



Departamento de Informática

Escuela Politécnica Superior
Universidad Carlos III de Madrid

INGENIERÍA EN INFORMÁTICA

PROYECTO FIN DE CARRERA

Estudio de la Utilización de Etiquetas en Aplicaciones para Buscar y Compartir Información

Autor

Fco. Javier González Ortega

Tutor

José Luis Martínez Fernández

12 de noviembre de 2008

Esta página se dejó intencionadamente en blanco

TÍTULO: Estudio de la Utilización de Etiquetas en Aplicaciones para Buscar y Compartir Información.

AUTOR: Francisco Javier González Ortega

TUTOR: Dr. D. José Luis Martínez Fernández

La defensa del presente proyecto se realizó el día de Noviembre de 2008; siendo calificada por el siguiente tribunal:

Presidente:

Secretario:

Vocal:

Habiendo obtenido la siguiente calificación:

Calificación:

Presidente

Secretario

Vocal

Esta página se dejó intencionadamente en blanco

Agradecimientos

Quisiera dedicar unas palabras a todos aquellos que han contribuido a que este proyecto se llevase a cabo. Primeramente, debo agradecer a mi tutor José Luis Martínez su dedicación, paciencia, amabilidad y consejos, que han permitido que este proyecto adquiriera forma y enfoque, y llegara finalmente a buen puerto, manteniendo siempre una sonrisa.

Quiero también agradecer a mi familia su paciencia y atención en los interminables años de carrera y más interminable si cabe año de proyecto: a mis padres, a los que ahora puedo devolver algo de lo que me han dado tantos años; a mis abuelos, por alegrarse generosamente por la más mínima ventura propia; y a mi hermana, tan generosa como para enseñar la aplicación a sus amigos como si de ella misma fuera. Y por supuesto también a Hámster (DEP) y a Cobayo, por su suavidad.

Quisiera también agradecer a los muchos amigos hechos durante estos cinco años su presencia, compañía y ayuda: Susana, Fraga, Dani, Irene, Chema, Johanna... Y de entre ellos, a Daniel Higuero, por lo mucho que he aprendido de él, su incomprensible interés por muchas cuitas surgidas durante el proyecto, y en general por estar siempre disponible para la conversación aleatoria; Juan Manuel Tirado, por la fraternidad creada en el progresivo, la chorrada, y la adversidad ante los sacos de yeso; y Julio Ayala, por estar siempre ahí.

A todos los conocidos y desconocidos que se pasaron por la aplicación del proyecto y dedicaron algunos de sus minutos a hacer alguna de sus tareas: muchas gracias, me hicisteis un favor.

Y en un apartado misceláneo, a esas cosas que hacen del mundo un lugar mejor: el Lateralus de Tool, Cthulhu, Dilbert, Lemmy, Terry Pratchett, Dethklok, The B-52's, Mr. T, el black/death/doom metal, rafabasa.com, y acabar una carrera.

“They sentenced me to twenty years of boredom”

Leonard Cohen, *First we Take Manhattan*

“Then when my tedious hours have past”

My Dying Bride, *Black God*

*“If there were no reward to reap (...),
This tedious path I've chosen here,
I certainly would've walked away by now”*

Tool, *The Patient*

Esta página se dejó intencionadamente en blanco

Resumen

En los últimos años, estamos asistiendo a un gran cambio en el modo en que Internet se relaciona con el mundo. Cada vez más popular, Internet ha dejado de limitarse a mostrar contenidos fijos para convertirse en un medio para la comunicación de los individuos. Una enorme cantidad de contenido web ha comenzado a ser generado a través de tecnologías sociales como weblogs, wikis o foros, tecnologías que permiten acceder a la comunicación y publicación de contenidos en web muy fácilmente, limitando cada vez más los conocimientos necesarios para su uso.

Estos nuevos patrones en el modo en que la información se genera y multiplica hacen necesarios métodos de clasificación de información, capaces de manejar y organizar estas cantidades. Los métodos de clasificación tradicionales usados hasta ahora (p.e. sistemas jerárquicos) muestran limitaciones al enfrentarse a este conjunto creciente de contenidos, debido entre otros motivos a estar controlados por una autoridad central.

Frente a ellos, un nuevo sistema de clasificación ha surgido al amparo precisamente de Internet, a raíz de su introducción en sitios web como del.icio.us o Flickr: las *folksonomías*. En ellas, es una comunidad de usuarios la encargada de aumentar los contenidos del sistema, añadiendo cada usuario recursos, y describiéndolos con una serie de *etiquetas*, esto es, palabras sueltas elegidas libremente, sin limitaciones de vocabularios controlados o de otro tipo.

Este proyecto dará una panorámica pormenorizada sobre variados aspectos relacionados con esta nueva tecnología, y de entre ellos, se hará especial énfasis en el estudio de las ideas de mejora aplicadas hasta el momento; después de todo, las folksonomías son aún en este momento una idea en desarrollo, alimentada y por muy diversos estudios académicos y sistemas web reales.

En este proyecto, hemos querido estudiar estas posibilidades de mejora, centrándonos en una: la recomendación automática de etiquetas. Se ha buscado comprobar hasta qué punto estos sistemas resultan de utilidad a los usuarios, si son utilizados, y qué provecho obtienen de ellos.

Para ello, se ha desarrollado una aplicación web publicada para su libre uso, en que diversas herramientas de recomendación son ofrecidas a los usuarios, encargados de ejecutar diversas tareas relacionadas con el etiquetado y búsqueda de imágenes. Los datos de su interacción son recogidos y estudiados, obteniendo diversas conclusiones acerca de su comportamiento general en el manejo de estos sistemas con ayuda de herramientas de recomendación.

Esta página se dejó intencionadamente en blanco

Abstract

In the last years, a great change has occurred in the way in which Internet is used. More and more popular, Internet has stopped limiting itself to show fixed contents, to truly become a mean for human communication. An enormous amount of Web content has begun being generated through social technologies like weblogs, wikis or forums, technologies that allow an easy communication and publication of contents in Web, limiting more and more the knowledge needed for its use.

These new patrons in the way in which the information is generated make necessary new methods for the classification of information, able to handle and to organize these large amounts. The traditional methods of classification used until now (v.g. hierarchic systems) show limitations when facing this increasing set of contents, for the reason (among others) of being controlled by a central authority.

In front of them, a new classification system has arisen, as a result of its introduction in Web sites like del.icio.us or Flickr: the *folksonomies*. In them, a community of users is in charge of increasing the contents of the system, each user adding resources, and describing them with a series of *tags*, a set of words chosen freely, without limitations of controlled vocabularies.

This project will give a general view on varied aspects related to this new technology, and among them, special emphasis on the different ideas for folksonomy improvement applied until the moment; after all, the folksonomies are still a developing idea, fed by very diverse academic studies and real Web systems.

In this project, we have tried to study these possibilities of improvement, focusing on one: the automatic recommendation of tags. We have tried to check to what extent these systems are useful to the users, whether they are used, and what benefit do they obtain from them.

For this reason, it has been developed a web application, which is published for its public use, in which diverse tools of recommendation are offered to the users, who are asked to execute diverse tasks related to the labelling and search of images. The data of their interaction are picked up and studied, obtaining diverse conclusions about the general behavior of users in the interaction with these systems with the help of recommendation tools.

Esta página se dejó intencionadamente en blanco

Índice de contenido

1.Introducción.....	1
1.1.Motivación.....	1
1.2.Objetivos.....	2
1.3.Resumen de contenidos.....	3
2.Estado del arte.....	5
2.1.Fundamentos.....	5
2.1.1.Software social.....	5
2.1.2.Clasificación de información.....	6
2.1.2.1.Jerarquías.....	7
2.1.2.2.Facetetas.....	8
2.2.Folksonomías.....	9
2.2.1.Características de las folksonomías.....	10
2.2.2.Concepto de etiqueta o tag.....	11
2.2.3.Ejemplos.....	12
2.2.3.1.Del.icio.us.....	12
2.2.3.2.Flickr.....	13
2.2.3.3.Gmail.....	15
2.2.3.4.Google Image Labeler: ESP Game.....	16
2.3.Propiedades de las folksonomías.....	17
2.3.1.Tipos de folksonomías.....	18
2.3.2.Motivación para el etiquetado.....	20
2.3.3.Ventajas.....	21
2.3.4.Defectos.....	22
2.3.5.Representación gráfica: nubes de etiquetas.....	23
2.3.6.Patrones comunes en el etiquetado.....	25
2.3.6.1.Ley de potencia.....	25
2.3.6.2.Distribuciones en las folksonomías.....	26
2.3.6.3.Análisis de co-ocurrencia.....	28
2.3.6.4.Características de las etiquetas utilizadas.....	29
2.3.6.4.1.Temática de las etiquetas.....	29
2.3.6.4.2.Tipos de etiquetas.....	30
2.3.6.4.3.Variaciones en las etiquetas.....	31
2.3.6.5.Otras propiedades.....	32
2.4.Mejora de las folksonomías.....	32
2.4.1.Educación de los usuarios.....	32
2.4.2.Mecanismos automáticos.....	33
2.4.2.1.Agrupaciones.....	33
2.4.2.2.Creación de estructuras jerárquicas o facetadas.....	36
2.4.2.3.Recomendación de etiquetas.....	38
2.4.2.4.Mejora en las búsquedas.....	39
3.Descripción del experimento.....	42
3.1.Antecedentes.....	43
3.2.Objetivo del experimento.....	44
3.3.Colección de imágenes.....	45
3.4.Ayudas al usuario.....	46

3.5.Tareas.....	46
3.6.Información guardada.....	48
3.7.Conclusiones a obtener.....	48
3.8.Recorrido por la aplicación.....	49
3.8.1.Pantalla de login.....	49
3.8.2.Tareas de añadir imágenes.....	50
3.8.3.Tareas de búsqueda de imágenes.....	54
4.Desarrollo de la aplicación.....	58
4.1.Extracción de imágenes.....	58
4.1.1.Extracción de imágenes en Flickr.....	60
4.2.Desarrollo de la aplicación: Arquitectura.....	61
4.3.Componente: Interfaz de Usuario.....	63
4.3.1.Funcionalidad.....	63
4.3.2.Tecnología: Ajax.....	63
4.3.2.1.Google Web Toolkit (GWT).....	65
4.3.2.2.Extensiones de GWT: MyGWT.....	66
4.4.Componente: Gestor de Acciones.....	68
4.4.1.Funcionalidad.....	68
4.4.2.Interfaces.....	68
4.5.Componente: Almacenamiento y Recuperación.....	73
4.5.1.Funcionalidad.....	73
4.5.2.Interfaces.....	74
4.5.3.Tecnología: FileUpload.....	77
4.5.4.Funcionamiento: Ayuda de sugerencias para autocompletado.....	77
4.5.5.Funcionamiento: Ayuda de etiquetas relacionadas.....	77
4.6.Componente: Motor de Búsquedas.....	78
4.6.1.Funcionalidad.....	78
4.6.2.Interfaces.....	79
4.6.3.Tecnología: Lucene.....	80
4.7.Componente: Traducción de Idiomas.....	81
4.7.1.Funcionalidad.....	81
4.7.2.Interfaces.....	81
4.7.3.Tecnología: Traductor CLEF.....	82
4.8.Componente: Imágenes Parecidas.....	83
4.8.1.Funcionalidad.....	83
4.8.2.Interfaces.....	83
4.8.3.Recuperación de imágenes basada en su contenido.....	85
4.8.4.Tecnología: GIFT.....	85
4.8.4.1.Funcionamiento.....	87
4.9.Componente: Almacenamiento.....	88
4.9.1.Funcionalidad.....	88
4.9.2.Interfaces.....	88
4.9.2.1.Interfaz IAccesoDatosGuardado.....	88
4.9.2.2.Interfaz IAccesoDatosConsulta.....	91
4.9.2.3.Interfaz IAccesoDatosLogs.....	94
4.9.3.Tecnología: Log4J.....	95
4.9.4.Tecnología: MySQL.....	96

4.9.5. Modelos de datos utilizados.....	96
4.9.5.1. Esquema de base de datos.....	96
4.9.5.2. Estructura de los logs de usuario.....	98
5. Resultados del experimento.....	104
5.1. Tareas de añadido de imágenes (de 1 a 4).....	106
5.2. Tareas de búsqueda (de 5 a 7).....	107
5.3. Comentarios de los usuarios.....	111
6. Conclusiones y líneas futuras.....	113
6.1. Resultados del experimento.....	113
6.2. Desarrollo de la aplicación.....	113
6.3. Ejecución del proyecto.....	114
6.4. Líneas futuras.....	114
7. Bibliografía.....	116

Esta página se dejó intencionadamente en blanco

Índice de tablas

Tabla 1: Interfaz ServiciosEtiquetado – método borrarClave.....	69
Tabla 2: Interfaz ServiciosEtiquetado – método buscarImágenes.....	69
Tabla 3: Interfaz ServiciosEtiquetado – método comenzarPrueba.....	70
Tabla 4: Interfaz ServiciosEtiquetado – método estaSesionIniciada.....	70
Tabla 5: Interfaz ServiciosEtiquetado – método guardaImagen.....	70
Tabla 6: Interfaz ServiciosEtiquetado – método haSeleccionadoImagen.....	70
Tabla 7: Interfaz ServiciosEtiquetado – método haVistoImagen.....	71
Tabla 8: Interfaz ServiciosEtiquetado – método imagenAleatoriaEscogida.....	71
Tabla 9: Interfaz ServiciosEtiquetado – método login.....	71
Tabla 10: Interfaz ServiciosEtiquetado – método obtenerEtiquetasRelacionadas.....	71
Tabla 11: Interfaz ServiciosEtiquetado – método obtenerImagenAleatoria.....	72
Tabla 12: Interfaz ServiciosEtiquetado – método obtenerImagenClave.....	72
Tabla 13: Interfaz ServiciosEtiquetado – método obtenerImágenesRelacionadas.....	72
Tabla 14: Interfaz ServiciosEtiquetado – método obtenerSugerencias.....	72
Tabla 15: Interfaz ServiciosEtiquetado – método obtenerTraducciones.....	73
Tabla 16: Interfaz ServiciosEtiquetado – método terminarPruebas.....	73
Tabla 17: Interfaz IAlmacenamientoRecuperacion – método borrarClaveImagen.....	74
Tabla 18: Interfaz IAlmacenamientoRecuperacion – método conseguirImagenAleatoria.....	75
Tabla 19: Interfaz IAlmacenamientoRecuperacion – método conseguirImágenes.....	75
Tabla 20: Interfaz IAlmacenamientoRecuperacion – método crearClusterTags.....	75
Tabla 21: Interfaz IAlmacenamientoRecuperacion – método crearUsuario.....	75
Tabla 22: Interfaz IAlmacenamientoRecuperacion – método existeYaUsuario.....	76
Tabla 23: Interfaz IAlmacenamientoRecuperacion – método guardarImagen.....	76
Tabla 24: Interfaz IAlmacenamientoRecuperacion – método guardarImagen.....	76
Tabla 25: Interfaz IAlmacenamientoRecuperacion – método obtenerSugerencias.....	76
Tabla 26: Interfaz IAlmacenamientoRecuperacion – método obtenerTagsRelacionados.....	77
Tabla 27: Interfaz IMotorBusquedas – método anadirImagenIndice.....	79
Tabla 28: Interfaz IMotorBusquedas – método buscarTags.....	80
Tabla 29: Interfaz IMotorBusquedas – método cierraIndice.....	80
Tabla 30: Interfaz IMotorBusquedas – método iniciarIndice.....	80
Tabla 31: Interfaz ITraduccionIdiomas – método obtenerTraducciones.....	82
Tabla 32: Interfaz IimagenesParecidas – método compararImagen.....	84
Tabla 33: Interfaz IimagenesParecidas – método conectarServicioImágenes.....	84
Tabla 34: Interfaz IAccesoDatosGuardado - método almacenarGrafoTags.....	89
Tabla 35: Interfaz IAccesoDatosGuardado - método anadirImagen.....	90
Tabla 36: Interfaz IAccesoDatosGuardado - método anadirImagenEtiquetada.....	90
Tabla 37: Interfaz IAccesoDatosGuardado - método borrarClaveImagen.....	90
Tabla 38: Interfaz IAccesoDatosGuardado - método crearUsuario.....	91
Tabla 39: Interfaz IAccesoDatosGuardado - método tomarClaveNuevaImagen.....	91
Tabla 40: Interfaz IAccesoDatosConsulta - método conseguirClaveAleatoria.....	92
Tabla 41: Interfaz IAccesoDatosConsulta - método conseguirImágenes.....	92
Tabla 42: Interfaz IAccesoDatosConsulta - método existeYaUsuario.....	93
Tabla 43: Interfaz IAccesoDatosConsulta - método obtenerListadosTagsImagen.....	93
Tabla 44: Interfaz IAccesoDatosConsulta - método obtenerTagsImagen.....	93
Tabla 45: Interfaz IAccesoDatosConsulta - método obtenerTagsRelacionados.....	93

Tabla 46: Interfaz IAccesoDatosConsulta - método obtenerTagsSubstring.....	94
Tabla 47: Interfaz IAccesoLogs - método prepararLoggerGeneral.....	95
Tabla 48: Interfaz IAccesoLogs - método prepararLoggerUsuario.....	95
Tabla 49: Interfaz IAccesoLogs - método prepararLoggerXMLUsuario.....	95

Índice de ilustraciones

Ilustración 1: Ejemplo de página con clasificación basada en jerarquía.....	7
Ilustración 2: Ejemplo de navegación sobre clasificación facetada.....	9
Ilustración 3: Vídeo de Youtube.....	10
Ilustración 4: Añadido de enlace en del.icio.us.....	12
Ilustración 5: Búsquedas en Flickr.....	14
Ilustración 6: Detalles de una imagen en Flickr.....	14
Ilustración 7: Etiquetas en Gmail.....	15
Ilustración 8: Google Image Labeler.....	17
Ilustración 9: Nube de tags creada por last.fm para describir la banda musical "Tool".....	24
Ilustración 10: Flickr Related Tag Browsing.....	24
Ilustración 11: Aspecto de la gráfica de una relación de ley potencial.....	26
Ilustración 12: Frecuencia de cada etiqueta en Flickr. Gráfica log-log: eje horizontal con las etiquetas ordenadas de mayor a menor frecuencia, eje vertical con la frecuencia de cada etiqueta. .	27
Ilustración 13: Número de etiquetas por imagen en Flickr. Gráfica log-log: eje horizontal con las imágenes ordenadas de mayor a menor nº de etiquetas, eje vertical con nº de etiquetas para cada imagen.....	27
Ilustración 14: Grafos MDS de co-ocurrencia para las URL "www.yousendit.com" y "www.bellybytes.com/articles/29foods.shtml", respectivamente.....	28
Ilustración 15: Distribución de categorías de WordNet para las etiquetas de Flickr.....	30
Ilustración 16: Clusters generados en Flickr para fotos con la etiqueta 'apple'.....	34
Ilustración 17: Ejemplo del resultado del algoritmo de agrupado.....	35
Ilustración 18: Creación de jerarquías personales de etiquetas en Rawsugar.....	37
Ilustración 19: Facetas creadas a partir de etiquetas en Rawsugar.....	38
Ilustración 20: Separación de imágenes etiquetadas como "apple" según su contenido gráfico.....	40
Ilustración 21: Pantalla de inicio de la aplicación.....	49
Ilustración 22: Enunciado de la tarea 1.....	50
Ilustración 23: El botón "Mostrar Enunciado" muestra el enunciado de la prueba actual.....	51
Ilustración 24: El botón "Ayudas" muestra textos de ayuda sobre la aplicación.....	51
Ilustración 25: Interfaz para la primera tarea.....	52
Ilustración 26: Tabla para añadir etiquetas. A la izquierda, tabla con dos etiquetas añadidas. A la derecha, inserción de una nueva etiqueta, siendo sugeridas algunas posibilidades.....	52
Ilustración 27: Ayuda de Imágenes Parecidas.....	53
Ilustración 28: Ayuda de Etiquetas Relacionadas.....	53
Ilustración 29: Selector de fichero.....	54
Ilustración 30: Ayuda de traducción automática.....	54
Ilustración 31: Búsqueda de imágenes.....	55
Ilustración 32: Detalles de imagen.....	55
Ilustración 33: Autocompletado en el cuadro de búsquedas.....	56
Ilustración 34: Ayudas de traducción y etiquetas relacionadas en tareas de búsqueda.....	56
Ilustración 35: Botón "Buscar parecidas".....	57
Ilustración 36: Botón "Abandonar Prueba".....	57
Ilustración 37: Cuadro de comentarios al fin del experimento.....	57
Ilustración 38: Esquema de la estructura XML.....	58
Ilustración 39: Arquitectura de la aplicación.....	62
Ilustración 40: Demo Explorer de MyGWT.....	67

Ilustración 41: Interfaz ServiciosEtiquetado.....	69
Ilustración 42: Interfaz IAlmacenamientoRecuperacion.....	74
Ilustración 43: Interfaz IMotorBusquedas.....	79
Ilustración 44: Interfaz ITraduccionIdiomas.....	82
Ilustración 45: Interfaz IImageneParecidas.....	84
Ilustración 46: Interfaz PHP de demostración de GIFT.....	86
Ilustración 47: Resultados de una consulta sobre GIFT.....	87
Ilustración 48: Interfaz IAccesoDatosGuardado.....	89
Ilustración 49: Interfaz IAccesoDatosConsulta.....	92
Ilustración 50: Interfaz IAccesoLogs.....	94
Ilustración 51: Esquema de base de datos.....	97
Ilustración 52: Estructura del xml de datos de interacción: elemento Pruebas.....	98
Ilustración 53: Estructura del xml de datos de interacción: elemento Prueba.....	99
Ilustración 54: Estructura del xml de datos de interacción: elementos ImagenAleatoriaEscogida, SeleccionadaImagen, VistaImagen, Abandono y VistaAyuda.....	100
Ilustración 55: Estructura del xml de datos de interacción: elemento Resultado.....	101
Ilustración 56: Estructura del xml de datos de interacción: elemento Búsqueda.....	101
Ilustración 57: Estructura del xml de datos de interacción: elemento Ayuda.....	102
Ilustración 58: Estructura del xml de datos de interacción: elemento AyudaEtiquetas.....	102
Ilustración 59: Estructura del xml de datos de interacción: elemento AyudaIdiomas.....	102
Ilustración 60: Estructura del xml de datos de interacción: elemento AyudaImagene.....	103
Ilustración 61: Estructura del xml de datos de interacción: elemento AyudaSugerencia.....	103
Ilustración 62: Número de usuarios que completó cada tarea.....	104
Ilustración 63: Tiempo medio de ejecución de cada tarea.....	105
Ilustración 64: Origen de las etiquetas utilizadas, en promedio.....	106
Ilustración 65: Número medio de llamadas a las ayudas en las tareas de búsqueda.....	108
Ilustración 66: N° de búsquedas textuales ejecutadas, y n° de palabras utilizadas en las mismas, en promedio.....	109
Ilustración 67: Origen de las palabras utilizadas en búsquedas, en promedio.....	110
Ilustración 68: Porcentaje de imágenes seleccionadas a partir de resultados obtenidos con cada ayuda.....	111

1. Introducción

Compleja tarea la de la clasificación de la información, si bien muy necesaria. Mediante ella, trata de darse estructura a una larga serie de contenidos, que permita organizarlos y dar una panorámica general de la información en ellos contenida, facilitando el acceso a la misma. Su dificultad radica en el modo en que esta estructura es creada: qué propiedades de los documentos se tendrán en consideración, en base a qué, cómo se agruparán... una larga serie de preguntas que finalmente desemboca en una larga serie de variables para esta clasificación.

Un ejemplo al respecto es el conocido artículo de Jorge Luis Borges “*El idioma analítico de John Wilkins*”, recogido en su libro *Otras Inquisiciones*, donde pondera las propiedades (o, por mejor decir, las arbitrariedades) de varios esquemas urdidos para la clasificación general de todos los contenidos, entre ellos el del Instituto Bibliográfico de Bruselas y el de un apócrifo enciclopedista chino, del que se decía:

“En sus remotas paginas está escrito que los animales se dividen en (a) pertenecientes al Emperador, (b) embalsamados, (c) amaestrados, (d) lechones, (e) sirenas, (f) fabulosos, (g) perros sueltos, (h) incluidos en esta clasificación, (i) que se agitan como locos, (j) innumerables, (k) dibujados con un pincel finísimo de pelo de camello, (l) etcétera, (m) que acaban de romper el jarrón, (n) que de lejos parecen moscas.”

El pasaje es evidentemente cómico, pero la conclusión es desesperanzada: “(...) notoriamente no hay clasificación del universo que no sea arbitraria y conjetural. La razón es muy simple: no sabemos qué cosa es el universo”. Pero después de todo, como él mismo comentaba poco después, “la imposibilidad de penetrar el esquema divino del universo no puede, sin embargo, disuadirnos de planear esquemas humanos, aunque nos conste que éstos son provisorios”.

Así pues, a pesar de sospechar que ningún sistema de clasificación podrá alcanzar el status de perfecto, pues ninguno podrá ser incuestionablemente válido en toda circunstancia y contexto, seguiremos buscando esquemas lo suficientemente adecuados como para aportarnos una estructura que nos pueda servir de guía en la exploración del conocimiento que nos es disponible.

Esencialmente ésa será la tarea de este proyecto, centrándonos en uno de los sistemas de clasificación de información con mayor crecimiento en los últimos años, debido a su expansión en Internet: las *folksonomías*.

1.1. Motivación

En los últimos años, y debido a la cada vez mayor disponibilidad de conexión a Internet de banda ancha, Internet ha sufrido una creciente popularización, que la han convertido más allá de mero muestrario de contenidos en un verdadero medio para la interacción y comunicación entre individuos; fenómenos como el de la Web Social lo corroboran. Esta popularización ha llevado a un gran aumento de la cantidad de contenido sobre Internet, posibilitando cada vez más que cualquier

individuo, tenga o no conocimientos informáticos, pueda contribuir a ello y aportar sus propios contenidos, sea a través de weblogs, wikis, foros o cualquier otro sistema.

Este aumento de la cantidad de información ha llevado a la necesidad de nuevos sistemas de clasificación de información, que permitan organizar toda esta masa dispersa de datos buscando obtener una panorámica general de los mismos, y sean capaces de manejar y organizar tales cantidades. Los métodos de clasificación tradicionales usados hasta ahora (p.e. sistemas jerárquicos) muestran limitaciones al enfrentarse a este conjunto creciente de contenidos, debido entre otros motivos a estar controlados por una autoridad central.

Las folksonomías representan un sistema novedoso de clasificación de información, surgido de forma espontánea en páginas web como Flickr o del.icio.us, y posteriormente desarrollado y analizado por numerosas otras páginas y variados estudios. En ellas, es una comunidad de usuarios la encargada de aumentar los contenidos del sistema, añadiendo cada usuario recursos, y describiéndolos con una serie de *etiquetas*, esto es, palabras sueltas elegidas libremente, sin limitaciones de vocabularios controlados o de otro tipo.

El estudio de las folksonomías resulta de interés por todo lo que tiene de cambio de perspectiva dentro de la clasificación de información. Dentro de un ámbito regido por vocabularios controlados y rígidas taxonomías formales, el confiar en una masa de usuarios para que a su libre albedrío añadan etiquetas sin ningún tipo de control formal, resulta particularmente revolucionario, y sorprendente que realmente los resultados conseguidos sean de utilidad.

Es además un campo en pleno desarrollo y expansión: las folksonomías surgieron hace pocos años, y constantemente nuevos estudios y páginas web aportan pequeñas mejoras o sugerencias para una experiencia más rica por parte de los usuarios. Es por ello un campo rico y con mucho material para la investigación, e interesante también por lo que tiene de uso generalizado: estos sistemas se utilizan muy comúnmente para la organización de contenidos en páginas dedicadas a compartir recursos por una comunidad de usuarios, y es tan fácil verlos en sistemas de puro entretenimiento para compartir vídeos o imágenes, como para usos más serios como organización de documentación.

En este proyecto, hemos querido estudiar estas posibilidades de mejora de las folksonomías, centrándonos en una: la recomendación automática de etiquetas. Se ha buscado comprobar hasta qué punto estos sistemas resultan de utilidad a los usuarios, si son utilizados, y qué provecho obtienen de ellos.

1.2. Objetivos

El proyecto se centra en el desarrollo y ejecución de un experimento sobre un sistema de recomendación de etiquetas para folksonomías, y la evaluación de su uso por parte de los usuarios. Para aclarar el cometido del proyecto y formalizar las tareas a ejecutar en el mismo, se planteó un listado de objetivos parciales para el completado del proyecto:

- **Estudio sobre las folksonomías, estado actual, características y posibles mejoras.**

Como primer paso a completar, sería necesario familiarizarse con la tecnología de las folksonomías: en qué se basa y cómo funciona, su implantación actual en Internet u otros medios, sus motivaciones, características, etcétera. Sería importante también estudiar diversos aspectos de las mismas susceptibles de mejora, así como propuestas o experimentos de mejora que hayan podido surgir.

- **Plantear un experimento sobre el manejo de las folksonomías por usuarios.**

Partiendo del estudio previo realizado, deberá plantearse un experimento de estudio sobre el modo en que los usuarios utilizan las folksonomías. Este experimento se centrará en la recomendación automática de etiquetas, buscando descubrir el modo en que los usuarios utilizan estas ayudas. Deberán detallarse las características del experimento, tareas a ser ejecutadas, datos a ser recogidos, resultados a comprobar, etcétera.

- **Desarrollo de la aplicación del experimento.**

Planteadas formalmente las características del experimento a ejecutar, se desarrollará la aplicación que dé soporte a tal experimento. Serán requisitos básicos de tal aplicación ajustarse a la descripción obtenida en el objetivo anterior, tanto en pruebas a ejecutar como en datos a recoger. Por otra parte, se buscará maximizar la usabilidad de su interfaz, tratando que esta sea intuitiva y manejable para los usuarios.

- **Pruebas del experimento con usuarios reales y evaluación de resultados.**

Una vez diseñada y desarrollada la aplicación, se deberá enviar a una serie de usuarios reales, que deberán ejecutar la serie de tareas que en ella les sean solicitadas. Se tomarán datos de cada una de las ejecuciones, que deberán ser estudiados según los criterios establecidos inicialmente en la definición del experimento, sacando las conclusiones que de ellos se deriven.

1.3. Resumen de contenidos

La estructura de este documento, junto con el propósito de cada una de sus secciones, es la que sigue:

- **Capítulo 1: Introducción.** Este capítulo. Explica brevemente la idea del proyecto, y aporta un encuadre general al mismo, estableciendo motivaciones y objetivos, que deberán servir de guía durante su ejecución.
- **Capítulo 2: Estado del arte.** Se describe el estado actual de desarrollo del ámbito técnico en que se enmarca el proyecto, el estudio de las denominadas *folksonomías*. Se aporta una definición de las mismas, con un estudio de los motivos de su surgimiento, ventajas, desventajas, ejemplos reales, y patrones de utilización. Finalmente, se detalla una serie de estudios para su mejora, tratando de dar una panorámica sobre las diferentes líneas de investigación actuales en el desarrollo de la folksonomías.

- **Capítulo 3: Descripción del experimento.** Se define el experimento que se planteará en el proyecto, para el estudio de la utilidad de las recomendaciones de etiquetas en el uso de una folksonomía. Se dan las pautas de las pruebas a ejecutar por una serie de usuarios, indicando los objetivos que guían el experimento, las tesis que se busca probar o refutar, las características de las pruebas, el entorno en que se realizarán, la información que se obtendrá de ellas y cómo se procesará.
- **Capítulo 4: Desarrollo de la aplicación.** Se aportan detalles técnicos sobre la aplicación desarrollada en el proyecto para la ejecución del experimento descrito en el capítulo anterior. Se aporta una descripción de alto nivel de la misma, detallando su arquitectura y componentes, la función de los mismos y el modo en que se conectan, las tecnologías utilizadas para su desarrollo, y en general el proceso llevado a cabo para su implementación.
- **Capítulo 5: Resultados del experimento.** Ejecutadas las pruebas con un conjunto de usuarios reales, se detallará en este capítulo la información recogida de su interacción, explicando aquellos aspectos de mayor interés en vista de los objetivos que guiaron las pruebas, y comprobando si finalmente las recomendaciones de etiquetas resultan de utilidad a los usuarios y en qué medida.
- **Capítulo 6: Conclusiones y líneas futuras.** En este capítulo conclusivo se tratará de evaluar el conjunto total del proyecto, recopilando las lecciones y experiencias aprendidas, buscando mostrar de forma resumida a los posibles lectores lo conseguido en el proyecto y lo que pueda convenir evitar en proyectos futuros. Se plantearán también una serie de líneas futuras, esto es, ideas de interés a explorar en futuros proyectos para completar el trabajo hecho en éste, que puedan enriquecer su estudio o aportar más información sobre campos relacionados.

2. Estado del arte

En esta sección ofreceremos una perspectiva sobre el estado actual de desarrollo de las *folksonomías*: sistemas de clasificación de información de forma social, mediante la asignación libre de etiquetas a recursos por parte de una comunidad de usuarios. Describiremos de forma concisa y clara sus fundamentos y características, detallando de igual modo los aspectos susceptibles de mejora y las líneas de investigación actuales, que servirán de motivación y base para el experimento llevado a cabo en este proyecto.

2.1. Fundamentos

Antes de la aparición de las folksonomías, existieron determinados fenómenos y necesidades que motivaron su surgimiento. En este apartado explicaremos unos de los factores más importantes, como fueron el software social y las redes sociales sobre Internet, que permitieron la creación de sistemas colaborativos formados por comunidades de usuarios, y los mecanismos de clasificación de información previos a su aparición, cuyas dificultades para adaptarse a un medio tan vasto y cambiante como es la información guardada en Internet hacían necesaria otra alternativa.

2.1.1. Software social

El software social engloba a un conjunto de herramientas de comunicación que permiten y facilitan la comunicación entre personas por medio de ordenadores. No se refiere tanto a un tipo concreto de software como al hecho de la utilización de una tecnología o combinación de tecnologías informáticas para la comunicación y formación de comunidades de usuarios.

Es sobretodo a través de Internet que este tipo de software se ha desarrollado y expandido, muy especialmente con la llegada del fenómeno Web 2.0, término acuñado para referirse a la tendencia de Internet de convertirse en una herramienta colaborativa y de comunicación entre sus usuarios, más allá de un mero escaparate de contenidos.

Entre estos sistemas, son los basados en *redes sociales* aquellos que más han tratado de mostrar el mundo social de sus usuarios sobre Internet. La filosofía de estas redes sociales se basa en conectar a personas con sus conocidos, familiares y amigos, que a su vez conectan a esa misma red a personas de su entorno. Para ello, los usuarios crean cuentas personales, con opciones para mostrar un perfil personal según los gustos del usuario y utilizar los diversos servicios que ofrezca la página, y son invitados a que agreguen a otras cuentas como “amigos” si lo consideran conveniente: de este modo, se hace evidente la estructura social de la comunidad. Estas páginas suelen tener un carácter eminentemente lúdico y suelen construirse sobre una determinada comunidad de intereses, para la cual ofrecen servicios relacionados: por ejemplo, Myspace, una de las más importantes redes sociales de la actualidad, tiene una importante base de bandas musicales emergentes, para lo que ofrece mecanismos para colgar sus canciones en sus perfiles. Otros ejemplos serían last.fm, comunidad de usuarios que muestran y comparten sus gustos musicales, o

una de las primeras creadas, classmates.com, que trataba de conectar a los usuarios con sus viejos compañeros de colegio.

Otros diferentes servicios surgidos para la comunicación entre usuarios son los foros, las wikis, los weblogs o la mensajería instantánea, que permiten una interacción cercana entre individuos, sea por mero componente lúdico o colaborando entre ellos para la consecución de alguna tarea. Sitios web que aglutinaban a comunidades de usuarios, colaborando unos con otros mediante la inclusión de recursos de información, para que otros pudiesen buscarlos y utilizarlos, fueron uno de los principales motores para el surgimiento de las folksonomías, como explicaremos más adelante.

2.1.2. Clasificación de información

Desde que el ser humano ha sido capaz de generar información y almacenarla, ha necesitado también de medios que le permitiesen organizarla y recuperarla cuando de ella tuviese necesidad. Métodos de clasificación y organización fueron estableciéndose para tales tareas, que permitiesen dar una imagen esquematizada del conocimiento disponible. El modo en que se organiza una biblioteca, dividiendo y organizando los libros por materias y autores, puede ser un buen ejemplo de esto.

En los últimos años, la cantidad de información a la que podemos tener acceso ha aumentado enormemente debido a Internet, y ha comenzado a ser creada con mucha mayor facilidad debido a tecnologías sociales como weblogs, wikis o foros. Estas tecnologías permiten acceder a la comunicación y publicación de contenidos en web muy fácilmente, limitando cada vez más los conocimientos tecnológicos necesarios para su uso. Estos nuevos patrones en el modo en que la información se genera y multiplica hacen necesarios métodos de organización y búsqueda, capaces de manejar estas cantidades.

Desde la aparición de los buscadores de Internet como Yahoo o Google, la recuperación de información por búsqueda textual ha sido el método más común para la obtención de información en este medio. Sin embargo, este medio no debe ser entendido como el único válido, al existir tareas y objetivos que no son cubiertos adecuadamente por el mismo.

En la clasificación de información, es de importancia la capacidad de agregar la información existente, permitiendo al usuario obtener una visión general del conocimiento presente. Dos serían las tareas básicas: la búsqueda de información concreta, y la exploración sobre un determinado área; los buscadores, centrados en esta primera tarea, no resuelven adecuadamente la segunda. La búsqueda implica el previo conocimiento de lo buscado, no requerido en la exploración, en la que el usuario pide al sistema recursos de un determinado área, o relacionados con otro recurso, tratando de construir una panorámica sobre ese aspecto.

Tradicionalmente, han existido dos esquemas básicos de clasificación de la información, basados en un acercamiento diferente:

- **Jerárquico**: método de organización que procede de arriba abajo, estableciendo un conjunto de categorías, subdivididas a su vez en otras categorías de ámbito cada vez más estrecho, en las cuales se encajan los elementos.
- **Analítico (facetas)**: método de organización que procede de abajo arriba, describiendo una serie de propiedades relevantes en cada uno de los elementos que la componen, y estableciendo medios para organizar y sintetizar los elementos a partir de tales valores.

Describimos estos acercamientos en los siguientes subapartados.

2.1.2.1. Jerarquías

El esquema más natural e intuitivo para la clasificación de elementos, donde el conocimiento se divide en categorías progresivamente más limitadas y específicas. Las categorías construidas se organizan en un esquema con forma de árbol, donde una categoría puede contener varias otras, pero pertenecer a sólo una categoría mayor. Este esquema es especialmente útil para ordenar objetos físicos y suele utilizarse en bibliotecas, por ejemplo, en las que los libros se colocan en un único lugar, con usualmente un solo camino en un esquema de categorías para llegar a él.

CATEGORÍAS

 [Todos los sitios](#) |  [Sólo sitios de España](#)

- [Actores y actrices](#) (593)
- [Animación y cómics](#) (431)
- [Artes escénicas@](#)
- [Centros de ocio](#) (5)
- [Chistes y humor](#) (241)
- [Ciencia ficción, fantasía y terror](#) (44)
- [Cine](#) (1273)
- [Comedia](#) (5)
- [Concursos de belleza](#) (21)
- [Concursos, encuestas y sondeos](#) (24)
- [Directorios y guías](#) (68)
- [Electrónica de consumo](#) (123)
- [Empleo](#) (2)
- [Escaparates](#) (1)
- [Eventos](#) (25)
- [Genial](#) (343)
- [Juegos@](#)
- [Literatura@](#)
- [Magia e ilusionismo](#) (38)
- [Medios de comunicación](#) (74)
- [Música](#) (4692)
- [Organizaciones](#) (5)
- [Parques de atracciones@](#)
- [Postales digitales](#) (103)
- [Radio@](#)
- [Recintos para espectáculos](#) (8)
- [Tauromaquia](#) (76)
- [Teatro@](#)
- [Televisión@](#)
- [Venta de entradas@](#)

Ilustración 1: Ejemplo de página con clasificación basada en jerarquía

Este sistema tiene la ventaja de ser fácilmente comprensible por los usuarios, modelando bien contextos de límites claros y más o menos estables en el tiempo. Su utilización ha sido común en páginas de Internet tales como directorios; en la ilustración puede verse las categorías raíz del directorio de Yahoo España [1].

Sin embargo, tiene una serie de desventajas asociadas, que han hecho que este modelo sea de aplicación cada vez más reducida para la clasificación de información sobre Internet. Primeramente, la estructura general carece de flexibilidad: una vez establecida, un cambio en las categorías resulta costoso, pues requiere estudiar cómo deberían moverse el resto de elementos ya categorizado.

Por otra parte, la creación de la estructura es un proceso centralizado, decidido por una autoridad central, lo cual choca con el hecho de que la elección de categorías es siempre un proceso subjetivo: si dos personas modelasen un mismo dominio pudieran llegar a muy distintas clasificaciones, dependiendo de los aspectos de los elementos que considerasen de relevancia. Esto implica que no siempre la búsqueda dentro de una estructura jerárquica será intuitiva: el usuario puede no entender la elección de categorías, o incluso buscar un elemento que en tal estructura pudiera encuadrarse en dos categorías diferentes.

Y finalmente, es importante notar el difícil mantenimiento del sistema. Se requieren usuarios entrenados para la inclusión de nuevos elementos dentro de la estructura, analizando sus características y estudiando en qué categoría encajarlos, y revisar, mover y actualizar los existentes en caso de que sea necesario. Por otra parte, si el número de elementos va a aumentar con el tiempo, cambios periódicos en la estructura serían necesarios, para equilibrar el árbol de categorías (quizá subdividiendo una categoría demasiado poblada, o uniendo dos categorías casi vacías), tarea costosa como ya se ha dicho.

2.1.2.2. *Facetas*

Los sistemas facetados, comparados con los jerárquicos, resuelven varios de sus problemas y son en general más adaptables a conjuntos de elementos amplios y variables. En estos sistemas, la clasificación de la información se desarrolla mediante la determinación de varios campos o aspectos relevantes de los objetos a clasificar, denominados *facetas*. Estas facetas serán atributos de los elementos del sistema, y tomarán un valor determinado para cada uno de ellos. Así por ejemplo, en un sistema de clasificación de canciones, podrían definirse las siguientes facetas:

- **Género:** *Pop, rock, folk, música clásica...*
- **Grupo**
- **Álbum al que pertenece**
- **Duración...**

Etcétera. Como puede verse, cada faceta representaría una propiedad de la canción, para la que debería indicarse el valor correspondiente al introducir un nuevo elemento al sistema.

De este modo, no se define una única jerarquía de elementos, sino que los elementos se clasifican implícitamente evaluando sus características, pudiendo ser recuperados por los usuarios usando esas mismas propiedades: un usuario podría decidir ver inicialmente el listado completo de canciones de género “rock”, y filtrarlo posteriormente por su duración. Estos esquemas resuelven los problemas anteriormente identificados, y son especialmente recomendables para sistemas en los que el *corpus* de elementos es muy numeroso. Sugieren un acercamiento explorativo, mediante el cual un conjunto de datos grande es filtrado según las elecciones del usuario.

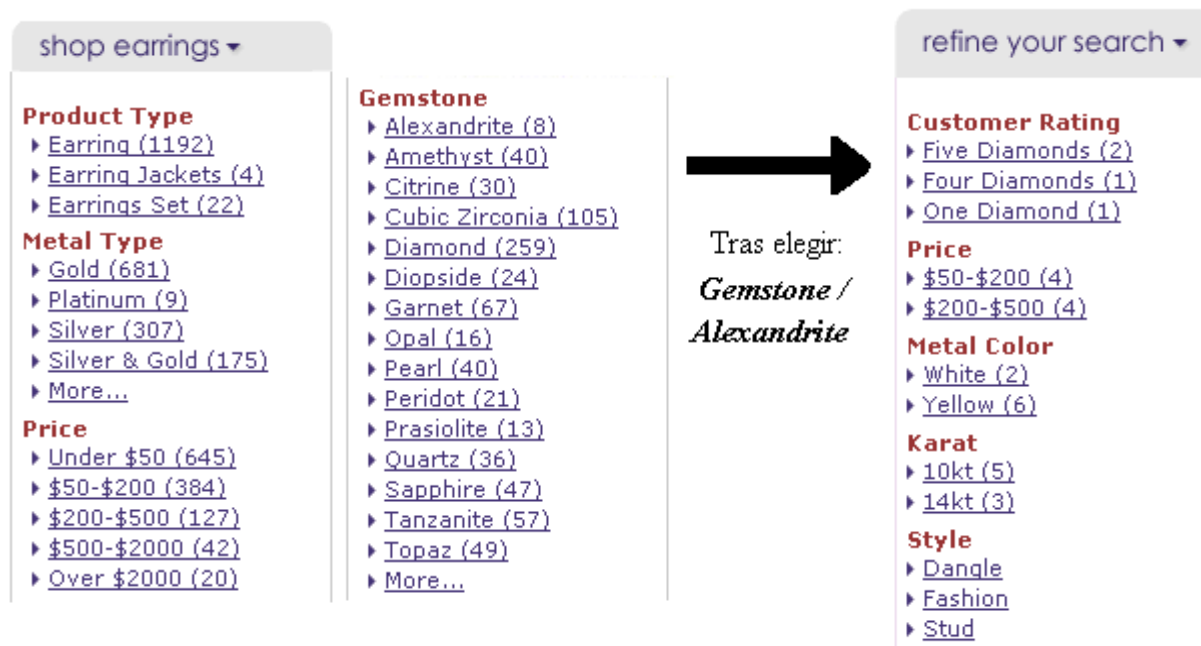


Ilustración 2: Ejemplo de navegación sobre clasificación facetada.

La ilustración anterior muestra cómo se muestra esta clasificación facetada en una página de Internet dedicada a la venta de pendientes [2]. La parte izquierda de la ilustración muestra una parte de la columna izquierda de la página, que se presenta al usuario para que pueda iniciar la exploración de la colección de elementos. Cuando éste selecciona un valor para una faceta (en la ilustración “*gemstone: alexandrite*”), se le mostrará el listado de elementos con ese valor para tal faceta, permitiéndole filtrar aún más el listado escogiendo valores para otras facetas.

Como se ha dicho, este sistema es mucho más adaptable que el jerárquico. No obstante, sigue presentando problemas cuando el número de elementos crece muy rápidamente: las facetas se definen para un determinado dominio, y el constante aumento del tamaño del sistema puede hacer que su significado se diluya. Además, sigue requiriendo del despliegue de recursos humanos, que continuamente eliminen, desplacen, actualicen, y revisen los recursos existentes. Normalmente, al inicio de un proyecto de este tipo, existen recursos humanos suficientes para suplir estas tareas de mantenimiento, estableciendo perfectamente las facetas y sus contenidos; sin embargo, según el tamaño del sistema crece, el mantenimiento manual se convierte en una labor tediosa e ineficiente.

2.2. Folksonomías

Frente a estos acercamientos básicos, ha surgido en los últimos años un nuevo paradigma para la clasificación colectiva de contenidos por parte de una comunidad de usuarios, que ha sido ampliamente aceptado y está sufriendo un gran crecimiento sobre Internet: son las folksonomías.

Se denomina *folksonomía* a la utilización de etiquetas para la clasificación de un conjunto de contenidos por una comunidad de usuarios. No nacen de una teoría o estudio previo, sino que surgieron y crecieron al ser introducidas sus bases en páginas web como Delicio.us, Flickr, 43 Things, Technorati, Furl, y varias otras. De hecho, el término *folksonomía* fue acuñado cuando el fenómeno ya había surgido y crecido: suele ser atribuido a Thomas Vander Wal, neologismo que mezcla las palabras *folk* (gente) y *taxonomy* (taxonomía), queriendo enfatizar de este modo su naturaleza de clasificación abierta, creada por el conjunto de usuarios.

2.2.1. Características de las folksonomías

Para precisar una definición formal, un modelo formal del concepto de folksonomía puede encontrarse en [3]. Se define una folksonomía como una tupla $F := (U, T, R, Y)$, donde U , T y R son conjuntos finitos cuyos elementos se denominan *usuarios*, *etiquetas* y *recursos*, respectivamente; Y es una relación ternaria definida sobre ellos ($Y \subseteq U \times T \times R$), cuyos elementos se denominan *asignaciones de etiqueta*.

Emperor-The Loss And Curse Of Reverence



De: **ecstasyindecay**
 Antigüedad: hace 2 años
 Vídeos: 4 [Suscribir](#)

Añadido: **05 de diciembre de 2005** [\(Menos información\)](#)
 This video comes from the Emperial Live Ceremony DVD.
 Categoría: **Música**
 Etiquetas: emperor black metal norway mysticism extreme
 ihsahn zyklon

URL:
http://es.youtube.com/watch?v=L_D3XqFgPSQ

Insertar: [Personalizar](#)
 <object width="425" height="344"><param name="movie" value="http:

► Más de: **ecstasyindecay**
 ▼ Vídeos relacionados

Ilustración 3: Vídeo de Youtube

En la ilustración de arriba puede verse la captura de un vídeo cualquiera en Youtube, página basada en una folksonomía. Youtube es una página para la clasificación de un conjunto de recursos, en este caso vídeos, por un conjunto de usuarios que se registran libremente. Pueden verse ejemplificados en la ilustración, que muestra la descripción de un vídeo en concreto, todos los elementos del sistema: el vídeo sería el recurso en cuestión, subido por un determinado usuario (en este caso, “ecstasyindecay”), que utiliza una serie de etiquetas para definirlo (“emperor black metal norway (...”). Estas etiquetas, como se ve, son simples palabras sueltas elegidas libremente por el usuario, no limitadas a un vocabulario u otro tipo de control. Estas etiquetas servirán posteriormente a otros usuarios para encontrar el vídeo en sus búsquedas.

Varias son las particularidades de este sistema, que lo diferencian de los sistemas de clasificación clásicos. La principal es el libre etiquetado: los usuarios están encargados de asignar a cada recurso del sistema una serie de etiquetas elegidas libremente, no restringidas a ningún

vocabulario. Una etiqueta consiste de este modo en una palabra (o conjunto de palabras, si se permiten etiquetas compuestas) cualquiera, que el usuario puede asignar a un recurso sin (en principio) restricciones en cuanto a número, de 0 a N etiquetas por recurso y usuario.

Es ésta una clasificación directamente generada por los usuarios, que carecen de una autoridad central que guíe este proceso de clasificación. De igual modo que en los sistemas facetados, la clasificación se crea desde los recursos, describiendo sus propiedades mediante etiquetas, que serán tomadas por los procesos de búsqueda y navegación para obtener visiones generales, de un modo escalable (esto es, la adición de contenidos no debe dificultar estos procesos). Los contenidos pueden ser de muy diverso tipo, siendo válido prácticamente cualquiera que pueda ser representado digitalmente: existen folksonomías dedicadas a imágenes, música, vídeos, entradas de blog, enlaces web, y un largo etcétera.

Así pues, las folksonomías necesitan de una comunidad de usuarios para su funcionamiento, encargados de aumentar el contenido del sistema y de su descripción. Esta relación de comunidad se refuerza por el sistema mediante diferentes modos de acceder a recursos relacionados: suele permitirse ver otros contenidos publicados por un cierto usuario, o por otros usuarios con gustos similares a los nuestros, o simplemente contenidos relacionados. Una folksonomía ideal debería ser accesible con un vocabulario originado por, y familiar a, sus usuarios primarios.

2.2.2. Concepto de etiqueta o tag

Como hemos dicho, una etiqueta (del inglés *tag*) es simplemente una palabra (o conjunto de palabras, en caso de que se permitan etiquetas compuestas) que el usuario asigna a un recurso. No existen reglas en su selección más que aquellas que el usuario considere oportunas.

Una etiqueta establece una relación entre un recurso y un concepto mental del usuario, expresada como una palabra. De este modo, la unión entre la foto de un árbol y el concepto “árbol” en la mente del usuario, se conseguirá mediante la etiqueta *árbol*. En otros casos, el usuario podrá etiquetar un elemento con información que él no contiene de forma obvia, pero sí despierta en el usuario; las etiquetas *genial*, *gracioso* o *debe_verse* expresan unos sentimientos del usuario, que sin embargo otro usuario podría no compartir.

Una etiqueta representa también un modo de focalizar la atención. Nos expresa aquello por lo que un recurso ha sido considerado útil, y nos invita a considerar si es útil para nosotros en ese mismo sentido. Las etiquetas son un método muy eficiente de distribución de la atención en un contexto de abundancia excesiva de información.

Sin embargo, aparte de esta asociación entre un concepto y el contenido que el usuario puede conseguir, no existe un modo directo de encontrar la semántica de una etiqueta. Debido a ello, uno de los principales problemas que deben controlarse en las folksonomías es el de la sinonimia (etiquetas diferentes con un mismo significado) y el de la homonimia (etiquetas iguales con diferente significado), que pueden llevar a uniones inapropiadas de elementos. Así por ejemplo, son en principio indistinguibles (aunque se han propuesto métodos para evitarlo, que se explicarán en secciones posteriores) usos de la etiqueta “rock” para referirse a un estilo musical o a una roca, o de la etiqueta “apple” para referirse a una manzana o a los ordenadores Macintosh.

De algún modo, pueden entenderse como un modo en que los sistemas facetados se desintegran y degradan, permitiéndose la inclusión de valores y facetas cada vez más laxos, perdiendo requisitos formales y de semántica en pos de facilidad y simplicidad en su manejo. El rigor necesario para la elección correcta de valores en las facetas no es necesario, y el usuario puede actuar a su antojo, añadiendo etiquetas sin necesidad de procesos mentales complejos.

2.2.3. Ejemplos

Mostraremos algunos ejemplos de sistemas reales de folksonomías, centrándonos para cada uno de ellos en un aspecto en concreto del comportamiento de la folksonomía, con la intención de aclarar lo máximo posible el concepto y ofrecer una primera panorámica del mismo.

2.2.3.1. *Del.icio.us*

Del.icio.us es una página para el guardado de marcadores (*social bookmarking*). Permite a los usuarios guardar sus marcadores (enlaces favoritos a páginas web) en la página, como lo harían en su explorador web, y consultarlos así desde cualquier ordenador. Estos marcadores se acompañan de una descripción y varias etiquetas, y pueden darse a conocer al resto de usuarios.

Mostraremos el modo en que se incluyen en esta página los marcadores de los usuarios:



del.icio.us

url

description

notes

tags

recommended tags
export **food** guide search

your network
for:joshua for:jwhiting

popular tags
cooking **food** **reference** recipes thesaurus Dictionary cook

Ilustración 4: Añadido de enlace en del.icio.us

En la ilustración podemos ver la pantalla mediante la cual se pueden añadir enlaces a del.icio.us, que muestra con bastante claridad los aspectos básicos de este tipo de interfaces. Los campos que pueden verse en esta pantalla son:

- **URL:** Dirección web donde se guarda la página a almacenar en marcadores. En este contexto, representa el recurso a etiquetar.
- **Description:** Nombre asignado al marcador. Automáticamente, se rellena con el título de la página, pero esto puede cambiarse.
- **Notes:** Campo opcional donde puede darse una descripción de la página, o cualquier otro contenido, pensamiento u opinión que el usuario considere necesario. Limitado a 255 caracteres.
- **Tags:** Etiquetas asociadas al marcador.

Pueden indicarse tantas etiquetas como se consideren necesarias, separadas por espacios (no se permiten etiquetas compuestas). Se muestran aquí los elementos más básicos de la interfaz: aparte del recurso en sí y el listado de etiquetas, es habitual la inclusión de breves resúmenes en forma de título y descripción.

Más abajo, pueden verse tres listas de etiquetas recomendadas, que ayudan a su selección:

- **Popular tags:** Etiquetas que otras personas han utilizado para etiquetar este enlace.
- **Your network:** Del.icio.us permite al usuario establecer una serie de usuarios vecinos. En esta lista, se dan posibles etiquetas con los que el usuario puede hacer indicar a sus vecinos que el contenido de este recurso les interesa (en formato *for:<vecino>*).
- **Recommended tags:** Selección de etiquetas basada en otras que el mismo usuario ha utilizado habitualmente, u otros usuarios han utilizado para etiquetar esta página.

Dentro de las pantallas de etiquetado de otras páginas basadas en folksonomías, estos listados no siempre se presentan, o en el caso de que lo hagan no tienen por qué ser los mismos. Son de utilidad para obtener de los usuarios un etiquetado de calidad. Existen variados mecanismos para la recomendación de etiquetas, que explicaremos más adelante.

2.2.3.2. Flickr

Flickr es una página web que ofrece servicios para compartir fotos entre la comunidad de usuarios. Esta página permite a sus miembros enviar sus fotos y hacerlas públicas para el resto de la comunidad, dando servicio para su clasificación y búsqueda. Flickr ha sido adoptado por muchos usuarios de Internet como almacenamiento de imágenes para su uso *online*, especialmente para los autores de weblogs.

Para esta página, detallaremos los principales modos de navegación que pueden utilizarse para explorar su contenido, que nos mostrarán los diferentes esquemas según los que se conectan los contenidos de un sistema folksonómico:



Ilustración 5: Búsquedas en Flickr

En la ilustración anterior, podemos ver el modo más básico para la navegación sobre los contenidos de Flickr: su buscador. Es ésta una herramienta básica en todo sistema folksonómico, como lo es en tantas otras páginas web: supone la primera puerta de entrada para el usuario a los contenidos del sistema, a los que accede incluyendo unas pocas palabras significativas sueltas.

Entre las opciones especiales que ofrece Flickr para estas búsquedas, están las de buscar en todos los textos asociados a una imagen (título, descripción y etiquetas) o limitarse a las etiquetas asignadas a la imagen. De igual modo, se permite la búsqueda de grupos de usuarios o usuarios en concreto. Búsquedas avanzadas permiten afinar más el tipo de contenido buscado.

Se nos presentará un listado con las imágenes relevantes para nuestra búsqueda, mostrando a su lado datos relevantes como título, usuario que la ha añadido o principales etiquetas utilizadas. Haciendo click sobre una de estas imágenes, llegaremos a su pantalla de descripción.

Björk's gusset

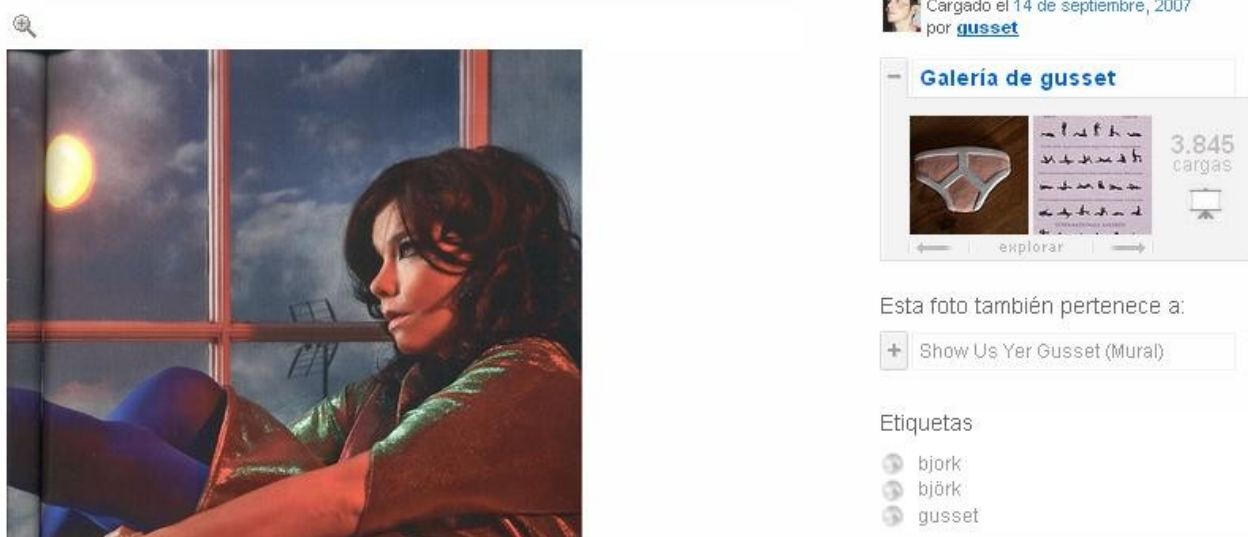


Ilustración 6: Detalles de una imagen en Flickr

En esta pantalla, se ofrece muy diversa información sobre la imagen, que la conecta a otros contenidos del sistema. Como puede verse en el margen derecho, se muestra el usuario que añadió la imagen y enlaces a su galería. Se incluyen también enlaces a colecciones de fotografías que incluyen la imagen, y a las etiquetas utilizadas. La navegación es intuitiva pues todo está conectado: haciendo click en el nombre de un usuario podremos llegar a su perfil, haciendo click sobre una etiqueta podremos ver todas las imágenes etiquetadas por ese usuario con esa etiqueta.

Flickr ofrece también una gran variedad de ayudas a la exploración que tratan de hacer la experiencia del usuario más grata y divertida. Se ofrece una selección de las imágenes más interesantes de los últimos días en <http://www.flickr.com/explore/interesting/7days/>, selecciones de fotos según el lugar del mundo donde hayan sido tomadas (relacionado con el *geotagging*: un formato acordado para la inclusión de etiquetas que representen coordenadas geográficas, para permitir su localización en el mundo) y como es habitual en los sitios folksonómicos, páginas para explorar las últimas imágenes añadidas o las etiquetas más populares. También, se incluye soporte para otros modos de acceso, como puedan ser los lectores de RSS.

2.2.3.3. Gmail

Gmail es el servidor de correo de Google. Es un servicio gratuito de webmail, que también permite interacción mediante POP3. Este servicio fue especialmente conocido por su gran capacidad del almacenamiento en el momento de salida al mercado, con 2 GB totales, y por el uso de programación en Ajax para su interfaz.

Entre las diferentes características de Gmail, se encuentra un potente mecanismo para la organización del correo, basado en etiquetas. Esta clasificación no está destinada a la búsqueda social, en contraste con las folksonomías, pues las cuentas de correo son de uso eminentemente personal; sin embargo, son un método muy eficiente para la clasificación flexible orientada a futuras búsquedas del usuario.

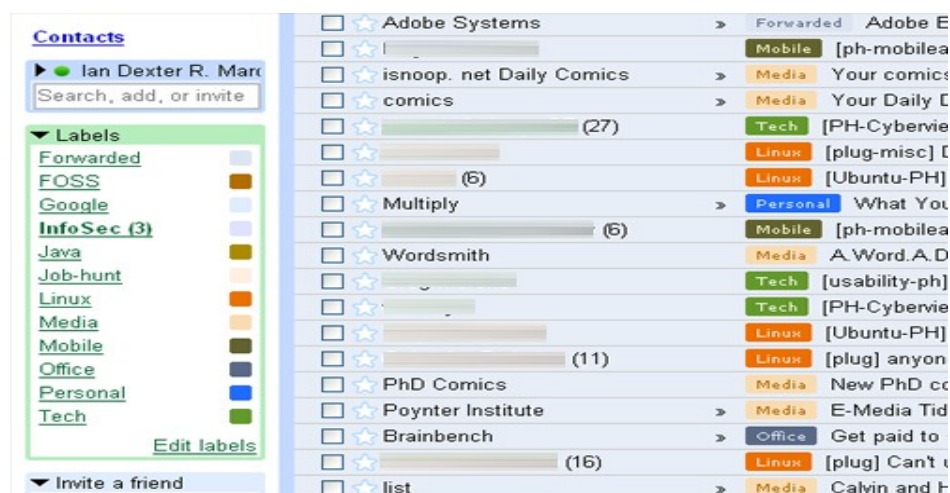


Ilustración 7: Etiquetas en Gmail

En la anterior ilustración, el recuadro verde izquierdo representa el área donde se muestran todas las etiquetas definidas por el usuario para la clasificación de sus correos. En Gmail, estas deben definirse previamente en *Settings\Labels*. Al pinchar sobre una de estas etiquetas, se filtrarán los correos del usuario, mostrándole sólo las conversaciones marcadas con esa etiqueta.

A la derecha pueden verse los correos del usuario, y cómo estas etiquetas pueden ser aplicadas en cada uno. En la imagen, cada etiqueta queda marcada en un recuadro de diferente color. El usuario puede añadir otras etiquetas a cada correo, o borrar las ya existentes; de este modo, una conversación puede almacenarse fácilmente en dos categorías distintas, evitando la complicación que ocurría en los sistemas de carpetas cuando un recurso podía ser almacenado en dos lugares distintos.

Si recordamos el modelo formal para las folksonomías antes presentado (una tupla $F := (U, T, R, Y)$, donde U , T y R son conjuntos finitos cuyos elementos se denominan *usuarios*, *etiquetas* y *recursos*, respectivamente; Y es una relación ternaria definida sobre ellos ($Y \subseteq U \times T \times R$), cuyos elementos se denominan *asignaciones de etiqueta*), podemos comprobar que esta clasificación supone un caso degradado de la misma, en que el conjunto de usuarios contiene un único elemento. En algunos *papers*, como puedan ser [3] (el mismo donde se aporta esta definición formal), ya se ha introducido un término para esta clasificación unipersonal, *personomía*.

2.2.3.4. Google Image Labeler: ESP Game

El reconocimiento de imágenes es una tarea para la cual los ordenadores no dan buenos resultados actualmente, a pesar de resultar de gran facilidad para usuarios humanos. Por este motivo, habitualmente los sistemas de búsqueda de imágenes no se basan directamente en su contenido, sino que aprovechan los metadatos o textos relacionados con la imagen para ejecutar búsquedas textuales sobre éstos, más desarrolladas actualmente. Las folksonomías para imágenes, como pueda ser Flickr [4], han supuesto una contribución a ello, al aportar una colección de etiquetas descriptivas para estas búsquedas.

Buscando enriquecer los sistemas de búsqueda basándose en esta idea, el *ESP Game* [5], experimento de computación humana ideado por Luis von Ahn de la universidad de Carnegie Mellon, trató de animar a los usuarios a jugar a etiquetar imágenes. Más tarde, Google lanzó Google Image Labeler [6], una nueva forma del ESP Game. Este experimento tenía una serie de particularidades que se diferenciaban del modo común en que son recogidas las etiquetas en las folksonomías, buscando obtener una mayor riqueza en las mismas.

Las reglas son: una vez iniciada sesión, el usuario es emparejado con otro usuario elegido aleatoriamente, que está jugando en ese mismo instante. Estos usuarios no se conocen entre sí y no pueden comunicarse. Una vez emparejados, se les muestra a ambos la misma imagen, y su tarea es etiquetarla, añadiéndose puntos por cada etiqueta en la que coincidan ambos usuarios. Estas etiquetas se considerarán buenos descriptores de la imagen. En total, tienen dos minutos y medio para etiquetar quince imágenes.



Ilustración 8: Google Image Labeler

Sin embargo, hay problemas que limitan su utilidad. Usualmente, los usuarios preferirán palabras excesivamente genéricas para describir el contenido. Una solución a esto es la inclusión de palabras *tabú*: cuando muchas parejas de usuarios coincidan en una misma etiqueta, esta pasará a ser *tabú* y no se darán puntos si futuras parejas la incluyen. Esto estimula en la búsqueda de nuevas palabras razonables, esperando coincidir en alguna de ellas.

Por otra parte, existen barreras idiomáticas. Los usuarios son de todo el mundo, por lo que para tener posibilidades de coincidir se utiliza por defecto el idioma inglés para describir las imágenes. Sin embargo, no todos los usuarios tienen el mismo nivel de inglés, lo que limita las etiquetas usadas, y puede evitar la aparición de etiquetas muy descriptivas pero poco conocidas.

Y existe el riesgo de *spamming*. Para Google Image Labeler, ocurrió que ciertas palabras exóticas como *abrasives*, *accretion* o *bequeathing* comenzaron a ser usadas por muchos usuarios para etiquetar cualquier imagen, bien fuera por diversión o para incrementar su puntuación. Estas palabras tuvieron que ser identificadas manualmente para evitar posterior abuso.

2.3. Propiedades de las folksonomías

En este apartado trataremos de aportar un análisis más detallado de ciertas características especiales de las folksonomías, buscando su comprensión en profundidad.

2.3.1. Tipos de folksonomías

A pesar de haber hablado hasta ahora de las folksonomías como un concepto genérico, no debe pensarse en ellas como sistemas uniformes e idénticos. Los diversos sitios web que las han aplicado han introducido diversas diferencias y matices según necesidad, adaptándolas mejor al determinado contexto de cada sistema, que de algún modo afectan a las asignaciones de etiquetas en ellas producidos. Dedicaremos este apartado a explicar diversos aspectos diferenciadores de particular relevancia, basándonos en la clasificación de los sistemas de etiquetado introducida por [7], y posteriormente matizada por [8].

1. Tipo de los objetos etiquetados

La diferenciación más evidente entre los diferentes sitios basados en folksonomía es el tipo de recursos que clasifican. Prácticamente cualquier tipo de objeto que pueda ser representado digitalmente es susceptible de ser utilizado como base de una folksonomía, lo que ha motivado que amplio rango de páginas web surgieran centrando su dominio a tipos muy diferentes. Ejemplos varios son Flickr para imágenes, Youtube para vídeos, del.icio.us para enlaces web o Technorati para entradas de blogs.

Las implicaciones del tipo de objeto elegido se extienden sobre diversos ámbitos. Importante es notar que ciertos recursos pueden sugerir al usuario ciertas etiquetas con preferencia sobre otras. Ejemplos evidentes serían aquellos tipos que contengan contenido textual, como documentos, entradas de blog o enlaces: puede verse como natural incluir en el etiquetado algunas de las palabras más relevantes incluidas en tales textos, opción de la que se carece en el etiquetado de imágenes o sonido. Relacionado con este mismo motivo, ciertas técnicas de recomendación de etiquetas serán aplicables o no dependiendo del tipo, con resultados dispares: para contenidos textuales podrán aplicarse técnicas de análisis de texto, en un estado bastante avanzado; en cambio, el reconocimiento de imágenes o audio requerirá de técnicas de mayor complejidad, que posiblemente devuelvan peores resultados.

2. Origen de los recursos etiquetados

El origen de los recursos puede variar dependiendo del sistema. En sistemas abiertos, los usuarios pueden ser los encargados de incluir los materiales, subiendo al sistema los vídeos, imágenes o cualquier otro tipo de recurso que ellos consideren oportunos. Otros sistemas cerrados aportan un conjunto de contenidos controlado, permitiendo a los usuarios su etiquetado, pero no su aumento. Como se ve, esta decisión afecta a su tamaño y posibilidad de expansión

3. Restricciones en los posibles recursos a etiquetar

Una de las caracterizaciones más importantes es ésta. Ya Thomas Wander Val se centró en esta característica para explicar diferencias entre varios sitios web basados en folksonomía, acuñando dos términos: folksonomías amplias y estrechas [9]. Estos dos tipos básicos hacen referencia a dos situaciones habituales: o bien un usuario tiene permiso para añadir etiquetas a cualquier recurso del sistema, incluso aquellos que no ha incluido él mismo (folksonomía amplia), o bien sólo se le permite etiquetar aquellos recursos que le sean propios (folksonomía estrecha). Otras situaciones son posibles, dependiendo del sistema: por ejemplo, podría permitirse etiquetar un

recurso por su dueño y otros usuarios que haya indicado como “amigos”, o limitarle a una serie de elementos escogidos por el sistema (ESP Game).

Dependiendo de las políticas escogidas, serán diferentes los esquemas de etiquetado obtenidos, como es evidente. Mientras que en un sistema donde un recurso sólo puede ser etiquetado por su creador, la elección de estas etiquetas reflejará su personal visión del objeto, en un sistema esto se permita a cualquier usuario acumulará en cada objeto las diferentes opiniones y perspectivas de cada uno, enriqueciendo su información y facilitando de este modo su recuperación por diferentes tipos de búsquedas.

Es importante notar que el modo en que las etiquetas se agreguen sobre un recurso será diferente dependiendo de esta elección. En caso de que sólo un usuario, el propietario, pueda etiquetar un determinado recurso, las etiquetas sobre tal objeto serán únicamente las suyas. Sin embargo, en caso de que varios usuarios puedan etiquetar un mismo recurso, podrá darse cierta agregación de las mismas sobre el objeto: una etiqueta podrá ser utilizada varias veces sobre el mismo por diferentes usuarios, descubriendo de este modo cierto acuerdo y relevancia.

4. Ayudas al etiquetado

Existe una variedad de ayudas al etiquetado que pueden influir en gran modo al modo en que los usuarios utilicen la folksonomía. Variedades de estas ayudas son ofrecer un conjunto de etiquetas utilizadas ya por otros usuarios, para el mismo recurso o en el conjunto general del sistema, permitiendo añadirlas con tan solo un click. Otras posibilidades pasan por procesar automáticamente el contenido del recurso y ofrecer posibles etiquetas significativas, sea por su contenido o por haber sido asignadas a recursos similares; se explicará más adelante en la sección “2.4.2.3.Recomendación de etiquetas”. Estos sistemas tienen la ventaja de permitir una mayor convergencia en la folksonomía, multiplicando el número de etiquetas utilizadas por los usuarios y reduciendo el número de etiquetas mal formadas (p.e. por mala ortografía). Como contrapartida, pueden coartar la libertad del usuario o molestarle, si se plantean de un modo demasiado invasivo. En cada caso deben estudiarse sus implicaciones.

5. Conectividad

Es interesante también observar el modo en que las diferentes partes de la folksonomía se conectan entre sí, que pueden formar diferentes patrones dependiendo del caso. Tres son los elementos básicos de una folksonomía, hemos dicho: usuarios, recursos y etiquetas, que se unen mediante asignaciones de etiquetas. Ahora bien, los elementos de cada uno de estos conjuntos pueden estar también relacionados entre sí de diferentes modos, aportando más información a la folksonomía.

Habida cuenta de que las folksonomías suelen montarse sobre sistemas de redes sociales, en que los usuarios forman una comunidad enlazada, ocurre habitualmente que estos usuarios mantienen entre sí lazos indicados de forma explícita, que informan de quiénes son sus conocidos y amigos. Similarmente, los recursos pueden estar unidos de muy diferentes formas aparte de por sus etiquetas: es posible que existan grupos de recursos de similar temática, o unidos en discusiones, donde unos sean respuesta a otros. Finalmente, pudieran establecerse también métodos para el enlazado o agrupado de etiquetas, como los clusters que ofrece Flickr.

En cada uno de estos tres casos, es interesante estudiar si existen estos enlaces y cuáles son sus características, pues esta información puede enriquecer la folksonomía. Por ejemplo, un modo de utilizar los enlaces entre usuarios sería dar ciertos permisos (etiquetado de imágenes propias, por ejemplo) a los usuarios amigos y denegárselos al resto. Las posibilidades son muchas y abiertas.

2.3.2. Motivación para el etiquetado

Durante mucho tiempo, se creyó que sistemas sociales para la clasificación de contenidos no podrían funcionar, pues la asignación de metadatos a un recurso es un proceso que requiere cierto entrenamiento y esfuerzo. Con la flexibilidad permitida en las etiquetas se ha conseguido evitar la primera barrera, aceptando cualquier etiqueta asignada por el usuario; sin embargo, cierto esfuerzo (si bien escaso) sigue siendo necesario por su parte. Habida cuenta del éxito de las folksonomías, vemos que los usuarios sí etiquetan sus recursos, por lo que cabe preguntarse qué ventajas obtienen los usuarios a cambio que les motiven a este esfuerzo adicional. Varios son los incentivos relacionados, señalados entre otros por [7]:

- **Utilidad:** Los usuarios utilizan las etiquetas para ser capaces de encontrar y recordar la información que introducen. Esto cambia el enfoque con respecto a las taxonomías tradicionales: los usuarios no están tan preocupados por la exactitud de los términos utilizados o por su calidad, sino por la utilidad que les reportan. El individuo se preocupa por sí mismo, y aun con ello aporta un bien a la comunidad.
- **Libertad:** La falta de una autoridad central que limite la selección de las etiquetas o exija ciertas obligaciones para su uso, da a los usuarios cierta tranquilidad en el momento del etiquetado. Se sienten seguros sabiendo que pueden utilizar las etiquetas como ellos deseen. A pesar de que esto puede llevar a una mala calidad en la clasificación, en realidad los usuarios tienden a buscar la utilidad en su selección de un modo filantrópico, buscando ayudar a la comunidad.
- **Feedback inmediato:** Los sistemas folksonómicos dan una respuesta inmediata al usuario. Son necesarios muy pocos pasos para categorizar un contenido, que queda automáticamente añadido al sistema, para su utilización por el mismo usuario o el resto de la comunidad.
- **Descubrimiento:** Estos sistemas recompensan al usuario con métodos de navegación que le permiten descubrir muchos recursos relacionados a un tema, explorando un área más que buscando un contenido en concreto. Los contenidos se relacionan por similitud de etiquetas o relaciones entre los usuarios que los publican.
- **Motivos sociales:** No deben obviarse tampoco los diversos factores sociales que influyen en las elecciones de etiquetas. Un usuario puede etiquetar un recurso propio de forma exhaustiva con la mera intención de que pueda ser encontrado por otros usuarios, atrayendo de este modo su atención; esto cobra especial sentido cuando estos recursos son obras personales del usuario, como fotografías o vídeos artísticos. De igual modo, pueden utilizarse las etiquetas como expresión de la personalidad del usuario y su particular opinión

sobre el recurso; un ejemplo podría ser la etiqueta “seen live” (visto en directo) de habitual uso en last.fm aplicada a artistas: paradójicamente, no aporta información útil sobre el artista, pero sí sobre el usuario que la ha aplicado.

2.3.3. Ventajas

Comparados con los sistemas de clasificación tradicionales, las folksonomías presentan una serie de ventajas no presentes en los primeros, que las hacen especialmente adecuadas para ámbitos tan cambiantes y amplios como pueda ser manejar la información sobre Internet. En estos sistemas, el contar con la opinión de muchos usuarios resulta la principal ventaja, siendo capaces de extraer algo que podríamos llamar “la sabiduría de la masa”, aunque la contribución aislada de un usuario pueda incurrir en pequeños errores, como mala ortografía o inconsistencias. Las principales ventajas fueron ya señaladas en [10], de las que se señalan algunas de las más relevantes:

- **Inclusivas:** Las folksonomías recogen el vocabulario de todos sus usuarios sin necesidad de estructurarlo o filtrarlo, como sería normal en el manejo de una taxonomía. Todas las etiquetas incluidas son tenidas en cuenta para responder a las consultas de los usuarios, reflejando de este modo todas sus perspectivas y necesidades, sean éstas populares o minoritarias. De hecho, como se verá más adelante al ver la distribución seguida por las etiquetas en las folksonomías, las visiones minoritarias tienden a superar en gran número a las pocas mayoritarias, que serían las principales a incluir en una taxonomía clásica.
- **Actualización:** Las folksonomías permiten una inmediatez y flexibilidad que no pueden alcanzar otros sistemas controlados. En el uso cotidiano del sistema los usuarios introducen sus nuevos contenidos aportando a su vez etiquetas para los mismos, registrándose estos cambios de inmediato, y permitiendo ser recuperados mediante consulta al momento. Esto les permite reflejar rápidamente cambios o sucesos recientes, algo que en sistemas controlados de importante tamaño nunca podría darse con inmediatez.
- **Automáticas:** Frente a los sistemas taxonómicos tradicionales, cuyo control se consigue de forma centralizada por una serie de usuarios con permiso para añadir contenido, las folksonomías tienden a moderarse y a corregir sus contenidos de forma “automática”, entendiéndose por esto que el esfuerzo acumulado antes en unos pocos usuarios se distribuye ahora por la totalidad de la comunidad, consiguiendo una mucha mayor escalabilidad y coste reducido.
- **Flexibles:** Frente a sistemas tradicionales donde un elemento debía pertenecer a una y solo una categoría de las definidas en el sistema, las folksonomías constituyen un sistema mucho más flexible, con restricciones mucho más limitadas. Las categorías, representadas por etiquetas, no están previamente definidas, sino que se crean y multiplican en el mismo instante de etiquetado, y un elemento no debe limitarse a una sola, sino que puede participar en varias al incluirse varias etiquetas. El mantenimiento, de este modo, es mucho más sencillo, y consigue evitarse la pérdida de semántica al intentar ajustar las etiquetas a un esquema predefinido.

- **Adaptadas a las necesidades del usuario:** Precisamente por estar contruidos por la comunidad de usuarios, estos sistemas son altamente flexibles, y sus líneas de contenidos se adaptarán por su propia naturaleza a las necesidades y deseos de la mayoría de sus usuarios. En las taxonomías clásicas, la separación entre el catalogador y el usuario podía producir desinterés en estos últimos si se sentían frustrados al respecto de la línea de contenidos. El catalogador tenía la responsabilidad de intuir qué contenidos pudieran resultar de interés para los usuarios, algo que podía llevar a error.

Como resumen de todas estas ventajas, puede decirse que las folksonomías resultan el sistema de clasificación actual mejor capacitado para las nuevas necesidades creadas por los fenómenos de web social. Unen a las ventajas anteriormente citadas y a una buena eficacia en estos ámbitos, un coste de creación y mantenimiento incomparablemente bajo, donde todo el esfuerzo queda distribuido entre la comunidad de usuarios, consumidores directos del contenido. Representan un fenómeno creciente: los usuarios de Internet ya han adaptado esta tecnología como suya y la hacen crecer.

2.3.4. Defectos

A pesar de las diferentes ventajas que su uso ofrece comentadas hasta ahora, las folksonomías no están faltas de defectos, que limitan su uso. Señalados como las ventajas anteriores en [10], mencionaremos los más importantes:

- **Falta de control de variaciones del lenguaje:** Siempre se produce cierta imprecisión en la clasificación, debida entre otros factores a la variabilidad del lenguaje, al variable comportamiento de los usuarios y a la falta de control de sinónimos en la mayoría de los casos. Los problemas básicos son el manejo de la *homonimia* (etiquetas iguales con diferente significado) y *sinonimia* (etiquetas diferentes con similar significado), que dificultan la recuperación de información. Por ejemplo, para hablar de teléfonos móviles, diferentes usuarios podrían utilizar las etiquetas *movil*, *móvil*, *celular*, *telefono.movil*, *moviles*, etcétera.

De cualquier modo, hay quien defiende que este defecto no es tal. La falta de restricciones permite a los usuarios elegir las etiquetas que consideren más oportunas para describir sus recursos sin la pérdida de semántica que ocurre en un vocabulario controlado. Muchos usuarios prefieren usar una cierta etiqueta, y podrían no aceptar otra que generalmente se considerase sinónima, como podría ser “cine” y “película”. De este modo, esta característica quedaría más como una opción de diseño que como un defecto.

- **Diferentes niveles de detalle en las etiquetas:** No todos los usuarios acuerdan un mismo modo de ver un objeto simplemente por su falta de conocimiento sobre él, y esto se muestra en el uso de términos más generales o más concretos. Por ejemplo, un usuario podría subir una foto de un árbol y etiquetarlo simplemente como “árbol”, cuando un usuario con conocimientos de botánica podría haberlo clasificado más adecuadamente como “aligustre” (que en un sistema jerárquico sería etiqueta hija de “árbol”). El problema viene cuando este segundo usuario quiere recuperar todas las fotos de aligustres presentes en el sistema, cuando estas han sido etiquetadas únicamente como “árbol”, produciéndose una

pérdida de semántica. Tampoco es posible para el usuario construir relaciones padre-hijo en sus etiquetas, aunque en un primer momento tienda a buscar esta utilidad por mera familiaridad. A pesar de la libertad que esto aporta, también impide la consecución de una clasificación más profunda y detallada.

- **Difícil buscar por un elemento en concreto, u obtener todos los recursos relevantes para una consulta:** Las folksonomías dan malos resultados para la búsqueda de un elemento en concreto de la base de contenidos, pues las etiquetas utilizadas para describir un contenido no lo identifican por completo. De igual modo, no es posible asegurar que para una determinada búsqueda se han conseguido todos los documentos relevantes contenidos, debido a lo informal de la selección de etiquetas que sirve para su recuperación. El ámbito en que mejor se desenvuelven es en el de descubrimiento de contenidos, ejecutando búsquedas sobre un determinado ámbito. Así, en toda búsqueda es probable que el usuario deba buscar en una serie de elementos relacionados con la etiqueta indicada, pero que quizá puedan no guardar relación entre sí.
- **Susceptibles de abuso:** Precisamente en cuanto las folksonomías se construyen a partir de las aportaciones individuales de los usuarios, es posible que usuarios lo utilicen en cambio para corromper el sistema abusando de sus posibilidades. Sus motivaciones son variadas, desde afán de notoriedad a mera diversión. Un ejemplo básico de cómo puede conseguirse este abuso es simplemente eligiendo etiquetas no relacionadas con el contenido almacenado, disminuyendo de este modo la calidad de las búsquedas. Puede ocurrir que el usuario intente asociar gran número de etiquetas a su contenido para aparecer en el mayor número de búsquedas posibles; por este motivo, el número de etiquetas suele estar limitado en los sistemas.

2.3.5. Representación gráfica: nubes de etiquetas

En muchas ocasiones, es de utilidad dentro un sistema folksonómico plantear un conjunto de etiquetas al usuario, como sugerencia o referencia. Para estos listados, es tendencia común en los sistemas utilizar una estructura que se ha venido a denominar de *nube de etiquetas*, donde las etiquetas se organizan de un modo ordenado y gráfico, permitiendo al usuario de un vistazo encontrar los conceptos principales y explorar los relacionados.

La representación más común de las nubes de etiquetas muestra un conjunto de etiquetas que han sido consideradas de interés (por ejemplo, las etiquetas más utilizadas en todo el sistema, o las más usadas para un determinado recurso), colocadas una tras otra y ordenadas alfabéticamente, variando el tamaño de la fuente para cada etiqueta en proporción directa con la relevancia de la misma dentro del listado. El usuario podrá interactuar con la nube de etiquetas haciendo click sobre una de ellas, ejecutándose la acción asociada, que dependiendo del contexto pudiera ser añadir la etiqueta a un listado, mostrar al usuario un conjunto de recursos etiquetados con la misma, o cualquier otra opción.

Tags populares de este artista

00s 90s **alternative** alternative metal **alternative rock** ambient an
avant-garde awesome california classic rock dark dark ambient electronic emo expe
favorite favorite artists favorite bands favorites favourite favourites great lyricists grunge
heavy heavy metal hip hop indie indie rock industrial industrial metal intense math roc
keenan **metal** nu metal post-rock prog prog metal prog roc
progressive metal **progressive**

Ilustración 9: Nube de tags creada por last.fm para describir la banda musical "Tool"

Este modelo básico es muy popular y ampliamente usado; sin embargo, se han planteado objeciones a aspectos tales como su accesibilidad o usabilidad.[11][12]. Se argumenta que el orden alfabético de las etiquetas es de escasa utilidad, pues no están pensadas para ser leídas una tras otra por el usuario, sino percibidas de un golpe de vista; sería de mucha mayor utilidad ordenar estas etiquetas de modo que etiquetas relacionadas estuviesen cerca entre sí, permitiendo de este modo enriquecer la interfaz permitiendo al usuario navegar por intereses relacionados.



Ilustración 10: Flickr Related Tag Browsing

Otras visualizaciones se han creado y utilizado tomando en consideración estas recomendaciones, aunque suelen encontrarse en estado de prototipo y no gozan de tanta popularidad. Ejemplos aplicados a Flickr son Tag Graph [13] y Flickr Related Tag Browsing [14]. Ambos parten de una etiqueta dada por el usuario, y presentan con una interfaz gráfica basada en

Flash otras etiquetas relacionadas. Su aplicación es menor, pues al no estar basados directamente en HTML su accesibilidad (i.e. posibilidad de ser accedidos y vistos por todas las personas y equipos, independiente de sus capacidades físicas y técnicas, respectivamente) es menor. En cualquier caso, son experimentos de interés, pues aparte de buscar mejores formas de representación de la información y navegación, potencian el componente lúdico con su atractivo aspecto, algo muy relevante en el contexto de ligero entretenimiento en que se mueven muchas folksonomías.

2.3.6. Patrones comunes en el etiquetado

Existen diferentes estudios que han tratado de extraer diferentes patrones y características en el modo en que los usuarios utilizan las folksonomías existentes en la actualidad. En este apartado resumiremos algunas de sus conclusiones más relevantes y frecuentes.

2.3.6.1. Ley de potencia

Una distribución estadística de especial importancia en el estudio de las folksonomías es la de **ley de potencia**, relación polinómica entre dos variables que cumple la propiedad de *invarianza de escala*. Se denomina como ley de potencia a la relación entre dos variables x y $f(x)$, donde ambas pertenecen al conjunto de los números reales, y se cumple que:

$$f(x) = ax^k$$

donde a y k son constantes. La propiedad de *invarianza de escala* hace referencia al hecho de que, en caso de la variable de entrada se multiplique por una constante, el valor de la función se verá también multiplicado por una constante. Esto es:

$$f(cx) = a(cx)^k = c^k (ax^k) = c^k f(x)$$

En la siguiente ilustración puede verse el perfil característico de las gráficas de tales distribuciones. Como puede verse, sólo unos pocos elementos toman valores altos, y la mayoría unos valores pequeños (la coloquialmente denominada *larga cola* en las gráficas de la distribución, que se alarga asintóticamente hasta cero y ocupa la mayor parte del dibujo).

La relación es más clara tomando el logaritmo de ambos lados:

$$\log(f(x)) = k \log(x) + \log(a)$$

Así transformada, puede comprobarse que la ecuación se corresponde con la de una recta, representando el factor k la pendiente de la misma. Es por esto que para demostrar que cierta relación entre variables sigue una ley potencial suele utilizarse una *gráfica log-log*, esto es, una gráfica en la que se represente la relación entre los logaritmos de los valores de ambas variables. Cuanto más se acerque a una línea recta, más se ajustará a una ley potencial.

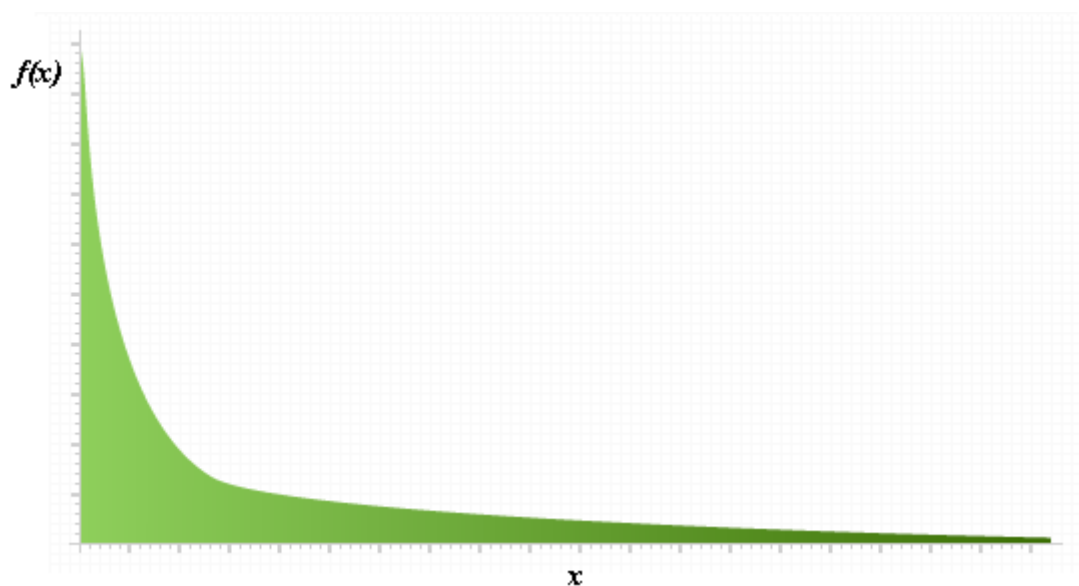


Ilustración 11: Aspecto de la gráfica de una relación de ley potencial

La relación de ley potencial es de especial interés por su habitual aparición en fenómenos naturales y humanos aparentemente sin relación, como las distribuciones de varias variables económicas, la intensidad de los terremotos, la frecuencia de las palabras, o las ventas de libros. Existen otras distribuciones relacionadas, como Zipf o Pareto, que modelan comportamientos similares [15].

2.3.6.2. Distribuciones en las folksonomías

Es un fenómeno conocido y respaldado por varios resultados experimentales [9][16][17][18], que la distribución estadística de múltiples factores de las folksonomías puede modelarse como una ley potencial.

En las ilustraciones inferiores, tomadas de [17], se muestran las gráficas log-log de dos diferentes aspectos en Flickr: la frecuencia de cada etiqueta (el número de recursos para los que ha sido asignada), y el número de etiquetas asignadas a cada imagen. Como puede verse, ambas se acercan con bastante precisión a la recta, aportando una evidencia estadísticamente significativa de ajustarse a tal distribución.

Esto puede mostrarse intuitivamente pensando que, para las frecuencias de cada etiqueta, existen unas pocas etiquetas muy genéricas, aplicables en gran variedad de casos, como puedan ser *2006* o *place*. A medida que las etiquetas son más concretas son menos aplicables, disminuyendo de este modo su frecuencia. Y debemos tener en cuenta las etiquetas mal formadas o muy personales, que suponen un amplio rango de etiquetas de uno o pocos usos.

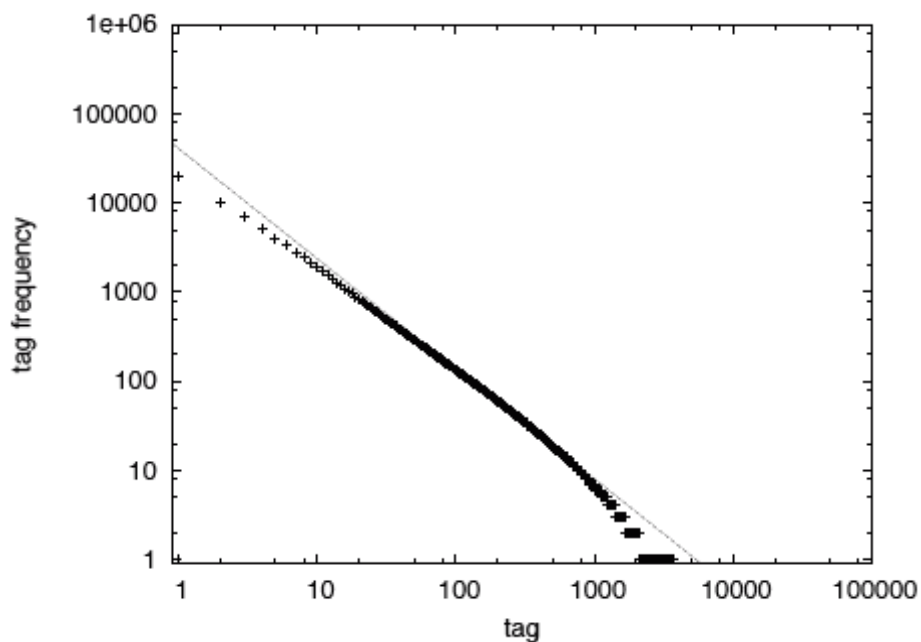


Ilustración 12: Frecuencia de cada etiqueta en Flickr. Gráfica log-log: eje horizontal con las etiquetas ordenadas de mayor a menor frecuencia, eje vertical con la frecuencia de cada etiqueta

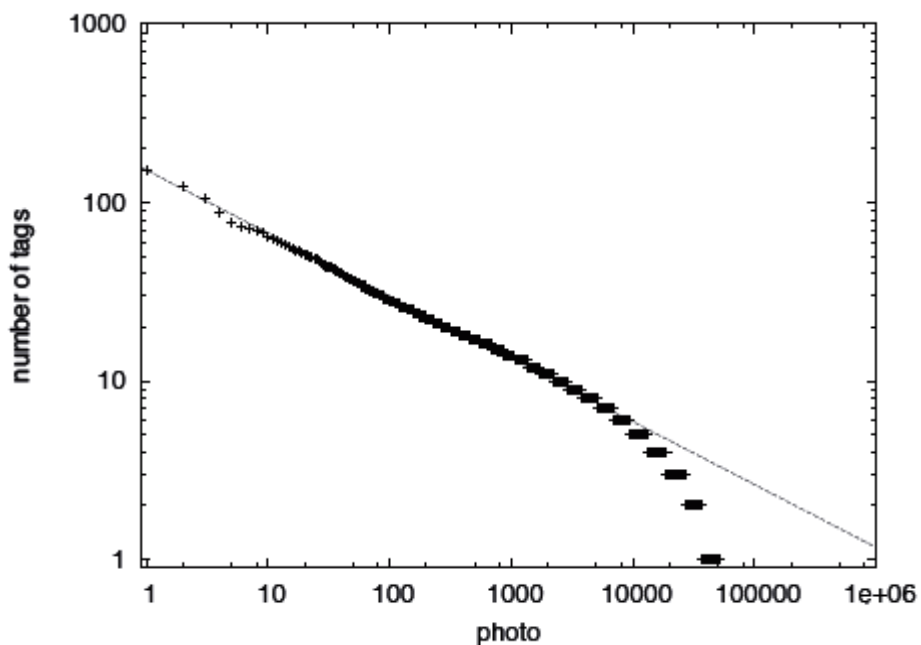


Ilustración 13: Número de etiquetas por imagen en Flickr. Gráfica log-log: eje horizontal con las imágenes ordenadas de mayor a menor n° de etiquetas, eje vertical con n° de etiquetas para cada imagen

2.3.6.3. Análisis de co-ocurrencia

El análisis de co-ocurrencia trata de descubrir relaciones entre elementos, estudiando la frecuencia con que se utilizan conjuntamente. Se basa en la idea de que conceptos relacionados tenderán a utilizarse conjuntamente con cierta frecuencia. Esta idea se ha aplicado en el estudio de las folksonomías para el descubrimiento de patrones en el uso de etiquetas, que permitan relacionar etiquetas sinónimas o intrínsecamente unidas por otro motivo.

En [16] se estudiaron estos patrones para un conjunto amplio de datos tomado de del.icio.us. En este contexto, el análisis de co-ocurrencia de etiquetas se centra en parejas de etiquetas que hayan sido asignadas a un determinado recurso. Esto es, si un determinado recurso R ha sido etiquetado por tres usuarios diferentes, utilizando cada uno las etiquetas X e Y, la pareja X e Y tendría en el grafo de co-ocurrencia un valor 3 (tres ocurrencias).

Estas frecuencias se tomaron para cada par de las etiquetas utilizadas para un mismo recurso (URLs en el caso de del.icio.us), limitándose, para evitar grafos demasiado poblados, a las 40 etiquetas utilizadas con más frecuencia. Esto resulta en una matriz simétrica de co-ocurrencia, conteniendo la distancia de cada etiqueta al resto. Para permitir una visión gráfica de las mismas, se utilizó una transformación MDS (*multi-dimensional scaling*), obteniendo como resultado un gráfico de dos dimensiones para cada matriz. En estos gráficos, se incluye en el plano cada una de las etiquetas en cuestión, situadas de tal modo que la distancia entre dos etiquetas cualesquiera representa su grado de relación: a mayor cercanía, mayor relación entre ambas.

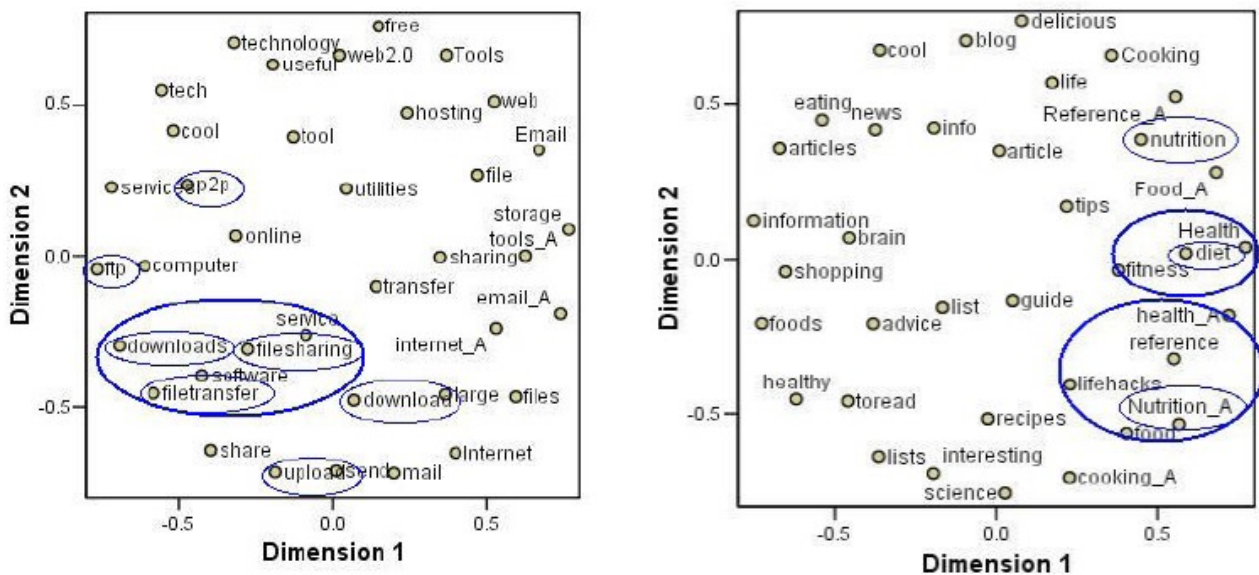


Ilustración 14: Grafos MDS de co-ocurrencia para las URL "www.yousendit.com" y "www.bellybytes.com/articles/29foods.shtml", respectivamente

Como puede verse en la ilustración anterior, pueden descubrirse ciertos conjuntos significativos aplicando cierto agrupamiento sobre las etiquetas así mostradas. En la figura de la izquierda, que muestra el grafo para la web "www.yousendit.com" dedicada a almacenar y compartir ficheros, varias etiquetas relacionadas con este tema se muestran cercanas entre sí,

agrupadas en un mismo área. Del mismo modo ocurre en el grafo de la derecha, de una web dedicada a temas de nutrición.

A pesar de todo, otras etiquetas relacionadas pueden verse en ambos apartados del conjunto. Esto puede entenderse si se piensa que el acuerdo entre etiquetas relacionadas se da sólo cuando un mismo usuario utiliza determinado par de etiquetas. Sin embargo, a pesar de que dos etiquetas puedan representar un mismo concepto, no siempre serán utilizadas conjuntamente en todos los casos: usuarios que utilicen la etiqueta “social.bookmarking” pueden no incluir la etiqueta “social_bookmarking”, que si bien es evidente que quiere decir lo mismo, no resulta relevante al usuario. Otros experimentos demostraron esta separación sobre, en concreto, variaciones ortográficas de una misma etiqueta; separación que, ahora bien, pudiera no ocurrir en otros casos si los usuarios conscientemente incluyeran ambas etiquetas para el enriquecimiento de la folksonomía.

En general, es éste un sistema adecuado para descubrir relaciones relevantes entre etiquetas. Ahora bien, debido a su método estadístico, funcionará tanto mejor cuanto mayor sea la muestra de usuarios y etiquetas que participen para un mismo recurso. De este modo, podrá asegurarse cierta convergencia y acuerdo; con pocos datos, puede ocurrir que no se refleje todo el conocimiento del dominio.

2.3.6.4. Características de las etiquetas utilizadas

En diversos estudios se ha tratado de analizar la masa de etiquetas utilizada por los usuarios de los sistemas folksonómicos, buscando encontrar patrones de interés y estudiar sus diferencias con la selección que pudiera hacerse en un sistema formal de metadatos. Se describen en este apartado algunas de las características encontradas.

2.3.6.4.1. Temática de las etiquetas

En [17] se llevó a cabo un estudio tratando de descubrir la temática general de las etiquetas utilizadas para la descripción de las imágenes en Flickr. Para ello, se utilizaron las etiquetas de un cierto número de imágenes, obteniendo para cada etiqueta la categoría a la que WordNet asignaba la palabra; en caso de asignarla a varias categorías, se tomaría solamente la de mayor prioridad para esa palabra.

Puede verse en la ilustración inferior los resultados de esta categorización, representando el círculo menor los resultados totales (con un importante grado de etiquetas inclasificadas, el 48%) y el mayor los porcentajes relativos a las etiquetas que han podido clasificarse con éxito. El alto porcentaje de inclasificados se comprende teniendo en cuenta que WordNet resulta tan sólo aplicable para etiquetas en inglés, que no representan el total de las etiquetas de Flickr; y más aún, para el subconjunto de las etiquetas en inglés existe también un número de etiquetas mal formadas, con faltas ortográficas o caracteres extraños que dificultan su reconocimiento.

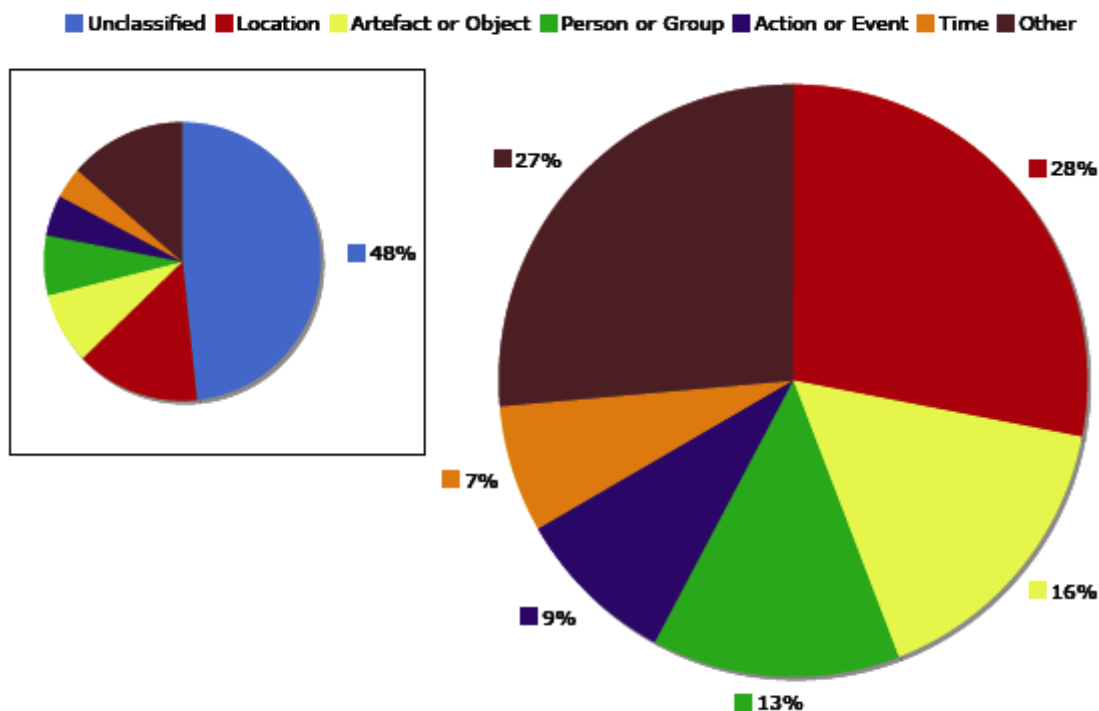


Ilustración 15: Distribución de categorías de WordNet para las etiquetas de Flickr

Dentro de las categorías descubiertas, vemos que “localización”, “artefactos y objetos” y “gente o grupos” representan los tipos más frecuentes. De esto se concluye en [17] que la tendencia general es utilizar las etiquetas para dar un contexto general a la imagen, con preferencia a describir sus contenidos, indicando por ejemplo en qué momento y lugar fuera tomada.

2.3.6.4.2. Tipos de etiquetas

Relacionado con la temática de las etiquetas investigada en el apartado anterior, otros estudios han tratado de descubrir qué función dan los usuarios a las etiquetas que utilizan, esto es, con qué intención, para informar de qué aspectos del recurso, escogen sus etiquetas.

En [19] se llevó a cabo un estudio sobre distribuciones y propiedades del etiquetado de recursos sobre del.icio.us, donde se trató de dividir las etiquetas utilizadas en ciertos grupos amplios dependiendo de la función que los usuarios daban a las mismas. En ella, podía apreciarse ya la tensión de los usos más personales de las etiquetas, identificando aspectos de interés principalmente para el mismo usuario, y usos orientados a la comunidad, facilitando la localización y recuperación de los recursos por otros usuarios. Las propiedades más relevantes señaladas eran:

- **Identificar el recurso:** Una de las funciones más básicas es el uso de etiquetas para describir lo que puede encontrarse directamente en el recurso, permitiendo de este modo su encuentro al ejecutar búsquedas que planteen tales características. Ejemplos sencillos serían, para etiquetar la foto de un elefante, utilizar “elefante” o “animal”.

- **Identificar el contexto del recurso:** Fuera de la descripción directa del recurso, suele existir otra información no presente en el mismo que le sirva de contexto y complete su significado. Dado el ejemplo anterior, otras etiquetas podrían completarlo dando su localización (“*zoo*”, “*Madrid*”) o momento (“*2007*”).
- **Identificar a su propietario:** Dependiendo del interés del propietario del recurso porque su autoría quede reflejada, pueden utilizarse etiquetas para resaltar la misma, incluyendo el nombre de su autor o su apodo.
- **Describir cualidades y características:** Otras etiquetas pueden mostrar la opinión del usuario sobre el contenido, como pueden mostrar etiquetas como “*divertido*” o “*interesante*”. En muchos casos, son bastante personales, y otros usuarios pueden no coincidir en las mismas.
- **Etiquetas de utilidad únicamente personal:** Uno de los tipos especiales de etiquetas son aquéllas cuyo significado está intrínsecamente relacionado con su creador, no pudiendo ser desligado. Ejemplo de ello puede ser, en Last.fm, dentro de las etiquetas asignadas a grupos de música por los usuarios, la etiqueta “*seen live*”, que identifica aquellos grupos que el usuario ha visto en directo; es evidente que el significado de esta etiqueta no puede desligarse del usuario que la incluye. Otros ejemplos pudieran ser “*mis_fotos*” o similares.

En este último grupo de etiquetas, pueden también incluirse aquellas que representan anotaciones de tareas a ejecutar por el usuario, como puedan ser “*por_leer*” o “*recordar*”. Estas etiquetas difieren de las que pudieran surgir en sistemas de clasificación tradicionales, pues representan una respuesta del usuario, más que información sobre el contenido del elemento. Plantan una unión activa con el elemento etiquetado, en la que el usuario utiliza el elemento con un cierto propósito en sus tareas o intereses.

2.3.6.4.3. Variaciones en las etiquetas

Se ha comentado ya que uno de los principales problemas de las folksonomías es su falta de control sobre las variaciones del lenguaje, al no detectar (si no se establecen sistemas de control) las etiquetas sinónimas. Por ello, es relevante identificar una serie de fuentes comunes de variaciones en las formas en que una misma etiqueta puede presentarse.

Una de las principales es el idioma. En los sitios folksonómicos lo suficientemente amplios, usuarios de muy diversas nacionalidades comparten espacio, lo que lleva a la pregunta de qué idioma deben utilizar para comunicarse. No hay un consenso general, de modo que diversas variaciones pueden encontrarse: etiquetar en el idioma nativo, en uno de consenso (usualmente el inglés) o mezclar etiquetas de los dos tipos. Esto lleva a que un mismo concepto pueda ser referido por etiquetas muy diferentes, como “*vaca*” y “*cow*”.

Otro factor importante es el de las variaciones tipográficas u ortográficas, muy habitual en estos casos: plurales y singulares, infinitivos y formas conjugadas de verbos son ejemplos siempre presentes. A ello, se le añaden las diferencias nacidas por la presencia en los idiomas de caracteres o

signos de puntuación especiales, que siempre generan la duda de si deben incluirse o no. Es típico que elementos como tildes o diéresis sean descartados por algunos usuarios y mantenidos por otros en sus etiquetas.

2.3.6.5. Otras propiedades

Otros estudios se han realizado sobre las propiedades de las folksonomías, abarcando un amplio ámbito. Mencionaremos como ejemplos significativos [20], donde se estudian características de redes, como la propiedad de mundo pequeño o el coeficiente de agrupado, para las folksonomías; [21], donde se estudian modos para la detección de “tendencias”, esto es, variaciones en la popularidad de usuarios, recursos o etiquetas.

2.4. Mejora de las folksonomías

Debido al estado experimental y de continuo crecimiento en que las folksonomías se encuentran en la actualidad, muchas son las ideas y propuestas que han surgido para la mejora de su utilidad. En esta sección daremos una panorámica sobre la diversa naturaleza de estas ideas, intentando aclarar qué ventajas pueden nacer de ellas.

2.4.1. Educación de los usuarios

Uno de los principales problemas de las folksonomías contra los que argumentan los defensores de los sistemas de clasificación clásicos es la falta de reglas en el método de elección de etiquetas, totalmente libre y a la entera elección de los usuarios, que facilita la aparición de etiquetas de escasa utilidad para la recuperación de información por ser redundantes, imprecisas o sólo inteligibles para su autor.

Una de las primeras soluciones ideadas para este problema pasa por la educación de los usuarios (planteada en [22], entre otros), tratando de conseguir que una cantidad suficiente de ellos sea capaz de aportar unas etiquetas “de calidad”, entendiendo por esto unas etiquetas descriptivas del elemento en cuestión y de utilidad para su recuperación. El paso más básico para conseguir algo semejante es ofrecer a los usuarios una serie de consejos para la buena elección de etiquetas; algunas reglas que abordarían pequeños problemas recurrentes en estos sistemas podrían ser:

- Evitar el uso de mayúsculas.
- Utilizar formas singulares con preferencia a formas plurales.
- Evitar errores tipográficos.
- Utilizar tanto etiquetas generales como específicas.
- Añadir sinónimos.

Subyace en el listado la idea de que las etiquetas, para su mayor provecho, deben ser escogidas considerando tanto la utilidad que reportan al usuario como a la comunidad. Para ello, es bueno buscar cierto consenso en los términos elegidos, como atestiguan las tres primeras reglas:

buscan hacer converger el conjunto de etiquetas de los usuarios hacia un vocabulario mínimamente controlado, no repleto de varias formulaciones para un concepto idéntico, sea por plurales o errores tipográficos varios. Luego, las dos últimas reglas buscan enriquecer y relacionar el conjunto de etiquetas, utilizando en una misma asignación diferentes puntos de vista y sinónimos.

Otras propuestas han estudiado la posibilidad de guiar a los usuarios en el momento de la asignación de etiquetas, tratando de este modo asegurar el cumplimiento de estos consejos, o darles ideas para la inclusión de una información más rica y matizada. Inspirados en el modo en que metadatos son asignados a imágenes, tratando de completar con un vocabulario controlado informaciones de diverso ámbito para su total contextualización, como puedan ser temática general, autor, tiempo, localización, etcétera, se han realizado experimentos tratando de forzar a los usuarios de las folksonomías a ajustarse a semejantes modelos.

A este respecto, podemos citar [23], en que se presentaban a los usuarios cuestionarios con campos como los anteriormente mencionados, para el etiquetado de una serie de imágenes pertenecientes a la cultura judía. Los usuarios no debían restringirse a un vocabulario controlado, y conservaban su libertad para el uso de cualesquiera etiquetas, pero debían hacerlo para cada una de las categorías: temática, símbolos, eventos, etcétera. En general, esto no es una práctica recomendable para las folksonomías, pues obligar a la inclusión de etiquetas puede llevar a etiquetas escogidas aleatoriamente y en general a la molestia de los usuarios, pero la existencia de tales cuestionarios como guía les puede servir de recordatorio de diversos aspectos a tener en cuenta en su selección de etiquetas, intentando abarcar un campo semántico más amplio.

En cualquier caso, todas estas medidas, consejos y precauciones estribarán su eficacia en el provecho que los usuarios hagan de ellos. Abrumar a los usuarios con consejos e indicaciones será con probabilidad contraproducente, pues no debe olvidarse que las ventajas de las folksonomías parten en buena parte de la esencial libertad de los usuarios, que queda coartada cuando se introducen demasiados filtros a su elección.

2.4.2. Mecanismos automáticos

La alternativa al esfuerzo por conseguir que los usuarios produzcan etiquetas de mayor calidad, es mejorar el sistema folksonómico para conseguir suplir de forma automática algunos de los defectos debidos a la naturaleza de las folksonomías o a etiquetados de baja calidad. Como se ha dicho, dificultades relacionadas con el manejo de etiquetas se deben a la falta de control de sinonimia y homonimia, o de las variaciones del lenguaje: una mejora a estudiar es, por tanto, sistemas que permitan suplir estas deficiencias de forma automática.

Este apartado tratará de dar una panorámica sobre diversos sistemas estudiados para la mejora de los folksonomías mediante mecanismos automáticos.

2.4.2.1. Agrupaciones

Como se mencionó anteriormente, dentro de las folksonomías, es dificultoso obtener todos

los recursos relevantes de un tema: recursos relacionados pueden estar etiquetados con palabras distintas, si bien relacionadas, como por ejemplo “cine” y “película”. Esta dificultad entronca con el problema de la sinonimia, pero no se limita a él: etiquetas de distinto significado pueden hacer referencia a conceptos dentro de un mismo ámbito de interés, como “zoo”, “animal” y “elefante”. Esto pudiera mejorarse permitiendo mostrar una panorámica de las etiquetas ya existentes en el sistema, reunidas en grupos significativos que permitan una rápida visión general. Digamos, ofrecer un “mapa” del conocimiento existente, para que el usuario guíe su exploración por donde más le interese.

Siempre tratando de aportar un tanto más de estructura y significado que ayude al usuario en sus tareas de búsqueda y navegación, existen varias propuestas e implementaciones para la creación de agrupaciones automáticas de etiquetas a partir de las existentes en el sistema en un determinado momento. Un ejemplo pueden ser los clústeres de Flickr, como se ve en la siguiente ilustración:

Explorar / Etiquetas / apple / clústeres



[mac](#), [macintosh](#), [ipod](#), [computer](#),
[powerbook](#), [laptop](#), [macbook](#), [imac](#), [ibook](#), [osx](#)

→ [Ver más en este clúster...](#)



[fruit](#), [red](#), [food](#), [green](#), [macro](#), [orange](#),
[apples](#), [banana](#), [yellow](#)

→ [Ver más en este clúster...](#)



[nyc](#), [newyork](#), [newyorkcity](#), [manhattan](#)

→ [Ver más en este clúster...](#)

Ilustración 16: Clusters generados en Flickr para fotos con la etiqueta 'apple'

Flickr ofrece el servicio de agrupar los recursos etiquetados con una misma palabra (“apple” en la ilustración anterior) en grupos significativos. Como puede verse, este sistema resuelve la polisemia de la etiqueta, y separa claramente cada uno de los significados en áreas distintas, agrupando tanto los recursos en que se aplica como otras etiquetas que suelen utilizarse junto a ésta (co-ocurrentes).

Para el desarrollo de esta estructura, diferentes *papers* han aportado una serie de algoritmos, básicamente similares en el acercamiento [24][25]. Los pasos principales son los siguientes:

- 1. Crear grafo de etiquetas relacionadas:** Primeramente, se tratará de crear una estructura general de todas las etiquetas del sistema, a modo de grafo, en que las etiquetas se unan entre sí con enlaces con valor numérico asociado, tanto más alto cuanto más relacionadas. Para ello, se suele recomendar basarse en la co-ocurrencia de etiquetas que, como ya se comentó en la sección “2.3.6.3 Análisis de co-ocurrencia”, es una medida fiable para

descubrir elementos relacionados. Así, en este primer paso, quedarían unidas las etiquetas utilizadas juntas en algún recurso, con enlaces con valor igual al número de ocasiones en que hayan aparecido unidas.

- 2. Normalizar el grafo:** El número de ocasiones que dos etiquetas hayan aparecido unidas es una medida engañosa para el grado de unión, pues siempre recompensará a las etiquetas más populares, que más tiendan a usarse, frente a las minoritarias. Por ello, es necesario establecer una normalización, mediante una medida de similaridad, que indicaría los nuevos valores de los enlaces. Existen diferentes fórmulas para calcular esta similaridad, pero la mayoría coincide en ponderar la co-ocurrencia, con el número de apariciones de las etiquetas en solitario. Por ejemplo, la similitud de *Dice* de dos etiquetas A y B, se calcula como $P = (2 |A \cap B|) / (|A| + |B|)$.
- 3. Calcular clusters:** Obtenido el grafo, sólo queda aplicar técnicas de cálculo automático de *clusters* o grupos para descubrir las agrupaciones adecuadas. Algoritmos clásicos para tal tarea pueden ser los mapas de Kohonen o k-medias, pero no se recomiendan al necesitar especificar el número de *clusters* buscado. En los *papers*, se recomienda el algoritmo de bisección espectral.

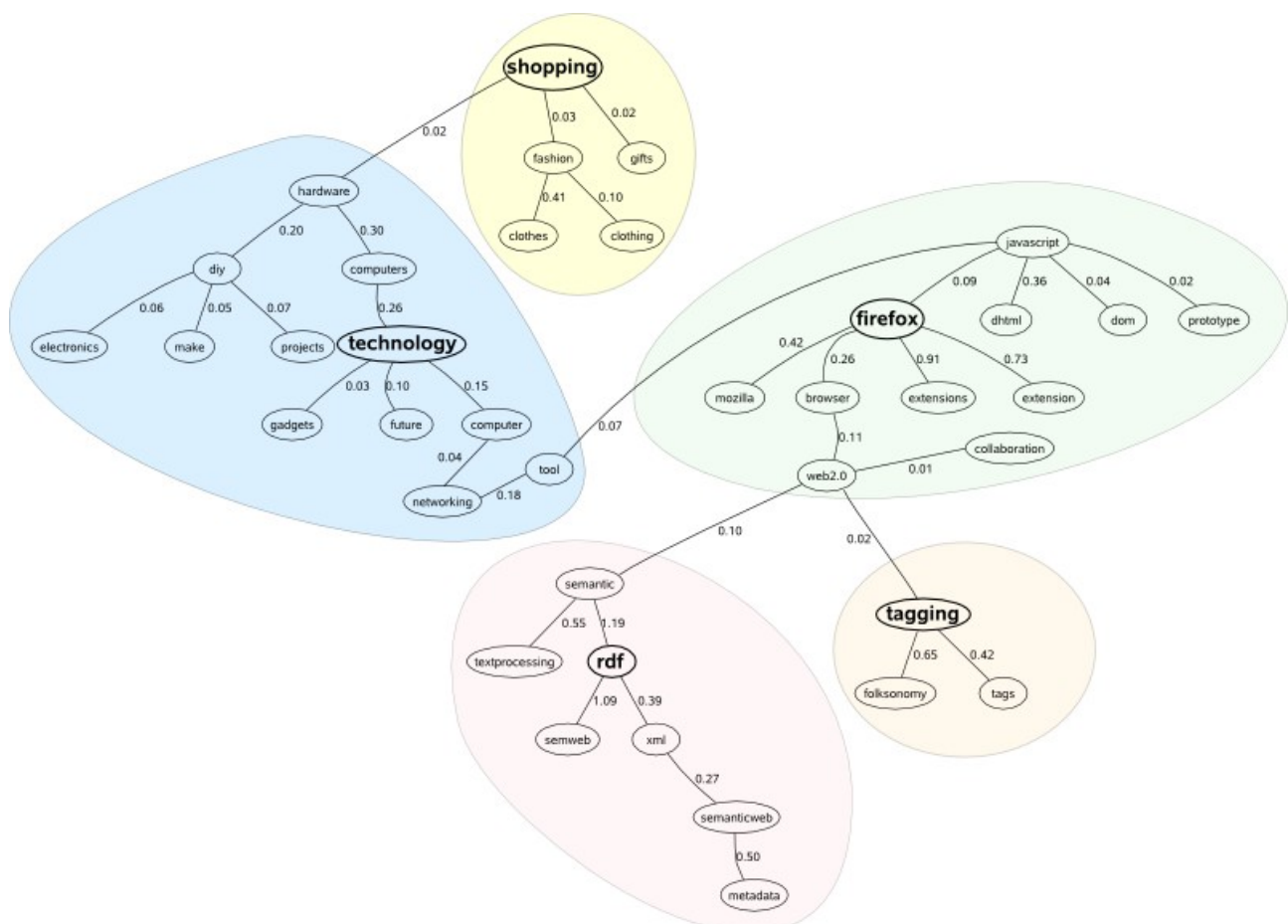


Ilustración 17: Ejemplo del resultado del algoritmo de agrupado

Otro problema relacionado de posible solución mediante este acercamiento es el de la ambigüedad de las etiquetas, como se mostró en [26], donde trató de descubrirse el significado de etiquetas ambiguas comprobando su relación de co-ocurrencia con otras etiquetas. Se plantea el caso de la etiqueta *sf*, iniciales de “*science-fiction*” o bien de “*San Francisco*”. En un grafo planteado como el anterior, esta etiqueta posiblemente quedase colocada a medio camino entre dos agrupaciones centradas en esos dos conceptos. Sería mediante el resto de etiquetas co-ocurrentes, comprobando hacia cuál de estas dos agrupaciones se acercan más, con las que podríamos decidir el verdadero significado de la etiqueta.

2.4.2.2. Creación de estructuras jerárquicas o facetadas

Otro campo que ha generado importante investigación ha sido el modo de generar automáticamente una estructura de clasificación clásica a partir de la masa de etiquetas de una folksonomía. Esto permitiría suplir varios de sus defectos, principalmente organizar de un cierto modo la masa de etiquetas y suplir su falta de jerarquía. En un ideal grado máximo de desarrollo, este sistema aportaría todas las ventajas de los sistemas de clasificación clásicos, como la organización formal o la falta de ambigüedad, y todas las de las folksonomías, como la adaptabilidad y el fácil crecimiento, evitando los inconvenientes de ambas partes. Estudios como [27] han aportado datos sobre su aplicabilidad y ventajas en los sistemas folksonómicos.

Varias han sido las propuestas para la creación de una ontología a partir de una folksonomía, que aportaría este grado de estructura del que carecen. En las ciencias de la información, se denomina *ontología* a la representación formal de una serie de conceptos incluidos en un determinado dominio y sus relaciones. Suelen incluir información tal como clases de objetos, atributos de las mismas, posibles restricciones, o incluso reglas para la inferencia lógica de conocimiento. Son de utilidad para la especificación formal, sin ambigüedad, de las características de un determinado dominio.

Así, en [28] se estudió la creación de una ontología facetada sobre un conjunto limitado de datos tomados de Flickr. Para su creación, se siguió un algoritmo basado en el estudio de la co-ocurrencia de etiquetas, concepto que ya hemos visto utilizado para la creación de agrupaciones. En este caso, se busca descubrir cuándo la definición de una etiqueta A contiene a otra etiqueta B, del mismo modo en que la idea “animales” pudiera contener el concepto “león”. Al identificar una relación semejante entre etiquetas, en la estructura de la jerarquía se colocaría por encima el término mayor (“animales”), hijo del cual sería el término menor (“león”).

La fórmula dada en este documento para el descubrimiento de una relación tal es:

Siendo X e Y dos etiquetas, y T un valor real entre 0 y 1, se deduce que Y es término contenido en X si:

$$P(X|Y) \geq T, \text{ y } P(Y|X) < T$$

$P(x|y)$ hace referencia a la probabilidad de X condicionada a Y, esto es, la probabilidad de que en una determinada asignación de etiquetas sobre un recurso se utilice la etiqueta X cuando ya se ha utilizado Y. El valor del umbral T puede tomarse a voluntad, pero debe recordarse que valores

altos (sobre 0,9) reducirán el error pero también el número de pares propuestos, e inversamente cuanto más se reduzca.

Uno de los mayores desarrollos fue llevado a cabo por la desaparecida página Rawsugar (cuyo dominio sigue activo, pero ha cambiado totalmente de propósito). Fue ésta una página al modo de del.icio.us o Technorati, destinada a la clasificación de enlaces web y entradas a weblogs. La distinguía del resto una serie de funcionalidades verdaderamente innovadoras, destinadas a facilitar la navegación por la estructura de etiquetas de la página de un modo similar al que pudiera hacerse en un sistema facetado clásico. La estructura de facetas era generada automáticamente por la misma página a partir de las etiquetas de los usuarios.

Tag Hierarchy

- ◆ Use the floating menu to create new tags or delete, rename, copy, cut and paste an existing tag. You do this by putting the mouse over the tag you wish to edit, and selecting the operation from the floating menu.
- ◆ Organize your tags in a hierarchy.



Ilustración 18: Creación de jerarquías personales de etiquetas en Rawsugar

Entre las opciones más innovadoras, estaba el permitir a los usuarios la creación de jerarquías propias para sus etiquetas, como puede verse en la ilustración anterior. Estas jerarquías servían al sistema para mejorar las estructuras facetadas que presentaba para la organización de los recursos del usuario y de la comunidad.

Como puede verse en la ilustración inferior, esta agregación de etiquetas permitía mostrar al usuario un mapa de los temas principales presentes en la folksonomía para guiar su navegación. Una vez seleccionado uno de ellos, podía del mismo modo presentar una estructura facetada para el refinamiento de la búsqueda, organizando así los resultados de la búsqueda.

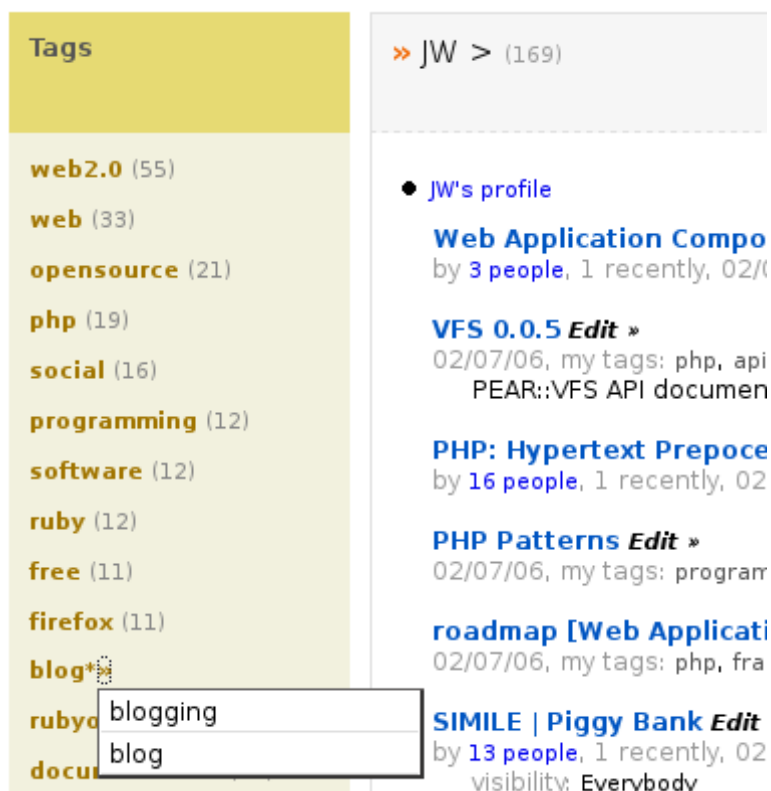


Ilustración 19: Facetas creadas a partir de etiquetas en Rawsugar

Ofrecía también servicios para la creación de organizaciones facetadas para los weblogs de los usuarios: a partir de su URL, Rawsugar leía las etiquetas utilizadas para cada entrada del blog, y generaba a partir de ellas una estructura que podía servir para navegar por el mismo, que el usuario podía incluir en su blog.

Por todo esto, Rawsugar fue una página de gran interés, que ofrecía unos servicios que no son actualmente ofrecidos por ninguna otra, y cuya desaparición por motivos económicos es de lamentar.

2.4.2.3. Recomendación de etiquetas

Otro campo de amplia investigación es el de las recomendaciones de etiquetas, que busca ayudar al usuario en la tarea de etiquetado, tratando de hacerla más sencilla y aportar ideas para la consecución de un etiquetado de mayor calidad. También, llevado a un extremo de desarrollo actualmente no alcanzado, permitiría el etiquetado automático de recursos, prescindiendo de la comunidad de usuarios en casos en que ésta no existiese o no fuese lo suficientemente amplia.

En realidad, las recomendaciones de etiquetas no son una propuesta de futuro, sino que ya en mayor o menor grado son incluidas por la mayoría de los sitios folksonómicos. Los listados de etiquetas más utilizadas para todo el sistema o para un tipo concreto de recursos son un buen

ejemplo. Del.icio.us, por ejemplo, muestra en el momento del etiquetado una serie de etiquetas utilizadas por otros usuarios para el mismo recurso, en caso de que éste ya esté presente en el sistema. La idea es adecuada: muy probablemente, algunas de ellas sean ya válidas para el usuario, eminentemente aquéllas dedicadas a la descripción de su contenido, como quizá puedan ser para la URL de un tutorial de Ajax “*tutorial*” y “*ajax*”.

Más allá de estas recomendaciones sencillas, las fuentes propuestas para las recomendaciones han sido muy diversas. Una posible fuente es la del análisis del contenido del recurso, tratando de obtener de forma automática palabras significativas que puedan encajar con el mismo. El problema que tiene esto asociado es que estas soluciones deben ser planteadas específicamente para cada tipo de recurso, pues cada uno requiere un tipo de proceso diferente: p.e., las técnicas de proceso de imágenes difieren totalmente de aquellas utilizadas para textos. De este modo, no pueden ser pensadas como soluciones generales para los sistemas folksonómicos, sino en exclusiva para aquellos dedicados al manejo de recursos del tipo estudiado.

Un ejemplo de este procedimiento es AutoTag [29], para la recomendación de etiquetas para entradas de weblogs. En su funcionamiento, ante una nueva entrada de blog, trata de obtener otras entradas parecidas de otros blogs, y una vez localizadas, obtener las etiquetas utilizadas para ellas, y devolver como recomendaciones las mayoritarias. El esquema puede ser válido para otros tipos de recursos, pero las técnicas para descubrir elementos similares habrán de ser necesariamente diferentes; en este caso, técnicas para análisis de textos son aplicadas.

Otra fuente habitual para las recomendaciones de etiquetas es el estudio de la co-ocurrencia. Tal y como hemos visto en el subapartado anterior correspondiente, un modo fiable para agrupar las etiquetas según su grado de relación es estudiar el grado de co-ocurrencia de las mismas, el número de veces en que ambas son utilizadas conjuntamente para un mismo recurso. Metodologías para la generación de recomendaciones a partir de esta información pueden verse en por ejemplo [17], donde a partir de unas etiquetas iniciales proporcionadas por el usuario, se generan otras como recomendaciones calculando las etiquetas con mayor co-ocurrencia para cada una de las iniciales, y devolviendo las que más se repitan.

Otras muchos sistemas de recomendación han sido planteados, como puede ser ejemplo [3], donde se estudian las posibilidades de sistemas de recomendación de etiquetas basados en las opiniones de usuarios de comportamiento similar al usuario en cuestión, y en un algoritmo basado en el PageRank de Google, tratando de dar puntuaciones de calidad a etiquetas y usuarios, tomando como supuesto que un usuario valioso tenderá a utilizar etiquetas valiosas. Como se ve, las posibilidades son muchas.

2.4.2.4. Mejora en las búsquedas

Otra idea fuente de investigación para la mejora de las folksonomías es el tratar de mejorar los sistemas de búsqueda de las folksonomías, buscando obtener en las búsquedas unos grados menores de *ruido* y *silencio* (esto es, documentos no relevantes obtenidos como resultado, y documentos relevantes no obtenidos en los resultados, respectivamente). Reflexionando, podemos ver que esta propuesta se relaciona de un modo íntimo con alguna de las anteriores, como la recomendación de etiquetas: esta última busca en última instancia una mejora en el etiquetado, una

mejor de selección de etiquetas por los usuarios tratando de recoger en éstas un mayor grado de información. Y en última instancia, estas mejores etiquetas buscan que los recursos a los que hacen referencia sean más fácilmente recuperables por otros usuarios, y más fácilmente relacionables entre sí; objetivos éstos, buscados por la mejora en los sistema de búsquedas y recuperación de información.

Varias propuestas han surgido para acometer esta mejora, de diversa naturaleza. Algunas tratan de adaptar el algoritmo de posicionamiento de Google, PageRank, a la peculiar estructura de las folksonomías, bautizando este nuevo algoritmo como FolkRank [30]. Mediante él, se busca obtener una noción de la “relevancia” de un recurso, de su validez y posible importancia como resultado a las consultas de un usuario; esta noción de relevancia no está en principio presente en la estructura misma de la folksonomía, y por ello supone un avance.

Las ideas básicas de PageRank y FolkRank son análogas y sirven para su comprensión. La idea básica que mueve el algoritmo de PageRank es que una página de importancia apunta mediante sus enlaces usualmente a otras páginas de importancia, de modo que un valor para ese grado de relevancia puede propagarse de una página a otra a través de sus hiperenlaces. Aplicado a las folksonomías, y por tanto principio de FolkRank, esto se traduce en que un recurso al que se le aplican etiquetas relevantes por usuarios relevantes, se convierte en importante.

En otros campos, también se ha buscado conseguir la desambiguación de los resultados de búsquedas ante casos de homonimia y sinonimia, mediante la obtención de información suplementaria sobre los recursos. Un ejemplo es [31], orientado a los sistemas de clasificación de imágenes, en que se trata de enriquecer los resultados de sistemas de agrupamiento y búsquedas mediante el procesamiento de los contenidos de las mismas imágenes, asumiendo que en conjuntos acotados (p.e. imágenes que comparten una misma etiqueta) esta información adicional puede resultar significativa.

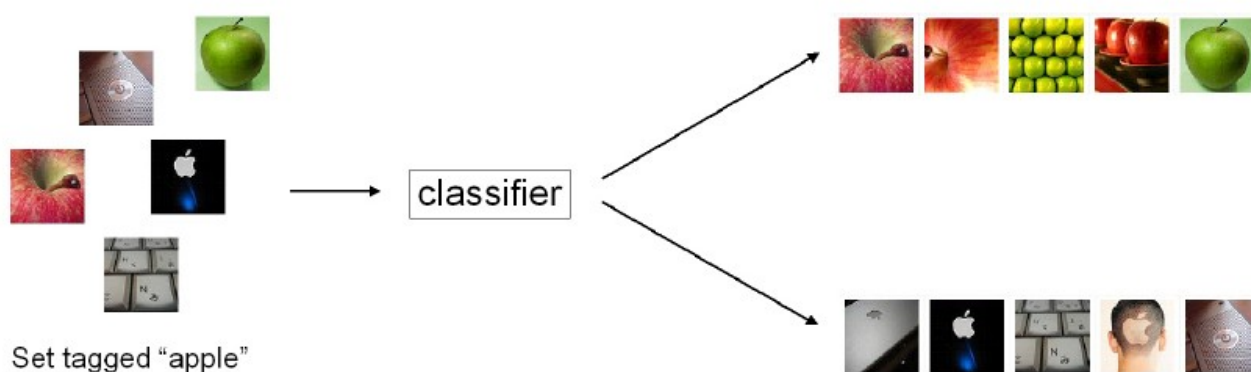


Ilustración 20: Separación de imágenes etiquetadas como "apple" según su contenido gráfico

También es de interés señalar la investigación motivada por el *Cross Language Evaluation Forum* (CLEF [32]). Este foro promueve la investigación en temas relacionados con la recuperación de información en sistemas multilingües, de los que podemos ver como ejemplo sistemas folksonómicos grandes con usuarios de todo el mundo. Organiza campañas anuales, planteando una

serie de objetivos orientados a la evaluación de diferentes aspectos de la recuperación de información, especialmente enfocados a las denominadas búsquedas *cross-language*, esto es, consultas ejecutadas en un determinado idioma, buscando objetos descritos en otros idiomas distintos.

Este foro, que inició sus actividades en el año 2000, busca potenciar la investigación en estos campos, planteando retos en cada una de sus ediciones progresivamente más complejos y variados. Muchos experimentos de interés han sido ejecutados a su abrigo, como por ejemplo estudios de usabilidad en búsquedas multilingües sobre Flickr. Estas ideas pueden ser de interés para la resolución del problema de los múltiples idiomas presentes en las folksonomías, que conllevan situaciones de homonimia y sinonimia a controlar.

3. Descripción del experimento

Analizado ya el grado de desarrollo actual de las folksonomías y sus diversos aspectos, plantearemos en este apartado el trabajo a ejecutar en este proyecto. Pretendemos contribuir al trabajo en evolución de mejora de las folksonomías, presentado en el capítulo anterior.

De todos los diversos ámbitos de mejora de las folksonomías, hemos decidido centrarnos en el campo de recomendación automática de etiquetas a los usuarios. Lo hemos considerado de particular interés por los diversos beneficios que puede aportar en varios aspectos de las folksonomías:

- Resulta de ayuda para superar las iniciales dudas del usuario ante el etiquetado, facilitándole una serie de sugerencias que considerar en sus etiquetas, que pueden enriquecerlas e incluso inspirarle la inclusión de otras aparte de las sugeridas. Esto ya minimizaría los casos de recursos poco o nada etiquetados que suelen ser frecuentes en las folksonomías.
- Contribuyen de este modo a la mejora de etiquetado de los recursos, aportando ideas para que las etiquetas cubran un ámbito semántico más amplio. Se enriquece así el contenido de las folksonomías, obteniéndose mejores resultados en operaciones tales como, por ejemplo, el cálculo de etiquetas relacionadas o recursos relacionados.
- Con la mejora del etiquetado, la recuperación de información se hace más sencilla, pues los recursos tendrán mayor información asociada, haciendo más fácil su búsqueda.

Para enfocar aún más nuestro trabajo, se decidió que se ejecutaría sobre una folksonomía para imágenes. El motivo de esta elección está en su mayor dificultad para la recuperación en los sistemas actuales de búsqueda, basados en su mayoría en búsquedas textuales: actualmente la posibilidad de recuperación de una imagen pasa por la inclusión de metadatos u otro tipo de descripciones textuales, interpretables por estos buscadores; la búsqueda de imágenes entonces pasaría por la búsqueda de textos dentro de esos metadatos coincidentes con los de la consulta, y devolver las imágenes que éstos tengan asociadas. Así pues, habida cuenta de que las etiquetas son una forma de estos metadatos para la búsqueda, la mejora en los mismos supone la mejora en su posibilidad de recuperación.

Partiendo de esta base, centraremos nuestra atención en la utilidad que los usuarios pueden sacar de estas herramientas de recomendación. Es éste un factor importante, pues en última instancia la ventaja que pueda obtenerse de tales herramientas podrá verse en si éstas son usadas por los usuarios, y si estos toman sus recomendaciones para el etiquetado de sus imágenes. Queremos conocer también, dado un conjunto de ayudas para el etiquetado, cuál es el manejo de las mismas por los usuarios, si son ignoradas o utilizadas, y qué provecho sacan de ellas.

Así pues, el acento del experimento estará puesto en descubrir el comportamiento de los usuarios al ejecutar una serie de tareas (añadido y búsqueda de imágenes) sobre un sistema folksonómico para la clasificación de imágenes, qué etiquetas eligen, qué ayudas utilizan y de qué

modo. En ello podrá también descubrirse si las herramientas concretas de ayuda propuestas en el experimento les resultan de utilidad, si se sirven de ellas y utilizan sus sugerencias.

3.1. Antecedentes

Para la definición de este experimento, se tomó como referencia relevante el conjunto de estudios llevados a cabo bajo iCLEF [33], parte de la conferencia anual CLEF [32]. La conferencia CLEF (*Cross-Language Evaluation Forum*) es un foro anual para la evaluación de sistemas de recuperación de información multilingüe; muchos estudios suelen ejecutarse a su abrigo, tomando como guía una serie de tareas planteadas anualmente por este foro para establecer los puntos comunes de estudio de los trabajos.

CLEF se divide en una serie de *tracks* o áreas temáticas, que investigan diferentes aspectos de la recuperación de información en varios idiomas; así, las búsquedas de imágenes son estudiadas en el *track* ImageCLEF, la búsqueda de información en la web en WebCLEF, o los sistemas de respuesta a preguntas en QA@CLEF. El *track* antes mencionado, iCLEF, se especializa en la interacción de los usuarios, las interfaces mediante las cuales se ofrece a los usuarios los servicios de búsqueda multilingüe. Se busca maximizar la utilidad de estas interfaces, tratando de que sean cada vez más intuitivas y sencillas de manejar, y faciliten el uso de los servicios de recuperación a los usuarios.

En particular, fijaremos nuestra atención en los experimentos desarrollados bajo el abrigo de iCLEF 2006. En él, la tarea presentada era el desarrollo de una interfaz para la búsqueda de imágenes de forma multilingüe sobre los datos de Flickr. Los resultados esperados de los estudios eran la generación de estas interfaces, que servirían de nuevo punto de entrada a los servicios de Flickr, y la recogida de datos de la interacción de los usuarios mediante las mismas, para el estudio de su comportamiento al llevar a cabo una serie de tareas que les es planteada. Como se ve, el énfasis está puesto en el estudio del proceso, no tanto en los resultados obtenidos de la ejecución de las tareas.

Se pedía a los usuarios la ejecución de una serie de tareas de búsqueda de imágenes, de diferente naturaleza y objetivos, que eran fijas para todos los estudios ejecutados bajo iCLEF 2006. Estas tareas eran:

1. **Recuperación basada en un tema:** Encuentre la mayor cantidad de edificios del parlamento europeo que le sea posible, sean de su interior o exterior.
2. **Recuperación creativa y abierta:** Encuentre cinco ilustraciones para el artículo “La historia del azafrán”. (Se incluía su texto).
3. **Tarea orientada visualmente:** ¿Cuál es el nombre de la playa en que se encuentra este cangrejo? (Se incluía la foto de un cangrejo sobre la arena de una playa. La imagen podía encontrarse en la base de Flickr, y en su descripción se indicaba el nombre de la playa. La particularidad era que la imagen estaba etiquetada en alemán).

Como se ve, trata de reflejarse en estas tareas diferentes modos de búsqueda recurrentes en el uso de Flickr u otras bases de información del mismo tipo. Así, las tres tareas se enfocan a la exploración de la colección pero con diferentes grados de libertad: limitada a un tema en la primera tarea, a lo que pueda sugerirle al usuario un texto en la segunda, y finalmente a la búsqueda de una imagen concreta.

Entre los experimentos presentados bajo esta base que fueron publicados en la conferencia, se ejecutaron estudios sobre la actitud de los usuarios de cara a la búsqueda multilingüe ([34]), ofreciéndoles ejecutar sus búsquedas para las tareas antes indicadas con herramientas de traducción automática y opciones de “no traducir”, “traducción automática” y “traducción asistida” (dejando elegir al usuario qué palabras traducidas tomar para sus búsquedas); también sobre la búsqueda multilingüe para usuarios árabes [35], o sobre la confianza que los usuarios mostraban en los resultados de tales búsquedas [36].

3.2. **Objetivo del experimento**

El objetivo principal del experimento es analizar el comportamiento de usuarios reales de una folksonomía basada en imágenes ante una serie de herramientas de recomendación de etiquetas, con el fin de descubrir hasta qué punto estas herramientas son de ayuda en sus acciones, cuándo y cómo son utilizadas, y aspectos relacionados.

El experimento consiste en varios elementos, que describimos a continuación para su total comprensión:

- El experimento requiere la intervención de una serie de voluntarios que ejecuten una serie de tareas relacionadas con acciones comunes dentro de las folksonomías de imágenes (p.e. Flickr), como son el añadido de una nueva imagen etiquetada, o la búsqueda de otra imagen. Estas tareas no requerirán en ningún caso de un esfuerzo especial por estos usuarios, y se calcula que el tiempo máximo que deberían llevarles sería de 20 minutos, para evitar su tedio y con él la obtención de resultados sesgados.
- El experimento se realizará a través de una interfaz gráfica, accesible a través de web para la mayor comodidad de los usuarios. Esta interfaz les presentará las tareas a ejecutar y diversas indicaciones para su total entendimiento, y les facilitará herramientas de recomendación de etiquetas para su uso si así lo desean.
- Esta interfaz recogerá los datos de la interacción de cada usuario en ficheros de *log* analizables posteriormente. Cuando todos los usuarios esperados hayan ejecutado las tareas, se tomarán estos ficheros y procesarán.
- Con los datos de cada interacción, se obtendrán conclusiones acerca del comportamiento de los usuarios y el uso que dan a las herramientas de recomendación: cuántas etiquetas utilizan por imagen, cuánto tardan en etiquetarlas, si utilizan las sugerencias de las herramientas y en qué medida, qué acciones ejecutan y en qué orden para las búsquedas, etcétera.

3.3. Colección de imágenes

En nuestro experimento, los usuarios deberán añadir imágenes y buscar otras dentro de una colección previa. Se ha de definir de dónde se tomará tal colección y cuáles serán sus características.

El origen de las mismas debería ser sitios folksonómicos dedicados a la clasificación de imágenes, para poder tomar además de las imágenes una serie de datos sobre las mismas, principalmente una serie de etiquetas, y de ser posible un título y descripción. De entre los varios sitios web existentes que cumplen estas condiciones, se pensó en el uso principal de Flickr, posiblemente la folksonomía de imágenes más importante, popular y extensa de Internet, con muchos servicios asociados que facilitarían el desarrollo.

Otra decisión tomada fue que, difiriendo de la elección de los experimentos de iCLEF 2006, nuestro experimento no hará consultas directamente sobre el repositorio de Flickr, sino que se recolectaría una cantidad suficiente de imágenes con sus datos asociados, y serían guardadas localmente en nuestros ordenadores. La principal motivación para esto estaba en los requisitos técnicos de las ayudas de recomendación de etiquetas a utilizar, que en más de un caso requerían el procesamiento de ficheros guardados localmente.

Decidido que las imágenes deberían guardarse localmente, faltaba definir el tamaño de la colección. Se optó por una colección más bien extensa, permitiendo que las herramientas de recomendación, varias de ellas basadas en el estudio del etiquetado previo para las sugerencias futuras, contasen con suficiente información para unos resultados adecuados. Se estableció este número en 20.000, lejos de los miles de millones de Flickr pero suficientemente extensa para el experimento. Estas imágenes se obtendrían de una forma aleatoria, tratando de mantener únicamente los siguientes requisitos:

- Toda imagen debería tener al menos dos etiquetas asignadas.
- En el total de imágenes, deberán distribuirse a partes iguales las imágenes etiquetadas en los siguientes cuatro idiomas: inglés, español, francés y alemán. Esto es, una cuarta parte de las imágenes etiquetadas en inglés, otra cuarta parte en español, etcétera.

El primer requisito trata de evitar obtener imágenes muy pobremente etiquetadas, que no aportarían información a las herramientas de ayuda, y el segundo mantener un equilibrio entre los idiomas representados en la colección, pensando que la pluralidad de idiomas suele ser frecuente en sistemas folksonómicos grandes.

De este modo, se decidió que la colección inicial de imágenes del experimento se compondría de 20.000 imágenes extraídas de otros sitios web basados en folksonomías para imágenes, eminentemente Flickr. De estas imágenes, se guardarán las siguientes informaciones:

- **URL:** URL donde puede encontrarse la imagen original.
- **Etiquetas:** Etiquetas asignadas a la imagen.
- **Título:** Título de la imagen, en caso de que exista.
- **Descripción:** Descripción de la imagen, en caso de que se haya escrito una.

3.4. Ayudas al usuario

Como se ha dicho, el experimento tratará de comprobar la utilidad que encuentran los usuarios en una serie de herramientas de recomendación de etiquetas. Existen muy diversos tipos de herramientas de recomendación, por lo que debieron delimitarse aquellas a utilizar en experimento, tratando de abarcar un ámbito lo suficientemente amplio como para que quedase reflejada la utilidad de las mismas atendiendo también a su diferente naturaleza.

Finalmente, las ayudas que se decidieron presentar en el experimento son las siguientes:

- **Autocompletado de etiquetas:** Se presentará al usuario, según escribe una etiqueta para su imagen, las etiquetas más empleadas en todo el sistema que contengan las mismas letras utilizadas para su autocompletado, a modo de Google Suggest [37]. Así, podría indicarse tras la escritura de “pho”, “photo” como etiqueta más probable. Estas sugerencias no interferirían con el usuario si no se corresponden con sus deseos, y podrían evitar errores ortográficos en otros casos.
- **Análisis de las similitudes de la nueva imagen con otras ya almacenadas:** Mediante algoritmos de comparación de imágenes según color o textura, se conseguirá una serie de imágenes similares a una dada entre las de la colección, ofreciendo sus etiquetas como sugerencias para la nueva imagen.
- **Etiquetas relacionadas:** Partiendo de una base de imágenes ya etiquetadas, se estudiará la co-ocurrencia de las etiquetas utilizadas en las mismas (esto es, qué etiquetas suelen acompañar a otra etiqueta dada), y ofrecer como sugerencias aquellas etiquetas que se consideren muy relacionadas, mediante este criterio, a las ya incluidas por el usuario (por ejemplo, el usuario podría indicar “ajax” y el sistema sugerirle “javascript”, que tiende a utilizarse conjuntamente).
- **Traducciones automáticas:** Para facilitar la recuperación de imágenes etiquetadas en idiomas diferentes al del usuario, se ofrecerá al usuario un sistema para traducir de forma automática sus etiquetas a otros idiomas de interés. Los idiomas escogidos son español, inglés, francés y alemán.

3.5. Tareas

Se plantearán al usuario una serie de tareas a ejecutar, que le hagan interactuar con el sistema en busca de unos determinados objetivos y nos permitan estudiar sus patrones de comportamiento. Estas tareas serán de dos naturalezas principales, añadido de imágenes etiquetadas, y búsqueda de imágenes.

Las tareas de añadido de imágenes son las cuatro primeras. Dos serán los elementos que varíen de una a otra: el origen de la imagen a etiquetar, que podrá ser de la colección o del usuario, y el número de ayudas presente.

Hay varias motivaciones para estas variaciones: el pedir al usuario compaginar el etiquetado de imágenes personales con otras elegidas aleatoriamente nos permitirá comprobar posibles diferencias de comportamiento en estos casos, como posibles tendencias a utilizar una ayuda frente a otras. Con respecto al número de ayudas, se ha optado por irlo aumentando progresivamente hasta alcanzar el total de ayudas; con esto se pretende que en cada una de las primeras pruebas el usuario fije su atención y se familiarice con una herramienta nueva cada vez, de este modo habituándole paulatinamente a su manejo, y permitiendo estudiar las posibles diferencias de comportamiento según la presencia de unas herramientas u otras.

- **Tarea 1:** Se escogerá una imagen aleatoria de la colección inicial del sistema, y se pedirá al usuario que la etiquete. Contará con dos ayudas: el autocompletado de etiquetas y la detección de imágenes parecidas.
- **Tarea 2:** Se escogerá una imagen aleatoria de la colección inicial del sistema, y se pedirá al usuario que la etiquete. Contará con tres ayudas: el autocompletado de etiquetas, la detección de imágenes parecidas y las etiquetas relacionadas.
- **Tareas 3 y 4:** El usuario deberá enviar un fichero de imagen de su disco duro al sistema, y se le pedirá que la etiquete. Contará con todas las ayudas: el autocompletado de etiquetas, la detección de imágenes parecidas, las etiquetas relacionadas y la traducción automática.

Las tareas restantes están enfocadas a la búsqueda de imágenes. En las búsquedas, basándose las consultas en palabras sueltas escritas por el usuario, pueden ser también de utilidad las herramientas de ayuda, dando sugerencias al usuario sobre qué palabras incluir para la mejor obtención de resultados. Pretende comprobarse con este conjunto de pruebas cuál es ese grado de utilidad.

Las tres tareas de búsqueda cubren un cierto rango de comportamientos habituales de búsqueda en estos sistemas. Se centran en la búsqueda por exploración, esto es, sin objetivo concreto, sino tratando de obtener recursos relacionados con un tema; y la búsqueda de un elemento en concreto, una tarea habitualmente costosa en las folksonomías si se desconocen las etiquetas utilizadas para el etiquetado del recurso, y por tanto de especial interés para el estudio de su posible mejora.

- **Tarea 5:** Se escogerá una imagen aleatoria de la colección inicial del sistema, y se pedirá al usuario que busque cinco imágenes que considere relacionadas en algún sentido con ella. No se pone ninguna condición a esta elección de imágenes, el criterio es totalmente libre por parte del usuario.
- **Tarea 6:** Se escogerá una imagen aleatoria de las añadidas por otros usuarios en las tareas 3 y 4, y se pedirá al usuario que la etiquete. En caso de que el usuario no consiga por más que lo intente encontrar la imagen, se le dará la opción de abandonar la tarea pulsando un botón, y pasar a la siguiente.
- **Tareas 7:** Se escogerá una imagen aleatoria de la colección inicial del sistema, y se pedirá al usuario que la etiquete. En caso de que el usuario no consiga por más que lo intente encontrar la imagen, se le dará la opción de abandonar la tarea pulsando un botón, y pasar a

la siguiente.

3.6. Información guardada

De la interacción del usuario con el sistema, se guardarán en archivos de *log* datos sobre sus acciones, que posteriormente analizados permitirán obtener conclusiones acerca de su comportamiento en el manejo de folksonomías con herramientas de ayuda.

Entre los datos que quedarán registrados sobre cada ejecución figuran los siguientes:

- Nombre del usuario
- Hora de comienzo de cada prueba.
- Claves de las imágenes mostradas al usuario al inicio de cada prueba para su etiquetado o búsqueda.
- Todas las ayudas que fueron ofrecidas al usuario, indicando la hora en que fueron mostradas, y todas las sugerencias que mostraron al usuario.
- Datos de las imágenes añadidas por el usuario, clave asignada y etiquetas.

Además, en las tareas de búsquedas:

- Consultas ejecutadas por el usuario, con resultados obtenidos.
- Descripciones de imágenes vistas por el usuario.
- Imágenes seleccionadas para cada prueba.
- Momento de abandono de las pruebas si éste ocurre.
- Comentarios finales del usuario.

3.7. Conclusiones a obtener

A partir de los datos recogidos de cada interacción, se tratarán de obtener conclusiones acerca del comportamiento general de los usuarios en el uso de folksonomías con herramientas de sugerencia de etiquetas.

Estas conclusiones, para las tareas de añadido de imágenes, versarán sobre aspectos como: ¿cuántas etiquetas utiliza el usuario para cada imagen? ¿Son propias, o utiliza alguna de las sugeridas? Si es así, ¿provenientes de qué herramientas, y en qué medida? ¿Cuánto tarda en etiquetar?

En las tareas de búsqueda, ¿qué pasos da el usuario en la búsqueda de su imagen? ¿Utiliza las ayudas, y si es así, cuándo? ¿Llega a encontrar las imágenes cuando se le pide buscar una en concreto, y a través de qué pasos? ¿Abandona, y si es así, cuánto tarda en hacerlo?

En general, tratará de obtenerse una panorámica sobre cómo los usuarios actúan ante las herramientas de sugerencia y qué provecho obtienen de ellas. Estas conclusiones se detallan en la sección “5. Resultados del Experimento” de este documento.

3.8. Recorrido por la aplicación

En esta sección, mostraremos las pantallas de las que consta la aplicación del experimento, describiendo sus diferentes elementos y qué acciones se esperan de los usuarios.

3.8.1. Pantalla de login

La pantalla inicial de la aplicación sirve de presentación de la misma y de punto de partida para el usuario. Se compone de un texto inicial en su parte superior, que da las pautas generales y objetivos del experimento, tratando de poner en contexto al usuario, indicarle qué va a encontrar y qué debe hacer. Más allá de esto, el único requisito para la entrada a la aplicación es que indique un nombre de usuario, que servirá para identificarle, y pulse entrar.

Bienvenido al Experimento de Etiquetado de Imágenes

Queremos comprobar la utilidad que los usuarios encuentran en nuestras herramientas. ¡Muchas gracias por participar!

Este estudio trata sobre **imágenes etiquetadas**, es decir, imágenes a las que se les ha asignado una serie de **palabras** (etiquetas), elegidas de forma **totalmente libre**, para describirlas. Por ejemplo, la foto de un árbol podrá tener asociadas las etiquetas 'árbol', 'naturaleza' o 'verde'. Como idea, puede pensar en las etiquetas de las fotos de Flickr.

Hemos desarrollado **herramientas para sugerir posibles etiquetas** al usuario, y queremos saber si resultan de utilidad. Para ello, le pediremos que **etiquete y busque una serie de imágenes**, ofreciendo para ello estas ayudas. Se le pedirá **subir 2 imágenes propias**, así que prepare 2 ficheros (JPG, GIF o PNG).

La duración aproximada es de **20 minutos**. Para entrar, **escriba un nombre** para identificarle.

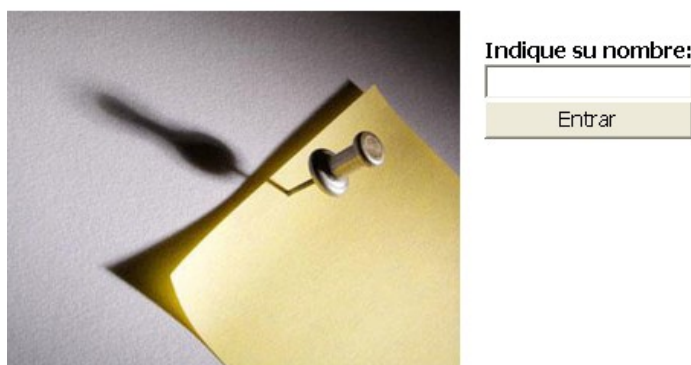


Ilustración 21: Pantalla de inicio de la aplicación

3.8.2. Tareas de añadir imágenes

Hecho esto, la primera pantalla que verá será la siguiente:

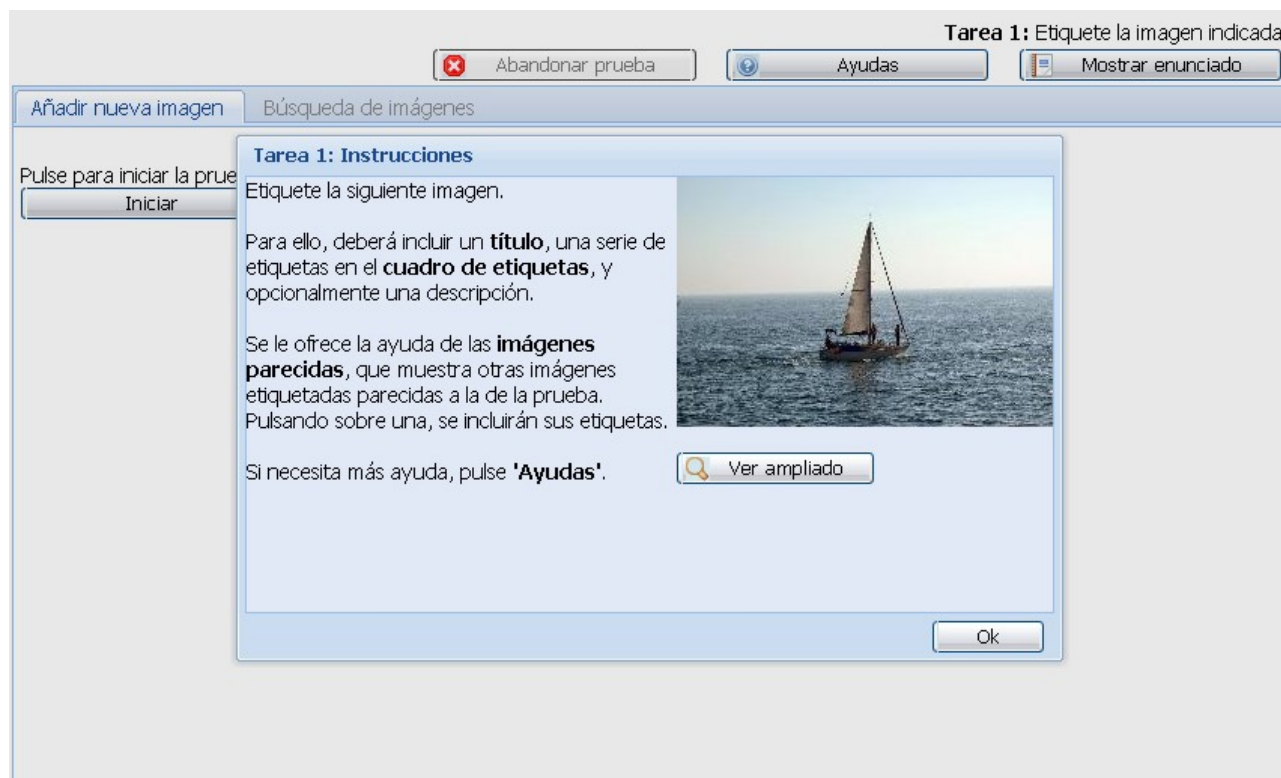


Ilustración 22: Enunciado de la tarea 1

Esta pantalla muestra el aspecto general de la interfaz de la aplicación. En primer plano se muestra el enunciado de la primera tarea. Este diálogo puede cerrarse pulsando OK. Una vez cerrado, podrá volver a accederse para consulta mediante el botón “Mostrar enunciado” de la esquina superior derecha.

En la franja superior de la pantalla pueden verse diversos elementos que dan un encuadre general al usuario. Un texto indica la tarea actual con una breve descripción, que puede completarse con el botón “Mostrar enunciado”, ya mencionado, para consultar en cualquier momento el enunciado de cada prueba. El botón de “Ayudas” le conducirá a un cuadro de diálogo con una serie de indicaciones de ayuda para la realización de las tareas. El botón de “Abandonar prueba” permitirá al usuario abandonar la actual prueba sin completarla y pasar a la siguiente, aunque estará sólo activo en las dos últimas tareas, de búsqueda de una imagen concreta.

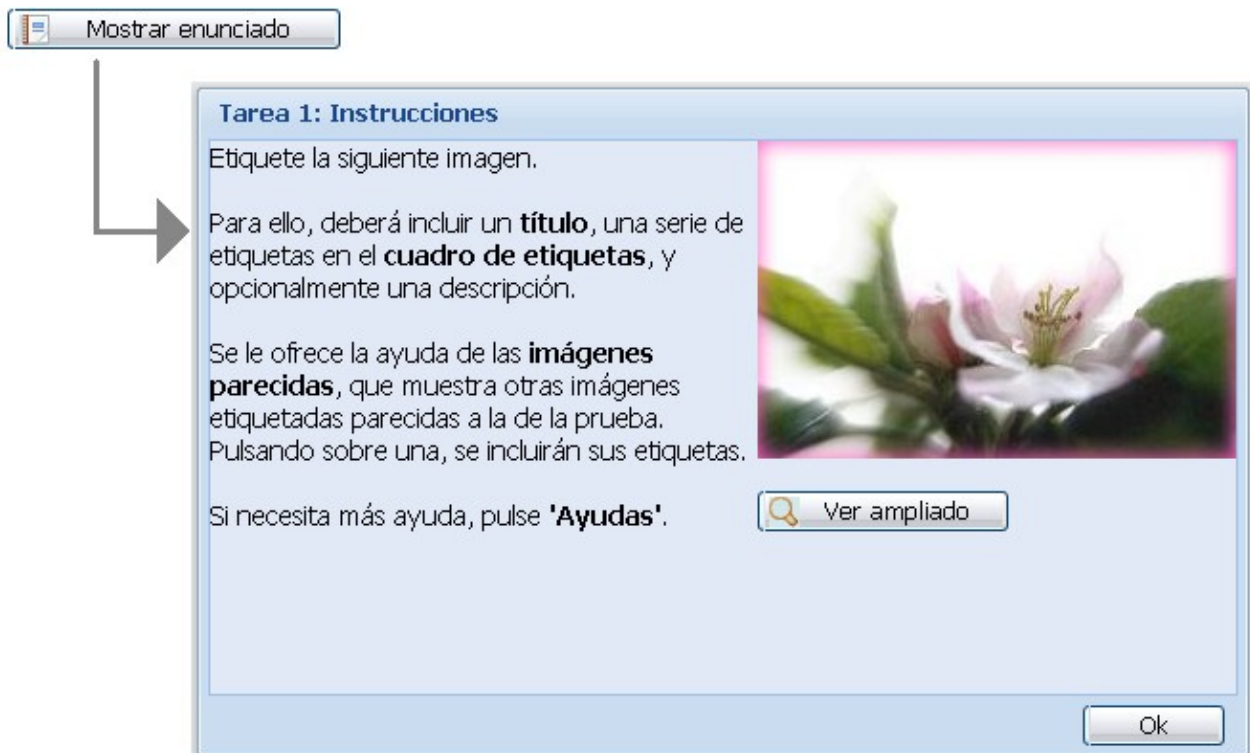


Ilustración 23: El botón "Mostrar Enunciado" muestra el enunciado de la prueba actual



Ilustración 24: El botón "Ayudas" muestra textos de ayuda sobre la aplicación

Una vez iniciada la primera prueba, el usuario se encontrará con la siguiente interfaz:



Ilustración 25: Interfaz para la primera tarea

En ella, se disponen una serie de elementos. La columna izquierda presenta en su parte superior dos botones, “Guardar” y “Eliminar imagen”, éste desactivado. El botón de “Guardar” permitirá al usuario guardar la imagen cuando haya dado por terminado su etiquetado; “Eliminar imagen”, en tareas que requieran que el usuario suba una imagen de su disco duro, permitirá eliminar la imagen enviada si el usuario quiere elegir otra. Más abajo, puede verse la imagen a etiquetar, y dos campos a rellenar, “Título” y “Descripción”, siendo éste último opcional.



Ilustración 26: Tabla para añadir etiquetas. A la izquierda, tabla con dos etiquetas añadidas. A la derecha, inserción de una nueva etiqueta, siendo sugeridas algunas posibilidades.

Las etiquetas pueden introducirse mediante la tabla de etiquetas, mostrada en la ilustración anterior. Las etiquetas pueden escribirse en el cuadro superior, siendo añadidas a la tabla mediante el botón “Añadir”. Para eliminar etiquetas ya introducidas, puede marcarse el *checkbox* de la izquierda de cada etiqueta, y pulsar “Eliminar marcados”.

En este sistema se presenta la primera ayuda al usuario. Al modo de Google Suggest [37], según el usuario escriba la etiqueta a introducir, el sistema le mostrará un listado con posibles etiquetas que puede escoger, tomadas de entre las etiquetas más utilizadas que contengan la cadena escrita por el usuario.

La otra ayuda presente en esta primera prueba es la de imágenes parecidas. El cuadro de esta ayuda mostrará al usuario una serie de imágenes presentes en la colección del sistema que se han juzgado parecidas a la imagen a etiquetar. El usuario puede pulsar sobre estas imágenes, haciendo con ello que las etiquetas asociadas a la misma se incluyan en las etiquetas para la imagen presente.



Ilustración 27: Ayuda de Imágenes Parecidas

Una vez pulsado el botón “Guardar”, la imagen se añadirá a la colección, y el usuario pasará a la **tarea 2**, que será similar a ésta pero añadiendo la ayuda de las etiquetas relacionadas.

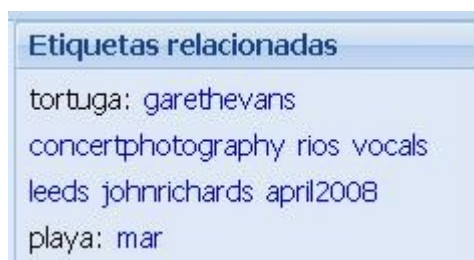


Ilustración 28: Ayuda de Etiquetas Relacionadas

Con esta ayuda, se presentan al usuario, para cada una de las etiquetas que haya utilizado, otras etiquetas que hayan sido utilizadas en conjunción con ellas en otras imágenes de la colección. Pulsando sobre una de estas etiquetas, se añadirá a las del usuario.

Las **tareas 3 y 4** siguen similar mecánica, pero con la particularidad de que las imágenes a etiquetar no serán escogidas aleatoriamente de la colección, sino que deberán ser escogidas por los usuarios desde sus ficheros. El cuadro inicial de selección de fichero es el que sigue:



Ilustración 29: Selector de fichero

Además, se añadirá una nueva ayuda, la de la traducción automática. Con ella, el usuario podrá traducir las etiquetas que haya utilizado a inglés, francés, alemán o español. Para ello, deberá escoger el idioma en que está escribiendo de un desplegable, y serán cargadas las traducciones en los idiomas restantes, que podrán ser añadidas a las etiquetas utilizadas pinchando sobre ellas.

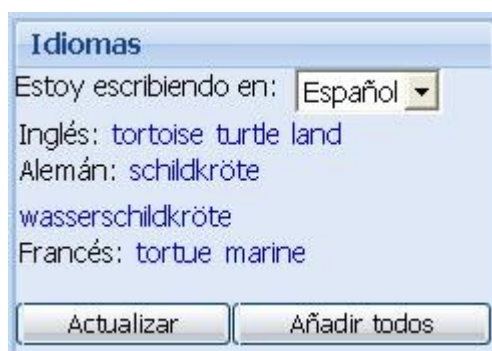


Ilustración 30: Ayuda de traducción automática

3.8.3. Tareas de búsqueda de imágenes

Tras estas cuatro tareas relacionadas con el añadido de imágenes, se entrará en el grupo de tareas de búsqueda de imágenes, compuesto de tres tareas en total. En ellas, la interfaz cambia, presentando al usuario un cuadro de texto con un botón “Buscar”, que le permitirá hacer búsquedas sobre la colección de imágenes del sistema.

En la siguiente ilustración puede verse esta interfaz, enseñando cómo se muestra una búsqueda y sus resultados. Para cada imagen resultante, se indica bajo ella su título, dimensiones y tamaño.

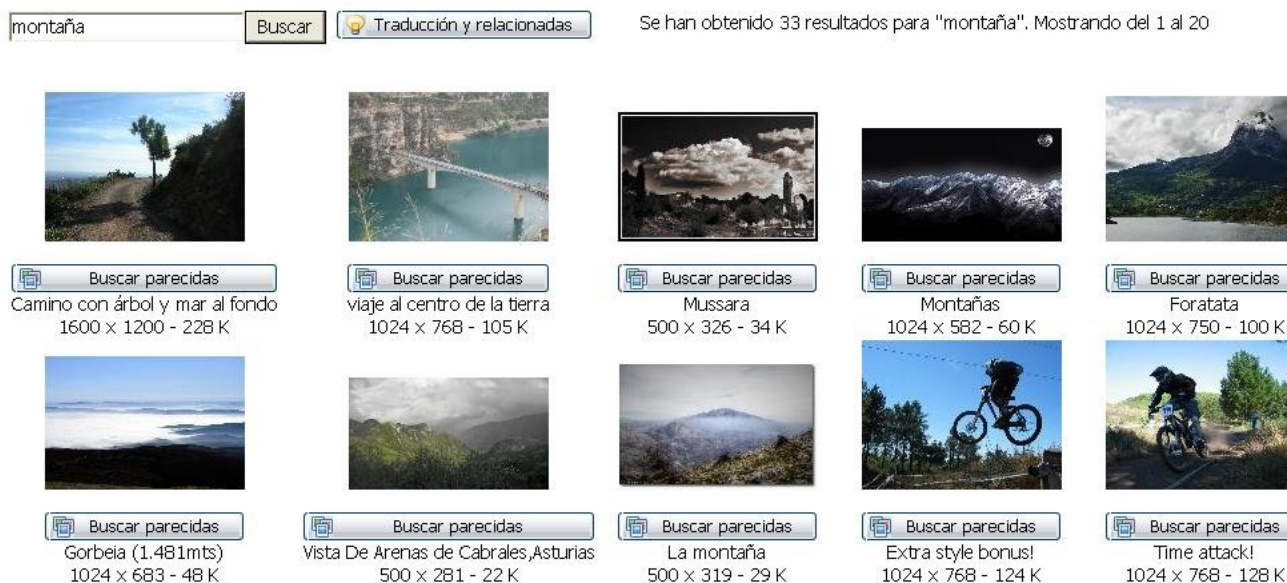


Ilustración 31: Búsqueda de imágenes

Pinchando sobre una de estas imágenes se abrirá una pestaña con su descripción:

viaje al centro de la tierra



Etiquetas:

- agua
- puente
- montaña
- embalse
- jalance

Ilustración 32: Detalles de imagen

En esta pestaña, las principales informaciones de la imagen son mostradas: así, título, etiquetas y descripción (en caso de que se haya escrito) se podrán ver junto a la imagen en sí. Tres botones permitirán una serie de acciones básicas: “Cerrar pestaña” permitirá al usuario cerrar esta descripción de la imagen cuando lo juzgue oportuno; con “Ver ampliado” podremos ver la imagen en su tamaño original; y finalmente “Es imagen buscada” permitirá al usuario señalar aquellas imágenes que cada tarea de la aplicación le haga buscar.

La primera tarea de búsquedas, **tarea 5**, propondrá al usuario una imagen elegida aleatoriamente de la colección del sistema, pidiéndole que explore las imágenes como mejor

considere, y seleccione cinco de ellas (pulsando su botón “Es imagen buscada”, ya indicado) que crea relacionadas con ella en algún sentido. La tarea es libre, como puede verse, y tiene el objetivo de invitar al usuario a explorar la colección para analizar sus preferencias a la hora de hacerlo.

Para ello, cuenta con una serie de ayudas a las búsquedas, las mismas que estaban presentes en las tareas de añadido de imágenes. Así, el cuadro de búsquedas también ofrecerá sugerencias para el completado de las palabras escritas, y las ayudas de traducciones y etiquetas relacionadas estarán disponibles en el botón “Traducción y relacionadas”.

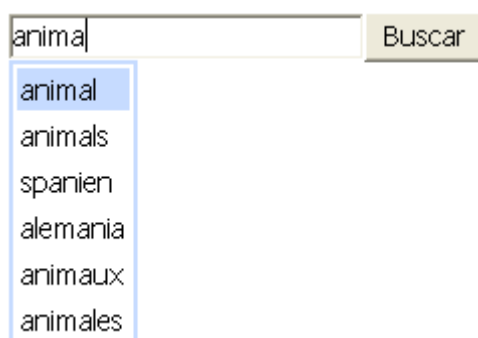


Ilustración 33: Autocompletado en el cuadro de búsquedas

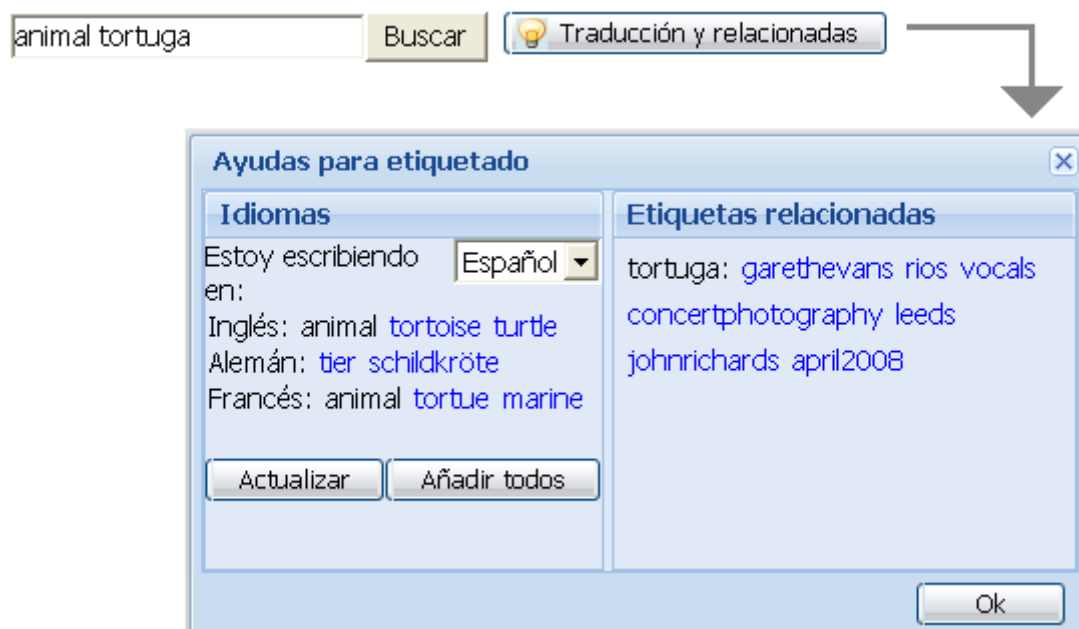
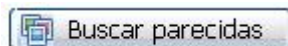


Ilustración 34: Ayudas de traducción y etiquetas relacionadas en tareas de búsqueda

El usuario también puede buscar imágenes parecidas a una concreta obtenida en los resultados de una búsqueda. Para ello, se mostrará un botón “Buscar parecidas” bajo cada resultado.



*Ilustración 35: Botón
"Buscar parecidas"*

Las **tareas 6 y 7** pedirán al usuario buscar sendas imágenes escogidas aleatoriamente en la colección mediante el uso del buscador y estas ayudas. Se diferencian entre sí en que la 6 tomará la imagen de entre las añadidas por anteriores usuarios, y la 7 de entre las de la colección inicial de imágenes.

Excepcionalmente en estas dos últimas tareas, comprendiendo que puede darse el caso de que el usuario, por más esfuerzos que haga, no consiga localizar las imágenes solicitadas, se le da la libertad de abandonar la prueba en el momento que considere oportuno, pulsando sobre el botón “Abandonar prueba”, y pasando con ello a la siguiente prueba.



*Ilustración 36: Botón
"Abandonar Prueba"*

Una vez terminadas todas estas pruebas, se presentará al usuario un cuadro de despedida, en el que se le pedirá que deje por escrito, si así lo desea, opiniones sobre la aplicación y el experimento, para nuestro mejor entendimiento de sus impresiones.

Un cuadro de diálogo con un título "Fin del experimento. Comentarios finales." y un cuerpo de texto que dice: "El experimento ha terminado. ¡Muchas gracias por su colaboración! Si así lo desea, escriba aquí los comentarios que estime oportunos (por ejemplo, ¿cree que la interfaz es fácil de usar? ¿siente que las recomendaciones le han sido de utilidad? ¿qué mantendría o cambiaría?). Su opinión es de mucha utilidad:". Debajo del texto hay un área de entrada de texto con un cursor y una barra de desplazamiento a la derecha. En la parte inferior derecha del cuadro hay un botón "Ok".

Ilustración 37: Cuadro de comentarios al fin del experimento

4. Desarrollo de la aplicación

En esta sección, describiremos de forma somera la aplicación preparada para la ejecución del experimento descrito en la sección anterior, detallando su estructura, partes y decisiones en su desarrollo.

4.1. Extracción de imágenes

Como se explicó en la descripción del experimento, la herramienta toma como base de información una colección de aproximadamente 20000 imágenes etiquetadas, tomadas de páginas de Internet dedicadas a las folksonomías para clasificación de imágenes. Se trataron de imponer las mínimas condiciones en la elección de las imágenes a extraer, tratando de reflejar la diversidad de material gráfico y modos de etiquetado que se hace visible en un sistema folksonómico abierto. Aún así, se definieron unas pocas restricciones para asegurar que los datos resultasen útiles para la ejecución del experimento:

- Toda imagen debería tener al menos dos etiquetas asignadas.
- En el total de imágenes, deberán distribuirse a partes iguales las imágenes etiquetadas en los siguientes cuatro idiomas: inglés, español, francés y alemán. Esto es, una cuarta parte de las imágenes etiquetadas en inglés, otra cuarta parte en español, etcétera. ¹

Se decidió guardar para cada imagen tanto el fichero de la imagen (formato JPEG), como una serie de metadatos descritos a continuación. Toda esta información quedaría recogida en un fichero XML, que serviría para su conservación, y que posteriormente habría de ser tratado por la aplicación para la inserción de estos datos en su base de información. La estructura del fichero XML sería la que sigue (se utiliza el modelo gráfico de XMLSpy [38] para la representación del esquema XML):

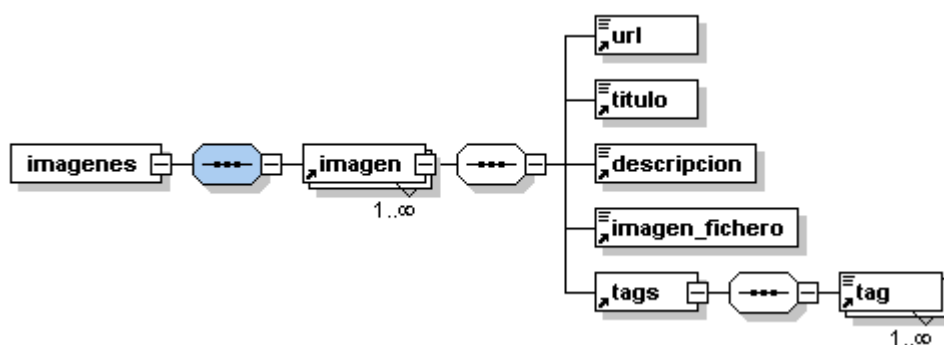


Ilustración 38: Esquema de la estructura XML

¹ Al ser esta última restricción de difícil cumplimiento exacto, al no indicarse entre los datos de cada imagen el idioma en que su información está escrita, el requisito se toma como indicativo, y su cumplimiento se hará de forma aproximada, como se indica más adelante.

Algunas de las etiquetas del fichero XML tienen contenido asociado, donde se guardan los datos de cada imagen. Éstas son:

- **URL:** URL donde puede encontrarse la imagen original.
- **Título:** Título de la imagen, en caso de que exista.
- **Descripción:** Descripción de la imagen, en caso de que se haya escrito una.
- **Imagen_fichero:** Ruta en el disco duro local donde se ha guardado la imagen tomada de Internet.
- **Tag:** Indica una etiqueta asignada a la imagen.

Para la extracción de imágenes, se manejaron dos diferentes sitios folksonómicos para imágenes, Arte Y Fotografía [39] y Flickr [4]. Ambos difieren en tamaño y enfoque: mientras Flickr es un sitio web con un número ingente de usuarios y fotografías, venidos ambos de todo el mundo, Arte Y Fotografía es un pequeño sitio web dirigido a usuarios argentinos, y en general hispanohablantes.

Se eligió este último sitio para asegurar un cierto número de imágenes etiquetadas en español. El algoritmo para la extracción de imágenes en este sitio web hubo de adaptarse a su particular estructura. Hubo de tenerse en cuenta que no se ofrecen servicios para programadores para la extracción automática de imágenes, como pudieran ser servicios web, que veremos al hablar de Flickr; de este modo, el acercamiento debía hacerse mediante análisis del código HTML de sus páginas. Los pasos del algoritmo fueron:

1. Las imágenes de la página pueden ser accedidas mediante la URL <http://www.arteyfotografia.com.ar/2493/fotos/<id>>, sustituyendo <id> por un número entre 1 y 40000 (aproximadamente el total de imágenes de la página). Partiendo de *id* =1000, y aumentando esta variable de 10 en 10 (esto es: 1010, 1020, etcétera), se fueron creando diferentes URL de imágenes a extraer. Se evitó acceder a imágenes con identificadores consecutivos pues tenderían a pertenecer a una misma colección, siendo probablemente parecidas entre sí. Espaciando de este modo los identificadores accedidos aumentábamos la variedad de orígenes y temas de las imágenes del sistema.
2. Para una determinada URL obtenida en el paso anterior, se obtenía su código HTML asociado. Mediante análisis del mismo, se obtenía la URL del fichero de la imagen, título, etiquetas y descripción. En caso de que la página estuviese bien formada y el número de etiquetas aplicadas fuera dos o mayor, se guardaban todos estos datos en una lista, y se descargaba el fichero de imagen.
3. Se repitieron pasos 1 y 2 hasta obtener una colección suficientemente amplia. Cada 10 imágenes extraídas, se generaba un fichero XML con todos los datos obtenidos, según la estructura antes descrita, previendo contra posibles paros inesperados del programa.

Para la colección final se obtuvieron unas 2000 imágenes de este modo, con utilización mayoritaria del idioma español en su etiquetado.

4.1.1. Extracción de imágenes en Flickr

Para obtener el resto de imágenes, se confió en el sitio web Flickr, uno de los más importantes para el guardado y etiquetado de imágenes, tanto en número de usuarios o imágenes, como en servicios adicionales para los usuarios.

Una de las mayores ayudas que Flickr ofreció para el proceso de extracción de imágenes es el hecho de contar con su propia API para desarrolladores [40] (*Application Programming Interface*, conjunto de funciones ofrecidas para atender peticiones hechas por programas informáticos). Esta API cuenta con servicios para ejecutar gran parte de las acciones que puede ejecutar manualmente un usuario, como puedan ser buscar imágenes indicando diferentes parámetros, acceder a los datos de una imagen en concreto o añadir contenidos. Está disponible para su uso libre con fines no comerciales; para fines comerciales, es necesario llegar a un acuerdo con Flickr.

Está construida mediante servicios web, es decir, funciona mediante peticiones HTTP ejecutadas sobre Internet, enviando los datos en formatos de intercambio como XML. Es por tanto accesible independientemente del lenguaje de programación utilizado. Ha surgido por este motivo una larga serie de proyectos desarrollados por terceros, generando librerías para una gran variedad de lenguajes de programación, que abstraen al programador del envío y traducción de estas peticiones, y le muestran una interfaz abstracta para el acceso a estos servicios. En este proyecto se utilizó FlickrJ [41], librería para Java con la utilidad antes comentada.

Para asegurar el segundo requisito planteado para la colección de imágenes, esto es, que la representación de los cuatro idiomas seleccionados en el etiquetado de las imágenes fuese aproximadamente equitativa, debía pensarse en algún modo para localizar las imágenes etiquetadas en tales idiomas. En Flickr, como en la mayoría de los sistemas folksonómicos, no existe información asociada a una imagen indicando cuál es el idioma en que se ha etiquetado o descrito, de modo que en principio no existe un modo automático para descubrir el idioma de las descripciones.

Para solucionar esto, se pensó en métodos aproximados. Recordando cómo en algunos experimentos (véase [22], sección 4. *Improving Tag Literacy*, subapartado *Tagging Observed*) se intentó determinar el idioma de las etiquetas comparando estas con listados de palabras presentes en los diccionarios de esos idiomas. Tratando de adaptar este procedimiento a un sistema de sencillo uso, se procedió en un primer experimento a obtener imágenes etiquetadas en español mediante un listado de las palabras del idioma español obtenidas de un tesoro. El algoritmo aplicado fue:

1. Se extrae la primera palabra habida en el tesoro, si la hay, caso en que se guarda en memoria y se borra del tesoro. De otro modo, fin.
2. La palabra obtenida en el paso 1 se utiliza para ejecutar una búsqueda en Flickr.
3. Del listado de resultados obtenido en el paso 2, se guardan los datos de las imágenes situadas en los puestos 3, 6 y 9.
4. Se vuelve al paso 1.

Sin embargo, este procedimiento se descartó finalmente, pues los resultados obtenidos no eran totalmente adecuados: se obtuvieron bastantes imágenes etiquetadas en portugués, idioma que

guarda cierta similitud con el español, y en otros idiomas no relacionados.

Otra idea, que finalmente fue la ejecutada, fue basarse en las colecciones de imágenes de Flickr. Los usuarios de Flickr pueden crear grupos de imágenes, usualmente con una temática común o idea unificadora, a los que cualquier otro usuario puede añadir sus imágenes si considera que encajan. Funcionan como algo parecido a colecciones temáticas de imágenes. Así, existen grupos para imágenes de mascotas, conciertos, edificios, amaneceres, y un largo etcétera, entre los que se encuentran grupos para imágenes de personas de determinados países, o que utilizan determinados idiomas para expresarse.

A este respecto, se decidió localizar una serie de grupos de expresión mayoritaria en los cuatro idiomas definidos, y extraer de ellos las imágenes del experimento. El algoritmo para ello fue:

1. Para un determinado grupo de imágenes, se localizaba su id, y ejecutaba una búsqueda mediante la API de Flickr para obtener todas las imágenes contenidas en él (aunque, por limitaciones de la misma, no podían obtenerse más de 10000).
2. Sobre los resultados obtenidos, se tomaba la primera imagen, y se saltaban cuatro (se tomaban de este modo las imágenes numeradas 1, 6, 11, etcétera). Si no había más resultados, fin.
3. Se guardaban los datos de cada imagen escogida en el paso 2, volviendo al mismo.

4.2. Desarrollo de la aplicación: Arquitectura

Planteada la descripción de la aplicación a ser desarrollada, dada en el anterior apartado de descripción del experimento, se explicarán en los apartados siguientes los detalles de su diseño e implementación, que permitan la total comprensión de la estructura de la aplicación creada.

Como primer paso en el diseño de la aplicación, y tomando como partida la descripción de su funcionalidad, se construyó una arquitectura lógica de la misma, definiendo sus componentes principales y las conexiones entre los mismos. Esta arquitectura serviría como una especie de mapa de la aplicación, definiendo y limitando su funcionalidad y confiriéndole una estructura.

Para esta arquitectura, se tomó como modelo principal la organización básica de *Modelo – Vista – Controlador* (MVC). Este patrón para el diseño de arquitectura es de especial interés en aplicaciones interactivas pensadas para ser utilizadas por usuarios, y separa la funcionalidad de la aplicación en tres elementos básicos: *modelo*, datos que la aplicación maneja; *vista*, interfaz de la aplicación con la que el usuario interactúa, y presenta la información del modelo de un modo interpretable por el usuario; y *controlador*, que toma los eventos y acciones producidas por el usuario en la vista, y ejecuta a raíz de ellos cambios en el modelo.

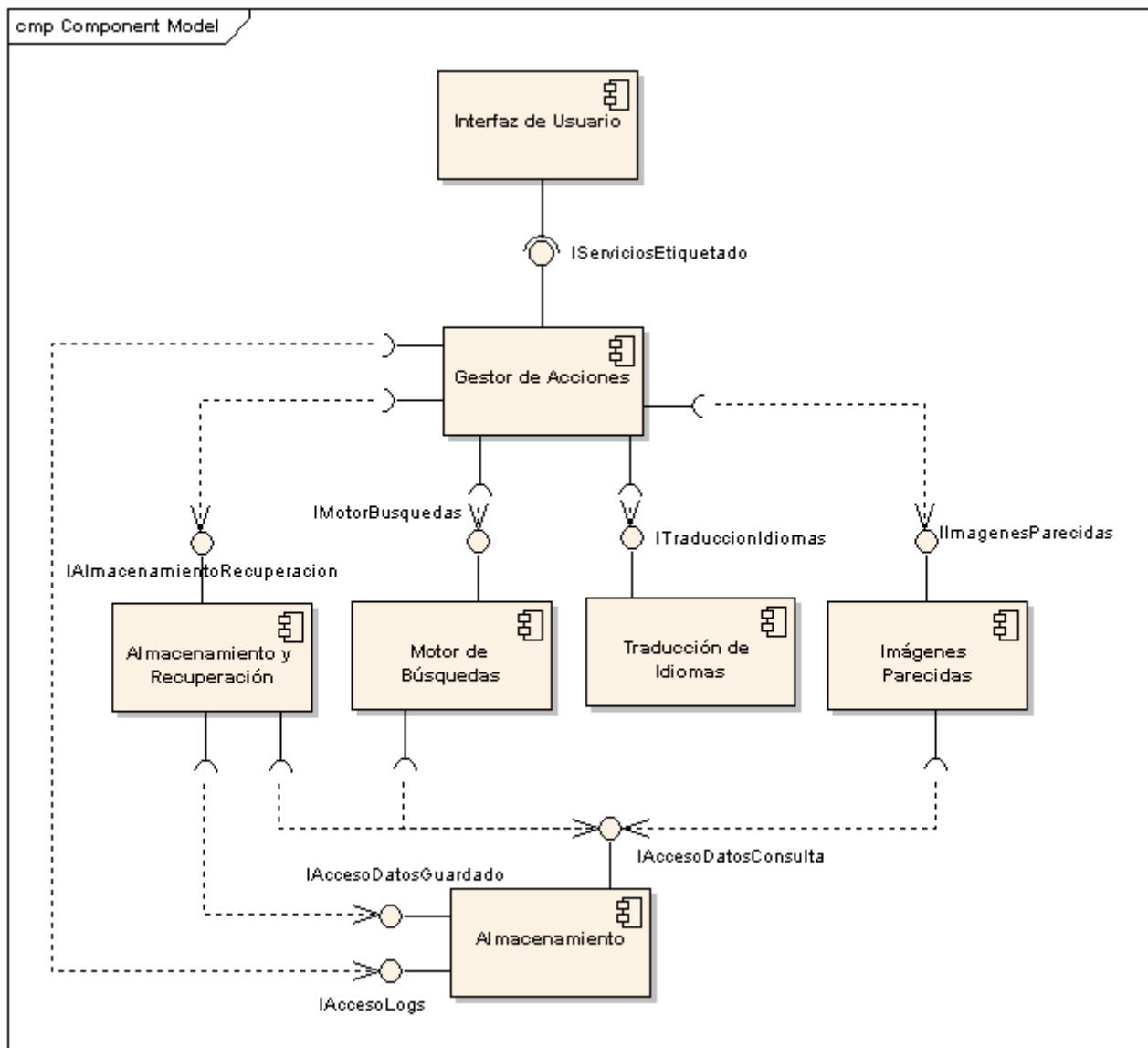


Ilustración 39: Arquitectura de la aplicación

Puede verse en la ilustración anterior la arquitectura planteada. Como aspectos reseñables, puede verse en el dibujo la estructura de cuatro capas de la arquitectura; la capa de controlador definida en el patrón MVC se ha dividido aquí en dos para su mejor gestión, utilizando la primera como un gestor de acciones, que coordinará las funcionalidades ofrecidas por la segunda (capa de negocio).

El componente de la primera capa, *Interfaz de Usuario*, se encarga de mostrar al usuario una interfaz para el manejo de la aplicación, presentándole pruebas, tareas y diversas informaciones, y en general guiándole y respondiendo a sus peticiones y acciones. Solicita servicios de la segunda capa en caso de necesitarlo: *Gestor de Acciones* escucha sus peticiones y las responde coordinando las acciones de los componentes de la tercera capa.

Los cuatro componentes de la tercera capa agrupan la lógica de la funcionalidad de la aplicación según su diversa naturaleza, buscando un equilibrio en la cantidad de responsabilidades asignada a cada uno, y el menor acoplamiento entre los mismos. Así, *Almacenamiento y Recuperación* se encarga de las tareas de guardado de nuevas imágenes, generando los ficheros necesarios, y de la obtención de los datos de otras imágenes de la colección. *Motor de Búsquedas* se encarga de la creación de índices y la gestión de las consultas, para la búsqueda de imágenes dentro de la colección del sistema, necesaria en las tareas de búsqueda planteadas al usuario. *Traducción de Idiomas* se encarga de la traducción automática de una serie de palabras a una serie de idiomas, y es necesario para el funcionamiento de las ayudas de traducción. *Imágenes Relacionadas* se encarga de buscar, a partir de una determinada imagen, otras imágenes dentro de la colección del sistema que guarden cierto parecido con la misma (sea por color, brillo o aspectos similares); es necesario para el funcionamiento de las ayudas de imágenes parecidas presentadas al usuario.

Finalmente, el componente de *Almacenamiento* se encarga del manejo a bajo nivel de los datos, su almacenamiento y organización en soporte físico. Ofrece interfaces para el manejo tanto de los datos del dominio como de logs de la aplicación.

En las siguientes secciones, describiremos en detalle cada uno de estos componentes, tratando de dar una visión detallada sobre el diseño de la aplicación.

4.3. Componente: Interfaz de Usuario

En esta sección describiremos el componente *Interfaz de Usuario* de la arquitectura de la aplicación.

4.3.1. Funcionalidad

La función del componente *Interfaz de Usuario* es la de presentar al usuario una interfaz gráfica que le permita el manejo de la aplicación y la ejecución de acciones sobre la misma. Las diferentes pantallas que se presentarán, con sus funciones y estructura, pueden verse en la sección “3.8 Recorrido por la aplicación” de este documento.

Esta interfaz presentará al usuario las diferentes tareas y ayudas, y solicitará los servicios de los componentes de las capas inferiores para cualesquiera acciones que necesiten de lógica de negocio, como el guardado de una imagen, la obtención de una ayuda, la ejecución de una búsqueda, etcétera.

4.3.2. Tecnología: Ajax

Para la implementación de este componente, hubo de decidirse la tecnología que sería utilizada para su desarrollo, dada la amplia gama existente para el desarrollo de interfaces gráficas. Como primer paso, se decidió que, siendo conveniente que los usuarios pudiesen acceder a la aplicación desde cualquier ordenador, evitando la instalación de software en sus máquinas locales,

el mejor acercamiento sería el desarrollo de una aplicación web.

Habiendo decidido que la aplicación sería web, las tecnologías barajadas para la implementación de la capa de interfaz fueron aquellas relacionadas con este tipo de presentación, basada en la generación de páginas HTML interpretables por navegadores web. Se estudió la posibilidad del desarrollo mediante JSP (tecnología Java que permite la generación de páginas HTML dinámicas), hojas de estilo CSS y posiblemente utilizando algún *framework* (esto es, una plataforma que aporta al programador una arquitectura y servicios básicos, a partir de los cuales puede desarrollar su aplicación completa) para la perfecta organización del código, como pueda ser Struts [42].

Sin embargo, esta opción contaba con el defecto de resultar excesivamente estática: cambios en la interfaz requerirían la recarga de una página HTML. Para la correcta usabilidad de la aplicación, era necesario que pudiesen mostrarse pequeños cambios en la interfaz sin necesidad de recargar completamente las páginas, pudiendo así, por ejemplo, actualizar los listados de recomendaciones de etiquetas sin perjudicar la interacción del usuario con la aplicación. Por ello, se pensó en una interfaz basada en Ajax.

Ajax, siglas para *Asynchronous JavaScript And XML* (JavaScript asíncrono y XML), no es tanto una nueva tecnología como una conjunción de tecnologías existentes, unidas para la creación de interfaces web con alto grado de fluidez e interacción. Su característica esencial es la posibilidad de actualizar fragmentos de una página sin necesidad de recargarla por completo: las páginas Ajax se ejecutan en el cliente, y pueden ejecutar peticiones asíncronas de información al servidor sin interferir la interacción. Esto permite la creación de interfaces web mucho más fluidas de lo habitual, acercándose a las prestaciones de un programa ejecutado en máquina local, como atestiguan ejemplos reales como Gmail o Google Maps.

Ajax es de este modo una combinación de tecnologías ya existentes, en concreto:

- XHTML (evolución de la codificación HTML, que fuerza a cumplir las restricciones definidas por XML, para una mayor facilidad de interpretación y manejo) y hojas de estilos CSS para la presentación.
- Utilización de DOM (*Document Object Model*), un modelo computacional independiente de plataforma y lenguaje para la representación de la estructura de HTML, que permite su modificación dinámica. Esta estructura es accedido por un lenguaje de script ejecutado en el cliente, por lo general JavaScript, para permitir la actualización de páginas sin necesidad de su recarga completa.
- El intercambio de datos con el servidor se ejecuta mediante el objeto XMLHttpRequest, una API para realizar peticiones HTTP al servidor web.
- El formato en que los datos se intercambian con estas peticiones es habitualmente XML, aunque cualquier formato está permitido, como puedan ser texto plano, HTML o JSON.

Con todo, una serie de defectos han de ser considerados antes de elegir utilizarlo. Ajax dificulta la accesibilidad de las aplicaciones web (esto es, el grado en que esa aplicación es capaz de

ser ejecutada por toda máquina, independientemente de sus características técnicas): no todas las máquinas son capaces de ejecutar JavaScript, como por ejemplo pueden ser los navegadores desde teléfonos móviles, o pueden simplemente tenerlo desactivado, caso en que sería imposible la ejecución de Ajax. Esto puede dificultar también su indexación por motores de búsquedas sobre Internet, si estos se ven incapaces de leer el texto de la página al ser éste cargado asincrónicamente.

Es necesario también tener en cuenta que el desarrollo directamente en Ajax es una tarea complicada, al estar implicada tal variedad de tecnologías, y faltando un entorno de desarrollo que facilite la depuración y detección de errores, que se suma a las dificultades usuales del desarrollo web, como son las incompatibles entre diferentes navegadores.

En cualquier caso, para este proyecto en concreto, juzgamos asumibles estas dificultades. Al no ser una aplicación destinada a un público masivo, sino a un pequeño conjunto de usuarios que en principio la ejecutarían una única vez, juzgamos que sería posible con facilidad para todos ellos el obtener un navegador con JavaScript con el que acceder a la aplicación. Para el resto de dificultades, estudiamos el uso de *frameworks* para el desarrollo de Ajax, como se explica en la sección que sigue.

4.3.2.1. Google Web Toolkit (GWT)

Para solucionar muchos de los problemas del desarrollo en Ajax antes mencionados, existen plataformas (*frameworks*) que permiten facilitar la programación y tareas de depuración sobre estas tecnologías. Existen diversos tipos, cada uno con diferentes servicios y utilidades. Por lo general, facilitan la programación mediante JavaScript, incluso permitiendo la generación de JavaScript a partir de código escrito en otros lenguajes, y aportan estructura a la información y llamadas ejecutadas entre cliente y servidor.

Para este proyecto, buscamos un *framework* lo suficientemente robusto y documentado que nos permitiese el fácil desarrollo de Ajax e integración con las funciones implementadas para la parte del servidor. Debido a nuestra mayor experiencia en lenguaje Java, buscamos en concreto plataformas orientadas a ese lenguaje. Algunas de las opciones encontradas fueron DWR [43], Ajax Tags [44] y GWT [45]. De entre ellas, se eligió esta última por el renombre de Google, compañía que lo ha desarrollado, que nos aseguraba un grado de documentación y soporte suficiente si dificultades eran encontradas en el desarrollo.

Google creó la plataforma Google Web Toolkit (GWT) en el año 2006, un *framework* para la creación de código Ajax a partir de Java, aportando funciones para la fácil integración entre las partes de cliente y servidor, y una librería de componentes ya desarrollados para la utilización en nuestras interfaces (*widgets*), como sean campos de texto, selectores de ficheros, paneles para ordenar vertical u horizontalmente otros *widgets*, etcétera. GWT permite el desarrollo de aplicaciones Ajax en un entorno de desarrollo similar al utilizado para el desarrollo Java, incluyendo utilidades para por ejemplo la integración con Eclipse.

GWT se compone de una serie de elementos básicos, dos componentes y dos librerías:

- **GWT Java-to-JavaScript Compiler:** Compilador de Java a JavaScript. Podríamos considerarlo el elemento básico de GWT, el encargado de la transformación del código Java desarrollado a código JavaScript ejecutable por un cliente web. El código generado está especialmente optimizado para su interpretación con diferencias gráficas mínimas por parte de los navegadores web mayoritarios, como Internet Explorer, Mozilla Firefox u Opera.
- **GWT Hosted Web Browser:** Navegador web para la emulación. Para facilitar el proceso de desarrollo, GWT permite la emulación del código Java sin necesidad de generar JavaScript. De este modo, se definen dos modos de trabajo: en desarrollo, el programador podrá probar su código rápidamente emulándolo mediante este navegador; una vez que esté completo y deba pasarse a producción, generará código JavaScript mediante el compilador GWT.
- **JRE emulation library:** Librería para la emulación del JRE: permite la traducción de clases y métodos básicos de Java a JavaScript. En GWT, no todos los paquetes de las librerías básicas de Java pueden ser utilizados en el código a ser traducido a JavaScript: sólo un subconjunto de estos paquetes es interpretable por GWT, concretamente `java.lang` en su mayor parte y algunas clases de `java.util`.
- **GWT Web UI class library:** Librería de *widgets* (componentes pre-desarrollados para su inclusión en interfaces gráficas) que permite el uso de elementos básicos de las interfaces web, como puedan ser cajas de texto, imágenes o botones.

Otra importante ventaja es que GWT ofrece bajo una licencia Apache 2.0 el uso de todo su código fuente, pudiendo de este modo ser libremente modificado y consultado por terceros. Google no impone ningún tipo de derecho intelectual sobre el software desarrollado mediante GWT, aunque sí mantiene derechos sobre el compilador, y permite la ampliación de sus librerías mediante otras desarrolladas por terceros. Una amplia comunidad de desarrolladores ajenos a Google han ido gracias a ello ampliando sus funcionalidades mediante librerías que incluyen numerosos otros *widgets* para las interfaces y diversas otras utilidades.

Se ha dicho antes que las aplicaciones Ajax se ejecutan totalmente en el cliente, el navegador web, llamando en contadas ocasiones al servidor para obtener determinadas informaciones, y mostrándolas al usuario sin necesidad de una recarga completa de la página. En GWT, el modo empleado es RPC (*Remote Procedure Call*, llamada a procedimiento remoto), integrado de un modo muy limpio en el *framework*, permitiendo su desarrollo de un modo claro y fácil.

4.3.2.2. **Extensiones de GWT: MyGWT**

En la experiencia, una de las dificultades del desarrollo con los elementos básicos de GWT es el contar con un conjunto de *widgets* insuficiente para la creación de interfaces lo suficientemente ricas. Las librerías básicas de GWT sólo ofrecen una serie de elementos básicos, no siempre parametrizables a gusto del desarrollador, en los que muchas funcionalidades de otras interfaces Ajax actuales no están presentes (los ejemplos son numerosos, pero podemos citar la falta de manejo para interfaces basadas en arrastrar y soltar).

Para solventar esto, el programador tiene siempre la posibilidad de ampliar GWT con desarrollos propios, permitiéndole la creación de otros componentes adaptados a sus necesidades. Amparada en la licencia de software libre de GWT, una larga serie de librerías externas han sido desarrolladas por terceras personas, que amplían sus funcionalidades supliendo una larga serie de necesidades, y evitando la inversión de tiempo y esfuerzo que requeriría su desarrollo en cada proyecto.

Para este proyecto, se consideró de utilidad la inclusión de una librería de ampliación, que nos permitiese contar con una colección mayor de componentes para el desarrollo de la interfaz. Dos fueron las principales candidatas: MyGWT [46] y GWT-Ext [47]. Ambas, de código libre, comparten una serie de componentes similares, algo más amplia en GWT-Ext; sin embargo, finalmente se decidió el uso en exclusiva de MyGWT por su desarrollo íntegro en GWT, que facilitaba la integración con el proyecto. GWT-Ext, por el contrario, requiere de la inclusión de una librería externa de JavaScript, no nativa de GWT.

MyGWT incluye una amplia variedad de *widgets* para la construcción de interfaces, como puedan ser listas, tablas, paneles arrastrables y desplegados o barras de herramientas. Muchos de ellos están parametrizados para permitir su fácil adaptación y manejo. En la ilustración siguiente puede verse una aplicación a modo de ejemplo incluida en la distribución de MyGWT, que permite ver y explorar estos nuevos componentes.

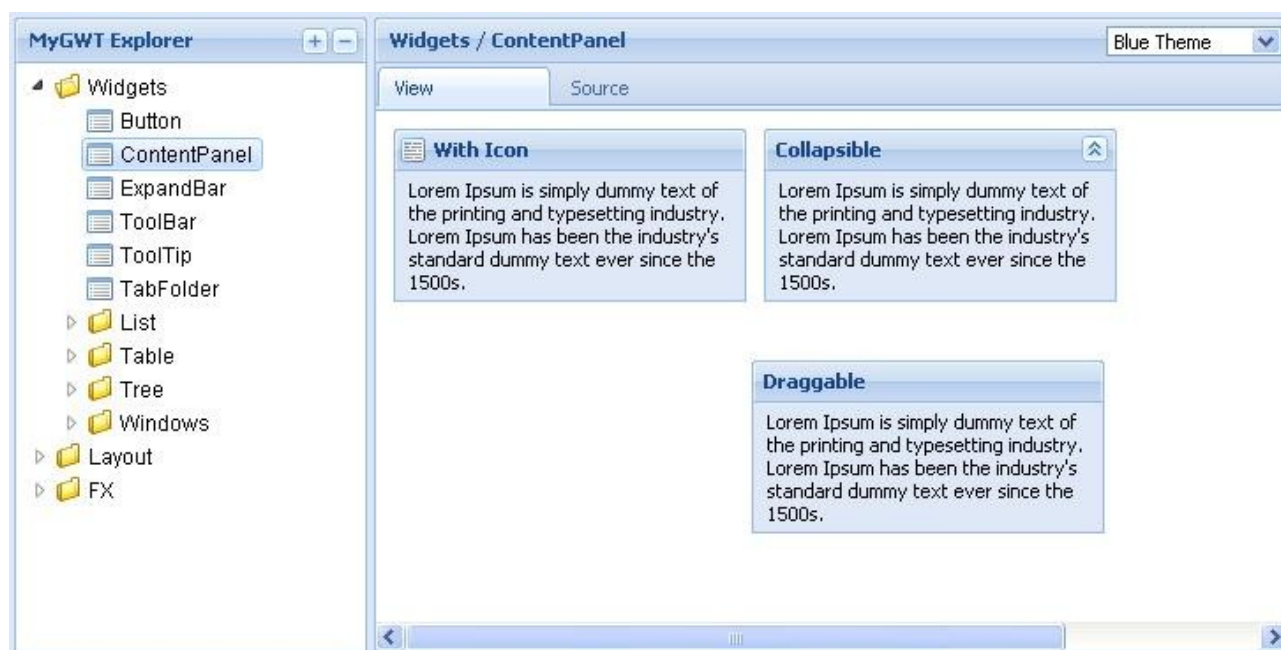


Ilustración 40: Demo Explorer de MyGWT

Otra de las ventajas aportadas por esta librería es contar con una hoja de estilos CSS muy cuidada y agradable a la vista, que permite la construcción de interfaces vistosas para el usuario a partir de sus elementos básicos, sin necesidad de entrar en tareas de diseño gráfico si éstas no resultan relevantes.

Debe decirse que, a día de hoy, esta librería ha dejado de existir como tal, fusionándose con otros proyectos y creando Ext GWT [48] (no relacionada con la antes mencionada GWT-Ext). El cambio le ha añadido funcionalidades pero ha supuesto también un cambio en la licencia de uso, haciéndola más restrictiva. En este proyecto, nos hemos limitado al uso de MyGWT en su versión 0.4.44, de libre distribución.

4.4. Componente: Gestor de Acciones

En este apartado explicaremos las características del componente *Gestor de Acciones*, de la arquitectura de la aplicación.

4.4.1. Funcionalidad

El componente *Gestor de Acciones*, perteneciente a la segunda capa de la arquitectura de la aplicación, es el encargado de la recepción de los eventos y mensajes producidos en la interfaz de la aplicación, y de dar una respuesta a los mismos utilizando para ello los servicios ofrecidos por los componentes de las capas inferiores de la arquitectura de la aplicación. Es, por así decirlo, un elemento director, que hace uso y orquesta los diferentes servicios y funciones ofrecidos por los componentes de las capas inferiores, en busca de dar una respuesta puntual y correcta a las peticiones de la interfaz.

4.4.2. Interfaces

Como se muestra en la arquitectura de la aplicación, este componente ofrece los servicios de la interfaz *IServiciosEtiquetado* a las capas superiores. Estos servicios encapsulan todas las tareas de la aplicación que deben ejecutarse del lado del servidor, y deben por tanto ser invocadas desde la interfaz. Se detallan a continuación:

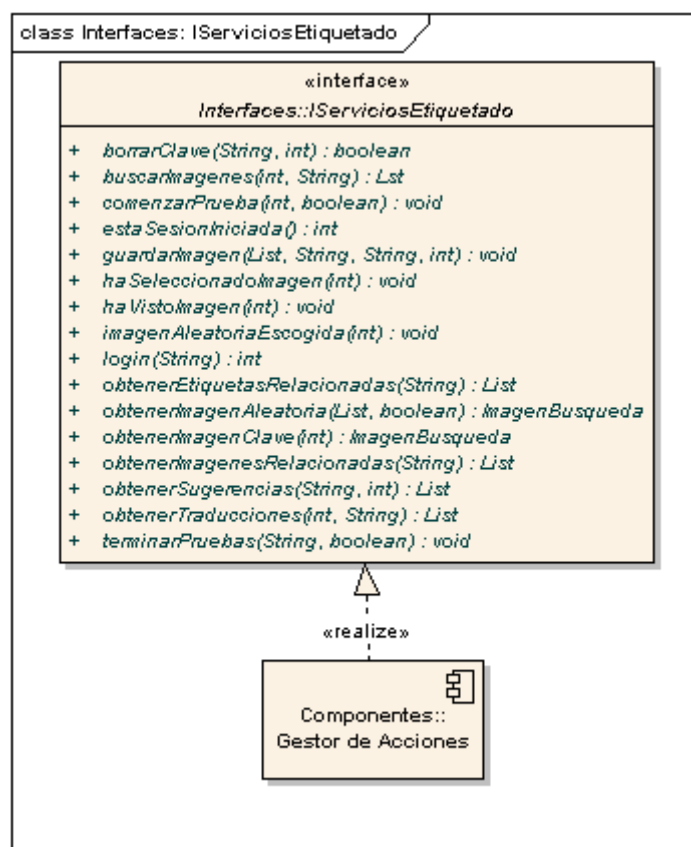


Ilustración 41: Interfaz ServiciosEtiquetado

Operación: borrarClave(tipo: String, clave: int) : boolean
Parámetros: tipo: Extensión del fichero de la imagen. clave: Clave a eliminar.
Descripción: Este método elimina una determinada clave de imagen del sistema, borrando los ficheros de imagen asociados.
Retorno: Valor booleano indicando si ha podido eliminarse la clave.

Tabla 1: Interfaz ServiciosEtiquetado – método borrarClave

Operación: buscarImagenes(query: String, pagina: int) : List
Parámetros: query: Consulta a ejecutar. pagina: Número de página dentro de los resultados (cada página incluye 20 resultados).
Descripción: Este método ejecuta una búsqueda dentro de la colección de imágenes.
Retorno: Listado de resultados de la consulta dentro de la página indicada.

Tabla 2: Interfaz ServiciosEtiquetado – método buscarImagenes

Operación: comenzarPrueba(abandonoAnterior: boolean, numPrueba: int) : void
Parámetros: <i>abandonoAnterior:</i> Valor booleano que indica si la prueba anterior fue abandonada por el usuario. <i>numPrueba:</i> Número de prueba a iniciar.
Descripción: Este método informa al sistema de que se ha iniciado una nueva prueba.
Retorno: Ninguno.

Tabla 3: Interfaz ServiciosEtiquetado – método comenzarPrueba

Operación: estaSesionIniciada() : int
Parámetros: Ninguno.
Descripción: Este método comprueba si el usuario ya ha iniciado sesión. Esto indicaría al sistema de que el usuario ha refrescado la página o recargado de algún modo.
Retorno: Entero con valor -1 si no se ha iniciado sesión, y el número de pruebas completadas por el usuario (valor mayor que -1) si se ha iniciado.

Tabla 4: Interfaz ServiciosEtiquetado – método estaSesionIniciada

Operación: guardaImagen(tags: List, titulo: String, descripcion: String, clave: int) : void
Parámetros: <i>tags:</i> Listado de etiquetas asignadas a la imagen. <i>titulo:</i> Título asignado a la imagen. <i>descripcion:</i> Descripción asignada a la imagen. <i>clave:</i> Clave de la imagen.
Descripción: Este método añade datos (título, descripción, etiquetas) a una determinada imagen, identificada con su clave.
Retorno: Ninguno.

Tabla 5: Interfaz ServiciosEtiquetado – método guardaImagen

Operación: haSeleccionadoImagen(clave: int) : void
Parámetros: <i>clave:</i> Clave de la imagen.
Descripción: Este método informa al sistema de que, en las tareas de búsqueda de imágenes, el usuario ha indicado que una determinada imagen es la imagen buscada.
Retorno: Ninguno.

Tabla 6: Interfaz ServiciosEtiquetado – método haSeleccionadoImagen

Operación: haVistoImagen(clave: int) : void
Parámetros: <i>clave:</i> Clave de la imagen.
Descripción: Este método informa al sistema de que, en las tareas de búsqueda de imágenes, el usuario ha pulsado en una determinada imagen para ver sus datos.
Retorno: Ninguno.

Tabla 7: Interfaz ServiciosEtiquetado – método haVistoImagen

Operación: imagenAleatoriaEscogida(clave: int) : void
Parámetros: <i>clave:</i> Clave de la imagen.
Descripción: Este método informa al sistema de que, al comienzo de un determinada prueba, el sistema ha escogido una determinada imagen para mostrársela al usuario.
Retorno: Ninguno.

Tabla 8: Interfaz ServiciosEtiquetado – método imagenAleatoriaEscogida

Operación: login(usuario: String) : int
Parámetros: <i>usuario:</i> Nombre del usuario.
Descripción: Este método inicia una sesión en el sistema, con el nombre de usuario indicado.
Retorno: Un valor menor que 0 si hay error, igual o mayor si no lo hay.

Tabla 9: Interfaz ServiciosEtiquetado – método login

Operación: obtenerEtiquetasRelacionadas(tag: String) : List
Parámetros: <i>tag:</i> Etiqueta.
Descripción: Este método obtiene etiquetas relacionadas con otra pasada como parámetro.
Retorno: Listado de etiquetas relacionadas con la indicada.

Tabla 10: Interfaz ServiciosEtiquetado – método obtenerEtiquetasRelacionadas

Operación: obtenerImagenAleatoria(deOtroUsuario: boolean, imgsYaMostradas: List) : ImagenBusqueda
Parámetros: <i>deOtroUsuario:</i> Valor booleano que indica si la imagen debe escogerse entre las añadidas por otros usuarios. <i>ImgsYaMostradas:</i> Listado de claves de otras imágenes ya mostradas, para evitar repeticiones.
Descripción: Este método devuelve los datos de una imagen escogida aleatoriamente de la colección del sistema.
Retorno: Datos de la imagen escogida.

Tabla 11: Interfaz ServiciosEtiquetado – método obtenerImagenAleatoria

Operación: obtenerImagenClave(clave: int) : ImagenBusqueda
Parámetros: <i>clave:</i> Clave de la imagen.
Descripción: Este método devuelve los datos de una imagen determinada, de la que se indica la clave.
Retorno: Datos de la imagen.

Tabla 12: Interfaz ServiciosEtiquetado – método obtenerImagenClave

Operación: obtenerImagenesRelacionadas(url: String) : List
Parámetros: <i>url:</i> URL de la imagen de la que se buscan las imágenes parecidas.
Descripción: Este método obtiene imágenes parecidas dentro de la colección del sistema, de otra imagen identificada con la URL desde la que es accesible.
Retorno: Listado con los datos de las imágenes parecidas resultantes.

Tabla 13: Interfaz ServiciosEtiquetado – método obtenerImagenesRelacionadas

Operación: obtenerSugerencias(textoParcial: String, limite: int) : List
Parámetros: <i>textoParcial:</i> Texto escrito por el momento en el recuadro para el autocompletado. <i>limite:</i> Número máximo de sugerencias a obtener.
Descripción: Este método obtiene sugerencias para el autocompletado de un recuadro de texto, basándose en las etiquetas más recuentes que contengan el texto ya escrito en el recuadro.
Retorno: Listado con las sugerencias.

Tabla 14: Interfaz ServiciosEtiquetado – método obtenerSugerencias

Operación: obtenerTraducciones(etiquetas: String, codIdiomOrig: int) : List
Parámetros: <i>etiquetas:</i> Etiquetas a traducir. <i>codIdiomOrig:</i> Código del idioma en que las etiquetas fueron escritas.
Descripción: Este método traduce automáticamente las etiquetas indicadas a una serie de idiomas.
Retorno: Listado con las traducciones.

Tabla 15: Interfaz ServiciosEtiquetado – método obtenerTraducciones

Operación: terminarPruebas(abandonoAnterior: boolean, comentario: String) : void
Parámetros: <i>abandonoAnterior:</i> Valor booleano que indica si la prueba anterior fue abandonada por el usuario. <i>comentario:</i> Último comentario escrito por el usuario al terminar las pruebas.
Descripción: Este método informa al sistema de que el usuario ha terminado todas las pruebas.
Retorno: Ninguno.

Tabla 16: Interfaz ServiciosEtiquetado – método terminarPruebas

4.5. Componente: Almacenamiento y Recuperación

En este apartado explicaremos las características del componente *Almacenamiento y Recuperación*, de la arquitectura de la aplicación.

4.5.1. Funcionalidad

El componente *Almacenamiento y Recuperación*, perteneciente a la capa de negocio de la arquitectura de la aplicación, se encarga de la lógica de las operaciones básicas de almacenado y recuperación de datos en la aplicación. Gestiona tareas tales como el guardado de imágenes, el control de los nombres de usuarios indicados al inicio, o la obtención de la información asociada a una determinada imagen. Hace uso de los servicios del componente *Almacenamiento*, encargado del manejo de los datos del sistema a bajo nivel y explicado posteriormente.

Es de especial importancia por encargarse también de la lógica de dos ayudas al usuario mostradas en la interfaz, el autocompletado de etiquetas y el cálculo de las etiquetas relacionadas, asignadas al mismo componente por ser tareas limitadas al procesamiento y consulta de los datos de la aplicación.

4.5.2. Interfaces

Como se muestra en la arquitectura de la aplicación, este componente ofrece los servicios de la interfaz *IAlmacenamientoRecuperacion* a las capas superiores. Estos servicios representan tareas para la consulta y modificación de datos del sistema, y se detallan a continuación:

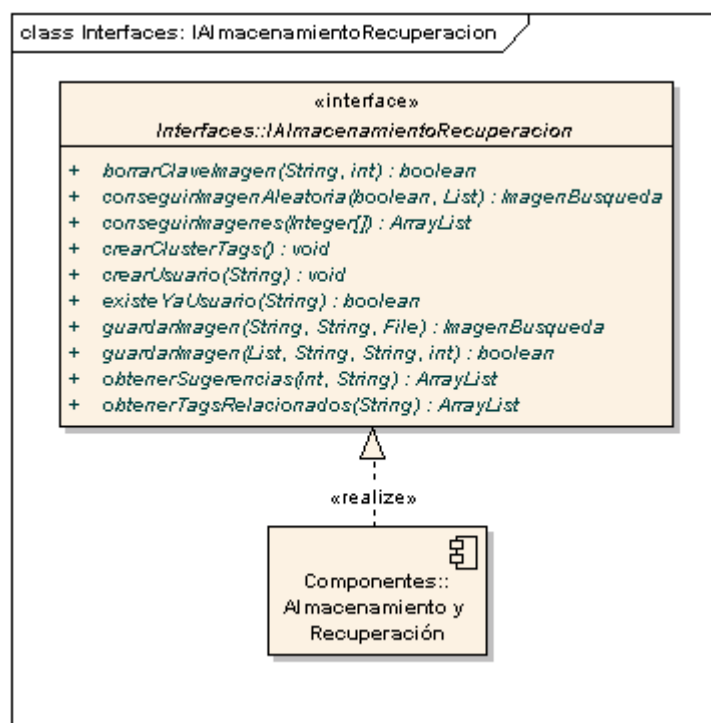


Ilustración 42: Interfaz IAlmacenamientoRecuperacion

Operación: borrarClaveImagen(tipo: String, clave: int) : boolean
Parámetros: <i>tipo</i> : Extensión del fichero de la imagen. <i>clave</i> : Clave de la imagen.
Descripción: Este método borra los datos de una imagen identificada por su clave, con los ficheros asociados incluidos.
Retorno: Valor booleano indicando si se ha borrado correctamente.

Tabla 17: Interfaz IAlmacenamientoRecuperacion – método borrarClaveImagen

Operación: conseguirImagenAleatoria(deOtroUsuario: String, clavesEvitar: List) : ImagenBusqueda
Parámetros: <i>deOtroUsuario</i> : Valor booleano indicando si la imagen a obtener debe haber sido introducida por otro usuario, o pertenecer a la colección inicial del sistema. <i>clavesEvitar</i> : Claves que no deben devolverse como resultado.
Descripción: Este método escoge aleatoriamente una imagen de la colección teniendo en cuenta los parámetros pasados.
Retorno: Datos de la imagen escogida.

Tabla 18: Interfaz IAlmacenamientoRecuperacion – método conseguirImagenAleatoria

Operación: conseguirImagenes(claves: Integer[]) : ArrayList
Parámetros: <i>claves</i> : Listado con las claves de las imágenes.
Descripción: Este método devuelve los datos de una serie de imágenes cuyas claves son pasadas como parámetro.
Retorno: Listado con los datos de las imágenes

Tabla 19: Interfaz IAlmacenamientoRecuperacion – método conseguirImagenes

Operación: crearClusterTags() : void
Parámetros: Ninguno.
Descripción: Este método inicia la ejecución del algoritmo para el cálculo de las etiquetas relacionadas.
Retorno: Ninguno.

Tabla 20: Interfaz IAlmacenamientoRecuperacion – método crearClusterTags

Operación: crearUsuario(usuario: String) : void
Parámetros: <i>usuario</i> : Nombre del usuario a crear.
Descripción: Este método guarda en el sistema el inicio de sesión de un nuevo usuario.
Retorno: Ninguno.

Tabla 21: Interfaz IAlmacenamientoRecuperacion – método crearUsuario

Operación: existeYaUsuario(usuario: String) : boolean
Parámetros: <i>usuario</i> : Nombre del usuario.
Descripción: Este método comprueba si un determinado usuario ha iniciado sesión previamente en el sistema.
Retorno: Valor booleano indicando si el usuario indicado ha iniciado sesión previamente.

Tabla 22: Interfaz IAlmacenamientoRecuperacion – método existeYaUsuario

Operación: guardarImagen(tipo: String, url: String, fichero: File) : ImagenBusqueda
Parámetros: <i>tipo</i> : Extensión del fichero de imagen. <i>url</i> : URL de donde se tomó previamente la imagen. <i>fichero</i> : Fichero de la imagen.
Descripción: Este método guarda los datos de una nueva imagen en el sistema.
Retorno: Datos de la imagen introducida.

Tabla 23: Interfaz IAlmacenamientoRecuperacion – método guardarImagen

Operación: guardarImagen(etiquetas: List, titulo: String, descripcion: String, clave: int) : boolean
Parámetros: <i>etiquetas</i> : Listado de etiquetas aplicadas a la imagen. <i>titulo</i> : Título de la imagen. <i>descripcion</i> : Descripción de la imagen. <i>clave</i> : Clave de la imagen.
Descripción: Este método añade datos textuales a una imagen ya introducida con determinada clave.
Retorno: Valor booleano indicando si los datos se han guardado correctamente.

Tabla 24: Interfaz IAlmacenamientoRecuperacion – método guardarImagen

Operación: obtenerSugerencias(cadena: String, limite: int) : ArrayList
Parámetros: <i>cadena</i> : Cadena de texto que deben contener las sugerencias. <i>limite</i> : Número máximo de sugerencias que deben ofrecerse.
Descripción: Este método devuelve sugerencias para el autocompletado de etiquetas, devolviendo un número de etiquetas entre las más utilizadas del sistema que contienen la cadena indicada.
Retorno: Listado de sugerencias.

Tabla 25: Interfaz IAlmacenamientoRecuperacion – método obtenerSugerencias

Operación: obtenerTagsRelacionados(etiqueta: String) : ArrayList
Parámetros: <i>etiqueta</i> : Etiqueta de la cual quieren encontrarse las etiquetas relacionadas.
Descripción: Este método devuelve etiquetas calculadas como relacionadas con la indicada.
Retorno: Listado de etiquetas relacionadas con la indicada.

Tabla 26: Interfaz IAlmacenamientoRecuperacion – método obtenerTagsRelacionados

4.5.3. Tecnología: FileUpload

Una de las principales tareas del componente es la recepción de nuevas imágenes por parte de la interfaz, en las tareas que requieren el envío de imágenes del usuario para su añadido. Es necesario para ello la inclusión de un mecanismo que permita la recepción de tales informaciones, recibiendo los paquetes enviados a través de la web y uniéndolos de nuevo en el fichero completo.

Para ello, fue de gran utilidad la librería de Java FileUpload [49], de libre distribución y perteneciente al proyecto Apache. Esta librería permite la recepción y análisis de peticiones HTTP producidas por el envío de ficheros a través de formularios HTML, esto es, con el tipo de contenido “multipart/form-data”, de forma transparente para el programador y con un buen rendimiento.

4.5.4. Funcionamiento: Ayuda de sugerencias para autocompletado

Una de las ayudas ofrecidas por el sistema es el autocompletado de las etiquetas y búsquedas tecleadas por el usuario, ofreciéndole sugerencias en tiempo real de modo similar a como lo haría Google Suggest [37]. Este componente se encarga de servir a la interfaz estas sugerencias que deberá presentar, haciendo para ello uso de los servicios del componente de *Almacenamiento*.

La lógica de esta ayuda es relativamente simple, pues se basa en una consulta a la base de información: para una determinada cadena introducida por el usuario, se buscarán aquellas etiquetas, ya utilizadas en imágenes previas del sistema, que contengan en su texto la cadena indicada. De estas etiquetas, se devolverán las N primeras (siendo en nuestra aplicación N fijado en seis), ordenadas según el número de imágenes en que hayan sido incluidas.

4.5.5. Funcionamiento: Ayuda de etiquetas relacionadas

La segunda ayuda al usuario manejada por este componente es la del cálculo de etiquetas relacionadas, que trata de descubrir para cada etiqueta introducida en el sistema, qué otras etiquetas se relacionan con ella. El grado de “relación” de dos etiquetas se medirá mediante su grado de co-ocurrencia, esto es, el número de imágenes que hayan sido etiquetadas con ambas etiquetas conjuntamente.

Para el desarrollo de esta ayuda, se siguió el algoritmo explicado en el apartado “2.4.2.1.Agrupaciones” de este documento, que dentro del contexto de mecanismos de mejora de las folksonomías, explicaba el modo de generar una estructura organizada de las etiquetas del sistema para generar agrupaciones de las mismas. En nuestra implementación juzgamos innecesario este último paso, interesándonos primordialmente el encontrar, a partir de una etiqueta, otras etiquetas del sistema con cierta relación con la misma.

De este modo, la aplicación de esta ayuda se divide en dos pasos: generación de un grafo que muestre el grado de relación entre cualquier par de etiquetas del sistema, y consulta del mismo para obtener las etiquetas relacionadas a otra. El grafo se genera una única vez, al inicio de la vida de la aplicación, mediante un algoritmo que requiere cierto esfuerzo computacional. Sin embargo, una vez calculado y almacenada su estructura en la base de información de la aplicación, las consultas de las etiquetas relacionadas tendrán un coste muy pequeño, pues supondrán simplemente la lectura del mismo. Los pasos para generar el grafo son los siguientes:

- 1. Crear grafo de etiquetas relacionadas:** Se creará una estructura general de todas las etiquetas del sistema, a modo de grafo, en que las etiquetas se unan entre sí con enlaces con valor numérico asociado. Este valor representará su co-ocurrencia, siendo igual al número de imágenes en que hayan utilizadas conjuntamente para su etiquetado.
- 2. Normalizar el grafo:** Para evitar recompensar a las etiquetas más populares frente a las minoritarias debido a esta medida, se establece una normalización. Por ello, se vuelven a calcular los valores de los enlaces del grafo mediante una medida de similitud *Dice*: para dos etiquetas A y B, se calcula como $P = (2 |A \cap B|) / (|A| + |B|)$.

Con estos dos pasos el grafo queda completo. Las consultas al mismo, para obtener las etiquetas relacionadas a A, se ejecutarán buscando la etiqueta A dentro del grafo, y viendo a qué otras etiquetas B, C, D... se relaciona, devolviéndose como sugerencias ordenadas según su grado de similitud.

4.6. Componente: Motor de Búsquedas

En este apartado explicaremos las características del componente *Motor de Búsquedas*, de la arquitectura de la aplicación.

4.6.1. Funcionalidad

El componente *Motor de Búsquedas*, perteneciente a la capa de negocio de la arquitectura de la aplicación, encapsula toda la lógica necesaria para el manejo de las búsquedas textuales de imágenes de la colección del sistema, a través de sus etiquetas o descripciones. Se encarga de la creación de índices para la rápida ejecución de las consultas, el añadido de nuevos elementos a tales índices, y el procesamiento de las búsquedas devolviendo los elementos coincidentes.

Esta funcionalidad es necesaria para el correcto funcionamiento de las tareas de búsqueda de imágenes planteadas al usuario en el experimento.

4.6.2. Interfaces

Como se muestra en la arquitectura de la aplicación, este componente ofrece los servicios de la interfaz *IMotorBusquedas* a las capas superiores. Estos servicios representan tareas básicas para la búsqueda de imágenes por texto, y se detallan a continuación:

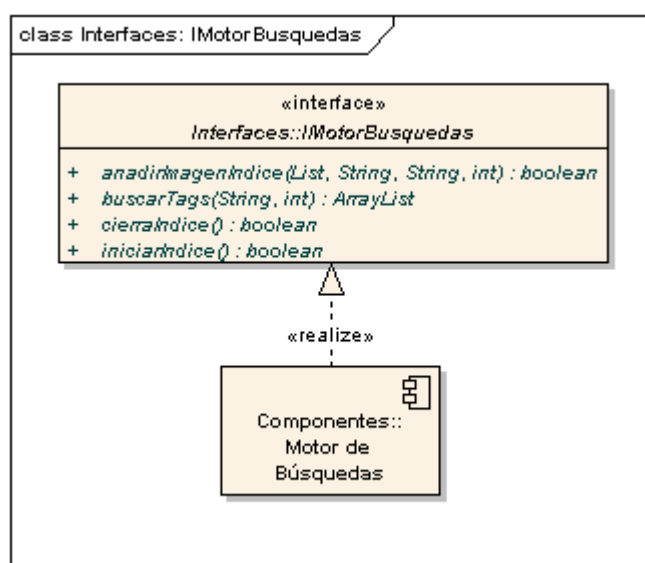


Ilustración 43: Interfaz *IMotorBusquedas*

Operación: anadirImagenIndice(tags: List, descrip: String, titulo: String, idImg: int) : boolean

Parámetros:

tags: Lista con las etiquetas utilizadas en la imagen.

descrip: Descripción de la imagen.

titulo: Título de la imagen.

idImg: Clave de la imagen.

Descripción: Este método añade una determinada imagen al índice del buscador. Se indican la clave de la imagen, e informaciones relacionadas como su título, descripción y etiquetas.

Retorno: Valor booleano indicando si se ha añadido correctamente.

Tabla 27: Interfaz *IMotorBusquedas* – método *anadirImagenIndice*

Operación: buscarTags(query: String, pagina: int) : ArrayList
Parámetros: <i>query</i> : Consulta ejecutada por el usuario. <i>pagina</i> : Número de la página de resultados a mostrar.
Descripción: Este método ejecuta una consulta del usuario. Se indican la búsqueda y el número de página de resultados a mostrar (cada página contiene 20 resultados), y se devuelven los resultados de tal búsqueda para tal página.
Retorno: ArrayList con los resultados. En su primera posición se indica el número total de resultados de la búsqueda, y el resto de posiciones indican en orden las claves de los resultados de la búsqueda para la página indicada.

Tabla 28: Interfaz IMotorBúsquedas – método buscarTags

Operación: cierraIndice() : boolean
Parámetros: Ninguno
Descripción: Este método cierra el índice para escritura una vez añadidos todos los elementos necesarios.
Retorno: Valor booleano indicando si se ha cerrado correctamente.

Tabla 29: Interfaz IMotorBúsquedas – método cierraIndice

Operación: iniciarIndice() : boolean
Parámetros: Ninguno
Descripción: Este método abre el índice para escritura, permitiendo añadir todos los elementos necesarios.
Retorno: Valor booleano indicando si se ha abierto correctamente.

Tabla 30: Interfaz IMotorBúsquedas – método iniciarIndice

4.6.3. Tecnología: Lucene

Para el desarrollo de las búsquedas textuales, se utilizó la librería de código abierto Apache Lucene [50]. Esta librería es de amplia utilización para la implementación de este tipo de búsquedas, pues permite obtener buenos resultados de forma rápida y sencilla. Otras opciones manejadas inicialmente pasaban por la implementación de un sistema propio de cálculo de índices TF/IDF, que fue descartada por innecesariamente compleja, no reportando beneficios frente a la utilización de esta librería y suponiendo en contrapartida un gran esfuerzo de implementación.

Las funcionalidades básicas de Lucene pasan por la creación de índices de documentos, y la ejecución de búsquedas textuales sobre estos índices para la recuperación de los documentos. Con el propósito de ser lo más independiente posible del formato de tales documentos, define interfaces para abstraerse de ellos, manejando una representación lógica de los mismos consistente en un registro con una serie de campos de texto, quedando en el programador la tarea de definir los campos a utilizar y rellenarlos con la información que considere adecuada; de este modo, la flexibilidad es máxima.

Definidas tales representaciones, Lucene ofrece servicios como descartar palabras vacías, lematización, búsqueda por un campo del documento o por una combinación de varios, devolviendo resultados de forma rápida y precisa.

4.7. Componente: Traducción de Idiomas

En este apartado explicaremos las características del componente *Traducción de Idiomas*, de la arquitectura de la aplicación.

4.7.1. Funcionalidad

El componente *Traducción de Idiomas*, perteneciente a la capa de negocio de la arquitectura de la aplicación, encapsula la funcionalidad necesaria para la traducción automática de palabras de un idioma a otros. Se encarga del manejo de servicios externos de traducción, incluyendo el envío y recepción de mensajes, y su interpretación para la obtención de resultados. Los idiomas considerados son español, inglés, francés y alemán.

Este componente es necesario para el funcionamiento de las ayudas de traducción automática de etiquetas ofrecidas al usuario en la interfaz de la aplicación.

4.7.2. Interfaces

Como se muestra en la arquitectura de la aplicación, este componente ofrece los servicios de la interfaz *ITraduccionIdiomas* a las capas superiores. Estos servicios representan tareas básicas para la obtención de traducciones automáticas, y se detallan a continuación:

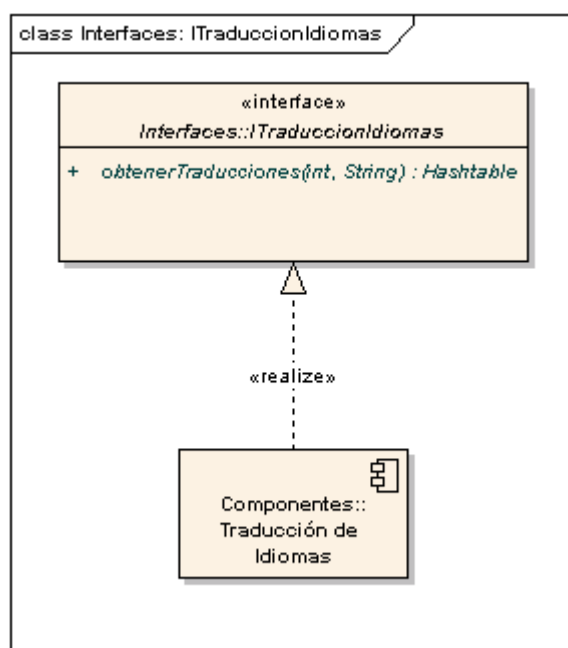


Ilustración 44: Interfaz ITraduccionIdiomas

Operación: obtenerTraducciones(codIdiomaOrig: int, etiquetas: String) : Hashtable
Parámetros: <i>codIdiomaOrig</i> : Código del idioma en que están escritas las etiquetas a traducir. <i>etiquetas</i> : Etiquetas a traducir.
Descripción: Este método traduce las etiquetas indicadas a los idiomas inglés, francés, español y alemán.
Retorno: Traducciones en los diferentes idiomas organizadas como Hashtable.

Tabla 31: Interfaz ITraduccionIdiomas – método obtenerTraducciones

4.7.3. Tecnología: Traductor CLEF

Para el desarrollo del componente, se pensó en el uso de herramientas de traducción automática ya existentes, para facilitar su desarrollo. En un análisis de las posibilidades existentes, se descubrió la existencia de una aplicación previa creada por Alejandro Blasco Plaza [51], antiguo alumno de la Universidad Carlos III, para su Proyecto Fin de Carrera, que ofrecía la funcionalidad necesaria: traducción automática de palabras sueltas a gran variedad de idiomas, entre los que se incluían los cuatro planteados como necesarios en nuestro experimento. Se decidió su utilización para el proyecto.

El funcionamiento de esta aplicación se apoya en páginas web desarrolladas por terceros y dedicadas a la traducción automática de documentos, como puedan ser Babelfish, FreeTranslation o Intertran. El programa envía peticiones a estas páginas (permitiendo al usuario escoger a cuáles)

con las palabras a traducir y los idiomas de origen y destino. Espera su respuesta y la procesa, devolviendo un resumen al usuario con los resultados, formateado en XML para su fácil proceso.

Esta aplicación funciona a través de web. Permite conectarse a la misma a través de una determinada URL, en la que se codifican los diversos parámetros de la petición: palabras a traducir, idiomas de origen y destino, y traductores a utilizar. De este modo, el acoplamiento entre aplicaciones es mínimo, reduciéndose a los mensajes enviados y recibidos.

4.8. Componente: *Imágenes Parecidas*

En este apartado explicaremos las características del componente *Imágenes Parecidas*, de la arquitectura de la aplicación.

4.8.1. Funcionalidad

El componente *Imágenes Parecidas*, perteneciente a la capa de negocio de la arquitectura de la aplicación, encapsula la lógica necesaria para la obtención de imágenes de la colección del sistema parecidas a otra dada. Se encarga del manejo de servicios para la creación de índices de imágenes y su procesado, que permitan la ejecución de consultas directas sobre imágenes parecidas.

Este componente es básico para el funcionamiento de las ayudas de imágenes parecidas ofrecidas al usuario en la interfaz durante el desarrollo del experimento.

4.8.2. Interfaces

Como se muestra en la arquitectura de la aplicación, este componente ofrece los servicios de la interfaz *ImágenesParecidas* a las capas superiores. Estos servicios representan tareas básicas para la búsqueda de imágenes parecidas, y se detallan a continuación:

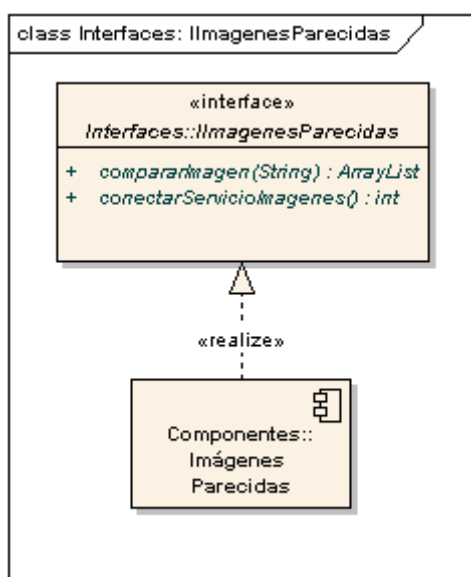


Ilustración 45: Interfaz IImágenesParecidas

Operación: compararImagen(urlImagen: String) : ArrayList
Parámetros: <i>urlImagen</i> : URL donde puede accederse a la imagen de la cual se deben buscar otras parecidas.
Descripción: Este método tomará la URL de una imagen y buscará otras imágenes de la colección del sistema que se consideren visualmente parecidas.
Retorno: ArrayList con los resultados, indicando en cada posición los nombres de las imágenes parecidas.

Tabla 32: Interfaz IImágenesParecidas – método compararImagen

Operación: conectarServicioImágenes() : int
Parámetros: Ninguno.
Descripción: Este método trata de conectar con el servicio para la obtención de imágenes parecidas. Debe ejecutarse previamente a las llamadas a compararImagen.
Retorno: Número de la sesión obtenida.

Tabla 33: Interfaz IImágenesParecidas – método conectarServicioImágenes

4.8.3. Recuperación de imágenes basada en su contenido

El desarrollo de este componente pasa por la utilización de servicios de recuperación de imágenes mediante el análisis directo de sus contenidos, en contraposición a los servicios más habituales de búsqueda basados en textos relacionados.

La recuperación de imágenes basada en su contenido (del inglés *content-based image retrieval*, *CBIR*) trata de resolver el problema de la búsqueda de imágenes mediante técnicas de reconocimiento visual. Como hemos comentado ya en este documento, actualmente la mayoría de los sistemas de búsqueda de imágenes están basados en búsquedas textuales sobre los metadatos o textos relacionados con la imagen; ejemplos son Google Images [52] o, como hemos visto, cualquier folksonomía para imágenes, como pueda ser Flickr [4]. Estos sistemas no son siempre adecuados, pues pudiera por ejemplo darse el caso en que sea difícil o costosa la obtención de metadatos para un conjunto de imágenes. La recuperación basada en el contenido, en cambio, busca por así decirlo “leer” en las mismas imágenes, para poder así responder más exactamente a las consultas de los usuarios.

El método de consulta en estos sistemas varía dependiendo de su tipo. Algunos, denominados de búsqueda por ejemplo, partirán de una determinada imagen elegida por el usuario y buscarán otras imágenes parecidas. Otros sistemas, de búsqueda semántica, pedirán al usuario una descripción de la imagen a buscar, como pueda ser “imágenes de caballos de pelo negro”. La complejidad de esta última tarea es mucho mayor a la primera, pues se une la interpretación de la petición del usuario a la búsqueda de imágenes coincidentes; los prototipos existentes suelen restringirse a un determinado dominio, tratando de simplificar una tarea de otro modo demasiado compleja.

Para la comparación de imágenes, las técnicas habitualmente utilizadas se centran en características visuales como la distribución del color, las texturas o la búsqueda de siluetas de figuras. Son técnicas basadas en características de bajo nivel, para las que existen técnicas aproximadas como la comparación de histogramas de color, detección de patrones o detección de bordes. Nuevamente, el ideal sería conseguir la correspondencia de esta información de bajo nivel con conceptos que describan tal contenido, como descubrir que una imagen representa un “perro” o un “amanecer”, pero este escalón semántico no es aún fácilmente salvable.

Aún a pesar de que su estado de desarrollo sea todavía experimental en la mayor parte de los casos, es un campo con mucho futuro y de interesante estudio. Prueba de ello es que, para dominios controlados como puedan ser comparación de radiografías, colecciones de arte o estudios topográficos, se han conseguido desarrollar sistemas de más que aceptable eficacia.

4.8.4. Tecnología: GIFT

Para la tarea de obtener imágenes parecidas a otra, se pensó en la utilización de un servicio de CBIR. Como se ha descrito antes, las tareas ejecutadas por los mismos serían de utilidad para nuestra labor. Habría de buscarse un sistema capaz de, dada una base de imágenes, recibir una imagen no necesariamente presente en tal base, y obtener a partir de ella las imágenes más

parecidas a la misma.

Los ejemplos de sistemas capaces de tal tarea son amplios, como imgSeek [53] o FIRE [54]. Finalmente, se escogió el uso de GNU Image-Finding Tool (GIFT) [55], por ajustarse a la funcionalidad requerida por el proyecto, y por recomendación del tutor, al haber sido utilizado en otros proyectos con buenos resultados.

GIFT es un sistema abierto de código libre para la búsqueda de imágenes basada en el contenido. Permite la búsqueda sobre colecciones de imágenes previamente indizadas en el sistema. Operaciones permitidas sobre ellas son la obtención de imágenes aleatorias, y obtener imágenes parecidas a una o varias indicadas como relevantes, sean las imágenes de partida parte de la colección o externas. La creación de índices de imágenes puede hacerse de forma automática mediante varios servicios presentes en GIFT.

GIFT utiliza un modelo de cliente-servidor para su funcionamiento, buscando facilitar y promover la aparición de interfaces de muy diverso tipo que hagan uso de sus servicios. El protocolo para el intercambio de mensajes, basado en XML y bastante documentado, se denomina MRML [56]. Ofrece una interfaz de demostración basada en PHP, que sirve para estudiar cómo se realiza el intercambio de mensajes, y como interfaz básica para la interacción con GIFT.

Collection ([details](#)): Algorithm: Return images

Algorithm options:
Modify default configuration:

Image uploading feature disabled

(fetch a random set of images) (launch the query) (Clear the query)

Ilustración 46: Interfaz PHP de demostración de GIFT

Result:

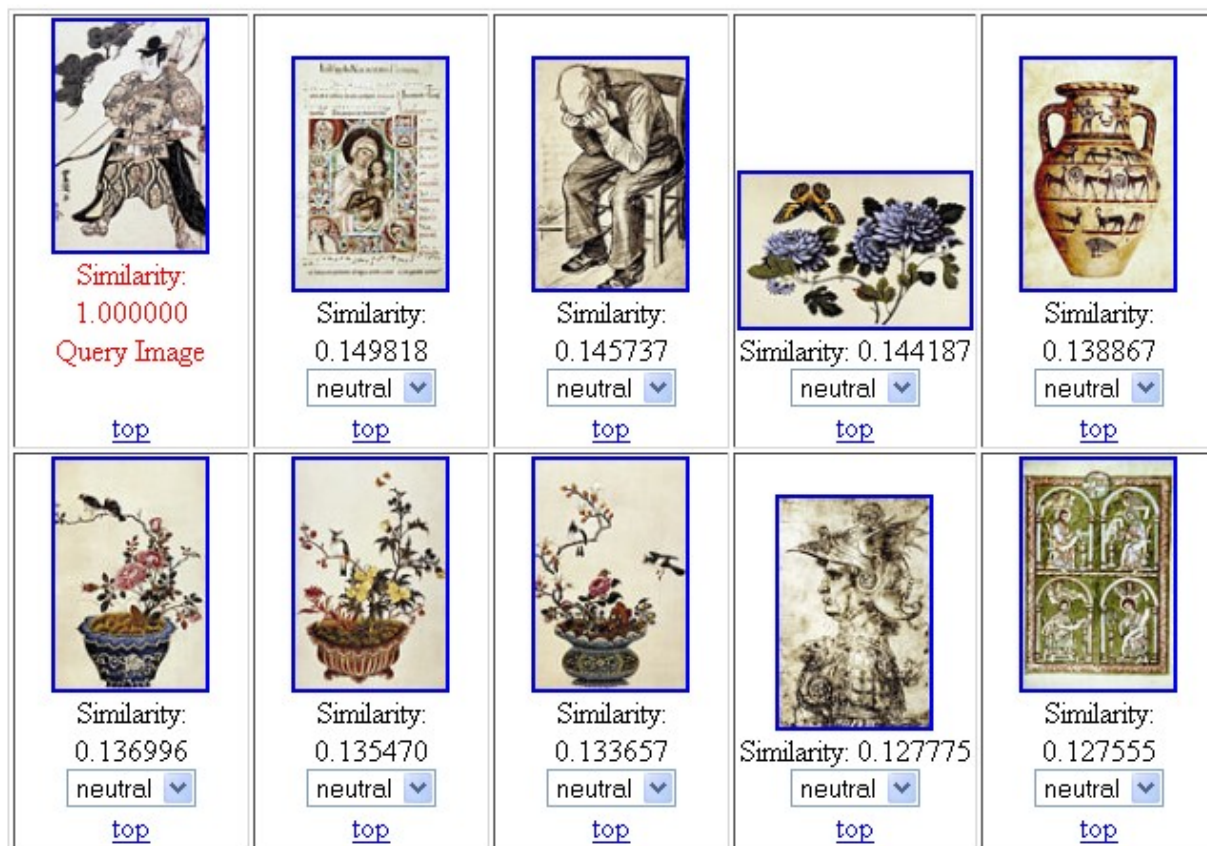


Ilustración 47: Resultados de una consulta sobre GIFT

4.8.4.1. Funcionamiento

Nuestra implementación hace uso de los servicios de GIFT para obtener una serie de imágenes de la base de conocimiento parecidas a una dada.

Como paso previo a cualquier otro, se debió crear un índice para la colección de imágenes, e instalar el servicio de escucha en una determinada máquina. Para ello, se guardaron todos los ficheros de imágenes en el servidor, asegurando que cada uno era accesible públicamente desde una URL, y se ejecutó el servicio de indización de GIFT. Tras ello, se debió activar el servicio de escucha de GIFT, que permite la espera de peticiones, y el atender cada una de ellas.

Hecho esto, en el proceso para la obtención de resultados, se siguen los siguientes pasos:

1. Se envía un mensaje a la URL y puerto donde escucha el servicio de GIFT, pidiéndole la apertura de una sesión.

2. Se espera la respuesta del servidor, que nos informará del número de la sesión abierta, los algoritmos y colecciones de imágenes disponibles.
3. Se envía un mensaje a la URL y puerto donde escucha el servicio de GIFT, con el número de sesión antes indicado y la URL de la imagen que se quiere comparar, pidiendo que nos envíe las imágenes parecidas a la indicada.
4. Se espera la respuesta del servidor, que nos informará de las URLs de las imágenes más parecidas, indicando un grado de acierto para cada una de ellas.

4.9. Componente: Almacenamiento

En este apartado explicaremos las características del componente *Almacenamiento*, de la arquitectura de la aplicación.

4.9.1. Funcionalidad

El componente *Almacenamiento*, perteneciente a la capa de datos de la arquitectura de la aplicación, encapsula la lógica necesaria para el manejo del almacenamiento persistente de los datos del sistema. Se encarga del control de los diversos sistemas de almacenamiento a bajo nivel, ejecutando consultas y añadiendo información utilizando los mecanismos específicos de cada uno de ellos, y ofrece servicios a las capas superiores para la utilización de estos datos abstrayéndose de los detalles puntuales de su estructura de almacenamiento.

4.9.2. Interfaces

Como se muestra en la arquitectura de la aplicación, este componente ofrece los servicios de tres interfaces, una dedicada al manejo de los datos de la aplicación para su guardado, otra para su consulta, y la última para el manejo de los logs, registros de acciones del sistema.

4.9.2.1. Interfaz *IAccesoDatosGuardado*

La interfaz *IAccesoDatosGuardado* ofrece a las capas superiores servicios para el guardado de información del dominio del sistema de forma persistente, como puedan ser datos de imágenes, usuario, o etiquetas relacionadas. Estos servicios se detallan a continuación:

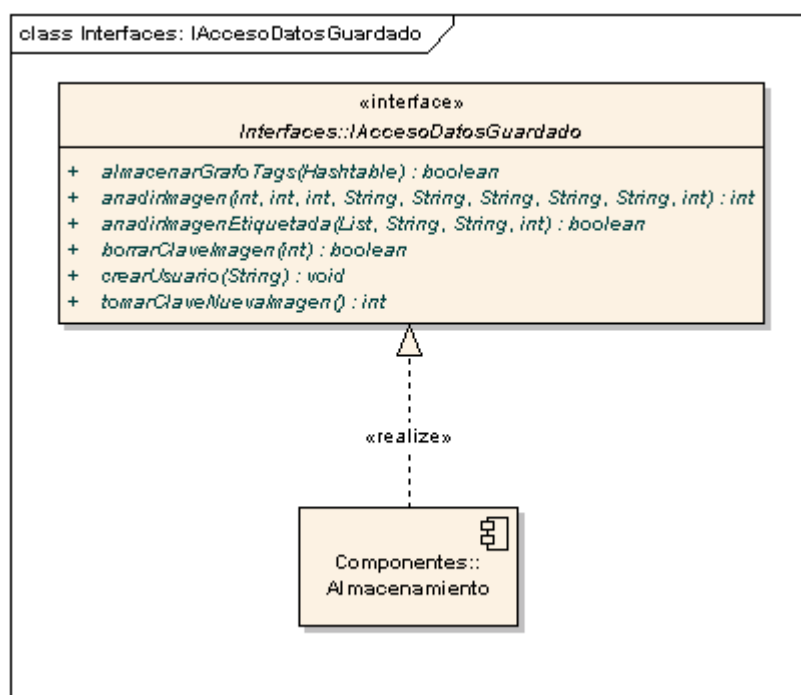


Ilustración 48: Interfaz IAcessoDatosGuardado

Operación: almacenarGrafoTags(grafoTags: Hashtable) : boolean

Parámetros:

grafoTags: Estructura que almacena el grafo de etiquetas relacionadas.

Descripción: Este método guardará la estructura del grafo de etiquetas relacionadas en almacenamiento persistente para su posterior consulta.

Retorno: Valor *true* si se ha completado correctamente la operación; *false* en caso contrario.

Tabla 34: Interfaz IAcessoDatosGuardado - método almacenarGrafoTags

Operación: <code>anadirImagen(kb: int, alto: int, ancho: int, thumb_nueva: String, thumb_ver: String, thumb_busq: String, url_orig: String, fichero: String, clave: int) : boolean</code>
Parámetros: <i>kb</i> : Tamaño en kilobytes de la imagen. <i>alto</i> : Número de píxeles de alto de la imagen. <i>ancho</i> : Número de píxeles de ancho de la imagen. <i>thumb_nueva</i> : Nombre del fichero que guarda la imagen reducida a un tamaño adecuado para su vista en el cuadro de “nueva imagen”. <i>thumb_ver</i> : Nombre del fichero que guarda la imagen reducida a un tamaño adecuado para su vista en el cuadro de detalles de la imagen. <i>thumb_busq</i> : Nombre del fichero que guarda la imagen reducida a un tamaño adecuado para su vista en los resultados de las búsquedas. <i>url_orig</i> : URL inicial de donde se tomó la imagen. <i>fichero</i> : Nombre del fichero que guarda la imagen. <i>clave</i> : Clave asignada a la imagen.
Descripción: Este método guardará los datos de una nueva imagen.
Retorno: Valor <i>true</i> si se ha completado correctamente la operación; <i>false</i> en caso contrario.

Tabla 35: Interfaz *IAccesoDatosGuardado* - método *anadirImagen*

Operación: <code>anadirImagenEtiquetada(tags: List, descripcion: String, titulo: String, clave: int) : boolean</code>
Parámetros: <i>tags</i> : Listado de etiquetas utilizadas con la imagen. <i>descripcion</i> : Descripción de la imagen. <i>titulo</i> : Título de la imagen. <i>clave</i> : Clave asignada a la imagen.
Descripción: Este método añadirá información de etiquetado a una imagen guardada.
Retorno: Valor <i>true</i> si se ha completado correctamente la operación; <i>false</i> en caso contrario.

Tabla 36: Interfaz *IAccesoDatosGuardado* - método *anadirImagenEtiquetada*

Operación: <code>borrarClaveImagen(clave: int) : boolean</code>
Parámetros: <i>clave</i> : Clave de la imagen.
Descripción: Este método borrará los datos de la imagen identificada con la clave indicada.
Retorno: Valor <i>true</i> si se ha completado correctamente la operación; <i>false</i> en caso contrario.

Tabla 37: Interfaz *IAccesoDatosGuardado* - método *borrarClaveImagen*

Operación: crearUsuario(usuario: String) : void
Parámetros: usuario: Nombre del usuario.
Descripción: Este método anota en el almacenamiento persistente la participación de un nuevo usuario.
Retorno: Ninguno.

Tabla 38: Interfaz IAccesoDatosGuardado - método crearUsuario

Operación: tomarClaveNuevaImagen() : int
Parámetros: Ninguno.
Descripción: Este método busca en el almacenamiento persistente una clave de imagen que no haya sido asignada a ninguna otra, y la reserva para una nueva imagen.
Retorno: Clave asignada a la nueva imagen.

Tabla 39: Interfaz IAccesoDatosGuardado - método tomarClaveNuevaImagen

4.9.2.2. Interfaz IAccesoDatosConsulta

La interfaz *IAccesoDatosConsulta* ofrece a las capas superiores servicios para la consulta de información del dominio del sistema almacenada previamente de forma persistente. Permite consultar la información relacionada a las imágenes del sistema, comprobar si un usuario ha iniciado ya sesión, obtener aleatoriamente una clave de una imagen del sistema, etcétera. Estos servicios se detallan a continuación:

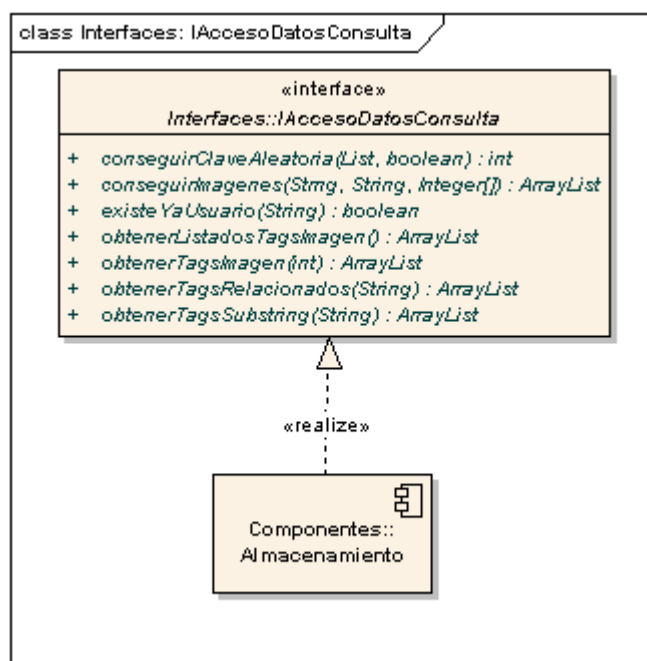


Ilustración 49: Interfaz IAcessoDatosConsulta

Operación: conseguirClaveAleatoria(clavesEvitar: List, deOtroUsuario: boolean) : int
Parámetros: <i>clavesEvitar</i> : Listado con diversas claves que no deben aparecer como resultado. <i>deOtroUsuario</i> : Valor booleano que indica si la imagen debe haber sido introducida por otro usuario, o ser de los elementos iniciales de la colección del sistema.
Descripción: Este método obtiene una clave de imagen aleatoria, atendiendo a las restricciones indicadas.
Retorno: Clave aleatoria obtenida.

Tabla 40: Interfaz IAcessoDatosConsulta - método conseguirClaveAleatoria

Operación: conseguirImagenes(listaClaves: Integer[], urlImg: String, urlThumbs: String) : ArrayList
Parámetros: <i>listaClaves</i> : Listado con las claves de las imágenes. <i>urlImg</i> : URL base del directorio donde se guardan las imágenes. <i>urlThumbs</i> : URL base del directorio donde se guardan las imágenes reducidas para su presentación.
Descripción: Este método los datos de una serie de imágenes, identificadas como parámetro indicando sus claves.
Retorno: Listado con los datos de las imágenes indicadas.

Tabla 41: Interfaz IAcessoDatosConsulta - método conseguirImagenes

Operación: existeYaUsuario(usuario: String) : boolean
Parámetros: <i>usuario</i> : Nombre del usuario buscado.
Descripción: Este método comprueba si se ha utilizado anteriormente determinado nombre de usuario para entrar a la aplicación.
Retorno: Valor <i>true</i> si el usuario ya existe, <i>false</i> en caso contrario.

Tabla 42: Interfaz *IAccesoDatosConsulta* - método *existeYaUsuario*

Operación: obtenerListadosTagsImagen() : ArrayList
Parámetros: Ninguno.
Descripción: Este método devuelve todas las etiquetas de todas imágenes guardadas en el sistema.
Retorno: Listado de todas las etiquetas de todas las imágenes guardadas en el sistema.

Tabla 43: Interfaz *IAccesoDatosConsulta* - método *obtenerListadosTagsImagen*

Operación: obtenerTagsImagen(clave: int) : ArrayList
Parámetros: <i>clave</i> : Clave de la imagen.
Descripción: Este método obtiene las etiquetas aplicadas a la imagen con clave igual a la indicada.
Retorno: Listado de etiquetas de la imagen.

Tabla 44: Interfaz *IAccesoDatosConsulta* - método *obtenerTagsImagen*

Operación: obtenerTagsRelacionados(tag1: String) : ArrayList
Parámetros: <i>tag1</i> : Etiqueta de la que se buscan otras etiquetas relacionadas.
Descripción: Este método obtiene las etiquetas relacionadas con otra en concreto, según los datos del grafo de etiquetas relacionadas almacenados.
Retorno: Listado de etiquetas relacionadas.

Tabla 45: Interfaz *IAccesoDatosConsulta* - método *obtenerTagsRelacionados*

Operación: obtenerTagsSubstring(cadena: String) : ArrayList
Parámetros: cadena: Cadena de texto.
Descripción: Este método obtiene etiquetas que contengan la cadena de texto indicada como parámetro.
Retorno: Listado de etiquetas conteniendo la cadena de texto.

Tabla 46: Interfaz *IAccesoDatosConsulta* - método *obtenerTagsSubstring*

4.9.2.3. Interfaz *IAccesoDatosLogs*

La interfaz *IAccesoLogs* ofrece a las capas superiores servicios para el manejo de los logs del sistema, añadiendo eventos e información a los mismos según consideren necesario. Estos servicios permiten la obtención de los controladores de estos ficheros de log, y se detallan a continuación:

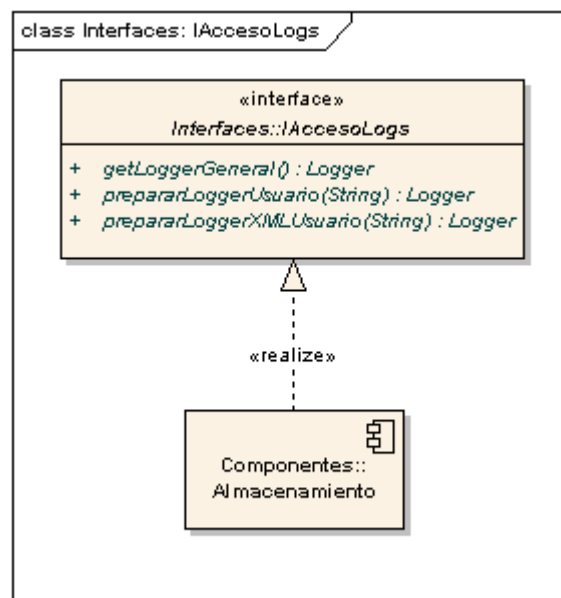


Ilustración 50: Interfaz *IAccesoLogs*

Operación: prepararLoggerGeneral() : Logger
Parámetros: Ninguno.
Descripción: Este método preparará el log general de la aplicación, esto es, el log que guardará las trazas de los mensajes de importancia en la ejecución de la aplicación en su conjunto, como errores de ejecución u otros estados excepcionales.
Retorno: Objeto Logger que representa el log y permite su manejo.

Tabla 47: Interfaz IAccesoLogs - método prepararLoggerGeneral

Operación: prepararLoggerUsuario(usuario: String) : Logger
Parámetros: <i>usuario</i> : Nombre del usuario.
Descripción: Este método preparará el log de la ejecución de la aplicación para un usuario en concreto, indicado como parámetro.
Retorno: Objeto Logger que representa el log y permite su manejo.

Tabla 48: Interfaz IAccesoLogs - método prepararLoggerUsuario

Operación: prepararLoggerXMLUsuario(usuario: String) : Logger
Parámetros: <i>usuario</i> : Nombre del usuario.
Descripción: Este método preparará el log de la ejecución de la aplicación para un usuario en concreto, indicado como parámetro. Este log se diferencia del otro presente para un usuario en que se destinará al guardado de información en formato XML, para permitir posteriormente su fácil proceso y análisis de las acciones del usuario.
Retorno: Objeto Logger que representa el log y permite su manejo.

Tabla 49: Interfaz IAccesoLogs - método prepararLoggerXMLUsuario

4.9.3. Tecnología: Log4J

En la creación de logs, fue de inestimable ayuda la utilización de la librería para Java de libre distribución Log4J [57]. Esta librería, proyecto de la Apache Software Foundation, es ampliamente utilizada para el manejo de ficheros de log en programas Java, gracias a su extensa funcionalidad, capacidad de configuración y fácil aprendizaje.

Log4J maneja automáticamente la escritura y creación de ficheros de log, permitiendo su configuración mediante un fichero externo de propiedades. Allí, es posible la especificación de *Loggers*, nombres lógicos para los logs que indican sus orígenes, *Appenders*, mecanismos para gestionar cómo se añadirá la información a log (opciones son indicar la ubicación del fichero, limitar su tamaño, hacer que se guarde una copia al fin de cada día, y demás), y *Layouts*, que

controlan el formato de la información guardada en cada log.

Una de las características más celebradas de Log4J es su capacidad para dar grados de importancia a los mensajes del log, según criterio del usuario. Estos niveles van desde TRACE, destinado a los mensajes con importancia únicamente para el testeado del correcto funcionamiento de la aplicación, hasta FATAL, destinado a los errores fatales que impidan la ejecución del programa. Un log puede ser configurado para filtrar los mensajes según su importancia desde el fichero de configuración de Log4J, sin necesidad de modificar el código de los programas.

4.9.4. Tecnología: MySQL

Para la gestión del almacenamiento persistente más allá de los ficheros de log, se consideró como opción más sencilla, práctica y útil el manejo de una base de datos relacional. Esta opción permite el guardado de forma completamente estructurada de los datos del dominio, y su posterior consulta mediante un lenguaje de interrogación plenamente estandarizado y potente (SQL).

Entre los diferentes gestores de bases de datos existentes en la actualidad, se optó por MySQL [58]. Esta opción es también muy común en proyectos como el nuestro, de pequeño tamaño y poca duración en el tiempo. Su principal ventaja es la de ser software libre, por tanto utilizable de forma gratuita, y con un rendimiento lo suficientemente bueno como para no necesitar un sistema gestor más potente dentro de proyectos pequeños.

4.9.5. Modelos de datos utilizados

En esta sección describiremos las estructuras de datos utilizadas para el guardado de la información del dominio del sistema, tanto por medio de bases de datos como de logs, buscando ofrecer una comprensión profunda de la información manejada por el sistema.

4.9.5.1. Esquema de base de datos

La información del dominio manejada por el sistema queda almacenada en un esquema de base de datos, que le aporta estructura y organización. La estructura de este esquema es la que sigue:

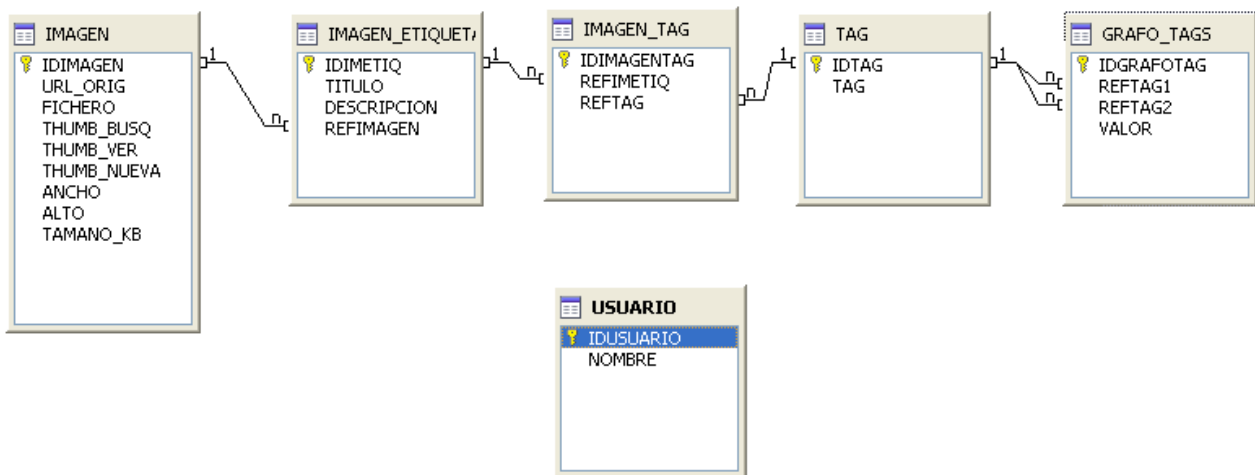


Ilustración 51: Esquema de base de datos

Las diferentes tablas presentes en el esquema, con sus datos y relaciones, son éstos:

- **Tabla Imagen:** Contiene información sobre las imágenes del sistema.
 - *IdImagen:* Identificador de cada tupla, generado automáticamente. Se toma como clave de la imagen.
 - *URL_Orig:* URL de donde fue tomada originalmente la imagen.
 - *Fichero:* Nombre del fichero de la imagen.
 - *Thumb_busq:* Nombre del fichero con la imagen reducida para ser mostrada como resultado de las búsquedas.
 - *Thumb_ver:* Nombre del fichero con la imagen reducida para ser mostrada en la vista de las propiedades de una imagen.
 - *Thumb_nueva:* Nombre del fichero con la imagen reducida para ser mostrada en el cuadro de nueva imagen.
 - *Ancho:* Tamaño en píxeles del ancho de la imagen.
 - *Alto:* Tamaño en píxeles del alto de la imagen.
 - *Tamano_kb:* Tamaño en kilobytes del fichero de la imagen.
- **Tabla Imagen Etiquetada:** Contiene información sobre los datos textuales añadidos a cada imagen del sistema.
 - *IdImEtiq:* Identificador de cada tupla, generado automáticamente.
 - *Título:* Título asignado a la imagen.
 - *Descripción:* Descripción asignada a la imagen.
 - *RefImagen:* Referencia a tabla Imagen, apuntando a la imagen descrita.
- **Tabla Tag:** Contiene información sobre las etiquetas presentes en el sistema.
 - *IdTag:* Identificador de cada tupla, generado automáticamente.
 - *Tag:* Texto de la etiqueta.
- **Tabla Imagen Tag:** Relaciona las imágenes etiquetadas con sus respectivas etiquetas.
 - *IdImagenTag:* Identificador de cada tupla, generado automáticamente.
 - *RefImEtiq:* Referencia a tabla Imagen Etiquetada, apuntando a la imagen

- correspondiente.
- *RefTag*: Referencia a tabla Tag, apuntando a la etiqueta aplicada.
 - **Tabla Grafo_Tags**: Une pares de etiquetas, indicando un grado de relación entre cada par, necesario para la consulta de etiquetas relacionadas en las ayudas del sistema.
 - *IdGrafoTag*: Identificador de cada tupla, generado automáticamente.
 - *RefTag1*: Referencia a tabla Tag, apuntando a la primera etiqueta.
 - *RefTag2*: Referencia a tabla Tag, apuntando a la segunda etiqueta.
 - *Valor*: Valor entre 0 y 1 que indica el grado de relación entre las dos etiquetas.
 - **Tabla Usuario**: Tabla que guarda los nombres de los usuarios que han iniciado sesión en el sistema.
 - *IdUsuario*: Identificador de cada tupla, generado automáticamente.
 - *Usuario*: Nombre del usuario.

4.9.5.2. Estructura de los logs de usuario

La interacción de cada usuario con la aplicación del experimento queda registrada en varios ficheros de log, de los cuales el más relevante es uno con estructura de fichero XML que recoge de forma estructurada cada uno de los datos de relevancia de las acciones del usuario en el sistema, y permite su posterior proceso y análisis.

En esta sección, describiremos la estructura de estos ficheros XML, que nos permitirá entender qué información se toma de cada interacción. Utilizaremos el modelo gráfico utilizado por XMLSpy [38] para su representación gráfica.

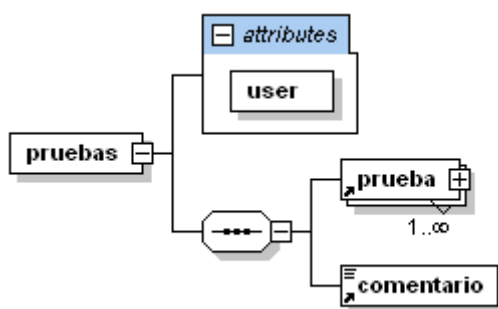
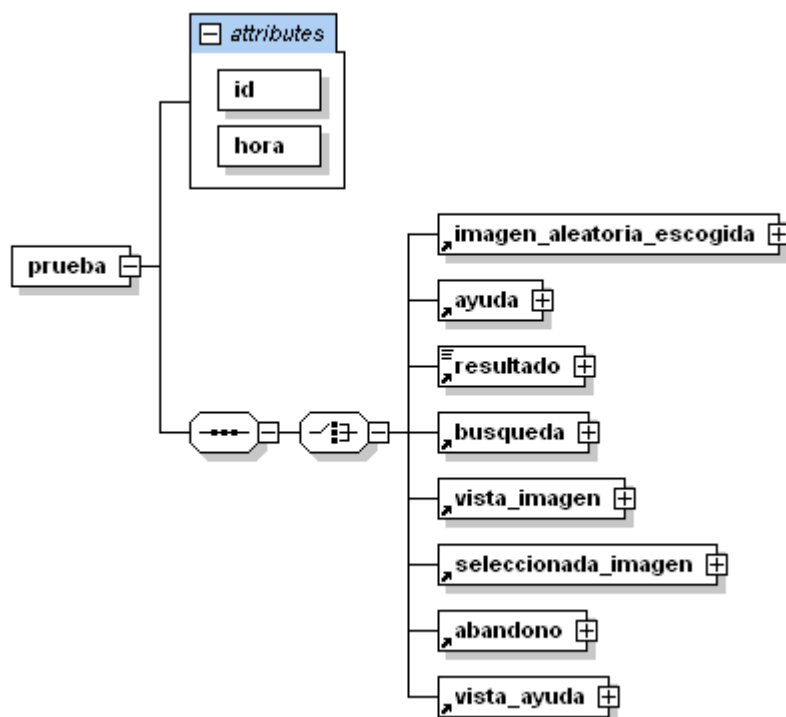


Ilustración 52: Estructura del xml de datos de interacción: elemento Pruebas

El elemento raíz del xml se denomina “pruebas”. Su atributo “user” indica el nombre del usuario indicado al inicio de la prueba. Contiene una serie de elementos “prueba”, con los datos de la interacción para cada prueba y que serán descritos en adelante, y un elemento final “comentario”, que recoge el texto que el usuario haya escrito como comentario final a las pruebas.



*Ilustración 53: Estructura del xml de datos de interacción:
elemento Prueba*

Cada elemento “prueba” queda identificado por dos atributos, “id”, número de 1 a 7 que indica el número de prueba de la que se recogen los datos, y “hora”, que indica el momento en que se inició. En ella se contienen una serie de etiquetas que indican diferentes eventos de la ejecución.

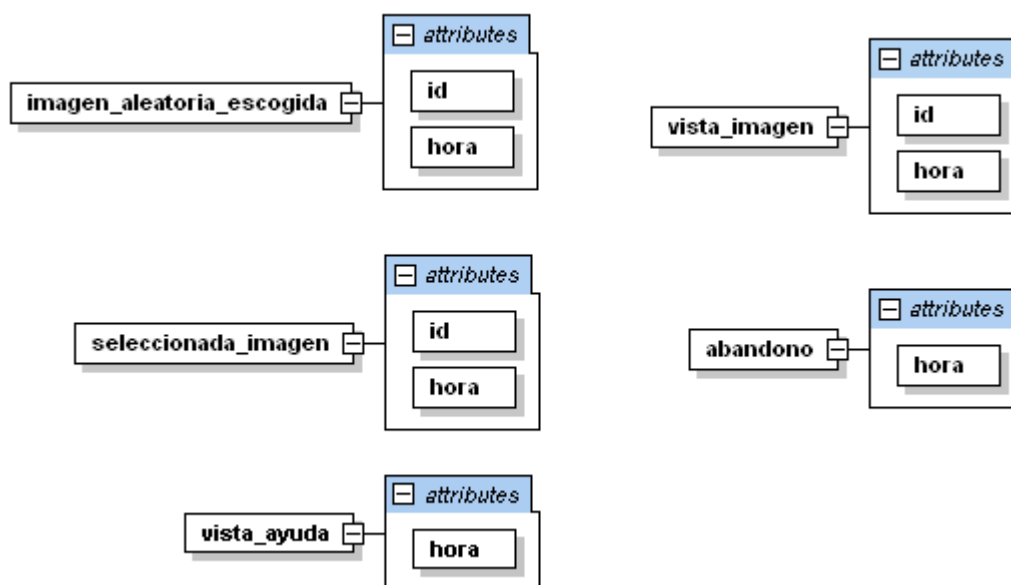


Ilustración 54: Estructura del xml de datos de interacción: elementos `ImagenAleatoriaEscogida`, `SeleccionadaImagen`, `VistaImagen`, `Abandono` y `VistaAyuda`

Cinco de estos elementos no tienen ningún contenido, sino simplemente una serie de atributos, y sirven para indicar eventos simples.

“`Imagen_aleatoria_escogida`” indica que se ha escogido aleatoriamente una imagen para iniciar una prueba (imagen aleatoria a buscar, por ejemplo), con atributos “`id`” indicando su clave y “`hora`” con el momento en que ocurrió.

“`Vista_imagen`”, en las pruebas de búsquedas, indica que el usuario ha pulsado sobre uno de las imágenes resultado de sus búsquedas, para ver sus detalles. El atributo “`id`” indica la clave de la imagen, y “`hora`” el momento en que ocurrió.

“`Seleccionada_imagen`”, en las pruebas de búsquedas, indica que el usuario ha pulsado sobre el botón “`Es imagen buscada`” de una imagen. El atributo “`id`” indica la clave de la imagen, y “`hora`” el momento en que ocurrió.

“`Abandono`”, en las pruebas de búsquedas, indica que el usuario ha decidido abandonar la prueba y pasar a la siguiente. El atributo “`hora`” indica el momento en que ocurrió.

“`Vista_ayuda`” indica que el usuario ha pulsado sobre el botón de “`Ayuda`” situado en la parte superior de la interfaz, mostrándosele textos de ayuda sobre el funcionamiento general de la aplicación. El atributo “`hora`” indica el momento en que ocurrió.

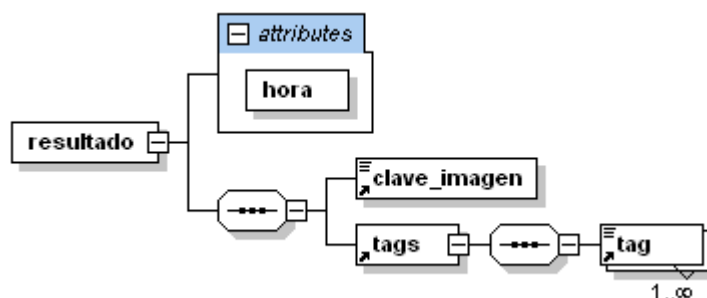


Ilustración 55: Estructura del xml de datos de interacción: elemento Resultado

Dentro de “prueba”, también puede aparecer el elemento “resultado”, que en las pruebas de añadido de imágenes indica las decisiones finales del usuario. El atributo “hora” indica la hora en que se guardó la imagen, el elemento ”clave_imagen” la clave que le fue asignada, y la serie de elementos “tag” las etiquetas que fueron utilizadas.

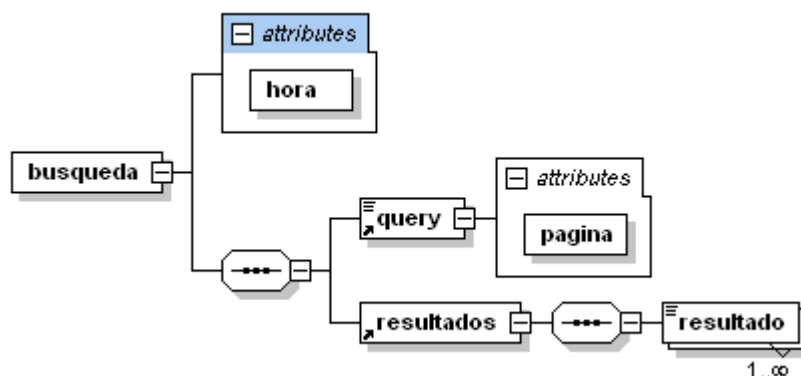


Ilustración 56: Estructura del xml de datos de interacción: elemento Búsqueda

Dentro de “prueba”, también puede aparecer el elemento “busqueda”, que en las pruebas de búsquedas indica la ejecución de una consulta. El atributo “hora” indica la hora en que se hizo la consulta, el elemento ”query” el texto de la consulta, con atributo “pagina” indicando el número de página vista, y la serie de elementos “resultado” las claves de las imágenes obtenidas como resultado.

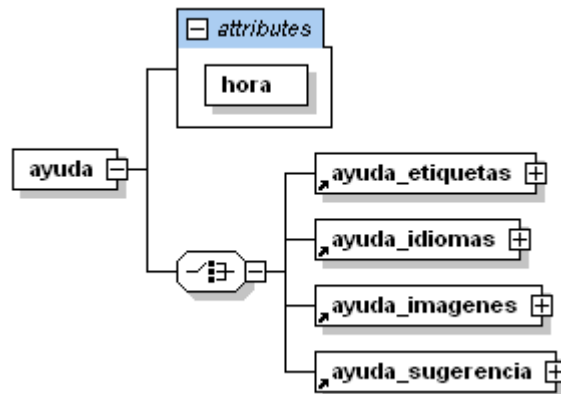


Ilustración 57: Estructura del xml de datos de interacción: elemento Ayuda

Dentro de “prueba”, también puede aparecer el elemento “ayuda”, que indica el ofrecimiento de una serie de sugerencias por las herramientas de ayuda. El atributo “hora” indica la hora en que ocurrió, y dependiendo del tipo de ayuda contendrá una de cuatro etiquetas, que describimos a continuación.



Ilustración 58: Estructura del xml de datos de interacción: elemento AyudaEtiquetas

Dentro de “ayuda”, figurará el elemento “ayuda_etiquetas” si la ayuda es de etiquetas relacionadas. El elemento “tag_orig” indica la etiqueta a partir de la que se calculan las relacionadas, que se guardan en una serie de elementos “tag”.

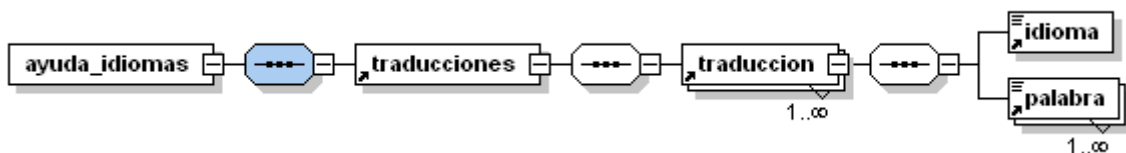


Ilustración 59: Estructura del xml de datos de interacción: elemento AyudaIdiomas

Dentro de “ayuda”, figurará el elemento “ayuda_idiomas” si la ayuda es de traducción automática. Contendrá una serie de elementos “traduccion”, conteniendo en cada uno de ellos las traducciones a un determinado idioma. Así, dentro de este elemento el elemento “idioma” indicará el idioma destino, y una serie de elementos “palabra” las sugerencias de palabras traducidas.

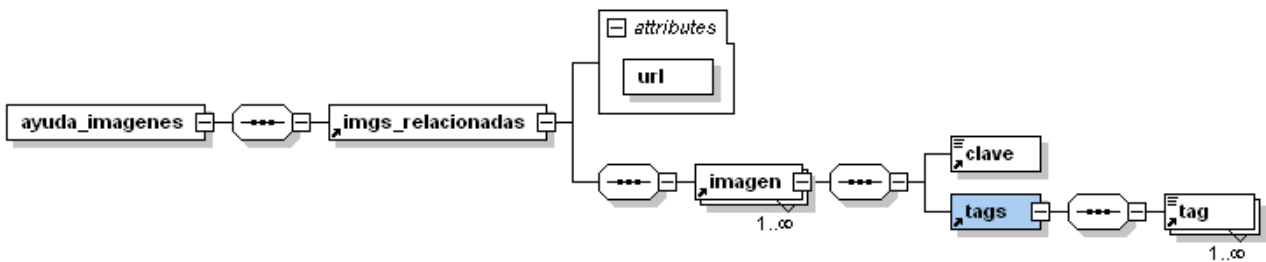


Ilustración 60: Estructura del xml de datos de interacción: elemento AyudaImagenes

Dentro de “ayuda”, figurará el elemento “ayuda_imagenes” si la ayuda es de imágenes parecidas. Dentro, un elemento “imgs_relacionadas” tendrá como atributo “url” la URL de la imagen pasada como parámetro al detector de imágenes parecidas, y una serie de elementos “imagen” con cada resultado. Cada uno de ellos tendrá un elemento “clave” con la clave de la imagen parecida, y una serie de elementos “tag” con cada una de sus etiquetas ofrecidas como sugerencia.

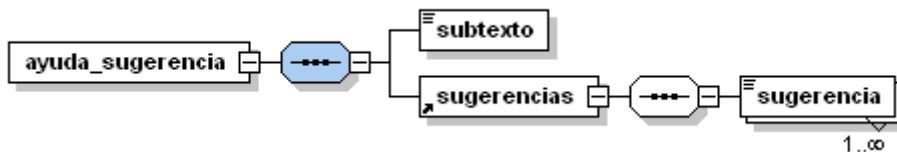


Ilustración 61: Estructura del xml de datos de interacción: elemento AyudaSugerencia

Dentro de “ayuda”, figurará el elemento “ayuda_sugerencia” si la ayuda es de sugerencias de autocompletado. El elemento “subtexto” indicará la cadena de texto escrita hasta el momento por el usuario, y una serie de elementos “sugerencia” indicarán las sugerencias de la herramienta para su completado.

5. Resultados del experimento

Completado el desarrollo de la aplicación, se pasó a entregar la aplicación a los usuarios para que estos pudieran ejecutar las tareas planteadas, y los datos de sus interacciones fuesen recogidos. Para ello, la aplicación se hizo pública desde la URL <http://miracle2.uc3m.es:8180/SugerenciaTags>, y se dio a conocer mediante correos y otros medios a una serie de usuarios, pidiéndoles que la utilizaran y distribuyesen a voluntad.

En el momento de escribir estas líneas y detallar los resultados obtenidos, tres semanas han transcurrido desde que se publicó la aplicación. Desde entonces, alrededor de 50 usuarios han al menos pinchado el enlace e iniciado la aplicación, pero no todos la ejecutaron hasta su fin: varios abandonaron sin completar la primera prueba, y otros terminaron algunas de las pruebas iniciales sin llegar a la final.

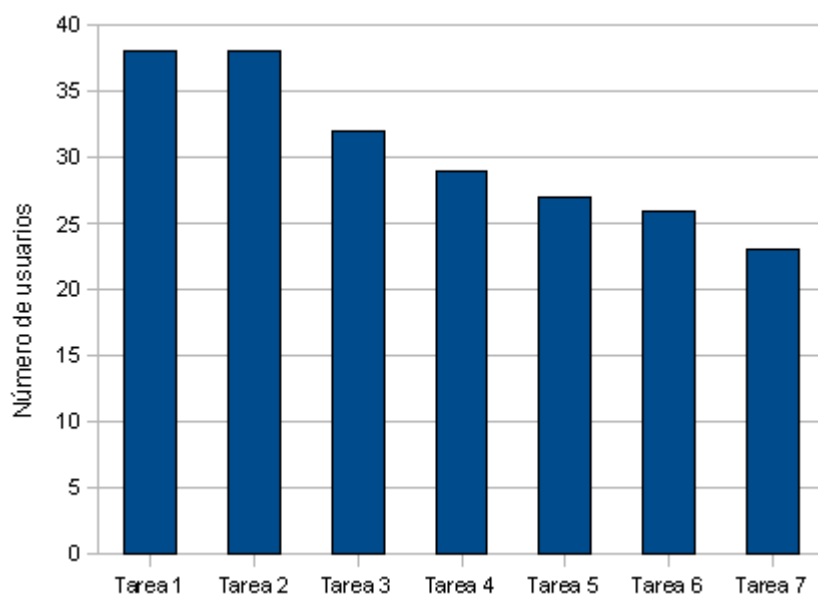


Ilustración 62: Número de usuarios que completó cada tarea

En la gráfica superior se muestra el número de usuarios que completaron cada una de las tareas de la aplicación. Como puede verse, el número disminuye con cada nueva prueba, partiendo de 38 y llegando a un mínimo de 23 en la última. Es éste un fenómeno esperable, pues cada nueva prueba supone un nuevo esfuerzo en el usuario, por humilde que éste sea: el de continuar atendiendo a la aplicación por unos minutos más. Dentro del contexto de Internet, donde la concentración en una única tarea es infrecuente y tiende a saltarse rápidamente de una página a otra, tanto éste como esfuerzos adicionales, como el subir imágenes propias en la tercera tarea, hacen abandonar a varios usuarios.

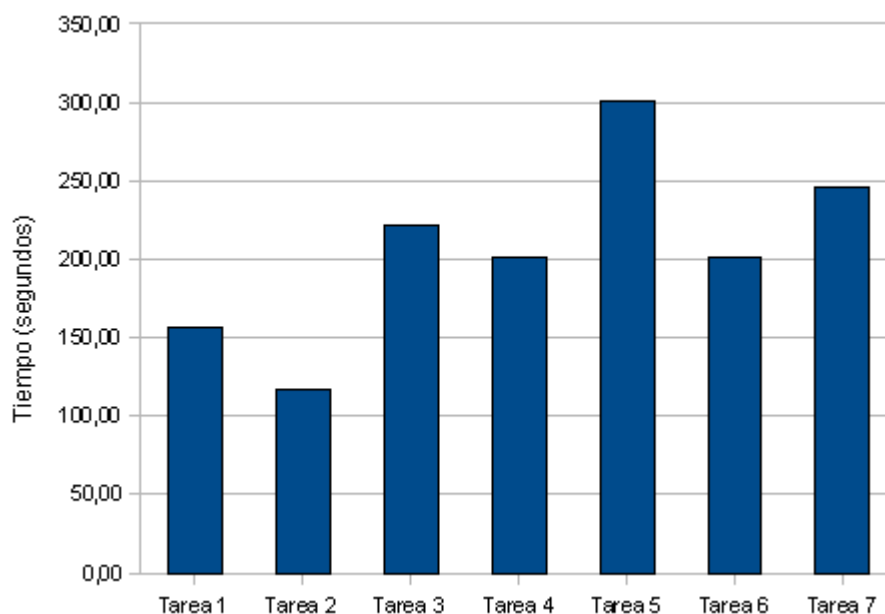


Ilustración 63: Tiempo medio de ejecución de cada tarea

En la ilustración superior se observa el tiempo medio dedicado a cada una de las tareas por los usuarios que las han completado. Puede verse que éste ronda sobre los tres minutos de media, con cierta variación de una a otra.

En las dos primeras tareas de añadido de imágenes se sitúan los mínimos, sin duda debido al esfuerzo limitado requerido por parte del usuario, que simplemente debe escoger una serie de etiquetas para imágenes que le son mostradas. Este tiempo aumenta en las dos siguientes, algo esperable debido al tiempo adicional requerido para escoger y enviar una imagen propia.

Las tres tareas siguientes son las de búsqueda. En una estimación previa, supusimos que el tiempo medio iría en aumento para cada una de las tres, debido a su dificultad creciente. La primera, de exploración libre, no pide al usuario más que navegue por la colección de imágenes y elija cinco que cree relacionadas en algún sentido con otra imagen que se le muestra, sean éstas cualesquiera. Las dos siguientes requieren mayor esfuerzo, al pedir al usuario encontrar una imagen en concreto, una tarea particularmente difícil de ejecutar en las folksonomías; de estas dos imágenes a buscar, se sospechaba la primera más sencilla, al escogerse entre las etiquetadas por otros usuarios de la aplicación, que en principio compartirían idioma y un vocabulario similar, frente a la segunda imagen, escogida aleatoriamente de la colección del sistema.

Sin embargo, el tiempo máximo se encuentra en la tarea de asociación libre: cabe imaginar que por enfrentar por vez primera a los usuarios con la interfaz de búsquedas, y por un empeño de exploración por parte de los usuarios, esforzándose por rebuscar en la colección tratando de encontrar imágenes directamente relacionadas con la mostrada. Las dos últimas, que suponen las pruebas de mayor dificultad, no llegan sin embargo a los tiempos máximos (acercándose a ellos en cualquier caso); posiblemente esto se deba a que los usuarios abandonaban estas pruebas (recordemos que en ellas se permite la opción Abandonar Prueba, y pasar automáticamente a la siguiente) cuando veían improbable encontrar la imagen indicada, o se cansaban de buscar.

Otro dato general digno de mención es que la ayuda de la aplicación, mostrada mediante el botón superior Ayudas, fue accedida por un total de 13 usuarios, de los 38 que completaron al menos una prueba.

5.1. Tareas de añadido de imágenes (de 1 a 4)

Las tareas 1 a 4 piden al usuario el añadido de una imagen al sistema, asociando a la misma una serie de etiquetas, un título y opcionalmente una descripción. Las tareas se distinguen unas de otras por el diferente origen de la imagen, y las ayudas al etiquetado ofrecidas. En las tareas 1 y 2, el sistema elige aleatoriamente la imagen a etiquetar de la colección del sistema, mientras que en las 3 y 4 es el usuario quien sube al sistema imágenes propias. Por otra parte, de las cuatro ayudas al etiquetado posibles (a saber: autocompletado, imágenes parecidas, etiquetas relacionadas y traducciones automáticas), en la tarea 1 se ocultan etiquetas relacionadas y traducciones automáticas, y en la tarea 2 únicamente traducciones automáticas; en el resto todas las ayudas están presentes.

Teniendo en cuenta esto, centraremos nuestra atención en el etiquetado que los usuarios han utilizado para las imágenes, distinguiendo si en él incluyen sugerencias de las ayudas ofrecidas.

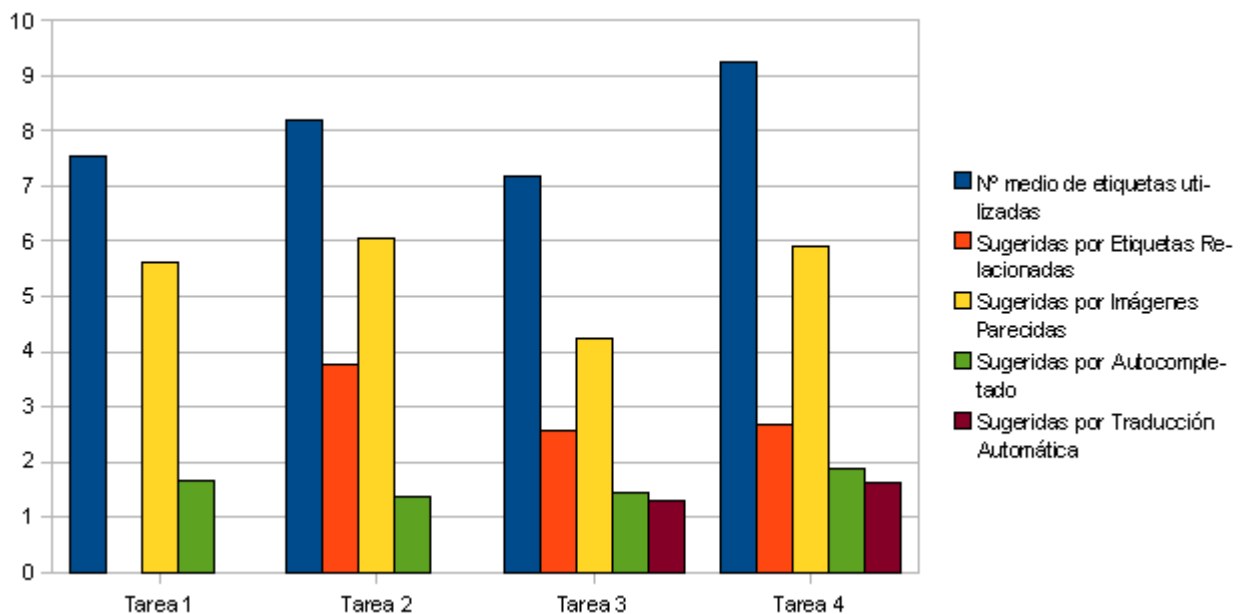


Ilustración 64: Origen de las etiquetas utilizadas, en promedio

En el gráfico anterior se muestran, para cada tarea, diversos promedios sobre su etiquetado: el número medio de etiquetas utilizadas por los usuarios para cada imagen, y de estas etiquetas, el número de las mismas que fueron ofrecidas como sugerencia por cada herramienta de ayuda. Obsérvese que si la etiqueta 'A' fue incluida por un usuario en sus etiquetas, y las herramientas de

ayuda B y C ofrecieron esta etiqueta como sugerencia, se contabiliza como válida para las dos herramientas B y C; de este modo, no necesariamente la suma del número de sugerencias aceptadas para cada herramienta será menor que el número de etiquetas utilizadas por los usuarios.

Los usuarios añadieron una media de 8 etiquetas por imagen, con un máximo de 62 etiquetas por imagen, y un mínimo de 1. Son sin duda unas cantidades elevadas, que ya habla de un primer efecto de las herramientas de ayuda, pues prescindiendo de ellas hubiera podido esperarse un número medio de etiquetas más reducido.

El gráfico habla también de un uso intenso de las herramientas de ayuda por parte de los usuarios para el etiquetado de sus imágenes. Ya en la primera tarea, se observa un número medio de 7,55 etiquetas por imagen, de las cuales 5,6 fueron previamente sugeridas por la herramienta de Imágenes Parecidas, y 1,6 por la de Autocompletado. Puede de ello deducirse que la mayoría de las etiquetas utilizadas por los usuarios les fueron ofrecidas por las herramientas de ayuda.

Puede verse una clara preponderancia de la ayuda de imágenes parecidas sobre el resto. Caben varias interpretaciones de ello, que hay que sopesar cuidadosamente. Por un lado, es con ello observable que los usuarios se interesan por esta ayuda y la utilizan. Por otro lado, debe tenerse en cuenta que la herramienta vierte todas las etiquetas de la imagen pulsada, como etiquetas de la imagen actual que añade el usuario; el número de etiquetas por imagen parecida habitualmente es igual o mayor a cuatro, por lo observado, y explicaría la predominancia de esta ayuda en las etiquetas del usuario. Queda la duda de si los usuarios han considerado todas las etiquetas incluidas relevantes, o simplemente han incluido todas despreocupándose de su revisión, caso en que la ayuda correría el riesgo de ser perjudicial al sistema por introducir ruido en el mismo.

El resto de ayudas ronda sobre las dos etiquetas por tarea, lo cual es una proporción suficiente para indicar que las sugerencias resultan de utilidad, si bien claramente inferior a la anterior ayuda. Por otra parte, se observa que la inclusión progresiva de nuevos tipos de ayuda no varía las proporciones de utilización del resto.

Por último, debe considerarse que en todas estas tareas, casi todas las sugerencias se muestran al usuario de forma automática, sin que él tenga que solicitarlas. Esto será ligeramente diferente para las tareas de búsquedas, y no se cumple para la ayuda de traducción automática, que necesita que se le indique el idioma actual para ejecutar las traducciones. Sabiendo esto, vemos analizando los datos que esta ayuda fue invocada una media de 0,57 veces en la tarea 3 (esto es, aproximadamente una vez por cada dos ejecuciones de la tarea), y 0,62 veces en la tarea 4 (recordemos que en las dos primeras no está presente): esto nos indica que, si bien no todos los usuarios hacen uso de la misma, sí se tiene en consideración y utiliza.

5.2. Tareas de búsqueda (de 5 a 7)

Las siguientes tareas presentadas al usuario tratan de ver qué uso hace de las herramientas de recomendación de etiquetas al ejecutar búsquedas sobre una folksonomía. Se distinguen en el enfoque: la tarea 5 le invita a navegar por la colección de imágenes del sistema, y a elegir cinco fotos que considere relacionadas con otra que se le muestra al inicio. Las tareas 6 y 7 piden buscar

una imagen concreta, siendo la de la tarea 6 una de las añadidas por usuarios anteriores, y la de la 7 una de las contenidas en la colección del sistema.

Las herramientas de ayuda se distribuyen en estas pruebas de forma diferente a las de añadido de imágenes, al adaptarse a la interfaz de búsquedas. Para conocer exactamente la disposición de cada ayuda puede consultarse el apartado “3.8.3 Tareas de búsqueda de imágenes”, donde se explica la interfaz mostrada en estas pruebas.

La diferencia principal de esta nueva disposición con respecto a las tareas de añadido de imágenes consiste en que, exceptuando la ayuda de autocompletado, el resto no se muestran de forma automática al usuario, sino que requieren que éste las solicite. Así, la ayuda de imágenes parecidas se muestra como un botón bajo cada imagen resultante de una búsqueda, que al pulsarse muestra las imágenes parecidas a la que tenía asociada. “Traducción automática” y “etiquetas relacionadas” se incluyen en otro botón al lado del cuadro de búsquedas, que el usuario debe pulsar para utilizarlas.

Así, la primera cuestión que surge es ¿accede el usuario a estas ayudas ahora que debe solicitarlas explícitamente?

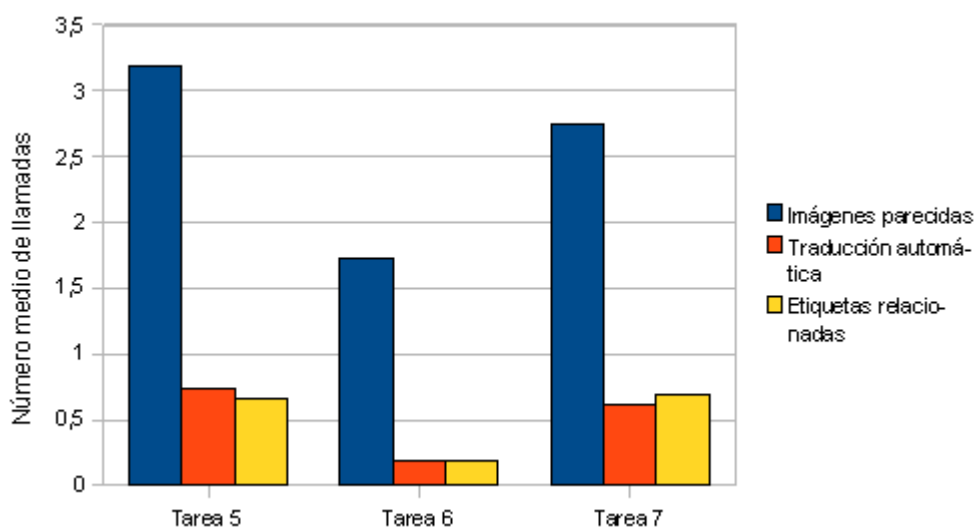


Ilustración 65: Número medio de llamadas a las ayudas en las tareas de búsqueda

Para responder a esta pregunta, en el gráfico anterior puede verse el número de llamadas a las tres ayudas mencionadas por cada prueba. Vemos en él que nuevamente las Imágenes Parecidas tienen clara predominancia, con entre 2 y 3 llamadas de media por prueba, alcanzando su máximo en la tarea 5 de exploración. Esto parece lógico: en estas pruebas el componente visual es básico, y resulta de mucha importancia, encontrada una imagen relevante pero no exactamente la buscada, localizar otras que se le parezcan. “Traducción automática” y “etiquetas relacionadas” van a la par en las tres tareas, algo comprensible habida cuenta de que se acceden a través del mismo botón. Su uso es más limitado, posiblemente por no ser una opción tan natural como las “imágenes parecidas”, y utilizándose en casos más restringidos, como medio para afinar las búsquedas cuando los resultados no son exactamente los deseados. Se observa también que la tarea en la que menos

llamadas se producen en la 6, algo que puede estar relacionado con una mayor facilidad, y necesidad de menos ayudas para completarla.

Otro importante punto a observar es el número de consultas que los usuarios ejecutan para cumplir el objetivo de cada tarea.

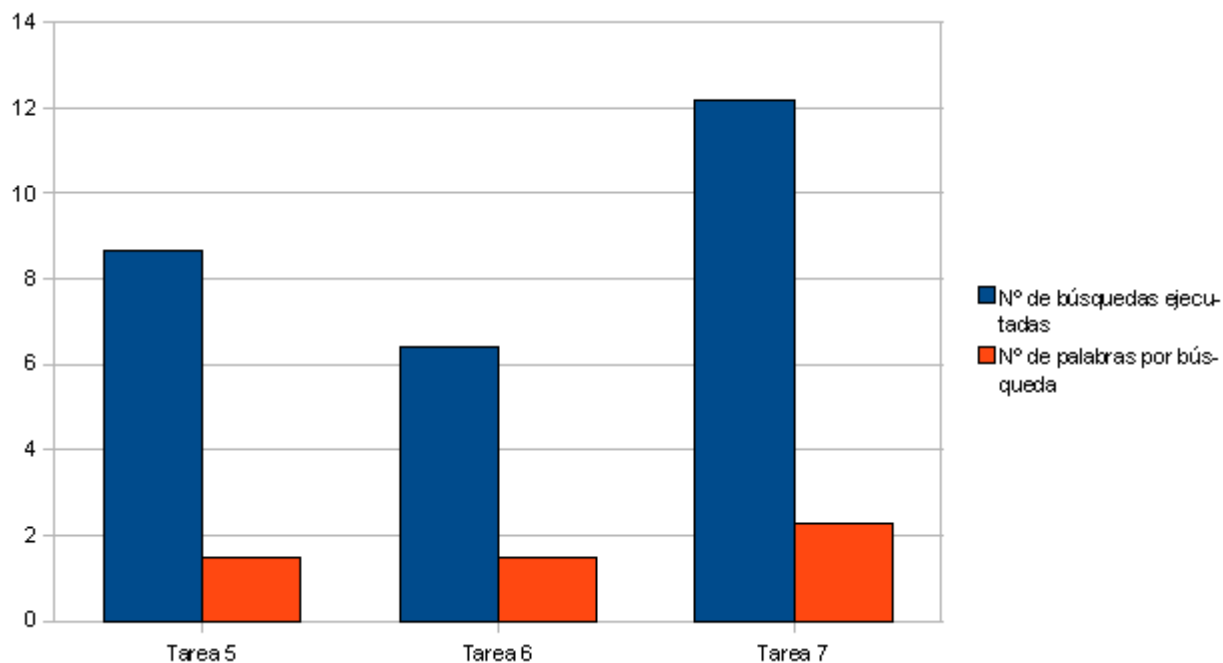


Ilustración 66: N° de búsquedas textuales ejecutadas, y n° de palabras utilizadas en las mismas, en promedio

Podemos ver en el gráfico superior el número medio de búsquedas ejecutadas por los usuarios en cada tarea, y el número medio de palabras utilizadas por búsqueda. Se contabilizan también como dos búsquedas distintas el acceder a diferentes páginas de resultados (p.e., ejecutar búsqueda con texto 'A', ver página 1 de resultados y luego ver página 2, se contarían como dos consultas).

Es un número a primera vista elevado, entre 6 y 12, que nos indica que las tareas no podían completarse con un par de búsquedas básicas, sino que requerían de cierta exploración. El número máximo de búsquedas se encuentra en la última tarea, duplicando el número de la sexta tarea, en la que se sitúa el mínimo. Podemos inferir de ello diferencias significativas en la dificultad de las dos tareas, y habida cuenta de que la única diferencia entre ambas era que la imagen de la tarea sexta estaba etiquetada por otros usuarios y la de la séptima tomada de la colección inicial, podemos deducir que es de vital importancia en la recuperación de una imagen concreta dentro de una folksonomía el compartir un cierto vocabulario o contexto, que permita imaginar qué etiquetas se habrán utilizado para marcar la imagen buscada. Esta diferente dificultad también queda reflejada en el número de abandonos de los usuarios en estas dos últimas tareas, siendo 10 los usuarios que abandonaron la sexta tarea, y 15 usuarios la séptima.

Se indica también el número medio de palabras utilizadas en las consultas, que es cercano a

2 en todas las tareas. Vemos con ello que los usuarios no tienden a utilizar búsquedas complejas, compuestas por muchas palabras, sino que prefieren ejecutar varias búsquedas con un número más reducido de palabras.

Visto el número y longitud de las consultas utilizadas por los usuarios, queda preguntarse qué número de estas palabras en las búsquedas eran motivadas por las herramientas de ayuda.

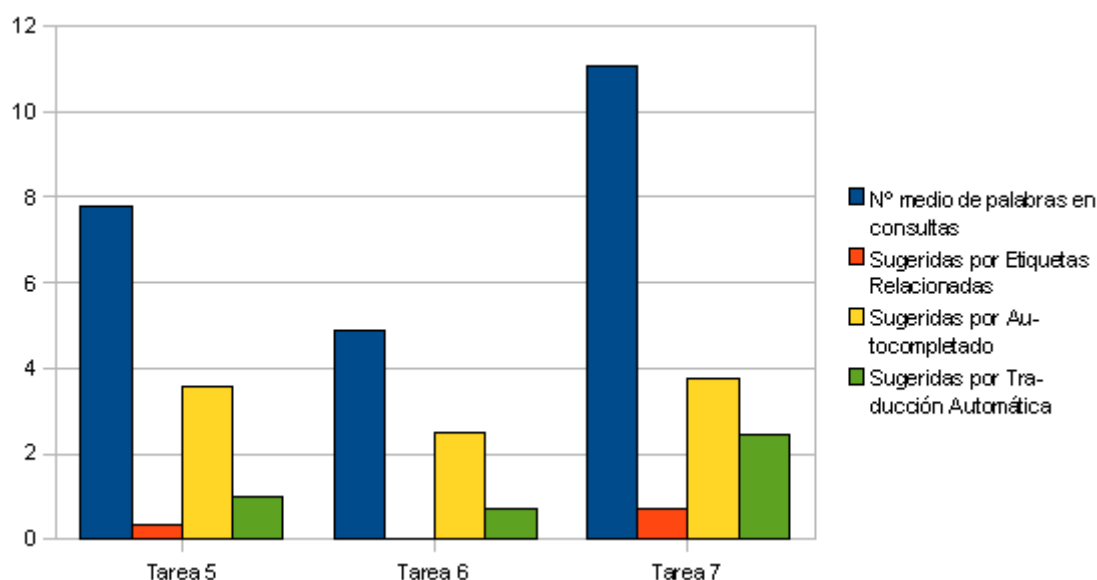


Ilustración 67: Origen de las palabras utilizadas en búsquedas, en promedio

En la gráfica anterior trata de responderse esta cuestión. En ella, se muestra el número medio de palabras distintas utilizadas en el conjunto de las búsquedas de un usuario para una tarea (p.e., si ha ejecutado una búsqueda con texto 'A', y otra búsqueda con texto 'B', se entenderá que ha utilizado dos palabras, 'A' y 'B'), y cuántas de estas palabras fueron sugeridas por las herramientas. No se muestra la ayuda de imágenes parecidas pues no sugería directamente palabras a incluir en las búsquedas.

Vemos que el número de elementos sugeridos escogidos no es tan alto como lo era en las tareas de añadido de imágenes, algo que puede explicarse por el menor número de accesos a las ayudas al estar más ocultas que en las tareas de añadido, como hemos mostrado antes. La ayuda más importante en este caso es la de autocompletado, que resulta de utilidad en caso de que la búsqueda a ejecutar sea una palabra común en el sistema; sin embargo, como se ve esto ocurre sólo en un número reducido de casos. Tras ella, se suelen utilizar las traducciones automáticas, aunque en menor medida, alcanzando máximo en la séptima tarea, como hemos dicho de mayor dificultad. Las etiquetas relacionadas no son apenas utilizadas por los usuarios.

Y por último, podemos preguntarnos si estas ayudas son verdaderamente de utilidad, entendiendo como tal si acercan de algún modo a los usuarios a imágenes que consideren importantes o relevantes.

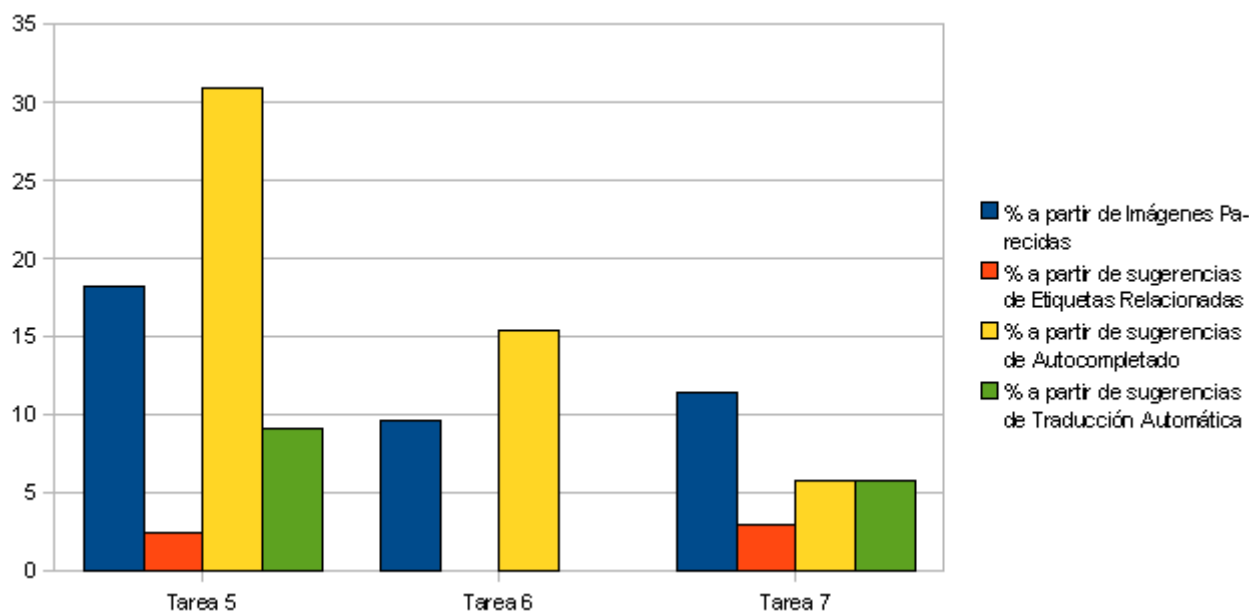


Ilustración 68: Porcentaje de imágenes seleccionadas a partir de resultados obtenidos con cada ayuda

En la gráfica anterior se mide, para las búsquedas que obtuvieron resultados posteriormente seleccionados por los usuarios (botón 'Es imagen buscada' dentro de la interfaz), qué ayudas contribuyeron a su formación, en porcentaje (esto es, si el usuario ejecuta dos búsquedas, seleccionando una imagen de los resultados de cada una, y habiéndose incluido sugerencias de la ayuda 'A' sólo en la primera búsqueda, el porcentaje para esta ayuda será del 50%).

En este caso, vemos que la herramienta de autocompletado lleva a los usuarios a imágenes relevantes con predominancia en la tarea 5, que disminuye en el resto. Esto es comprensible: en la tarea 5 los usuarios son libres de elegir las imágenes que prefieran, mientras que en las siguientes dos tareas están buscando una en concreto. Del resto de ayudas ofrecidas, parece resultar de cierta utilidad la ayuda de imágenes parecidas, con un resultado similar en las tres tareas. Traducciones y etiquetas relacionadas tienen una relevancia mínima, particularmente esta última, que tiende a cero.

5.3. Comentarios de los usuarios

Algunos de los usuarios dejaron comentarios acerca de la aplicación, bien mediante el formulario al fin de la misma establecido al efecto, o hablándolo personalmente. Estos comentarios solían enfocarse a unos pocos aspectos en los que habían reparado de forma especial, sea por suponerles una dificultad o por haberles gustado particularmente.

En general, a los usuarios les gustó la aplicación, aunque sólo fuera por lo que tenía de colorido, la posibilidad de añadir imágenes propias y buscar las de otros, que era algo divertido. Se quejaron un poco del tiempo necesario para completar todas las tareas, algo que explica el número decreciente de usuarios según se avanzaba en el número de tareas completadas. Pudo comprobarse también que los textos de ayuda dispersos por la aplicación sólo eran atendidos por determinados usuarios, quejándose otros por no haber podido localizar o entender algo que aparecía explicado en

esos textos; esto es comprensible, sabiendo que en las interfaces web la información textual tiende a ser leída rápidamente y de forma salteada.

Sobre la usabilidad, aparecieron opiniones contrapuestas. Muchos coincidían en indicar que la aplicación era en líneas generales intuitiva, de interfaz sencilla y agradable de utilizar, pero mencionaban también algunos aspectos que hubiesen preferido más claros, si bien no coincidían en cuáles eran estos. Algunos decían que no habían entendido completamente alguna de las pruebas, otros que los botones del margen superior debían resaltarse mucho más para que los usuarios reparasen en ellos, otros tardaban en entender el mecanismo de la tabla para el añadido de etiquetas, otros esperaban más funcionalidad en el buscador, etcétera. Un aspecto que sí sugirió cierto acuerdo fue en el uso del botón Anterior del navegador, que en esta aplicación no se maneja, y en las tareas de búsquedas llevó a confusión a algún usuario al intentar utilizarlo.

Un usuario señaló que no estaba de acuerdo con el modo en que las imágenes parecidas eran obtenidas, pues más que la comparación superficial de color y textura sería preferible un análisis de su contenido para el reconocimiento de objetos. En esto estamos de acuerdo, pero sobrepasaba los límites del proyecto.

Hay también acuerdo en resaltar la dificultad de las dos últimas pruebas, de búsqueda de una imagen en concreto, detalle que ya había sido descubierto en nuestras pruebas previas a la publicación de la aplicación. Si bien era esperable y comprensible, los usuarios se sentían a veces frustrados por ello: algunos pusieron verdadero empeño en encontrar la imagen y se negaban a abandonar la prueba, y sólo cuando probaban todas las ayudas y combinaciones de búsquedas posibles sin obtener resultado desistían.

6. Conclusiones y líneas futuras

Una vez finalizado el proyecto, explicaremos en este apartado las conclusiones y lecciones aprendidas a lo largo del mismo, que nos puedan servir de consejo y guía en proyectos futuros, y propondremos una serie de líneas de investigación para proyectos futuros nacidos a raíz de éste.

6.1. Resultados del experimento

En general, el análisis de los resultados del experimento ejecutado demuestra que los usuarios están dispuestos a utilizar las ayudas que les ofrece el sistema. En las pruebas ejecutadas por los usuarios, estos consideraron e incluyeron en sus etiquetados y búsquedas sugerencias aportadas por las herramientas de recomendación, de forma consistente en todas las tareas propuestas, como se ve en el más detallado análisis presentado en la sección “5.Resultados del experimento”. La herramienta con más aceptación fue la de imágenes parecidas, posiblemente por el componente gráfico general de todo el experimento, seguida de autocompletado, que ayudaba a adecuar las etiquetas utilizadas por los usuarios a las ya presentes en el sistema, facilitando el acuerdo entre usuarios.

Cabe sin embargo hacer una observación sobre su uso: las herramientas de recomendación parecen sólo ser utilizadas de forma mayoritaria cuando se les presentan a los usuarios de forma claramente visible; al hacer que tengan que pulsar un botón para que se muestren (recuérdense las ayudas de traducciones y etiquetas relacionadas en las tareas de búsquedas) el número de usos disminuía. Esto puede resultar natural estando dentro del contexto de Internet, donde un click de más ya supone la pérdida de cierto número de usuarios, pero es interesante a considerar.

Reflexionando acerca del concepto de herramientas de recomendación más allá de las presentadas en nuestro experimento, cabe preguntarse si la presencia demasiado en primer plano de una ayuda puede facilitar la introducción de ruido en el sistema; esto es así particularmente en herramientas que requieran de los usuarios un cierto grado de atención, sea para distinguir las sugerencias que resultan adecuadas de las que no, para evitar que usuarios despreocupados apliquen sugerencias inadecuadas en sus etiquetados. Puede establecerse un grado de visibilidad recomendado para cada ayuda a este respecto, dejando otras más para el contexto de “usuarios expertos”, que en sistemas reales con extensión en el tiempo, conocieran los atajos y ayudas del sistema más adecuados a la ejecución de sus tareas, aunque éstas requiriesen cierto aprendizaje. En futuros estudios, deberían estudiarse en más detalle estas implicaciones.

6.2. Desarrollo de la aplicación

Una de las partes más interesantes del proyecto fue el diseño y programación de la aplicación del experimento. Esto fue así principalmente por la variedad de módulos y funcionalidades necesarias para la misma, que debían entremezclarse en el manejo del usuario de forma coherente y consistente. Esto supuso un gran reto y una gran fuente de aprendizaje,

permitiendo familiarizarme con tecnologías y módulos de rica funcionalidad, con muchas posibilidades de aplicación en otros proyectos. Entre ellos, me resulta grato haber manejado y conocido los entresijos del componente para la detección de imágenes parecidas GIFT, la librería para búsquedas Lucene, o el manejador para el envío de ficheros por HTTP FileUpload.

Una decisión particularmente relevante en su desarrollo fue la utilización del *framework* GWT para el desarrollo de código Ajax mediante Java, para la programación de la interfaz. Esta decisión fue motivada, al sopesar las necesidades del experimento, por la necesidad de la creación de una interfaz fluida y dinámica, que permitiera al usuario manejar con soltura los diversos elementos que habrían de mostrarse, sin necesidad de esperar a recargas de página o adaptarse a estructuras en exceso estáticas. Y además de esto, me permitió explorar un nuevo modo de desarrollo de interfaces web que creo ha resultado muy enriquecedor: su implementación se ha facilitado con mucho con respecto al esfuerzo que hubiera sido necesario de haber enfrentado la programación en Ajax directamente, pasando a un paradigma similar al de la programación con Java Swing, obteniendo una presentación muy agradable para el usuario y una mejor estructuración de código para el programador.

6.3. Ejecución del proyecto

Uno de los mayores problemas enfrentados durante el desarrollo del proyecto fue debido a mi decisión de compaginar su ejecución con un trabajo a jornada completa. Aunque puedo imaginar que esta dificultad ha sido enfrentada por numerosos otros alumnos, y al tomar esta decisión ya asumía que supondría cierto retraso en su ejecución, el resultado fue que hizo alargarse el tiempo de ejecución mucho más de lo inicialmente planeado, alrededor de siete meses más.

Aunque el balance final es positivo, pues el proyecto ha podido llevarse a término adecuadamente, obtengo como conclusión personal, quizá no extrapolable a otros proyectos y personas, que no es conveniente el compaginar varios proyectos de envergadura a un mismo tiempo, pues la tendencia será que ambos se resientan. Es preferible, en la medida de lo posible, ejecutar cada proyecto de importancia separadamente, centrando todos los esfuerzos en él hasta concluirlo, aunque sea tan sólo por concentrar la atención y potencias en su desarrollo, que de otro modo se ven dispersas y más dadas al agotamiento.

6.4. Líneas futuras

Terminada la ejecución de este proyecto, consideramos que otros pueden ser planteados buscando una profundización en sus líneas de investigación. Resaltaremos en este apartado algunas posibles propuestas, que creemos de especial utilidad:

- **Ampliación del ámbito del experimento:** Este proyecto se ha limitado a un ámbito relativamente reducido para la ejecución de sus pruebas, tanto en el aspecto del número de herramientas de recomendación observadas como el número de usuarios y duración en el tiempo de la evaluación. Para la obtención de unos resultados mucho más detallados y con

ello más precisos, sería conveniente la ampliación del experimento, ampliando la aplicación para permitir la obtención de información mucho más detallada sobre patrones de comportamiento, y su ejecución por muchos más usuarios.

- **Aplicación de las ayudas de sugerencia de etiquetas a sistemas reales:** Para realmente poder evaluar la utilidad y usabilidad de las herramientas de recomendación de etiquetas, sería deseable su inclusión en un sistema real con un número suficiente de usuarios. De este modo, un número amplio de usuarios, siempre superior al que pudiera conseguirse mediante un prototipo para evaluación como el nuestro, harían un verdadero uso del sistema utilizándolo para sus fines personales, y utilizarían estas nuevas herramientas de recomendación preocupándose exclusivamente de la utilidad que les pudieran reportar. De este modo, los defectos y problemas que pudieran presentar serían rápidamente detectados, y si demostrasen ser de utilidad para tan amplio conjunto de usuarios, demostrarían una utilidad real.
- **Evaluación de otros tipos de herramientas de recomendación:** En nuestro experimento nos hemos limitado al desarrollo y estudio de cuatro herramientas de recomendación de etiquetas. Sería interesante la ampliación del estudio con otras posibles herramientas, relacionadas con las presentadas aquí o no. Ejemplos serían sistemas de análisis de los contenidos etiquetados, dependiendo de su tipo: herramientas de minería de textos en contenidos textuales (entradas de blog, artículos, relatos...), o de reconocimiento de imágenes en clasificaciones de imágenes, que traten de identificar la temática y elementos de la imagen incluida sin intervención humana. Y sería más interesante aún el planteamiento de tipos novedosos de herramientas de recomendación, buscando otros modos de mejorar el etiquetado y los sistemas folksonómicos.
- **Etiquetado automático de recursos:** Una de las ventajas derivadas de las herramientas de recomendación de etiquetas es que, si éstas aportan resultados lo suficientemente relevantes, pueden utilizarse sus sugerencias para el etiquetado de recursos sin necesidad de intervención humana, algo muy útil para, por ejemplo, la clasificación automática de grandes conjuntos de contenidos. Sería interesante el estudio de qué herramientas y bajo qué condiciones permitirían semejante uso, y con qué fiabilidad.

7. Bibliografía

- [1] Directorio de Yahoo España (página web, accedida en Julio de 2008): <http://es.dir.yahoo.com>
- [2] Jewelry Television (página web, accedida en Julio de 2008): <http://www.jewelrytelevision.com>
- [3] Jäschke, Robert; Marinho, Leandro; Hotho, Andreas; Schmidt-Thieme, Lars; Stumme, Gerd (2007): *Tag Recommendations in Folksonomies*. 11th European Conference on Principles and Practice of Knowledge Discovery in Databases, Warsaw, Poland.
- [4] Flickr (página web, accedida en Julio de 2008): <http://www.flickr.com/>
- [5] ESP Game (página web, accedida en Julio de 2008): <http://www.gwap.com/gwap/gamesPreview/espgame/>
- [6] Google Image Labeler (página web, accedida en Julio de 2008): <http://images.google.com/imagelabeler/>
- [7] Marlow, Cameron; Naaman, Mor; Boyd, Danah; Davis, Marc (2006): *Position Paper, Tagging, Taxonomy, Flickr, Article, ToRead*. HYPERTEXT '06: Proceedings of the seventeenth conference on Hypertext and hypermedia, Nueva York, EEUU.
- [8] Voß, Jakob (2007): *Tagging, Folksonomy & Co: Renaissance of Manual Indexing?*. 10th international Symposium for Information Science, Colonia, Alemania.
- [9] Vander Val, Thomas (2005): *Explaining and Showing Broad and Narrow Folksonomies*. http://personalinfocloud.com/2005/02/explaining_and_.html
- [10] Kroski, Ellyssa (2005): *The Hive Mind: Folksonomies and User-Based Tagging*. <http://infotangle.blogspot.com/2005/12/07/the-hive-mind-folksonomies-and-user-based-tagging/>
- [11] Hassan-Montero, Yusef; Herrero-Solana, Víctor (2006): *Improving Tag-Clouds as Visual Information Retrieval Interfaces*. I International Conference on Multidisciplinary Information Sciences and Technologies.
- [12] Hassan-Montero, Yusef (2006): *Visualización y Recuperación de Información*. II Encontro de Ciências e Tecnologias da Documentação e Informação.
- [13] Tag Graph (página web, accedida en Julio de 2008): <http://taggraph.com/>
- [14] Flickr Related Web Browsing (página web, accedida en Julio de 2008): http://www.airtightinteractive.com/projects/related_tag_browser/app/
- [15] Adamic, L. A.: Zipf, Power-laws, and Pareto - a ranking tutorial . 2002.
- [16] Kipp, Margaret E.I.; Campbell, D. Grant (2006): *Patterns and Inconsistencies in Collaborative Tagging Systems: An Examination of Tagging Practices*. Proceedings American Society for Information Science and Technology, Austin, Texas.
- [17] Sigurbjörnsson, Börkur; van Zwol, Roelof (2008): *Flickr Tag Recommendation based on Collective Knowledge* . WWW2008, Beijing, China.
- [18] Lux, M.; Granitzer, M.; Kern, R. (2007): *Aspects of Broad Folksonomies*. Database and Expert Systems Applications, 2007. DEXA '07. 18th International Conference on.
- [19] Golder, S.; Huberman, B.A. (2006): *The structure of collaborative tagging systems*. Journal of the American Society for Information Science and Technology.
- [20] Cattuto, C.; Schmitz, C.; Baldassarri, A.; Servedio, V.D. P.; Loreto, V.; Hotho, A.; Grahl, M.; Stumme, G (2007): *Network properties of folksonomies*. AI Commu-nications Special Issue on "Network Analysis in NaturalSciences and Engineering".
- [21] Hotho, A.; Jäschke, R.; Schmitz, C.; Stumme, G. (2006): *Trend Detection in Folksonomies*. Prof. 1st International Conference on SAMT, Atenas, Grecia.

- [22] Guy, Marieke; Tonkin, Emma (2006): *Folksonomies: Tidying up Tags?*. <http://www.dlib.org/dlib/january06/guy/01guy.html>
- [23] Bar-Ilan, J.; Shoham, S.; Idan, A.; Miller, Y.; Shachak, A. (2006): *Structured vs. unstructured tagging – A case study*. Conferencia WWW2006, Edimburgo, Escocia.
- [24] Simpson, Edwin (2008): *Clustering Tags in Enterprise and Web Folksonomies*. HP Labs Technical Reports.
- [25] Begelman, G.; Keller, P.; Smadja, F. (2006): *Automated TagClustering: Improving search and exploration in the tag space*. Proceedings of the WWW2006.
- [26] Au Yeung, C. M.; Gibbins, N.; Shadbolt, N. (2007): *Tag Meaning Disambiguation through Analysis of Tripartite Structure of Folksonomies*. The 2007 IEEE / WIC / ACM International Conference on Intelligence Agent Technology - Workshops, 2-5 November, Silicon Valley, California, USA.
- [27] Christiaens, S. (2006): *Metadata Mechanisms: From Ontology to Folksonomy ... and Back*. On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops.
- [28] Schmitz, P. (2006): *Inducing ontology from Flickr tags*. Collaborative Web Tagging Workshop at WWW2006.
- [29] Mishne, Gilad (2006): *AutoTag: A Collaborative Approach to Automated Tag Assignment for Weblog Posts*. Conferencia WWW2006, Edimburgo, Escocia.
- [30] Hotho, A.; Jaschke, R.; Schmitz, C.; Stumme, G. (2006): *FolkRank: A ranking algorithm for folksonomies*. FGIR 2006.
- [31] Aurnhammer, M.; Hanappe, P.; Steels, L. (2006): *Integrating Collaborative Tagging and Emergent Semantics for Image Retrieval*. Conferencia WWW2006, Edimburgo, Escocia.
- [32] CLEF: Cross-Language Evaluation Forum (página web, accedida en Julio de 2008): <http://www.clef-campaign.org/>
- [33] ICLEF (página web, accedida en Agosto de 2008): <http://nlp.uned.es/iCLEF/>
- [34] Artiles, J.; Gonzalo, J.; López-Ostenero, F.; Peinado, V. (2006): *Are Users Willing to Search Cross-Language? An Experiment with the Flickr Image Sharing Repository*. Conferencia WWW2006, Edimburgo, Escocia.
- [35] Clough, P.; Al-Maskari, A.; Darwish, K. (2006): *Providing Multilingual Access to FLICKR for Arabic Users*. Conferencia WWW2006, Edimburgo, Escocia.
- [36] Karlgren, J.; Olsson, F. (2006): *Trusting the results in crosslingual keyword-based image retrieval*. Conferencia WWW2006, Edimburgo, Escocia.
- [37] Google Suggest (página web, accedida en Agosto de 2008): <http://www.google.com/webhp?complete=1&hl=en>
- [38] XMLSpy (página web, accedida en Septiembre de 2008): http://www.altova.com/products/xmlspy/xml_editor.html
- [39] Arte y Fotografía (página web, accedida en Julio de 2008): <http://www.arteyfotografia.com.ar/>
- [40] API de Flickr (página web, accedida en Julio de 2008): <http://www.flickr.com/services/api/>
- [41] FlickrJ (página web, accedida en Julio de 2008): <http://sourceforge.net/projects/flickrj/>
- [42] Struts (página web, accedida en Julio de 2008): <http://struts.apache.org/>
- [43] DWR (página web, accedida en Julio de 2008): <http://directwebremoting.org/>
- [44] AjaxTags (página web, accedida en Julio de 2008): <http://ajaxtags.sourceforge.net/>
- [45] Google Web Toolkit (página web, accedida en Julio de 2008): <http://code.google.com/webtoolkit/>
- [46] MyGWT (página web, accedida en Julio de 2008):

<http://sourceforge.net/projects/mygwtlgpl/>

[47] GWT-Ext (página web, accedida en Julio de 2008): <http://gwt-ext.com/>

[48] Ext GWT (página web, accedida en Julio de 2008): <http://extjs.com/products/gxt/>

[49] FileUpload (página web, accedida en Septiembre de 2008):

<http://commons.apache.org/fileupload/>

[50] Proyecto Apache Lucene (página web, accedida en Julio de 2008):

<http://lucene.apache.org>

[51] Alejandro Blasco Plaza (2006): *Proyecto Fin de Carrera: ASISTE*. Universidad Carlos III

[52] Google Images (página web, accedida en Julio de 2008): <http://images.google.com/>

[53] ImgSeek (página web, accedida en Julio de 2008): <http://www.imgseek.net/>

[54] FIRE: Flexible Image Retrieval Engine (página web, accedida en Julio de 2008):

<http://thomas.deselaers.de/FIRE>

[55] GIFT: GNU Image Finding Tool (página web, accedida en Julio de 2008):

<http://www.gnu.org/software/gift/>

[56] Multimedia Retrieval Markup Language (página web, accedida en Julio de 2008):

<http://www.mrml.net/>

[57] Log4J (página web, accedida en Septiembre de 2008): <http://logging.apache.org/log4j/>

[58] MySQL (página web, accedida en Septiembre de 2008): <http://www.mysql.com/>