.
RF-18	Permitir la posibilidad de copiar registros para evitar cumplimentar los mismos campos cuando solo se necesita modificar uno.
RF-19	La interfaz deberá ser sencilla, manejable e intuitiva.

Continuación de la tabla

ID.Req	Requisito
RNF-1	Agilizar el uso del tesauro cuando se cumplimenta una ficha de contenido.
RNF-2	Mejorar la interfaz del tesauro para el usuario.
RNF-3	Mejorar la presentación de los registros seleccionados.
RNF-4	La herramienta ha de ser eficaz, rápida y sencilla.
RNF-5	Aumentar la productividad del departamento de documentación.

Tabla A.1: Requisitos de Cliente.

A.2. Descripción del sistema

A continuación se adjunta un diagrama de despliegue del sistema a implementar (Figura A.1):

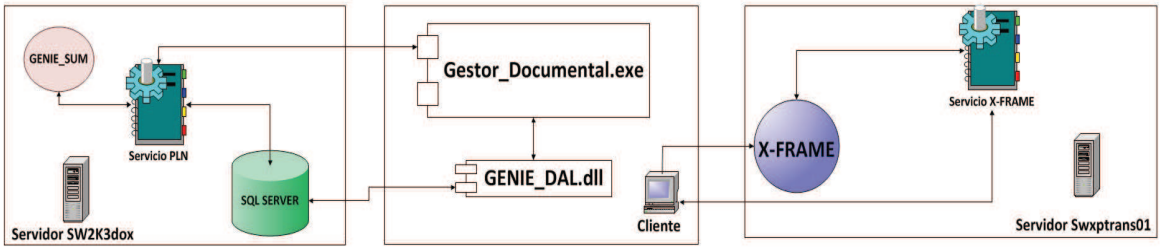

Figura A.1: Diagrama de despliegue del sistema

Los elementos que se desarrollarán son los siguientes:

- Servicio de Windows que automatice las tareas con X-FRAME desde el cliente.
- Ejecutable de escritorio que inicie la herramienta de gestión.

Y se reutilizarán los siguientes componentes ya desarrollados:

- Biblioteca "itextsharp.dll" que permiten crear, adaptar, inspeccionar y mantener documentos en formato PDF para C#. Esta biblioteca ha sido desarrollada por uno de los principales desarrolladores de *iText*, Paulo Soares.

- Librería de clase dinámica que conecta con la base de datos y permite realizar las consultas de manera sencilla (GENIE_DAL.dll). Esta biblioteca, en cuyo desarrollo participé, se ha reutilizado en proyectos del grupo SID de la Universidad de Zaragoza con éxito.

- Servicio de Procesamiento del Lenguaje Natural (Servicio PLN) que contiene todo lo necesario para lematizar la base de datos. Este servicio ha sido desarrollado en el departamento de Sistemas de Información Distribuida (SID) de la Universidad de Zaragoza.

- GENIE_SUM es la unidad que he desarrollado para generar resúmenes automáticos de contenido. Esta unidad hace uso del Servicio PLN para su funcionamiento, y se ha utilizado en varios proyectos del grupo SID de la Universidad de Zaragoza.

- Ficheros *"Segoeuil Segoeuil.ttf"* y *"Segoe Ui Semilight.ttf"*, los cuales permiten que la fuente de datos utilizada sea igual a la de *Windows 8* [2], lo que da un aspecto más moderno a la interfaz de la herramienta. La diferencia entre ambos ficheros es el grosor de las letras, lo cual es útil para diferenciar títulos de otro contenido dentro los formularios que se presentan al usuario. Estos ficheros están desarrollados por *Microsoft Corporation*.

A.3. Diagramas UML de Análisis

En este apartado, se adjuntan los diagramas UML que se han estimado necesarios para ayudar en la comprensión del sistema:

A.3.1. Casos de uso

En esta sección se van a analizar los posibles casos que podrán darse en la herramienta, mediante algunos diagramas de caso de uso. En primer lugar, las diferentes opciones que ofrece la herramienta pueden verse en la Figura A.2.

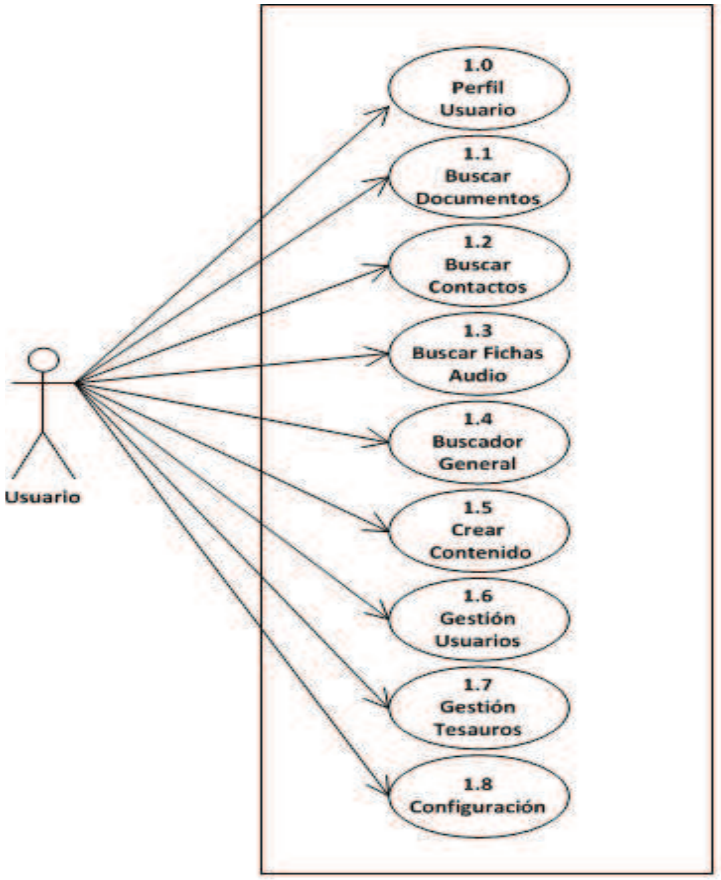

Figura A.2: Diagrama de casos de uso de la herramienta

Caso de uso: 1.0 Perfil Usuario

Propósito:	Acceder al perfil del usuario
Actor principal:	Administrador, o usuario con permisos de gestión de usuarios y cambio de clave.
Precondiciones:	Estar conectado correctamente.
Descripción:	En esta opción el usuario conectado puede acceder a sus datos personales.
Flujo Básico:	Acceder a la opción de perfil.
Flujo Alternativo:	Hay problemas al leer de la base de datos. Se notificará de ello al usuario, y se cancelará la operación.

Caso de uso: (1.1, 1.2, 1.3, 1.4) Búsquedas

Propósito:	Acceder a la pantalla de búsqueda correspondiente.
Actor principal:	Administrador, o usuario con permisos de lectura de documentos.
Precondiciones:	Estar conectado correctamente.
Descripción:	En esta opción el usuario conectado puede proceder a realizar una búsqueda de contenido.
Flujo Alternativo:	Hay problemas al leer de la base de datos. Se notificará de ello al usuario, y se cancelará la operación. Datos inválidos al realizar la búsqueda: se informa al usuario.

Caso de uso: 1.5 Crear Contenido

Propósito:	Acceder a la pantalla crear contenido.
Actor principal:	Administrador, o usuario con permisos de gestión de documentos.
Precondiciones:	Estar conectado correctamente.
Descripción:	En esta opción el usuario conectado puede proceder crear nuevo contenido e insertar el mismo en la base de datos. Para ello debe seleccionar la categoría a la que pertenece el nuevo contenido, y dependiendo de la categoría escogida el formulario varía sus campos a rellenar.
Flujo Alternativo:	Hay problemas al leer de la base de datos. Se notificará de ello al usuario, y se cancelará la operación. Datos inválidos al crear el contenido: se informa al usuario.

Caso de uso: 1.6 Gestión Usuarios

Propósito:	Acceder a la pantalla gestión de usuarios
Actor principal:	Administrador, o usuario con permisos de gestión de usuarios y cambio de clave.
Precondiciones:	Estar conectado correctamente.
Descripción:	En esta opción el usuario conectado puede proceder a gestionar la información almacenada para todos los usuarios. Esa gestión consiste en editar datos actuales de un usuario, borrar un usuario o crear uno nuevo. Si solo se posee privilegio de cambio de clave, únicamente se permite acceder a la edición de clave, prohibiendo la eliminación de otros usuarios o la creación de uno nuevo.
Flujo Alternativo:	Hay problemas al leer de la base de datos. Se notificará de ello al usuario, y se cancelará la operación. Datos inválidos al insertar un nuevo usuario: se informa al usuario.

Caso de uso: 1.7 Gestión Tesauros

Propósito:	Acceder a la pantalla gestión de usuarios.
Actor principal:	Administrador, o usuario con permisos de gestión de tesauros.
Precondiciones:	Estar conectado correctamente.
Descripción:	En esta opción el usuario conectado puede proceder a gestionar la información almacenada de los tesauros. Esa gestión permite editar los términos existentes en el tesauro (cambiar su nombre o moverlo de un posición a otra dentro del árbol jerárquico de tesauros), buscar un término, añadir un nuevo término en la posición deseada, y eliminar un término en concreto.
Flujo Alternativo:	Hay problemas al leer de la base de datos. Se notificará de ello al usuario, y se cancelará la operación. Datos inválidos al insertar o editar un término: se informa al usuario.

Caso de uso: 1.8 Configuración

Propósito:	Acceder a la pantalla gestión de usuarios.
Actor principal:	Administrador, o usuario con permisos de gestión de campos.
Precondiciones:	Estar conectado correctamente.
Descripción:	En esta opción el usuario conectado puede proceder a gestionar la información sobre los distintos tipos y categorías que puede tener un contenido en concreto. El usuario puede borrar/añadir tipos de contenido y categorías. En el caso de las categorías, se permite editar los diversos campos que debe poseer esa categoría en concreto.
Flujo Alternativo:	Hay problemas al leer de la base de datos. Se notificará de ello al usuario, y se cancelará la operación. Datos inválidos al insertar o editar tipos y categorías: se informa al usuario.

Para cada uno de estos casos de uso se va a mostrar un ejemplo. En la Figura A.3 podemos ver una posible traza de eventos para el caso de búsqueda de contenido en la herramienta, ya sean documentos, contactos, audios, o una búsqueda general por todos los campos disponibles.

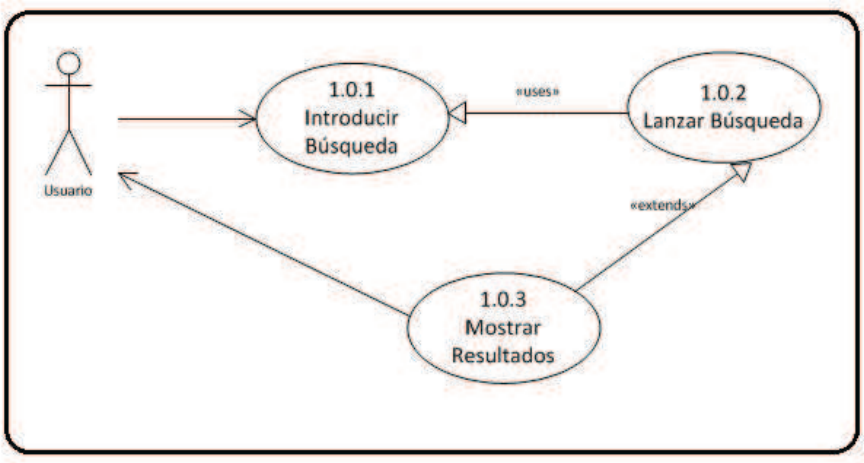

Figura A.3: Caso de uso *buscar* (1.1, 1.2, 1.3, 1.4)

Caso de uso: 1.0.1 Introducir búsqueda

Propósito:	Introducir la búsqueda deseada.
Actor principal:	Administrador, o usuario con permisos de lectura de documentos.
Precondiciones:	Estar conectado correctamente.
Descripción:	En esta opción el usuario conectado puede realizar una búsqueda según los campos deseados. Para ello debe rellenar el formulario mostrado en la pantalla.
Flujo Alternativo:	Datos inválidos al insertar o editar un tesauro: se informa al usuario.

Caso de uso: 1.0.2 Lanzar Búsqueda

Propósito:	Lanzar la búsqueda realizada a la base de datos.
Precondiciones:	Estar conectado correctamente.
Descripción:	Cuando el usuario ha pulsado el botón buscar, se procede a lanzar la consulta a la base de datos para que esta devuelva los resultados encontrados.
Flujo Alternativo:	Hay problemas al leer de la base de datos. Se notificará de ello al usuario, y se cancelará la operación.

Caso de uso: 1.0.3 Mostrar Resultados

Propósito:	Mostrar los resultados encontrados al usuario.
Descripción:	Una vez realizada la consulta a la base de datos, si esta se ha efectuado con éxito, se mostrará en una pantalla un listado con todos los resultados encontrados. En caso de no encontrar ningún resultado, se informará de ello mediante un mensaje por pantalla.
Flujo Alternativo:	Hay problemas al leer de la base de datos. Se notificará de ello al usuario, y se cancelará la operación.

En la Figura A.4, se puede ver un ejemplo de posible escenario de uso cuando se ha realizado una búsqueda y se muestran los resultados encontrados. Desde

el listado de resultados, se puede acceder a toda la información disponible del resultado seleccionado, un ejemplo de posible escenario en este caso, es el mostrado en la Figura A.5.

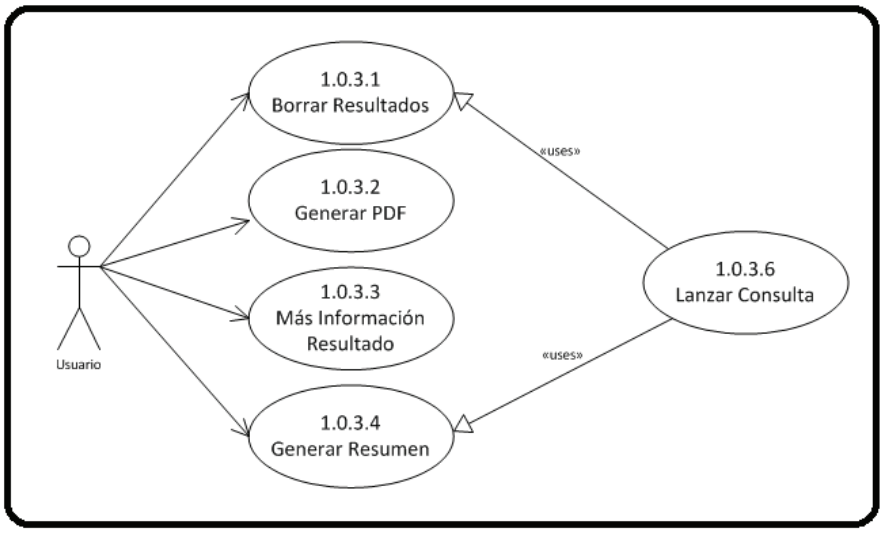

Figura A.4: Caso de uso *listar* (1.0.3)

Caso de uso: 1.0.3.1 Borrar Resultados

Propósito:	Borrar los resultados deseados del listado.
Descripción:	En el listado de resultados devuelto al realizar una búsqueda en concreto, se permite eliminar aquel contenido que no se desea. Para ello se dispone de un botón para dicho propósito, así como la posibilidad de pulsar la tecla *"Supr"* del teclado para efectuar el borrado. Para asegurarse de que no es una equivocación por parte del usuario, se le muestran varios mensajes de aviso.
Flujo Alternativo:	Hay problemas al leer de la base de datos. Se notificará de ello al usuario, y se cancelará la operación.

Caso de uso: 1.0.3.2 Genera PDF

Propósito:	Generar un fichero PDF de los resultados deseados del listado.
Descripción:	En el listado de resultados devuelto al realizar una búsqueda en concreto, se permite generar un fichero PDF con el contenido de uno o varios resultados seleccionados. Para ello el usuario debe seleccionar aquel directorio donde desea almacenar el fichero PDF.
Flujo Alternativo:	Hay problemas al generar el fichero PDF. Se notificará de ello al usuario, y se cancelará la operación.

Caso de uso: 1.0.3.3 Más Información Resultado

Actor principal:	Administrador, o usuario con permisos de lectura de documentos. Si no posee permisos de gestión de documentos, no podrá editar ningún dato.
Propósito:	Acceder a la información almacenada de un contenido en concreto.
Descripción:	En el listado de resultados devuelto al realizar una búsqueda en concreto, se permite acceder a toda la información almacenada sobre el resultado seleccionado.

Caso de uso: 1.0.3.4 Genera Resumen

Propósito:	Generar un resumen de los resultado deseados del listado.
Descripción:	En el listado de resultados devuelto al realizar una búsqueda en concreto, se permite generar un resumen con el contenido de uno o varios resultados seleccionados. Para ello el usuario debe seleccionar aquellos resultados que se desean agrupar y resumir, junto al directorio donde se desea almacenar el resumen generado.
Flujo Alternativo:	Hay problemas al conectarse con el servidor externo, o al generar el resumen. Se notificará de ello al usuario, y se cancelará la operación.

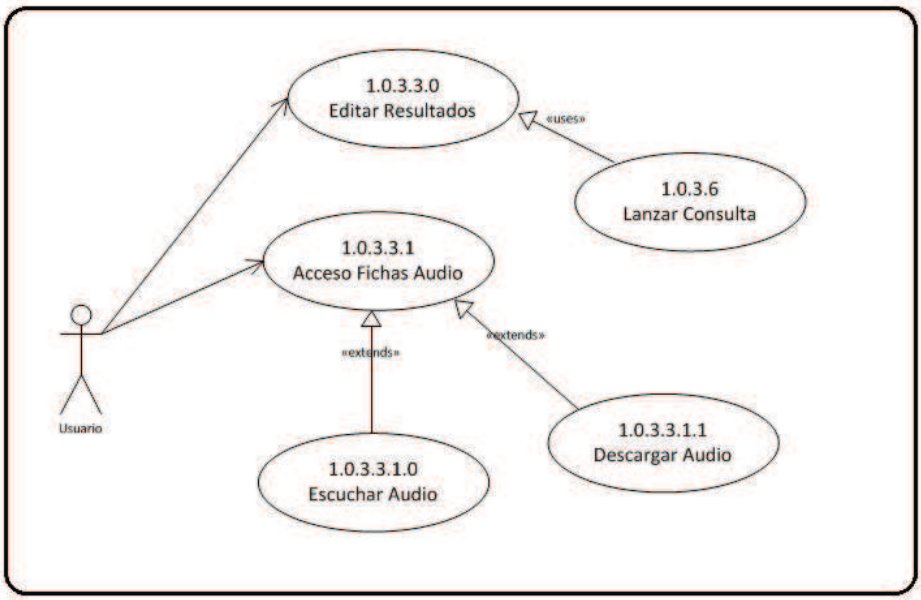

Figura A.5: Caso de uso *más información* (1.0.3.3)

Caso de uso: 1.0.3.3.0 Editar Resultados

Actor principal:	Administrador, o usuario con permisos de gestión de documentos.
Propósito:	Editar los campos deseados de la información almacenada del contenido seleccionado.
Descripción:	En la pantalla que contiene toda la información acerca del contenido seleccionado en el listado de resultados, se permite editar cualquiera de los campos almacenados y actualizarlos en la base de datos.
Flujo Alternativo:	Hay problemas al conectarse con la base de datos, o algún dato introducido no es válido. Se notificará de ello al usuario, y se cancelará la operación.

Así mismo, en los diagramas de caso de uso siguientes, se mostrará un ejemplo de uso cuando el usuario accede a: crear nuevo contenido (Figura A.7), gestionar usuarios (Figura A.6) y gestionar tesauros (Figura A.8).

Caso de uso: 1.0.3.3.1 Acceso Fichas Audio

Actor principal:	Administrador, o usuario con permisos de gestión de documentos.
Propósito:	Acceso a la información almacenada de las fichas de audio que contiene el contenido seleccionado.
Descripción:	En la pantalla que contiene toda la información acerca del contenido seleccionado en el listado de resultados, se permite visualizar todas las fichas de audio vinculadas a dicho contenido. Ese contenido también puede ser editado.
Flujo Alternativo:	Hay problemas al conectarse con la base de datos, o algún dato introducido no es válido. Se notificará de ello al usuario, y se cancelará la operación.

Caso de uso: 1.0.3.3.1.0 Escuchar Audio

Propósito:	Poder reproducir el fichero de audio seleccionado.
Descripción:	En la pantalla que contiene toda la información acerca del contenido seleccionado en el listado de resultados, se permite visualizar todas las fichas de audio vinculadas a dicho contenido. Si se desea se puede acceder mediante un botón a la reproducción automática del fichero de audio seleccionado. El fichero se reproduce mediante VLC media.
Flujo Alternativo:	Hay problemas al conectarse con VLC media o al traer el fichero de audio desde el servidor al ordenador cliente. Se notificará de ello al usuario, y se cancelará la operación.

Caso de uso: 1.0.3.3.1.1 Descargar Audio

Propósito:	Poder descargar el fichero de audio seleccionado.
Descripción:	Desde la pantalla de información del contenido seleccionado, se encuentra habilitada la opción de descargar el fichero de audio en el directorio deseado.
Flujo Alternativo:	Hay problemas al traer el fichero de audio desde el servidor al ordenador cliente. Se notificará de ello al usuario, y se cancelará la operación.

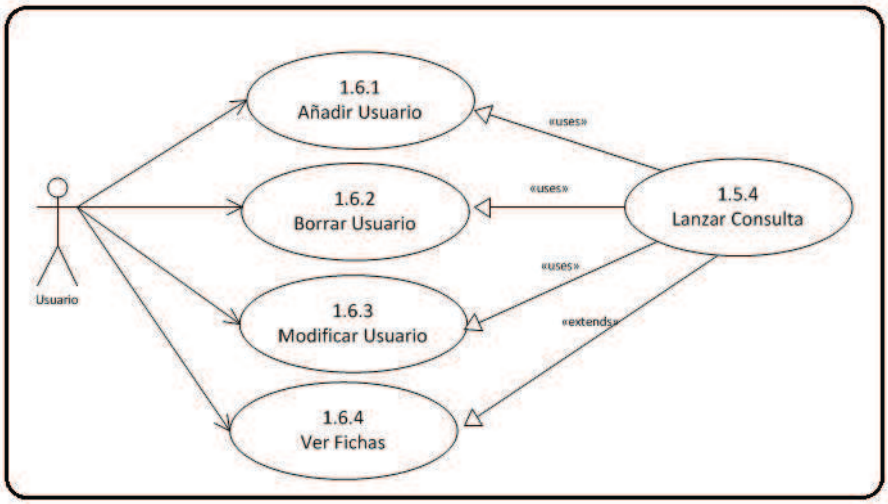

Figura A.6: Caso de uso *gestión usuarios* (1.6)

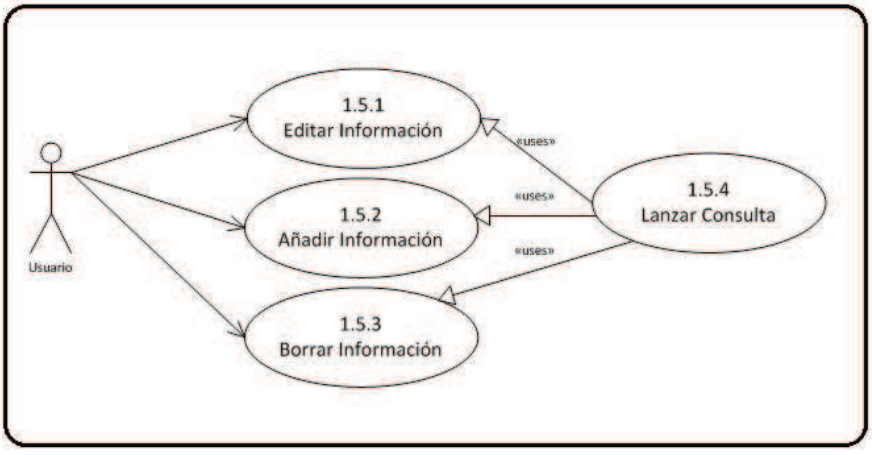

Figura A.7: Caso de uso *crear contenido* (1.5)

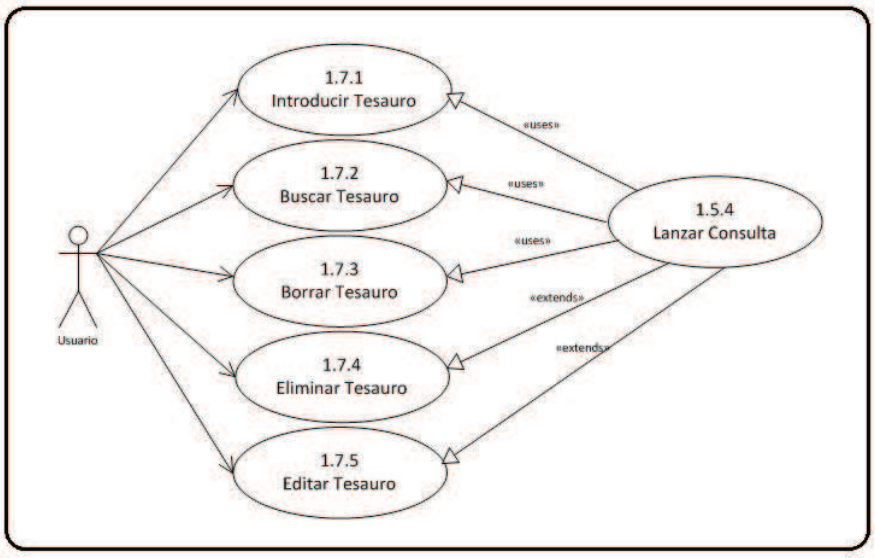

Figura A.8: Caso de uso *gestión tesauros* (1.7)

A.3.2. Diagramas de secuencia de eventos

En esta sección se van a mostrar diferentes diagramas de secuencia de la herramienta. En concreto, el primer diagrama de secuencia A.9 muestra la traza de eventos al iniciar el sistema.

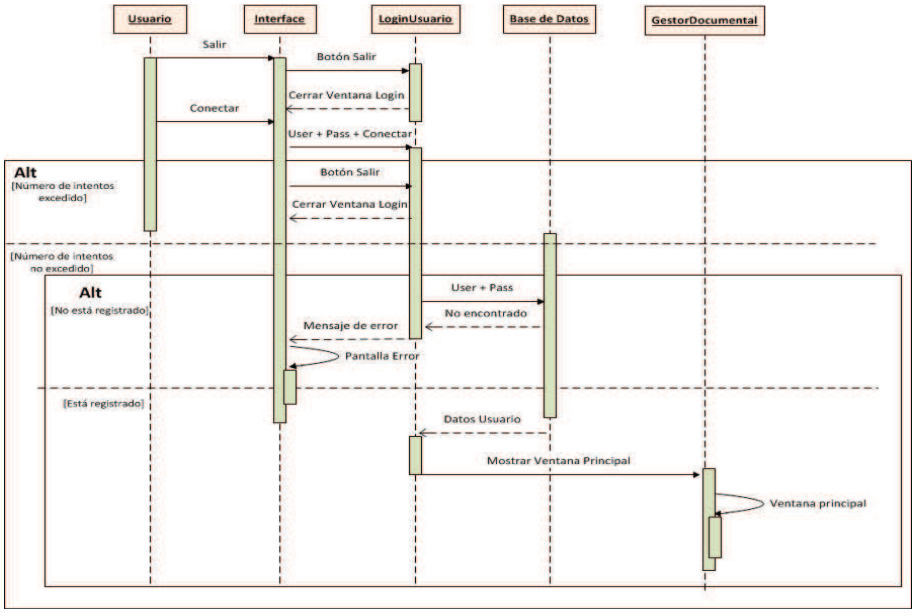

Figura A.9: Ejemplo de traza de eventos para la iniciación del sistema

Dado que una de las opciones más complejas dentro de la herramienta es la búsqueda de contenido (porque ofrece muchas opciones una vez mostrados los resultados) se ha decidido mostrar la traza de eventos genérica para cualquiera de los 4 buscadores disponibles en la aplicación (buscador de contenido, buscador de audio, buscador de contactos y buscador general).

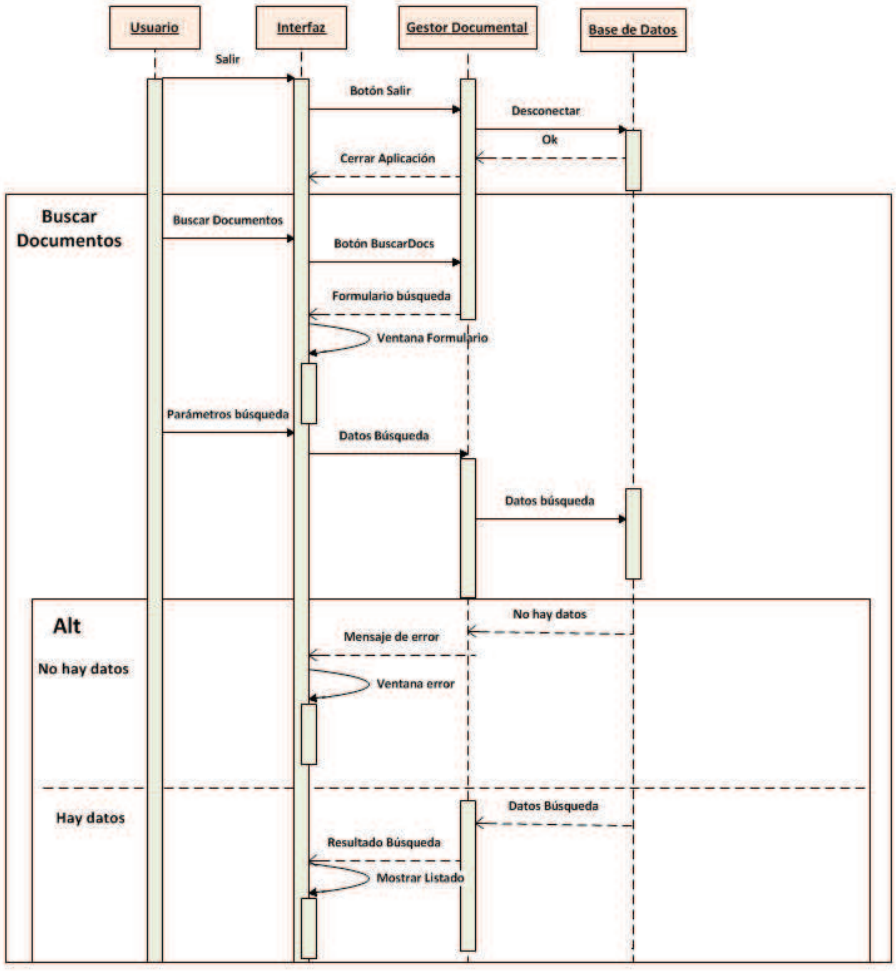

Figura A.10: Diagrama de secuencia *búsqueda*

Como la traza conlleva muchos eventos, se ha decidido dividir el diagrama de secuencias en tres partes. La primera de ellas, la cual se puede ver en la Figura A.10, muestra la traza de eventos que se lleva a cabo para mostrar al usuario los resultados de la búsqueda efectuada. En la segunda Figura A.11 se puede observar las diferentes trazas que pueden darse dentro de la pantalla de listado de resultados, y por último en la Figura A.12 se puede ver en más detalle la opción

"Más Información del resultado", a la cual se puede acceder desde el listado de resultados, haciendo clic sobre el resultado deseado.

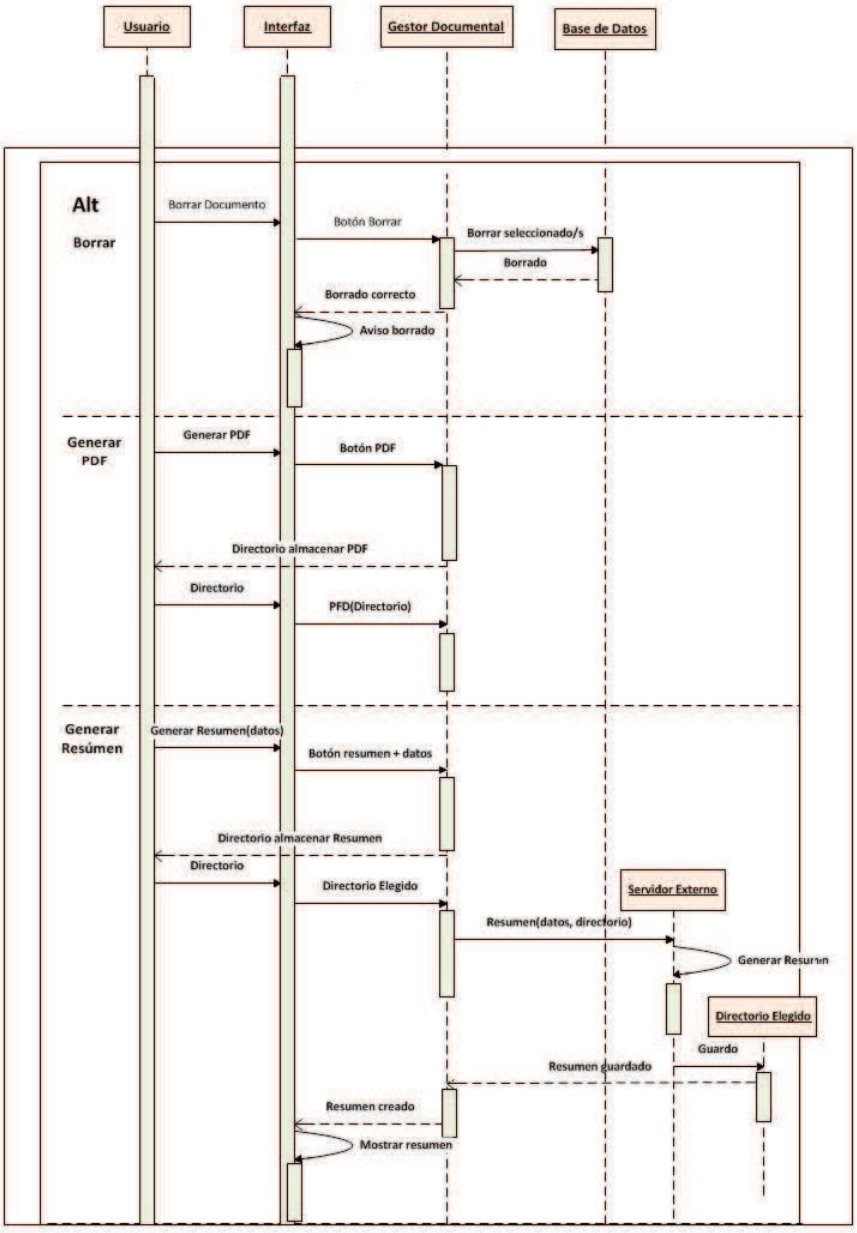

Figura A.11: Diagrama de secuencia *listado de resultados*

Por último mostrar la traza de eventos para la opciones de creación de contenido (Figura A.13), gestión de usuarios (Figura A.14) y gestión de tesauros (Figura A.15). Remarcar que en estos dos últimos diagramas no se han añadido todas las

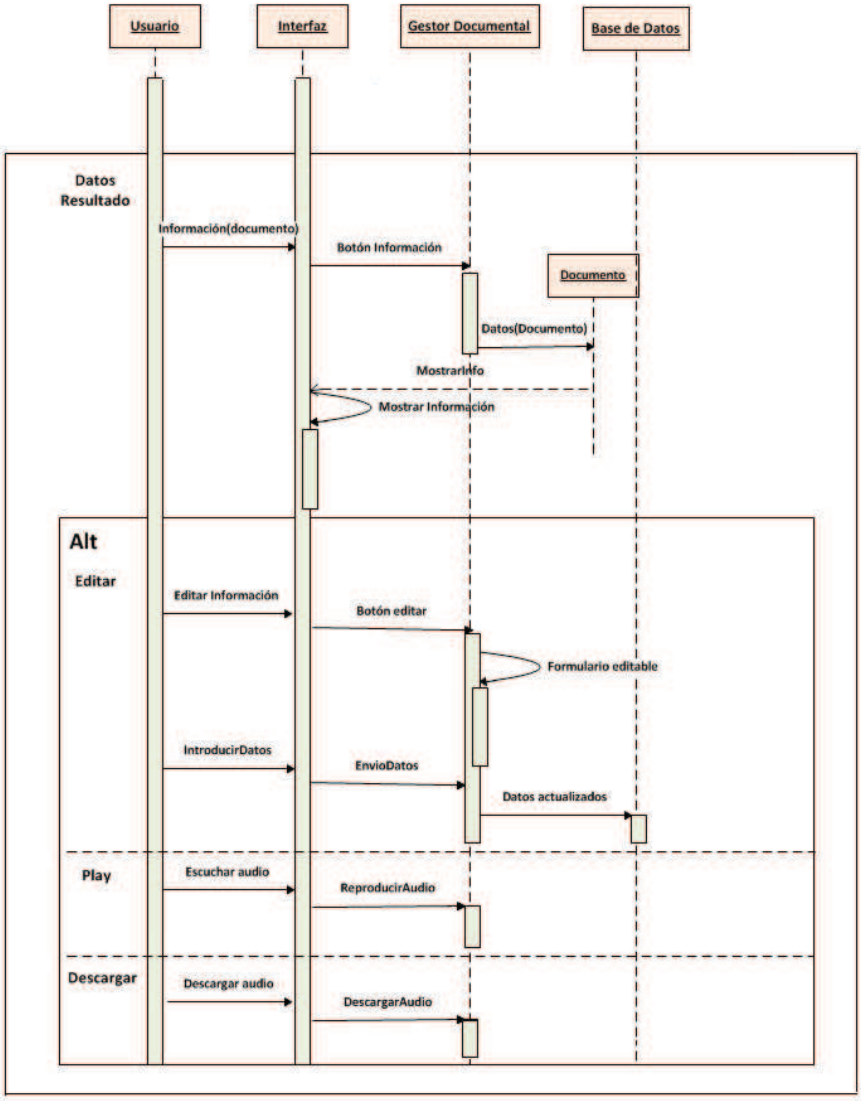

Figura A.12: Diagrama de secuencia *información del resultado*

trazas de eventos posibles por no recargar el diagrama de secuencia demasiado. En ambos, se dispone también de las opciones de editar y borrar y la traza de eventos sería similar a la de añadir que ambos muestran en sus diagramas de secuencia.

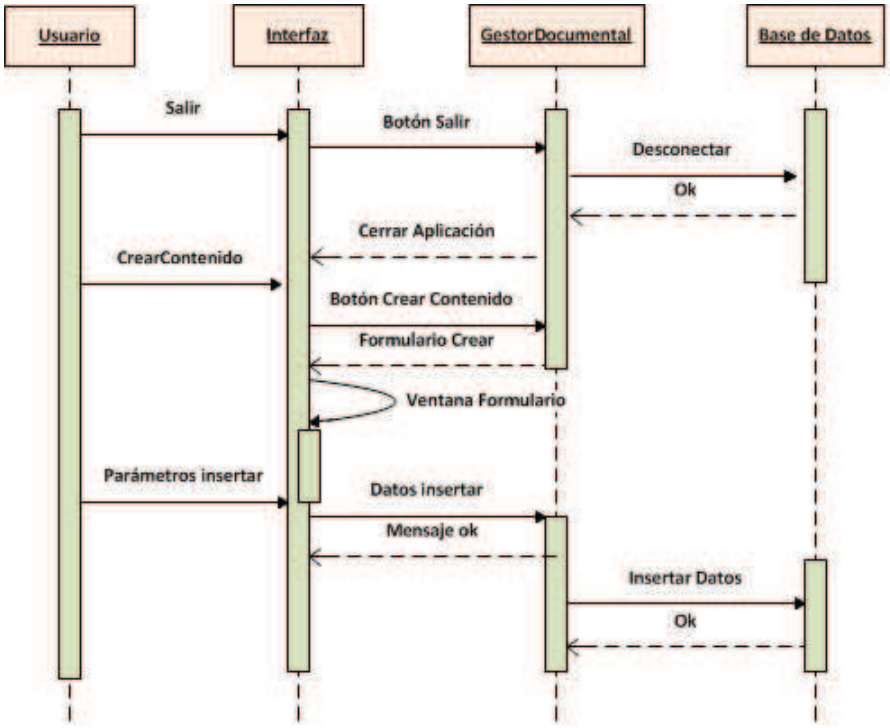

Figura A.13: Diagrama de secuencia *creación de contenido*

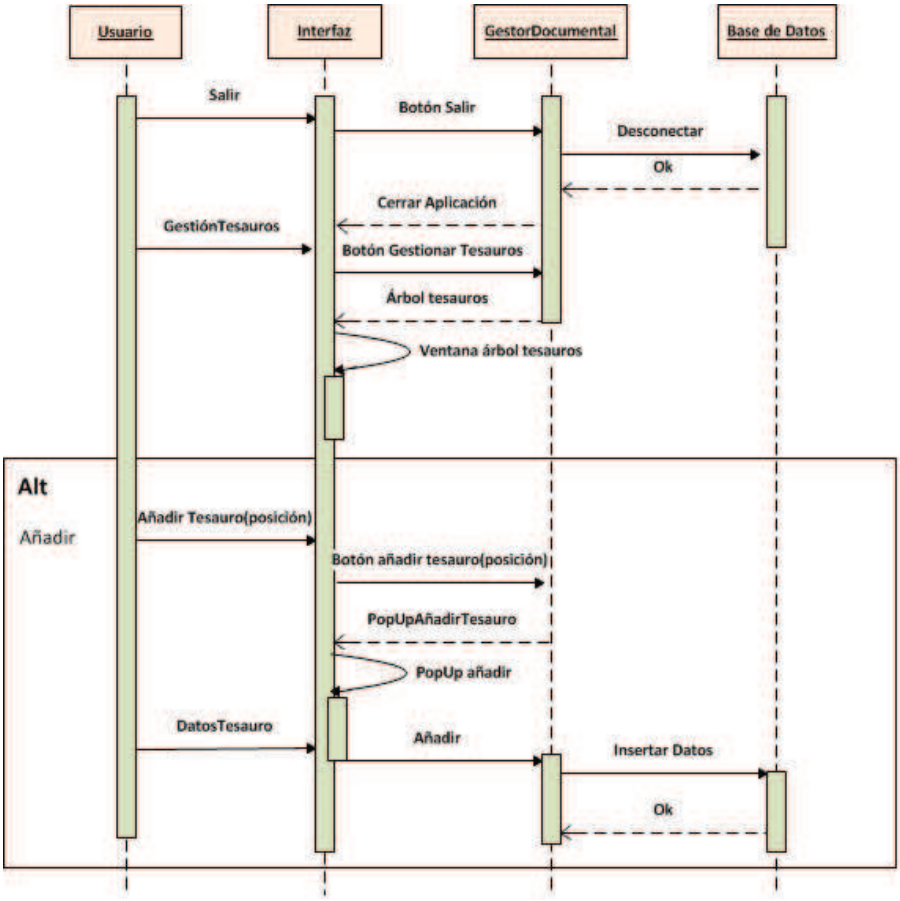

Figura A.14: Diagrama de secuencia *gestión de usuarios*

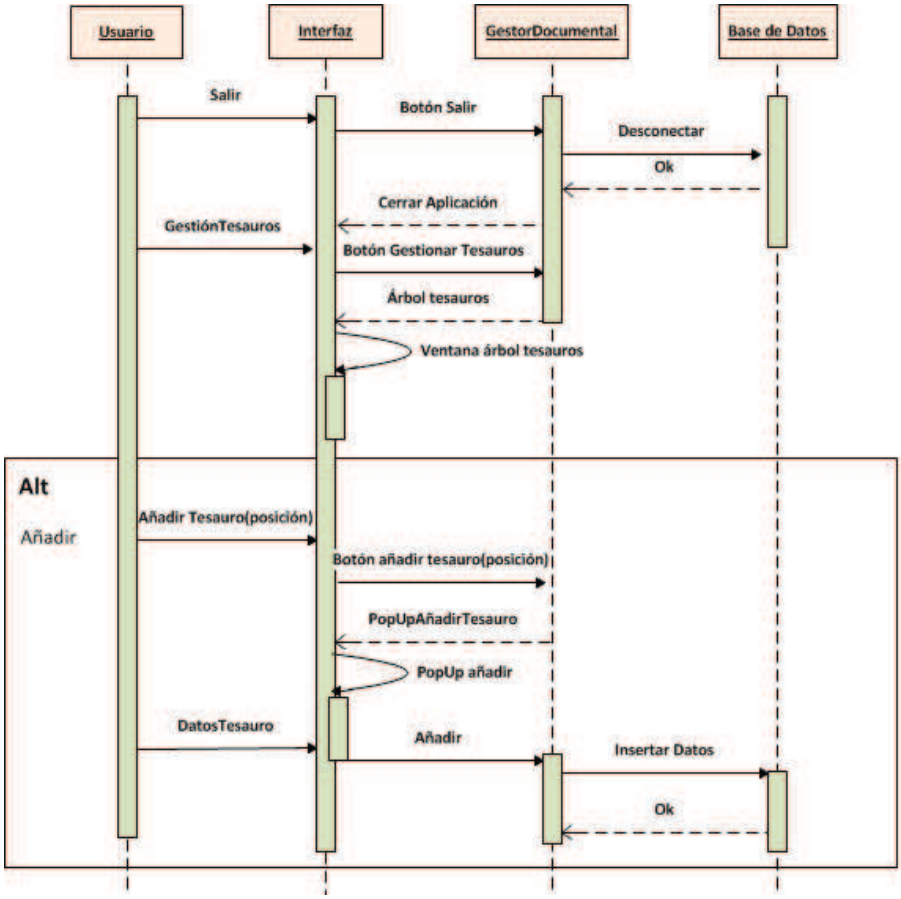

Figura A.15: Diagrama de secuencia *gestión de tesauros*

A.3.3. Clases y objetos

Dada la complejidad del diagrama de clases y objetos, debido al gran número de estas, se ha decidido mostrar por un lado el diagrama de clases para la ventana principal, y por otro el diagrama de clases para cada opción disponible en la herramienta. La nomenclatura utilizada en todos los diagramas es la siguiente: el prefijo "Pantalla" se añade en aquellas clases que implementan formularios, el prefijo "Data" representa la capa de datos de la herramienta, y por último el prefijo "Configuración" indica las clases que pertenecen a los formularios de la opción de configuración del sistema.

- **Ventana Principal:** En la Figura A.16 se puede observar el diagrama de clases de la ventana principal, la cual está formado por un menú que contiene todas las opciones disponibles en la herramienta (las clases en color naranja son proporcionadas por C# .Net y la tecnología WPF).

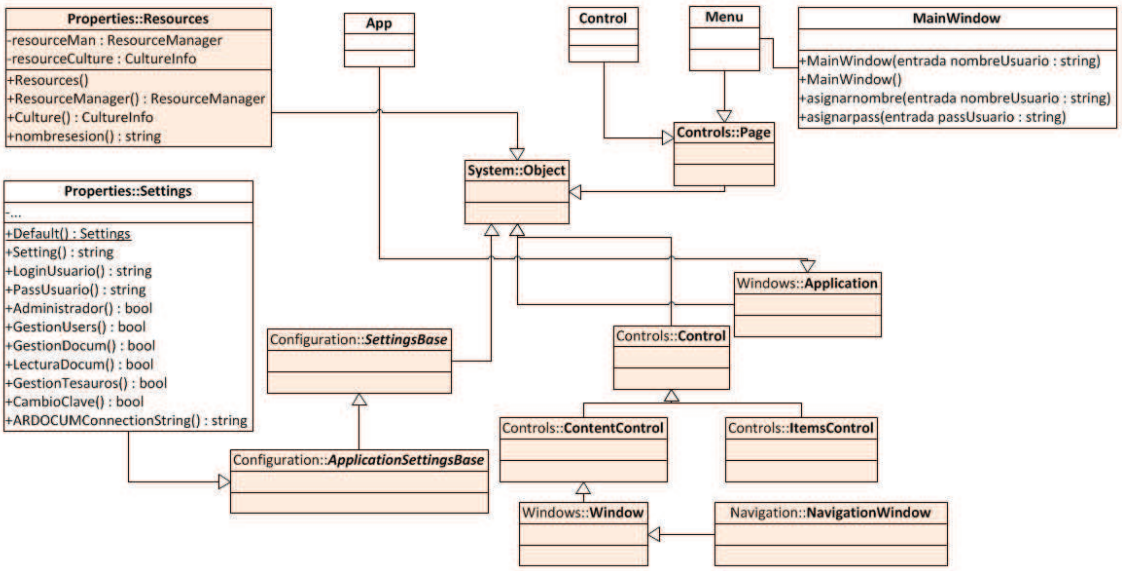

Figura A.16: Diagrama de clases ventana principal

- **Opciones de búsqueda en el menú** En la Figura A.17 se puede observar el diagrama de clases para las opciones de búsqueda de la herramienta.

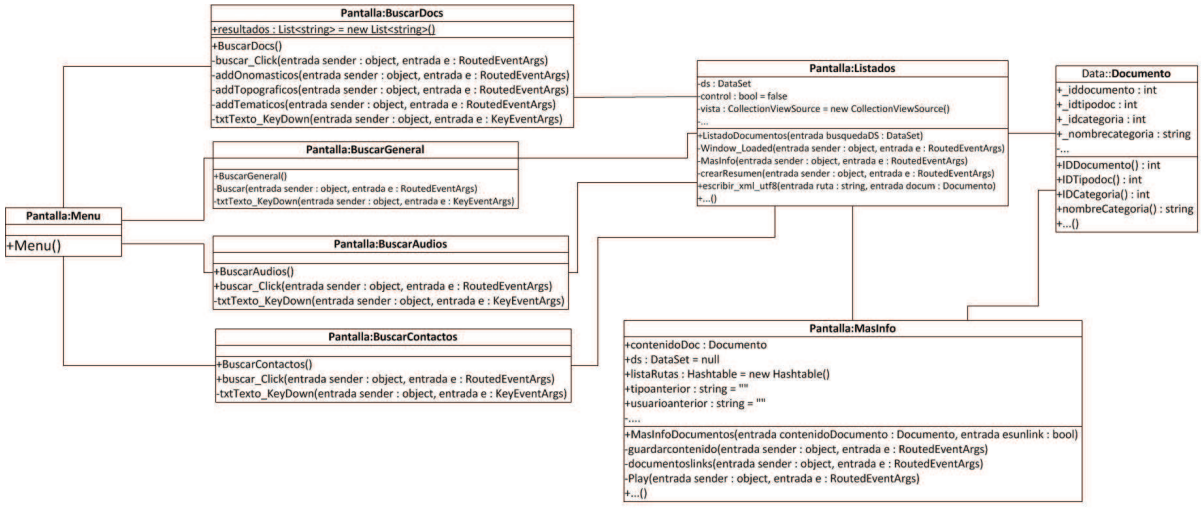

Figura A.17: Diagrama de clases y objetos para las opciones de búsqueda

- **Opción crear nuevo contenido** En la Figura A.18 se puede observar el diagrama de clases para la opción de crear nuevo contenido.

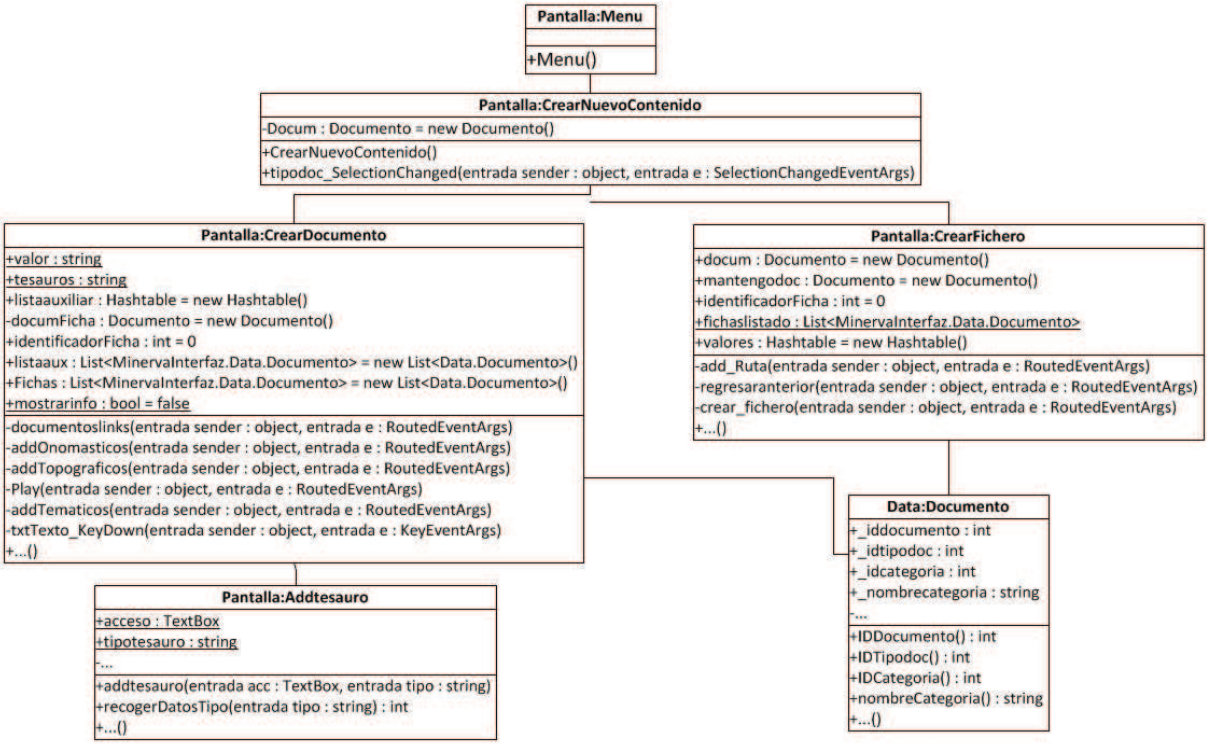

Figura A.18: Diagrama de clases y objetos para la opción *crear contenido*

- **Opción gestionar tesauro** En la Figura A.19 se puede observar el diagrama de clases para la opción de gestionar el tesauro de la herramienta.

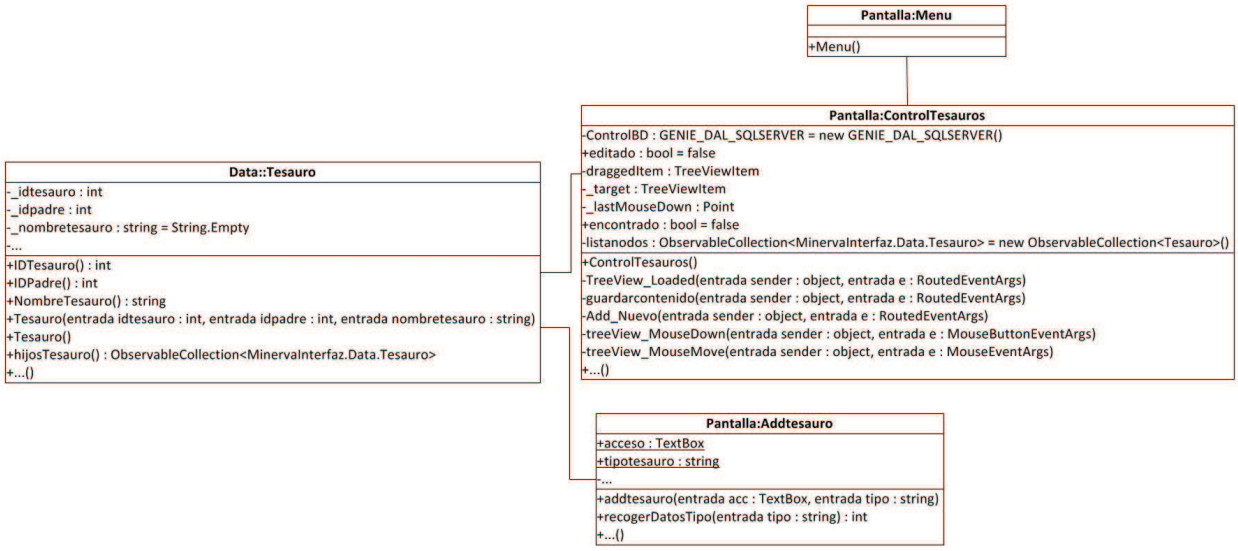

Figura A.19: Diagrama de clases y objetos para la opción *gestión de tesauros*

- **Opción configurar** En la Figura A.20 se puede observar el diagrama de clases para la opción de configuración de la herramienta.

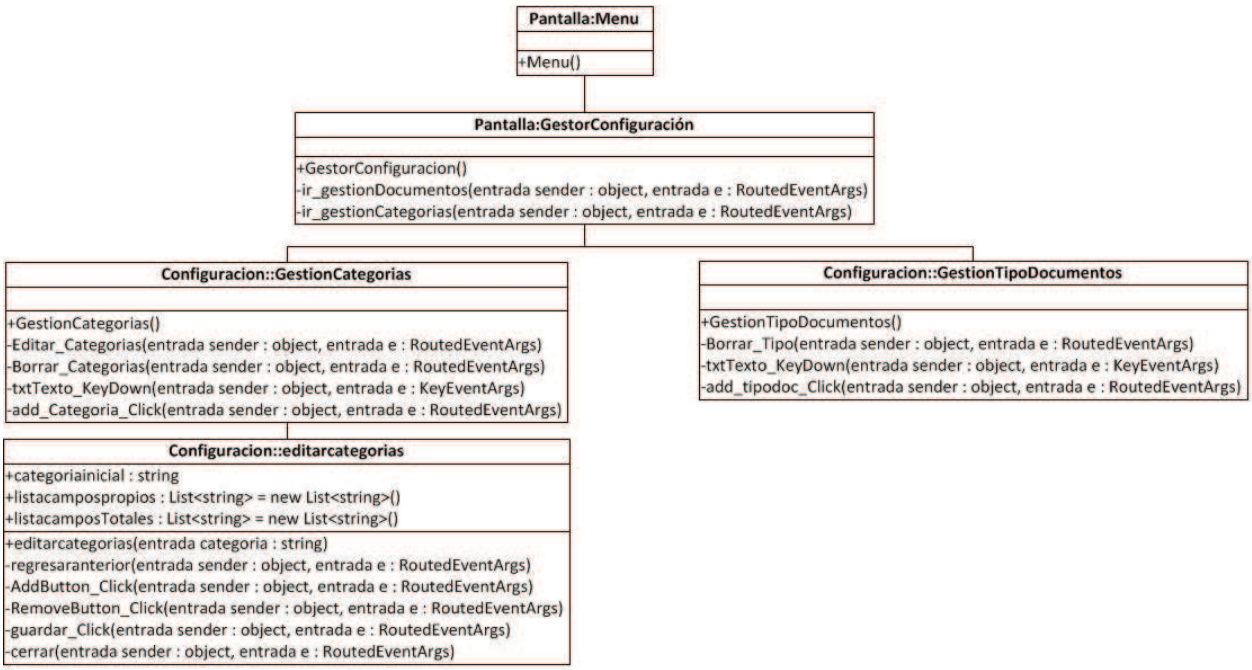

Figura A.20: Diagrama de clases para la opción *configurar*

- **Otras opciones** En la Figura A.21 se puede observar el diagrama de clases para el resto de opciones de la herramienta.

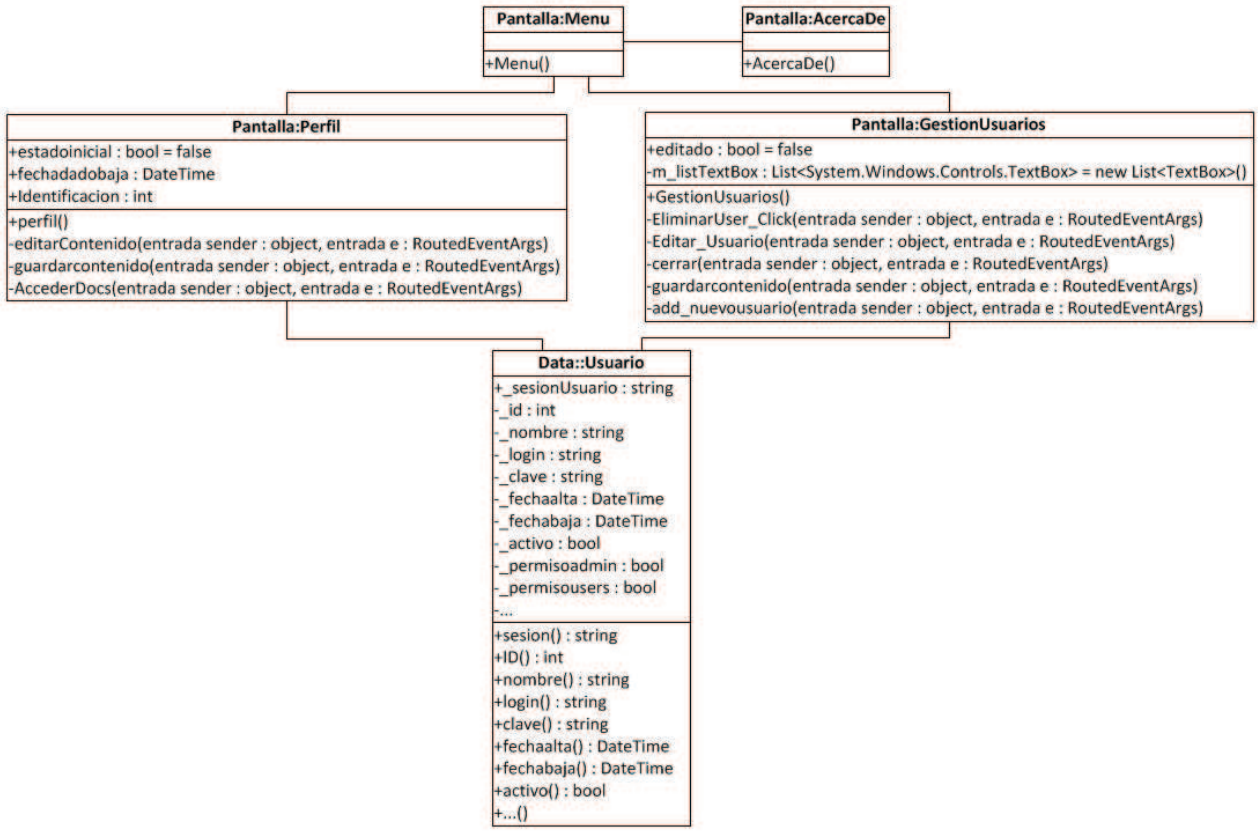

Figura A.21: Diagrama de clases y objetos para el resto de opciones del sistema

A.3.4. Navegación

La navegación del sistema puede tomar diversos caminos según la opción seleccionada por el usuario. Por ello, para aclarar la navegación del sistema se ha añadido la Figura A.22, en la cual se puede observar un diagrama general de la navegación del sistema.

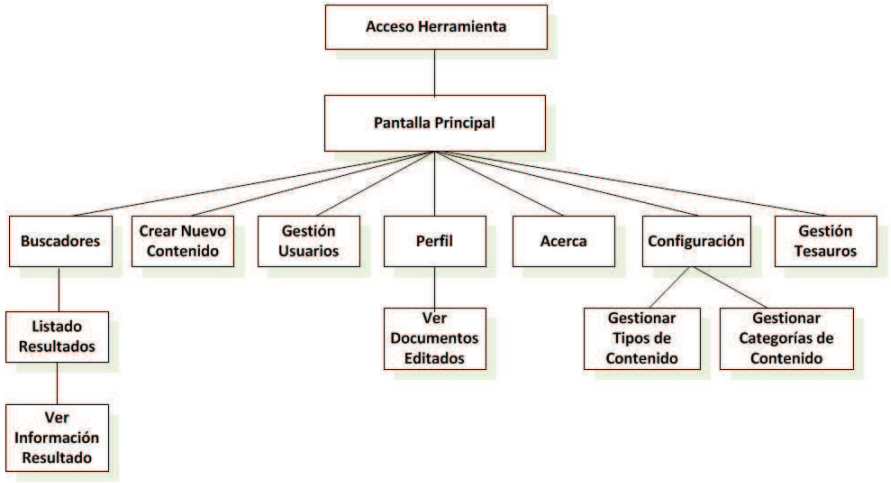

Figura A.22: Diagrama de navegación del sistema

A.4. Base de datos

A.4.1. Migración de la base de datos

Se ha visto necesario diseñar una nueva base de datos, y por consiguiente migrar los datos de la antigua base de datos a la nueva. El motivo por el cual se decidió diseñar una nueva base de datos se debe a que la antigua base de datos almacenaba redundancias, ambigüedades, tablas vacías, etc., lo que dificultaba el uso del sistema. El hecho de que la base de datos tuviera un conjunto de datos duplicados es debido a que el sistema actual de Aragón Radio se vio modificado en el año 2010 por una empresa, la cual fusionó la estructura que el sistema tenía anteriormente con la nueva que desarrollaron, lo que ha provocado un gran número de problemas que influyen a toda la aplicación y dificultan enormemente el uso de la misma. En la nueva base de datos se han eliminado todos los datos duplicados, se han unificado los contenidos, el diseño es más sencillo y permite acceder a los campos deseados de manera más rápida y sencilla, lo cual mejora las consultas a la base de datos. Además, se ha mejorado notablemente el tesauro que se tenía anteriormente, permitiendo un tesauro facetado agrupado por campos, con una estructura multinivel (sin límite de niveles) y unas relaciones padre-hijo entre términos, dado que el tesauro antiguo no permitía más de dos niveles en su estructura, no estaba facetado y las relaciones entre padre-hijo eran la mayoría incorrectas o estas no existían. El esquema conceptual con la base de datos que tenía Aragón Radio se puede ver en la Figura A.23 y el esquema relacional en la Figura A.24.

El proceso de migración que se ha seguido para adaptar la base de datos actual a la propuesta en esta sección ha consistido en la creación de un programa en Visual Basic .Net que se encarga de crear la nueva base de datos con todas sus tablas, y de insertar todos los datos que se encontraban en la base de datos vieja en la nueva. De esta forma se automatiza la migración, facilitando la misma. Para mejorar la base de datos actual de Aragón Radio, se propone el desarrollo de un esquema Entidad-Relación (E-R) que contenga las entidades que se definen en los puntos siguientes. El diagrama E-R de la Figura A.25 muestra de forma simplificada las entidades y las relaciones existentes entre ellas.

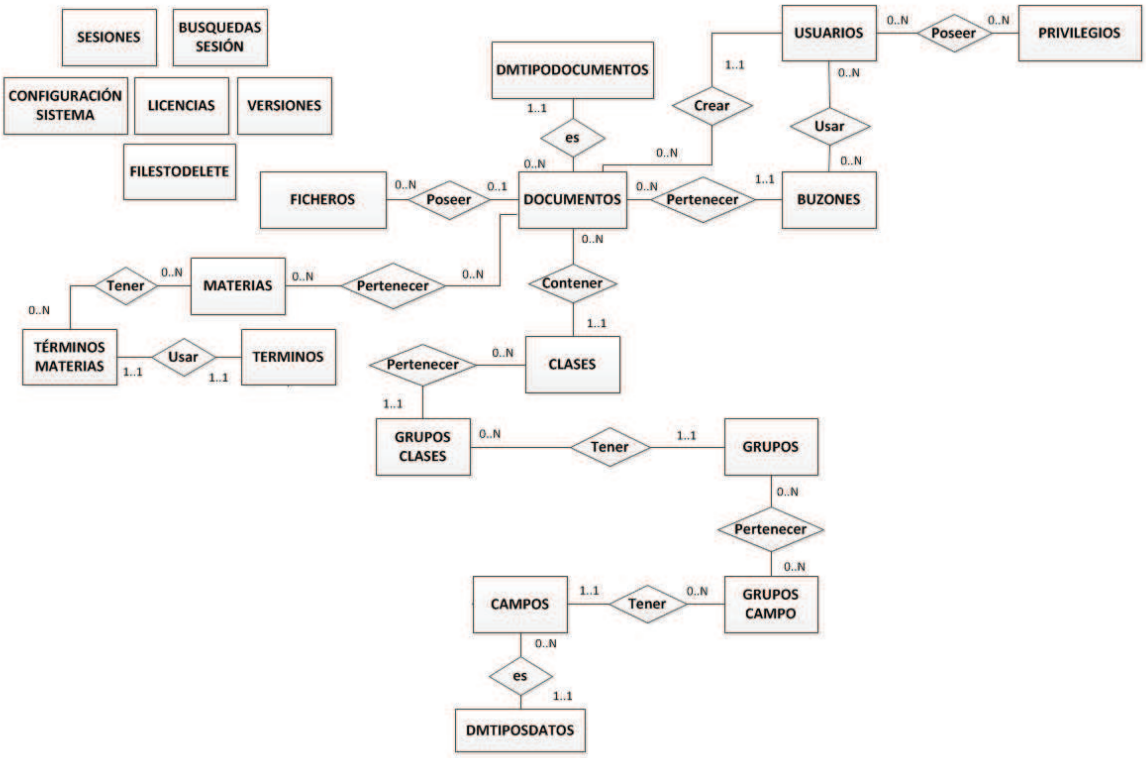

Figura A.23: Esquema conceptual de la base de datos antigua

A.4.2. Nueva base de datos

1. Entidades a desarrollar:

   - USUARIO: entidad que describe la información correspondiente a cada uno de los usuarios que hacen uso del sistema.
   - PRIVILEGIO: contiene la información correspondiente a cada uno de los privilegios de los que dispone cada usuario del sistema. Este conjunto de permisos es fijo y son: Administrador, Gestión de Usuarios, Gestión de Campos, Gestión de Documentos, Lectura de Documentos, Gestión de Tesauro y Cambio de Clave.
   - GRUPO_USUARIO: describe a que grupo de usuarios pertenece cada usuario. Un usuario sólo pertenece a un grupo determinado, y a un grupo pueden pertenecer varios usuarios. A su vez, cada grupo tiene unos privilegios asociados.
   - DOCUMENTO: incluye la información correspondiente a cada uno de los documentos que almacena el departamento de documentación.

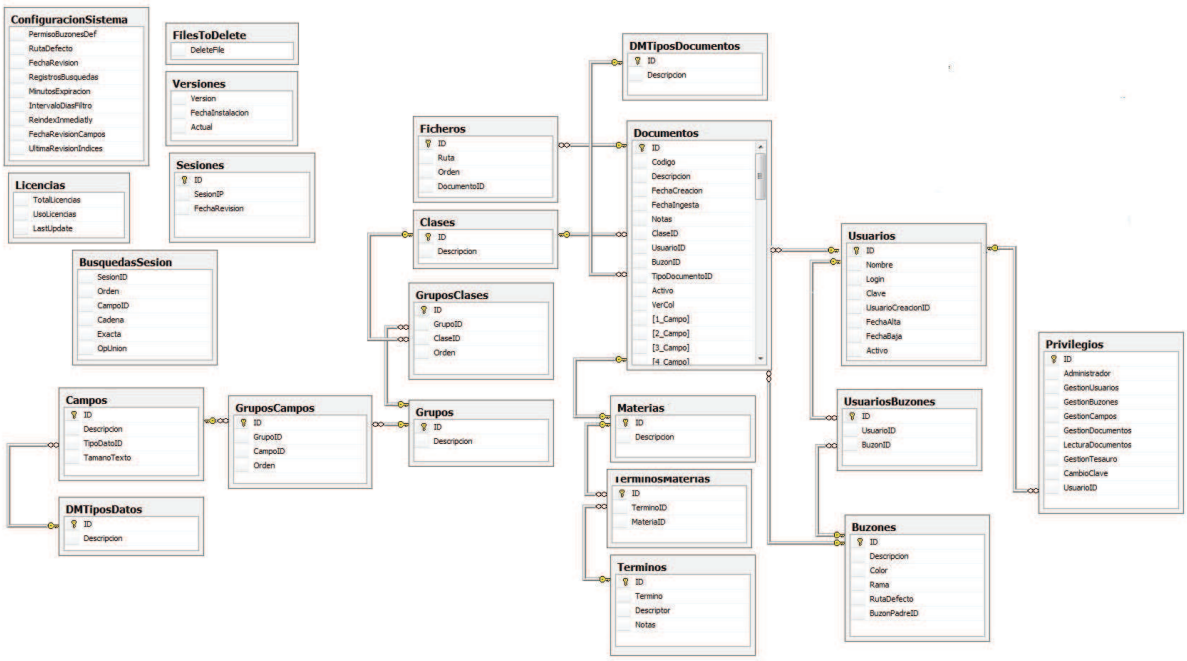

Figura A.24: Esquema relacional de la base de datos antigua

- FICHERO: contiene la información correspondiente a cada uno de los ficheros de audio asociados a cada documento.

- TESAURO: describe la información correspondiente a los términos que clasifican cada documento.

- CATEGORIAS: contiene la información correspondiente a la categoría o ámbito al que pertenece un documento. Por ejemplo, programas, informativos, música, biografías,...

- CAMPOS_CATEGORIA: expresa qué campos pertenecen a cada categoría. Estos campos son: título, programa, cuadro técnico, entre otros.

- TIPO_DOC: incluye la información correspondiente al tipo al que pertenece un documento. Un documento podrá ser de tipo texto, audio, imagen, video, etc.

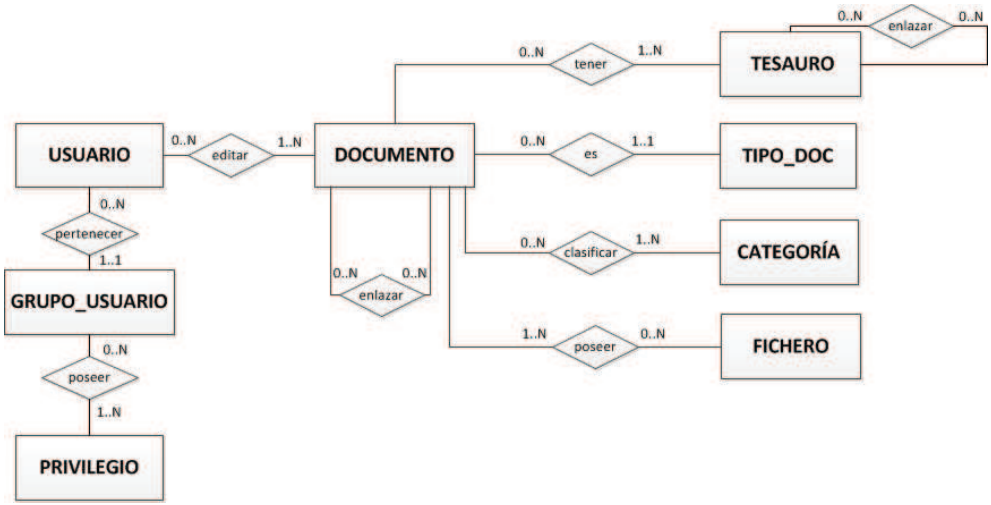

Figura A.25: Esquema conceptual de la base de datos nueva

2. Relaciones existentes:

   - Poseer: indica la relación entre un usuario y un privilegio. Un usuario podrá tener uno o varios privilegios. Estos privilegios corresponden con las diferentes opciones que puede realizar un usuario dentro del sistema.
   - Poseer: describe la relación entre un documento y un fichero. Un documento puede tener asociado uno o varios ficheros de audio.
   - Clasificar: relaciona un documento y una categoría. Un documento pertenecerá a una categoría determinada según el contenido del documento.
   - Es: muestra la relación entre un documento y su tipo (TIPO_DOC). Un documento siempre será de un tipo determinado, estos tipos pueden ser de texto, audio, imagen, texto, etc.
   - Tener: relaciona un documento con una o varias etiquetas del tesauro. Un documento puede tener un conjunto de etiquetas del tesauro vinculadas a este. Estas etiquetas se almacenarán en los campos del documento de nombre: onomásticos, temáticos y topográficos, dependiendo del tipo de etiqueta.
   - Enlazar: detalla la relación que puede existir entre varios documentos.

En este apartado, se describen las tablas que se deben desarrollar para la nueva base de datos y su correspondencia con las entidades y relaciones explicadas en el esquema conceptual del apartado anterior. La Figura A.26 muestra el esquema completo de estas tablas, junto con los atributos definidos para cada una, su

formato, sus restricciones y la relación existente entre ellas. Teniendo en cuenta que la abreviaturas PK y FK corresponden respectivamente a *PRIMARY KEY* y *FOREIGN KEY*.

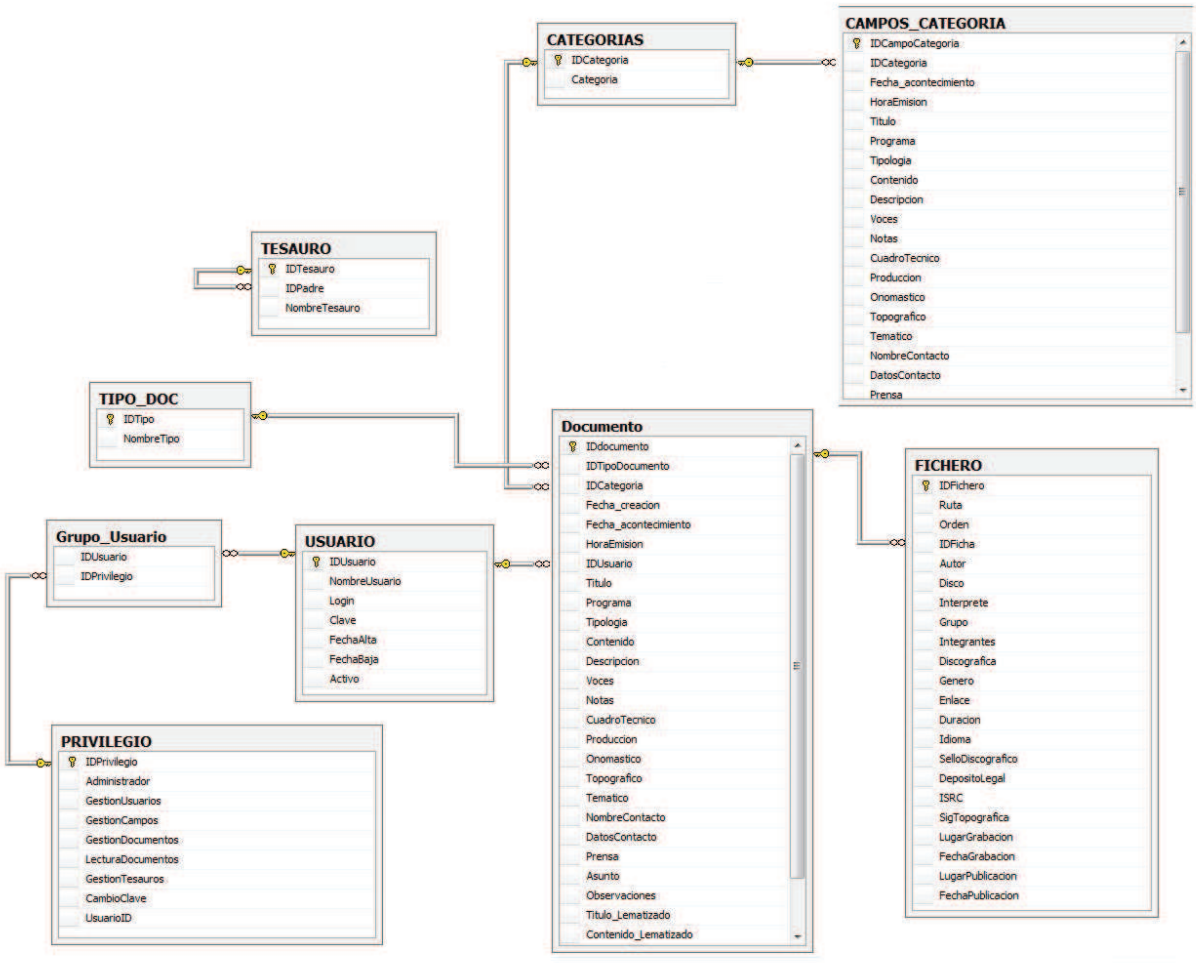

Figura A.26: Modelo lógico de la base de datos nueva

Atributos	Descripción	Formato
IDUsuario	Identificador del Usuario **(PK NO NULL)**.	Integer
NombreUsuario	Nombre del Usuario.	Varchar
Login	Login con el cual un usuario accede a la aplicación. Este login es único **(NO NULL)**.	Varchar
Clave	Clave con la cual un usuario accede a la aplicación y está asociada al login **(NO NULL)**.	Varchar
FechaAlta	Fecha en la cual se ha dado de alta al usuario en la base de datos.	DateTime
FechaBaja	Fecha en la cual se ha dado de baja al usuario en la base de datos.	DateTime
Activo	Indica si el usuario se encuentra activo y disponible en el sistema.	Bit

Tabla A.2: Tabla entidad Usuario

Entidad Documento

Atributos	Descripción	Formato
IDdocumento	Identificador del Documento **(PK NO NULL)**.	Integer
IDTipoDocumento	Identificador del tipo de documento al que pertenece ese documento **(FK de TIPO_DOC NO NULL).**	Integer
IDCategoría	Identificador de la categoría a la que pertenece el documento **(FK de CATEGORIAS NO NULL)**.	Integer
Fecha_creación	Fecha en la que se crea el Documento **(NO NULL)**.	DateTime
Fecha_acontecimiento	Fecha del acontecimiento al que hace referencia el documento.	DateTime
HoraEmisión	Hora a la que se emite un programa.	Varchar
IDUsuario	Identificación del usuario que edito el documento **(FK de USUARIO NO NULL)**.	Integer
Título	Título o nombre del documento.	Varchar

Continuación de la tabla

Atributos	Descripción	Formato
Programa	Programa de emisión de los ficheros que describe el documento.	Varchar
Tipología	Tipo de programa que describe el documento (Programa divulgativo, informativo, etc).	Varchar
Contenido	Contenido del documento.	Varchar
Descripción	Descripción general del contenido del documento.	Varchar
Voces	Nombre de las personas que aparecen en las fichas de audio.	Varchar
Notas	Notas adicionales del documento.	Varchar
Cuadro_técnico	Descripción del cuadro técnico de edición de los documentos.	Varchar
Producción	Producción de las fichas de audio que describe el documento.	Varchar
Onomástico	Contiene los términos del documento que pertenecen a entidades o nombres propios.	Varchar
Topográfico	Contiene los términos del documento que pertenecen a lugares.	Varchar
Temático	Contiene los términos del documento que pertenecen a temas diversos que no son onomásticos ni topográficos.	Varchar
NombreContacto	Nombre del contacto al que hace referencia el documento, en caso de que el documento sea de tipo Contactos.	Varchar
DatosContacto	Número de teléfono, fax o email, del contacto al que hace referencia el documento, en caso de que el documento sea de tipo Contactos.	Varchar
Prensa	Datos de contacto adicionales para el contacto al que hace referencia el documento, por ejemplo el número de teléfono de la secretaría a la cual hay que acceder para llegar a esa persona, etc.	Varchar
Asunto	Motivo por el cual se ha almacenado dicho contacto.	Varchar
Observaciones	Si hay alguna observación adicional a tener en cuenta sobre ese contacto.	Varchar
Título_Lematizado	Titulo del documento lematizado mediante el servicio PLN situado en el servidor.	Varchar

Continuación de la tabla

Atributos	Descripción	Formato
Contenido_Lematizado	Contenido del documento lematizado mediante el servicio PLN situado en el servidor.	Varchar
Descripción_Lematizado	Descripción del documento lematizado mediante el servicio PLN situado en el servidor.	Varchar
Notas_Lematizado	Notas del documento lematizado mediante el servicio PLN situado en el servidor.	Varchar
Observaciones_Lematizado	Observaciones del documento lematizado mediante el servicio PLN situado en el servidor.	Varchar

Tabla A.4: Tabla entidad Documento

Entidad Tipo_Doc

Atributos	Descripción	Formato
IDTipo	Identificador del tipo de documento **(PK NO NULL)**.	Integer
NombreTipo	Nombre del tipo de documento **(NO NULL)**.	Integer

Tabla A.5: Tabla entidad Tipo_Doc

Entidad Fichero

Atributos	Descripción	Formato
IDFichero	Identificador del Fichero **(PK NO NULL)**.	Integer
Ruta	Ruta del Fichero.	Integer
Orden	Orden en el cual se deben escuchar los ficheros de audio en caso de que un documento contenga varios.	Integer
IDFicha	Identificador del documento al cual pertenece dicha ficha de audio **(FK de DOCUMENTO)**.	Integer
Autor	Nombre de la persona que ha editado la ficha de audio.	Varchar

Continuación de la tabla

Atributos	Descripción	Formato
Disco	Nombre del disco al que pertenece la ficha de audio.	Integer
Intérprete	Nombre de intérprete al que pertenece la ficha de audio.	Integer
Grupo	Nombre del grupo al que pertenece la ficha de audio.	Integer
Integrantes	Nombre de los integrantes del grupo al que pertenece la ficha de audio.	Integer
Discográfica	Nombre de la discográfica a la cual pertenece la ficha de audio.	Integer
Género	Género musical al que pertenece el audio de la ficha.	Integer
Enlace	Enlace auxiliar a la información del grupo, por ejemplo, la URL de la página del grupo, etc.	Integer
Duración	Horas, minutos y segundos que dura la ficha de audio.	Integer
Idioma	Idioma al que pertenece la ficha de audio.	Integer
SelloDiscográfico	Sello discográfico al que pertenece en caso de que el audio sea una ficha musical.	Integer
DepositoLegal	Depósito Legal del audio musical.	Integer
ISRC	ISRC o Código Estándar Internacional de Grabación al que pertenece la ficha musical.	Integer
SigTopografica	Signatura Topográfica en la cual se almacena la ficha musical en caso de que Aragón Radio disponga de ese disco en su biblioteca de discos musicales.	Integer
LugarGrabación	Lugar donde se realizó la grabación de la ficha de audio.	Integer
FechaGrabacion	Fecha en la que se grabó la ficha de audio.	Integer
LugarPublicacion	Lugar donde se publicó la ficha de audio.	Integer
FechaPublicacion	Fecha en la que se publicó la ficha de audio.	Integer

Tabla A.6: Tabla entidad Fichero

Entidad Privilegio

Atributos	Descripción	Formato
IDPrivilegio	Identificador del Privilegio **(PK NO NULL)**.	Integer
UsuarioID	Identificador del usuario al que está asociado el privilegio. **(FK de USUARIO NO NULL)**.	Integer
Administrador	Permiso para gestionar todas las opciones de la aplicación **(NO NULL)**.	Bit
GestiónUsuarios	Permiso de gestión de usuarios de la aplicación **(NO NULL)**.	Bit
GestiónCampos	Permiso de gestión de campos para una categoría determinada **(NO NULL)**.	Bit
GestiónDocumentos	Permiso de gestión y edición de documentos de la aplicación **(NO NULL)**.	Bit
Lecturadocumentos	Permiso de lectura de documentos de la aplicación **(NO NULL)**.	Bit
GestiónTesauro	Permiso de gestión de tesauros de la aplicación **(NO NULL)**.	Bit
CambioClave	Permiso de cambio de clave a los usuarios **(NO NULL)**.	Bit

Tabla A.3: Tabla entidad Privilegio

Entidad Categorías

Atributos	Descripción	Formato
IDCategoría	Identificador de la Categoría **(PK NO NULL)**.	Integer
Categoría	Nombre de la Categoría **(NO NULL)**.	Integer

Tabla A.7: Tabla entidad Categorías

Campos Categoría

Atributos	Descripción	Formato
IDCampoCategoría	Identificador del campo **(PK NO NULL)**.	Integer

Continuación de la tabla

Atributos	Descripción	Formato
IDCategoría	Identificador de la categoría a la que pertenecen los campos **(FK de CATEGORIAS NO NULL)**.	Integer
Fecha_acontecimiento	Indica si esta categoría tiene activo el campo fecha de acontecimiento en los formularios de crear y editar contenido **(NO NULL)**.	Bit
HoraEmision	Indica si esta categoría tiene activo el campo hora de emisión en los formularios de crear y editar contenido **(NO NULL)**.	Bit
Titulo	Indica si esta categoría tiene activo el campo título en los formularios de crear y editar contenido **(NO NULL)**.	Bit
Programa	Indica si esta categoría tiene activo el campo programa en los formularios de crear y editar contenido **(NO NULL)**.	Bit
Tipología	Indica si esta categoría tiene activo el campo tipología en los formularios de crear y editar contenido **(NO NULL)**.	Bit
Contenido	Indica si esta categoría tiene activo el campo contenido en los formularios de crear y editar contenido **(NO NULL)**.	Bit
Descripción	Indica si esta categoría tiene activo el campo descripción en los formularios de crear y editar contenido **(NO NULL)**.	Bit
Voces	Indica si esta categoría tiene activo el campo voces en los formularios de crear y editar contenido **(NO NULL)**..	Bit
Notas	Indica si esta categoría tiene activo el campo notas en los formularios de crear y editar contenido.	Bit
Cuadro Técnico	Indica si esta categoría tiene activo el campo cuadro técnico en los formularios de crear y editar contenido **(NO NULL)**.	Bit
Producción	Indica si esta categoría tiene activo el campo producción en los formularios de crear y editar contenido **(NO NULL)**.	Bit

Continuación de la tabla

Atributos	Descripción	Formato
Onomástico	Indica si esta categoría tiene activo el campo onomástico en los formularios de crear y editar contenido **(NO NULL)**.	Bit
Topográfico	Indica si esta categoría tiene activo el campo topográfico en los formularios de crear y editar contenido **(NO NULL)**.	Bit
Temático	Indica si esta categoría tiene activo el campo temático en los formularios de crear y editar contenido **(NO NULL)**.	Bit
NombreContacto	Indica si esta categoría tiene activo el campo nombrecontacto en los formularios de crear y editar contenido **(NO NULL)**.	Bit
DatosContacto	Indica si esta categoría tiene activo el campo datoscontacto en los formularios de crear y editar contenido **(NO NULL)**.	Bit
Prensa	Indica si esta categoría tiene activo el campo prensa en los formularios de crear y editar contenido **(NO NULL)**.	Bit
Asuntos	Indica si esta categoría tiene activo el campo asunto en los formularios de crear y editar contenido **(NO NULL)**.	Bit
Observaciones	Indica si esta categoría tiene activo el campo observaciones en los formularios de crear y editar contenido **(NO NULL)**.	Bit

Tabla A.8: Tabla entidad CamposCategoría

Entidad Tesauro

Atributos	Descripción	Formato
IDTesauro	Identificador del Tesauro **(PK NO NULL)**.	Integer
IDPadre	Identificador del Tesauro padre **(NO NULL)**.	Integer
NombreTesauro	Nombre del Tesauro.	Varchar

Tabla A.9: Tabla entidad Tesauro

Apéndice B

Unidad de Resúmenes Automáticos

Vivimos en una sociedad en la que la información y el conocimiento son fundamentales. Por ello, el acceso a grandes cantidades de información se ha convertido en algo habitual en nuestras vidas, planteándose la necesidad de disponer de herramientas para su obtención, organización, análisis y distribución. Una herramienta útil para satisfacer todas esas necesidades sería un generador automático de resúmenes. Esta herramienta podría tener numerosos ejemplos de uso, por ejemplo:

- Para resumir noticias de SMS (Short Message Service) para teléfonos móviles o tabletas.
- En los motores de búsqueda para presentar descripciones comprimidas de los resultados de búsqueda.
- Para buscar en otros lenguajes y poder obtener un resumen traducido automáticamente.
- Resumir la información que aparece en algunos blogs sobre los temas principales de los que se habla, con enlaces a las entradas de interés.
- Recesiones bibliográficas.
- Resúmenes de historiales médicos, de correo electrónico, de páginas web, de noticias, extracción de titulares (headlines), apoyo a los sistemas de recuperación de información y resúmenes de reuniones.

El esquema básico que debería tenerse para generar un resumen automático sería el que se muestra en la Figura B.1. Es decir, primero se debe introducir un conjunto de documentos, un documento o una consulta a sintetizar, y según las condiciones configuradas (por ejemplo, el grado de compresión del resumen), se procedería a localizar los fragmentos más relevantes (segmentos, oraciones, párrafos,...). Después se interpretaría tanto a nivel conceptual (conocimiento semántico)

como a nivel estructural (conocimiento sintáctico), y finalmente se procedería a ordenar los fragmentos extraídos por orden de relevancia, para producir el resumen.

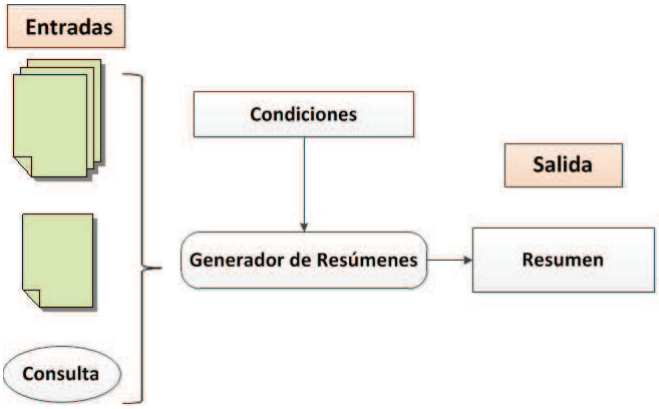

Figura B.1: Esquema general de la generación de resúmenes

B.1. Factores a tener en cuenta

Desde 1998 [11] ya se estableció un marco para el estudio de los factores que intervienen a la hora de generar un resumen. No obstante, en 2007 [25] se volvieron a revisar estos factores. A día de hoy muchos prototipos y sistemas no tienen en cuenta todos estos factores. Estos factores son los siguientes: fuente, idioma, género, dominio (debe tener conocimiento y técnicas de tratamiento del lenguaje natural adaptadas y específicas para el dominio que trata), contenido a extraer, formato de entrada, formato de salida, grado de compresión (este puede variar del 0 % al 100 %), por lo que un resumen con grado de compresión del 1 % sería el que deseche el 99 % del volumen total de la fuente), genericidad y tipo de resumen (informativo, descriptivo, abstracto o de síntesis).

B.2. Estado del arte

Podemos encontrar diversas metodologías para conseguir la información más importante dentro de un texto dado. Todas tienen como objetivo seleccionar las palabras clave (keywords), palabras individuales o frases para "etiquetar" un documento, y posteriormente usarlo para seleccionar frases enteras y crear con ellas un resumen breve. Dentro de esta metodología se pueden encontrar las siguientes estrategias:

1. **Aproximación estadística clásica:** se enfoca en fragmentar el contenido en palabras y oraciones, a las que se les aplica una medida de relevancia relativa, o peso, para la que se tiene en cuenta la frecuencia relativa y aspectos como la localización de los elementos. Un ejemplo de fórmula de puntuación podría ser la ecuación B.2:

$$S = w1 \times C + w2 \times K + w3 \times T + w4 \times L \tag{B.1}$$

   Donde S sería la puntuación total para una oración, teniendo en cuenta que C es el número de palabras pragmáticas (aquellas expresiones que proporcionan información a nivel del discurso, por ejemplo: además, entonces, ahora o por lo tanto) que acentúan la importancia de una sección, K es las palabras clave (estadísticamente seleccionadas del texto), T son las palabras que componen el título, y L corresponde a la localización de la frase. Por último w1, w2, w3 y w4 son los pesos predispuestos para cada categoría.

2. **Aproximación aplicando aprendizaje de máquina:** se trata de buscar patrones dentro de la representación matemática del texto, que permitan encontrar las frases más relevantes del mismo. Se puede llevar a cabo de dos maneras:

   a) Enfoque supervisado: se entrena un modelo sobre la representación vectorial de resúmenes realizados por personas. Posteriormente se plantea un modelo Naive-Bayes que decide que frase es relevante en el texto y cual no, calculando cuál de los dos posibles eventos es más probable dado un conjunto de características de las oraciones, por ejemplo su tamaño.

   b) Enfoque no supervisado: se trabaja directamente sobre una representación del texto en busca de relaciones subyacentes, que permitan seleccionar las oraciones. Para ello se aplican técnicas de clustering o agrupamiento utilizando el algoritmo K-Mean. En este caso se consideran representaciones vectoriales de cada oración del texto, se realiza agrupamiento de las frases y finalmente se aplica una medida de similitud. Para generar el resumen, se definen un numero k de clústers o grupos y se halla el centroide de cada uno de ellos, es decir la oración que se considera más representativa dentro de cada clúster. Al final los k centroides correspondientes a k oraciones conforman el resumen.

Después de todas las metodologías encontradas, se ha decidido utilizar una metodología de enfoque no supervisado utilizando herramientas estadísticas y herramientas de procesamiento de lenguaje natural (PLN). El motivo de esta elección, es por la multitud de ventajas que ofrece esta técnica:

- Implementación sencilla.

- Bajo coste en esfuerzo humano, económico y computacional [17].

- Consistente y evita la subjetividad de los redactores de resúmenes humanos [14].

- Suelen dar mejores resultados que la técnica abstractiva [16]. Esta técnica consiste en construir una representación semántica interna y luego utilizar técnicas naturales de generación del lenguaje para crear un resumen que está más cerca de lo que un humano puede generar. Este resumen puede contener palabras no presentes explícitamente en el texto original. Es necesario tener en cuenta el tipo de palabras que se evalúan y sus relaciones, por ello se necesitan herramientas de procesamiento del lenguaje natural (PLN) (tesauros, analizadores gramaticales y técnicas de representación del conocimiento (ontologías) de dominio específico).

Como inconvenientes se pueden encontrar los siguientes [16]:

- Falta de coherencia del resumen generado.

- Redundancia del contenido por tratar las oraciones de manera independiente.

Para solucionar el primer inconveniente se ha utilizado un analizador sintáctico, y para solucionar el segundo inconveniente se dispone de un simplificador de oraciones redundantes.

B.3. Arquitectura

Para poder generar resúmenes de forma automática he desarrollado una unidad de generación de resúmenes llamada GENIE_SUM, que posee una arquitectura con los siguientes componentes: herramientas estadísticas y herramientas de procesamiento de lenguaje natural (PLN). En la Figura B.2 puede verse el diagrama que representa las etapas que he implementado para poder generar el resumen.

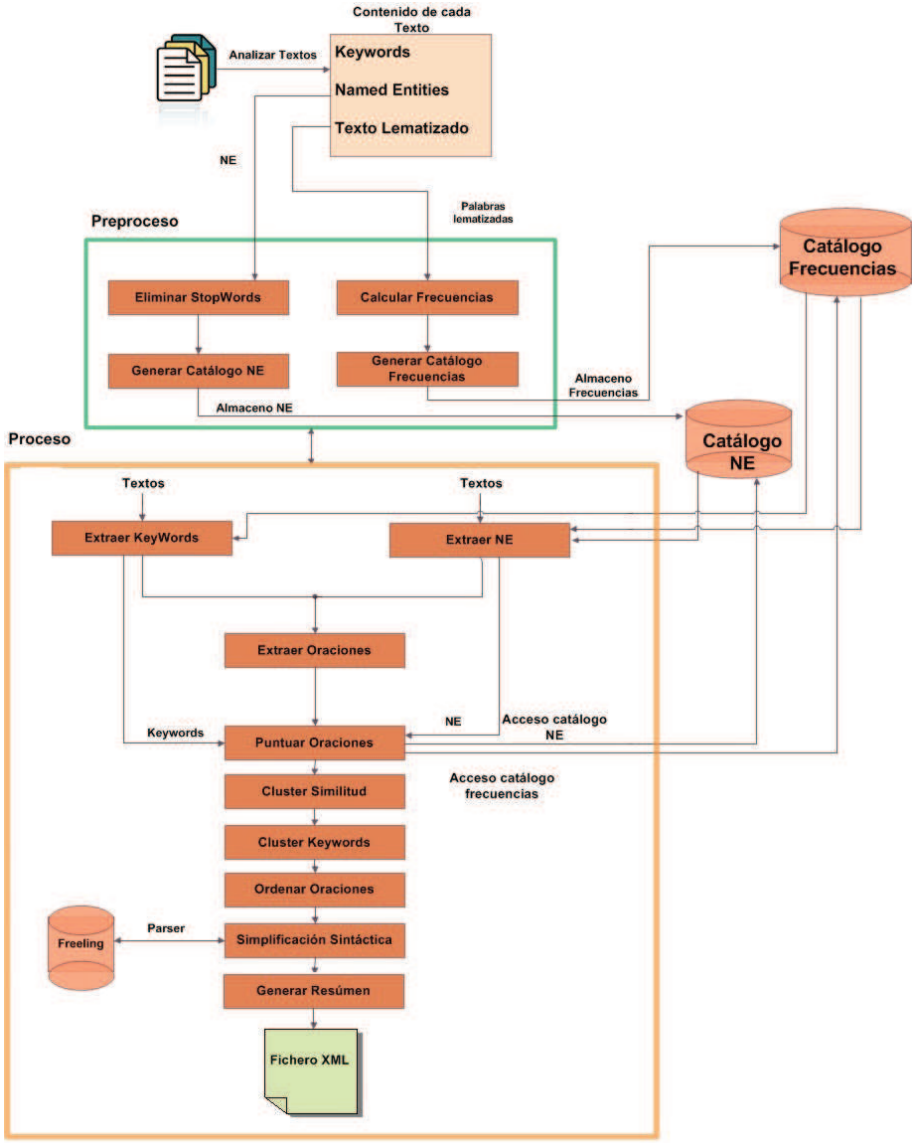

Figura B.2: Diagrama etapas del generador de resúmenes

B.3.1. Entradas y salidas del sistema

Como entradas del sistema podemos encontrar:

- **Fichero de Configuración:** fichero con toda la información que necesita el sistema para realizar su función (ruta donde recoger el texto a resumir, ruta donde dejar el resumen generado y grado de compresión deseada.
- **Textos:** documentos de los que se quiere obtener un resumen.

Y como salidas las siguientes:

- **Resumen:** fichero que contiene el resumen de los textos introducidos en la entrada del sistema (puede realizarse de un fichero o de varios).
- **Índice Temático:** se genera un vocabulario controlado que sirva para catalogar documentos posteriores. Este índice es útil para conocer que palabras son más utilizadas dentro de un conjunto de textos.

La unidad desarrollada se compone de dos etapas diferentes: una de preprocesamiento y otra de proceso, las cuales se van a explicar detalladamente en los puntos siguientes.

B.3.2. Etapa de preproceso

En esta etapa se realiza un análisis de los textos dados, obteniendo la información relativa a sus palabras (esto se requerirá en tareas posteriores). Para hacer esta tarea, se usa una herramienta PLN (Procesamiento del Lenguaje Natural) para extraer y lematizar las palabras que constituyen los textos e identificar las Named Entities presentes en ellos. Esta herramienta es Freeling [5].

La información extraída se almacena en dos catálogos:

- Una catálogo de Named Entities (NE) [21] identificadas mediante una herramienta de PLN. Cada entidad puede corresponder a nombres de personas, lugares, organizaciones, etc. Antes de almacenar estos, el preproceso ejecuta un análisis de las NE para eliminar aquellas que pueden haber sido identificadas como tal y no lo son (por ejemplo: artículos, preposiciones, etc.), es decir, se eliminan las palabras conocidas como *stop words*. Además a cada NE se le asigna un peso para indicar cuánto es de relevante en el texto dado.
- Una catálogo de frecuencias de las palabras que aparecen en los textos. El proceso usa un método que permite calcular estadísticas (frecuencias) de las palabras extraídas con PLN.

B.3.3. Etapa de proceso

En este apartado se describe la etapa de proceso en detalle.

1. **Extracción de keywords:** se encarga de extraer una lista de keywords para cada texto dado, usando el método TF-IDF (Term Frequency - Inverse Document Frequency). Más tarde esas palabras permiten identificar qué partes del texto proporcionan más conocimiento que otras. En esta fase se usan las frecuencias de las palabras extraídas en la etapa de preprocesamiento.

El algoritmo estadístico tf-idf[20], el cual es muy conocido en ámbitos de recuperación de información y minería de textos, permite obtener una medida numérica que expresa cuán relevante es una palabra para un documento dentro de una colección. tf indica la frecuencia de un término e idf la frecuencia inversa de documento, es decir, indica si un término es común o no en la colección de documentos, como por ejemplo la palabra "Aragón" dentro de la base de datos de Aragón Radio. El valor tf-idf aumenta proporcionalmente al número de veces que una palabra aparece en el documento, pero es compensada por la frecuencia de la palabra en la colección de documentos dados, lo que permite manejar el hecho de que algunas palabras son generalmente más comunes que otras. La ecuación que representa el cálculo tf-idf puede verse en la ecuación B.2, siendo t el término, d el documento y D el número de documentos en la colección.

$$tfidf(t, d, D) = tf(t, d) \times idf(t, D) \tag{B.2}$$

2. **Extracción de Named Entities:** se extrae una lista de NE para cada uno de los textos dados usando la herramienta PLN *Freeling*[5]. Freeling es una biblioteca de código abierto desarrollado en la Universidad Politécnica de Cataluña que permite analizar lingüísticamente diversos idiomas. Ofrece funciones de análisis y anotación lingüística de textos. También se tienen en cuenta las frecuencias de las Named Entities extraídas en la etapa de preprocesamiento.

3. **Extraer oraciones:** se encarga de dividir el texto en oraciones para identificar posteriormente cuáles tienen información más relevante. Se considera oración todo aquel conjunto de palabras que termina en un punto.

4. **Puntuar oraciones:** Se puntúa cada oración para diferenciar las oraciones relevantes de las que no lo son. El método propuesto para poder determinar las oraciones relevantes y las que no lo son consiste en buscar en una oración las palabras que coinciden con las *keywords* y las *named entities* que aparecen en todo el texto. El procedimiento de calificar las oraciones permite extraer la oración más relevante dentro de la colección general de oraciones obtenidas para cada texto de entrada dado. Esa calificación es la suma de la puntuación `tf-idf` (que se calculará teniendo en cuenta el lema de la palabra) de cada palabra que forma parte de la oración, dividido por el número de palabras totales que componen la oración (*totPalabras*). Esta métrica indica que la relevancia de una oración depende no sólo de la frecuencia de las palabras

presentes en ella, sino también del número de textos en los que aparecen las palabras. La Ecuación B.3 describe el cálculo de la puntuación.

$$Score_O = \frac{\sum_P (tfidf_w)}{totPalabras} \tag{B.3}$$

Las siguientes fases del proceso tienen como objetivo identificar la información relevante del conjunto de oraciones en dos etapas basándome en el algoritmo [24]:

- **Clúster de similitud:** para identificar oraciones que transmiten la misma información, los clústers del proceso consideran el grado de similitud de cada oración. La similitud entre dos oraciones comprende dos dimensiones calculadas teniendo en cuenta los lemas de las palabras: las subsecuencias de las oraciones (haciendo uso del algoritmo ROUGE-L, el cual busca la subsecuencia más común en una oración) y el solapamiento de palabras (con el algoritmo Jaccard, busca las palabras comunes compartidas en dos oraciones utilizando los lemas de las palabras).

  Estas dos dimensiones se combinan en un valor de similitud. Este valor de similitud se calcula para todas las oraciones de las que disponemos, y se guarda en un array de similitud. El array de similitud se ordena según el grado de similitud y finalmente se procede a recuperar la primera oración dentro del array, es decir, la que tiene mayor grado de similitud. Si el valor de similitud es mayor que un umbral de similitud dado, las oraciones se agrupan en el mismo grupo (mismo clúster); si no, la oración es añadida a un nuevo grupo (clúster). La oración con la puntuación más alta de cada grupo generado se selecciona para utilizarse en la siguiente etapa, estas oración se denominan centroides. Las oraciones que no han sido agrupadas se guardan para posteriores comprobaciones. Un diagrama que refleja el funcionamiento de este clúster se muestra en la Figura B.3, siendo "ClusSim" un clúster de similitud.

- **Clúster de Keywords:** el algoritmo que agrupa las oraciones por *keywords* es una versión adaptada del algoritmo K-means ampliamente utilizado en minería de datos, que permite agrupar oraciones según el grupo al que pertenezcan.

  El agrupamiento de *keywords* agrupa las oraciones basadas en las apariciones de las keywords. En esta etapa se utilizan las oraciones que se han considerado como centroides en la etapa anterior. Para cada centroide se extraen sus respectivas keywords haciendo uso de la unidad de

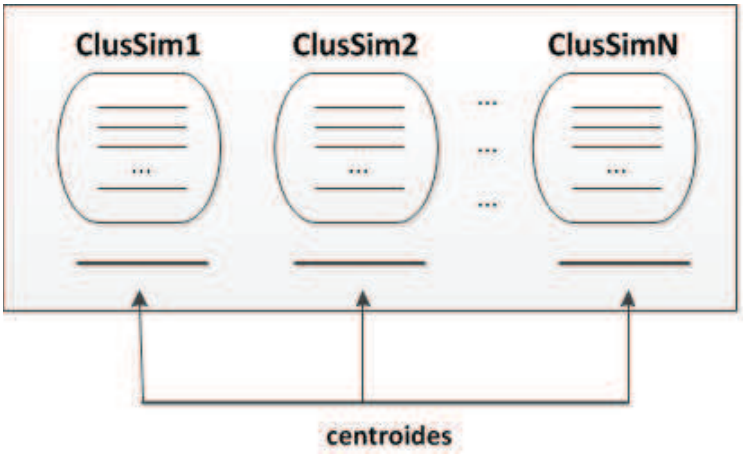

Figura B.3: Diagrama clúster de similitud

extracción de las mismas, y con aquellas keywords cuya frecuencia es mayor frente a otras, se genera un nuevo clúster, que se denomina clúster de *keywords*. Ahora se procede a añadir las oraciones a los clústers de *keywords* generados, teniendo en cuenta para ello que es añadida a un clúster o no si esa oración tiene como *keyword* más frecuente, aquella *keywords* que representa al clúster. Las oraciones que no contienen ninguna *keyword* se ignoran, al considerarse irrelevantes. Un diagrama que refleja el funcionamiento de este clúster se muestra en la Figura B.4, siendo "Clusk" un clúster de keywords.

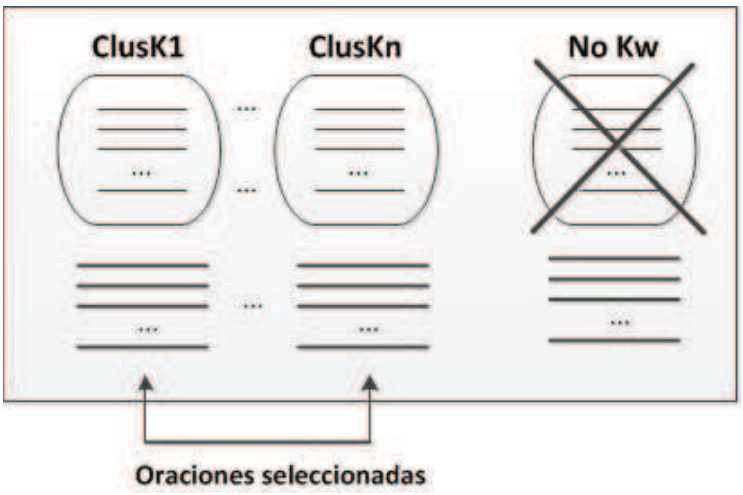

Figura B.4: Diagrama clúster de keywords

5. **Ordenar oraciones:** una vez que todas las oraciones han sido puntuadas se procede a ordenarlas. En este punto, el conjunto de oraciones que se encuentran agrupadas en los clústers de keywords poseen una determinada puntuación, la cual ha sido calculada previamente. Según esa puntuación se procede a ordenar las oraciones de mayor puntuación a menos, constituyendo esta ordenación un resumen del texto dado.

6. **Simplificación sintáctica:** este proceso es el encargado de simplificar cada una de las oraciones que forman el resumen, es decir eliminando contenido irrelevante. Para conseguir esta simplificación, se ha hecho uso de la biblioteca GENIE_TSIMPLY.dll creada dentro del Grupo SID de la Universidad de Zaragoza. Esta biblioteca utiliza un módulo de procesamiento de lenguaje natural para efectuar la simplificación. La simplificación se realiza según una serie de reglas impuestas en un fichero XML; una regla podría ser, por ejemplo, eliminar aquel contenido que se encuentre entre comas.

7. **Obtención del resumen:** el resumen generado se devuelve en formato XML en el directorio de salida elegido en el fichero de configuración de la unidad de resumen.

Esta unidad ha sido probada en diversas aplicaciones para comprobar su correcto funcionamiento como por ejemplo, en un generador de conocimiento a través de topic maps[1] obtenidos gracias a un resumen previo del contenido seleccionado por el usuario [7], y en una aplicación[6] que permite lanzar desde la misma una consulta a varios motores de búsqueda seleccionados por el usuario, y obtener un resumen para cada uno de ellos teniendo en cuenta además de las NE y las keywords, los synsets del texto[2] y el contenido HTML[3] de las páginas web encontradas.

[1]Topic Map: estándar para la representación y el intercambio de conocimiento mediante un mapa conceptual que plasme las ideas más importantes.

[2]Synsets: conjunto de sinónimos de una palabra dada.

[3]HTML: (**H**yperText **M**arkup **L**anguage) lenguaje de marcado para la elaboración de páginas web.

Apéndice C

Pruebas

Una vez que se han concluido las fases de implementación e implantación, se han realizado todas las pruebas del sistema para verificar el correcto funcionamiento de la aplicación y comprobar así que todos los requisitos expuestos con anterioridad se cumplen.

En este documento se da una descripción detallada de las distintas pruebas realizadas sobre cada una de las funcionalidades que posee la aplicación, comprobando así el correcto funcionamiento de la misma.

En la primera sección se van a describir las pruebas unitarias utilizadas, las cuales se encargan de verificar el correcto funcionamiento de un módulo de código por sí sólo, y en la segunda y la tercera se encarga de verificar el correcto funcionamiento del sistema.

C.1. Pruebas unitarias a realizar.

- **Búsqueda**: como la aplicación contiene varios formularios de búsqueda para facilitar al usuario la localización de información, estos formularios han de cumplir una serie de requisitos mínimos que se detallan a continuación:

  Sistema de Validación

  1. Comprobar que no se producen errores por desbordamiento en los campos. Estado: correcto.
  2. Validación de campos obligatorios: comprobar que el sistema de validación obliga a informar sobre todos los campos de carácter obligatorio. Estado: correcto.
  3. Validación de formato numérico: comprobar que todos los campos de tipo numérico en caso de ser informados son validados como tal (deci-

males, enteros). Estado: correcto.

4. Validación de campos tipo fecha: comprobar la correcta validación de los campos fecha en caso de ser informados. Estado: correcto.
5. Comprobar que el buscador no distinga entre mayúsculas y minúsculas. Estado: correcto.
6. Comprobar que al introducir comillas no se rompe la cadena de búsqueda, y permite buscar de forma exacta. Estado: correcto.

Otras consideraciones:

1. Comprobar que funcionan correctamente todos los criterios de búsqueda informados. Estado: correcto.
2. En caso de que la búsqueda se complemente con una ordenación: realizar una búsqueda utilizando al menos un criterio de búsqueda y ordenar el listado por un campo (fecha de creación del contenido) comprobando que no se pierda el criterio de búsqueda durante el proceso de ordenación. Estado: correcto.
3. Realizar una búsqueda utilizando un criterio de búsqueda y recorrer las distintas páginas del listado comprobando que no se pierda el criterio de búsqueda durante el proceso de paginación. Estado: correcto.

- **Inserción:** la pantalla de creación de nuevo contenido así como varias pantallas con opciones de insertar datos, permiten registrar nuevos datos en la aplicación completando los campos de un formulario de entrada de datos.

  Sistema de Validación

  1. Comprobar que no se producen errores por desbordamiento en los campos. Estado: correcto.
  2. Validación de campos obligatorios: comprobar que el sistema de validación obliga a rellenar todos los campos del formulario de carácter obligatorio. Estado: correcto.
  3. Validación de formato numérico: comprobar que todos los campos de tipo numérico son validados como tal (decimales, enteros). Estado: correcto.
  4. Validación de campos tipo fecha: comprobar que todas las fechas introducidas son válidas. En caso de hacer una búsqueda entre fechas, comprobar que la fecha *desde* es menor que la fecha *hasta* para que el intervalo a buscar sea el correcto. Estado: correcto.

Sistema de grabación

1. Si el proceso de grabación tarda varios segundos (o minutos), comprobar que se ha puesto alguna marca (reloj de arena, barra de progreso...) para indicar al usuario que la aplicación está trabajando. Estado: correcto.
2. Control de errores: comprobar que se muestra un mensaje indicando el error producido y que no se graba ningún dato hasta que no se solvente el problema. Estado: correcto.

- **Borrado:** como varias pantallas tienen habilitada la opción de eliminación de registros, se hará la siguiente serie de comprobaciones en ellas:

  Sistema de validación:

  1. Eliminar un registro. Estado: correcto.
  2. Eliminar varios registros. Estado: correcto.
  3. Pulsar el botón eliminar sin seleccionar ningún registro a eliminar y comprobar que no se producen errores. Estado: correcto, se avisa con un mensaje por pantalla.

  Sistema de grabación:

  1. Control de errores: comprobar que se muestra un mensaje indicando el error producido y que no se borre ningún dato hasta que no se solvente el problema. Estado: correcto.

- **Redimensionado:** se han de poder visualizar correctamente todas las opciones de la aplicación en las configuraciones de escritorio utilizadas en Aragón Radio (1280x1024). Estado: correcto.

  Sistema de validación:

  1. Comprobar que al ejecutar la aplicación no se salga de los márgenes de la pantalla. Estado: correcto.
  2. Comprobar que al minimizar y maximizar la aplicación ésta se siga visualizando correctamente. Estado: correcto.
  3. Comprobar que al reducir o ampliar el tamaño de la aplicación su contenido se siga visualizando correctamente y se pueda seguir accediendo a las distintas opciones. Estado: correcto.

- **Operaciones sobre listados (ordenación, paginación, generación de PDF, generación de resúmenes):** en todas las opciones de búsqueda disponibles en la aplicación, se obtiene un listado de datos que pueden ser paginados, ordenables, generados en PDF y/o generados en forma de resumen.

  Sistema de paginación:

  1. Comprobar el correcto funcionamiento de las opciones de paginación: primera página, anterior, siguiente, última página. Estado: correcto.
  2. Comprobar que no se produce ningún error si se pulsan las opciones de paginación sin existir ningún dato a paginar. Estado: correcto, se muestra un mensaje informando de que no hay resultados y no se permite pulsar los botones de paginación.

  Sistema de ordenación:

  1. Ordenar el listado por fecha de creación del contenido siguiendo los criterios de ordenación ascendente y descendente. Estado: correcto.
  2. Ordenar el listado por fecha de creación del contenido y recorrer distintas páginas del listado comprobando que el sistema guarda correctamente el criterio de ordenación. Estado: correcto.

  Sistema de impresión en PDF:

  1. Comprobar que el listado de impresión no se salga de los márgenes. Estado: correcto.
  2. Comprobar que si no existen datos a listar la aplicación no imprima nada. Estado: correcto, si no hay datos no se permite la opción de generación de PDF.

  Sistema de generación de resúmenes:

  1. Comprobar que si no existen datos a listar la aplicación no genere ningún resumen. Estado: correcto, si no hay datos no se permite la opción de generación de resumen.
  2. Comprobar si el resumen obtenido es correcto: comprobar si un resumen es correcto o no es algo muy subjetivo, puesto que no existe un único resumen válido para un texto. Así pues, resulta difícil confiar en un solo método de evaluación, esto hace necesario la combinación de diferentes métodos de evaluación para garantizar una mínima calidad en el resumen. Para ello he utilizado un método intrínseco que consiste

en medir la calidad del resumen obtenido comparándolo con uno llevado a cabo por una persona, y comprobando que realmente el resumen obtenido contiene toda la información relevante. Dado que he utilizado el método ROUGE-L en la generación del resumen, este me asegura que el resumen contendrá un alto nivel de correspondencia entre los conceptos de ambos documentos al existir un alto nivel de solapamiento. Por ejemplo, si introducimos el siguiente texto:

> Adolfo Suárez, primer presidente del Gobierno de la democracia, descansa ya junto a su esposa, Amparo Illana, en la catedral de Ávila. El cuerpo del expresidente ha recibido sepultura en el claustro de la catedral de El Salvador. Con esta ceremonia, concluyen tres días en los que la familia de Suárez ha velado sus restos en la clínica donde falleció y la capilla ardiente instalada en el Congreso de los Diputados, a la espera del funeral de Estado que tendrá lugar el 31 de marzo. Así la ciudad de Ávila ha ofrecido una última muestra de respeto y afecto al primer presidente de la democracia en España... APLAUSOS LLEGADA AVILA "¡Viva Suárez!", han exclamado al unísono centenares de personas que han esperado ateridas durante horas la llegada del féretro con los restos mortales del expresidente del Gobierno, que finalmente ha tenido lugar a las dos menos cuarto de la tarde. Una ovación ha acompañado al coche fúnebre en los últimos metros antes de su llegada a la plaza de la catedral, donde ha sido recibido por el jefe del Ejecutivo, Mariano Rajoy, el presidente de la Junta de Castilla y León y el alcalde de Ávila. El obispo de Ávila, Jesús García Burillo, ha deseado en su homilía su firme fe cristiana, y su capacidad de diálogo, que ya manifestó en su tierra natal durante su juventud y que trasladó a la política. OBISPO AVILA HOMILIA SUAREZ En la plaza, desde las ocho de la mañana, comenzaban a reunirse abulenses que querían despedir al "mejor presidente que ha tenido España" y que han soportado el frío y el viento que ha azotado durante toda la mañana la ciudad. Pero el último viaje de Suárez ha comenzado poco después de las 11.00 horas, cuando su féretro ha descendido por la escalinata de la Puerta de los Leones, donde se han colocado las numerosas autoridades presentes en este acto y la familia del ex presidente. Su ataúd ha salido al son de música fúnebre, portado por un piquete de honor y en presencia de compañías de los tres ejércitos y de la Guardia Civil. El piquete, formado por diez soldados del Regimiento Inmemorial del Ejército de Tierra se ha parado en la escalinata del

Congreso para escuchar el himno nacional. HIMNO NACIONAL SUAREZ TVE ((((pisar)))Al acabar el himno, el respetuoso silencio se ha tornado en numerosos aplausos de la multitud congregada en la Carrera de San Jerónimo y se ha iniciado el desfile militar que rinde honores de Estado a Adolfo Suárez. El piquete de honor ha colocado el ataúd con los restos mortales de Suárez, envuelto en la bandera española, en un armón de artillería tirado por cuatro caballos, que ha iniciado su recorrido solemne por la Carrera de San Jerónimo y en dirección a la Plaza de Cibeles. Detrás, un soldado del Ejército del Aire portaba el Toisón de Oro que concedió el Rey a Suárez y un marinero llevaba el Collar de la Real y Distinguida Orden Española de Carlos III, otorgada ayer por el Gobierno al ex presidente a título póstumo. Y detrás del féretro sus familiares así como representantes de las principales instituciones, personalidades políticas y miles de ciudadanos se han volcado en la despedida al ex presidente.

y el texto:

La capilla ardiente con los restos mortales de Santiago Carrillo, permanecerá abierta hasta las nueve y media de la noche de hoy en el auditorio de la sede de CCOO en Madrid para que el histórico dirigente del PCE reciba el último adiós. Ayer fallecía a sus 97 años, mientras dormía la siesta, tras deteriorarse en los últimos días su delicado estado de salud. Hoy todos los demócratas han reconocido el papel fundamental que jugo en la Transición. Así con este homenaje se abría la sesión del Congreso de los Diputados tras conocerse la noticia. Allí todos los grupos políticos han destacado de Santiago Carrillo su capacidad de diálogo y su esfuerzo por ampliar los derechos de los ciudadanos. Escuchamos a Joan Tarda de Esquerra, Cayo Lara de IU, Alfonso Guerra del PSOE y Carlos Floriano del PP. Mientras tanto, en el auditorio de CC.OO el féretro con sus restos ha sido recibido con aplausos por decenas de personas que se encontraban en el exterior. Permanecerán allí hasta esta noche para que todo el que quiera pueda acercarse a darle el último adiós. Su muerte le sobrevino después de meses en los que las visitas al hospital eran constantes. Escuchamos a Santiago Carrillo hijo. Los restos mortales del histórico dirigente comunista serán incinerados mañana, jueves, en el cementerio de La Almudena. La familia llevará luego sus cenizas a la costa asturiana de Gijón para esparcirlas en el mar, como era su deseo. Hasta la capilla ardiente se han sus

Majestades los Reyes de España así como representantes políticos y amigos de la familia. Y aquí en Aragón Radio, también hemos querido recordar la figura de Santiago Carrillo. Lo hemos hecho con Ángel Cristóbal Montes, diputado de las Cortes Constituyentes en la primera legislatura, y con Miguel Ángel Loriente, que fue concejal del PCE en el Ayuntamiento de Zaragoza. Ambos han destacado que Carrillo es clave en nuestra historia pero también en la de Europa en cuánto a la democratización del comunismo. Políticos de toda índole recuerdan la figura de Santiago Carrillo, que ha muerto a los 97 años en su domicilio de Madrid a causa de una insuficiencia cardíaca, mientras dormía la siesta. Los restos mortales del histórico dirigente del PCE, que permanecen en su residencia, serán trasladados a la sede de Comisiones Obreras, donde se instalará la capilla ardiente.

Ambos textos han sido extraídos de Aragón Radio. El resumen obtenido es el siguiente:

Hoy todos los demócratas han reconocido el papel fundamental que jugo en la Transición. Así con este homenaje se abría la sesión del Congreso de los Diputados tras conocerse la noticia. Allí todos los grupos políticos han destacado de Santiago Carrillo su capacidad de diálogo y su esfuerzo por ampliar los derechos de los ciudadanos. El féretro con sus restos ha sido recibido con aplausos por decenas de personas que se encontraban en el exterior. Permanecerán allí hasta esta noche para que todo el que quiera pueda acercarse a darle el último adiós. Escuchamos a Santiago Carrillo hijo. Los restos mortales del histórico dirigente comunista serán incinerados mañana en el cementerio de La Almudena. Hasta la capilla ardiente se han sus Majestades los Reyes de España así como representantes políticos y amigos de la familia. Y aquí en Aragón Radio, también hemos querido recordar la figura de Santiago Carrillo. Lo hemos hecho con Ángel Cristóbal Montes, diputado de las Cortes Constituyentes en la primera legislatura, y con Miguel Ángel Loriente, que fue concejal del PCE en el Ayuntamiento de Zaragoza. Ambos han destacado que Carrillo es clave en nuestra historia pero también en la de Europa en cuánto a la democratización del comunismo. El obispo de Ávila, Jesús García Burillo, ha deseado en su homilía su firme fe cristiana, y su capacidad de diálogo, que ya manifestó en su tierra natal durante su juventud y que trasladó a la política. HIMNO NACIONAL SUAREZ TVE ((((pisar))) Al acabar

> el himno, el respetuoso silencio se ha tornado en numerosos aplausos de la multitud congregada en la Carrera de San Jerónimo y se ha iniciado el desfile militar que rinde honores de Estado a Adolfo Suárez. El piquete de honor ha colocado el ataúd con los restos mortales de Suárez, envuelto en la bandera española, en un armón de artillería tirado por cuatro caballos, que ha iniciado su recorrido solemne por la Carrera de San Jerónimo y en dirección a la Plaza de Cibeles. Detrás, un soldado del Ejército del Aire portaba el Toisón de Oro que concedió el Rey a Suárez y un marinero llevaba el Collar de la Real y Distinguida Orden Española de Carlos III, otorgada ayer por el Gobierno a el ex presidente a título póstumo. Y detrás del féretro sus familiares así como representantes de las principales instituciones, personalidades políticas y miles de ciudadanos se han volcado en la despedida a el ex presidente.

El resumen obtenido es legible, coherente, y cohesionado (la relación entre los elementos del texto es correcta, hay conexión léxica entre los elementos) por tanto cumple las propiedades que debe tener un resumen para una correcta evaluación del mismo [8]. Además el índice de compresión solicitado (50 %) es correcto.

- **Operaciones sobre componentes:** como todas la pantallas están formadas por componentes (botones, combos,..), que realizan funciones distintas a las mencionadas en los apartados anteriores, será necesario probarlos.

  Control de eventos:

  1. Comprobar que no se produce ningún error al pulsar sobre el componente. Estado: correcto.

  Otras funcionalidades:

  1. Verificar que al pulsar la tecla de tabulación, el cursor se desplaza correctamente y en el orden indicado por los campos de los formularios. Estado: correcto.
  2. Comprobar que no se produce ningún error al pulsar ciertas teclas del teclado (ej. F1, F2, Enter, ESC,...). Estado: correcto.
  3. Comprobar que, de activarse los eventos que se lanzan al pulsar algunas teclas, se obtienen los resultados deseados. Estado: correcto (las teclas habilitadas en la aplicación son suprimir, Intro, flechas de navegación y tabulador).

La batería de pruebas unitarias expuestas anteriormente ha sido completada con éxito.

C.2. Pruebas de integración

Las pruebas de integración se han ido realizando a lo largo del desarrollo de la aplicación, conforme se iban añadiendo funcionalidades a la misma. Se ha comprobado que las diferentes capas de la arquitectura de la aplicación interaccionan entre sí correctamente. Para ello se han realizado pruebas estructurales centradas en las llamadas entre los diferentes procedimientos y funciones que forman las diferentes capas de la arquitectura, y pruebas funcionales centradas en encontrar fallos entre módulos. Todas las pruebas de integración han sido satisfactorias.

C.3. Pruebas del sistema y de aceptación

Cuando han sido verificados todos los módulos de la herramienta por separado, se ha procedido a comprobar el correcto funcionamiento del sistema bajo problemas típicos a la hora de utilizar una herramienta en la cual se deben rellenar distintos formularios, como es el caso. Los problemas típicos probados han sido entradas erróneas de datos, buscar con datos vacíos, etc.

Posteriormente, se ha revisado uno a uno todos los requisitos definidos con anterioridad en el anexo A, comprobando que se han cumplido cada uno de ellos satisfactoriamente.

Apéndice D

Glosario

En este anexo se van a reflejar los términos que aparecen a lo largo del proyecto y que se considera importante conocer su significado para comprender completamente todas las explicaciones realizadas. Los términos son:

- **.Net:** Framework de Microsoft que permita un rápido desarrollo de aplicaciones con independencia de plataforma de hardware.
- **C#:** lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma .NET.
- **GENIE**[1]**:** propuesta arquitectónica perteneciente al grupo SID de la Universidad de Zaragoza, que implementa un conjunto de componentes cuyo objetivo es proporcionar herramientas para hacer más fácil la extracción de información sobre fuentes de información desestructuradas mediante Aprendizaje Automático, técnicas de Inteligencia Artificial, herramientas de Procesamiento del Lenguaje Natural (PLN) y Semántica.
- **Jaccard:** útil para generar resúmenes automáticos, dado que mide la similitud o disimilitud que existe entre dos oraciones [10].
- **K-means:** método utilizado en el clúster de keywords para generar resúmenes automáticos. Su objetivo es la partición de n observaciones en observaciones de k grupos en el que cada observación pertenece al grupo más cercano a la media [15].
- **Lematizar:** proceso lingüístico que consiste en reducir las diferentes formas flexivas de una palabra a la forma canónica que se selecciona como lema [3]. Siendo el lema (o raíz léxica) la forma canónica a la que se reduce todo

[1]Página web GENIE: http://sid.cps.unizar.es/SEMANTICWEB/GENIE/index.html)

paradigma flexivo y que se toma como representante de todas las variaciones morfológicas de la palabra. Por ejemplo, dada la palabra dije, su lema es decir.

- **ROUGE-L:** métrica para evaluar un resumen automático teniendo en cuenta las subsecuencias comunes más largas.
- **Servicio Windows:** programa o aplicación que se encuentra cargado en el sistema operativo Windows, y que tiene como particularidad que se ejecuta en segundo plano mientras se realizan otras tareas.
- **SQL:** lenguaje para declarar bases de datos relacionales.
- **Microsoft SQL Server 2008:** sistema para la gestión de bases de datos basado en el modelo relacional.
- **Stop words:** son aquellas palabras que son filtradas al procesar un texto, debido a su irrelevancia. Un ejemplo de estas palabras son preposiciones o artículos.
- **Tesauro:** lista de palabras o términos utilizados para representar conceptos de forma jerárquica.
- **Visual Basic .Net:** lenguaje de programación orientado a objetos implementado sobre la plataforma .NET.
- **Visual Studio .Net:** entorno de desarrollo para sistemas operativos Windows.

Apéndice E

Manual de usuario

El presente documento tiene como propósito describir de manera clara las funcionalidades de la aplicación *Desarrollo de un Gestor Documental para un Medio de Comunicación*. Contiene información detallada para su correcto manejo, de manera que cualquier persona que lo lea sea capaz de utilizar la herramienta con éxito. El manual de usuario consta de las siguientes partes:

- **Descripción:** incluye una descripción general de la herramienta junto a los requerimientos necesarios para hacerla funcionar.
- **Guía de instalación:** se describen los pasos básicos que se deben seguir para una correcta instalación y puesta en marcha de la aplicación.
- **Tutorial completo:** detalla todas las funcionalidades que ofrece la aplicación.

E.1. Descripción

La herramienta desarrollada para el departamento de documentación de la empresa Aragón Radio, es una aplicación de escritorio que permite realizar todo lo necesario para una completa gestión de la documentación. Los requerimientos de la aplicación son los siguientes:

- Un PC con sistema operativo Windows.
- Base de datos SQL Server 2008 o superior que contenga los datos.
- Framework .NET versión 3.5 o superior instalado (en caso de no estar instalado, el programa de instalación lo instalará automáticamente).
- Pantalla con una resolución de 1280 x 1024.

E.2. Guía de instalación

A continuación se detallan los pasos a seguir para la instalación del gestor documental en el equipo:

1. Introduzca el CD de la aplicación en el lector correspondiente.

2. Hacer clic en el ejecutable "Setup" contenido en el CD.

3. Se procederá a configurar Windows para el correcto funcionamiento de la aplicación (Ver Figura E.1).

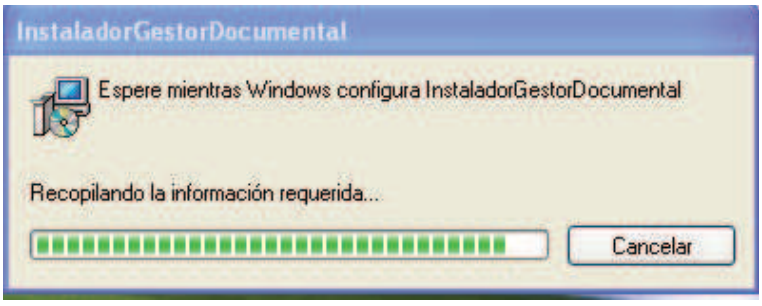

Figura E.1: Iniciando configuración

4. Una vez configurado, se procede a la instalación (Ver Figura E.2).

5. Se debe elegir el directorio donde se desea instalar la aplicación (Ver Figura E.3).

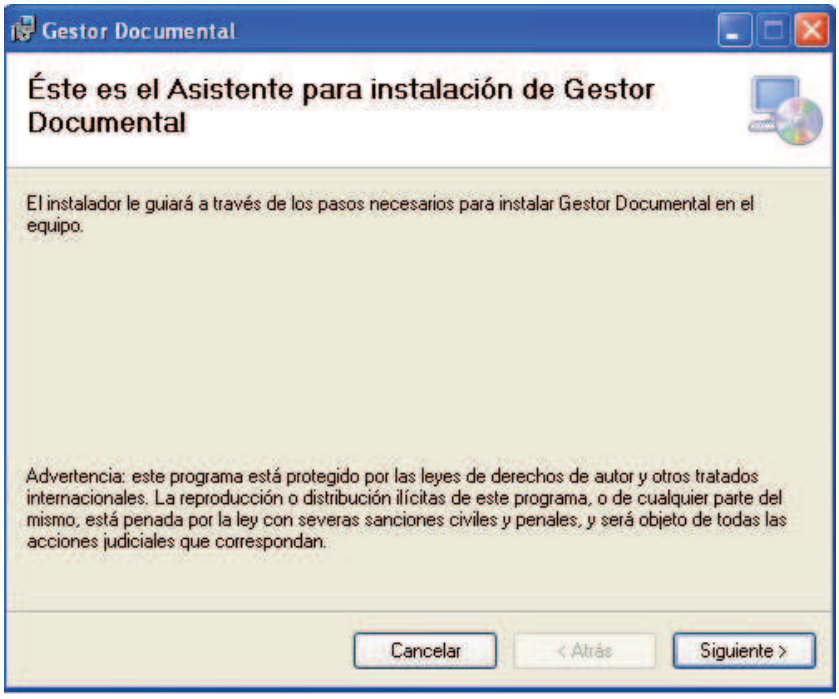

Figura E.2: Instalación gestor documental

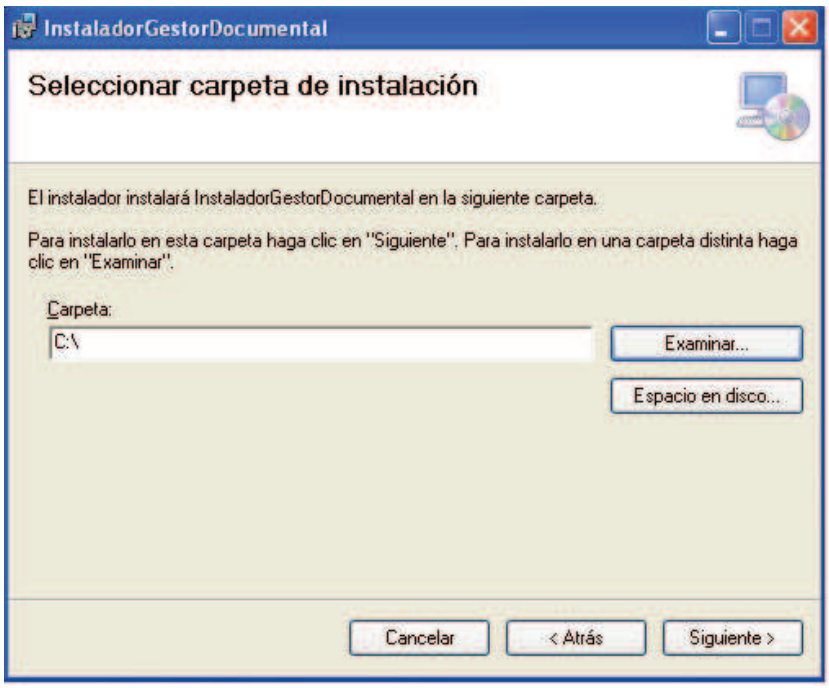

Figura E.3: Seleccionar directorio instalación

6. Cuando se ha elegido el directorio, debemos hacer clic en siguiente hasta que la instalación finalice.

7. Una vez completada con éxito la instalación, podremos ver el icono de la aplicación en nuestro escritorio (Ver Figura E.4).

Figura E.4: Icono de la aplicación

E.3. Tutorial completo

Cuando iniciamos el ejecutable "Gestor Documental Minerva" situado en el escritorio (o en Inicio/Todos los Programas/Gestor Documental Minerva), lo primero que se nos pide es nuestro usuario (login) y contraseña (Ver Figura E.5). En esta pantalla se deben introducir los datos de acceso de manera correcta para poder acceder a la herramienta de gestión. Para validar los datos introducidos se debe pulsar el botón "Acceder". En caso de introducir unos datos incorrectos, se dispondrá de tres intentos para intentarlo de nuevo, superados los tres intentos esta ventana se cerrará. Si se quieren recordar los datos de usuario para futuros accesos a la herramienta, se debe activar la opción "Recordarme". Por último, si se desea salir de la ventana de acceso debemos hacer clic en el botón "Salir".

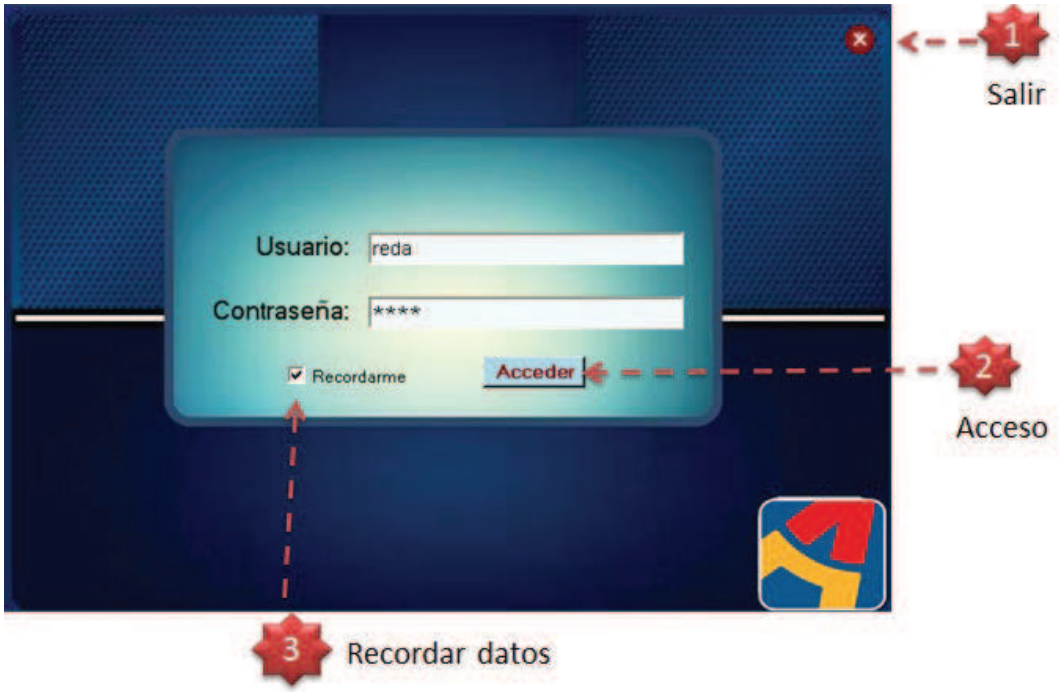

Figura E.5: Pantalla de acceso a la herramienta

Una vez introducidos correctamente los datos de usuario, se accederá a la herramienta de gestión (Ver Figura E.6) en la cual se pueden encontrar los siguientes elementos:

- **Flechas navegación:** estas flechas permiten navegar de forma cómoda por toda la herramienta a medida que avanzamos por sus diversas opciones. Para regresar a una pantalla anteriormente visitada se debe hacer clic en ellas

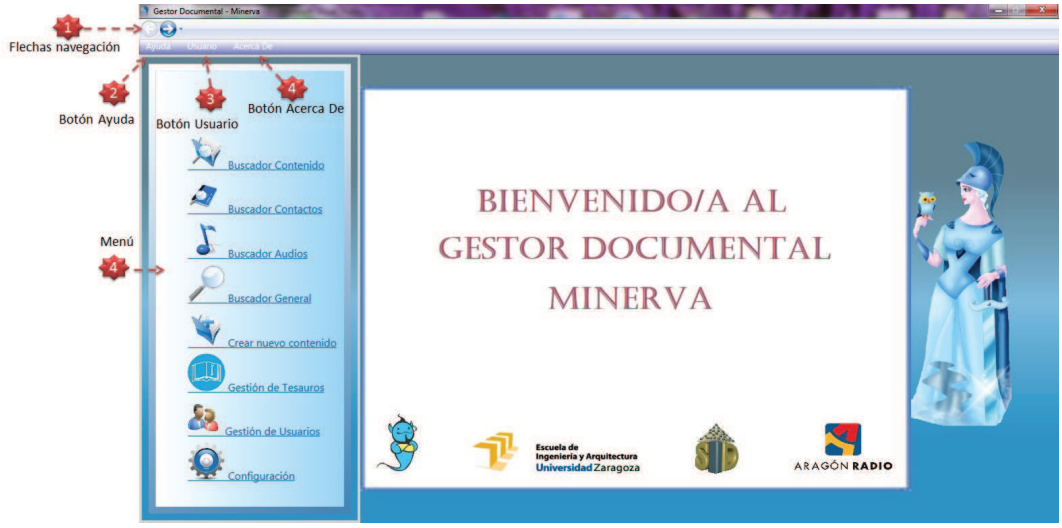

Figura E.6: Pantalla principal de la herramienta

tantas veces como queramos hasta llegar a la pantalla deseada (la flecha izquierda es para retroceder y la derecha para avanzar) (Ver Figura E.7).

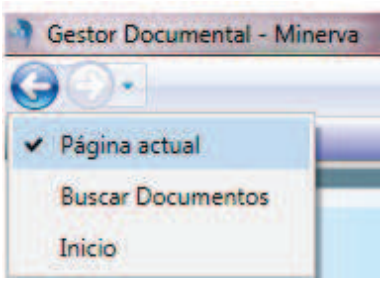

Figura E.7: Flechas de navegación

- **Botón Ayuda**: contiene una opción que abre un PDF con este mismo manual (Ver Figura E.8).

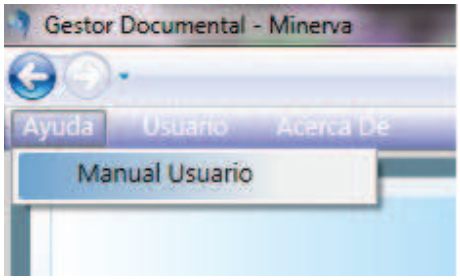

Figura E.8: Opciones del botón ayuda

- **Botón Usuario:** permite acceder al perfil del usuario que se encuentra conectado. En caso de tener privilegios de gestión se podrán editar dichos datos haciendo clic en el botón "*Editar*". Si se quiere visualizar un listado con los documentos que han sido editados o creados por este usuario, se debe hacer clic en el botón "*Acceso a tus documentos*". El formato de esta pantalla se puede ver en la Figura E.9.

Figura E.9: Pantalla perfil del usuario

- **Botón Acerca De:** muestra información sobre la aplicación.

- **Menú:** muestra las distintas opciones que contiene la aplicación.

Se va a proceder a explicar con detalle cada una de las opciones que forman parte de la herramienta de gestión en las subsecciones siguientes:

E.3.1. Buscador de contenido

Esta opción permite buscar cualquier tipo de documento almacenado en el sistema. Para poder acceder a esta opción, el usuario debe poseer permisos de lectura de documentos. Para realizar la búsqueda se dispone de un formulario divido en dos pestañas, como se puede ver en la Figura E.10.

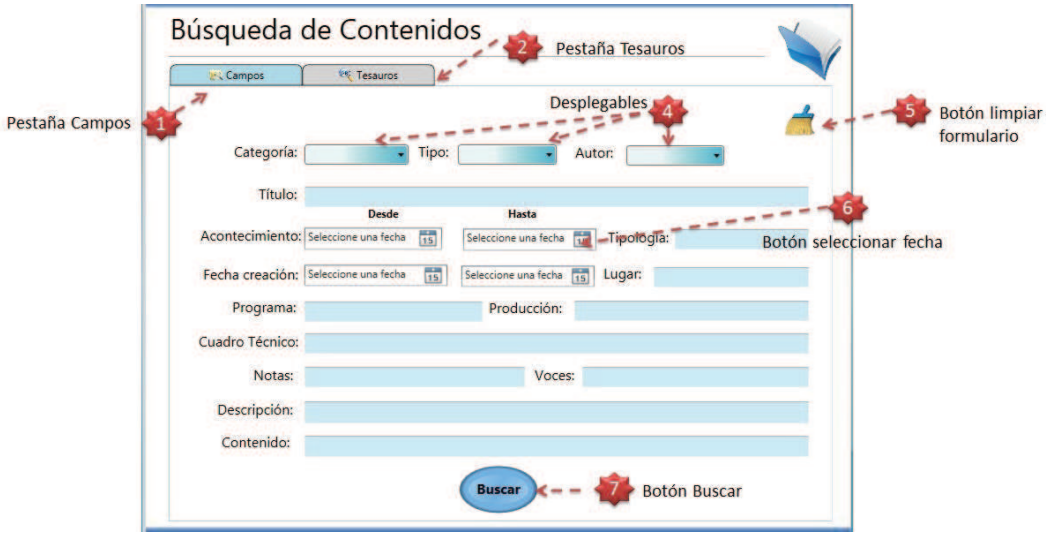

Figura E.10: Pantalla buscador de contenidos

1. **Pestaña Campos:** esta pestaña muestra un formulario con todos los campos disponibles para poder realizar una búsqueda de contenido, siendo esos campos: categoría del contenido, tipo de documentos, autor, título, tipología, fecha de acontecimiento (se permite la búsqueda entre un rango de fechas), fecha de creación (se permite la búsqueda entre un rango de fechas), lugar, programa, producción, cuadro técnico, notas, voces, descripción y contenido.

2. **Pestaña Tesauros:** esta pestaña permite añadir términos del tesauro a la búsqueda. Los términos se encuentran divididos por su tipo: onomásticos (relativos a nombres propios y entidades), topográficos (relativos a lugares) y temáticos (Ver Figura E.11).

Figura E.11: Pestaña búsqueda de términos

Para añadir un término se debe hacer clic en el botón "*Añadir término*" del tipo que deseemos añadir. Este botón se encuentra debajo de cada de tipo. Una vez hagamos clic nos aparecerá la siguiente ventana:

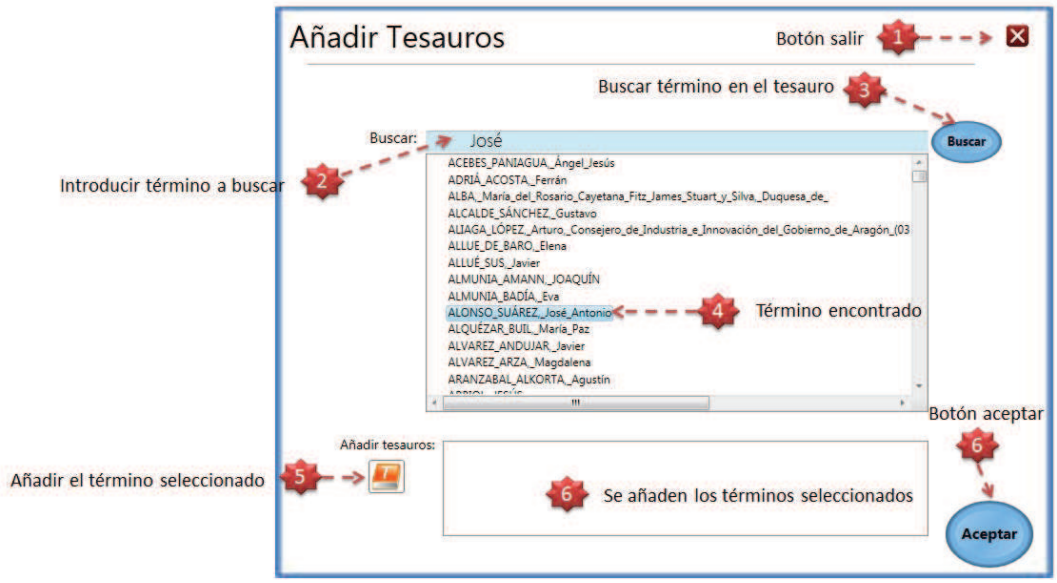

Figura E.12: Ventana añadir términos del tesauro

Dado que en este caso hemos seleccionado el tipo *onomástico* se nos muestra un listado que contiene únicamente los términos de este tipo. Para facilitar la

búsqueda del término o términos deseados, esta pantalla ofrece un buscador donde podemos indicar el término que deseamos buscar dentro del tesauro. Una vez introducido el término deseado se debe hacer clic en el botón "*Buscar*". En caso de encontrar un término que coincida con la búsqueda realizada se seleccionará (de color azul) en el tesauro, en caso contrario, se avisará de ello con un mensaje por pantalla.

Una vez seleccionado el término deseado se debe pulsar el botón "*Añadir término seleccionado*" para añadirlo a la lista de términos seleccionados, y finalmente se pulsará el botón "*Aceptar*" para confirmarlos. Si se desea salir de esta pantalla sin guardar cambios, disponemos de un botón "*Salir*" el cual nos regresa a la pantalla anterior.

3. **Desplegables:** para facilitar la introducción de datos al formulario de búsqueda, se dispone de tres desplegables: uno de categorías, otro de tipos de documento y por último otro de autores. En cada uno de estos desplegables se puede seleccionar más de un elemento para realizar la búsqueda de contenido. Un ejemplo de desplegable puede verse en la Figura E.13.

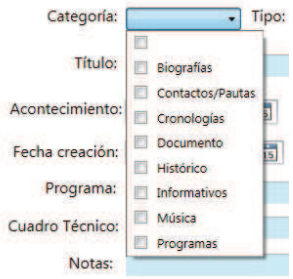

Figura E.13: Desplegable de categorías

4. **Botón limpiar formulario:** este botón permite eliminar todo el contenido del formulario, facilitando así la realización de una nueva búsqueda sin tener que eliminar manualmente uno a uno el contenido de los campos.

La búsqueda se puede realizar rellenando todos los datos del formulario para afinar la misma, o rellenando uno solo si se desea buscar un contenido que contenga un campo en concreto. En caso de que ningún campo del formulario de búsqueda se halla rellenado, se informará del error por pantalla mediante un mensaje. Para iniciar la búsqueda se debe hacer clic en el botón "*buscar*" o pulsar la tecla "*Intro*" del teclado.

Una vez realizada la búsqueda se mostrará un listado con todos los resultados encontrados. En caso de no encontrar ningún resultado siempre se avisará de ello con un mensaje. Un ejemplo de listado de resultados puede verse en la Figura E.14:

Figura E.14: Pantalla con listado de resultados

El listado de resultados ofrece distintas opciones. Estas son:

1. **Resultados encontrados:** indica el número de resultados que se han encontrado en la búsqueda.

2. **Generación de informes en PDF:** permite generar un fichero PDF con el contenido de uno o varios resultados del listado, estos se deben seleccionar y posteriormente hacer clic en el botón *"Generación de PDF"*. Una vez pulsado el botón, se preguntará donde se desea guardar el fichero PDF resultante (Ver Figura E.15). Cuando el fichero PDF haya sido generado, se nos preguntará si se desea abrirlo para ver su contenido o no. En caso afirmativo, el fichero PDF se abrirá con el formato que se puede ver en la Figura E.16.

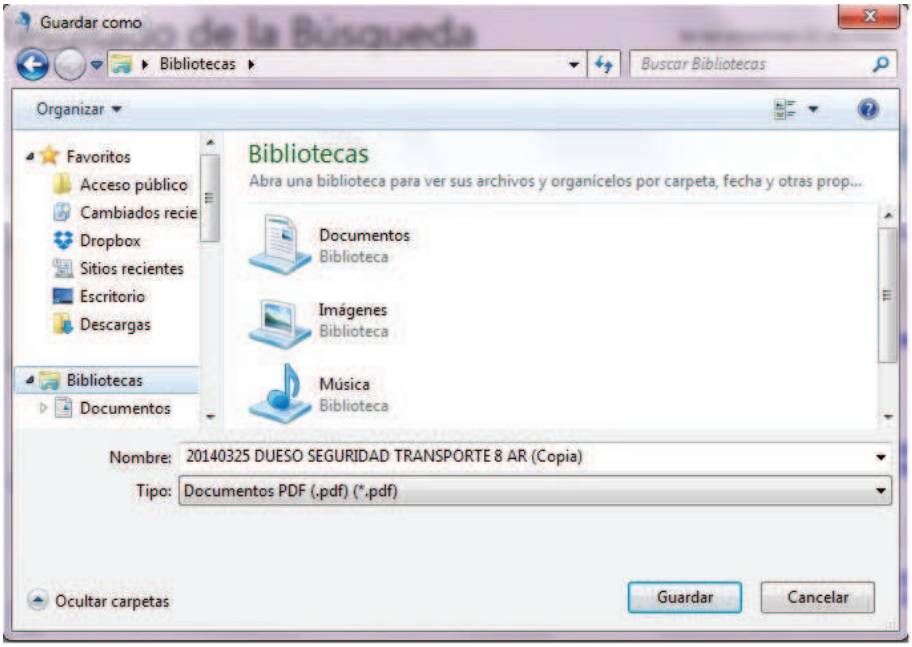

Figura E.15: Pantalla guardar fichero PDF

Informe del documento [ID:30745]

ARAGÓN RADIO

Titulo: balance del primer año de funcionamiento de la línea completa del tranvía de zaragoza desde parque goya hasta valdespartera con carmen dueso, concejal de servicios públicos del ayuntamiento.
Fecha Creación: 02/04/2014

Programa: Aragón noticias
Nombre Usuario: Documentación
NombreCategoria: Informativos
Tipo documento: Audio
Fecha acontecimiento: 26/03/2014
Cuadro Técnico: Jacobo Elizalde
Voces: DUESO, CARMEN
Producción: Aragón Radio
Contenido: El tranvía ha recorrido, en los últimos 365 días, 1 millon 295 mil kilómetros... Es uno de los balances del funcionamiento de la línea completa entre Valdespartera y Parque Goya. El servicio comenzó a prestarse hace justo un año, transportando durante todo este tiempo unos 25 millones de viajeros. Jacobo Elizalde. El pasado día del Pilar fue la jornada que más usos se han registrado, casi 138.000... y de las paradas con las que cuenta la línea, la más utilizada es plaza de España, con 12.000 personas diarias. A nivel global, desde que comenzó su andadura, en abril de 2011, ha dejado 29 accidentes. Carmen Dueso, concejal de Servicios Públicos. DUESO SEGURIDAD TRANSPORTEDe cara al futuro, elAyuntamiento sigue pensando en la línea Este - Oeste. Está redactando el pliego de condiciones pra sacar a concurso el estudio de viabilidad.

Figura E.16: Pantalla con listado de resultados

En caso de que el documento contenga fichas de audio vinculadas a él, estas aparecerán en un campo llamado "Ficheros enlazados" dentro del fichero PDF tal y como se puede observar en la Figura E.17:

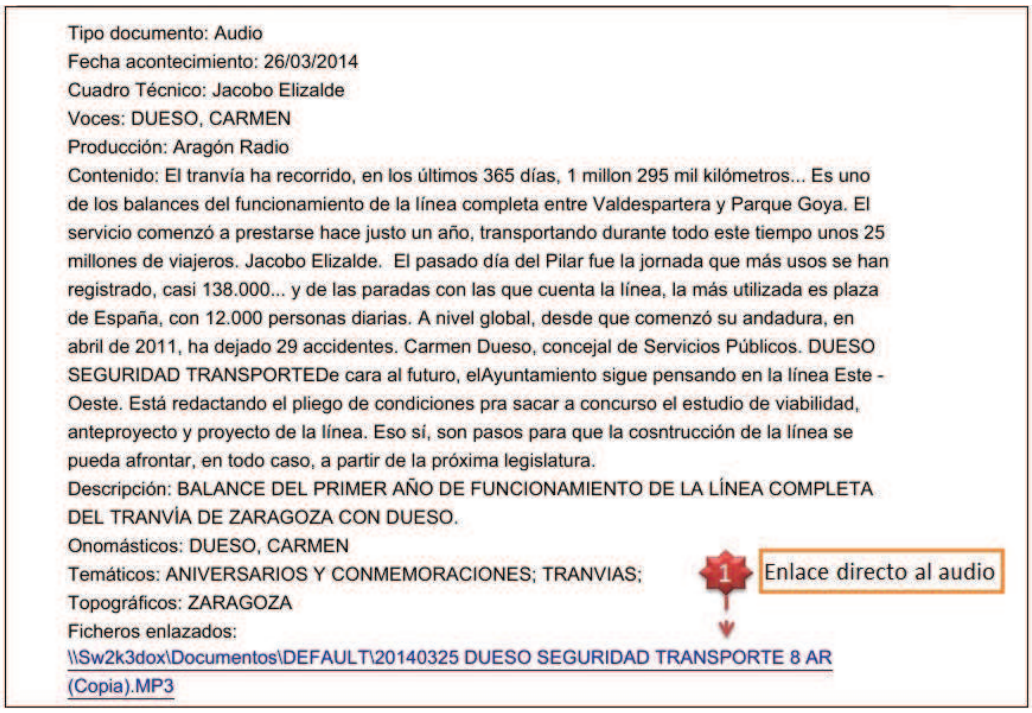

Figura E.17: Pantalla con listado de resultados

Si se desea escuchar el contenido de alguna ficha de audio, se debe hacer clic en el hipervínculo asociado a esa ficha de audio.

3. **Generación de resúmenes**: para obtener el resumen de varios resultados del listado debemos seleccionar (1 o varios) resultados, y una vez seleccionados estos debemos hacer clic en el botón "*generar resumen*". Cuando el resumen se genere (este proceso puede tardar unos segundos) se informará de ello con un mensaje y se preguntará donde se desea almacenar el fichero TXT generado, cuando se seleccione la ruta de almacenamiento el resumen se mostrará por pantalla al usuario.

4. **Borrado de contenido**: si se desea eliminar uno o varios resultados del listado, se debe proceder a seleccionar aquellos resultados que queremos eliminar y posteriormente hacer clic en el botón "*borrar contenido*" o bien pulsar la tecla "*Supr*" del teclado. Después se nos preguntará por pantalla si realmente se quiere proceder a eliminar el contenido seleccionado. En caso afirmativo se

preguntará en otro mensaje si se desea eliminar todo el contenido vinculado a este documento, es decir, las fichas de audio que se encuentran vinculadas a él.

5. **Ordenación de contenido**: por defecto los resultados se muestran ordenados de mayor a menor fecha de acontecimiento, pero si se desea se puede cambiar este orden haciendo clic en el botón "*ordenar la búsqueda*". Para volver a colocarlos como al principio, se debe hacer clic en el mismo botón otra vez.

6. **Acceso más información:** cada resultado del listado contiene a su lado un botón de información. Este botón permite acceder a toda la información que contiene ese documento, tal y como podemos ver en la Figura E.18. También podemos acceder a esta información haciendo clic en el título del documento deseado. Esta pantalla contiene tres pestañas (campos, tesauros y fichas de audio).

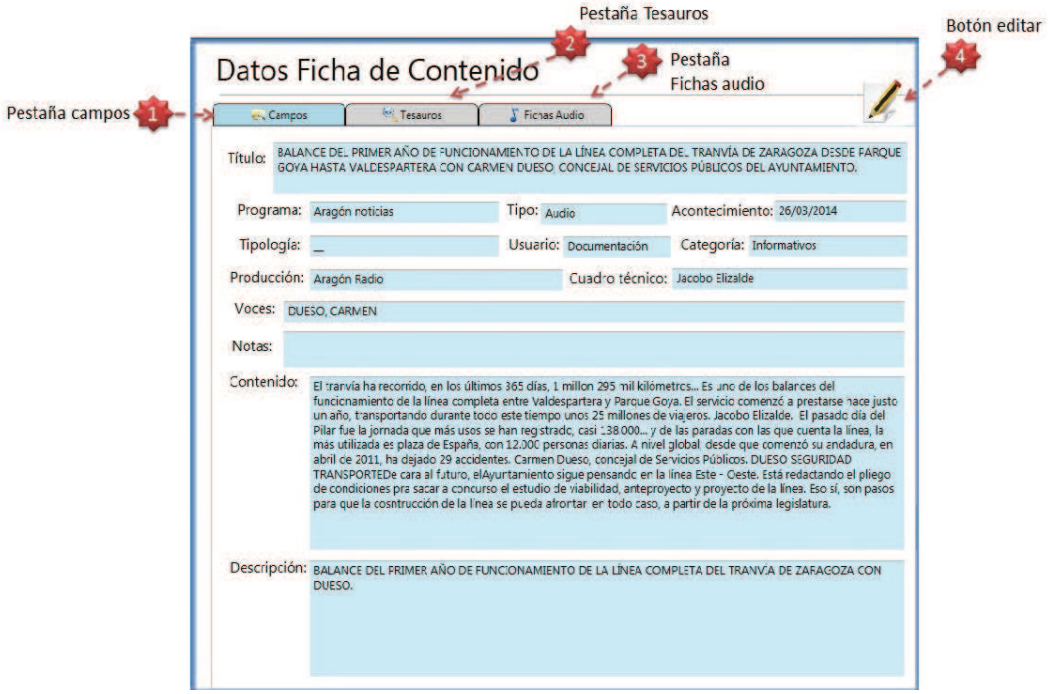

Figura E.18: Pantalla información del documento

a) **Pestaña campos:** esta pestaña muestra toda información almacenada para el documento seleccionado. Si se desea editar esta información, se debe hacer clic en el botón "*editar*" para que los campos del formulario se

habiliten y estos puedan ser modificados. En caso de no querer guardar ninguno de los cambios realizados, se debe hacer clic en la aspa roja situada en la esquina derecha superior para cancelar, en caso contrario debemos hacer clic en el botón "*guardar*".

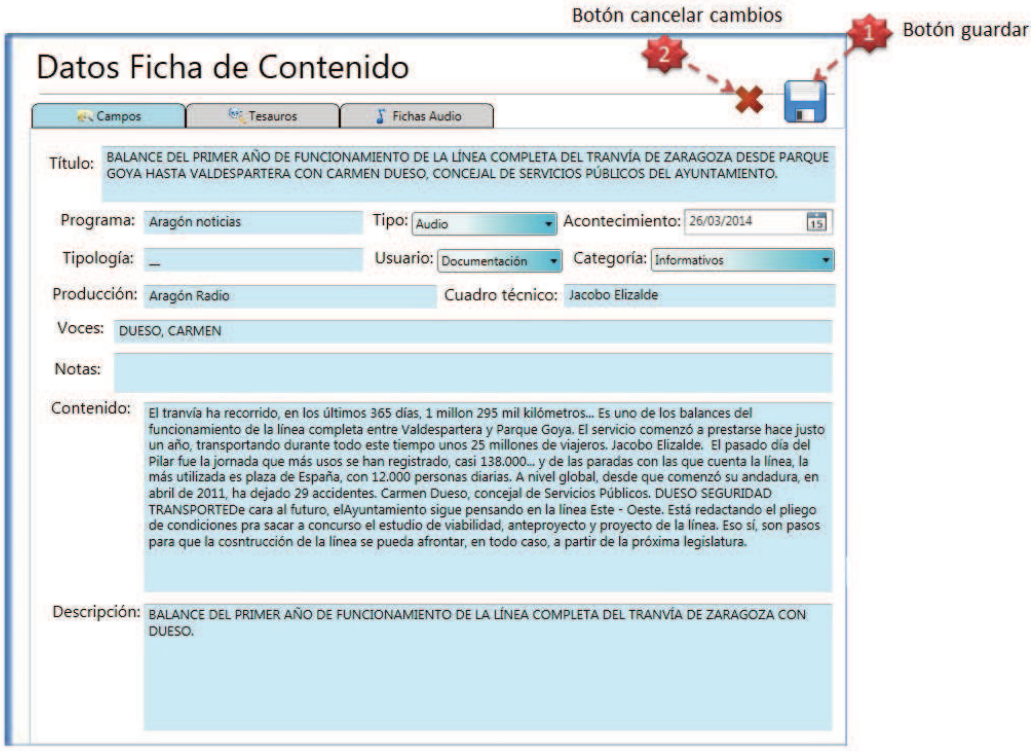

Figura E.19: Pantalla buscador de contenidos

b) **Pestaña Tesauros:** esta pestaña muestra todos los términos del tesauro que se encuentran relacionados con este documento separados por categorías.

Si se pulsa el botón "*editar*" los términos pueden ser editados. Para ello aparecerá un botón de añadir términos para cada tipo disponible (temático, topográfico y onomástico). En caso de hacer clic en alguno de estos botones, se nos mostrará la misma ventana anteriormente explicada para añadir términos en la búsqueda de contenido (Ver Figura E.21). En caso de no querer almacenar los cambios realizados se debe hacer clic en el botón "*cancelar*", en caso contrario debemos pulsar el botón "*guardar*".

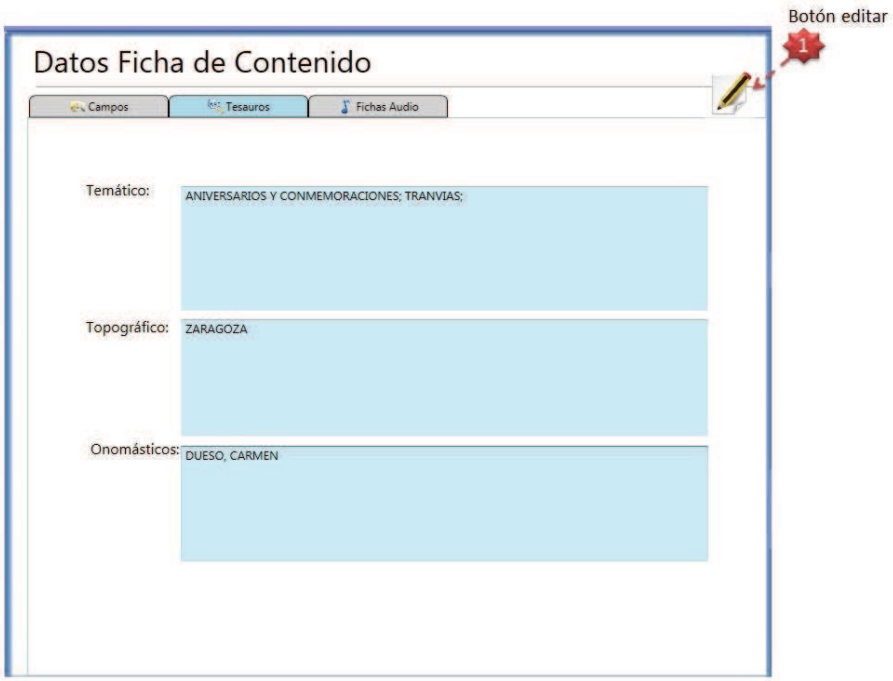

Figura E.20: Pantalla pestaña tesauros en información de contenido

Figura E.21: Pantalla editar tesauros en información de contenidos

c) **Pestaña fichas de audio**: esta pestaña muestra un listado con las fichas de audio que están vinculadas a este documento, tal y como se muestra en la Figura E.22:

Figura E.22: Pantalla fichas de audio vinculadas

Para cada ficha de audio vinculada se dispone de tres botones: información, reproducción y descargar.

Si se pulsa el botón "Información" accederemos a la información almacenada sobre esta ficha de audio (Ver Figura E.23). En esta pantalla, si se hace clic en el botón "*Editar*" se habilitarán todos los campos del formulario para poder modificarlos. Al igual que en pantallas anteriores, se deberá pulsar el botón "*Guardar*" para almacenar los cambios realizados, y el botón "Cerrar" para no almacenar dichos cambios. Esta pantalla también tiene un botón de "*Regreso*" para volver a la pantalla inmediatamente anterior.

En caso de hacer clic en el botón "*escuchar*", el audio de la ficha seleccionada se reproducirá con el reproductor multimedia VLC Media (Ver Figura E.24). En caso de hacer clic en "*descargar*", se procederá a descargar el audio en formato MP3 en el directorio que el usuario seleccione.

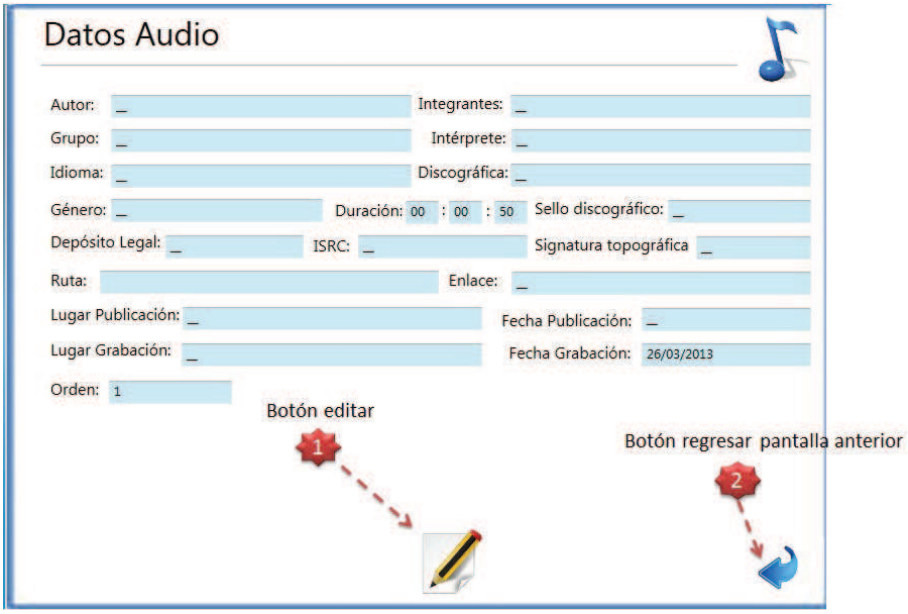

Figura E.23: Pantalla datos ficha audio

Figura E.24: Pantalla reproducir ficha de audio

7. **Navegación por los resultados encontrados:** si el número de resultados encontrados es muy numeroso, la pantalla de resultados ofrece una navegación sencilla gracias a las flechas situadas en la parte baja del listado. El primer botón permite ir a la primera página de forma directa, el segundo y el tercer botón permiten ir a la anterior y a la siguiente página respectivamente, y el último botón permite ir a la última página de resultados.

E.3.2. Buscador de audios

Este opción se asemeja al buscador de contenido anteriormente explicado, pero en este caso las búsquedas están centradas en las fichas de audio disponibles en el sistema. El usuario debe poseer permiso de lectura de documentos para poder acceder. La búsqueda se puede realizar rellenando todos los datos del formulario para afinar la misma, o rellenando uno solo si se desea buscar una ficha de audio que contenga un campo en concreto. En caso de que ningún campo del formulario de búsqueda haya sido rellenado, se informará del error por pantalla mediante un mensaje. El formato de esta opción puede verse en la Figura E.25.

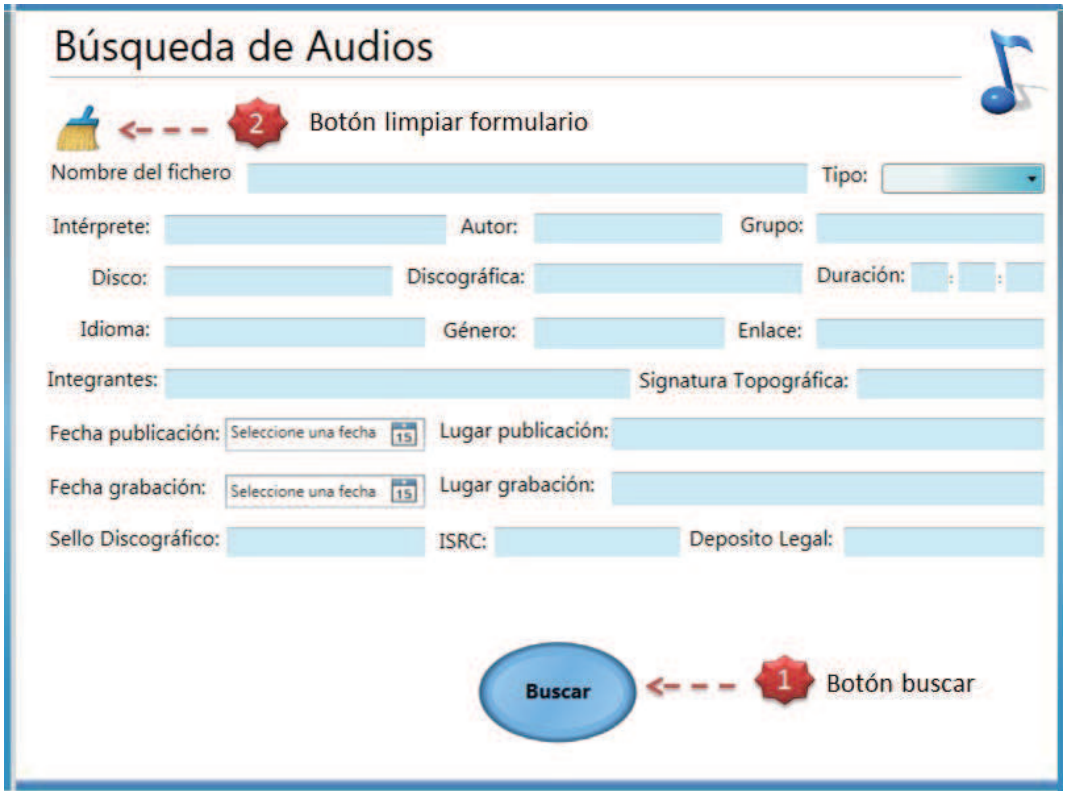

Figura E.25: Pantalla buscador fichas de audio

Para iniciar la búsqueda se debe hacer clic en el botón "*buscar*" o pulsar la tecla "*Intro*" del teclado. Una vez efectuada con éxito la búsqueda, se mostrarán los resultados en un listado como el de la Figura E.26.

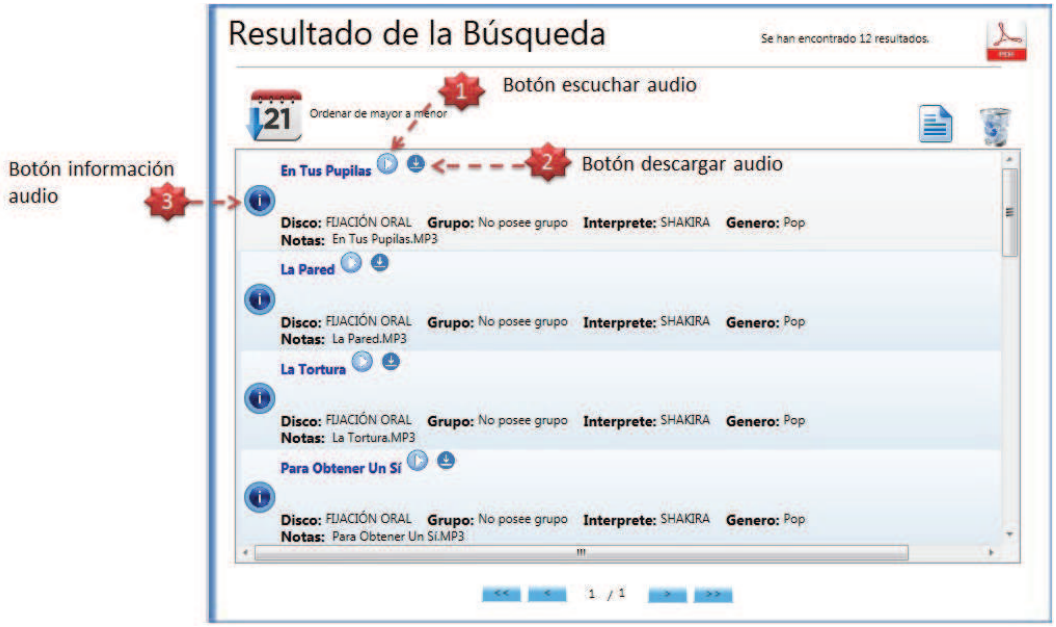

Figura E.26: Pantalla resultados búsqueda fichas audio

Esta pantalla contiene las mismas opciones que las explicadas anteriormente para la búsqueda de contenido, con la diferencia de los botones de reproducción y descarga de audio, que en este caso se encuentran al lado del título del audio y no dentro de la pantalla de información como en el caso anterior.

Figura E.27: Pantalla información ficha de audio

Si hacemos clic en el botón "*información*" de una ficha de audio cualquiera contenida en el listado (o hacemos clic en su título), podemos ver su información (Ver Figura E.27).

En esta pantalla se puede proceder a editar cualquier campo haciendo clic en el botón "*Editar*" para habilitar todos los campos del formulario. Al igual que en pantallas anteriores, se hará clic en el botón "*Guardar*" para almacenar los cambios y en el botón "*Cancelar*" para desecharlos. Un campo importante en esta pantalla es "*utilizado en*", el cual indica con qué documento está vinculado la ficha de audio, por ejemplo en este caso el documento con el que está vinculado es "En tus Pupilas". Para acceder a los datos de este documento debemos hacer clic en el botón "*Información contenido vinculante*" como podemos ver en la Figura E.28.

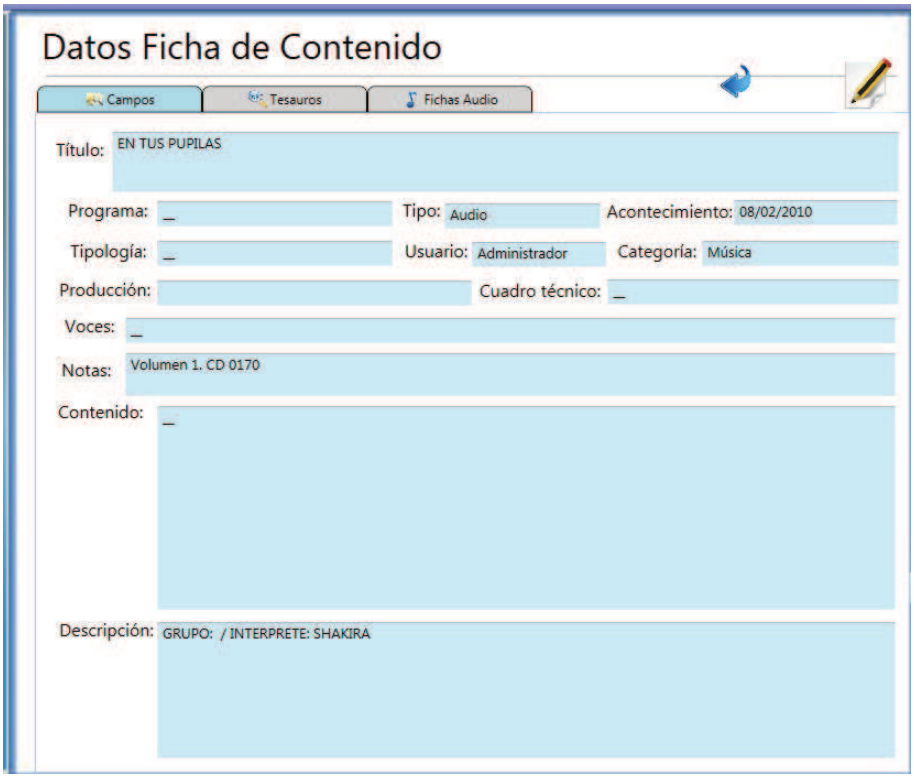

Figura E.28: Pantalla información documento vinculado

La pantalla con el contenido del documento vinculado puede ser editada y posteriormente guardada. Los datos que se pueden editar son: campos del documento y términos del tesauro. También podemos añadir nuevas fichas de audio a este documento dentro de la pestaña “Fichas de Audio”.

E.3.3. Buscador de contactos

Esta opción permite buscar cualquier contacto o pauta almacenado en el sistema. El usuario debe tener activo el permiso de lectura de documentos para poder acceder a esta opción. El funcionamiento de esta opción es similar al de la búsqueda de contenido o búsqueda de audio. En este caso los campos del formulario se encuentran adaptados a la búsqueda de contactos o pautas (Ver Figura E.29). Para realizar la búsqueda se deben introducir los campos que se deseen para acotar la búsqueda y posteriormente hacer clic en el botón "*Buscar*". Para eliminar los campos introducidos, en caso de habernos equivocado en la introducción de alguno de ellos, se dispone del botón "*limpiador de formularios*" el cual vacía el contenido de todos los campos.

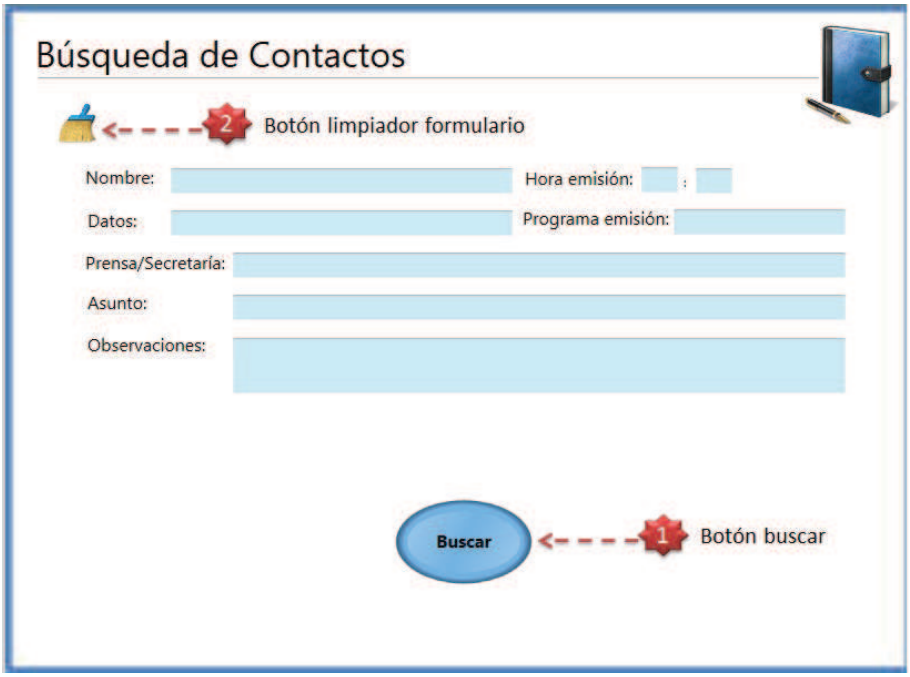

Figura E.29: Pantalla buscador de contactos

Una vez efectuada la búsqueda, obtendremos un listado de resultados análogo al anterior en apariencia y en funcionamiento. Si tal y como se ha explicado con anterioridad hacemos clic en el botón "*Información*", o en el nombre del contacto, accederemos a los datos del contacto seleccionado como se ve en la Figura E.30.

Figura E.30: Pantalla información del contactos

Estos datos pueden ser editados haciendo clic en el botón "*Editar*". En caso de no querer guardar los cambios efectuados, se debe pulsar el botón "*Cancelar*" y en caso contrario el botón "*Guardar*" para actualizar en la base de datos los datos modificados. Estos dos últimos botones aparecerán una vez pulsado el botón editar "*Editar*". Si queremos regresar a la pantalla anterior disponemos de un botón de regreso.

E.3.4. Buscador general

Permite encontrar cualquier tipo de contenido almacenado en el sistema utilizando en la búsqueda todos los campos disponibles. Para poder utilizar está opción, el usuario debe tener activo el permiso de lectura de documentos. El funcionamiento de este buscador en un poco distinto a los tres anteriormente explicados, dado que en este caso no se debe rellenar ningún formulario en particular, sino que su funcionamiento se asemeja a un buscador web.

Por ello basta con introducir una consulta y se procederá a buscar en la base de datos. El funcionamiento del buscador es el mismo si se utiliza el operador lógico AND. Si se quiere realizar una consulta exacta, la consulta deberá ir entre dobles comillas.

El buscador puede verse en la Figura E.31.

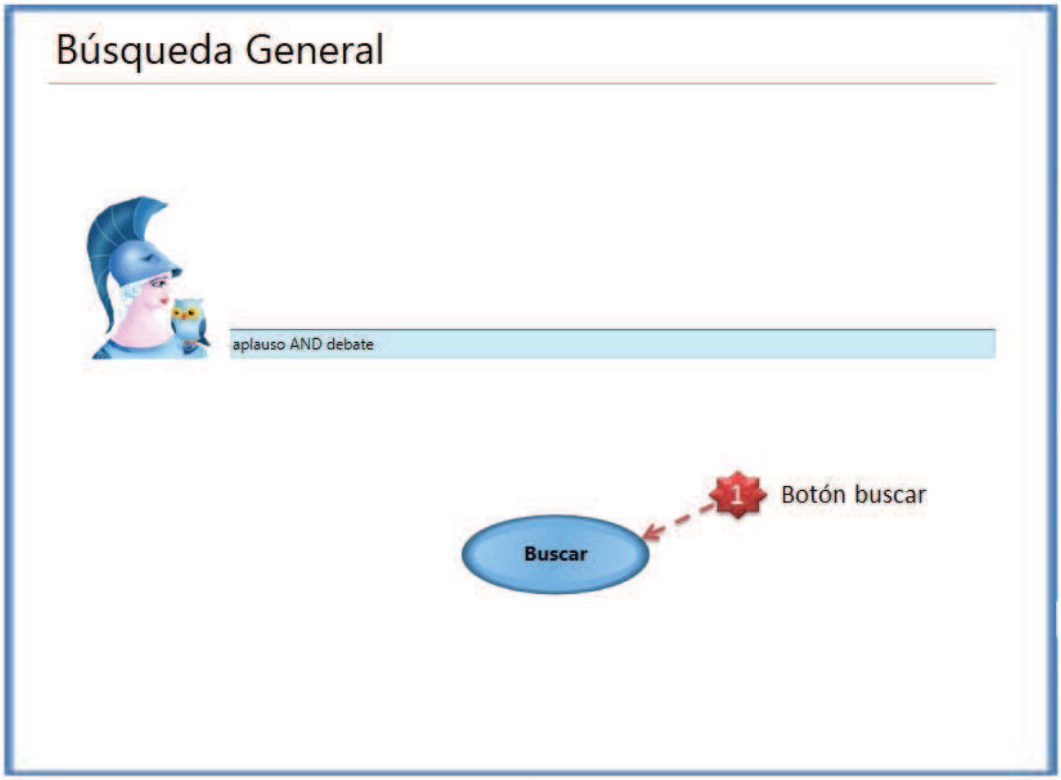

Figura E.31: Pantalla buscador general

Para proceder a realizar la búsqueda general se debe pulsar el botón "*Buscar*", y se nos mostrará un listado con todos los resultados encontrados, tal y como puede verse en la Figura E.32.

Figura E.32: Pantalla listado resultados buscador general

El funcionamiento del listado de resultados es análogo a los explicados anteriormente.

E.3.5. Crear nuevo contenido

Esta opción permite crear nuevo contenido en el sistema. El usuario debe tener activo el permiso de gestión de documentos para poder acceder.

Para poder crear nuevo contenido, primero debemos seleccionar la categoría a la que pertenece el contenido que se desea crear (puesto que los campos del formulario a rellenar son diferentes según la categoría). Para seleccionar la categoría deseada disponemos de un desplegable situado al inicio de la pantalla de creación (Ver Figura E.33).

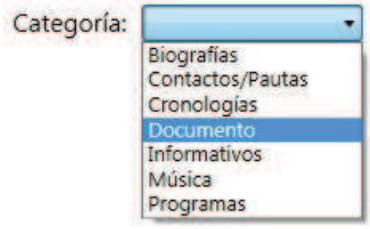

Figura E.33: Pantalla desplegable seleccionar categoría

Una vez seleccionada la categoría deseada se mostrará el formulario con los campos a rellenar como se puede ver en la Figura E.34 para la categoría "Biografías".

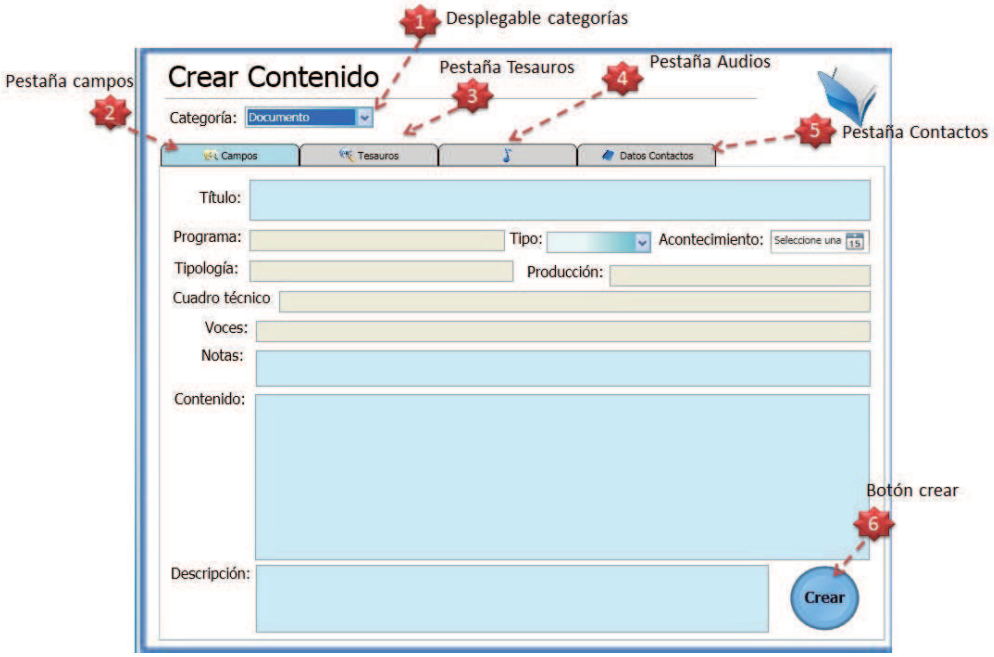

Figura E.34: Pantalla crear nuevo contenido

Esta pantalla contiene 4 pestañas:

1. **Pestaña campos:** muestra todos los campos disponibles para añadir el contenido según la categoría.

2. **Pestaña Tesauros:** permite añadir los términos deseados (onomásticos, temáticos y topográficos) que categorizan al documento.

3. **Pestaña fichas de audio:** permite vincular fichas de audio a este documento. Para añadir una nueva ficha, se debe hacer clic en el botón "*Añadir ficha audio*" (Ver Figura E.35) y accederemos a la siguiente pantalla (Ver Figura E.36):

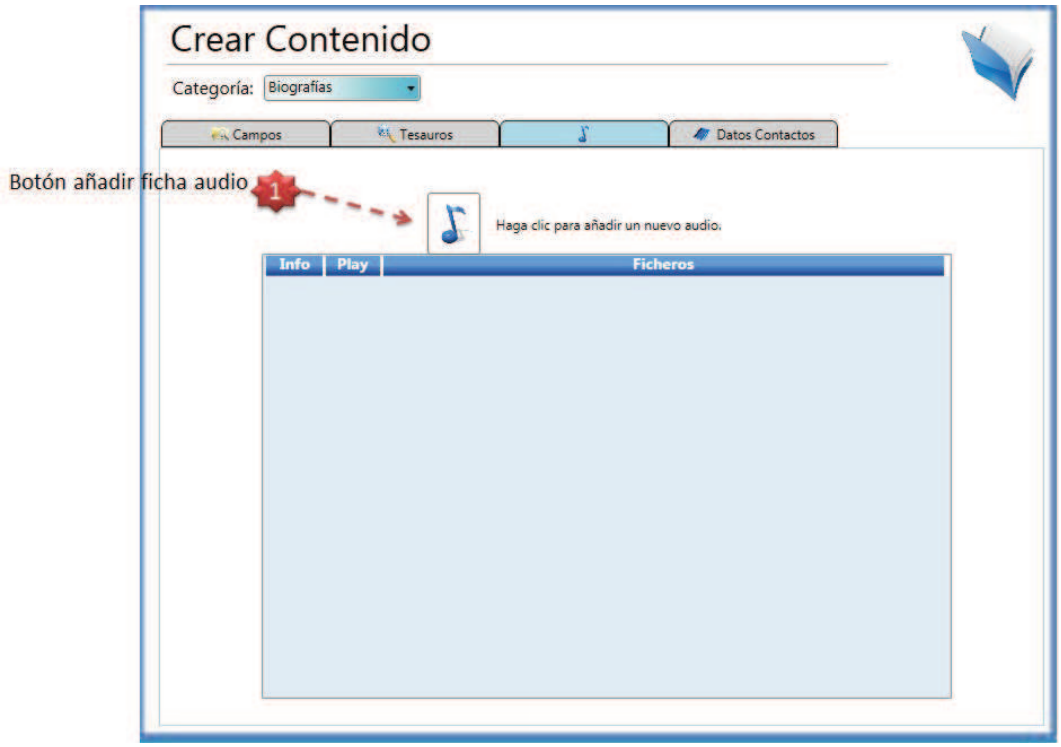

Figura E.35: Pantalla pestaña ficha de audio

En esta pantalla se deben rellenar todos los datos deseados para la ficha de audio. Si se desea añadir la ruta donde se encuentra el audio, haremos clic en el botón "*Añadir ruta*" al lado de la opción del formulario llamada "Ruta", en ese momento nos aparecerá la opción de escoger el fichero deseado entre todas las carpetas de nuestro ordenador:

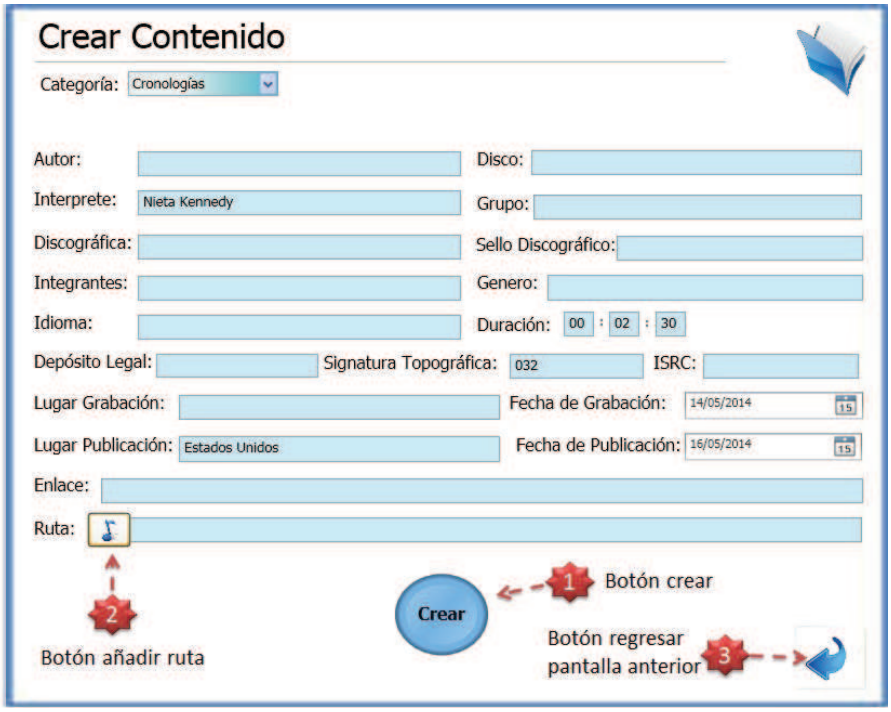

Figura E.36: Pantalla seleccionar ficha de audio

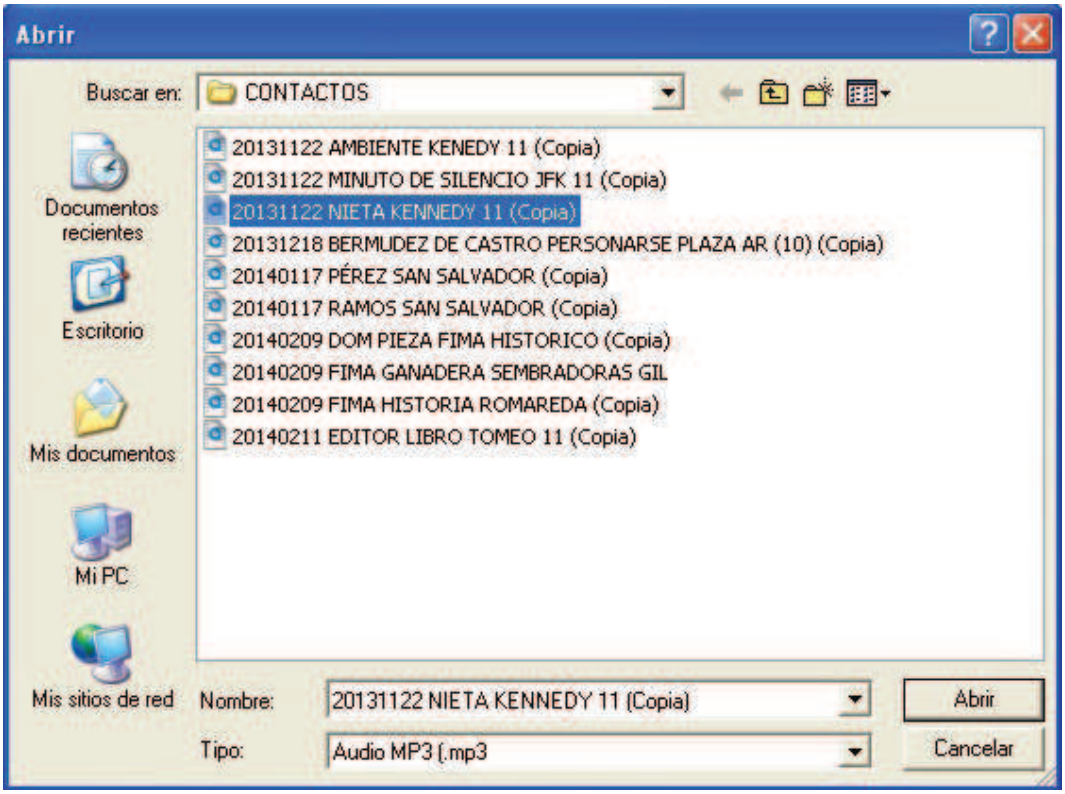

Figura E.37: Pantalla añadir ruta

Deberemos hacer clic en el botón "*Abrir*", y entonces la ruta del fichero escogido quedará añadida al formulario. Si queremos regresar a la pantalla anterior sin añadir la nueva ficha de audio, debemos hacer clic en el botón "*Regresar*", en caso contrario se hará clic en el botón "*Crear*" y la ficha quedará añadida al documento creado.

4. **Pestaña datos contacto:** permite añadir los datos de algún contacto interesante que este relacionado con este documento.

E.3.6. Gestión de usuarios

Con esta opción se puede gestionar toda la información almacenada de los usuarios que se encuentren dados de alta en el sistema. Para poder acceder a esta opción el usuario debe tener activo el permiso de gestión de usuarios, así como el permiso de cambio de clave si se desean editar las claves almacenadas para cada usuario.

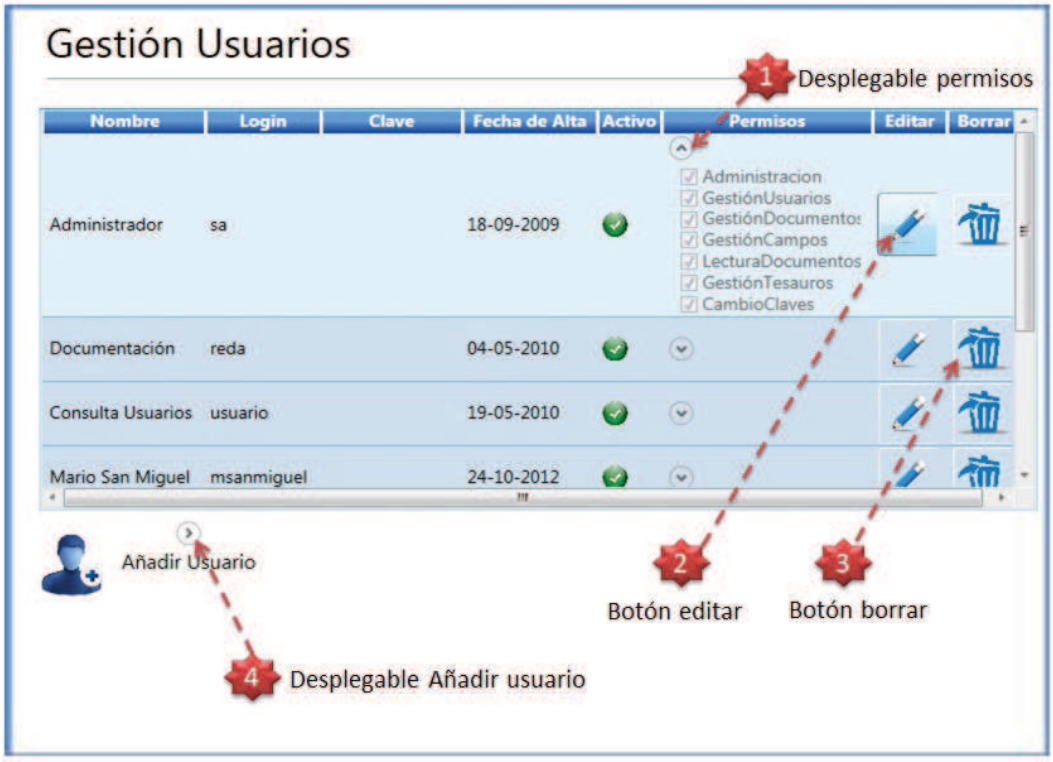

Figura E.38: Pantalla gestión de usuarios

En esta pantalla podemos desplegar los permisos de cada usuario para visualizar cuáles de ellos tiene activo y cuáles no, también podemos eliminar un usuario en concreto haciendo clic en su botón "*Borrar*". Del mismo modo, si se quieren editar los datos almacenados de un usuario se puede proceder haciendo clic en el botón "*Editar*" de dicho usuario, y nos aparecerá la siguiente ventana (Ver Figura E.39):

Figura E.39: Pantalla editar usuario

En esta ventana se pueden editar todos los campos del usuario que deseemos. En caso de no querer guardar los cambios haremos clic en el botón "*Cancelar*", en caso contrario pulsaremos el botón "*Guardar*". Para añadir un nuevo usuario al sistema, debemos desplegar la opción "*Desplegable añadir usuario*", y aparecerá lo siguiente:

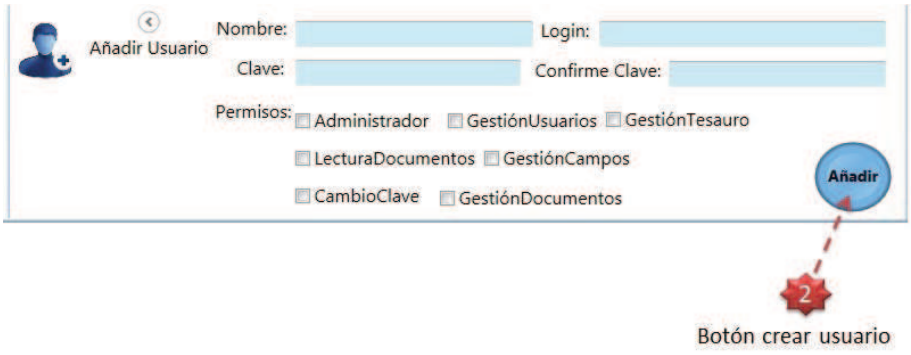

Figura E.40: Pantalla añadir nuevo usuario

En este caso se deben rellenar los datos del usuario y posteriormente hacer clic en el botón "*Añadir*". El campo "*Login*" es obligatorio.

E.3.7. Gestión de tesauros

Con esta opción se puede gestionar toda la información relacionada con el tesauro asociado al sistema. Para poder acceder a esta opción el usuario debe tener activo el permiso de gestión de tesauro.

La primera opción que nos ofrece esta pantalla es la de buscar términos dentro del tesauro. Para ello se debe introducir el término a buscar en el formulario (en este caso por ejemplo hemos introducido "España") y posteriormente debemos hacer clic en el botón "*Buscar*". En caso de encontrar alguna coincidencia en el tesauro, ese término se marcará de color azul tal y como se puede ver en la Figura E.41.

Figura E.41: Pantalla gestión de tesauros

Las otras opciones que ofrece son:

- **Añadir término:** si se quiere añadir un término al tesauro, se debe seleccionar la posición donde se quiere añadir dentro del tesauro, es decir, el término padre (por ejemplo, si quiero añadir un término a AFRICA, seleccionaré este) y posteriormente hacer clic en el botón "*Añadir término*", lo cual abrirá la siguiente ventana emergente:

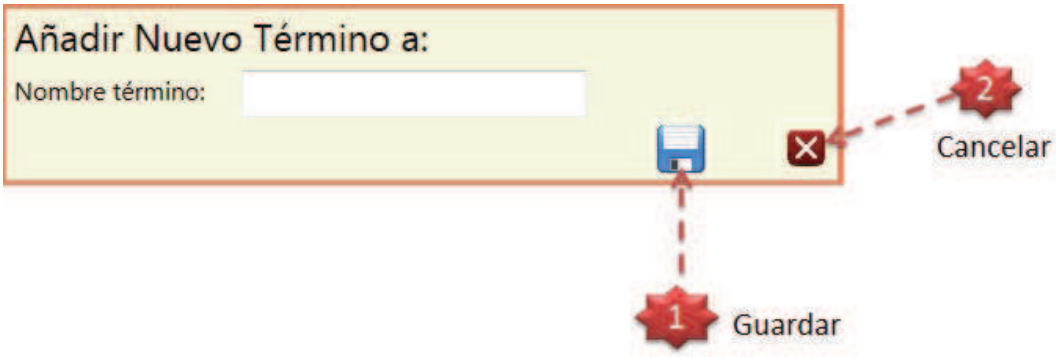

Figura E.42: Pantalla inserción nuevo término

- **Editar término:** si se quiere editar un término del tesauro, se debe seleccionar el término a editar y posteriormente hacer clic en el botón "*Editar término*", lo cual abrirá la siguiente ventana emergente:

Figura E.43: Pantalla edición de un término

Únicamente se pueden editar aquellos términos del tesauro que no se encuentren vinculados con algún documento del sistema, para evitar problemas de integridad de datos. En caso de estar vinculado se informará de ello en un mensaje por pantalla y se cancelará la edición.

- **Borrar término:** si se quiere eliminar un término del tesauro, se debe seleccionar el término a eliminar y posteriormente hacer clic en el botón "*Borrar término*". Si se da el caso de que el término a borrar tiene hijos (Por ejemplo queremos eliminar EUROPA), se preguntará con un mensaje adicional por pantalla si se desea eliminar todo el contenido de este término, es decir, si queremos eliminar EUROPA y aceptamos se borrará ALBANIA, ALEMANIA, ANDORRA,...

Únicamente se pueden eliminar aquellos términos del tesauro que no están vinculados con un documento del sistema, para evitar así problemas de integridad de datos. En caso de estar vinculado se informará de ello en un mensaje por pantalla y se cancelará la eliminación.

- **Mover término:** esta opción es muy útil cuando se quiere modificar la posición actual donde se encuentra un término en el tesauro por una nueva. Para ello se debe seleccionar el tesauro que se desea mover y arrastrarlo hasta la posición deseada. Para evitar errores, se mostrarán dos mensajes de aviso antes de efectuar el movimiento.

E.3.8. Configuración

Con esta opción se pueden configurar todos los tipos y categorías almacenados actualmente en el sistema. Para poder acceder a esta opción se deben poseer permisos de administrador o de cambio de campos. Dado que se puede elegir entre configurar tipos o categorías, la pantalla ofrece un menú en el cual se debe seleccionar que opción queremos:

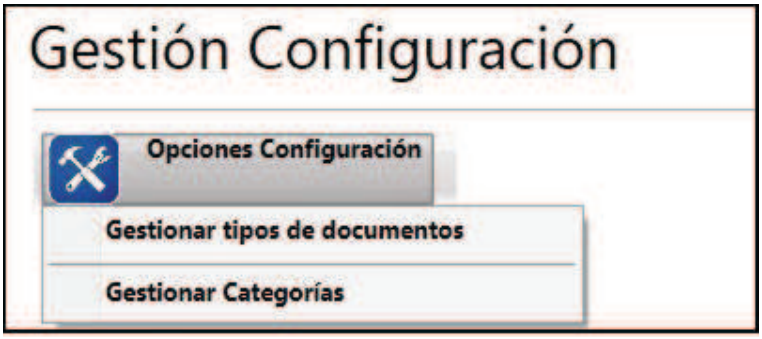

Figura E.44: Menú opciones de configuración

Si elegimos la opción "Configurar tipos de documentos" aparecerá la siguiente pantalla:

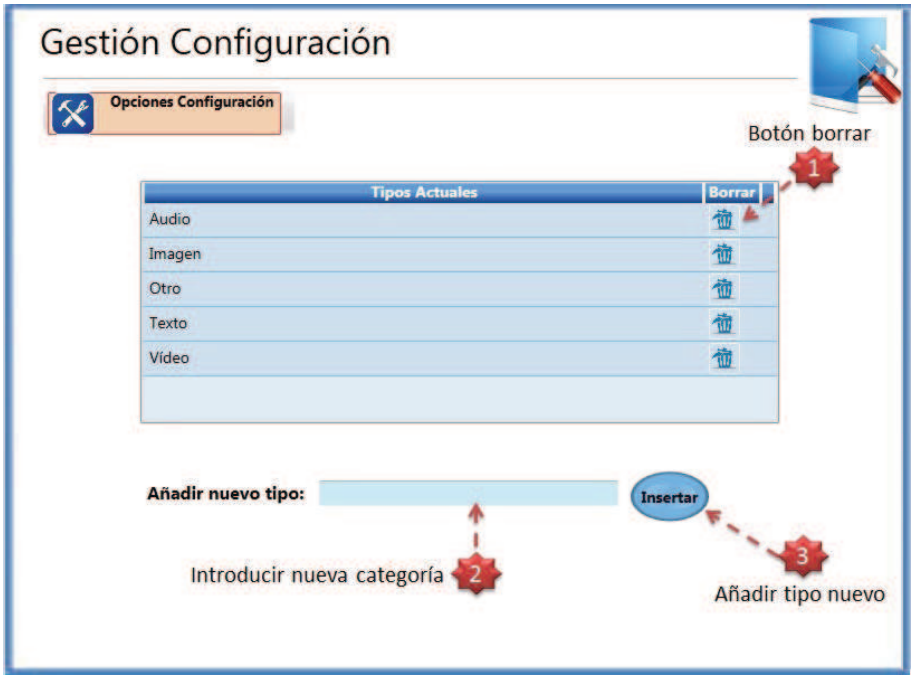

Figura E.45: Pantalla configurar tipos de documentos

En esta pantalla se pueden borrar tipos de documentos haciendo clic en el botón "*Borrar*" del tipo o tipos correspondientes. Únicamente se eliminarán aquellos tipos que no se encuentren vinculados con ningún contenido del sistema para evitar errores de integridad de datos. Para añadir un nuevo tipo, se debe introducir el nombre en el formulario situado en la parte baja de la pantalla y finalmente pulsar el botón "*Insertar*".

Si elegimos la opción "Configurar categorías" aparecerá la pantalla de la Figura E.46. En esta pantalla se pueden borrar categorías haciendo clic en el botón "*Borrar*" de la categoría deseada. Al igual que en la gestión de tipos, en este caso también se eliminarán únicamente aquellas categorías que no se encuentren vinculadas con algún contenido del sistema para evitar errores de integridad de datos. Si queremos añadir una nueva categoría, se debe introducir el nombre de esta en el formulario situado en la parte baja de la pantalla y finalmente pulsar el botón "*Insertar*". Cuando insertemos una nueva categoría, automáticamente nos aparecerá la pantalla de la Figura E.47 para editar los campos de esta nueva categoría. En caso de querer editar una categoría en concreto, deberemos pulsar el botón "*Editar*", y accederemos a la pantalla de edición:

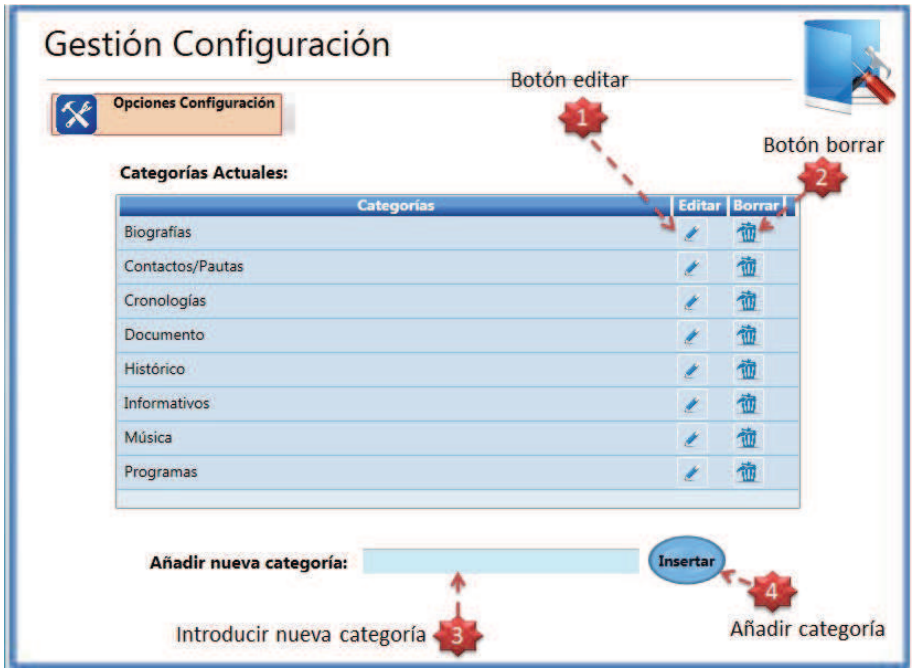

Figura E.46: Pantalla configurar categorías

Figura E.47: Pantalla editar categorías

En esta pantalla podemos cambiar el nombre de la categoría, así como los campos actuales que posee. Para cambiar los campos actuales se debe hacer uso

de las flechas situadas en el medio de la pantalla. La primera tabla muestra todos los campos que quedan disponibles para esa categoría, y en la segunda tabla (la de la derecha) muestra los campos que actualmente posee la categoría seleccionada (en este caso "Biografías"). Una vez editado se debe pulsar el botón "*Guardar*" para almacenar los datos, en caso de no querer almacenar los cambios efectuados, se debe pulsar el botón "*Cancelar*".

Apéndice F

Artículos publicados

F.0.9. TM-Gen: A Topic Map Generator from Text Documents (ICTAI 2013) (Página 164)

F.0.10. Obtaining Knowledge from the Web using Fusion and Summarization Techniques (FUSION 2014) (Página 170)

TM-Gen: A Topic Map Generator from Text Documents

Angel L. Garrido, María G. Buey, Sandra Escudero, Sergio Ilarri, Eduardo Mena
IIS Department
University of Zaragoza
Zaragoza, Spain
Email: {***garrido, mgbuey, sandra.escudero, silarri, emena***}***@unizar.es***

Sara B. Silveira
Computer Department
University of Lisbon
Lisbon, Portugal
Email: ***sara.silveira@di.fc.ul.pt***

***Abstract*—The vast amount of text documents stored in digital format is growing at a frantic rhythm each day. Therefore, tools able to find accurate information by searching in natural language information repositories are gaining great interest in recent years. In this context, there are especially interesting tools capable of dealing with large amounts of text information and deriving human-readable summaries. However, one step further is to be able not only to summarize, but to extract the knowledge stored in those texts, and even represent it graphically.**

In this paper we present an architecture to generate automatically a conceptual representation of knowledge stored in a set of text-based documents. For this purpose we have used the topic maps standard and we have developed a method that combines text mining, statistics, linguistic tools, and semantics to obtain a graphical representation of the information contained therein, which can be coded using a knowledge representation language such as RDF or OWL. The procedure is language-independent, fully automatic, self-adjusting, and it does not need manual configuration by the user. Although the validation of a graphic knowledge representation system is very subjective, we have been able to take advantage of an intermediate product of the process to make an experimental validation of our proposal.

***Keywords*-Knowledge acquisition; text mining; ontologies; topic maps; linguistics.**

I. Introduction

Nowadays, any organization owns text document collections in digital format that are constantly growing. This, coupled with the explosion of written information generated through the Internet leads us to have a huge amount of text information, so vast that it often takes a great effort to find what is really needed since most of it is hidden, messy and unsorted. So, proposals related to improving information and knowledge retrieval tools are receiving a great interest.

In the context of information systems, the usual way of representing knowledge is by using ontologies. An ontology is defined as a formal and explicit specification of a shared conceptualization [1] and it can be stored in various digital formats suitable for machines, so it is often desirable to have a tool that shows knowledge through a schema which is easier to interpret by humans.

There are different types of schemas for knowledge representation, but we have focused our attention on topic maps [2], since their simplicity makes them understandable by anyone, and besides they have a large semantic expressiveness, which allows us to store and transmit knowledge [3].

Our proposal is an automatic system called TM-GEN ("Topic Map GENerator"), capable of generating a topic map from scratch, relying on a set of input texts, and using tools without any information related to the context of the documents, in order to make the system as automatic as possible. The experimental results obtained after working with Spanish texts belonging to a news corpus from the newspaper *Heraldo de Aragón*[1] are very promising and show the interest of the proposal.

Therefore, this paper provides two main contributions:

- Firstly, we present a new architecture valid for any knowledge extraction task from unstructured text-based information, without using any context information.
- Secondly, we have faced the problem of the evaluation of topic maps by proposing a new objective method.

This paper is structured as follows. Section II describes the state of the art. Section III explains the general architecture of our solution and presents the proposed algorithm. Section IV discusses the results of our experiments with real data. Finally, Section V provides our conclusions and some lines of future work.

II. Related Work

In this section we describe the state of the art by reviewing the general aspects of topic maps, listing some existing tools, and introducing the issue of automatic summaries.

A. Context

A concept map is a diagram that shows relationships between concepts within a context. The concepts are represented in a hierarchical graph with the most inclusive and most relevant concepts at the top of the map, and the more specific and less relevant concepts below. Concept maps facilitate sense-making and learning by the individuals who make them, and by those who use them, because they are constructed to reflect the organization of the memory

[1] http://www.heraldo.es/

system. Concept maps are used in various fields, such as Biology [4], or Engineering [5]. Novak and Musonda claimed that concept maps improve learning and teaching, as they facilitate extracting knowledge and representing it graphically [6].

There is a set of standard rules to manage the representation and exchange of knowledge. The standard ISO is known as ISO/IEC 13250:2003, or more commonly as XTM (XML Topic Maps) [2]. A topic map in XML is composed of topics, associations, and occurrences. Topics are the elements of the topic map, associations are the relations among the different topics, and occurrences are the appearances of each topic in the texts. Topic maps are similar to concept maps in several aspects, although only topic maps are standardized.

B. Topic map tools

There are many tools for building automatically topic maps, for example [7], which perform an automatic extraction of topic map from web pages by crawling, through the mining of information about the topics and the relationships present in the web pages. These approaches apply heuristics for extracting this information from web sites, such as statistical and linguistic analysis, and they use annotation for the automatic extraction of linguistic entities. Other studies, like [8], propose a semi-automatic generation incorporating machine learning techniques in the process.

Although many methods for concept extraction from texts and many tools to generate topic maps have been proposed, most of them do not take so much into account the semantic relations between the topics and the associations. They usually apply semi-automatic processes that need machine learning techniques and preprocessing, but we miss in these works the fact of taking into account the semantic aspects and working them in more detail. This point is one of the distinguishing features of our work.

C. Automatic summaries

In some ways, the creation of a topic map from a document also relates to the preparation of a summary in text format. After all, in both cases, the goal is to collect the key ideas of a text and rewrite it synthetically with a new format. We have exploited this similarity for the construction and evaluation of the topic maps.

Regarding the creation of automatic summaries, it is important to mention classical works that use statistical criteria to detect sentences that contain high-frequency terms [9]. Other recent works use positional criteria [10]. Due to our interest in working with multi-documents, we have analyzed other similar works, for example [11], that use domain-independent techniques based mainly on fast statistical processing, a metric for reducing redundancy and maximizing diversity in the selected passages or that use a cluster centroid with techniques such as graph matching, maximal marginal relevance, and language generation. Also we find very interesting the recent contributions of SIMBA [12], which has a smart procedure to simplify sentences to ensure the compression of the summary. To carry out this task, SIMBA applies a two-stage process of clusterization: clustering sentences by similarity and clustering sentences by keyword.

III. Methodology

Our purpose is to create a fully automated system able to extract information and knowledge from a set of texts and collect that information in a topic map. To do its work, the first task is preprocessing the texts to find relevant information that the process needs later. Then, each text is processed separately to obtain its own topic map. This is done by dividing the text into sentences with the purpose of analyzing them separately and assigning them a relevance score, in order to find those that are most important in the text (i.e., they will give us more information and knowledge). Afterwards, TM-Gen analyzes syntactically the sentences to find the best candidates to be a topic, and then the system establishes associations between them. Next, TM-Gen carries out a semantic simplification in case there exist redundancies, incompatible associations, or ambiguities. Finally, when all the texts have been analyzed and their corresponding topic map is generated, the process merges them all into a single topic map.

A. Process pipeline

In this subsection we describe the pipeline of the process in detail:

1) *Preprocessing*: TM-Gen performs an analysis of the texts to obtain information related to their words, that is required in later tasks. To do this, we use an NLP tool to extract and lemmatize the words that constitute the texts, and to identify named entities present in them. The extracted information is stored in two catalogs:
   - A list of named entities [13] identified by the natural language processing tool, that correspond to names of people, places, organizations, etc. Before storing them, during the preprocessing task these named entities are assigned with a weight because they are relevant information present in a text. In this phase is where TM-Gen uses a gazetteer [14] to identify which named entities are locations.
   - A list of frequencies of the words present in the texts. The process uses a method that calculates statistics from words lemmatized by the NLP tool and included in the texts and calculates their frequencies.
2) *Extraction of keywords*: It extracts a list of keywords from each text using the well-known TF-IDF (Term

Frequency - Inverse Document Frequency) [15]. Later, these words allow us to identify which parts of the text provide more knowledge. In this phase we use the frequencies of the words extracted in the previous preprocessing task.

3) *Extraction of Named Entities*: TM-Gen extracts a list of named entities from each text by using an NLP tool. These named entities are important candidates to be concepts in the topic map, as they provide relevant information and each of them constitutes a concept itself. It also takes into account the frequencies of these named entities extracted in the preprocessing phase.
4) *Text split in sentences*: It divides each text into sentences to identify later which of those sentences have more relevant information.
5) *Sentences scoring*: TM-Gen uses a method to score the sentences in order to identify which ones are the most relevant. For this task, it takes into account the keywords and named entities that appear in a sentence and it scores each one using the frequencies of the words extracted in the preprocessing step. This method is described in more detail in Section III-C.
6) *Sort sentences*: Once the sentences have been scored, they are ordered from the highest relevance to the lowest. At this point, the set of the sentences that are placed higher on the list (i.e., with higher scores) constitutes already a summary of the text.
7) *Sentence analysis*: The process performs a syntactic and grammatical analysis of each sentence in order to identify the function of each word in the sentence and its type. For this purpose we use a parser tool [16]. It is important to extract this information from the sentences because the best candidates to be concepts in the topic map are the words whose function is to be the subject of the sentence; a subject is typically a noun, noun phrase, or pronoun. It is also important to identify verbs in the sentence, as they establish associations between topics.
8) *Addition of topics*: TM-Gen adds to the topic map the concept candidates found in the previous step. These candidates are the subjects identified. For each candidate, if it has been previously included in the topic map, then the topic is not added but it will be assigned with a higher weight. The intuition is that the topics that are repeated more than once in several sentences tend to have more relevance in the text and they usually provide more information.
9) *Addition of associations*: TM-Gen adds to the topic map the associations between topics found in each sentence. These associations are given by the verbs present in the sentence. TM-Gen performs this task by searching the subject included as topic, and then it adds the verb as its association. Finally, it links its verb complement with the topic and with the association as a new topic.
10) *Semantic simplification*: Once topics and associations are added into the topic map, the process performs a semantic analysis of them and it makes a simplification of the topic map in case it finds redundancies, incompatible associations, or ambiguities. This task is described in more detail in Section III-D.
11) *Evaluation*: Each time the system adds elements corresponding to a sentence into a topic map, an evaluation process is performed. In this process it checks if the number of topics added is enough, and if so it stops processing sentences and it continues with the next task. This number can be adjusted according to the desired depth level of the topic map.
12) *Validation*: The process checks if the topic map has been built correctly according to the standards for topic maps [2], and it also searches for possible mistakes generated in previous phases of the process and fixes them.
13) *Merge*: Once the topic map for each text in the input set has been generated, TM-Gen performs a merging of all of them into a single topic map. To do this, we use a method previously developed in [17]. The method for merging is called SIM (Subject Identity Measure) and it is responsible for describing the relation among two subjects or topics.
14) *Creation of concept maps and ontologies*: The process of design and creation of ontologies can be a pretty complex process. For this reason, we suggest the utilization of a representation that can be integrated with ontologies, so these can be obtained automatically. For the conversion from topic maps to ontologies we need a concept map in the XTM language to create a corresponding file in the OWL language [18]. Furthermore, to perform the graphic representation of topic maps we use ONTOPIA [19], as it automatically draws topic maps from an XTM file.

B. Algorithm

The next algorithm that is detailed below in pseudo-code explains the process detailed in the previous sections:

```
PROGRAM TMGen (
        INPUT: g as Gazetteer,
               LDB as LexicalDB,
               txt_list as ListOf(string);
        OUTPUT: tm_ok as topic_map);
BEGIN
   tm_list := empty_list;
   Preprocessing(txt_list);
   FOR EACH text IN txt_list DO
       tm := new topic_map;
       kw_list := GetKeywords(text);
       ne_list := GetNamedEntities(text,g);
       sentence_list := SplitText(text);
       FOR EACH s IN sentence_list DO
```

```
            ScoreSentence(s,kw_list,ne_list);
        END FOR
        SortSentences(sentence_list);
        WHILE NOT fit(topic_map) DO
            s := sentence_list.next;
            c := TopicAnalysis(s);
            assoc := AssocAnalysis(s);
            tm := AddTopic(c, tm);
            tm := AddAssoc(assoc, tm);
            SemanticSimplif(tm,LDB);
        END WHILE
        Validate(tm);
        tm_list.add(tm);
    END FOR
    tm_ok := Merge(tm_list);
    Output := tm_ok;
END.
```

C. Determining the relevant sentences

The proposed method searches in a sentence the words that match with keywords and named entities that appear in the entire text. We use an approximation of the method proposed in [12].

The sentence scoring procedure defines the sentence relevance in the overall collection of sentences obtained from each input text, and it is the sum of the `tf-idf` scores ($tfidf_w$) [15] (computed considering the word lemmas obtained previously) of each word of the sentence s, smoothed by the number of words in the sentence (*totW*). This metric states that the relevance of a sentence not only depends on the frequency of the words present in it, but also on the number of texts in which the words appear. Equation 1 describes the computation of this score:

$$score_s = \frac{\sum_w (tfidf_w)}{totW} \quad (1)$$

The next stages of processing aim at identifying relevant information in the collection of sentences, in two steps:

1) *Similarity clustering*: In order to identify sentences conveying the same information, the process clusters them considering their degree of similarity. The similarity between two sentences comprises two dimensions, computed considering the word lemmas: the sentences subsequences and the word overlap. These two dimensions are combined in a similarity value. If the similarity value is higher than the similarity threshold, the sentences are grouped in the same cluster. The sentence with the highest score from each cluster is selected to be used in the next step.
2) *Keyword clustering*: The algorithm that clusters sentences by keywords is an adapted version of the K-means algorithm [20]. Keyword clustering groups sentences based on the occurrence of the keywords. The keywords extracted previously represent the clusters. A sentence is added to the cluster represented by the keyword that occurs more often in it. The sentences that do not contain any keywords are ignored.

The next task of the process orders the sentences based on their score after both clustering phases have been executed, defining the order of the sentences to be processed in the next tasks. These sentences are considered more significant than the others, since they address the main topics conveyed by the collection of sentences.

D. Semantic simplification

Most methods and techniques developed to extract concept maps from texts usually take into account syntactic and grammatical information of the words that compose a sentence (or a text), and statistical analysis of the most relevant words, but there are very few proposals that exploit semantic aspects. Thus, it is possible that the topic map generated includes topic redundancies. This means that there could exist several topics that relate to the same concept, i.e., they are synonyms. At this task, TM-Gen conducts an analysis to search this type of redundancies and, in case of finding them, removes and reduce them into a single concept, using for this purpose a lexical database that contains semantic information of the words of a language. For example, if the process has generated a topic map where the topics "objective" and "purpose" appear, this method will search and select their best meaning and reduce them into only one topic, gathering and connecting their associations with it.

At the same time, the process can identify ambiguities in the topics. Thus, two or more topics can have the same meaning in a certain context but not in others. In this case, we use a disambiguation engine [21] taking into account associations and topics that are close to those concepts in the topic map hierarchy. If the process finds that they are synonyms, then it reduces them in the way described above.

Moreover, the process is also responsible for identifying the most general concepts, in order to extend and generalize the knowledge content extracted. For example, if it finds "kids" in the topic map it substitutes the word with "children", which is more general and formal. At this phase, TM-Gen also recognizes the possible incompatible associations between topics. This occurs when two completely opposite associations leave or arrive at a topic, or when an association does not share a relation with a topic. In this case, it tries to solve it by using the disambiguation engine or remove them if it does not find a solution. This method helps to correct errors derived from the previously mentioned simplification or errors in the text itself.

IV. Experimental evaluation

This section discusses the results of experimental tests carried out to check the performance of the proposed algorithm. For testing in a real environment we have used a corpus of 4,433 news of *Heraldo de Aragón*, a major Spanish media, containing sets of texts in Spanish and its

corresponding summaries made by the professional documentation department of the newspaper. The purpose of our system is to approximate the automatic generation of knowledge to those professional summaries. To evaluate this, we used an intermediate product of our process: the natural language summary used as a basis to construct the topic map of each text. So, we will compare our automatic summary to the abstract elaborated by expert humans.

We compared our summaries with the summaries elaborated by an external tool: SweSum[2] [22]. In our system, we have used Freeling [23] as Spanish NLP tool, both for morphological analysis as for syntactic analysis of the texts. As a lexical database we have used EuroWordNet [24] concerning the evaluation itself. After both summaries (the one generated by SweSum and the one obtained by our system) had been built for each input, they were compared with the corpus of ideal summaries using ROUGE [25], which is a package for automatic evaluation of summaries.

The results of the experiments are shown in Fig. 1.

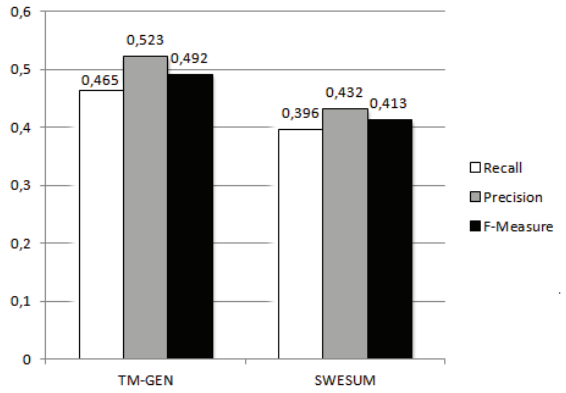

Figure 1. ROUGE-L metrics for TM-Gen and Swesum summaries

We analyze the following measures, commonly used in the Information Retrieval context [26]: the precision, the recall, and the F-Measure. By observing Fig. 1, we can see that TM-Gen has a better performance than SweSum. The F-Measure value for TM-Gen overcomes the one of SweSum in eight percentage points, meaning that TM-Gen summaries have more significant information than the ones of SweSum. The precision value obtained by TM-Gen is very interesting. A high precision value means that, considering all the information in the input texts, the retrieved information is relevant. Thus, obtaining the most relevant information in the sentences by discarding their less relevant data, which is ignored during the construction of the topic map, ensures that the summary contains indeed the most important information conveyed by each of its sentences. The recall values of the two systems are closer than the ones concerning precision. Thus, considering the evaluation results, we can conclude that this approach produces better summaries when compared to the one used by SweSum.

Regarding the graphical representation, the impression is that the topic maps generated are very suitable with respect to clarity, accuracy, and usefulness of the information represented. As the topic map is encoded in XTM, the results can be displayed in graphical form by using any design tool for concept maps, such as Cmap[3] or ONTOPIA. An example of the final appearance of the topic map can be seen in Fig. 2.

V. Conclusions and Future Work

In this paper, we have presented a new method to obtain knowledge information from a set of unstructured documents in natural language, with the only help of a morphological analyzer, a lexical database, a syntactic parser, and a generic Gazetteer, and without any help of the context. The idea is to summarize the set of texts using statistics methods and NLP tools, and then to use the parser to build a topic map, and the lexical database and semantics rules to simplify it. After that, we merge the results and we generate automatically an RDF/OWL file to store the knowledge. Besides, we have also developed an evaluation process to quantify how appropriate a topic map is to describe the embedded knowledge of a set of texts, so we have been able to compare our method with others. Our main contributions are:

- Developing an algorithm to fully automatically obtain a topic map from a set of texts, and therefore a knowledge representation of them using RDF or OWL.
- Introducing a new method to build a topic map from a set of sentences using a syntactic parser.
- Proposing a new way to simplify topic maps using the semantic relationships between its elements.
- Measuring quantitatively the quality of our knowledge representation by using the summaries (an intermediate product in the generation process).

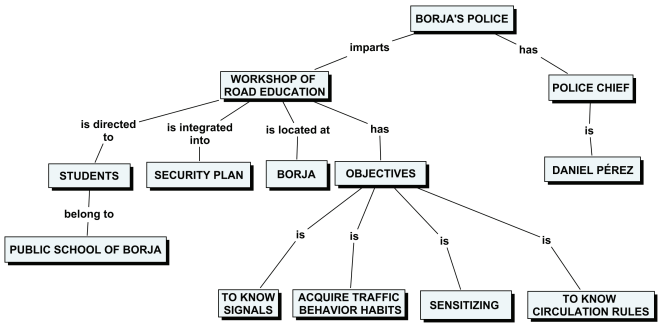

Figure 2. Example of a topic map automatically obtained from a set of texts

The main and most valuable application of this technique is to create catalogs, thesauri and ontologies that serve to

[2]http://swesum.nada.kth.se/index-eng.html

[3]http://ftp.ihmc.us

automatically categorize a repository of unstructured text-based information. The proposal has been tested in the real environment of a major Spanish newspaper and we were able to take advantage of the fact that we have thousands of texts summarized by a professional team of archivists, which has allowed us to verify accurately the results of our tests. In any case, we believe that the algorithm used is independent of the language, question that we plan to test in a short-term.

ACKNOWLEDGMENT

This research work has been supported by the CICYT project TIN2010-21387-C02-02 and DGA-FSE.

REFERENCES

[1] T. R. Gruber *et al.*, "A translation approach to portable ontology specifications," *Knowledge Acquisition*, vol. 5, no. 2, pp. 199–220, 1993.

[2] S. Pepper and G. Moore, "XML Topic Maps (XTM) 1.0 - TopicMaps. org specification," *TopicMaps. Org Authoring Group, http://www.topicmaps.org/xtm*, 2001.

[3] A. Simón, L. Ceccaroni, and A. Rosete, "Generation of OWL ontologies from concept maps in shallow domains," in *Current Topics in Artificial Intelligence*, pp. 259–267, Springer, 2007.

[4] K. M. Markham, J. J. Mintzes, and M. G. Jones, "The concept map as a research and evaluation tool: Further evidence of validity," *Journal of research in science teaching*, vol. 31, no. 1, pp. 91–101, 1994.

[5] J. Turns, C. J. Atman, and R. Adams, "Concept maps for engineering education: A cognitively motivated tool supporting varied assessment functions," *IEEE Transactions on Education*, vol. 43, no. 2, pp. 164–173, 2000.

[6] J. D. Novak and D. Musonda, "A twelve-year longitudinal study of Science concept learning," *American Educational Research Journal*, vol. 28, no. 1, pp. 117–153, 1991.

[7] K. Böhm, G. Heyer, U. Quasthoff, and C. Wolff, "Topic map generation using text mining," *Journal for Universal Computer Science (J. UCS)*, vol. 8, no. 6, pp. 623–643, 2002.

[8] L. Kásler, Z. Venczel, and L. Z. Varga, "Framework for semi automatically generating topic maps," in *Third International Workshop on Text-based Information Retrieval (TIR'06)*, vol. 205, pp. 24–30, CEUR Workshop Proceedings (CEUR-WS.org), 2006.

[9] H. P. Luhn, "The automatic creation of literature abstracts," *IBM Journal of Research and development*, vol. 2, no. 2, pp. 159–165, 1958.

[10] C.-Y. Lin and E. Hovy, "Identifying topics by position," in *Fifth Conference on Applied Natural Language Processing (ANLC'97)*, pp. 283–290, Association for Computational Linguistics, 1997.

[11] J. Goldstein, V. Mittal, J. Carbonell, and M. Kantrowitz, "Multi-document summarization by sentence extraction," in *2000 NAACL-ANLP Workshop on Automatic Summarization (NAACL-ANLP-AutoSum 2000), volume 4*, pp. 40–48, Association for Computational Linguistics, 2000.

[12] S. B. Silveira and A. Branco, "Extracting multi-document summaries with a double clustering approach," in *Natural Language Processing and Information Systems*, pp. 70–81, Springer, 2012.

[13] S. Sekine and E. Ranchhod, *Named Entities: Recognition, Classification and Use*. John Benjamins, 2009.

[14] L. L. Hill, "Core elements of digital gazetteers: placenames, categories, and footprints," in *Research and Advanced Technology for Digital Libraries*, pp. 280–290, Springer, 2000.

[15] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information Processing and Management*, vol. 24, no. 5, pp. 513–523, 1988.

[16] D. Grune and C. J. Jacobs, "Parsing techniques-a practical guide," 1990.

[17] L. Maicher and H. F. Witschel, "Merging of distributed topic maps based on the Subject Identity Measure (SIM) approach," *Proceedings of Berliner XML tags*, vol. 4, pp. 301–307, 2004.

[18] D. L. McGuinness, F. Van Harmelen, *et al.*, "OWL web ontology language overview," *W3C recommendation*, vol. 10, no. 2004-03, p. 10, 2004.

[19] S. Pepper, "The tao of topic maps," in *Proceedings of XML Europe*, vol. 3, 2000.

[20] S. B. Silveira and A. Branco, "Using a double clustering approach to build extractive multi-document summaries," in *Text, Speech and Dialogue*, pp. 298–305, Springer, 2012.

[21] R. Navigli, "Word sense disambiguation: A survey," *ACM Computing Surveys*, vol. 41, no. 2, p. 10, 2009.

[22] H. Dalianis, *SweSum: A Text Summerizer for Swedish*. KTH, 2000.

[23] X. Carreras, I. Chao, L. Padró, and M. Padró, "FreeLing: An open-source suite of language analyzers," in *Fourth International Conference on Language Resources and Evaluation (LREC'04)*, pp. 239–242, European Language Resources Association, 2004.

[24] P. Vossen, *EuroWordNet: A multilingual database with lexical semantic networks*. Kluwer Academic Boston, 1998.

[25] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *ACL Workshop on Text Summarization Branches Out (WAS'04)*, pp. 74–81, 2004.

[26] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to information retrieval*, vol. 1. Cambridge University Press, 2008.

Obtaining Knowledge from the Web using Fusion and Summarization Techniques

Sandra Escudero, Angel L. Garrido, Sergio Ilarri
IIS Department
University of Zaragoza
Zaragoza, Spain
*Email: {**sandra.escudero, garrido, silarri**}**@unizar.es***

***Abstract*—Nowadays, information and knowledge are fundamental in our society. This has induced an information overload problem in the Internet. For this reason, we propose to create an automatic system to retrieve, select, and extract information from the Web whose methodology is based on fusion techniques. The system, called Diana, facilitates and improves the identification of interesting contents, and it allowsto extract the most relevant information about a certain topic from the Web as a summary. To do this, we have developed algorithms that use semantic tools, Natural Language Processing (NLP) techniques, statistics, a generic gazetteer, and fusion methods. The development of the system is undergoing, but the preliminary results that we have obtained so far are very promising and show the interest of our proposal.**

***Keywords*-Summaries; fusion; text mining; NLP.**

I. Introduction

Currently, the number of web pages keeps growing at impressive rates. The explosion of written information generated through the Internet, such as blogs, social networks, or news, leads to the availability of a huge amount of text information. This makes it difficult to find what is really needed, since most of this information is hidden, messy, and unsorted. For this reason, proposals related to improving information and knowledge retrieval from the Web are receiving a great interest in the last years.

In this paper, we propose a multilingual system called Diana, motivated by the interest to create a program that allows to collect all the results that a web search returns, gathering the most relevant information correctly and showing it to the user in a compact and easy-to-consume format. With Diana, the user is able to have a global vision of the general contents of the set of web pages that the search engine has returned, without the need to perform a time-consuming exploration of the contents of each one separately.

Data fusion is defined as a "multilevel process that deals with the automatic detection, the association, the correlation, the estimation, and the combination of data and information from single and multiple sources" [1]. Information fusion systems deal with large quantities of heterogeneous data. The data fusion of multiples sources implies a set of techniques, similar to the cognitive process that a human performs, to integrate data of multiple sensors/observers in order to make inferences about the outside world. Therefore, data fusion aims at obtaining a better and precise overview of the relevant world.

The automatic integration of information from multiple and diverse sources is a central topic for many data fusion applications. In our system, we will have a very specific type of data items, which are documents (web pages) that contain textual information. So, in this paper we perform a data fusion of several sources of text-based information with the goal to provide a general overview of their contents through a suitable summarization process.

In our approach, on the one hand, the set of web pages obtained from the search are gathered, and they are converted to plain text in order to process their contents using Natural Language Processing (NLP) tools, semantic techniques, and statistics. On the other hand, we also carry out the same text extraction process but taking into account the HTML labels of this aforementioned text. The relevance of each sentence in the text is evaluated by using *four different approaches*, obtaining different values/scores for each of them. Three of these values are obtained from the analysis of the plain text, and the other one from the analysis of the HTML marks. These scores are then combined by using classical fusion techniques based on probabilistic logics. The final combined score is used to decide which sentences are the most relevant. Those sentences will be used to compose a summary that synthesizes the most important knowledge collected in the different web pages. Finally, this information is shown to the user.

In order to build the summary, Diana carries out a double clustering approach in order to process the sentences extracted from the web pages. The first clustering process ensures that the content is not repetitive, and the second clustering process assures the selection of the most relevant content, the preservation of the general concepts of the texts, and the organization of the final summary.

To produce highly-informative summaries, the summarization process also includes a simplification step at the end of its processing pipeline. This text simplification aims at clarifying natural language texts, by simplifying their sentences structurally into shorter and simpler ones, while keeping the meaning and information that the sentences contain. Both the simplification and the duplication removal are typical steps in a data fusion process [2]. Moreover, it is

important to remark that the user can configure the search of the information using a set of filters as, for example, setting the number of pages that the search will have to return, or the size of the summary performed.

To sum up, Diana is an application that allows to perform a summary of several web pages and present it to a user to provide him/her with a general overview of the results. The summary is created through a search of similar content among the information collected in all the web pages found, using fusion techniques. Diana also uses relevant attributes of the web pages (such as links and heading tags) to obtain a better summary, as certain HTML tags may contain relevant content that should be especially assessed for potential consideration as part of the summary.

Therefore, the main contribution of this paper is the development of a new methodology of information retrieval that uses NLP, semantics, statistics, and fusion techniques, to obtain informative summaries that contain the main knowledge stored in a set of web pages.

The rest of this paper is structured as follows. Section II is a brief description of the state of the art. Section III explains our proposed methodology. Section IV shows a working example. Finally, Section V provides our conclusions and explains our lines of future work on this topic.

II. Related Work

In this section, we briefly review some work related to information extraction from web pages and fusion for information retrieval.

A. Information Extraction from Web Pages

A web page usually contains several elements related to navigation, decoration, interaction and contact information, which are usually not relevant to a user's search. Therefore, detecting the content structure of a web page could potentially improve the performance of tasks related to information retrieval from web pages. HTML filtering tools allow us to extract only the relevant information. For example, the well-known PageRank [3] is a tool able to classify web pages objectively and effectively, in order to measure the interest of their content.

There are several proposals dedicated to information extraction from the Internet, such as for example [4], which presents a portal that provides a software agent configured to perform searches of web pages and summarize them. These summaries can be accessed by subscribers of the service. Users can specify their information needs by filling templates. The information obtained can be sent immediately to the subscriber or saved by the portal server for later retrieval.

It is interesting to mention that we can find many tools that automatically obtain summaries from web sources, such as for example: (i) summarization services of news in SMS (Short Message Service) messages or e-mail format for mobiles phones, (ii) searching services that obtain a summary automatically translated from a foreign language text, and (iii) search engines (like Google) that presents compressed descriptions of the search results.

Another interesting and important aspect is the readability in web summarization algorithms and the readability quality in real-time monitoring of search engines. In this sense, we can find proposals such as [5], which uses machine learning with a corpus of random queries and their corresponding search results. Then, the features of the summaries that are judged to potentially characterize readability are extracted. The authors try to obtain a model (gradient boosted decision tree) that predicts human judgments based on the features in order to avoid an alternative time-consuming solution where the quality of the summary is judged by a human. The model obtained can then be compared with other models of readability, such as the Collins-Thompson-Callan model [6].

Finally, we can find a work that also studies the use of abstracts in the context of the Web is [7]. This work uses hierarchies to provide a means of organizing, summarizing, and accessing correct information. This technique allows summarizing the documents retrieved by a search.

B. Fusion Applied to Information Retrieval

Data fusion has two main goals: increasing the integrity and boosting the conciseness of data. An enhance of integrity is achieved by adding more data sources to the system, and an increase in conciseness is reached by removing redundant data and ambiguities [2]. These goals are explained in works such as Motro [8] or Naumann [9]. In the context of data fusion, two main quality measures are considered: conciseness and completeness. These measures are analogous to the classical precision and recall measures used in the context of information retrieval. The results obtained should be complete, concise, and consistent. A result is complete if it contains all the relevant objects and all the relevant attributes present in the sources; it is concise if all the real-world objects and all the semantically-equivalent attributes present are described only once; finally, it is consistent if it contains all the tuples from the sources that are consistent regarding a specified set of integrity constraints (inclusion or functional dependencies) [10].

To conclude this section, it is interesting to mention two additional interesting systems that perform information retrieval over multiple documents. First, *SIMS* [11] is an information mediator that provides access and integration of multiples sources of information. It determines the relevant data sources and how to efficiently retrieve the desired information. Second, *Ariadne* [12] consists of wrappers for web sources and a central mediator which is responsible for query processing. The domain model and query language are the same as in SIMS, whereas Ariadne additionally includes binding patterns in the source descriptions.

Our methodology of information retrieval is different from the approaches mentioned previously, as we have merged NLP, semantics, statistics, and fusion techniques, to obtain summaries that contain the main knowledge stored in a set of web pages. Our proposal offers a configurable tool that allows to obtain a summary for each search engine selected from a query introduced by the user. With Diana the users can obtain easily a general idea about the topic searched.

III. Knowledge Fusion with Diana

The aim of our system Diana is to facilitate the identification of interesting contents available in the Internet. This system allows to extract the most relevant information about a certain topic in the Web in the form of a summary. We have developed an algorithm based on *the method of the three steps* proposed by F. Naumann [2]. First, we retrieve a set of web pages from the Internet using a search engine. Then, we extract from each web page a set of features that are considered useful for the data fusion process. Afterwards, we normalize the data obtained. Next, Diana proceeds to combine all the data obtaining a text summary of the contents as a final product. Finally, the summary goes through a simplification process, in which duplicates and ambiguities are eliminated, so that only the most relevant parts are extracted and the rest is discarded. In the rest of this section, we describe this process in detail.

A. Configuration and User Data Entry

Before performing a search, the user can configure the search engines on which the query is executed, as well as the number of web pages to be returned, when the default configuration is not desired. The default configuration in our prototype uses Google as the search engine, and as the number of web pages to return we have chosen 5 to simplify the results. After gathering all the settings, Diana shows a window where the user can introduce a set of keywords to perform the query. Diana launches the query over each of the selected search engines, searching in the Internet the aforementioned number of web pages with relevant content to the user's query. Finally, all those web pages are collected in a local repository for further analysis.

B. Extraction of Relevant Features

At this stage, Diana will extract all the features offered by the contents obtained in HTML format. First, the text in HTML must be transformed to legible text without tags. To do this, Diana uses a module that performs a filtering process which removes the tags and keeps the plain texts.

The text of each web page is divided into sentences to identify which of those sentences have more relevant information. Then, Diana scores each sentence. This score depends on the terms that compose the sentence. For this, Diana lemmatizes the text. The sentence scoring procedure defines the sentence relevance in the overall collection of sentences obtained from each input web page text. The overall score is computed as the sum of the `tf-idf` (Term Frequency - Inverse Document Frequency) scores [13] (it is obtained in turn by considering the word lemmas obtained previously) of each word of the sentences s ($tfidf_w$), smoothed by the number of words in the text of the Web page (*totw'*). With this metric, the relevance of a sentence not only depends on the frequency of the words present in it, but also on the number of texts in which the words appear. Equation 1 describes the computation of this score, where "w" refers to words that belong to the sentence "s":

$$score_s = \frac{\sum_w (tfidf_w)}{totw'} \quad (1)$$

The features that Diana obtains in this stage for each web page are: *named entities*, *keywords*, *synsets*, and *HTML content*. Named entities are fragments of text that represent names of persons, organizations, or locations [14]. For example, if we have the sentence "Peter drives a bus in New York", the named entities in this case are "Peter" and "New York", because the first one is a name of a person and the second one is a location. The keywords are the relevant words that are used to reveal the internal structure of a document. The synsets are provided by WordNet [15], that is a large lexical database. They are a set of cognitive synonyms expressing a distinct concept depending on the word given. The synsets are interlinked by means of conceptual, semantic and lexical relations. The synsets allow to reduce the number of relevant keywords of a sentence. An example of synset would be: *good*, *right* and *ripe*, where each word is most suitable or convenient than the other according to the context in which you are:

- "Now is a good time to go to the mountain"
- "Now is the right time to act"
- "The time is ripe to great changes"

Two analysis processes are performed to obtain the features mentioned above. The first one considers only the plain text of each web page. In this first analysis Diana detects named entities, synsets, and keywords. The second one uses the full information of the web pages, which includes both the plain text and the HTML tags referring to the format of those texts.

For each feature, a different scoring method is applied. The score for the keywords depends on the *tf-idf*, i.e., it depends on the frequency with which the term appears. The score for the named entities is computed in a similar way, but dividing the total score for the named entities by the total number of named entities. In the same way, the score obtained for the synsets is the sum of the score of each synset divided by the number of synsets contained in the text.

The scores based on the HTML tags are computed quite differently than the ones obtained for the other features. The

main idea is to give a score to each sentence according to the HTML format of its text. Headers, underscores, and italics receive a higher score than plain text. The size of the font used is also taken into account, as it is assumed that text shown more prominently on the web page is considered more important by the authors of the page. The weights of every text class can be configured by the user, allowing for more appropriate settings according to the context.
The extraction of each feature is performed as follows:

1) *Keywords* are identified by performing a morphological analysis to extract a list of important words from each text. Later, these words allow to identify which parts of the text "provide more knowledge" than others.
2) *Named Entities* are extracted using a morphological analysis with natural language processing tools and a gazetteer (to know if an entity is a place). We only take into account these kinds of elements in every sentence to obtain the score using the aforementioned statistical method `tf-idf`.
3) *Synsets* are obtained through a semantic analysis using a lexical database. So, if we know the synset of the words that compose the sentences, we can reduce the number of relevant keywords in order to perform a statistical analysis. This is due to the fact that some sets of words (for example: way, path, road, track and trail) correspond to the same synset.
4) *HTML Tags* such as images, tags, footers and headers, semantic tags (i.e., markup tags contained in an HTML file, to reinforce the semantics or meaning of data instead of just defining their presentation), hyperlinks, and comments, are extracted from each HTML file collected. These features are useful to know if the corresponding web page is related or not with the topic the user is interested in, as we rely on the assumption that everything that appears somehow highlighted in the web page is more relevant and therefore deserves a greater weight in its score.

C. Schema mapping

Once Diana has collected all the features of the web pages found, it proceeds to adjust the received data to a common measure. For this, the system normalizes all the scores obtained from the different features between 0 and 1. In this way the scores can be easily compared in the data fusion stage.

D. Data Fusion

In this stage, the fusion of all the data collected is performed, i.e., Diana unifies the multiple heterogeneous results obtained in the previous stage. The integration system responsible for carrying out this task combines the results, and then the combined output is shown to the user.

At this point, we use techniques related to data fusion based on statistics. We can see in Figure 1 a diagram that shows the process. To perform data fusion, each feature receives a modifier (weight) for its corresponding score before obtaining the unified final value.

Given our features, which are keywords (K), named entities (N), synsets (S) and HTML content (H), and their weights (wk, wn, ws, and wh, respectively), we can consider the general expression 2 to calculate the unified final value (V) of each sentence found.

$$V = \frac{(K * wk) + (N * wn) + (S * ws) + (H * wh)}{4} \quad (2)$$

Suitable values for the weights were obtained experimentally for our prototype, after testing it, so the context definitely plays an important role in order to tune the system. For example, if the scope of the search are news, the headlines are very important, because they capture much of the information contained in the text. In that case, the scores of the content of the HTML tags that belong to the header (h1, h2, ...) will be increased. In our tests, we have seen that when the scope is news, the scores of the headers can be multiplied up to 5, in order to acquire good results. As another example, in the scope of news we would also give significant weight to named entities that appear in the text, like important people or places.

Although the expression 2 shows an aggregation of values, it is important to remark that the aggregation is not the only data fusion action performed, as other aspects must also be considered, such as the data pre-processing, simplification and duplication removal, data alignment, and semantic interpretation.

When all the data have been combined, we have a set of sentences with a consolidated score. Then, we use an algorithm of automatic summarization [16] to obtain the final summary that contains the most relevant information found in all the web pages. This method consist of identifying which sentences are the most relevant ones according to their overall score.

Now, the aim is to identify relevant information in the collection of all the sentences contained in the web pages selected, and this is performed in two steps: a similarity clustering and a keyword clustering.

1) *Similarity clustering*. The similarity between two sentences is calculated considering the lemmas of the words. We obtain a similarity value taking account the subsequences of sentences and the overlapping words. If the similarity value is higher than the similarity threshold, then the sentences are grouped in the same cluster, to identify sentences conveying the same information. Finally, the sentence with the highest score from each cluster is selected to be used in the next step.

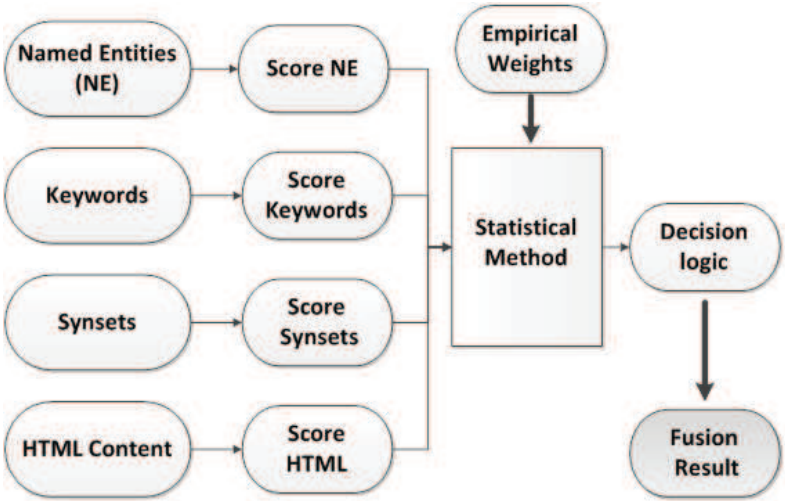

Figure 1. Process diagram

2) *Keyword clustering*. In this step, we have used an adapted version of the K-means algorithm to cluster sentences by keywords. As mentioned before, the input of this clustering process is composed by the sentences with the highest score selected in the previous clustering process. The keywords extracted previously represent the output clusters. A sentence is added to the cluster represented by the keyword that occurs more often in it. The sentences that do not contain any keyword are ignored.

After the two-step clustering process described above, Diana sorts the sentences based on their score, from the most relevant one to the less relevant one. Sentences at the top of the list are considered more significant than the others, since they address the main topics conveyed by the collection of sentences. At this point, the set of the most significant sentences constitutes already a summary of the text. Then, a syntactic and grammatical analysis of each sentence is performed, in order to identify the function of each word in the sentence and its type. For this purpose, we use an NLP tool. Afterwards, we carry on a simplification step in which duplicates and ambiguities are eliminated, so that only the most relevant parts are extracted and the rest is discarded. In this simplification step, we use an NLP tool too. Finally, we have a summary that it is composed by the most significant sentences.

The algorithm used in this section is motivated by its good performance in other applications such as TM-GEN [17], SOLE-R [18], and TMR [19]. For instance, in the first case, the algorithm was used to obtain a summary of multiple documents, and then used it to graphically represent knowledge through topic maps; in the second and in the third case, the applications are based on the use of TM-GEN, but in these cases the concept maps represent items to recommend. The concept maps are used too to model user profiles, and finally they are compared between them.

E. Summary of the Process

Summing up, the first thing that the user must do is to configure the system. Then, he/she launches the query. Next, Diana extracts all the features of the web documents obtained in the results using the selected search engines. Afterwards, it computes the score of each relevant feature, and the scores are normalized. Then, a data fusion is performed to combine the scores. Finally, it obtains a summary by selecting the most relevant sentences (the ones with the higher scores) of the web pages.

IV. A Complete Illustrative Example

The complete example we explain below has been performed with real users who have tested the system (see Figure 2 for a snapshot of the user interface of Diana). In our experiments, we have used Freeling [20] as NLP tool, both for morphological analysis as for syntactic analysis of the texts. As a lexical database we have used EuroWordNet [21].

To perform our experiments, we used a specific set of searches for our tests, which have helped us to prepare the system. This set is obtained by repeating the same questions all the time in the search engines and then analyzing the results (the summaries). 25% of the results of these searches have been used to train the system concerning the data fusion modifiers and the remaining 75% was used for real testing.

To carry out our examples, we have used several queries using different search engines. In the first test, the user introduced in Diana the search string “new mobile iphone 5”. The search engine found several web pages with relevant content. An example of the results obtained can be seen in Figure 3.

Figure 2. Diana interface

Many of the results obtained belong to the official website of Apple and the mobile company T-MOBILE. Diana allows us to choose among the selected search engines, and it shows us a first summary with the data obtained for each search engine. Then, we can access the results through a button whose name is "Summary".

In this figure, we can see the URL found, and part of the content of the web page. Moreover, we can access a little window where we can see the content of the page found, such as the title, the number of words that compose the page, the links, the anchor text along with its link, the keywords, and the matches, as can be seen in Figure 4. Of course, the user can access more related information by clicking on the corresponding link (for instance, in this case it points to a web shop). This information can be about the topic of the page (in this case, technology and communications). Moreover, if the user wants, he/she can export the result obtained.

If we launch the query "new mobile iphone 5", Diana returns the following summary:

The new iPhone 5 is in black or white with 16GB or 32GB. iPhone 5 is the thinnest, and lightest. The iPhone is so simple and intuitive, you don't need a manual. iPhone 5 features a 4-inch Retina display, ultrafast wireless and the powerful A6 chip.

In the second example, we have used Yahoo as the only search engine. The search introduced in Diana has been "stores new iphone". Now, the results also belong to the official web of Apple as well as to other web pages with information about the high demand for these mobile phones. For this query, Diana returns this summary:

The Apple Stores will start taking reservations for the iPhone 5s and 5c on September 17. Apple has announced. iPhone 5s went on sale in 10 countries on Friday, September 20. The new Iphone was initially available in the U.S. with shipping estimates of 1-3 days, however the gold iPhone 5s was quickly sold out, and the shipping estimates of all the other models slipped to 7-10 business days.

In the third example, the only search engine used is Bing. In this case, the string introduced has been "iphone new launch". The results obtained for this query are a little different from the other cases. For example, we can find web pages discussing the characteristics of the new iPhone, web pages with videos about its release, and also news about this. The summary obtained for this query is:

Apple is expected to unveil its next iPhone on Sept. 10. The WSJ's Deborah Kan and Yun-Hee Kim discuss what to expect at Apple's big event. The hoopla with the new Iphone will be big. On September 10, 2013, Apple unveiled two new iPhone models. As thousands stood in long-lines outside Apple stores, iPhone users eagerly switched to the new operating system. The iPhone 5S has a brand new A7 processor, that's 40 per cent faster. With its new A7 chip, iPhone 5s is the first smartphone with 64-bit technology, providing blazing-fast performance when launching apps, editing photos. The iPhone 5S, which starts at $199.

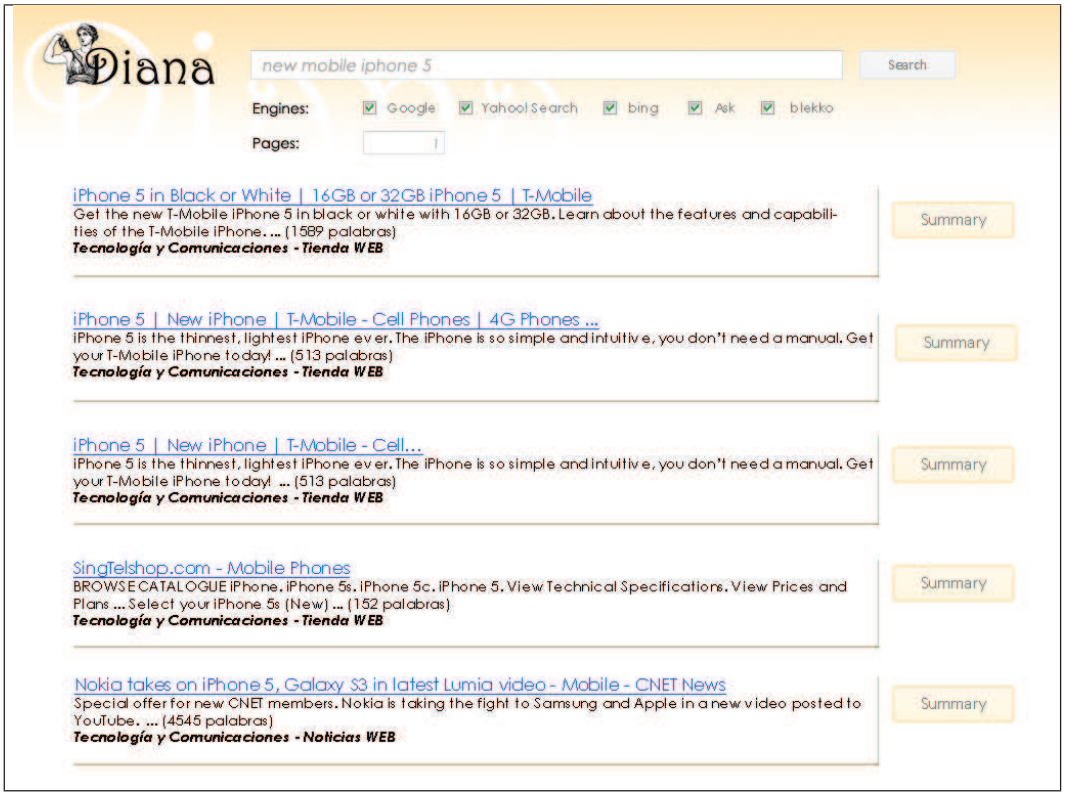

Figure 3. Results of the search

V. Conclusions and Future Work

In this paper, we have presented a multilingual system called Diana, that allows to obtain knowledge from a set of web pages using data fusion techniques with the help of a morphological analyzer, a lexical database, semantic tools, a syntactic parser, statistics methods, and a generic gazetteer. The proposal has been tested using well-known search engines.

Our main contributions are:

- The development of an algorithm to automatically obtain a summary from a set of web pages.
- The introduction of a new method to build summaries from a set of web documents using fusion techniques.

The possible scenarios where Diana can be used are diverse. For example, to improve the documental search engines, to facilitate tasks related with SEO (Search Engine Optimization), to increment the performance of Web browsers, or to automate and digitize the massive information. Finally, it may also be useful for end users.

The system is currently being tested and the results obtained so far are promising. However, this is still a seminal work and our next step will be to perform a more rigorous user testing to quantify the effectiveness of the solution in different contexts.

Another interesting idea for future work is to improve the interface showing the final summary, to look like a "report", combining texts and images of different websites found. In other words, the interface should be more friendly and intuitive for the user.

Acknowledgment

This research work has been supported by the CICYT project TIN2010-21387-C02-02 and DGA-FSE.

References

[1] E. Waltz, J. Llinas *et al.*, *Multisensor data fusion*. Artech house Boston, 1990, vol. 685.

[2] J. Bleiholder and F. Naumann, "Data fusion," *ACM Computing Surveys (CSUR)*, vol. 41, no. 1, p. 1, 2008.

[3] L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank citation ranking: Bringing order to the web." *Stanford InfoLab*, 1999.

[4] S. K. Inala, P. V. Rangan, R. Satyavolu, and S. P. Rajan, "Method and apparatus for obtaining and presenting web summaries to users," Dec. 14 2000, U.S. Patent App. 09/737,404.

Figure 4. Diana Result

[5] T. Kanungo and D. Orr, "Predicting the readability of short web summaries," in *Second ACM International Conference on Web Search and Data Mining*. ACM, 2009, pp. 202–211.

[6] M. Heilman, K. Collins-Thompson, and M. Eskenazi, "An analysis of statistical models and features for reading difficulty prediction," in *Third Workshop on Innovative Use of NLP for Building Educational Applications*. Association for Computational Linguistics, 2008, pp. 71–79.

[7] D. J. Lawrie and W. B. Croft, "Generating hierarchical summaries for web searches," in *Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*. ACM, 2003, pp. 457–458.

[8] A. Motro, "Completeness information and its application to query processing." in *Very Large Data Bases Conference*, vol. 86, 1986, pp. 170–178.

[9] F. Naumann, J.-C. Freytag, and U. Leser, "Completeness of integrated information sources," *Information Systems*, vol. 29, no. 7, pp. 583–615, 2004.

[10] A. Fuxman, E. Fazli, and R. J. Miller, "Conquer: Efficient management of inconsistent databases," in *Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data*. ACM, 2005, pp. 155–166.

[11] Y. Arens, C.-N. Hsu, and C. A. Knoblock, "Query processing in the sims information mediator," in *ARPA/Rome Laboratory Knowledge-Based Planning and Scheduling Initiative Workshop*, 1996.

[12] C. A. Knoblock, S. Minton, J. L. Ambite, N. Ashish, I. Muslea, A. G. Philpot, and S. Tejada, "The Ariadne approach to web-based information integration," *International Journal of Cooperative Information Systems*, vol. 10, no. 01n02, pp. 145–169, 2001.

[13] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information Processing and Management*, vol. 24, no. 5, pp. 513–523, 1988.

[14] S. Sekine and E. Ranchhod, *Named Entities: Recognition, Classification and Use*. John Benjamins, 2009.

[15] G. A. Miller, "WordNet: a lexical database for English," *Communications of ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[16] S. B. Silveira and A. Branco, "Extracting multi-document summaries with a double clustering approach," in *Natural Language Processing and Information Systems*. Springer, 2012, pp. 70–81.

[17] A. L. Garrido, M. G. Buey, S. Escudero, S. Ilarri, E. Mena, and S. B. Silveira, "TM-gen: A topic map generator from text documents," in *25th International Conference on Tools with Artificial Intelligence*. IEEE Computer Society, ISBN 978-1-4799-2971-9, ISSN 1082-3409, November 2013, pp. 735–740.

[18] A. L. Garrido, M. S. Pera, and S. Ilarri, "SOLER-R, a semantic and linguistic approach for Book Recommendations," in *14th IEEE International Conference on Advanced Learning Technologies - ICALT*, To be published in July 2014.

[19] A. L. Garrido and S. Ilarri, "TMR: A Semantic recommender system using topic maps on the items descriptions," in *11th European Conference of Web Semantic*, To be published in May 2014.

[20] X. Carreras, I. Chao, L. Padró, and M. Padró, "FreeLing: An open-source suite of language analyzers," in *Fourth International Conference on Language Resources and Evaluation (LREC'04)*. European Language Resources Association, 2004, pp. 239–242.

[21] P. Vossen, *EuroWordNet: A multilingual database with lexical semantic networks*. Kluwer Academic Boston, 1998.

Bibliografía

[1] Windows 7. `http://es.wikipedia.org/wiki/Windows_7`. Último acceso: 2 de Mayo de 2014.

[2] Windows 8. `http://es.wikipedia.org/wiki/Windows_8`. Último acceso: 2 de Mayo de 2014.

[3] Enrique Alcaraz Varó and María Antonieta Martínez Linares. *Diccionario de lingüística moderna*. Ariel, 1997.

[4] Tim Bray, Jean Paoli, C Michael Sperberg-McQueen, Eve Maler, and François Yergeau. Extensible Markup Language (XML). *World Wide Web Consortium Recommendation REC-xml-19980210. http://www.w3.org/TR/1998/REC-xml-19980210*, 1998.

[5] Xavier Carreras, Isaac Chao, Lluıs Padró, and Muntsa Padró. FreeLing: An open-source suite of language analyzers. In *Fourth International Conference on Language Resources and Evaluation (LREC'04)*, pages 239–242. European Language Resources Association, 2004.

[6] Sandra Escudero, Angel Luis Garrido, and Sergio Ilarri. Obtaining knowledge from the web using fusion and summarization techniques. In *17th International Conference on Information Fusion (FUSION 2014), Salamanca (España)*, To be published in July 2014.

[7] Angel Luis Garrido, Maria G. Buey, Sandra Escudero, Sergio Ilarri, Eduardo Mena, and Sara B. Silveira. Tm-gen: A topic map generator from text documents. In *25th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2013), Washington (USA), IEEE Computer Society*, pages 735–740, November 2013.

[8] Jade Goldstein, Mark Kantrowitz, Vibhu Mittal, and Jaime Carbonell. Summarizing text documents: sentence selection and evaluation metrics. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 121–128. ACM, 1999.

[9] Thomas R Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199–220, 1993.

[10] Paul Jaccard. Nouvelles recherches sur la distribution florale. volume 44, pages 223–270. Bulletin de la Sociéte Vaudense des Sciences Naturelles, 1908.

[11] K Sparck Jones et al. Automatic summarizing: factors and directions. *Advances in automatic text summarization*, pages 1–12, 1999.

[12] Video Lan. `http://www.videolan.org/vlc`. Último acceso: 23 de Mayo de 2014.

[13] Jesse Liberty. *Programming C#: Building. NET Applications with C#*. O'Reilly Media, Inc., 2005.

[14] Hans P Luhn. The automatic creation of literature abstracts. *IBM Journal of research and development*, 2(2):159–165, 1958.

[15] James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, page 14. University of California, USA, 1967.

[16] Inderjeet Mani. *Automatic summarization*, volume 3. John Benjamins Publishing, 2001.

[17] Inderjeet Mani and Mark T Maybury. *Advances in automatic text summarization*, volume 293. MIT Press, 1999.

[18] Visual Basic .Net. `http://es.wikipedia.org/wiki/Visual_Basic.NET`. Último acceso: 2 de Mayo de 2014.

[19] Steve Pepper and Graham Moore. XML Topic Maps (XTM) 1.0 - TopicMaps. org specification. *TopicMaps. Org Authoring Group, http://www.topicmaps.org/xtm*, 2001.

[20] Juan Ramos. Using Tf-idf to determine word relevance in document queries. In *Proceedings of the First Instructional Conference on Machine Learning*, 2003.

[21] Satoshi Sekine and Elisabete Ranchhod. *Named Entities: Recognition, Classification and Use*. John Benjamins, 2009.

[22] Chris Sells and Ian Griffiths. *Programming Windows presentation foundation*. O'Reilly Media, Inc., 2005.

[23] SQL Server. http://es.wikipedia.org/wiki/Microsoft_SQL_Server. Último acceso: 2 de Mayo de 2014.

[24] Sara Botelho Silveira and António Branco. Extracting multi-document summaries with a double clustering approach. In *Natural Language Processing and Information Systems*, pages 70–81. Springer, 2012.

[25] Karen Spärck Jones. Automatic summarising: The state of the art. *Information Processing & Management*, 43(6):1449–1481, 2007.

[26] Visual Studio. http://es.wikipedia.org/wiki/Microsoft_Visual_Studio. Último acceso: 2 de Mayo de 2014.

[27] M Wilson. Migrating from thesauri to ontologies. *Available in: http://www. w3c. rl. ac. uk/ukofficepasttalksindex. html*, 2002.

[28] Windows XP. http://es.wikipedia.org/wiki/Windows_XP. Último acceso: 2 de Mayo de 2014.