



UNIVERSIDAD DE ZARAGOZA
CENTRO POLITÉCNICO SUPERIOR



Ingeniería Informática

Proyecto Final de Carrera

Publindex: Análisis de la calidad de publicaciones de investigación

Autor: Sara Subijana Gracia

Director: Eduardo Mena Nieto

*Departamento de Informática e Ingeniería de
Sistemas*

Septiembre de 2010

*Dedicado a mi abuela Elisa,
gracias por mandarme tus fuerzas allí donde estés.
Siempre te llevaré conmigo.*

Agradecimientos

Quiero agradecer todo su esfuerzo y dedicación a mi director, Eduardo, que siempre ha estado disponible para responder a mis dudas y que me ha hecho aprender cómo hacer las cosas mejor. Gracias por ayudarme a que esta memoria no parezca una novela del Señor de los Anillos y por invitarme a tantas consumiciones en tu oficina particular.

Agradecer su ayuda a todo el grupo de investigación Sistemas Informáticos Distribuidos (SID), que hicieron una gran labor de pruebas desde las primeras versiones de la aplicación ayudándome así a mejorarla.

Mil Gracias a ti, Rafa, por darme todo tu apoyo y confianza en los momentos de desesperación y flaqueza. Sin tus ánimos, tu cariño y paciencia no hubiera afrontado las cosas de la misma manera que lo he hecho. Te quiero.

Gracias a mi Comando, es especial a mis chicas, los momentos de desconexión, cervezas y risas con vosotros que me daban fuerzas renovadas para afrontar los días siguientes mucho más relajada. Espero que no nos separemos nunca, por muy lejos que estemos las unas de las otras.

Quiero agradecer a mis chicos Deloitte su compañía y amistad durante el periodo que estuve trabajando con ellos. El conocerlos fue una de las mejores experiencias que gané mientras estuvimos en Walqatraz.

Gracias a mi familia, en especial a mis padres y mi abuelo, que siempre han creído en mí y se han interesado por este proyecto, aunque no entendieran qué estaba haciendo. Espero que os sintáis muy orgullosos de vuestra ingeniera.

Anexo A: Especificación de requisitos

Publindex: Análisis de la calidad de
publicaciones de investigación

Tabla de contenido

A.1 INTRODUCCIÓN.....	47
A.1.1 PROPÓSITO Y ALCANCE	47
A.1.2 ACRÓNIMOS.....	47
A.1.3 RESUMEN	47
A.2 DESCRIPCIÓN FUNCIONAL DEL SISTEMA.....	49
A.2.1 FUNCIONES PRINCIPALES DEL SISTEMA.....	49
A.2.2 USUARIOS DE LA APLICACIÓN.....	50
A.3 REQUISITOS DEL SISTEMA.....	51
A.3.1 REQUISITOS FUNCIONALES	51
A.3.2 REQUISITOS NO FUNCIONALES	55
A.4 REFERENCIAS	57
A.4.1 REFERENCIAS WEB.....	57
A.4.2 REFERENCIAS BIBLIOGRÁFICAS	57

A.1 Introducción

Este apartado pretende informar del propósito y alcance del documento, con objeto de mostrar la finalidad de su elaboración dentro del proceso de desarrollo de Publindex.

A.1.1 Propósito y alcance

El propósito de este documento consiste en desglosar y describir de manera clara y detallada todo el proceso de captura de requisitos. Los requisitos han sido divididos en funcionales y no funcionales. Los requisitos funcionales permiten dar una visión de todas las funcionalidades que presenta el sistema, mientras que los requisitos no funcionales especifican ciertas exigencias a nivel de rendimiento, nivel tecnológico, entorno de desarrollo, de consistencia, etc.

El alcance de este documento comprende todo el ciclo de vida del proyecto, ya que en cualquier momento la funcionalidad puede ampliarse añadiendo nuevos requerimientos al sistema, a excepción de en la fase de finalización.

A.1.2 Acrónimos

Los acrónimos utilizados a lo largo del documento son:

- SID: Sistemas de Información Distribuidos.
- JCR: Journal Citation Reports
- CORE: COmputing Research & Education
- SJR: Scientific Journal Rankings
- ISSN: International Standard Serial Number

A.1.3 Resumen

El documento se compone de tres partes claramente diferenciadas. En un primer lugar, se realiza una breve introducción al objetivo y alcance del documento; en segundo lugar, se describe el sistema de manera funcional y finalmente se enumeran los requisitos tanto funcionales como no funcionales.

A.2 Descripción funcional del sistema

Este apartado presenta funcionalmente el sistema, además de identificar los distintos usuarios del sistema. Este es el primer paso antes de identificar los casos de uso con las funcionalidades principales y poder recopilar así los requisitos funcionales.

A.2.1 Funciones principales del sistema

El usuario podrá realizar analizar la calidad de tres maneras diferentes:

- Analizar una publicación, bien sean revistas o conferencias.
- Analizar un listado de publicaciones a las que hacen referencia una serie de citas bibliográficas.
- Analizar la calidad de un autor a través de las publicaciones en las que ha publicado sus trabajos, referenciados a través de citas bibliográficas.

Estas tres maneras de analizar la calidad están ligadas entre sí, ya que el análisis de un listado de publicaciones necesita analizar individualmente cada una y el análisis del trabajo de autor requiere analizar las publicaciones a las que hacen referencia sus citas.

Estos análisis de la calidad darán como resultado las valoraciones que dichos eventos han registrado según las fuentes de calidad JCR, SJR y CORE. La valoración asignada por cada índice es única para un año en concreto, algo que resulta un baremo poco preciso de la calidad. Por ejemplo, decir que una publicación tiene valoración 'A' cuando el resto de publicaciones de una categoría tiene valoración 'C' es bueno, pero si el resto tuvieran valoración 'A+' sería negativo.

Como consecuencia de esta problemática, cada publicación tendrá una valoración en cada categoría a través de un porcentaje. Dicho porcentaje se calcula en función de la posición de la publicación en la categoría respecto del total de publicaciones de ésta.

El sistema también puede ser utilizado como fuente de información. Se permite consultar información propia de los journals y conferencias, como el lenguaje en el que está escrito, el país o ISSN. Además, también puede consultarse información relativa a las categorías de un índice de calidad, visualizando las publicaciones que componen una categoría y su posición en ella para un año en concreto.

El análisis de un listado de publicaciones se realiza a través de un conjunto de citas bibliográficas. Este sistema permite al usuario dado un conjunto de citas, con un cierto formato texto o formato *BibTex*, extraer la información de cada evento o revista a los que hacen referencia y así poder consultar la calidad de cada publicación.

Dado que Publindex va a ser utilizado en un primer momento por un grupo reducido de investigadores pertenecientes al grupo Sistemas de Información Distribuidos (SID), la búsqueda por autor sólo va a realizarse dentro del ámbito de este grupo de investigadores.

La aplicación también proporciona un mecanismo de mantenimiento de la base de datos, para eliminar duplicados de publicaciones y fusionarlas para crear una sola.

A.2.2 Usuarios de la aplicación

El investigador que consultará la calidad de las publicaciones es el único usuario de la aplicación. Para poder acceder a la aplicación web a través de cualquier URL de Publindex es necesario introducir un usuario y contraseña. El acceso a la aplicación está protegido porque a través de este sistema pueden consultarse índices de calidad como JCR que tienen restringido el acceso. Además, el uso de la aplicación debe limitarse a los usuarios de Publindex que tengan un nivel elevado de experiencia dentro del ámbito de la investigación para poder realizar labores de mantenimiento en los datos almacenados.

A.3 Requisitos del sistema

A continuación se muestran los requisitos funcionales y no funcionales del sistema. De cada requisito funcional se presenta, en modo tabla, una descripción, criticidad, coste y planificación, riesgos, implicaciones técnicas, dependencias y comentarios. Cada requisito no funcional sólo presenta una descripción.

A.3.1 Requisitos funcionales

Funcionalidad	Analizar una publicación por nombre/siglas/ISSN
Descripción	<p>El usuario puede realizar búsquedas de la publicación de la que consultar la calidad a través del nombre o siglas de la publicación o bien por ISSN y poder seleccionar las macrocategorías dentro de cada índice donde buscar.</p> <p>La búsqueda a través de nombre/siglas se realiza por <i>substring</i>, de manera que puede haber más de una coincidencia. En ese caso, se mostrará el listado de journals y conferencias coincidentes para que el usuario pueda elegir una de ellas y ver los resultados.</p> <p>La búsqueda por ISSN, sin embargo, sólo lleva a un resultado posible.</p> <p>Tras la búsqueda se redirige a las valoraciones de cada índice de calidad para una publicación, año a año, mostrando también su top dentro de cada categoría.</p>
Criticidad	Alta
Implicaciones técnicas	
Coste y planificación	Alta
Riesgos	Bajo
Dependencias	Para poder realizar las búsquedas de valoraciones, deben haber sido insertados los datos de los índices de calidad y las publicaciones. Además, requieren de una actualización periódica para evitar que las valoraciones se queden obsoletas y perder así fiabilidad.
Comentarios	La selección de las macrocategorías permite una búsqueda más concreta.

Tabla 1: Descripción de la funcionalidad relativa al analizador de una publicación

Funcionalidad	Mostrar la información de una categoría de índice.
Descripción	Deben mostrarse las publicaciones que, junto con sus valoraciones, componen cada categoría por año.
Criticidad	Baja
Implicaciones técnicas	
Coste y planificación	Baja
Riesgos	Baja
Dependencias	Debe estar implementado el Analizador de publicaciones y haber sido almacenada información de las categorías y valoraciones.
Comentarios	Esta funcionalidad complementa el Analizador de publicaciones para dar mucha más información al usuario.

Tabla 2: Descripción de la funcionalidad relativa a las categorías de índice

Funcionalidad	Asignar una valoración independiente del índice.
Descripción	Aportar a cada índice de calidad una presentación independiente del criterio de puntuación seguido, otorgando así al usuario una visión mucho más genérica acerca de la calidad de las publicaciones.
Criticidad	Baja
Implicaciones técnicas	
Coste y planificación	Baja
Riesgos	Baja
Dependencias	Deben existir datos relativos a las categorías y valoraciones para cada índice.
Comentarios	La valoración debe considerar otros aspectos calificativos que no dependan directamente de la puntuación asignada por cada índice.

Tabla 3: Descripción de la funcionalidad de valoraciones independientes del índice

Funcionalidad	Analizar un listado de publicaciones a través de referencias bibliográficas
Descripción	<p>La calidad de un artículo de investigación viene determinada por la calidad de la revista o conferencia donde está publicado.</p> <p>La aplicación también debe ofrecer la posibilidad de realizar análisis de un conjunto de publicaciones, a través de una serie de citas bibliográficas. Esta serie de citas bibliográficas podrán presentarse en un formato texto concreto o formato BibTex para facilitar el reconocimiento de la información.</p> <p>Tras la búsqueda del listado de publicaciones y sus valoraciones, la aplicación debe presentar el listado de publicaciones al que hacen referencia las citas e indicar la valoración del evento en el año de la cita.</p>
Criticidad	Alta
Implicaciones técnicas	Debe construirse algún mecanismo de procesamiento de texto para acceder a la publicación dentro de la cita bibliográfica.
Coste y planificación	Medio
Riesgos	Baja
Dependencias	Interactúa con el Analizador de publicaciones a través de nombre/ISSN, que es el que verdaderamente realiza la búsqueda en el sistema.
Comentarios	Para cada cita, debe seleccionarse la publicación que más se asemeje a la información del título de la revista o conferencia de la referencia que se ha procesado.

Tabla 4: Descripción de la funcionalidad del analizador de un listado de publicaciones por citas bibliográficas

Funcionalidad	Analizar la calidad de un autor.
Descripción	Decidir si el trabajo de una persona es bueno depende de los eventos y revistas donde haya publicado sus trabajos. Publindex también debe ofrecer un servicio de análisis de la calidad de los trabajos de un autor dado su nombre. Esta búsqueda de la calidad se centra en recuperar las referencias bibliográficas relativas al autor y de ahí acceder a las publicaciones a las que referencian dichas citas. La búsqueda de los trabajos de un autor debe realizarse de manera automática y a través de la red, sin acceder a base de datos. Los autores sobre los que realizar la búsqueda serán los integrantes del grupo de investigación SID.
Criticidad	Alta
Implicaciones técnicas	Debe construirse algún mecanismo que acceda a la información de la web y procese la información online.
Coste y planificación	Medio
Riesgos	Baja
Dependencias	Interactúa con el analizador de referencias bibliográficas.
Comentarios	El hecho de acceder de manera automática a la información evita acceder a base de datos y las penalizaciones de tiempo que estos accesos conllevan (conexiones, consultas...)

Tabla 5: Descripción de la funcionalidad del analizador de un listado de publicaciones por autor

Funcionalidad	Ofrecer un servicio de mantenimiento de datos.
Descripción	Existe una problemática relativa a la información que se almacena de cada índice: puede ocurrir que existan dos publicaciones con nombre diferente pero que en realidad sean la misma. En estas situaciones el usuario dispone de un subsistema que permite realizar la fusión de dos publicaciones en una sola, y disponer en un solo evento de toda la información que anteriormente estaba distribuida en dos registros diferentes.
Criticidad	Media
Implicaciones técnicas	Debe diseñarse algún mecanismo que guarde la información de las fusiones realizadas y poder restaurar así el estado anterior en el caso de que deba reiniciarse la base de datos.
Coste y planificación	Bajo
Riesgos	Baja
Dependencias	Deben existir datos relativos a publicaciones y valoraciones de índices.
Comentarios	Este subsistema facilita las labores de búsqueda del investigador que, una vez realizadas las labores de mantenimiento, sabe exactamente a qué publicaciones han sido asignadas las valoraciones.

Tabla 6: Descripción de la funcionalidad del mantenimiento de publicaciones

A.3.2 Requisitos no funcionales

Los requisitos no funcionales del sistema aportan especificaciones de carácter tecnológico, de rendimiento...en general aquellas exigencias que no van a afectar a la funcionalidad definida en el sistema.

Requisito	Descripción
Entorno tecnológico	La aplicación web debe estar implementada en Java versión 6.0 en adelante.
Disponibilidad	El sistema debe ser accesible las 24h del día y ser usado por varios usuarios a la vez sin que se vea afectado el tiempo de respuesta.
Accesibilidad	El sistema tiene limitado el acceso, siendo necesario autenticarse al acceder a cualquier URL relativa a la web de Publindex.

Compatibilidad de navegadores	El sistema debe tener un comportamiento satisfactorio, tanto a nivel funcional como de apariencia, en los navegadores Internet Explorer 7.0 en adelante, Mozilla Firefox y Google Chrome.
Rendimiento	El tiempo de espera del usuario no debe sobrepasar los 15 segundos.
Otros requisitos	El sistema debe presentar información de estadísticas relativas a la página como visitas totales recibidas, media de visitas día, etc.

Tabla 7: Requisitos no funcionales del sistema

A.4 Referencias

A.4.1 Referencias web

En este documento no se han utilizado referencias web.

A.4.2 Referencias bibliográficas

[PU]: I. Jacobson, G. Booch, J. Rumbaugh. 2000. *El proceso Unificado de desarrollo de Software*. Pearson Education.

Anexo B: Caso de estudio del proyecto

Publindex: Análisis de la calidad de
publicaciones de investigación

Tabla de contenido

B.1	INTRODUCCIÓN	63
B.1.1	PROPÓSITO Y ALCANCE	63
B.1.2	REFERENCIAS	63
B.1.3	ACRÓNIMOS	63
B.1.4	RESUMEN.....	64
B.2	ÍNDICES DE CALIDAD	65
B.2.1.1	<i>Journal Citation Reports (JCR)</i>	65
B.2.1.2	<i>Computing Research and Education (CORE)</i>	66
B.2.1.3	<i>Scientific Journal Rankings (SJR)</i>	69
B.2.1.4	<i>Computer Science Conference Ranking (CSCR)</i>	70
B.3	PRIMERA FASE: IDEACIÓN	73
B.3.1	DIAGRAMA DE CONTEXTO	74
B.3.2	MODELOS DE CASOS DE USO	74
B.3.2.1	<i>CU Analizar publicación</i>	74
B.3.2.2	<i>CU Analizar lista de publicaciones</i>	76
B.3.2.3	<i>Analizar calidad de un autor</i>	77
B.3.2.4	<i>CU Mantenimiento de publicaciones</i>	78
B.4	SEGUNDA FASE: ELABORACIÓN (ANÁLISIS)	81
B.4.1	PRIMERA ITERACIÓN	81
B.4.1.1	<i>Modelo de dominio parcial</i>	81
B.4.1.2	<i>Diagramas de secuencia</i>	82
B.4.2	SEGUNDA ITERACIÓN.....	85
B.4.2.1	<i>Diagramas de secuencia</i>	85
B.5	SEGUNDA FASE: ELABORACIÓN (DISEÑO)	89
B.5.1	PATRONES DE DISEÑO Y PRINCIPIOS GRASP	89
B.5.2	DIAGRAMAS DE COMUNICACIÓN	90
B.5.2.1	<i>Analizador de una publicación</i>	90
B.5.2.2	<i>Analizador de una lista de publicaciones</i>	92
B.5.2.3	<i>Analizador de la calidad de un autor</i>	93
B.5.2.4	<i>Mantenimiento de publicaciones</i>	94
B.5.3	DIAGRAMA DE CLASES	96
B.5.4	ESTRUCTURA DE LA SOLUCIÓN	98
B.5.5	DISEÑO DE LA BASE DE DATOS	98

B.5.6	PROTOTIPADO DE VENTANAS	102
B.5.6.1	<i>Menú principal</i>	102
B.5.6.2	<i>Analizador de una publicación</i>	103
B.5.6.3	<i>Analizador de publicaciones a través de citas bibliográficas</i>	105
B.5.6.4	<i>Analizador de la calidad de un autor</i>	106
B.5.6.5	<i>Mantenimiento de publicación</i>	107
B.5.6.6	<i>Mensajes de error</i>	108
B.6	SEGUNDA Y TERCERA FASE: ELABORACIÓN Y CONSTRUCCIÓN (IMPLEMENTACIÓN)	109
B.7	CUARTA FASE: TRANSICIÓN	111
B.8	REFERENCIAS	113
B.8.1	REFERENCIAS WEB	113
B.8.2	REFERENCIAS	113

B.1 Introducción

La finalidad de este apartado es dar una visión general del documento, indicando al igual que en cualquier otro texto de este proyecto su objetivo, alcance, referencias, etc.

B.1.1 Propósito y alcance

Este documento tiene como propósito mostrar de manera clara y concisa todo el proceso de desarrollo del proyecto, con especial énfasis en el análisis y diseño que tuvo que realizarse previamente a la implementación del sistema.

El alcance del caso de estudio comprende todo el proyecto. Como se indica en la memoria del proyecto, la metodología seguida ha sido el Proceso Unificado. Esta metodología está basada en un modelo iterativo e incremental con objeto de ir construyendo la aplicación a la vez que se van refinando o definiendo nuevas funcionalidades. El proceso ha sido dividido en cuatro fases: inicialización o ideación, elaboración, construcción y finalización. La fase de ideación pretende captar la mayor parte de las funcionalidades del sistema; durante la fase de elaboración se aplican las técnicas de análisis y diseño necesarias para ir definiendo de manera concreta cada módulo de trabajo y sus necesidades y la implementación de aquellos aspectos de mayor riesgo, la arquitectura y base de datos; la fase de construcción comprende las labores de implementación y construcción de los aspectos y módulos de menor riesgo; finalmente la fase de finalización comprende las labores de documentación [PU].

B.1.2 Referencias

A lo largo de este documento se hace referencia al anexo de Requisitos y de Implementación, ya que recogen parte del trabajo realizado durante la fase de ideación, elaboración y construcción.

B.1.3 Acrónimos

Se han utilizado los siguientes acrónimos a lo largo del documento:

- CU: Casos de Uso
- PU: Proceso Unificado
- MVC: Model-View-Controller
- DAO: Data Access Object
- JCR: Journal Citation Report
- GRASP: General Responsibility Assignment Software Patterns
- GUI: Graphic User Interface

B.1.4 Resumen

La estructura que presenta el documento contiene una introducción (apartado actual), un estudio de los índices de calidad que fueron considerados como baremo de la calidad y a continuación la descripción de las fases de ideación, elaboración, implementación y finalización del proceso de desarrollo. Finalmente, se muestran las referencias bibliográficas a los textos relativos a todo este proceso que fueron consultados.

B.2 Índices de calidad

Esta sección detalla las principales características de los índices de calidad de publicaciones que serán consideradas a utilizar por su prestigio, versiones actuales y fiabilidad. Las fuentes descritas son JCR, SJR, CORE y CSCR.

B.2.1.1 Journal Citation Reports (JCR)

JCR es un índice de calidad de revistas de divulgación científica y social. Dentro de cada uno de estos grupos clasifica las revistas en categorías según la temática a la que pertenezcan [WebJCR].

La calidad se presenta anualmente a través de un factor de impacto numérico que indica el número de referencias a la revista en los últimos dos años dividido por el número de artículos aceptados en dicho revista. Este factor de impacto es único para una publicación en un año concreto y podría considerarse como una especie de media de citas recientes por artículo aceptado en la revista. Según el factor de impacto la calidad de una revista varía considerablemente.

Debido a que el número de categorías es elevado, Publindex sólo considerará la carga de datos de aquellas categorías científicas pertenecientes al área de informática y que son las siguientes:

- *Computer Science, Artificial Intelligence*
- *Computer Science, Cybernetics*
- *Computer Science, Hardware & Architecture*
- *Computer Science, Information Systems*
- *Computer Science, Interdisciplinary Applications*
- *Computer Science, Software Engineering*
- *Computer Science, Theory & Method*
- *Multidisciplinary Sciences*
- *Robotics*
- *Telecommunications*

Cada categoría presenta las revistas que la componen ordenados por factor de impacto, indicando para cada publicación datos importantes como son su título, ISSN, país o lenguaje. Una revista puede incluirse en más de una categoría dentro de un mismo año. La imagen siguiente muestra un fragmento de la categoría *Computer Science, Artificial Intelligence* del año 2008:

1	Title	ISSN	Impact	Country	Language
2	IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE	0162-8828	5,96	UNITED STAT	ENGLISH
3	INTERNATIONAL JOURNAL OF COMPUTER VISION	0920-5691	5,358	UNITED STAT	ENGLISH
4	IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION	1089-778X	3,736	UNITED STAT	ENGLISH
5	IEEE TRANSACTIONS ON NEURAL NETWORKS	1045-9227	3,726	UNITED STAT	ENGLISH
6	IEEE TRANSACTIONS ON FUZZY SYSTEMS	1063-6706	3,624	UNITED STAT	ENGLISH
7	MEDICAL IMAGE ANALYSIS	1361-8415	3,602	NETHERLANE	ENGLISH
8	ARTIFICIAL INTELLIGENCE	0004-3702	3,397	NETHERLANE	ENGLISH
9	IEEE TRANSACTIONS ON IMAGE PROCESSING	1057-7149	3,315	UNITED STAT	ENGLISH
10	COMPUTATIONAL INTELLIGENCE	0824-7935	3,31	UNITED STAT	ENGLISH
11	PATTERN RECOGNITION	0031-3203	3,279	ENGLAND	ENGLISH
12	JOURNAL OF MACHINE LEARNING RESEARCH	1532-4435	3,116	UNITED STAT	ENGLISH
13	Journal of Web Semantics	1570-8268	3,023	NETHERLANE	ENGLISH
14	EVOLUTIONARY COMPUTATION	1063-6560	3	UNITED STAT	ENGLISH
15	International Journal of Innovative Computing Information and Control	1349-4198	2,791	JAPAN	ENGLISH
16	COMPUTATIONAL LINGUISTICS	0891-2017	2,656	UNITED STAT	ENGLISH
17	NEURAL NETWORKS	0893-6080	2,656	ENGLAND	ENGLISH
18	EXPERT SYSTEMS WITH APPLICATIONS	0957-4174	2,596	UNITED STAT	ENGLISH
19	IEEE Computational Intelligence Magazine	1556-603X	2,535	UNITED STAT	ENGLISH
20	DATA MINING AND KNOWLEDGE DISCOVERY	1384-5810	2,421	NETHERLANE	ENGLISH
21	NEURAL COMPUTATION	0899-7667	2,378	UNITED STAT	ENGLISH
22	IEEE TRANSACTIONS ON SYSTEMS MAN AND CYBERNETICS PART B-CYBERNETICS	1083-4419	2,361	UNITED STAT	ENGLISH
23	MACHINE LEARNING	0885-6125	2,326	UNITED STAT	ENGLISH
24	IEEE INTELLIGENT SYSTEMS	1541-1672	2,278	UNITED STAT	ENGLISH
25	IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING	1041-4347	2,236	UNITED STAT	ENGLISH
26	COMPUTER VISION AND IMAGE UNDERSTANDING	1077-3142	2,22	UNITED STAT	ENGLISH
27	AUTONOMOUS AGENTS AND MULTI-AGENT SYSTEMS	1387-2532	2,125	UNITED STAT	ENGLISH
28	Information Fusion	1566-2535	2,057	NETHERLANE	ENGLISH
29	ARTIFICIAL INTELLIGENCE IN MEDICINE	0933-3657	1,96	NETHERLANE	ENGLISH

Figura 1: Ejemplo de categoría JCR

JCR presenta anualmente nuevos índices de calidad pudiendo variar los factores de impacto y las categorías a las que pertenece una publicación. Debido a que cargar datos con cierta antigüedad no es fiable, el sistema sólo considerará la carga de las valoraciones a partir del año 2002.

El proceso de consulta de JCR no está exento de pasos a realizar. Los pasos a dar por un usuario de JCR son los siguientes:

1. Acceder a la página web de *ISI Web of Knowledge*.
2. Introducir usuario y contraseña para acceder a la base de datos.
3. Seleccionar la base de datos de JCR.
4. Elegir el año en el cual analizar la revista y el nombre concreto.
5. Del listado de resultados, elegir la revista deseada (en esta parte ya puede consultarse el factor de impacto).
6. Para saber su clasificación en una categoría, acceder al enlace '*Journal rank in categories*'.

Esta manera de consultar la calidad de una revista implica tener conocimientos de en qué años exactos ha sido valorada. Además, la información de la posición que ocupa en cada categoría sólo está disponible para el año consultado resultando muy pesado consultar varios años en una categoría teniéndose que hacer estos cinco pasos para cada año.

B.2.1.2 Computing Research and Education (CORE)

Este índice da valoraciones de la calidad de revistas y conferencias que han sido referenciadas por investigadores australianos [WebCORE]. Desde el año 2008, las valoraciones CORE son recogidas de los rankings que se corresponden con el área de la

informática facilitados por el gobierno Australiano. No obstante, existen otras áreas de especial interés que serán tenidas en cuenta con objeto de poder realizar búsquedas de publicaciones pertenecientes a otra temática, como por ejemplo ingeniería o tecnología.

CORE asigna anualmente valoraciones únicas para cada publicación basadas en letras con el siguiente orden decreciente: A*/A+, A, B, C, L, U. La valoración A+ era asignada a aquellas publicaciones de mayor calidad hasta el año 2008, cuando esa valoración cambió a A*. El resto de valoraciones se han mantenido a lo largo del tiempo.

Aunque sólo se facilitan dos rankings, uno para revistas y otro para conferencias, las publicaciones se agrupan por macrocategorías y categorías a través de su código FoR. Cada macrocategoría tiene un código FoR de dos dígitos y puede estar compuesta por varias categorías. Las principales macrocategorías son las siguientes:

- *Mathematical Sciences (FoR 01)*
- *Physical Sciences (FoR 02)*
- *Chemical Sciences (FoR 03)*
- *Earth Sciences (FoR 04)*
- *Environmental Sciences (FoR 05)*
- *Biological Sciences (FoR 06)*
- *Agricultural and Veterinary Sciences (FoR 07)*
- *Information and Computing Sciences (FoR 08)*
- *Engineering (FoR 09)*
- *Technology (FoR 10)*
- *Medical and Health Sciences (FoR 11)*
- *Built Environment and Design (FoR 12)*
- *Education (FoR 13)*
- *Economics (FoR 14)*
- *Commerce, Management, Tourism and Services (FoR 15)*
- *Studies In Human Society (FoR 16)*
- *Psychology and Cognitive Sciences (FoR 17)*
- *Law and Legal Studies (FoR 18)*
- *Studies In Creative Arts and Writing (FoR 19)*
- *Language, Communication and Culture (FoR 20)*
- *History and Archaeology (FoR 22)*
- *Philosophy and Religious Studies (FoR 23)*

Cada macrocategoría engloba las categorías cuyos dos primeros dígitos del código FoR se corresponden con los dos primeros del código de la macrocategoría. Las macrocategorías más importantes y que contienen la mayor parte de las publicaciones hacen referencia al área de informática, ingeniería y tecnología (códigos 08, 09 y 10). Por ejemplo, las categorías correspondientes a la macrocategoría 'Information and Computing Sciences':

- *Artificial Intelligence and Image Processing (FoR 0801)*

- *Computation Theory and Mathematics (FoR 0802)*
- *Computer Software (FoR 0803)*
- *Data Format (FoR 0804)*
- *Distributed Computing (FoR 0805)*
- *Information Systems (FoR 0806)*
- *Library and Information Studies (FoR 0807)*
- *Other Information and Computing Sciences (For 0899)*

Dentro de una categoría las publicaciones con una misma valoración no tienen orden, pudiendo una publicación pertenecer a varias categorías diferentes.

Una vez realizado un análisis de CORE, la información disponible en la web que puede ser recuperada corresponde desde el año 2007 en adelante. A continuación se presenta un fragmento del ranking CORE para revistas en el año 2009:

ERAID	Rank	Title	FoR1
17833	A*	Decision Support Systems	08
4456	A*	IEEE Transactions on Medical Imaging	08
32143	A*	Journal of the ACM	08
17794	A*	The Computer Journal	08
17734	A*	ACM Transactions on Graphics	0801
17761	A*	Artificial Intelligence	0801
6131	A*	Behavioral and Brain Sciences	0801
6187	A*	Cognition	0801
6193	A*	Cognitive Psychology	0801
6194	A*	Cognitive Science	0801
8765	A*	Computational Linguistics	0801
9940	A*	Digital Creativity	0801

Figura 2: Fragmento de ranking de revistas CORE

Para realizar la consulta de la calidad a través de CORE deben darse los siguientes pasos:

1. Acceder al sitio web de COmputing Research & Education (<http://core.edu.au/>).
2. Seleccionar la opción del menú *Journal Rankings* o *Conference Rankings*.
3. Los listados de valoraciones pueden ordenarse según varios criterios: ordenados por valoración, alfabéticamente, por código FoR, por valoración y código FoR...
4. Una vez se seleccione el criterio de ordenación, se descarga el fichero PDF que contiene la información.
5. La revista o conferencia debe analizarse en el fichero "a mano".

CORE es un índice poco práctico para consultar la calidad de publicaciones. En primer lugar, cuando actualiza la información de la página web se pierde la antigua, de modo que si se desea analizar la calidad de otros años es conveniente tener un histórico de cada versión

que va siendo publicada. Por otro lado, no puede verse la información relativa a una categoría, además de que es necesario conocer a qué categoría hace referencia cada código FoR.

Este índice también presenta una problemática relativa a la consistencia de la información. De un año a otro los nombres de las publicaciones pueden variar, ocasionando que información relativa a una publicación se cargue en el sistema a desarrollar como información de publicaciones diferentes en distintos años.

B.2.1.3 Scientific Journal Rankings (SJR)

SJR es un portal web que recopila los indicadores de calidad de revistas de la base de datos *Scopus*. Al igual que JCR agrupa las publicaciones por categoría, pudiendo también organizarlos por país al que pertenecen. [WebSJR]

Este índice también asigna anualmente un factor numérico de la calidad único para cada revista. La medida es el número de referencias a la publicación dividido por el número de artículos aceptados en los últimos tres años.

SJR también presenta muchas clasificaciones por categorías, por lo que el sistema sólo recopilará la información de aquellas relacionadas con el área de informática y que son las siguientes:

- *Artificial Intelligence*
- *Communication*
- *Computational Mathematics*
- *Computational Mechanics*
- *Computational Theory and Mathematics*
- *Computer Graphics and Computer-Aided Design*
- *Computer Networks and Communications*
- *Computer Science (miscellaneous)*
- *Computer Science Applications*
- *Computer Vision and Pattern Recognition*
- *Computers in Earth Sciences*
- *Discrete Mathematics and Combinatorics*
- *Hardware and Architecture*
- *Human-Computer Interaction*
- *Information System*
- *Multidisciplinary*
- *Signal Processing*
- *Software*
- *Theoretical Computer Science*

Un ejemplo de la categoría SJR '*Hardware and Architecture*' del año 2008 se muestra en la siguiente imagen:

1	Title	ISSN	SJR	Country
2	Digest of Technical Papers - IEEE International Solid-State Circuits Conference	0193-6530	0,457	United States
3	IBM Journal of Research and Development	0018-8646	0,373	United States
4	IEEE Transactions on Nanotechnology	1536-125X	0,233	United States
5	IEEE Circuits and Systems Magazine	1531-636X	0,164	United States
6	Microelectronic Engineering	0167-9317	0,15	Netherlands
7	Displays	0141-9382	0,114	Netherlands
8	ACM Journal on Emerging Technologies in Computing Systems	1550-4832	0,106	United States
9	Journal of the ACM	0004-5411	0,104	United States
10	IEEE/ACM Transactions on Networking	1063-6692	0,098	United States
11	IEEE Network	0890-8044	0,092	United States
12	IEEE Micro	0272-1732	0,085	United States
13	Journal of the Society for Information Display	1071-0922	0,081	United States
14	VLDB Journal	1066-8888	0,079	Germany
15	IEEE Transactions on Very Large Scale Integration (VLSI) Systems	1063-8210	0,075	United States
16	ACM Transactions on Multimedia Computing, Communications and Applications	1551-6865	0,075	United States
17	IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems	0278-0070	0,074	United States
18	Mobile Networks and Applications	1572-8153	0,07	Netherlands
19	IEEE Design and Test of Computers	0740-7475	0,07	United States
20	Journal of Optical Networking	1536-5379	0,07	United States
21	IEEE Transactions on Computers	0018-9340	0,068	United States

Figura 3: Ejemplo de categoría SJR

La existencia de este índice también se remonta tiempo atrás y debido a los mismos motivos de fiabilidad que JCR se utilizan los datos de las categorías enumeradas desde el año 2002.

Los pasos a dar para consultar la información de calidad de SJR son los siguientes:

1. Acceder a la página web de *Scimago Revista & Country Rank* (<http://www.scimagojr.com/index.php>).
2. Seleccionar la opción del menú *Journal Search*.
3. Introducir el nombre o ISSN de la revista de la que encontrar la calidad.
4. Seleccionar del resultado del análisis la revista deseada.
5. Para saber su clasificación en las categorías, debe accederse al enlace de cada una de ellas.

Esta web muestra mucha más información que JCR en una sola búsqueda. Por ejemplo, puede consultarse la evolución del factor SJR de calidad y las citas por año para todos los años desde 1999 a 2008 a través de una gráfica. Sin embargo, al igual que ocurría para JCR, en el caso de que se quiera consultar la posición de una revista en una categoría para un año en concreto no puede hacerse con una sola búsqueda. Para ello, es necesario consultar la información de cada categoría año a año.

B.2.1.4 Computer Science Conference Ranking (CSCR)

CSCR es un índice de calidad para conferencias pertenecientes al área de informática. Al igual que SJR y JCR, asigna una valoración numérica a cada evento entre 0.0 y 1.0 agrupándolos por categoría.

Una conferencia sólo es incluida en una sola categoría, mostrando sólo aquellas con valoración mayor o igual a 0.5. Cuanta más alta es la valoración mayor es el prestigio de la conferencia. Cada categoría agrupa varias temáticas y se listan a continuación:

- *Databases / Knowledge and Data Management / Data Security / Web / Mining*

- *Artificial Intelligence / Machine Learning / Robotics / Human Computer Interaction*
- *Architecture / Hardware / High-Performance Computing / Tools / Operating Systems*
- *Applications / Education / Software / Theory / Communications / Graphics / Bioinformatics*

Las conferencias aparecen ordenadas por valoración, indicando para cada una de ellas su título, siglas y valoración. Un ejemplo de la categoría '*Applications / Education / Software / Theory / Communications / Graphics / Bioinformatics*' para el año 2009 es el siguiente:

Top 159 conferences are listed (642 considered):	
SIGCOMM: ACM Conf on Comm Architectures, Protocols & Apps (0.99)	
STOC: ACM Symp on Theory of Computing (0.99)	
INFOCOM: Annual Joint Conf IEEE Comp & Comm Soc (0.99)	
CSCW: Conference on Computer Supported Cooperative Work (0.99)	
FOCS: IEEE Symp on Foundations of Computer Science (0.99)	
RECOMB: Annual Intl Conf on Comp Molecular Biology (0.99)	
COLT: Computational Learning Theory (0.98)	
LICS: IEEE Symp on Logic in Computer Science (0.98)	
SCG: ACM Symp on Computational Geometry (0.98)	
SODA: ACM/SIAM Symp on Discrete Algorithms (0.98)	
ISSAC: Intl. Symp on Symbolic and Algebraic Computation (0.98)	
CRYPTO: Advances in Cryptology (0.98)	
SOSP: ACM SIGOPS Symp on OS Principles (0.98)	
SOSDI: Usenix Symp on OS Design and Implementation (0.97)	
IEEE Visualization (0.97)	
CCS: ACM Conf on Comp and Communications Security (0.97)	
POPL: ACM-SIGACT Symp on Principles of Prog Langs (0.97)	
EUROCRYPT: European Conf on Cryptography (0.97)	
PLDI: ACM-SIGPLAN Symp on Prog Lang Design & Impl (0.97)	
OOPSLA: OO Prog Systems, Langs and Applications (0.97)	
ICFP: Intl Conf on Function Programming (0.97)	
JICSLP/CLP/LPS: (Joint) Intl Conf/Symp on Logic Prog (0.97)	
SIGGRAPH: ACM SIGGRAPH Conference (0.97)	
FSE: ACM Conf on the Foundations of Software Engineering (inc: ESEC-FSE) (0.97)	
FM/FME: Formal Methods, World Congress/Europe (0.97)	
ICDCS: IEEE Intl Conf on Distributed Comp Systems (0.97)	
ISDG: ACM-SIGGRAPH Interactive 3D Graphics (0.97)	
ACM-MM: ACM Multimedia Conference (0.97)	
DCC: Data Compression Conf (0.97)	
PECCS: IFIP Intl Conf on Perf Eval of Comp & Comm Sys (0.97)	
ISTCS: Israel Symp on Theory of Computing and Systems (0.96)	

Figura 4: Ejemplo de categoría CSCR

Los índices son actualizados trimestralmente, apareciendo nuevas versiones en Enero, Abril, Julio y Octubre. El principal problema que presenta CSCR es que las valoraciones se muestran en la página web no pudiendo consultar valoraciones anteriores a la versión que actualmente se muestra en el sitio web.

Para realizar la consulta de la calidad de una conferencia deben realizarse las siguientes acciones:

1. Acceder al sitio web de *Computer Science Conference Ranking* (<http://www.cs-conference-ranking.org/conferencerrankings/alltopics.html>).
2. Analizar en el listado de categorías aquella a la que se sepa que pertenece la conferencia.
3. Analizar dentro de esa categoría la conferencia a través de su nombre.

CSCR parece estar desfasado. Desde finales del 2009 su página web no está disponible, no pudiendo consultar información actualizada. Al igual que CORE es necesario guardar un historial de las versiones antiguas y sólo puede consultarse la valoración de aquellas publicaciones con un factor mayor que 0.5.

B.3 Primera fase: Ideación

La fase de ideación pretende definir en torno al 80% de la funcionalidad del sistema y así determinar la viabilidad de éste. Esta fase se compone de la identificación y definición de los casos de uso y se apoya en los requisitos funcionales y no funcionales capturados. Estos requisitos del sistema están descritos de manera clara y concisa en el anexo de captura de requisitos.

Como consecuencia de la problemática de publicaciones duplicadas al importar los datos desde las fuentes de calidad, se amplió la funcionalidad del sistema realizándose dos iteraciones en esta fase:

- Durante la primera iteración fueron identificadas las tres principales funcionalidades de análisis de la calidad de publicaciones: análisis de una publicación por nombre/siglas o ISSN, una lista de publicaciones a través de citas bibliográficas que las referencian y publicaciones en las que ha publicado un autor su trabajo. Además, se incorporó la funcionalidad de acceder a las categorías de índice a través del analizador de una publicación, ya que es el único punto donde se muestra información de las categorías.
- Posteriormente, en la segunda iteración, se definió la nueva funcionalidad del mantenimiento de publicaciones duplicadas.

B.3.1 Diagrama de Contexto

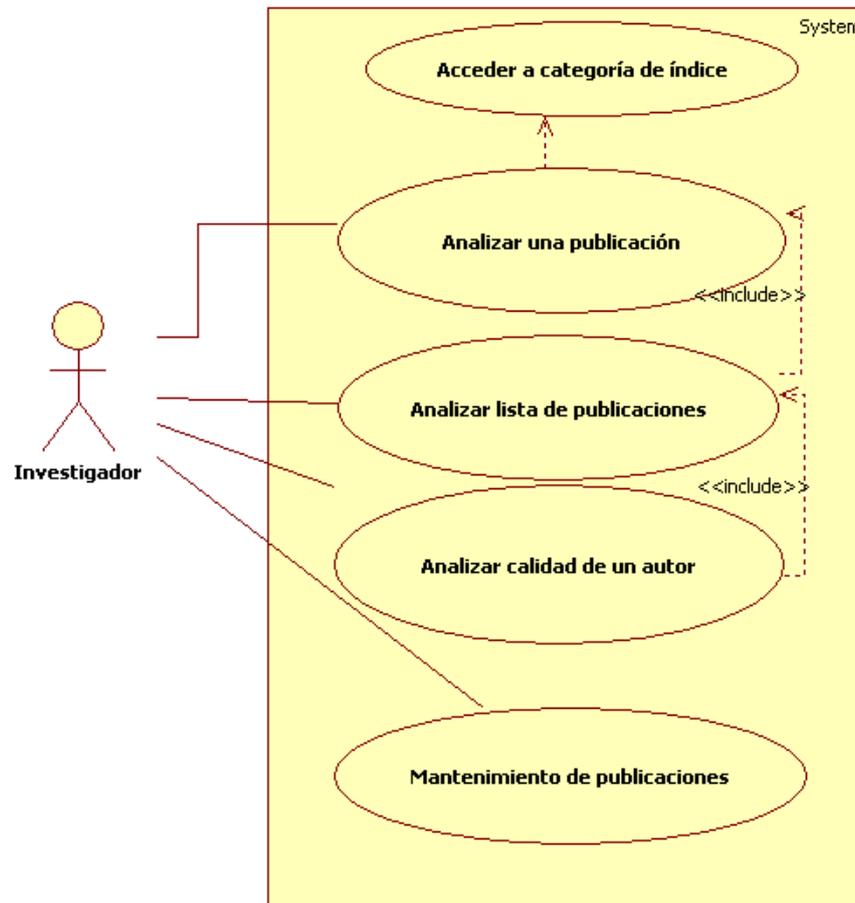


Figura 5: Diagrama de contexto con los principales casos de uso

B.3.2 Modelos de casos de uso

A continuación se describe detalladamente cada caso de uso siguiendo un formato tabla:

B.3.2.1 CU Analizar publicación

Alcance	Sistema Publindex
Nivel	Objetivo usuario
Actor principal	Investigador
Partes interesadas	Investigador: quiere encontrar las valoraciones de una publicación.
Pre-condiciones	Deben existir datos almacenados relativos a publicaciones e índices de calidad en el sistema.

Garantías de éxito	La publicación a analizar puede o no existir
Escenario principal de éxito	<ol style="list-style-type: none"> 1. El investigador selecciona la opción de analizar una publicación por nombre/siglas. 2. El investigador introduce el nombre de una publicación y selecciona las macro categorías donde buscar. 3. El sistema muestra las coincidencias en la base de datos para ese nombre/siglas. 4. El usuario selecciona una publicación del listado de publicaciones. 5. El sistema muestra toda la información relativa a la publicación, valoraciones y categorías a las que pertenece para cada fuente. 6. El investigador termina el proceso.
Extensiones	<ol style="list-style-type: none"> 1a. El investigador selecciona la opción de analizar una publicación por ISSN. <ol style="list-style-type: none"> 1. El investigador introduce el ISSN y selecciona las macro categorías donde buscar. 2. El sistema muestra toda la información relativa a la publicación, valoraciones y categorías a las que pertenece para cada fuente. 3. El investigador termina el proceso. 3a. El sistema no encuentra ninguna coincidencia. <ol style="list-style-type: none"> 1. El sistema muestra un mensaje de error. 2. El Investigador termina el proceso 6a. El investigador selecciona una categoría para visualizar sus publicaciones. <ol style="list-style-type: none"> 1. El investigador selecciona una nueva publicación a analizar. 2. Mismos pasos que en escenario de éxito, desde punto 5.
Requerimientos especiales	-
Otras informaciones	Riesgo: existencia de varias publicaciones con nombre similar pero que son tratadas como publicaciones diferentes y en realidad

	son la misma. Solucionado con el subsistema de mantenimiento de publicaciones.
--	--

Tabla 1: Descripción del caso de uso de analizar una publicación

B.3.2.2 CU Analizar lista de publicaciones

Alcance	Sistema Publindex
Nivel	Objetivo usuario
Actor principal	Investigador
Partes interesadas	Investigador: quiere encontrar las valoraciones de una lista de publicaciones a través de una serie de citas bibliográficas.
Pre-condiciones	Deben existir datos almacenados relativos a publicaciones e índices de calidad en el sistema.
Garantías de éxito	Las publicaciones a las que las citas hacen referencia pueden existir o no en el sistema.
Escenario principal de éxito	<ol style="list-style-type: none"> 1. El investigador selecciona la opción de analizar una lista de publicaciones. 2. El investigador introduce un listado de citas en formato texto o BibTex. 3. El sistema analiza las citas y su formato y procesa la información para buscar las publicaciones a las que más se asemejan los nombres/siglas procesados. 4. El sistema muestra un listado de cada cita y la publicación a la que hace referencia. 5. El investigador selecciona una de las publicaciones del listado para analizar su calidad. 6. El sistema muestra toda la información relativa a la publicación, valoraciones y categorías a las que pertenece para cada fuente. 7. El investigador termina el proceso.
Extensiones	<p>3a. Las citas no poseen un formato correcto</p> <ol style="list-style-type: none"> 1. El sistema muestra un mensaje de error. 2. El investigador termina el proceso. <p>7a. El investigador selecciona una categoría</p>

	<p>para visualizar las publicaciones.</p> <ol style="list-style-type: none"> 1. El investigador selecciona una nueva publicación a analizar. 2. Mismos pasos que en escenario de éxito, desde punto 5.
Requerimientos especiales	-
Otras informaciones	<p>Problema abierto: el analizador de citas no reconozca todos los formatos posibles o reconozca erróneamente el nombre de una publicación.</p> <p>Riesgo: obtener la publicación cuyo título se asemeje más al extraído de la cita. Solución: método que calcule la semejanza entre dos cadenas (p.e. distancia lexicográfica).</p>

Tabla 2: Descripción del caso de uso de analizar publicaciones por citas bibliográficas

B.3.2.3 Analizar calidad de un autor

Alcance	Sistema Publindex
Nivel	Objetivo usuario
Actor principal	Investigador
Partes interesadas	Investigador: quiere encontrar las valoraciones de un autor, a través de una serie de referencias bibliográficas, dado su nombre.
Pre-condiciones	Deben existir datos almacenados relativos a publicaciones e índices de calidad en el sistema.
Garantías de éxito	El autor del que se quiere analizar la calidad puede no existir.
Escenario principal de éxito	<ol style="list-style-type: none"> 1. El investigador selecciona la opción de analizar la calidad de un autor. 2. El sistema busca las referencias bibliográficas del autor. 3. El sistema analiza las citas y su formato y procesa la información para analizar las publicaciones a las que más se asemejan los nombres/siglas procesados. 4. El sistema muestra el trabajo del autor: un listado de cada cita y la publicación a la que

	<p>hace referencia.</p> <p>5. El investigador selecciona una de las publicaciones del listado para visualizar su calidad.</p> <p>6. El sistema muestra toda la información relativa a la publicación, valoraciones y categorías a las que pertenece para cada fuente.</p> <p>7. El investigador termina el proceso.</p>
Extensiones	<p>2a. No existe el autor seleccionado.</p> <ol style="list-style-type: none"> 1. El sistema muestra un mensaje de error. 2. El investigador termina el proceso. <p>3a. Una de las citas no presenta un formato correcto.</p> <ol style="list-style-type: none"> 1. El sistema no la incluye en el listado de citas resultado. 2. Mismos pasos de escenario de éxito desde 4. <p>7a. El investigador selecciona una categoría para visualizar las publicaciones.</p> <ol style="list-style-type: none"> 1. El investigador selecciona una nueva publicación de la que analizar la calidad. 2. Mismos pasos que en escenario de éxito, desde punto 5.
Requerimientos especiales	-
Otras informaciones	Problema abierto: el analizador de citas no reconozca todos los formatos posibles o reconozca erróneamente el nombre de una publicación.

Tabla 3: Descripción del caso de uso de analizar publicaciones por autor

B.3.2.4 CU Mantenimiento de publicaciones

Alcance	Sistema Publindex
Nivel	Objetivo usuario
Actor principal	Investigador
Partes interesadas	Investigador: quiere fusionar en una sola publicación dos publicaciones con nombres

	diferentes.
Pre-condiciones	Deben existir datos almacenados relativos a publicaciones e índices de calidad en el sistema.
Garantías de éxito	Se fusionan las dos publicaciones seleccionadas.
Escenario principal de éxito	<ol style="list-style-type: none"> 1. El investigador busca a través del analizador una publicación. 2. El sistema muestra el listado de coincidencias de publicaciones. 3. El usuario selecciona una publicación para fusionar. 4. El sistema muestra la página inicial del mantenimiento de publicaciones para que seleccione la publicación destino de la fusión. 5. El usuario introduce el nombre/siglas/ISSN para buscar la publicación destino. 6. El sistema muestra el listado de coincidencias de la búsqueda. 7. El usuario selecciona una coincidencia del listado. 9. El usuario confirma la fusión. 10. El sistema muestra mensaje de éxito. 11. El usuario termina el proceso.
Extensiones	<p>6a. El sistema no encuentra la publicación destino</p> <ol style="list-style-type: none"> 1. El sistema lanza un mensaje de error. 2. El usuario termina el proceso.
Requerimientos especiales	-
Otras informaciones	<p>Problema abierto: El usuario podría introducir como publicación destino cualquiera realizando una fusión no correcta a nivel de información.</p> <p>Riesgo: al reiniciar el estado de la base de datos pueden perderse los cambios sobre ésta. Solución: fichero de log.</p>

Tabla 4: Descripción del caso de uso de mantenimiento de publicaciones

B.4 Segunda fase: Elaboración (Análisis)

La fase de elaboración de Publindex consiste en dos iteraciones. Debido a que los analizadores de publicaciones y por autor dependen directamente del analizador de una publicación, se planificó realizar dos iteraciones:

- La primera iteración comprende el análisis y diseño del escenario del analizador de una publicación, junto con la implementación de la interfaz y base de datos.
- La segunda iteración comprende el análisis y diseño de los analizadores de lista de publicaciones, por autor y del subsistema de mantenimiento. También se realiza la implementación de los aspectos de mayor riesgo, la arquitectura del sistema y base de datos.

Este apartado recoge todas las labores de análisis realizadas durante la fase de elaboración en cada iteración definida previamente.

B.4.1 Primera iteración

Esta primera iteración contiene el modelo de dominio y los diagramas de secuencia del CU del analizador de una publicación.

B.4.1.1 Modelo de dominio parcial

El modelado de dominio permite representar visualmente las clases conceptuales o lo que es lo mismo, los objetos reales del dominio y las asociaciones entre ella. Se identificaron las siguientes clases y los siguientes atributos:

- Publicación: Nombre, País, Lenguaje, Siglas
- Revista: ISSN
- Conferencia
- Índice de calidad: Nombre
- Categoría: Nombre, Año
- Valoración de calidad: Valoración, Posición
- Autor: Nombre
- Cita: Título, Año
- Macro categoría: Nombre

En el siguiente diagrama de clases se muestran las asociaciones establecidas entre las clases identificadas:

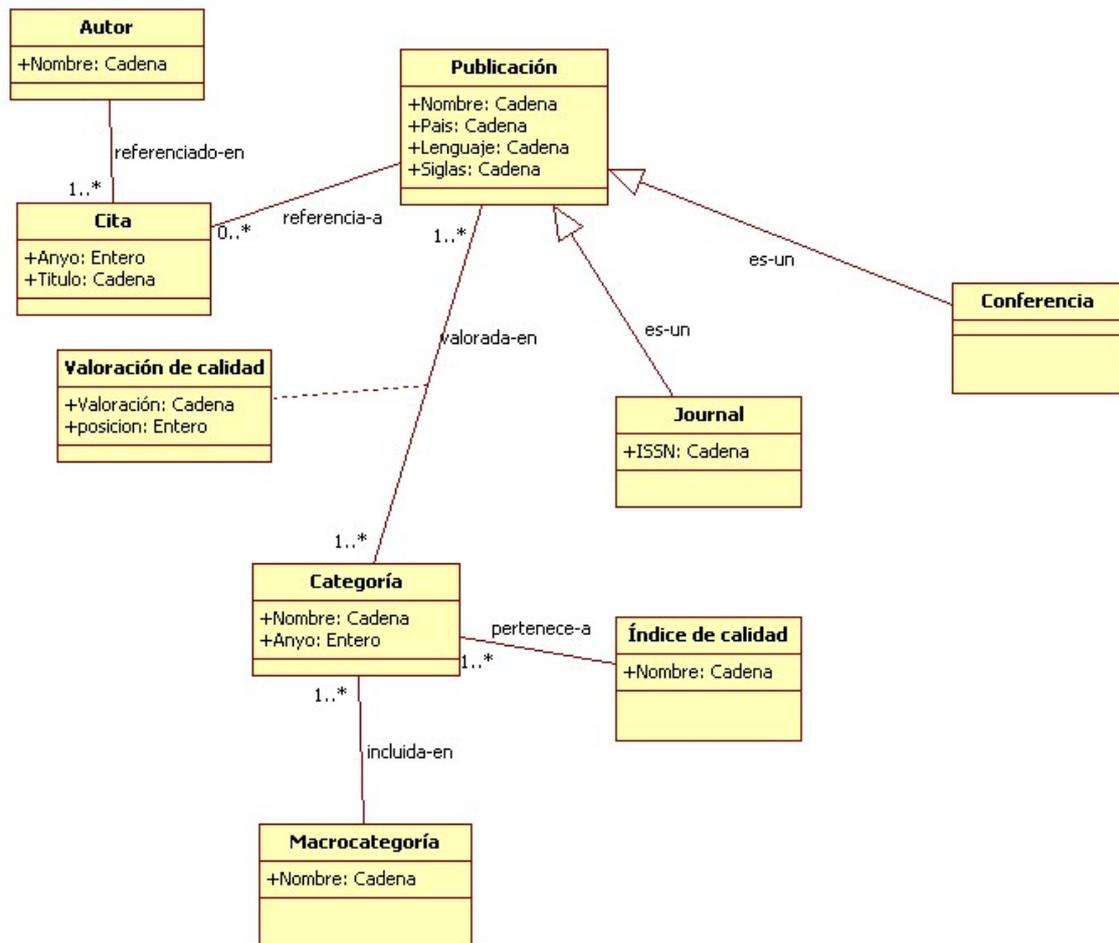


Figura 6: Modelo de dominio del problema

B.4.1.2 Diagramas de secuencia

A través de estos diagramas se representan los eventos de entrada y salida del sistema, y así poder identificar los eventos que en el diseño se refinarán para su posterior implementación.

A continuación se presentan los diagramas de secuencia para cada caso de uso definido durante la fase de ideación y correspondientes a la primera iteración:

Analizador de publicación

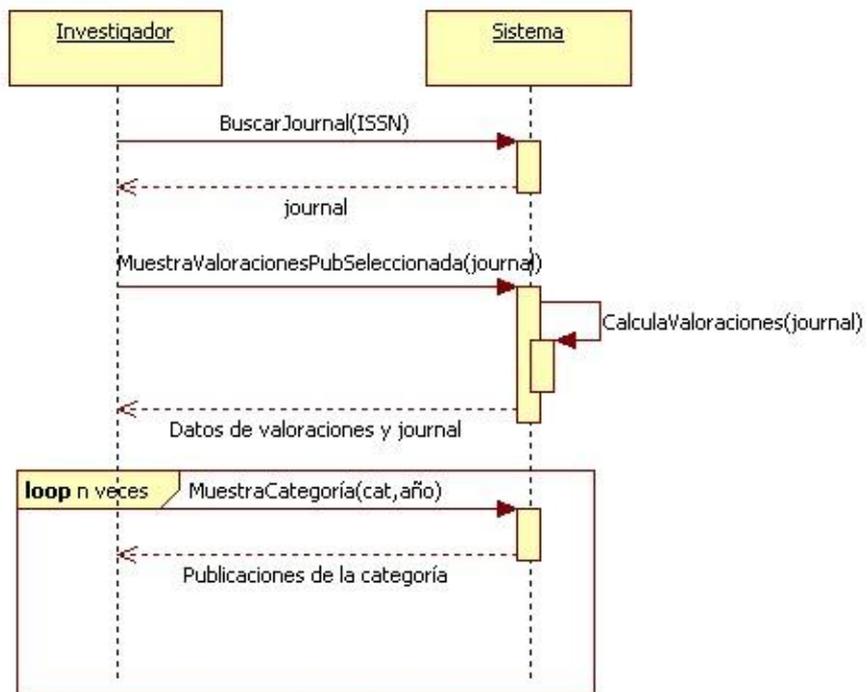
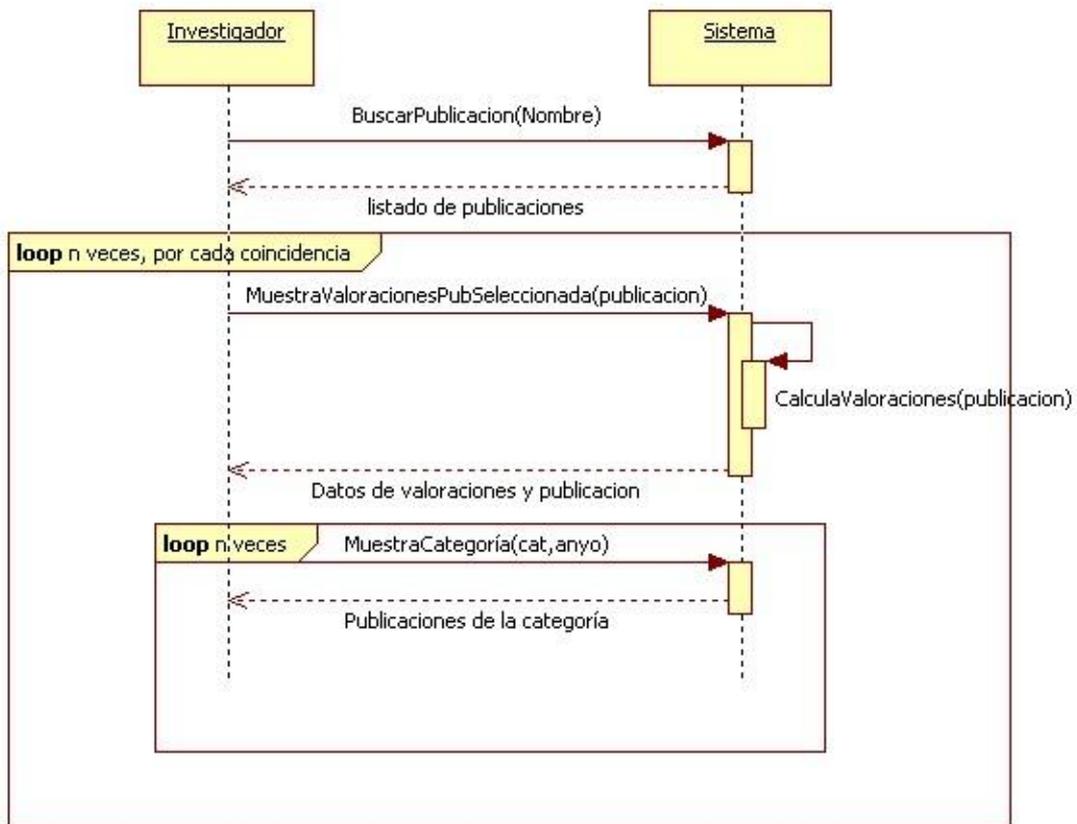


Figura 7: Diagrama de secuencia para el analizador de una publicación dado su nombre/siglas o ISSN

Los contratos de operaciones mostradas en el anterior diagrama son los siguientes:

- Contrato C1
 - Operación: BuscarPublicacion(cadena Nombre)
 - Referencia: CU “Analizar una publicación”
 - Pre-condiciones:
 - Nombre no es una cadena vacía.
 - Post-condiciones:
 - Se obtiene una lista de publicaciones de tipo Revista/Conferencia con todos aquellos que contengan en su nombre o siglas la cadena pasada por parámetro.
 - Se obtiene una lista de publicaciones de tipo Conferencia con todos aquellos que tengan como sub cadena en su nombre o siglas la cadena pasada por parámetro.
- Contrato C2
 - Operación: MuestraValoracionesPubSeleccionada(publicación P)
 - Referencia: CU “Analizar una publicación”
 - Pre-condiciones:
 - P es una publicación válida.
 - Post-condiciones:
 - Se obtienen las valoraciones para la publicación en los índices de calidad.
- Contrato C3
 - Operación: CalculaValoraciones(publicación P)
 - Referencia: CU “Analizar una publicación”
 - Pre-condiciones: P es una publicación válida
 - Post-condiciones:
 - Se calculan las valoraciones de la publicación P en cada categoría a la que pertenezca.
- Contrato C4
 - Operación: MuestraCategoria(categoría C, entero año)
 - Referencia: CU “Analizar una publicación”
 - Pre-condiciones:
 - C es una categoría válida
 - Año es un número válido
 - Post-condiciones:
 - Devuelve las publicaciones de la categoría pasada como parámetros en el año indicado.

- Contrato C5
 - Operación: BuscarJournal(cadena ISSN)
 - Referencia: CU “Analizar una publicación”
 - Pre-condiciones:
 - ISSN es una cadena no vacía
 - Post-condiciones:
 - Devuelve la revista asociada al ISSN pasado por parámetro

B.4.2 Segunda iteración

Esta segunda iteración presenta los diagramas de secuencia de los casos de uso de análisis de una lista de publicaciones y de un autor. Además, se incluyen también los diagramas para el subsistema de mantenimiento de publicaciones. No se incluye el modelo de dominio ya que no presenta cambios en esta iteración.

B.4.2.1 Diagramas de secuencia

Analizar una lista de publicaciones

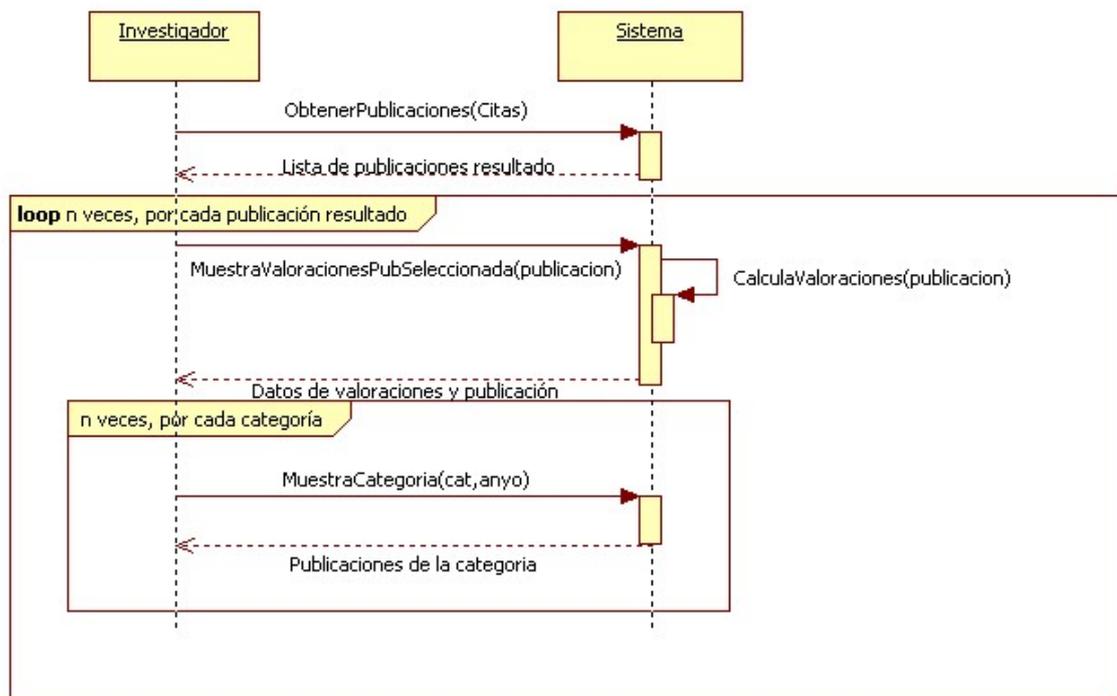


Figura 8: Diagrama de secuencia para el analizador de una lista de publicaciones a través de citas bibliográficas

Los contratos de operación nuevos que se han presentado en el anterior diagrama:

- Contrato C6

- Operación: ObtenerPublicaciones(Lista<Citas> citas)
- Referencia: CU “Analizar lista publicaciones”
- Pre-condiciones:
 - Citas es una lista no vacía
- Post-condiciones:
 - Devuelve la lista de publicaciones correspondiente. Para cada cita obtener aquella publicación que más se asemeje al título de la publicación extraído de la cita.

Analizar publicaciones por autor

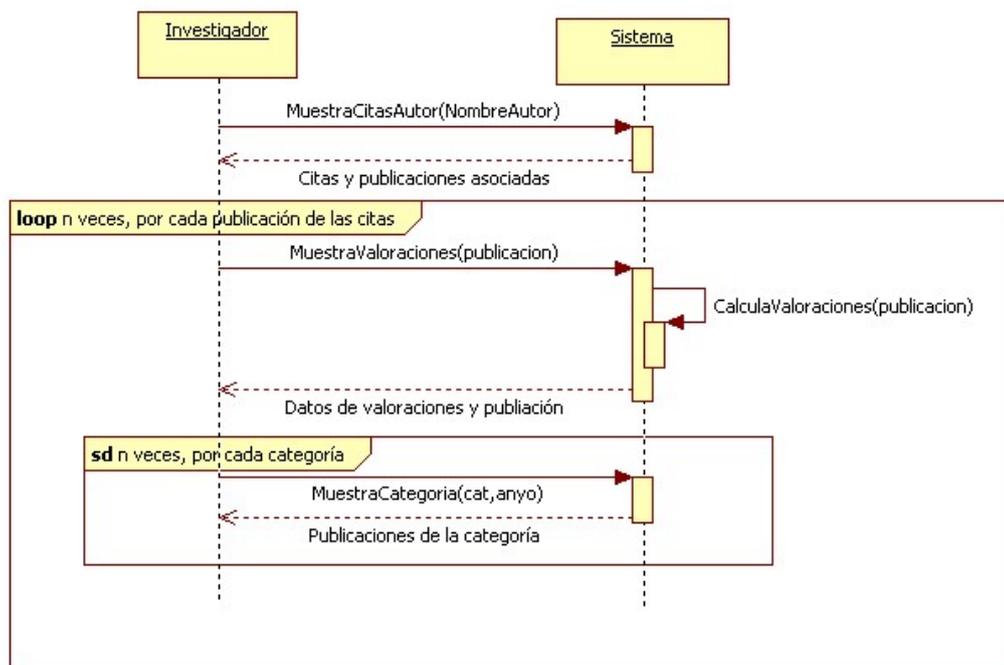


Figura 9: Diagrama de secuencia para el analizador de una lista de publicaciones a partir de un autor

La nueva operación introducida tiene el siguiente contrato:

- Contrato C7
 - Operación: MuestraCitasAutor(cadena NombreAutor)
 - Referencia: CU “Analizar calidad de un autor”
 - Pre-condiciones:
 - NombreAutor no es una cadena vacía
 - Post-condiciones:
 - Devuelve la lista de las publicaciones correspondientes a las citas del autor. Para cada una obtener aquella publicación que más se asemeje al título de la publicación extraído de la cita.

Mantenimiento de publicaciones

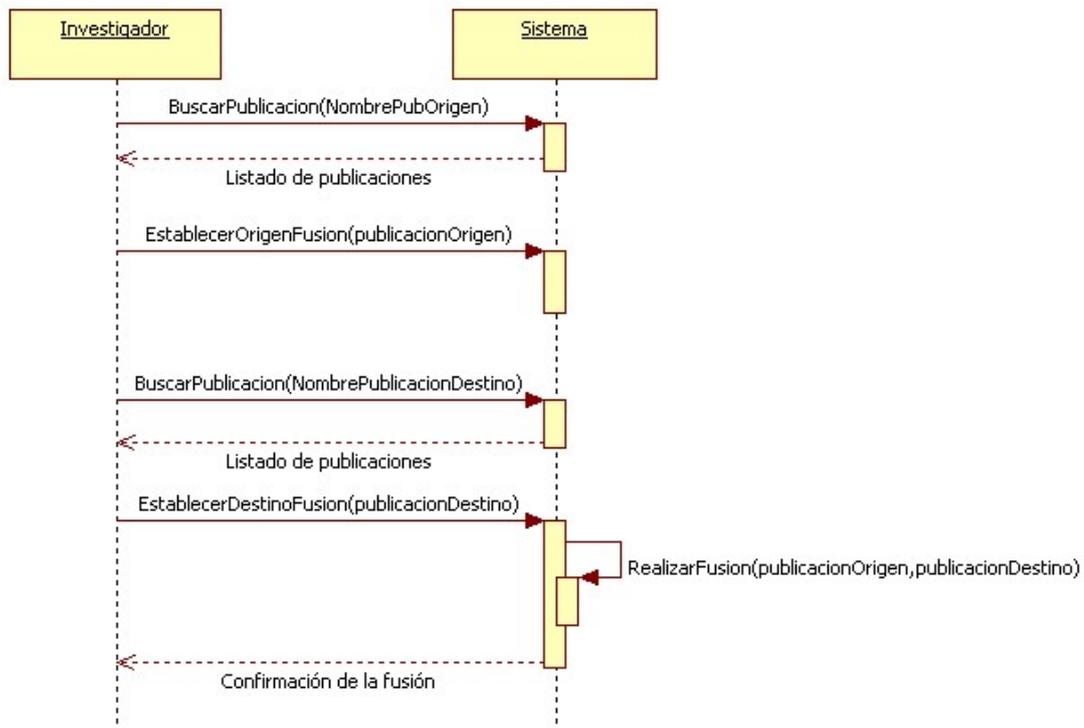


Figura 10: Diagrama de secuencia para el subsistema de mantenimiento de publicaciones

Los nuevos contratos de operación son los siguientes:

- **Contrato C8**
 - Operación: EstablecerOrigenFusion(publicación pubOrigen)
 - Referencia: CU “Mantenimiento de publicaciones”
 - Pre-condiciones:
 - pubOrigen es una publicación válida.
 - Post-condiciones:
 - Establece pubOrigen como publicación origen a fusionar y de la que existe duplicado.
- **Contrato C9**
 - Operación: EstablecerDestinoFusion(publicación pubDestino)
 - Referencia: CU “Mantenimiento de publicaciones”
 - Pre-condiciones:
 - pubDestino es una publicación válida.
 - Post-condiciones:
 - Establece pubDestino como publicación destino de la fusión.
- **Contrato C10**
 - Operación: RealizarFusion(publicación pubOrigen,publicación pubDestino)
 - Referencia: CU “Mantenimiento de publicaciones”

- Pre-condiciones:
 - pubOrigen, pubDestino son publicaciones válidas.
- Post-condiciones:
 - pubDestino tiene asociadas además de las valoraciones suyas propias las de pubOrigen.
 - La publicación pubOrigen ha sido eliminada del sistema.

B.5 Segunda fase: Elaboración (Diseño)

Este apartado presenta el modelo de diseño desarrollado en la fase de elaboración. Indicar que se presenta la parte del proceso de diseño (diagramas de comunicación) que permite definir la interacción entre las clases definidas del sistema y asignar ciertas responsabilidades y funciones para cada una de estas. Toda esta labor permite un refinamiento del modelo de dominio y construir así el primer modelo de clases del sistema.

Durante el proceso de diseño, se aplicaron un conjunto de patrones y principios. Todos ellos serán descritos en detalle en el sub apartado 5.1. Esta sección contiene también la estructura del sistema, el diseño de la base de datos y el prototipado de interfaces de la aplicación (sub apartados 5.4, 5.5 y 5.6 respectivamente).

B.5.1 Patrones de diseño y principios GRASP

El Proceso Unificado permite aplicar principios y patrones de diseño a lo largo del modelado e implementación para identificar y asignar responsabilidades a los objetos software. Estos principios reciben el nombre de principios GRASP (= “agarrar, comprender”).

Los principios GRASP utilizados fueron:

- Principio de separación Modelo-Vista (Controller): este principio permite identificar y/o definir el primer objeto bajo la capa GUI que recibe y controla una operación del sistema [UMLPatterns] [GRASP].
- Principio de High-Cohesion: permite asignar las responsabilidades de modo que la cohesión sea elevada, es decir, identificando los conjuntos de responsabilidades con objeto de englobar en diferentes clases cada conjunto. Siguiendo este principio, se definieron varias clases Controller [UMLPatterns] [GRASP].

Estrechamente relacionado con la aplicación de estos principios, estuvo el uso de ciertos patrones de diseño:

- El patrón de diseño MVC: hizo separar en tres capas la estructura del sistema para separar interfaz, lógica y datos y poder identificar así las clases que se incluirán en cada capa [MVC].
- El patrón de diseño DAO: permite abstraer al sistema del acceso a base de datos. Cada clase DAO ofrece un conjunto de operaciones de consulta o edición sobre la base de datos, de manera que el acceso a datos queda separado de la lógica, sin tener visión desde esta última capa del modelo físico de datos [DAO].
- El patrón de diseño POJO: identifica las entidades de negocio que representan la información intercambiada entre las capas definidas en el sistema.

La aplicación de estos patrones hizo que se identificaran nuevas clases en el sistema:

- AutorDAO, CitaDAO, RevistaDAO, ConferenciaDAO, PublicacionDAO, CategoriaDAO, MacrocategoriaDAO, ValoracionDAO para acceso a datos y control de operaciones básicas del sistema que son llamadas desde interfaz.
- GUI AnalizadorPublicacion, GUI AnalizadorPublicacionesCitas, GUI AnalizadorPublicacionesAutor, GUI MantenimientoPublicaciones para englobar en una clase las interfaces gráficas de cada caso de uso definido.
- Las clases de dominio Cita, Autor, Publicación, Conferencia, Revista, Valoración, Categoría, Macrocategoria como clases POJO de la aplicación. Estas clases deben tener los atributos correspondientes junto con sus métodos *get* y *set* de consulta.

B.5.2 Diagramas de comunicación

Este apartado recoge los diagramas de comunicación, separados por caso de uso, de las dos iteraciones realizadas durante el proceso de elaboración e incluyendo las nuevas clases del sistema definidas.

B.5.2.1 Analizador de una publicación

BuscarPublicación (Nombre)

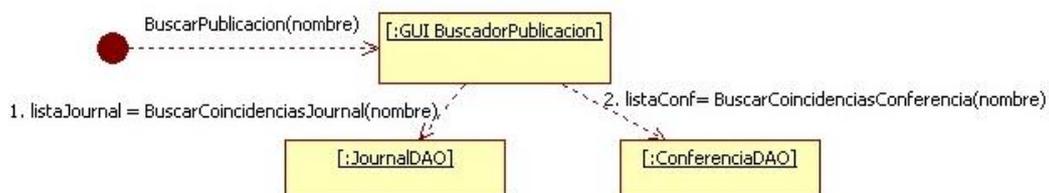


Figura 11: Diagrama de comunicación para el evento *BuscarPublicación (Nombre)*

BuscarJournal (ISSN)

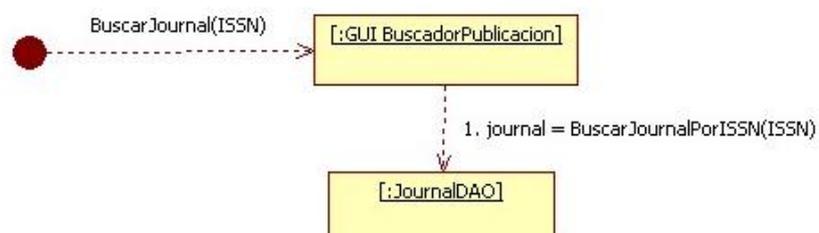


Figura 12: Diagrama de comunicación para el evento *BuscarJournal (ISSN)*

MuestraValoracionesPubSel (publicación)

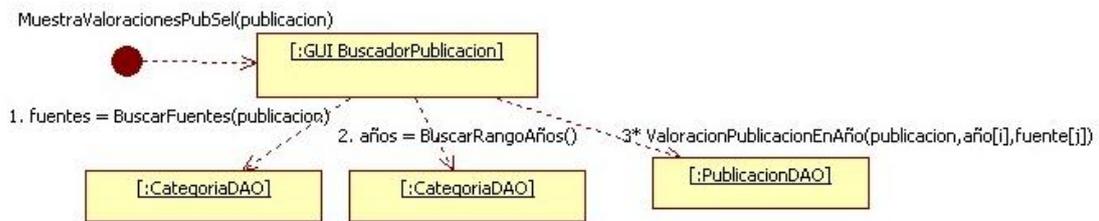


Figura 13: Diagrama de comunicación para el evento MuestraValoracionesPubSel (publicación)

CalculaValoraciones (publicación)

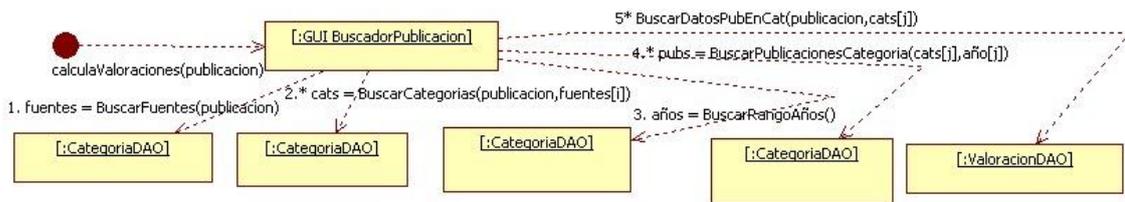


Figura 14: Diagrama de comunicación para el evento CalculaValoracion (publicación)

MuestraCategoria (categoría, año)

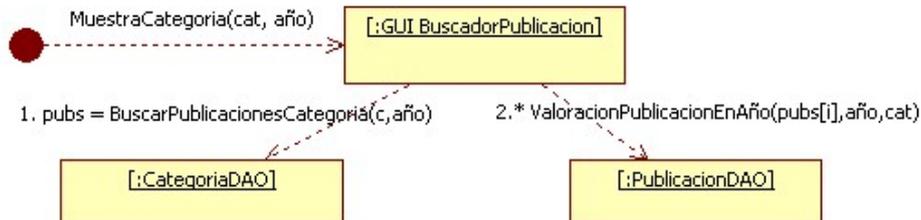


Figura 15: Diagrama de comunicación para el evento MuestraCategoria

B.5.2.2 Analizador de una lista de publicaciones

ObtenerPublicaciones (citas)

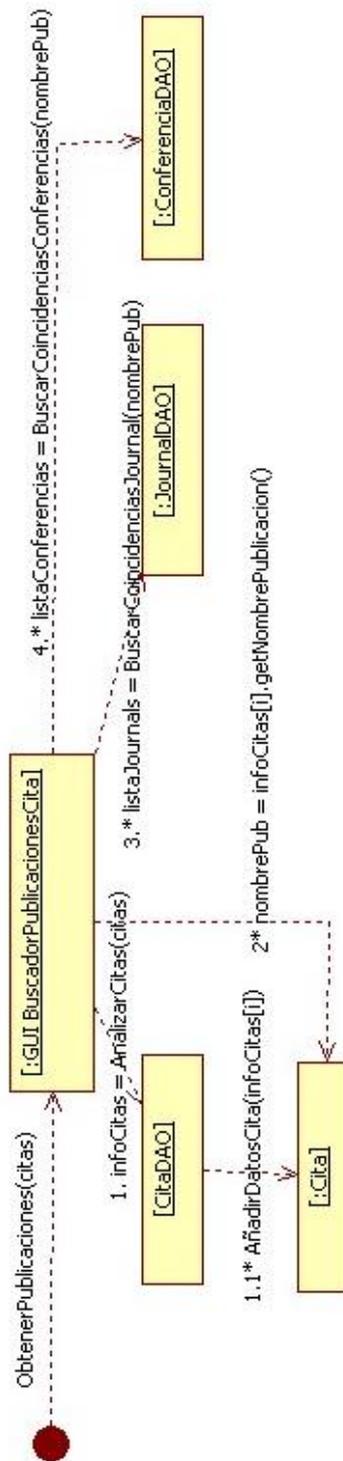


Figura 16: Diagrama de comunicación del evento *ObtenerPublicaciones (citas)*

B.5.2.4 Mantenimiento de publicaciones

EstablecerOrigenFusion (publicación)



Figura 18: Diagrama de comunicación del evento *EstablecerOrigenFusion (publicación)*

EstablecerDestinoFusion (publicación)

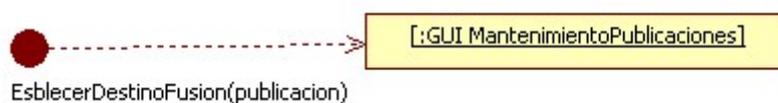


Figura 19: Diagrama de comunicación del evento *EstablecerDestinoFusion*

RealizarFusion (pubOrigen, pubDestino)

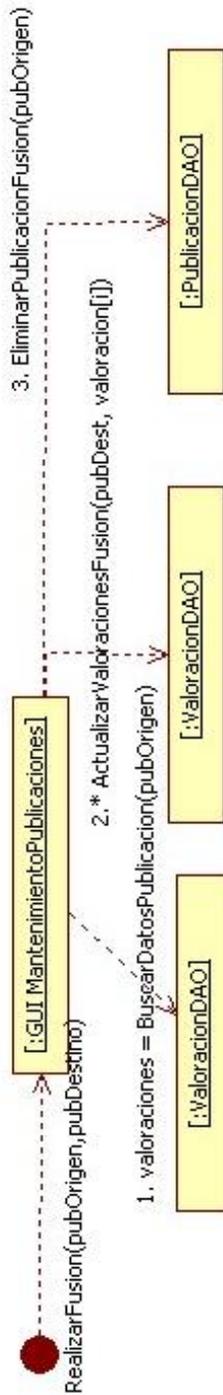


Figura 20: Diagrama de actividades del evento RealizarFusion (pubOrigen, pubDestino)

B.5.3 Diagrama de clases

Este diagrama es el resultado de haber diseñado los diagramas de comunicación anterior y haber aplicado los patrones y principios de diseño previamente comentados. Los cambios que se presentan son:

- Las nuevas clases DAO y GUI que son introducidas con respecto al modelo de dominio.
- Los métodos añadidos en todas clases, incluidas las que previamente fueron definidas en el modelo de dominio.
- Añadidos nuevos atributos a la clase de dominio Cita.

El diagrama es el siguiente:

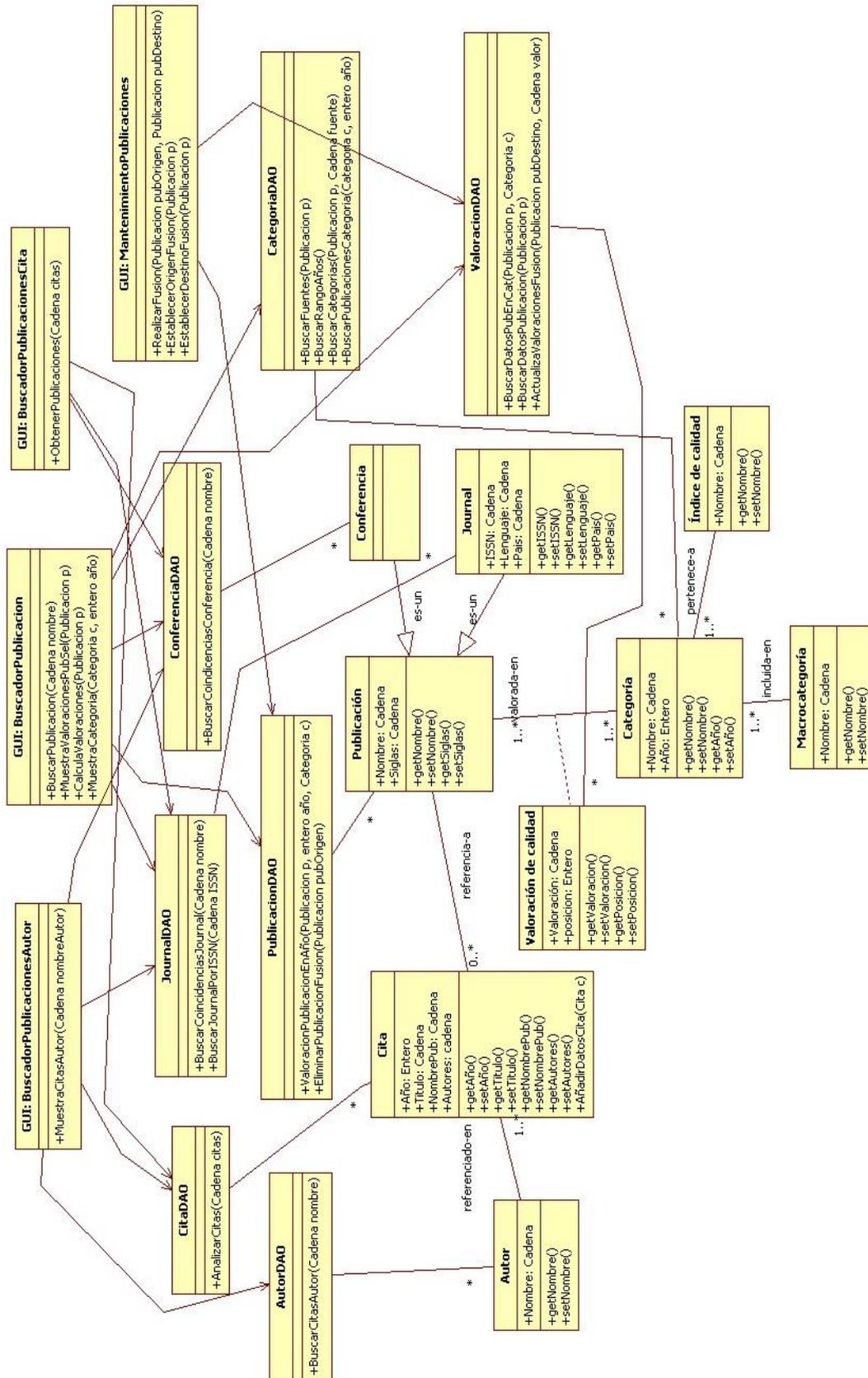


Figura 21: Diagrama de clases. Diseño

B.5.4 Estructura de la solución

El sistema presenta una estructura basada en tres capas: la vista, que contiene todas las clases de interfaz, la lógica que contiene los POJOs y DAOs para conectar interfaz y base de datos, y el modelo físico de los datos, gestionado por el SGBD.

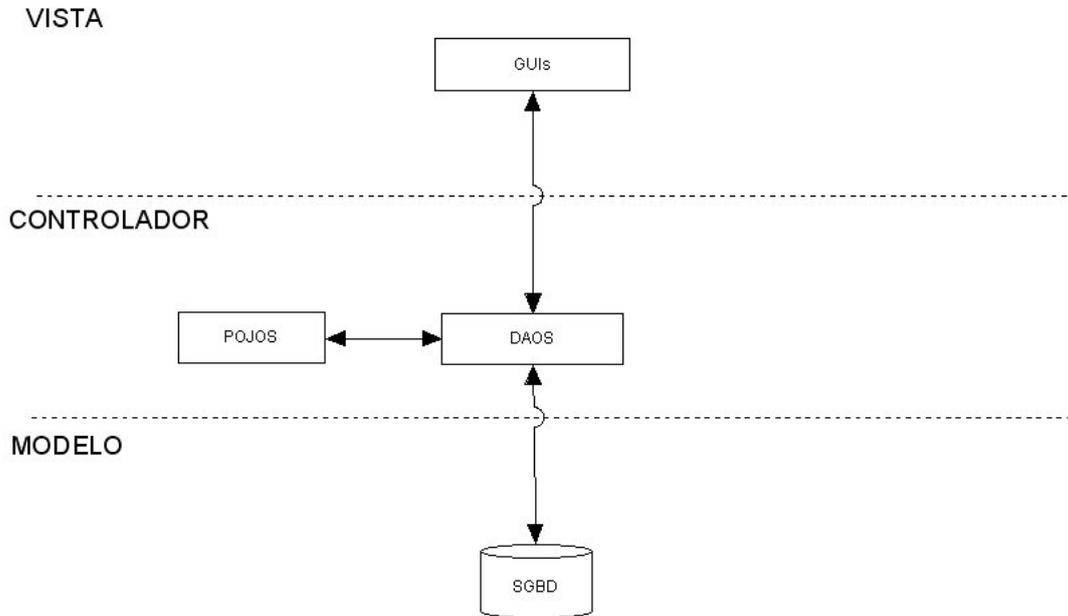


Figura 22: Diseño de la estructura del sistema

B.5.5 Diseño de la base de datos

El diseño de base de datos consiste en identificar las entidades físicas de las cuales es necesario guardar información, utilizando para ello el modelo de clases definido. Además de identificar las entidades, es necesario definir sus campos y las relaciones entre ellas.

La base de datos fue evolucionando a la vez que era refinado el problema y era necesario almacenar información de nuevas entidades. En un primer momento, el modelo presentaba el siguiente aspecto:

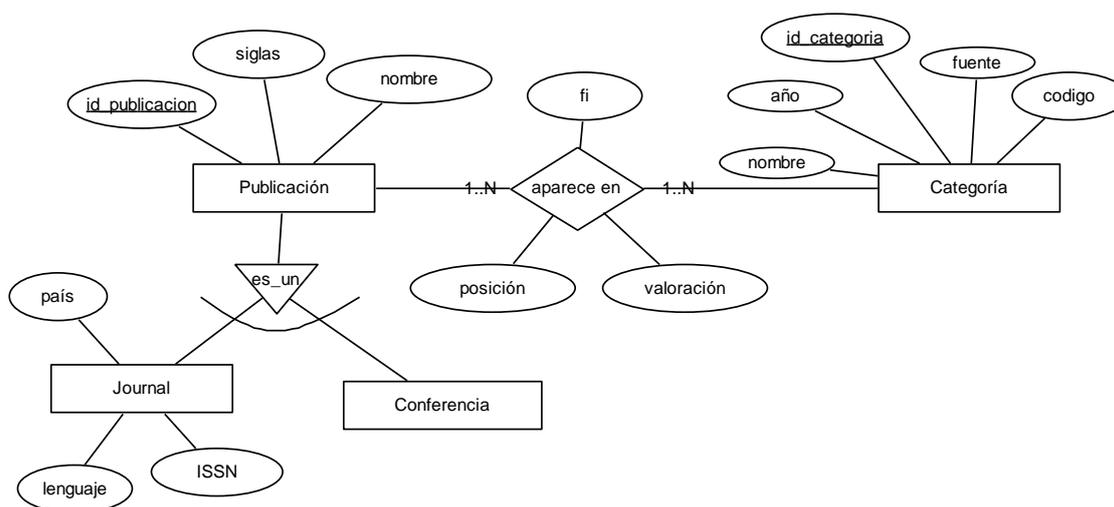


Figura 23: Esquema ER de la versión 1.0 de la base de datos

Conforme se fue definiendo de manera más concreta el problema, se decidió guardar información de los índices cargados cada año e identificar las macrocategorías a las que pertenecían las categorías para poder concretar el espacio de búsqueda. La versión 2.0 de la base de datos es la siguiente:

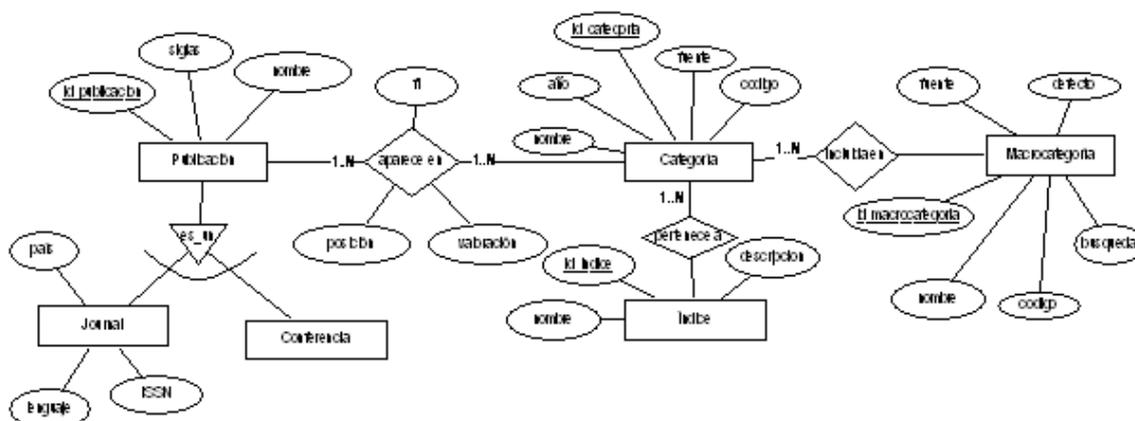


Figura 24: Esquema ER de la versión 2.0 de la base de datos

El sistema fue completado una vez que se definió el análisis de listado de publicaciones por cita y el análisis de la calidad de un autor, quedando la versión definitiva de base de datos. A continuación se presentan las entidades a almacenar y sus campos:

- Autor:
 - Id_autor: entero (clave primaria)
 - nombre: cadena
- Cita:
 - Id_cita: entero (clave primaria)
 - año: entero
 - título: cadena
- Publicación
 - Id_publicación: entero (clave primaria)

- Siglas: cadena
- nombre: cadena
- Journal
 - Atributos y clave primaria de publicación
 - ISSN: cadena
 - lenguaje: cadena
 - país: cadena
- Conferencia
 - Atributos de publicación
- Categoría
 - Id_categoria: entero (clave primaria)
 - fuente: cadena
 - código: cadena
 - año: entero
 - nombre: cadena
- Índice
 - Id_indice: entero (clave primaria)
 - nombre: cadena
 - descripción: cadena
- Macrocategoría
 - Id_macro categoría: entero (clave primaria)
 - nombre: cadena
 - código: cadena
 - fuente: cadena
 - búsqueda: booleano
 - defecto: booleano

El siguiente esquema Entidad-Relación muestra gráficamente el diseño final de base de datos.

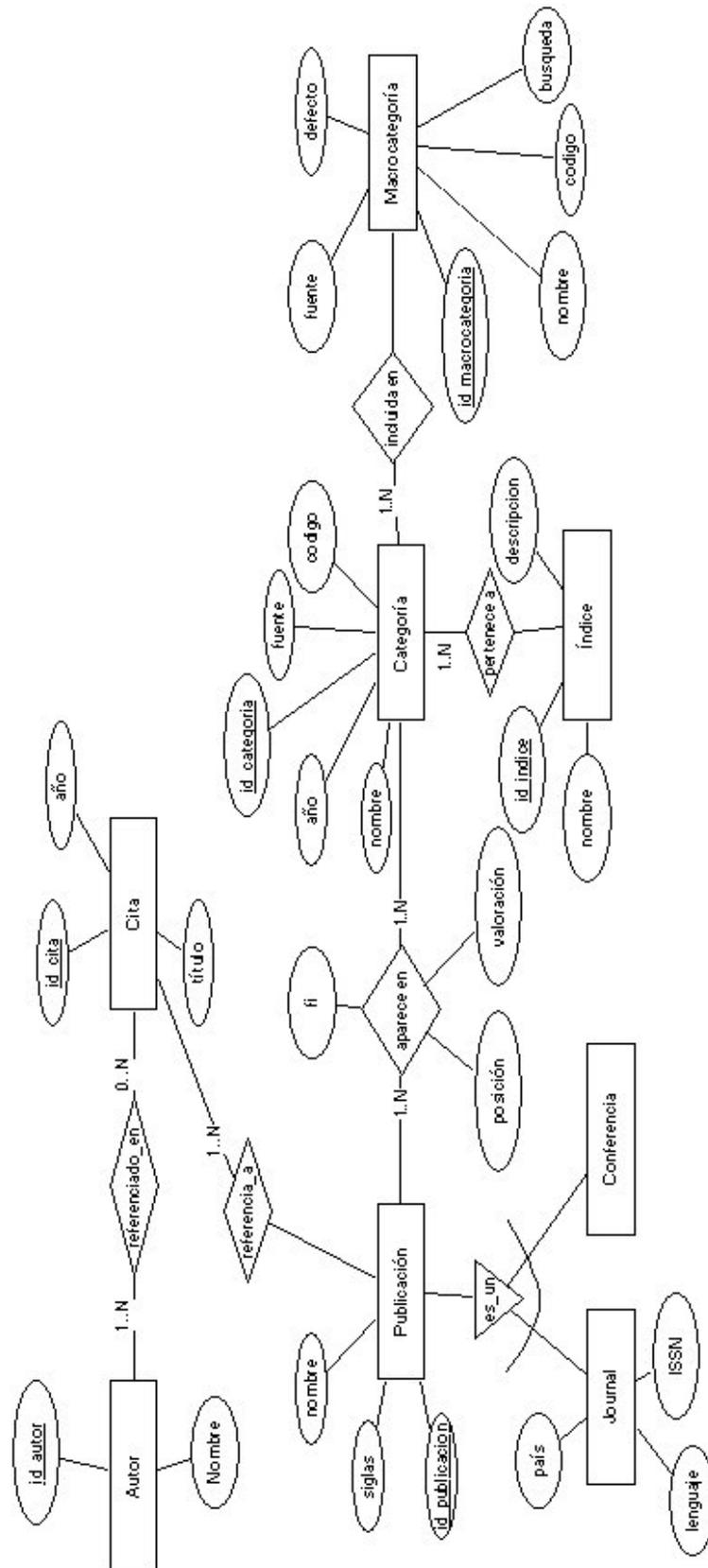


Figura 25: Esquema de diseño Entidad Relación de la base de datos

B.5.6 Prototipado de ventanas

El proceso de diseño también comprendió realizar un prototipado de la apariencia que tendrían las ventanas de la aplicación. Las interfaces se distinguen por poseer una maquetación sencilla compuesta por:

- Cabecera: Contendrá títulos significativos y ocupará todo el espacio de izquierda a derecha de la pantalla. En título que aparecerá en la cabecera de la página web del navegador será el nombre del sistema.
- Contenido: ocupará el resto de la página y aparecerá inmediatamente después de la cabecera. En él aparecerá el contenido de la sección a la que hace referencia. Contendrá elementos tales como enlaces, formularios...

B.5.6.1 Menú principal

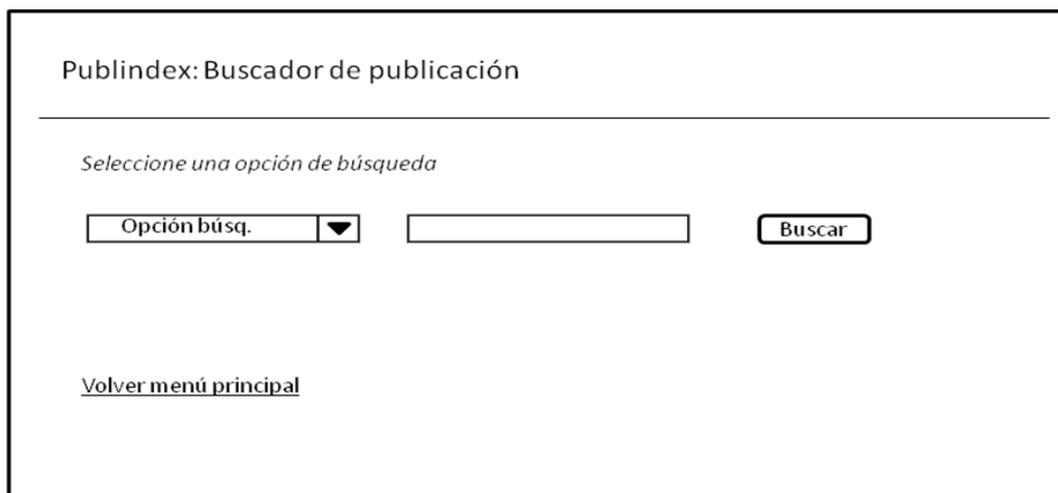


Figura 26: Prototipado de la ventana de menú principal

El menú principal se compone de un listado de enlaces, cada uno redirigiendo a los tres tipos de analizadores que ofrece el sistema.

B.5.6.2 Analizador de una publicación

Página principal



Publindex: Buscador de publicación

Seleccione una opción de búsqueda

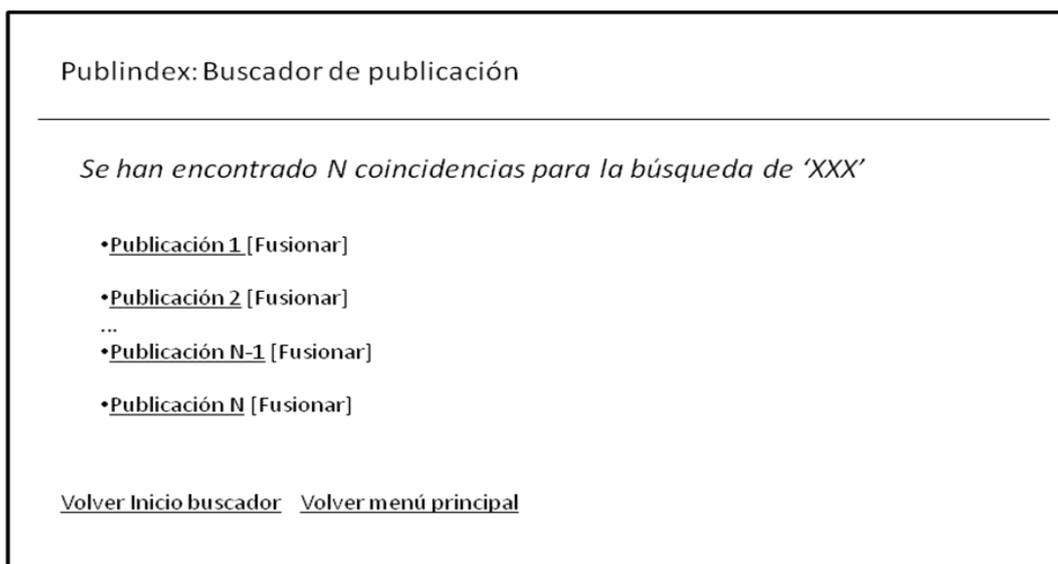
Opción búsq. ▼ Buscar

[Volver menú principal](#)

Figura 27: Página de inicio del analizador de una publicación

Esta página ofrece una lista desplegable, de manera que puede seleccionarse el tipo de búsqueda, bien por nombre/siglas o por ISSN. El siguiente elemento es un cuadro de texto donde introducir la clave y finalmente un botón de búsqueda.

Listado de coincidencias



Publindex: Buscador de publicación

Se han encontrado N coincidencias para la búsqueda de 'XXX'

- [Publicación 1](#) [Fusionar]
- [Publicación 2](#) [Fusionar]
- ...
- [Publicación N-1](#) [Fusionar]
- [Publicación N](#) [Fusionar]

[Volver Inicio buscador](#) [Volver menú principal](#)

Figura 28: Página de resultado de la búsqueda

Esta página muestra un listado con las publicaciones coincidentes según los criterios de búsqueda introducidos en la página anterior. Cada entrada de la lista es un link que redirige a la página de valoraciones y resultados de la publicación a la que referencia el nombre del enlace.

Además, cada entrada de publicación posee un link que redirige al subsistema de mantenimiento de publicaciones y que selecciona como publicación de la que realizar el mantenimiento a la que corresponde dicho enlace.

Resultados de la publicación

Publicacion 'Título'				
País: [País de la publicación (journal)]				
Lenguaje: [Lenguaje de la publicación (journal)]				
ISSN: [ISSN de la publicación (journal)]				
Fuente1	Año 1	Año 2	...	Año N
Valoración	V1	V2	...	Vn
Categorías				
<i>Categ. 1</i>	Top1.1%	Top1.2%	...	Top1.N%
<i>Categ. 2</i>	Top2.1%	Top2.2%	...	Top2.N%
Fuente2	Año 1	Año 2	...	Año N
Valoración	V1	V2	...	Vn
Categorías				
<i>Categ. 1</i>	Top1.1%	Top1.2%	...	Top1.N%
<i>Categ. 2</i>	Top2.1%	Top2.2%	...	Top2.N%

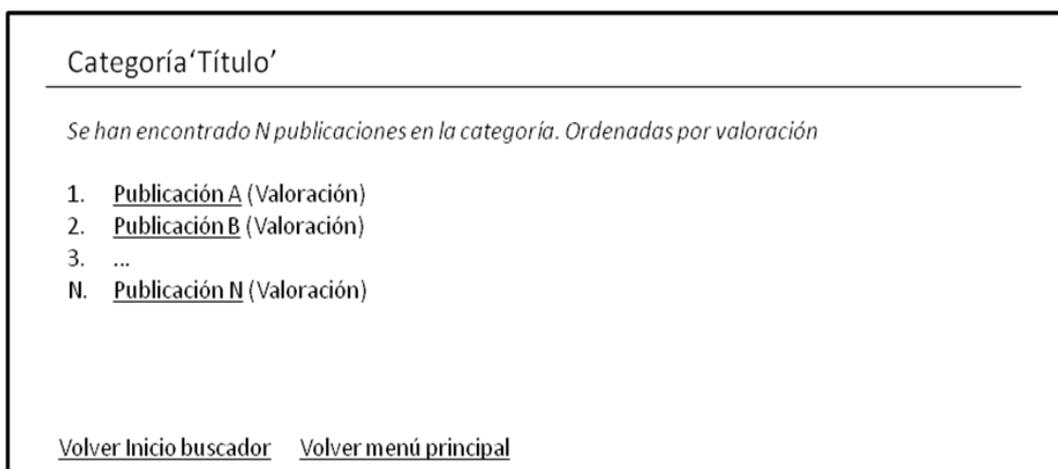
[Volver Inicio buscador](#) [Volver menú principal](#)

Figura 29: Página de datos de la publicación

Esta página muestra todos los datos relativos a la publicación seleccionada. Se compone de información propia de la publicación (país, lenguaje, ISSN) y una tabla donde se muestran todas las valoraciones y puntuaciones de los índices de calidad. La tabla está compuesta por subsecciones, una por cada índice que presenta valoraciones para la publicación:

- Nombre de la fuente y listado de años de los que existen datos almacenados.
- Valoración que es asignada por la fuente a la publicación en cada año del listado.
- Listado de enlaces a las categorías del índice a la que pertenece la publicación y el top % que ocupa dicha publicación según su valoración en esa categoría y ese año.

Datos de una categoría



Categoría 'Título'

Se han encontrado N publicaciones en la categoría. Ordenadas por valoración

1. Publicación A (Valoración)
2. Publicación B (Valoración)
3. ...
- N. Publicación N (Valoración)

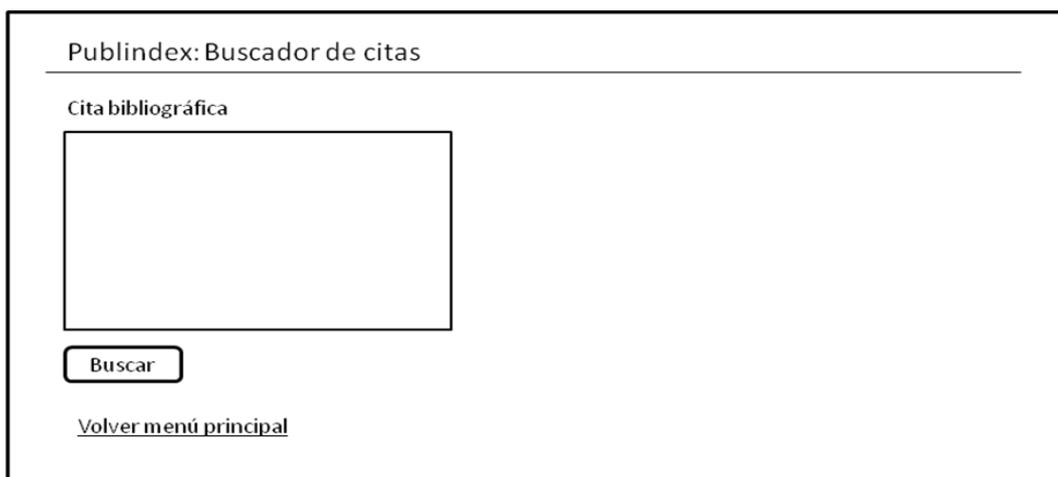
[Volver Inicio buscador](#) [Volver menú principal](#)

Figura 30: Página de datos de una categoría

En esta página se muestra el listado de enlaces a las publicaciones de una categoría, ordenadas decrecientemente según la valoración que presenten. La publicación desde la que se ha llegado a la categoría aparecerá resaltada.

B.5.6.3 Analizador de publicaciones a través de citas bibliográficas

Página de inicio



Publindex: Buscador de citas

Cita bibliográfica

[Volver menú principal](#)

Figura 31: Página de inicio del analizador de publicaciones.

Esta página presenta un formulario compuesto por un cuadro de texto donde introducir el listado de citas del cual consultar las publicaciones y de un botón de búsqueda que redirige a la página de resultados.

Página de resultado de búsqueda

Citas bibliográficas resultado

Se muestran las citas analizadas y la publicación a la que hacen referencia

Datos referencia	Publicación resultado
Cita 1	Publicación Cita 1
Cita 2	Publicación Cita 2
...	...
Cita N	Publicación Cita N

[Volver Inicio buscador](#) [Volver menú principal](#)

Figura 32: Página de resultado del análisis de publicaciones

Esta página se compondrá de una tabla de dos columnas. La columna de la izquierda contiene los datos de la cita bibliográfica y en la columna de la derecha un link a la publicación resultado del análisis de la cita y búsqueda de la publicación. Este enlace redirige a la página de datos y valoraciones de la publicación que corresponde al prototipado de la Figura 29.

B.5.6.4 Analizador de la calidad de un autor

Página inicial del analizador

Publindex: Buscador de publicaciones de un autor

Indique el nombre del autor del que buscar las referencias

Nombre de autor:

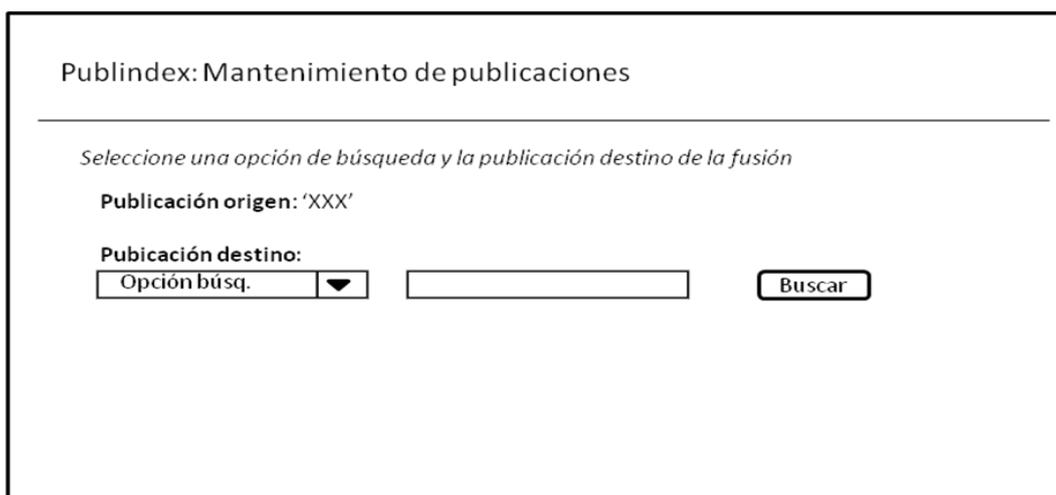
[Volver menú principal](#)

Figura 33: Página de inicio del analizador por autor

Esta página se compone de un formulario que contiene un cuadro de texto en el cual introducir el nombre del autor y el botón de búsqueda que redirigirá a la página de resultados cuyo prototipado corresponde a la Figura 32.

B.5.6.5 Mantenimiento de publicación

Búsqueda de la publicación resultado del mantenimiento



Publindex: Mantenimiento de publicaciones

Seleccione una opción de búsqueda y la publicación destino de la fusión

Publicación origen: 'XXX'

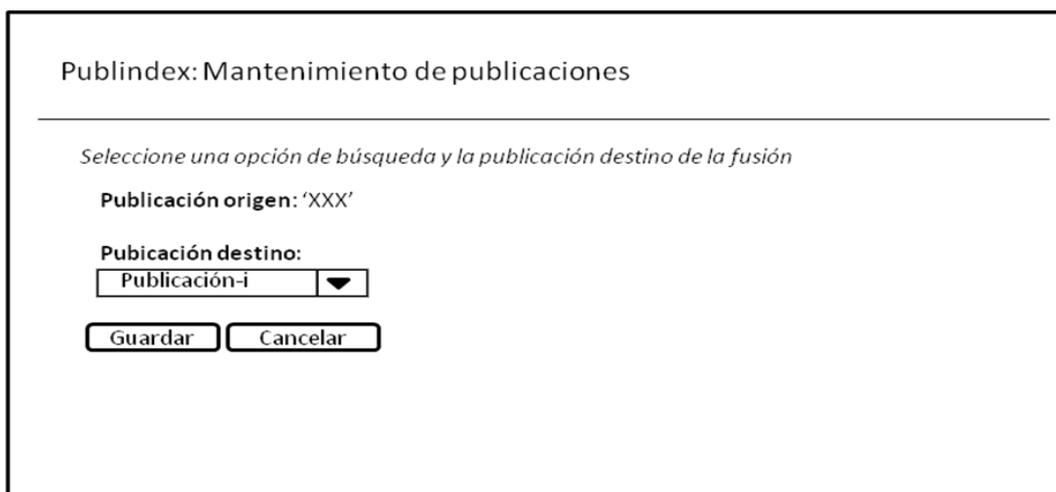
Publicación destino:

Opción búsq. ▼

Figura 34: Página de inicio del mantenimiento de publicaciones

La página se compone de un formulario en el cual se indica la publicación origen, que es aquella a la que pertenecía el enlace del cual se ha redirigido al mantenimiento, y un buscador de la publicación destino del mantenimiento. Al igual que en el analizador de una publicación, se puede seleccionar el criterio de búsqueda por nombre/siglas o ISSN.

Selección de la publicación destino



Publindex: Mantenimiento de publicaciones

Seleccione una opción de búsqueda y la publicación destino de la fusión

Publicación origen: 'XXX'

Publicación destino:

Publicación-i ▼

Figura 35: Página de selección de la publicación resultado

Esta pantalla presenta un desplegable con las publicaciones resultado de la búsqueda y los botones para aceptar o cancelar la operación de mantenimiento. Si se pulsa 'Guardar', se procederá a realizar la fusión entre las dos publicaciones. Si se pulsa 'Cancelar' se regresa a la página inicial del analizador de una publicación.

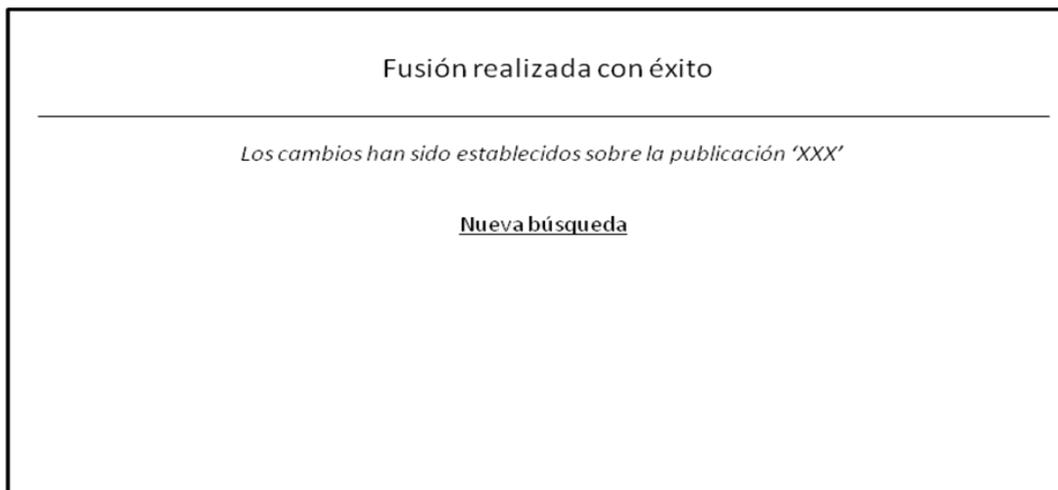
Mensaje de éxito de la operación

Figura 36: Mensaje de éxito de la operación

B.5.6.6 Mensajes de error

La interfaz de mensaje de error presentará la siguiente estructura, dependiendo del tipo de error presentará un título y mensaje de error diferente:



Figura 37: Pantalla de error

B.6 Segunda y tercera fase: Elaboración y construcción (Implementación)

La fase de elaboración también comprende parte del proceso de implementación, concretamente la implementación iterativa del núcleo de la arquitectura y solución de los riesgos mayores. Dentro de esta fase se llevaron a cabo las siguientes tareas:

- Definición e implementación de la arquitectura de la solución.
- Implementación de la base de datos.
- Decisiones de implementación relativas a cómo realizar la carga de los datos de los índices de calidad JCR, CORE y SJR.

Durante la fase de construcción se realizó una implementación iterativa de la funcionalidad definida para la aplicación y la carga de los índices de calidad. Cada iteración comprendía aproximadamente dos semanas de desarrollo, generando una nueva versión que incorporaba solución de fallos y/o nueva funcionalidad. Cabe destacar los siguientes hitos:

- 7 de Diciembre de 2010: inicio de la fase de construcción.
- 17 de Enero de 2010: carga completa de JCR.
- 2 de Marzo de 2010: primera versión estable del analizador de una publicación a través de su nombre/siglas o ISSN con los datos de JCR cargados. Se pone a disposición del usuario esta versión beta.
- 1 de Mayo de 2010: carga completa de CORE y nueva versión del analizador de una aplicación.
- 30 de Mayo de 2010: versión estable del subsistema de mantenimiento de publicaciones.
- 6 de Junio de 2010: carga completa de SJR.
- 25 de Junio: primera versión estable del analizador de una lista de publicaciones a través de una colección de citas bibliográficas.
- 10 de Julio de 2010: versiones estables de los analizadores de publicaciones por citas y autor.
- 1 de Agosto de 2010: versión definitiva de la aplicación Publindex.

Toda la información detallada puede consultarse en el documento anexo C de implementación del sistema.

B.7 Cuarta fase: Transición

La documentación del proyecto se compone de los siguientes textos:

- Memoria del proyecto: recoge un resumen de todo el proceso, explicando el análisis realizado a la vista de los objetivos, diseño de alto nivel, riesgos, conclusiones, etc. Su objetivo es dar una visión precisa pero no tediosa ni muy detallada del proyecto y su desarrollo al lector interesado.
- Documentación anexa: preparación y recopilación detallada en varios documentos de toda la documentación relativa a cada proceso de cierta relevancia del desarrollo. Por ejemplo, especificación de requisito, análisis, diseño o implementación.
- Manual de usuario: explica paso a paso cómo utilizar la funcionalidad que ofrece el proyecto.

La fase de transición comprendió toda la labor de elaboración de esta documentación.

B.8 Referencias

B.8.1 Referencias Web

[JCR]: http://en.wikipedia.org/wiki/Revista_Citation_Reports

[CORE]: <http://www.core.edu.au/>

[SJR]: <http://www.scimagojr.com/index.php>

[CSCR]: <http://www.cs-conference-ranking.org/conferencerankings/alltopics.html>

[MVC]: http://es.wikipedia.org/wiki/Modelo_Vista_Controlador

[DAO]: http://en.wikipedia.org/wiki/Data_access_object

[GRASP]: <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=grasp>

Todas las referencias web han sido probadas a fecha de 25/08/2010.

B.8.2 Referencias

[PU]: I. Jacobson, G. Booch, J. Rumbaugh (2000). *El proceso Unificado de desarrollo de Software*. Pearson Education.

[UMLPatterns]: Larman C. (2001). *Applying UML and patterns: An introduction to O-O design and iterative development*. Prentice Hall.

Anexo C: Implementación

Publindex: Análisis de la calidad de
publicaciones de investigación

Tabla de contenido

C.1	INTRODUCCIÓN	119
C.1.1	PROPÓSITO Y ALCANCE	119
C.1.2	REFERENCIAS	119
C.1.3	ACRÓNIMOS	119
C.1.4	RESUMEN.....	119
C.2	ESTRUCTURA DE DESARROLLO (FRAMEWORK)	121
C.3	DECISIONES DE IMPLEMENTACIÓN	123
C.3.1	CARGA DE DATOS	123
C.3.1.1	<i>Carga de datos de JCR.....</i>	<i>123</i>
C.3.1.2	<i>Carga de datos de SJR.....</i>	<i>123</i>
C.3.1.3	<i>Carga de datos de CORE.....</i>	<i>124</i>
C.3.2	ANALIZADOR DE LA CALIDAD DE PUBLICACIONES A TRAVÉS DE CITAS	125
C.3.2.1	<i>Analizador de citas bibliográficas.....</i>	<i>125</i>
C.3.2.2	<i>Búsqueda de la publicación referencia de la cita</i>	<i>127</i>
C.3.3	BUSCADOR DE LAS REFERENCIAS DE UN AUTOR	128
C.3.4	MANTENIMIENTO DE PUBLICACIONES.....	129
C.3.5	ACCESO A ESTADÍSTICAS DE LA PÁGINA WEB.....	129
C.4	BASE DE DATOS	133
C.4.1	TABLA PUBLICACIÓN.....	133
C.4.2	TABLA JOURNAL	133
C.4.3	TABLA CONFERENCIA.....	134
C.4.4	TABLA ÍNDICE	134
C.4.5	TABLA MACROCATEGORÍA	134
C.4.6	TABLA CATEGORÍA	134
C.4.7	TABLA PUBLICADOENCAT (VALORACIÓN).....	135
C.5	ARQUITECTURA DE LA SOLUCIÓN	137
C.6	CLASES DEL SISTEMA	139
C.6.1	CLASES DE LA CAPA DE INTERFAZ	139
C.6.1.1	<i>Analizador de una publicación</i>	<i>139</i>
C.6.1.2	<i>Analizador de un listado de publicaciones a través de citas.....</i>	<i>139</i>
C.6.1.3	<i>Buscador de publicaciones a través de autor.....</i>	<i>140</i>
C.6.1.4	<i>Mantenimiento de publicaciones</i>	<i>140</i>

C.6.1.5	<i>Páginas modales</i>	140
C.6.1.6	<i>Ficheros auxiliares</i>	140
C.6.2	CLASES DE LA CAPA DE LÓGICA (CONTROLADOR)	140
C.6.2.1	<i>Actions</i>	140
C.6.2.2	<i>ActionForms</i>	141
C.6.2.3	<i>POJOS</i>	142
C.6.2.4	<i>DAOS</i>	142
C.6.2.5	<i>Utilidades</i>	143
C.6.3	CLASES DEL MODELO DE DATOS	144
C.6.4	FICHEROS DE CONFIGURACIÓN.....	144
C.7	APARIENCIA DE LA APLICACIÓN	145
C.7.1	PÁGINA DE INICIO DEL ANALIZADOR DE UNA PUBLICACIÓN	145
C.7.2	PÁGINA DE RESULTADO DE LA BÚSQUEDA DE UNA PUBLICACIÓN	146
C.7.3	PÁGINA DE VALORACIONES DE UNA PUBLICACIÓN.....	146
C.7.4	PÁGINA DE RESULTADOS DE LA BÚSQUEDA DE LAS PUBLICACIONES A TRAVÉS DEL TRABAJO DE UN AUTOR. 147	
C.8	EVALUACIÓN DE LA IMPLEMENTACIÓN DEL EXTRACTOR DE PUBLICACIONES A TRAVÉS DE CITAS BIBLIOGRÁFICAS	149
C.8.1	RECONOCEDOR DE CITAS BIBLIOGRÁFICAS	149
C.8.2	RECONOCEDOR DE CITAS BIBLIOGRÁFICAS A TRAVÉS DE AUTOR	149
C.9	REFERENCIAS	153
C.9.1	REFERENCIAS WEB	153
C.9.2	REFERENCIAS	153

C.1 Introducción

La finalidad de este apartado es dar una visión general del documento, indicando al igual que en cualquier otro texto de este proyecto su objetivo, alcance, referencias, etc.

C.1.1 *Propósito y alcance*

Este documento tiene como propósito mostrar las principales decisiones de implementación que fueron tomadas durante el proceso de desarrollo del sistema. Además, muestra la estructura de clases final y la implementación de base de datos que tras las decisiones tomadas refinaron los modelos elaborados en la fase de diseño.

El alcance de este documento comprende toda la fase de implementación de la aplicación que fue realizado durante la elaboración y construcción del sistema final.

C.1.2 *Referencias*

Se realizan referencias a los documentos anexos de *Caso de estudio* y *Requisitos*.

C.1.3 *Acrónimos*

Se han utilizado los siguientes acrónimos a lo largo del documento:

- MVC: Model-View-Controller
- DAO: Data Access Object
- JSP: Java Server Pages
- CSS: Class Style Sheet
- SQL: Structured Query Language
- LALR: Look-Ahead LR Parser
- CSV: Comma-Separated Values
- JCR: Revista Citation Reports
- CORE: COmputing Research & Education
- SJR: Scientific Revista Rankings

C.1.4 *Resumen*

Este documento se compone de una breve introducción y de una descripción de los aspectos más relevantes en la fase de implementación: la estructura o framework de desarrollo, las principales y más importantes decisiones de implementación, la arquitectura de

la solución y la implementación de clases y base de datos del sistema. Finalmente, se presenta el listado de bibliografía consultada para la fase de estudio de la implementación.

C.2 Estructura de desarrollo (Framework)

Dadas sus múltiples ventajas, el framework que se ha utilizado para trabajar con Java ha sido Struts. Las ventajas son las siguientes:

- Intuitivo y centralizado control del flujo de la aplicación en un fichero de configuración, de manera que los cambios relacionados con el flujo sólo deben realizarse sobre dicho fichero.
- Posibilidad de implementar una aplicación en varios idiomas a través de ficheros de recursos.
- Separa la aplicación en tres capas independientes: el Modelo, la Vista y el Controlador (se ajusta al patrón MVC que se decidió utilizar en el diseño).
- Gestión sencilla de errores en formularios (campos vacíos, tipos de datos incorrectos, etc.)
- Se encarga del desarrollo de la capa de Vista y Control de la arquitectura, abstrayéndose de la implementación y desarrollo del modelo de datos.
- Distribución gratuita.

Struts proporciona una serie de tecnologías que serán utilizadas para el desarrollo de las aplicaciones web. Concretamente, el sistema usa:

- Actions: clases que separan la lógica de negocio de la petición del cliente, pudiendo llevar a cabo funciones antes de llamar a negocio para que realice las operaciones necesarias (logging, validación de sesión...).

Publindex presenta un conjunto de clases que heredan de Action que, sobrescribiendo la función *execute*, se encargan de llamar a la lógica y de redirigir hacia un sitio u otro.

Por ejemplo, cuando se introduce el nombre de una publicación y se pulsa buscar, una clase Action, definida en el fichero de configuración, será la encargada de determinar qué tipo de búsqueda se pide (nombre o ISSN) y así llamar a un método u otro de negocio para realizar la búsqueda de coincidencias. En caso de devolver datos, redirige a la página del listado de resultados y sino a la página de error.

- ActionMappings: clase que representa una acción de mapeado en el fichero de configuración de Struts. Indica al controlador de flujo que clase Action debe llamarse según la petición realizada.
- ActionForward: clase que permite indicar a qué página destino debe redirigirse el flujo una vez que un Action se ha completado. Estas clases están definidas en el fichero de configuración y se incluyen dentro de las ActionMapping, pudiendo definirse más de una en el caso de que el flujo tenga varias direcciones por tomar.
- ActionForms: clases que permiten llevar a cabo labores de validación de formularios, de manera que, en el caso de error, se impide la llamada a la Action

oportuna y puedan mostrarse errores al usuario. Por ejemplo, en el caso de que no se introduzca un nombre en el analizador de publicaciones, no se realiza la búsqueda de coincidencias y se redirige al usuario a la misma página de inicio mostrando un mensaje de error de campo vacío.

En cuanto a la capa de Vista del modelo, se utilizaron tecnologías permitidas por Struts:

- Páginas JSP para la implementación de las interfaces de la aplicación.
- JavaScript para mostrar información de manera dinámica en las páginas web y ejecutarse desde el lado de cliente.
- CSS para definir los estilos utilizados en las páginas.

Todo el conjunto de clases Action, ActionForm, páginas JSP, ficheros de configuración de Struts, etc. serán enumeradas y descritas junto con el resto de clases en el apartado C.6 de clases del sistema [Struts].

C.3 Decisiones de implementación

Este apartado reúne las principales decisiones que tuvieron que tomarse a nivel de implementación y que influyeron en el modelo de clases y estructura final del sistema. Todas estas decisiones no influyen a nivel de análisis y diseño, de manera que si hubieran sido tomadas otras alternativas de implementación estos modelos no se verían afectados.

C.3.1 Carga de datos

La carga de datos fue una de las tareas más importantes durante el proceso de implementación, ya que los analizadores precisaban de tener información almacenada para poder realizar sus operaciones y consultas. El principal problema era que cada índice de calidad presentaba sus datos en un formato determinado, teniendo que considerar varios procedimientos de carga de los índices.

C.3.1.1 Carga de datos de JCR

El formato que presentaban los datos de JCR era Excel. Para cada una de las categorías y en un año concreto, se presentaba un fichero que contenía todas las revistas pertenecientes junto con los datos relativos al factor de impacto, ISSN, lenguaje, país y más datos que no eran necesarios. Para realizar la carga de los datos de una categoría los pasos a seguir son los siguientes:

1. Manipular el fichero Excel para dejar todos los datos ordenados decrecientemente según el factor de impacto y sólo dejar la información importante. Por cada fila deben aparecer los siguientes datos por columnas en este orden: título, ISSN, factor de impacto, país, lenguaje y categoría.
2. Guardar el fichero en formato CSV.
3. Procesar el fichero CSV a través de Java e ir guardando cada registro de éste.

Indicar que los datos de las categorías y macrocategorías pertenecientes a JCR ya deben estar almacenados en la base de datos.

C.3.1.2 Carga de datos de SJR

SJR también presentaba los datos almacenados por categoría y año en formato Excel. Los pasos a realizar son:

1. Manipular el fichero Excel para dejar todos los datos ordenados decrecientemente según el índice SJR y por cada fila presentar los siguientes datos en este orden: título, ISSN, índice de calidad y país.
2. Guardar el fichero en formato CSV.

3. Procesar el fichero CSV a través de Java.

Igual que JCR, los datos de las categorías y macrocategorías deben estar previamente cargados en el sistema.

C.3.1.3 Carga de datos de CORE

Cargar los datos relativos a CORE supuso un mayor problema, ya que el formato de los ficheros de los que cargar los datos es PDF. Estos ficheros no podían ser procesados directamente, por lo que debieron ser convertidos a formato XML que sí puede ser procesado. Como XML es un lenguaje que permite representar la información a través de etiquetas de manera estructurada, tuvo que implementarse un mecanismo de reconocimiento y procesamiento de ésta información para poder almacenarla en base de datos. Para ello, se implementó un reconocedor léxico y sintáctico a través de JFlex y CUP. Los pasos a seguir para almacenar la información son:

1. Guardar el fichero PDF en formato XML.
2. Modificar el fichero XML de manera que sólo quede la información relativa a las valoraciones. Cada entrada del fichero viene encapsulada en un registro TR de un Table. A continuación se presenta el formato que siguen los datos:

```
<Table>
...
  <TR>
    <TD>Info no relevante</TD>
    <TD>Valoración</TD>
    <TD>Título publicación</TD>
    <TD>ISSN (Si es revista) </TD>
    <TD>Código de categoría</TD>
  </TR>
...
</Table>
```

3. Debido a que en estos ficheros no se indica qué tipo de publicación se está tratando o año de la valoración, es necesario añadir una información en el fichero XML que represente estos datos:

```
<Categoria>
  <Fuente>CORE</Fuente>
  <Anyo>Año</Anyo>
</Categoria>
<Publicacion>Tipo de publicación<Publicacion>
<Table>
```

Información de las valoraciones
</Table>

4. Procesar el fichero a través del analizador, que guardará la información del índice.

Para poder procesar la información correctamente es necesario guardar información de las categorías CORE, ya que en el fichero XML por cada registro se indica el código de la categoría a la que pertenece la valoración.

Sin embargo, no todos los ficheros de valoraciones CORE se presentaban en este formato. En una ocasión los ficheros PDF no pudieron ser convertidos a XML, por lo que fue necesario extraer el texto del PDF e implementar un analizador que reconociera esta información para poder almacenarla. Las valoraciones CORE del año 2010 se presentaron en formato Excel, teniendo que extraer la información de la misma manera que para JCR o SJR.

C.3.2 Analizador de la calidad de publicaciones a través de citas

A continuación se detallan las decisiones que tuvieron que ser tomadas para implementar la funcionalidad del buscador de publicaciones a través de citas bibliográficas.

C.3.2.1 Analizador de citas bibliográficas

Con objeto de poder extraer la información relativa a la publicación a la que hacía referencia una cita, fue necesario implementar algún mecanismo de reconocimiento. Se barajaron las siguientes posibilidades:

- Usar la clase StringTokenizer de Java para detectar patrones de cadenas.
- Separar la cita en subcadenas e identificar qué hacía referencia a la publicación.
- Presentar las citas en formato XML y recuperar la información usando XPATH.
- Implementar un reconocedor léxico y sintáctico que procese la información y recupere la relativa a la publicación.

Los principales requisitos de implementación son conseguir un sistema flexible ante cambios y generar un código que sea entendible y modificable por otras personas aparte del autor. Dado que las tres primeras opciones suponían generar un código bastante complejo, se tomó la decisión de implementar el reconocedor léxico y sintáctico utilizando las herramientas JFlex y CUP.

Formatos reconocidos

El analizador implementado permite reconocer citas bibliográficas en formato texto y BibTex que presenten una estructura de formato concreta. La estructura en formato que es procesada por el analizador es el siguiente:

[Autores], "[Título cita]", [Título publicación], [Resto de información]

En el resto de información se indica el año de la cita, el ISSN u otra información no relevante, separada por comas. Cabe destacar que si la cita posee más de un autor, se presentarán los nombres separados también por comas. Un ejemplo de cita en formato texto es la siguiente:

O. Urra, S. Ilarri and E. Mena, "Agents Jumping in the Air: Dream or Reality?", 10th International Work-Conference on Artificial Neural Networks (IWANN'09), Special Session on Practical Applications of Agents and Multi-Agent Systems, Salamanca (Spain), Springer Verlag, ISSN 0302-9743, ISBN 978-3-642-02477-1, volume 5517, pp. 627-634, June 2009.

En cuanto al formato BibText, hay que destacar que dependiendo del tipo de evento en el que se publique, el formato que se presenta es diferente y por tanto, el analizador deberá reconocer unos patrones concretos. En concreto, el analizador reconoce los formatos BibText: @INPROCEEDINGS y @ARTICLE. A continuación se muestran unos ejemplos:

```
@INPROCEEDINGS{iwann09-Urra,
author={O. Urra and S. Ilarri and E. Mena},
booktitle={10th International Work-Conference on Artificial Neural Networks (IWANN'09), Special
Session on Practical Applications of Agents and Multi-Agent Systems, Salamanca (Spain)},
month={June},
pages={627--634},
publisher={Springer Verlag, ISSN 0302-9743, ISBN 978-3-642-02477-1},
series={Lecture Notes in Computer Science},
title={Agents Jumping in the Air: Dream or Reality?},
volume={5517},
year={2009},
idbdi={yes},
file={iwann09.pdf},
keywords={Mobile Agents},
}
```

```
@ARTICLE{is08,
author={S. Ilarri and E. Mena and A. Illarramendi},
journal={Information Sciences, ISSN 0020-0255},
month={May},
number={9},
pages={2105--2127},
publisher={Elsevier, doi:"10.1016/j.ins.2007.12.015"},
title={Using Cooperative Mobile Agents to Monitor Distributed and Dynamic Environments},
volume={178},
year={2008},
idbdi={yes},
```

```
file={is08.pdf},  
keywords={Mobile Computing, ANTARCTICA, Mobile Agents, Location-dependent queries},  
}
```

Para más información, consultar la referencia web [WikiBibTex].

Problemas sintácticos

Aunque el analizador que se ha implementado es bastante flexible en cuanto al orden en que aparezcan los patrones y caracteres reconocidos, no presenta un 100% de éxito ni acepta todo tipo de formatos. Los problemas encontrados han sido:

- En el formato texto de las citas los autores, título de la cita y título de la publicación no pueden cambiarse de orden, ya que si se cambiaran no habría manera de reconocer concretamente que patrón es el título de la publicación.
- En el formato BibTex los caracteres '{' y '}' son especiales, por tanto no se pueden reconocer aquellas citas que contienen esos caracteres en la información (título de la cita, publicación, año...).
- El título de la publicación presenta en ocasiones comas que el analizador interpreta como separador del título y otra información, por lo que en esos casos el título de la publicación extraído no es completo. Esto implica que la búsqueda de resultados de las publicaciones sea menos precisa al buscar una subcadena más pequeña en las coincidencias.

Para comprobar la precisión y cómo de bueno es el reconocedor se realizaron una serie de pruebas, analizando un conjunto de citas en formato BibText y texto. Este estudio se presenta con más detalle en el apartado número 5 de evolución del prototipo.

C.3.2.2 Búsqueda de la publicación referencia de la cita

Una vez extraído el título de la publicación de la cita, es necesario buscar la publicación en la base de datos para su posterior análisis. En el caso de búsqueda por título o siglas se realiza la búsqueda de coincidencias. Puede ocurrir que los resultados obtenidos sean más de uno e incluso tener un listado grande de coincidencias. Además, el resultado de la búsqueda debería dar como resultado aquella publicación que más se asemeje en título o siglas a los extraídos de la cita.

Esta situación hizo que fuera necesario implementarse algún mecanismo que permitiera identificar en los resultados aquella publicación a la que realmente se hace referencia. Para ello se decidió estudiar qué porcentaje de similitud poseían los títulos con el nombre extraído de la cita utilizando la distancia lexicográfica. Esta función es facilitada por una librería de Java, *simmetrics*, que ofrece métodos que calculan la semejanza entre dos cadenas. El método que se utilizó fue *getSimilarity*, que acepta dos cadenas como parámetro y devuelve el porcentaje de similitud en tanto por uno de ambas cadenas.

Para saber cuál de los elementos de la lista es el que más se asemeja, es necesario ordenar sus elementos en orden creciente en función del porcentaje de similitud. Ordenar una lista de diez elementos no supone un coste muy elevado, pero sin embargo ordenar una de seiscientos elementos sí, por lo que fue necesario también utilizar criterios de ordenación.

Para resolver este problema, se implementó una clase que presentara criterios de ordenación de conjuntos de elementos usando como función de ordenación *getSimilarity*. Dicha clase, presenta los siguientes métodos de ordenación:

- Método Burbuja de ordenación: consiste en ordenar el vector comparando elementos adyacentes entre sí e intercambiarlos hasta que todos se encuentran ordenados. Este método en el caso de listas grandes, es muy deficiente, presentando un coste $O(n^2)$; Sin embargo, en el caso de listas pequeñas o parcialmente ordenadas es uno de los métodos más utilizados.
- Método de ordenación *Quicksort*: método de ordenación que a través de un elemento pivote divide el vector en trozos para ordenar cada uno de manera independiente, teniendo un coste medio de ordenación $O(n \log n)$. En el caso de listas de tamaño considerable y no ordenadas es uno de los métodos más eficientes [QuickSort].

Una vez que se realiza la búsqueda de coincidencias, se utilizan estos métodos para ordenar los resultados. Al completarse la ordenación, se selecciona la publicación que se encuentra en la última posición de la lista, es decir, la que presenta mayor porcentaje de similitud.

C.3.3 Buscador de las referencias de un autor

Una de las decisiones más relevantes que se tomó fue la obtención de manera online de las referencias bibliográficas del trabajo de un autor. El objetivo de esta decisión fue incorporar una funcionalidad nueva al sistema, ahorrar costes de tiempo en acceso a datos y evitar el tiempo que hubiera sido necesario invertir cargando la información de referencias y autores en el sistema.

La búsqueda online se limitó a un grupo concreto de personas. Publindex sólo va a poder facilitar información relativa al personal investigador del grupo de investigación de Sistemas de Información Distribuidos [SID].

La búsqueda online se compone de los siguientes pasos:

1. Se obtiene el código HTML de la página web que muestre las referencias bibliográficas del autor. Para descargar el código es necesario conocer la URL de la página. La URL presenta siempre la misma estructura indicando en el parámetro *authors* el nombre del autor al que pertenecen las referencias mostradas en la página.
2. Se reconoce y procesa el código HTML a través de un analizador léxico y sintáctico, obteniendo así el listado de las citas.

3. Una vez obtenidas las citas, se reconocen para obtener la información de las publicaciones a las que hacen referencia.
4. Se redirige al analizador de un listado de publicaciones para mostrar el listado de citas del autor y los eventos donde se publicaron los trabajos.

La búsqueda podría ampliarse añadiendo nuevas páginas de autores en las que buscar. Esto implicaría modificar el analizador de código HTML para reconocer el formato de estas nuevas páginas y procesar la información de las citas.

C.3.4 Mantenimiento de publicaciones

En un primer momento, los cambios en los datos producidos por la fusión de publicaciones se perdían si se restauraba la base de datos a su estado inicial o cargando un script que no tuviera los últimos cambios registrados. Esto hizo tomar la decisión de que se debían de registrar los cambios en un fichero de log. Este fichero almacena todas las operaciones que se realizaron sobre el modelo físico.

Para recuperar los cambios lo más rápido y cómodamente posible, el fichero de log es un fichero SQL que contiene todas las sentencias que se realizan sobre base de datos para realizar el mantenimiento, de manera que para volver al estado anterior al reinicio sólo es necesario ejecutar dicho script. Se implementó una clase auxiliar de nombre *LogUtils* que presenta métodos para la edición del fichero de log.

Así pues, cada vez que se realiza una fusión, el subsistema de mantenimiento también añade los cambios al fichero de log. Este fichero presenta el nombre de *FusionLog.sql* y se encuentra en una carpeta que recibe el nombre de Publindex en el directorio raíz donde se encuentra la aplicación ejecutándose.

C.3.5 Acceso a estadísticas de la página web

La aplicación debe facilitar una serie de datos estadísticos, tales como el número de visitas recibidas en el día, media diaria, últimos visitantes, estimaciones... Se decidió implementar esta funcionalidad a través de las estadísticas gratuitas de *Motigo Webstats*. Esta web proporciona contadores web gratuitos, de modo que añadiendo un pequeño fragmento de código HTML a la página web que se desea que muestre la información, ya se tiene acceso a estas estadísticas. Publindex presenta en las páginas de inicio de los analizadores y en el menú principal acceso a ellas:

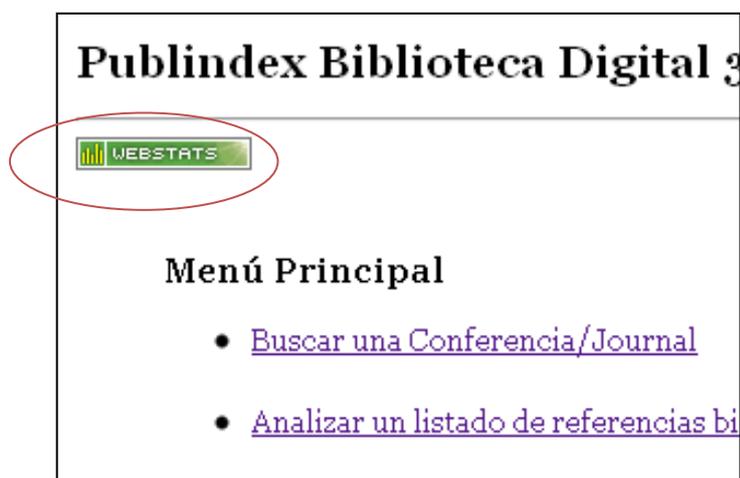


Figura 1: Ubicación de las estadísticas en la aplicación

Si se pincha sobre este icono verde, se accede directamente a la página informativa de las estadísticas relativas a la URL de acceso a la aplicación. La siguiente captura de pantalla muestra cómo se presentan estas estadísticas:

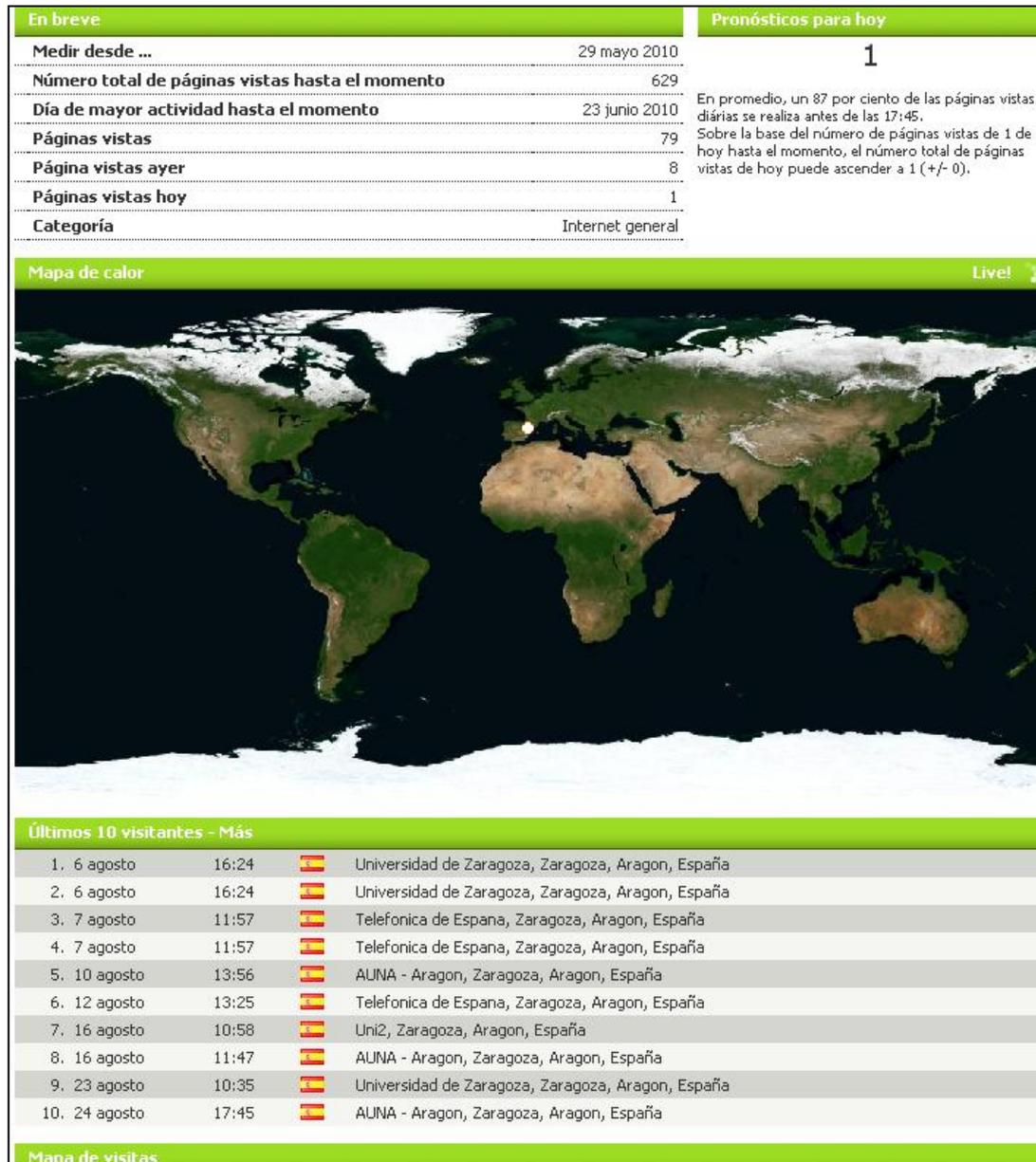


Figura 2: Estadísticas de Publindex

Dentro de estas estadísticas destacan el número total de visitas que ha recibido la página web, el día de mayor actividad, los pronósticos para el día de hoy o los últimos 10 visitantes. Desde el día 29 de Mayo de 2010 que se incorporaron las estadísticas, se han recibido un número total de 629 visitas, verificándose así que la aplicación ha sido utilizada antes de disponer de la versión final de la aplicación.

C.4 Base de datos

El modelo que finalmente se implementó de la base de datos es diferente al presentado en el diseño, debido a las decisiones tomadas relativas a adquirir online la información de los autores. Esto hizo que las entidades de autor y cita desaparecieran, ya que no es necesario guardar información de ellas al no recuperarse del modelo físico.

Por otro lado, se decidió utilizar Hibernate para realizar el mapeo de las entidades de negocio o POJOS con las tablas en base de datos. Esto permite realizar consultas en lenguaje HQL mucho más intuitivo y abstrayéndonos de la representación física. Cada clase POJO está mapeada con su correspondiente tabla en BBDD a través de una serie de ficheros de configuración.

A continuación se presentan las tablas junto con sus claves primarias, campos y claves extranjeras para indicar las relaciones entre ellas.

C.4.1 Tabla publicación

Campo	Tipo	Nulo	Descripción
id_publicacion	Int	No	Clave primaria de la publicación. Se autoincrementa cada vez que se almacena un nuevo registro.
siglas	Varchar(20)	Si	Campo que identifica las siglas de la publicación, en el caso de que tenga.
nombre	Varchar(300)	No	Campo que identifica el nombre de la publicación.

Tabla 1: Campos de la tabla publicación en base de datos

C.4.2 Tabla journal

Campo	Tipo	Nulo	Descripción
id_publicacion	Int	No	Clave ajena perteneciente a la tabla publicación.
pais	Varchar(20)	Si	Campo que identifica el país al que pertenece la revista.
lenguaje	Varchar(20)	Si	Campo que identifica el lenguaje en el que está escrito la revista
ISSN	Varchar(10)	Si	Campo que identifica el ISSN de la revista.

Tabla 2: Campos de la tabla revista en base de datos

C.4.3 Tabla conferencia

Campo	Tipo	Nulo	Descripción
id_publicacion	Int	No	Clave ajena perteneciente a la tabla publicación.

Tabla 3: Campos de la tabla conferencia en base de datos

C.4.4 Tabla índice

Campo	Tipo	Nulo	Descripción
id_indice	Int	No	Clave primaria del índice. Se autoincrementa cada vez que se almacena un nuevo registro.
nombre	Varchar(15)	No	Campo que identifica el nombre del índice.
descripción	Varchar(15)	No	Campo que identifica la descripción del índice.

Tabla 4: Campos de la tabla índice en base de datos

C.4.5 Tabla macrocategoría

Campo	Tipo	Nulo	Descripción
id_macrocategoria	Int	No	Clave primaria de la macrocategoría. Se autoincrementa cada vez que se almacena un nuevo registro.
nombre	Varchar(300)	No	Campo que identifica el nombre de la macrocategoría.
código	Varchar(30)	Si	Campo que identifica el código de la macrocategoría.
búsqueda	Tinyint(1)	No	Campo que indica si la macrocategoría está incluida en el conjunto de búsqueda actual.
Fuente	Varchar(30)	No	Campo que indica la fuente a la que pertenece la macrocategoría.
defecto	Tinyint(1)	No	Campo que indica si la macrocategoría está seleccionada por defecto en la búsqueda.

Tabla 5: Campos de la tabla macrocategoría en base de datos

C.4.6 Tabla categoría

Campo	Tipo	Nulo	Descripción
Id_categoria	Int	No	Clave primaria de la categoría. Se

			autoincrementa cada vez que se almacena un nuevo registro.
id_macrocategoria	Int	Si	Clave ajena perteneciente a la tabla macrocategoria en la que está incluida.
Id_indice	Int	No	Clave ajena perteneciente a la tabla índice a la que pertenece.
nombre	Varchar(100)	No	Campo que identifica el nombre de la categoría.
ano	Int	No	Campo que identifica el año de la categoría.
fuelle	Varchar(10)	No	Campo que indica la fuente de la categoría.
codigo	Varchar(10)	Si	Campo que indica el código de la categoría.

Tabla 6: Campos de la tabla categoría en base de datos

C.4.7 Tabla publicadoencat (valoración)

Campo	Tipo	Nulo	Descripción
Id_publicada	Int	No	Clave primaria de la valoración. Se autoincrementa cada vez que se almacena un nuevo registro.
Id_categoria	Int	No	Clave ajena de la categoría en la que se ha dado una valoración a la publicación.
id_publicacion	Int	No	Clave ajena de la publicación de la que se da la valoración.
posicion	Int	No	Campo que identifica la posición de la publicación en la categoría según la valoración.
valoracion	Varchar(20)	No	Campo que indica la valoración asignada a la publicación.
fi	Double	Si	Campo que indica la valoración en formato numérico.

Tabla 7: Campos de la tabla publicadoencat (valoración) en base de datos

El esquema que muestra gráficamente la base de datos con las tablas y relaciones entre ellas a través de claves extranjeras:

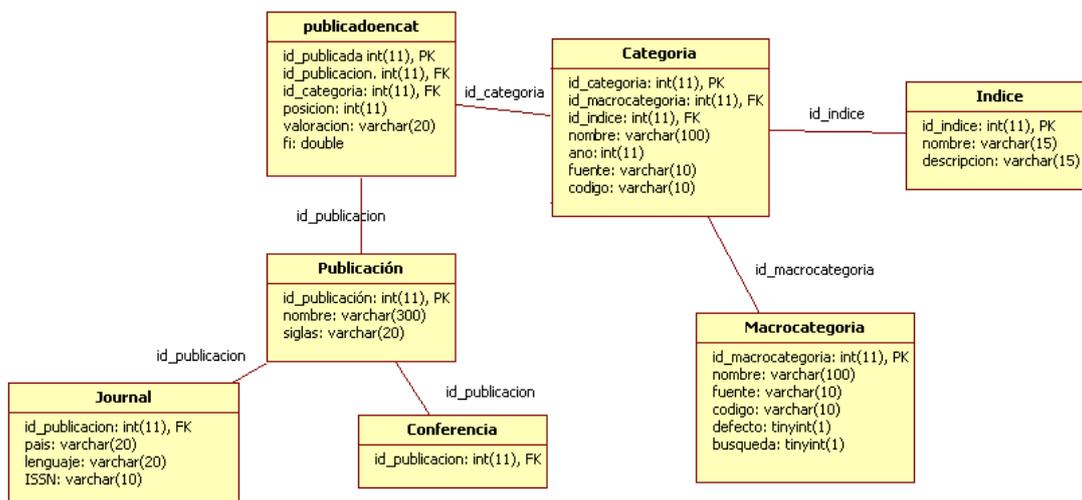


Figura 3: Base de datos implementada

Como puede verse, existe una tabla, *publicadoencat*, para almacenar la valoración y posición que presenta una publicación en una categoría. Esto es debido a que en el diseño la relación entre publicación y categoría era N-N, por lo que para implementarla era necesario añadir una tabla que registrara las ocurrencias de publicación-categoría. La tabla *journal* y *conferencia* heredan de *publicación* a través de su *id* como la clave ajena.

La tabla *macrocategoría* contiene unos campos booleanos (tipo *tinyint*) para indicar cuáles de ellas intervienen en la búsqueda de publicaciones de los analizadores de calidad. En el caso de que estos campos tengan valor falso, las categorías que están asociadas a estas *macrocategorías* no serán incluidas en la búsqueda de coincidencias de publicaciones y sus valoraciones.

Notar que esta base de datos no contendrá todas las publicaciones de los índices de calidad, ya que no todas las categorías de JCR, SJR y CORE han sido cargadas para todos los años. Por ejemplo, de JCR sólo están almacenadas las valoraciones, categorías y publicaciones relativas al área de informática (ver anexo B de caso de estudio).

C.5 Arquitectura de la solución

La arquitectura final del sistema está estructurada en tres capas tal y como presenta Struts en su definición y como se estableció en el proceso de diseño: el modelo, la vista y el controlador.

La vista y el controlador son las capas que contienen las clases implementadas en el sistema. Por un lado, la vista contiene las páginas JSP que conforman la interfaz de la aplicación como además las hojas de estilo CSS y los ficheros JavaScript que las cumplimentan. El controlador contiene todas las clases que conforman la lógica de la aplicación, desde las clases DAO, POJOS o entidades de negocio y aquellas propias de Struts (Actions y ActionForms).

La capa de modelo hace referencia a la representación física de los datos, como además los ficheros de configuración Hibernate que enlazan las entidades de negocio con las tablas de base de datos.

A continuación se muestra la arquitectura que presenta el sistema, agrupando las clases según su funcionalidad:

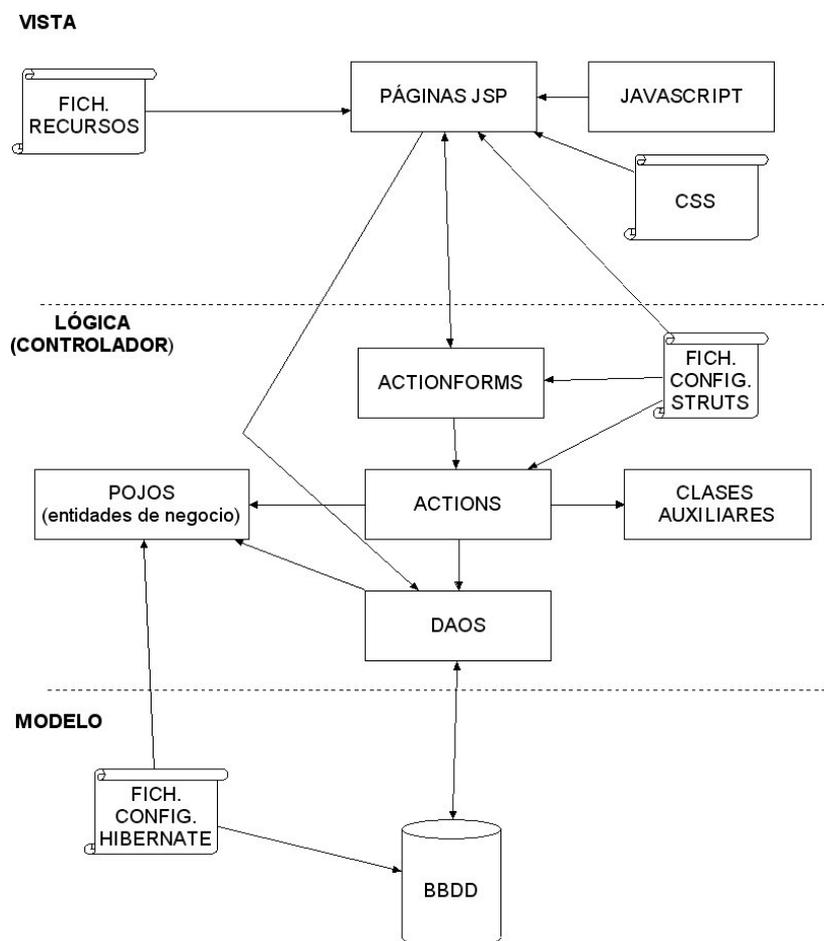


Figura 4: Arquitectura de la solución

Dentro del conjunto de clases auxiliares se encuentran todas aquellas que sean necesarias para realizar tareas cuya implementación sea significativa. Por ejemplo, los analizadores de autores y citas o la clase que contiene las funciones de ordenación.

C.6 Clases del sistema

Este apartado describe el conjunto de clases del sistema, junto con una descripción de la funcionalidad que implementan en el sistema. Las clases serán agrupadas según si pertenecen a la capa de vista, modelo o lógica.

C.6.1 Clases de la capa de interfaz

Contiene las páginas JSP que implementan las interfaces de la aplicación. Mostramos las interfaces agrupadas dependiendo del subsistema al que pertenezcan.

La página de inicio, [index.jsp](#), presenta un menú principal a través del cual se podrá acceder a todas las opciones de búsqueda de la aplicación.

C.6.1.1 Analizador de una publicación

- [indexBuscador.jsp](#): página de inicio del buscador de una publicación donde introducir los criterios de búsqueda.
- [muestraCoincidencias.jsp](#): página que muestra los resultados de la búsqueda.
- [muestraDatosJournal.jsp](#): página que muestra los datos y valoraciones de una publicación de tipo revista.
- [muestraDatosConferencia.jsp](#): página que muestra los datos y valoraciones de una publicación de tipo conferencia.
- [muestraDatosCategoriaJCR.jsp](#): página que muestra las publicaciones de una categoría perteneciente al índice JCR en un año en concreto.
- [muestraDatosCategoriaSJR.jsp](#): página que muestra las publicaciones de una categoría perteneciente al índice SJR en un año en concreto.
- [muestraDatosCategoriaCORE.jsp](#): página que muestra las publicaciones de una categoría perteneciente al índice CORE en un año en concreto.

C.6.1.2 Analizador de un listado de publicaciones a través de citas

- [indexCitas.jsp](#): página de inicio del buscador de una lista de publicaciones.
- [muestraResultadosCitas.jsp](#): página que muestra el listado de citas con las publicaciones resultado de la búsqueda.

El resto de páginas ya han sido definidas en la subsección anterior.

C.6.1.3 Buscador de publicaciones a través de autor

- [indexAutor](#): página de inicio del buscador de una lista de publicaciones a través del autor.

El resto de páginas ya han sido definidas en la subsección anterior.

C.6.1.4 Mantenimiento de publicaciones

- [mantenimientoPublindex.jsp](#): página de inicio del subsistema de mantenimiento.
- [seleccionarPublicacionDestino.jsp](#): página de selección la publicación destino de la fusión.

C.6.1.5 Páginas modales

- [error.jsp](#): página que muestra los mensajes de error al no encontrar una publicación, al introducir una cita no válida o al no encontrar información del autor.
- [errorInterno.jsp](#): página que muestra los mensajes de error al ocurrir una excepción no controlada.

C.6.1.6 Ficheros auxiliares

Las páginas JSP también hacen uso de las siguientes herramientas:

- [utilesPresentacion.js](#): fichero que contiene algunas funciones JavaScript que se ejecutan en algunas páginas ante ciertos eventos, como por ejemplo al cambiar la selección de un desplegable mostrar un mensaje informativo.
- [SID.css](#): contiene las clases CSS para dar apariencia a la página.
- [MessageResources.properties](#): el fichero de recursos que utilizan las páginas. En el caso de querer añadir la funcionalidad de ser una aplicación multi idioma, se presentaría un fichero de recursos por cada lenguaje permitido.

C.6.2 Clases de la capa de lógica (controlador)

La capa de lógica presenta la mayor parte de las clases del sistema, agrupándolas por los paquetes declarados en el sistema:

C.6.2.1 Actions

Estas clases heredan de la clase Action de Struts, implementando todas ellas el método *execute* (*ActionMapping*, *ActionForm*, *HttpServletRequest*, *HttpServletResponse*):

- [AnalizarCita.java](#): Se encarga de obtener del formulario las citas introducidas por el usuario y llamar al reconocedor para procesar la información. En el caso de que las citas no sean válidas, redirigirá a la página de error.

- BuscarCitasAutor.java: Se encarga de obtener las citas de un autor y analizarlas para obtener los datos de las publicaciones. En el caso de que el autor no exista o no pueda ser procesada la información, se redirige a la página de error.
- BusquedaCategoriaAction.java: Se encarga de obtener las publicaciones que forman parte de la categoría en el año seleccionado.
- BusquedaConferenciaAction.java: Se encarga de buscar los datos de la conferencia como además todas las valoraciones asignadas y almacenadas en el sistema.
- BusquedaJournalAction.java: Se encarga de buscar los datos de la revista como además todas las valoraciones asignadas y almacenadas en el sistema.
- BusquedaPublicacionFusionAction.java: Realiza la búsqueda de la publicación destino de la fusión en el mantenimiento de publicaciones. Buscará las coincidencias en el sistema para el nombre/siglas o ISSN introducidos por el usuario.
- RealizarFusionAction.java: se encarga de fusionar las publicaciones seleccionadas por el usuario, de manera que la publicación destino presenta los datos de las dos publicaciones antiguas. También se encarga de modificar el fichero de log.
- SeleccionarFiltroAnyos.java: obtiene las valoraciones de la publicación en el rango de años seleccionado en el filtro.
- SeleccionarTipoBusqueda.java: se encarga de realizar la búsqueda de coincidencias de publicaciones por nombre/siglas o ISSN según la opción seleccionada en el formulario de la página de inicio de búsqueda.

C.6.2.2 ActionForms

Estas clases heredan de `ActionForm` y se encargan de definir los campos del formulario al cual hacen referencia junto con sus métodos `get` y `set`. Estas clases implementan el método `validate` (`ActionMapping`, `HttpServletRequest`) con objeto de comprobar que se han introducido correctamente los métodos en el formulario.

- AutorForm.java: representación del formulario de búsqueda del trabajo de un autor.
- BusquedaPublicacionFusionForm.java: formulario de búsqueda de la publicación destino de la fusión en el mantenimiento de publicaciones.
- CitaForm.java: formulario en el cual el usuario introduce las citas bibliográficas para obtener el listado de publicaciones y poder analizarlas.
- PublicaciónDestionFusionForm.java: formulario en el cual el usuario selecciona la publicación destino de la fusión de un desplegable donde aparecen todos los resultados de la búsqueda de dicha publicación.
- SeleccionarFiltroAnyosForm.java: formulario que representa el filtro de años en el cual buscar las valoraciones a mostrar de la publicación a la que pertenecen los resultados.

- SeleccionarTipoBusquedaForm.java: formulario de búsqueda de la publicación a analizar a partir del nombre/siglas o ISSN.

C.6.2.3 POJOS

Estas clases hacen referencia a las entidades de negocio que serán los datos a intercambiar entre las capas de la aplicación. Cada clase es mapeada con una tabla de base de datos a través de Hibernate. Las clases POJO son las siguientes:

- Categoria.java: Entidad de negocio Categoría.
- Publicacion.java: Entidad de negocio Publicación.
- Journal.java: Entidad de negocio Journal. Hereda de Publicacion.
- Conferencia.java: Entidad de negocio Conferencia. Hereda de Publicación.
- Indice.java: Entidad de negocio Índice.
- Macrocategoria.java: Entidad de negocio Macrocategoria.
- Publicadas.java: Entidad de negocio para representar la valoración de un objeto Publicación con uno Categoría.

C.6.2.4 DAOS

Estas clases son las encargadas de abstraer a la vista y el controlador del modelo físico de datos. Se presenta una clase DAO por tabla de base de datos y ofrece un conjunto de métodos para editar y consultar la información almacenada en ella. Los parámetros y resultados se presentan en POJOS o colecciones de éstos.

Al igual que sucedió con la implementación de base de datos, las decisiones de implementación tomadas hicieron que no se presentaran clases DAO para las citas y los autores, ya que no existen tablas en el modelo de datos para ellos.

- CategoriaDAO.java: Contiene todas las operaciones de acceso a datos para la entidad Categoría.
- ConferenciaDAO.java: Contiene todas las operaciones de acceso a datos para la entidad Conferencia.
- IndiceDAO.java: Contiene todas las operaciones de acceso a datos para la entidad Índice.
- JournalDAO.java: Contiene todas las operaciones de acceso a datos para la entidad Journal.
- MacrocategoriaDAO.java: Contiene todas las operaciones de acceso a datos para la entidad Macrocategoria.
- PublicacionDAO.java: Contiene todas las operaciones de acceso a datos para la entidad Publicación.
- PublicadasDAO.java: Contiene todas las operaciones de acceso a datos para entidad Publicadas (Valoración).

C.6.2.5 Utilidades

Dentro de esta subsección se presentan todas aquellas clases auxiliares que utilizan las clases definidas previamente para implementar tareas de cierta complejidad como por ejemplo el analizador de citas bibliográficas.

Analizadores y reconocedores de texto

Dentro de este grupo de incluyen los ficheros que implementan los analizadores léxicos y sintácticos como además las clases java generadas automáticamente por estas:

- ParserCitas.lex: Analizador léxico de citas bibliográficas.
- ParserCitas.cup: Analizador sintáctico de las citas. Contiene la gramática que define la estructura reconocida del formato texto y BibTex definidos en el apartado C.3.2.
- CitaSingleton.java: El analizador sintáctico es el que procesa la información, guardándola en los atributos de la instancia de ésta clase. Es una clase de tipo Singleton es decir, sólo existiendo una instancia de ésta, de manera que tras el análisis pueda accederse a los atributos donde estará guardada la información [UMLPatterns].
- Conversor.java, ParserCitasCup.java y ParserCitasSym.java: las clases generadas de manera automática por JFlex y CUP al compilarse los analizadores léxico y sintáctico de citas.
- ParserAutores.lex: Analizador léxico para procesar el código HTML de la página web que muestra el trabajo del autor.
- ParserAutores.cup: Analizador sintáctico que se encarga de procesar el código HTML para recopilar el listado de citas del autor.
- AutorSingleton.java: clase de tipo Singleton que se encarga de guardar en sus atributos el listado de citas bibliográficas del autor de manera que a través de esta única instancia pueda accederse a dicha información una vez reconocida.
- ConversorAutores.java, ParserAutoresCup.java, ParserAutoresSym.java: las clases generadas de manera automática por JFlex y CUP al compilarse los analizadores de HTML léxico y sintáctico.

Otras utilidades

Dentro de este grupo se presentan el resto clases definidas en el sistema y que aportan las siguientes funcionalidades:

- LogUtils.java: contiene los métodos para crear/modificar el fichero de Log que guarda los mantenimientos realizados en base de datos.
- OrdenacionUtils.java: contiene los métodos de ordenación Burbuja y QuickSort para ordenar listas de revistas y conferencias.
- ParserException.java: define una excepción que es lanzada por los analizadores sintácticos para indicar que existe un error de formato.

- Fuentes.java, ProcedenciaAplicacion.java y TipoPublicación.java: clases de tipo enumeración que indican los tipos de índice cargados, los tipos de analizador y los tipos de publicación respectivamente.

C.6.3 Clases del modelo de datos

La clase Database.java contiene los métodos de inicialización y atributos necesarios para realizar la configuración inicial de la base de datos con Hibernate. Los ficheros de mapeo son:

- Categoria.hbm.xml: mapea la tabla categoría de base de datos con el POJO del mismo nombre.
- Conferencia.hbm.xml: mapea la tabla conferencia de base de datos con el POJO del mismo nombre.
- Indice.hbm.xml: mapea la tabla índice de base de datos con el POJO del mismo nombre.
- Journal.hbm.xml: mapea la tabla journal de base de datos con el POJO del mismo nombre.
- Macrocategoria.hbm.xml: mapea la tabla macrocategoria de base de datos con el POJO del mismo nombre.
- Publicacion.hbm.xml: mapea la tabla publicación de base de datos con el POJO del mismo nombre.
- Publicadas.hbm.xml: mapea la tabla publicadoencat de base de datos con el POJO Publicadas.

C.6.4 Ficheros de configuración

Struts e Hibernate precisan de sus respectivos de configuración, cada uno con una determinada funcionalidad:

- Struts-config.xml: Fichero de configuración de Struts. A través de este fichero se definen las clases ActionMapping y ActionForward que permiten configurar el flujo de la aplicación. Para los mapeos utiliza las clases Action y ActionForm definidas anteriormente.
- Hibernate.cfg.xml: Fichero de configuración de Hibernate donde se incluye la conexión a base de datos.
- Web.xml: Fichero de configuración predefinido de Struts el cual debe modificarse para añadir restricciones de seguridad y así limitar el acceso a la publicación protegiendo con usuario y contraseña.

C.7 Apariencia de la aplicación

Este apartado muestra la apariencia final de la aplicación una vez finalizado el proceso de implementación. Las páginas poseen las siguientes características:

- Todas las páginas poseen una cabecera separada del contenido, el cual ocupa el resto de la página.
- No se han incorporado colores (sólo blanco y negro) ya que no era un requisito de la aplicación.
- Los enlaces se muestran en azul subrayado, los títulos y subtítulos en negrita y los mensajes informativos en cursiva.
- La página de inicio y las principales de los analizadores presentan unos enlaces, en la parte superior izquierda, a estadísticas de la página (en color verde) como por ejemplo número de visitas diarias, promedio, pronósticos, etc.

Los siguientes sub apartados muestran la apariencia de las páginas más relevantes de la aplicación.

C.7.1 Página de inicio del analizador de una publicación

Publindex: Buscador de publicaciones

Búsqueda de:

 Ej: al teclear 'artificial intelligence' buscará dicho substring en el campo indicado

Índices de JCR donde realizar la búsqueda

Unknown Category Computer Science Multidisciplinary Sciences
 Robotics Telecommunications

Índices de CORE donde realizar la búsqueda

Mathematical Sciences Physical Sciences Chemical Sciences
 Earth Sciences Environmental Sciences Biological Sciences
 Agricultural and Veterinary Sciences Information and Computing Sciences Engineering
 Technology Medical and Health Sciences Built Environment and Design
 Education Economics Commerce, Management, Tourism and Services
 Studies In Human Society Psychology and Cognitive Sciences Law and Legal Studies
 Studies In Creative Arts and Writing Language, Communication and Culture History and Archaeology
 Philosophy and Religious Studies Multidisciplinary Unknown Category

Índices de SJR donde realizar la búsqueda

Artificial Intelligence Communication Computational Mathematics
 Computational Mechanics Computational Theory and Mathematics Computer Graphics and Computer-Aided Design
 Computer Networks and Communications Computer Science (miscellaneous) Computer Science Applications
 Computer Vision and Pattern Recognition Computers in Earth Sciences Discrete Mathematics and Combinatorics
 Multidisciplinary Theoretical Computer Science Hardware and Architecture
 Human-Computer Interaction Information Systems Signal Processing
 Software

Publindex contiene registros de los siguientes índices:

Año	Índice	Descripción
2002	JCR-Sc	JCR Science
2003	JCR-Sc	JCR Science

Figura 5: Página de inicio del buscador de una publicación

C.7.2 Página de resultado de la búsqueda de una publicación



Figura 6: Página de resultados de coincidencias

C.7.3 Página de valoraciones de una publicación

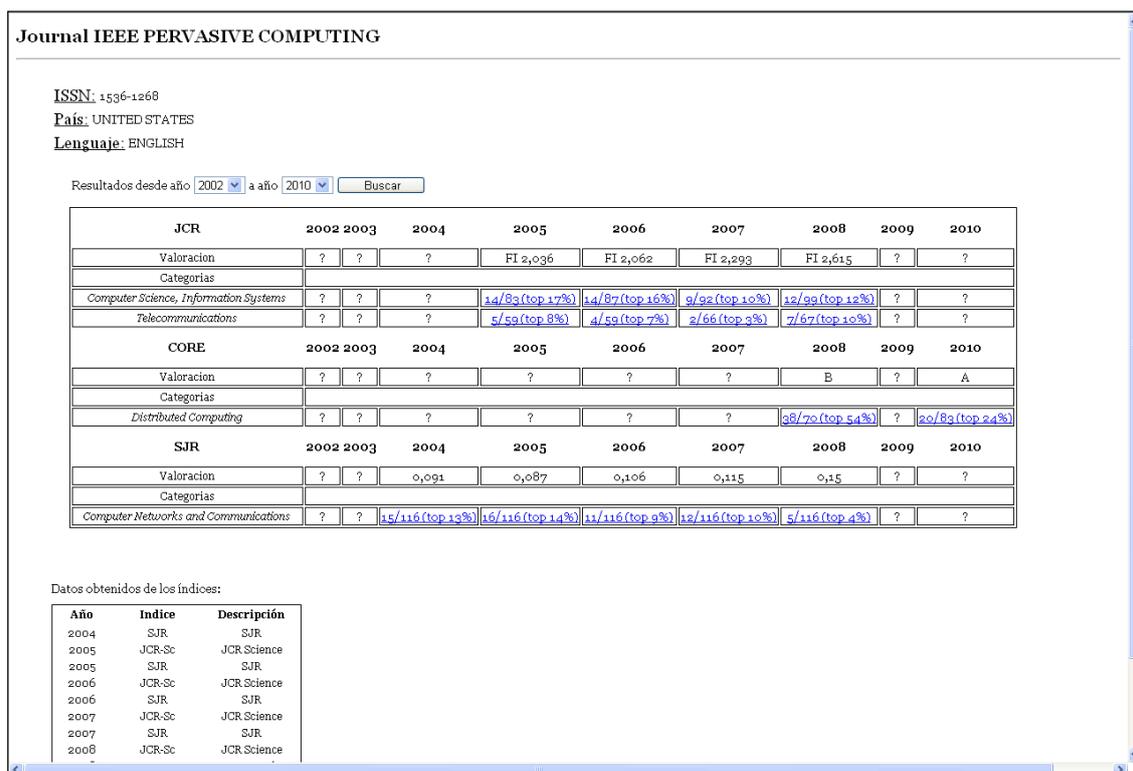


Figura 7: Página que muestra las valoraciones y datos de una revista

C.7.4 Página de resultados de la búsqueda de las publicaciones a través del trabajo de un autor

Citas bibliográficas de Carlos Bobed

A continuación se muestra cada una de las citas asociadas al evento que más similitud presenta con el indicado en la referencia

Datos referencia	Publicaciones resultado
Autores: Carlos Bobed and Sergio Ilarri and Eduardo Mena Año ref: 2010 Cita: Exploiting the Semantics of Location Granules in Location-Dependent Queries Journal/Conf: East-European Conference on Advances in Databases and Information Systems (ADBIS'10)	SYMPOSIUM ON ADVANCES IN DB AND INFORMATION SYSTEMS (ADBIS) (CORE B)
Autores: Carlos Bobed, Sergio Ilarri, Eduardo Mena Año ref: 2010 Cita: Distributed Mobile Computing: Development of Distributed Applications Using Mobile Agents Journal/Conf: International Conference on Parallel and Distributed Computing (PDP'10)	LASTED INTERNATIONAL CONFERENCE ON PARALLEL AND DISTRIBUTED COMPUTING AND SYSTEMS (PDCS) (CORE C)
Autores: Carlos Bobed and Eduardo Mena Año ref: 2010 Cita: Enhancing the Discovery of Web Services: A Keyword-oriented Multontology Reconciliation Journal/Conf: International Conference on Parallel and Distributed Computing (PDP'10)	LASTED INTERNATIONAL CONFERENCE ON PARALLEL AND DISTRIBUTED COMPUTING AND SYSTEMS (PDCS) (CORE C)
Autores: S. Ilarri and A. Corral and C. Bobed and E. Mena Año ref: 2009 Cita: Probabilistic Granule-Based Inside and Nearest Neighbor Queries Journal/Conf: East-European Conference on Advances in Databases and Information Systems (ADBIS'09)	SYMPOSIUM ON ADVANCES IN DB AND INFORMATION SYSTEMS (ADBIS) (CORE B)
Autores: Carlos Bobed and Raquel Trillo and Eduardo Mena and Jorge Bernad Año ref: 2008 Cita: Semantic Discovery of the User Intended Query in a Selectable Target Query Language Journal/Conf: International Conference on Web Intelligence (WI 2008)	IEEE/WIC/ACM INTERNATIONAL CONFERENCE ON WEB INTELLIGENCE AND INTELLIGENT AGENT TECHNOLOGY (WI-IAT) (CORE C)
Autores: S. Ilarri and E. Mena and C. Bobed Año ref: 2007 Cita: Processing Location-Dependent Queries with Location Granules Journal/Conf: On the Move to Meaningful Internet Systems 2007: OTM 2007 Workshops (PerSys)	Cannot extract the event
Autores: C. Bobed and E. Mena Año ref: 2007	

Figura 8: Página de resultados tras analizar las citas y buscar las publicaciones correspondientes

C.8 Evaluación de la implementación del extractor de publicaciones a través de citas bibliográficas

Este apartado trata en detalle todo el proceso de estudio de la eficiencia que fue realizado durante el proceso de desarrollo de la aplicación. Este estudio se basa en evaluar la calidad de la implementación de los analizadores léxicos y sintácticos de citas bibliográficas y código HTML.

Como ya se ha comentado a lo largo del documento, la aplicación utiliza una serie de herramientas para la obtención de resultados que no es del todo precisa. Estas herramientas corresponden con los reconocedores sintácticos y léxicos de referencias bibliográficas. Las pruebas realizadas permitieron ir refinando los analizadores hasta llegar a un considerable porcentaje de éxito en el reconocimiento.

C.8.1 Reconocedor de citas bibliográficas

Para determinar la precisión de reconocimiento y éxito del analizador de citas en formato BibTex se tomó como entrada un fichero que contenía 26 citas correspondientes a varios tipos de eventos en dicho formato:

- Revista (*@Article {...}*)
- Conferencia (*@Inproceedings {...}*)
- Capítulo o parte de un libro (*@Inbook {...}*)

De este conjunto se analizaron correctamente 19 de las citas, fallando aquellas que tenían los caracteres '{' y '}' incluidos en los campos de la cita BibTex y no tratados como caracteres delimitadores especiales. El porcentaje de acierto para formato BibTex alcanzado es de un 73% (19 aciertos /27 analizadas).

Además de realizar pruebas con el formato BibTex, se realizaron pruebas con citas en formato texto. El reconocedor tomó como entrada un fichero con 21 citas en formato simple, de las cuales reconoció todas a excepción de una en la que no se incluía para la fecha el formato "Mes (En inglés) Año (4 dígitos)". La tasa de aciertos para formato texto es de un 95% (20 aciertos/21 analizadas).

C.8.2 Reconocedor de citas bibliográficas a través de autor

Estas pruebas implican determinar la calidad que conjuntamente ofrecen el reconocedor de código HTML, que representa a la página web de un autor, y el analizador de citas bibliográficas, ya que el porcentaje de éxito viene determinado por el número de citas de la página web que han sido procesadas y de las que además la búsqueda de las publicaciones a las que hacen referencia ha sido correcta.

Se realizó la búsqueda del trabajo de los investigadores pertenecientes al SID Carlos Bobed, Sergio Ilarri, Raquel Trillo y Jorge Gracia. Se tomaron las siguientes mediciones:

Autor	Publicaciones	Conf. /Revistas	Analizadas	No encontradas	Éxitos de búsqueda	Búsqueda errónea
Carlos Bobed	8	7	7	2	5	0
Sergio Ilarri	54	46	34	12	19	3
Raquel Trillo	18	17	11	2	7	2
Jorge Gracia	15	14	11	2	7	2
Total	95	84	63	18	38	7

Tabla 8: Mediciones relativas a las pruebas sobre el buscador de publicaciones por autor

De estas mediciones se llegó a los siguientes parámetros que sirven como índice de la calidad de los analizadores:

- Número de publicaciones encontradas: 45 (38 éxitos + 7 erróneas)
- Número de citas referentes a revistas y conferencias: 84
- Tasa de reconocimiento de citas: 75% (63 analizadas/ 84 publicaciones válidas)
- Tasa de acierto de búsqueda de publicaciones: 84% (38 búsquedas válidas/ 45 pubs, encontradas)

De este estudio se identificaron tres tipos de búsqueda: no encontrada (cuando la publicación no se encuentra almacenada en el sistema), éxito (publicación encontrada y que corresponde con la referenciada en la cita) y errónea (publicación encontrada pero que no se corresponde con la indicada en la cita). A continuación se muestra un gráfico circular que muestra qué porcentaje del total de publicaciones analizadas corresponde a cada tipo de búsqueda:

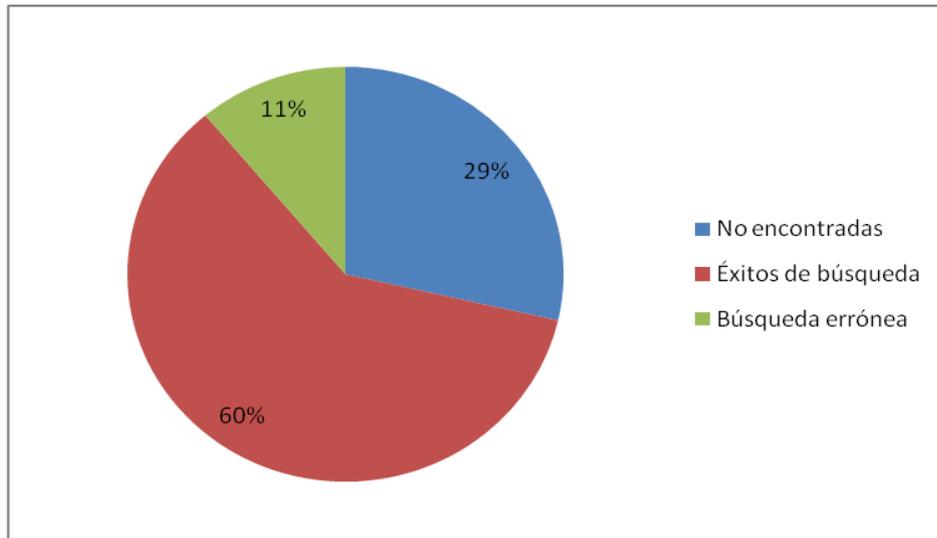


Figura 9: Gráfico circular que representa el porcentaje de las publicaciones analizadas que pertenecen a cada tipo de búsqueda

Todas estas pruebas y estadísticas llevaron a la conclusión de que los analizadores tienen un porcentaje de acierto elevado y que por tanto pueden ser considerados como herramientas aptas para implementar la funcionalidad definida en la aplicación.

C.9 Referencias

C.9.1 Referencias Web

[WikiBibTeX]: <http://es.wikipedia.org/wiki/BibTeX>

[WikiQuickSort]: <http://es.wikipedia.org/wiki/Quicksort>

[SID]: <http://sid.cps.unizar.es/>

Todas las referencias web han sido probadas a fecha de 25/08/2010.

C.9.2 Referencias

[Java]: David Flanagan (2005). *Java In A Nutshell*. O'Reilly.

[Java2]: Patrick Niemeyer, Jonathan Knudsen (2005). *Learning Java*. O'Reilly.

[Apache]: Ben Laurie, Peter Laurie (2002). *Apache: The Definitive Guide*. O'Reilly.

[MySQL]: Paul DuBois (2008). *MySQL*. Addison-Wesley.

[Struts]: James Holmes (2004). *Struts: The Complete Reference*. McGraw-Hill y Osborne Media.

[CUP]: Sergio Gálvez Rojas, Miguel Ángel Mora Mata (2005). *Java a Tope: Traductores Y Compiladores Con Lex/yacc, Jflex/cup Y Javacc*. Universidad de Malaga.

[UMLPatterns]: Larman C. (2001). *Applying UML and patterns: An introduction to O-O design and iterative development*. Prentice Hall.

[JSP]: Jayson Falkner, Ben Galbraith, Romin Irani (2002). *Desarrollo web con JSP*. Anaya Multimedia.

[Hibernate]: Christian Bauer and Gavin King (2004). *Hibernate in action*. Manning Publications.

Anexo D:

Fase de pruebas

Publindex: Análisis de la calidad de
publicaciones de investigación

Tabla de contenido

D.1	INTRODUCCIÓN	159
D.1.1	PROPÓSITO Y ALCANCE	159
D.1.2	REFERENCIAS	159
D.1.3	ACRÓNIMOS	159
D.1.4	RESUMEN.....	159
D.2	DESCRIPCIÓN DEL PROCESO	161
D.3	PRUEBAS UNITARIAS	163
D.3.1	PUBLICACIONDAO	163
D.3.1.1	<i>ExistePublicacion()</i>	163
D.3.1.2	<i>BuscarPublicacion()</i>	163
D.3.1.3	<i>ValoracionPublicacionEnAnyo</i>	164
D.3.1.4	<i>BuscarCoincidenciasPorNombre()</i>	164
D.3.1.5	<i>ExistePublicacionEnAnyoCat()</i>	164
D.3.2	JOURNALDAO.....	165
D.3.2.1	<i>ExisteJournal()</i>	165
D.3.2.2	<i>BuscarJournal()</i>	165
D.3.2.3	<i>BuscarCoincidenciasJournalPorNombre()</i>	166
D.3.2.4	<i>BuscarJournals()</i>	166
D.3.3	CONFERENCIADAO	166
D.3.3.1	<i>ExisteConferencia()</i>	166
D.3.3.2	<i>BuscarConferencias()</i>	166
D.3.3.3	<i>BuscarCoincidenciasConferenciaPorNombre()</i>	167
D.3.3.4	<i>BuscarConferencia()</i>	167
D.3.4	CATEGORIADAO.....	167
D.3.4.1	<i>BuscarCategoria()</i>	168
D.3.4.2	<i>BuscarCategorias()</i>	168
D.3.4.3	<i>BuscarFuentes()</i>	168
D.3.4.4	<i>BuscarPublicacionesCategoria()</i>	169
D.3.4.5	<i>BuscarPublicacionesCategoriaCORE()</i>	169
D.3.4.6	<i>PubsCategoriaEnValoracionCORE()</i>	170
D.3.4.7	<i>ValoracionesEnCategoria()</i>	170
D.3.4.8	<i>MaxAnyoCategorias()</i>	171
D.3.4.9	<i>MinAnyoCategorias()</i>	171
D.3.4.10	<i>MaxAnyoEnFuente()</i>	171

D.3.4.11	<i>NumPublicacionesCategoria()</i>	171
D.3.5	PUBLICADASDAO	172
D.3.5.1	<i>ExistePublicacionEnCat()</i>	172
D.3.5.2	<i>BuscarDatosPublicacionEnCat()</i>	172
D.3.5.3	<i>BuscarDatosPublicacion()</i>	172
D.3.6	MACROCATEGORIADAO.....	173
D.3.6.1	<i>ObtenerMacrocategorias()</i>	173
D.3.6.2	<i>ObtenerCategoriasMacrocategoria()</i>	173
D.3.6.3	<i>ObtenerMacrocategoria()</i>	173
D.3.7	INDICEDAO.....	174
D.3.7.1	<i>BuscarIndice()</i>	174
D.3.7.2	<i>BuscarIndicesAnyo()</i>	174
D.3.7.3	<i>BuscarIndicesPublicacion()</i>	174
D.3.7.4	<i>BuscarIndices()</i>	174
D.4	PRUEBAS DE INTEGRACIÓN Y SISTEMA	177
D.4.1	ANALIZADOR DE UNA PUBLICACIÓN	177
D.4.2	ANALIZADOR DE UN LISTADO DE PUBLICACIONES A TRAVÉS DE CITAS	179
D.4.3	ANALIZADOR DE LA CALIDAD DE UN AUTOR	179
D.4.4	MANTENIMIENTO DE PUBLICACIONES.....	180
D.5	PRUEBAS DE USUARIO.....	181
D.5.1	ESCENARIO. ANÁLISIS DE UNA PUBLICACIÓN	181
D.5.2	ESCENARIO. ANÁLISIS DE UN LISTADO DE PUBLICACIONES.....	182
D.5.3	ESCENARIO. ANÁLISIS DE LA CALIDAD DE UN AUTOR.....	182
D.5.4	ESCENARIO. MANTENIMIENTO DE PUBLICACIONES	182
D.6	RESULTADOS FINALES.....	185
D.6.1	ANALIZADOR DE UNA APLICACIÓN	185
D.6.2	ANALIZADOR DE UNA LISTA DE PUBLICACIONES POR REFERENCIAS Y AUTOR.....	185
D.6.3	MANTENIMIENTO DE PUBLICACIONES.....	185
D.6.4	CONCLUSIONES FINALES.....	186
D.7	REFERENCIAS	187
D.7.1	REFERENCIAS WEB.....	187
D.7.2	REFERENCIAS	187

D.1 Introducción

La finalidad de este apartado es dar una visión general del documento presentando su objetivo, alcance, referencias, etc.

D.1.1 Propósito y alcance

La finalidad de este documento es la de presentar detalladamente la fase de pruebas realizada durante el desarrollo de la aplicación. Esta fase engloba un conjunto de pruebas a distinto nivel cuyo propósito es el de comprobar el correcto funcionamiento y cumplimiento de los requisitos por parte del sistema.

El alcance de este documento comprende toda la fase de pruebas del proceso de desarrollo. Esta fase se realiza a la vez que la fase de implementación, de manera que pueda comprobarse la validez de las funcionalidades recientemente implementadas.

D.1.2 Referencias

Se realizan referencias a los documentos anexos de *Requisitos*, *Caso de estudio e Implementación*.

D.1.3 Acrónimos

Los acrónimos utilizados durante el documento son los siguientes:

- JCR: Journal Citation Reports
- CORE: COmputing Research & Education
- SJR: Scientific Journal Rankings
- DAO: Data Access Object
- DBLP: Digital Bibliography and Library Project

D.1.4 Resumen

Este documento detalla toda la fase de pruebas a través de una serie de apartados, además de la introducción que se presenta en ese apartado. El segundo apartado contiene una descripción de todo el proceso de pruebas, con objeto de mostrar todos los pasos que se dieron para determinar la validez del sistema. Los tres siguientes apartados presentan las pruebas unitarias, de sistema y usuario que se realizaron, incluyendo su finalidad y enumerando las pruebas realizadas en cada grupo. Finalmente, se presentan las conclusiones recogidas tras completar todo el proceso de pruebas.

D.2 Descripción del proceso

La fase de pruebas tiene como objetivo comprobar el correcto funcionamiento de la aplicación como además verificar que se han implementado todas las funcionalidades definidas durante el proceso de captación de requisitos.

Las pruebas pueden ser de tres tipos diferentes en función del nivel al que se aplican: unitarias, de integración y de usuario.

Las pruebas unitarias tienen como finalidad comprobar el comportamiento satisfactorio de cada clase probando cada método perteneciente a ella, asegurándonos así de que cada acceso a datos o método de lógica funcionan sin errores y proporcionan los resultados estimados. Estas pruebas son realizadas cada vez que una clase haya sido completada o modificada su funcionalidad.

Las pruebas de integración comprueban que todas las clases trabajan correctamente a nivel de sistema, es decir se comunican e intercambian información correctamente entre ellas e implementan las funcionalidades definidas. Estas pruebas se realizan por parte del desarrollador una vez que se haya implementado una funcionalidad principal de la aplicación.

Cuando haya sido implementada totalmente la aplicación o se tenga una versión entregable de ella, el usuario realiza una serie de pruebas con objeto de determinar si el resultado es el esperado. También pueden considerarse en este grupo las labores de mantenimiento una vez que ha sido entregada la aplicación, cuando el usuario reporta errores al desarrollador en el caso de suceder un error no controlado.

Todas las pruebas proporcionan una retroalimentación al desarrollador, de manera que si los resultados no son lo esperados se procede a una modificación, si aplica, en análisis, diseño e implementación. Estos cambios se incluyen en una nueva iteración en la fase de elaboración construcción o transición del sistema, generando una nueva versión de la aplicación (Ver anexo de *Caso de estudio*) [Test].

Todo este proceso puede repetirse hasta que los resultados sean aceptables a cada nivel de pruebas. Estos resultados son registrados y documentados en los siguientes apartados de este documento.

Para finalizar se muestra un diagrama que muestra gráficamente todo este proceso de pruebas:

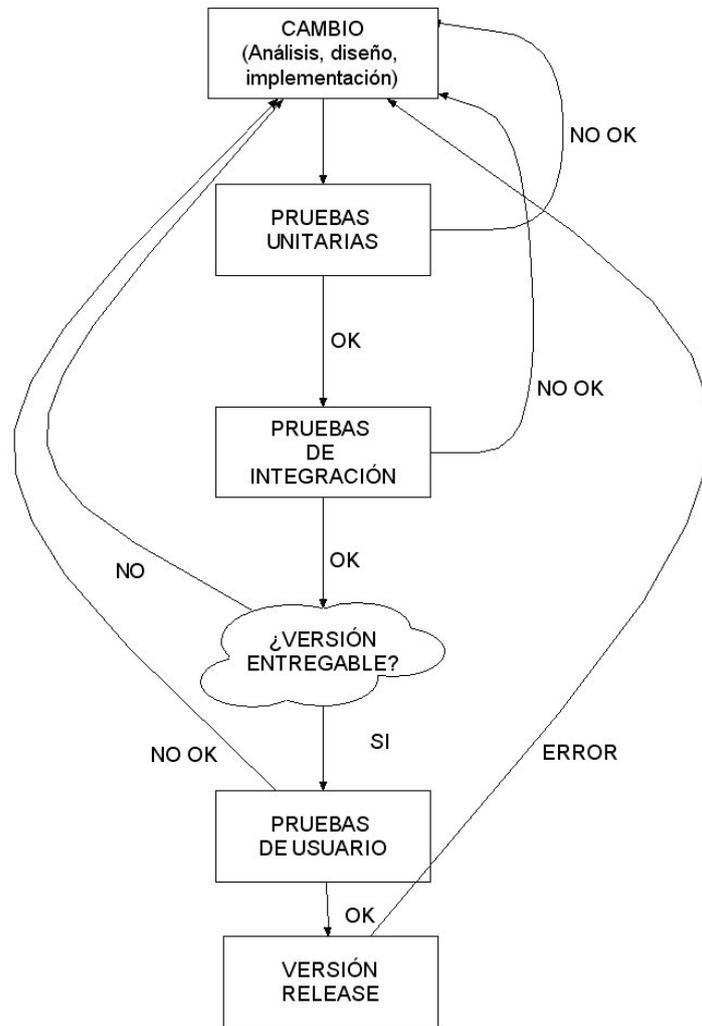


Figura 1: Proceso de pruebas y fases que lo componen

D.3 Pruebas unitarias

Las pruebas unitarias del sistema afectaron al conjunto de clases DAO de la aplicación (Ver anexo de *Implementación*). Estas clases son las encargadas del acceso a datos y reúnen la lógica necesaria para que sean utilizadas desde interfaz para acceder a los datos físicos, de manera que es necesario validar individualmente cada clase comprobando el comportamiento de cada uno de sus métodos.

La herramienta utilizada para realizar las pruebas unitarias fue *JUnit*. JUnit es un conjunto de bibliotecas que permite evaluar el comportamiento de los métodos de una clase en función de las entradas y salidas de éste. [JUnit], En el caso de las clases DAO, se prueban los métodos de consulta hacia base de datos.

Los siguientes apartados muestran las pruebas unitarias realizadas en cada clase, presentándose cada test en formato tabla. Dicha tabla contiene el nombre del test, el método de la clase a probar, la entrada al método, el resultado esperado y si es correcto.

D.3.1 PublicacionDAO

Este apartado contiene todas las pruebas unitarias para los métodos de la clase PublicacionDAO.

D.3.1.1 ExistePublicacion()

Nombre test	testExistePublicacion()
Método a probar	boolean ExistePublicacion (string, boolean)
Entrada del método	Siglas de una publicación almacenada y booleano a cierto para indicar que son siglas.
Resultado esperado	El método debe devolver cierto.
Funcionamiento	Correcto

Tabla 1: Prueba unitaria para el método ExistePublicacion()

D.3.1.2 BuscarPublicacion()

Nombre test	testBuscarPublicacionId()
Método a probar	Publicacion BuscarPublicacion (int)
Entrada del método	Identificador de una publicación almacenada en el sistema.
Resultado esperado	El método debe devolver una publicación correcta.
Funcionamiento	Correcto

Tabla 2: Prueba unitaria para el método BuscarPublicacion() dado su id por parámetro

Nombre test	testBuscarPublicacionNombre()
Método a probar	Publicacion BuscarPublicacion (string, boolean)
Entrada del método	Siglas de una publicación almacenada y booleano a cierto para indicar que son siglas.
Resultado esperado	El método debe devolver una publicación correcta.
Funcionamiento	Correcto

Tabla 3: Prueba unitaria para el método *BuscarPublicacion()* por nombre/siglas

D.3.1.3 ValoracionPublicacionEnAnyo

Nombre test	testValoracionPublicacionEnAnyo()
Método a probar	String ValoracionPublicacionEnAnyo (Publicacion,int)
Entrada del método	Una publicación válida de base de datos y un año en el cual presente valoraciones.
Resultado esperado	El método debe devolver una cadena correcta no nula.
Funcionamiento	Correcto

Tabla 4: Prueba unitaria para el método *ValoracionPublicacionEnAnyo()*

D.3.1.4 BuscarCoincidenciasPorNombre()

Nombre test	testBuscarCoincidenciasPorNombre()
Método a probar	List<Publicacion> BuscarCoincidenciasPorNombre (string)
Entrada del método	Una subcadena que contengan más de una de las publicaciones almacenadas en el sistema (por ejemplo 'ACM' o 'IEEE').
Resultado esperado	El método debe devolver una lista no vacía de publicaciones.
Funcionamiento	Correcto

Tabla 5: Prueba unitaria para el método *BuscarCoincidenciasPorNombre()*

D.3.1.5 ExistePublicacionEnAnyoCat()

Nombre test	testExistePublicacionEnAnyoCat()
Método a probar	boolean ExistePublicacionEnAnyoCat (Publicacion, int,Fuente)
Entrada del método	Publicación válida como además un año en el cual presente una valoración y la fuente que ha asignado dicho valor.
Resultado esperado	El método debe devolver cierto.
Funcionamiento	Correcto

Tabla 6: Prueba unitaria para el método *ExistePublicacionEnAnyoCat()*

D.3.2 JournalDAO

Este apartado contiene todas las pruebas unitarias para los métodos de la clase JournalDAO.

D.3.2.1 ExisteJournal()

Nombre test	testExisteJournal()
Método a probar	boolean ExisteJournal(string)
Entrada del método	ISSN en formato válido de una revista almacenada en el sistema.
Resultado esperado	El método debe devolver cierto.
Funcionamiento	Correcto

Tabla 7: Prueba unitaria para el método ExisteJournal()

D.3.2.2 BuscarJournal()

Nombre test	testBuscarJournalISSN()
Método a probar	Journal BuscarJournal(string)
Entrada del método	ISSN en formato válido de una revista almacenada en el sistema.
Resultado esperado	El método debe devolver una revista válida.
Funcionamiento	Correcto

Tabla 8. Prueba unitaria para el método BuscarJournal() dado su ISSN por parámetro

Nombre test	testBuscarJournalNombre()
Método a probar	Journal BuscarJournal(string,boolean)
Entrada del método	Nombre de una revista almacenada en el sistema y booleano a falso para indicar que es un nombre.
Resultado esperado	El método debe devolver una revista válida.
Funcionamiento	Correcto

Tabla 9: Prueba unitaria para el método BuscarJournal() dado su nombre/siglas por parámetro

Nombre test	testBuscarJournalId()
Método a probar	Journal BuscarJournal(int)
Entrada del método	Identificador de una revista almacenada en el sistema.
Resultado esperado	El método debe devolver una revista válida.
Funcionamiento	Correcto

Tabla 10: Prueba unitaria para el método BuscarJournal() dado su id por parámetro

D.3.2.3 BuscarCoincidenciasJournalPorNombre()

Nombre test	testBuscarCoincidenciasJournalPorNombre()
Método a probar	List<Journal> BuscarCoincidenciasJournalPorNombre (string)
Entrada del método	Subcadena que tengan en los títulos algunas revistas del sistema (por ejemplo 'ACM').
Resultado esperado	El método debe devolver una lista no vacía de revistas.
Funcionamiento	Correcto

Tabla 11: Prueba unitaria para el método BuscarCoincidenciasJournalPorNombre()

D.3.2.4 BuscarJournals()

Nombre test	testBuscarJournals()
Método a probar	List<Journal> BuscarJournals()
Entrada del método	No tiene parámetros el método.
Resultado esperado	El método debe devolver una lista no vacía de revistas.
Funcionamiento	Correcto

Tabla 12: Prueba unitaria para el método BuscarJournals()

D.3.3 ConferenciaDAO

Este apartado contiene todas las pruebas unitarias para los métodos de la clase ConferenciaDAO.

D.3.3.1 ExisteConferencia()

Nombre test	testExisteConferencia()
Método a probar	boolean ExisteConferencia(string,boolean)
Entrada del método	Siglas de una conferencia almacenada en el sistema y booleano a cierto para indicar que son siglas.
Resultado esperado	El método debe devolver un valor cierto.
Funcionamiento	Correcto

Tabla 13: Prueba unitaria para el método ExisteConferencia()

D.3.3.2 BuscarConferencias()

Nombre test	testBuscarConferencias()
Método a probar	List<Conferencia> BuscarConferencias()
Entrada del método	No tiene parámetros el método.
Resultado esperado	El método debe devolver una lista no vacía de conferencias.

Funcionamiento	Correcto
-----------------------	----------

Tabla 14: Prueba unitaria para el método `BuscarConferencias()`

D.3.3.3 `BuscarCoincidenciasConferenciaPorNombre()`

Nombre test	<code>testBuscarCoincidenciasConferenciaPorNombre()</code>
Método a probar	<code>List<Conferencia></code> <code>BuscarCoincidenciasConferenciaPorNombre (string)</code>
Entrada del método	Subcadena que tengan en los títulos algunas conferencias del sistema (por ejemplo 'conference').
Resultado esperado	El método debe devolver una lista no vacía de conferencias.
Funcionamiento	Correcto

Tabla 15: Prueba unitaria para el método `BuscarCoincidenciasConferenciaPorNombre()`

D.3.3.4 `BuscarConferencia()`

Nombre test	<code>testBuscarConferenciaNombre()</code>
Método a probar	<code>Conferencia BuscarConferencia(string,boolean)</code>
Entrada del método	Siglas de una conferencia almacenada en el sistema y booleano a cierto para indicar que son siglas.
Resultado esperado	El método debe devolver una conferencia válida.
Funcionamiento	Correcto

Tabla 16: Prueba unitaria para el método `BuscarConferencia()` dado su nombre/siglas por parámetro

Nombre test	<code>testBuscarConferenciaId()</code>
Método a probar	<code>Conferencia BuscarConferencia(int)</code>
Entrada del método	Identificador de una conferencia almacenada en el sistema.
Resultado esperado	El método debe devolver una conferencia válida.
Funcionamiento	Correcto

Tabla 17: Prueba unitaria para el método `BuscarConferencia()` dado su id por parámetro

D.3.4 CategoríaDAO

Este apartado contiene todas las pruebas unitarias para los métodos de la clase CategoríaDAO.

D.3.4.1 BuscarCategoría()

Nombre test	testBuscarCategoríaNombreAnyo()
Método a probar	Categoría BuscarCategoría (string,int,fuente)
Entrada del método	Nombre de una categoría almacenada en el sistema, un año en el que existan validaciones y la fuente a la que pertenezca.
Resultado esperado	El método debe devolver una categoría válida.
Funcionamiento	Correcto

Tabla 18: Prueba unitaria para el método BuscarCategoría() dado el nombre de la categoría, año y fuente

Nombre test	testBuscarCategoríaCodigo()
Método a probar	Categoría BuscarCategoría (string,int)
Entrada del método	Código de una categoría almacenada en el sistema y un año en el que existan validaciones.
Resultado esperado	El método debe devolver una categoría válida.
Funcionamiento	Correcto

Tabla 19: Prueba unitaria para el método BuscarCategoría() dado el nombre de la categoría y año

D.3.4.2 BuscarCategorías()

Nombre test	testBuscarCategorías()
Método a probar	Categoría BuscarCategorías (Publicacion,fuentes)
Entrada del método	Publicación válida y fuente en la que existen valoraciones para ella
Resultado esperado	El método debe devolver una lista no vacía de categorías.
Funcionamiento	Correcto

Tabla 20: Prueba unitaria para el método BuscarCategorías()

D.3.4.3 BuscarFuentes()

Nombre test	testBuscarFuentesPublicacion()
Método a probar	List<string> BuscarFuentes (Publicacion)
Entrada del método	Publicación válida.
Resultado esperado	El método debe devolver una lista no vacía de nombres de fuentes.
Funcionamiento	Correcto

Nombre test	testBuscarFuentes ()
Método a probar	List<string> BuscarFuentes()
Entrada del método	No tiene parámetros.
Resultado esperado	El método debe devolver una lista no vacía de nombres de fuentes.
Funcionamiento	Correcto

Tabla 21: Prueba unitaria para el método BuscarFuentes()

D.3.4.4 BuscarPublicacionesCategoria()

Nombre test	testBuscarPublicacionesCategoriaEnAnyo()
Método a probar	List<Publicacion> BuscarPublicacionesCategoria (Categoria, int,string)
Entrada del método	Categoría almacenada en el sistema, un año en el que aparezcan valoraciones y el nombre de la fuente a la que pertenezca la categoría.
Resultado esperado	El método debe devolver una lista no vacía de publicaciones.
Funcionamiento	Correcto

Tabla 22: Prueba unitaria para el método BuscarPublicacionesCategoria()dado el nombre de la categoría, año y fuente

Nombre test	testBuscarPublicacionesCategoriaNombreEnAnyo()
Método a probar	List<Publicacion> BuscarPublicacionesCategoria (string, int,string,boolean)
Entrada del método	Nombre de la categoría almacenada en el sistema, un año en el que aparezcan valoraciones, el nombre de la fuente a la que pertenezcan las valoraciones y booleano indicando si se quieren ordenados los resultados por factor de impacto (no afecta al resultado).
Resultado esperado	El método debe devolver una lista no vacía de publicaciones.
Funcionamiento	Correcto

Tabla 23: Prueba unitaria para el método BuscarCategorías()dada la categoría, año y fuente

D.3.4.5 BuscarPublicacionesCategoriaCORE()

Nombre test	testBuscarPublicacionesCategoriaCORECategoriaAnyo()
Método a probar	List<Publicacion> BuscarPublicacionesCategoriaCORE (Categoria,int,TipoPublicacion)
Entrada del método	Categoría almacenada en el sistema, un año en el que

	aparezcan valoraciones y el tipo de publicación que se desea devolver.
Resultado esperado	El método debe devolver una lista no vacía de publicaciones del tipo pasado por parámetro.
Funcionamiento	Correcto

Tabla 24: Prueba unitaria para el método BuscarPublicacionesCategoriaCORE () dada la categoría, año y tipo de publicación

Nombre test	testBuscarPublicacionesCategoriaCORECategoriaValoracion()
Método a probar	List<Publicacion> BuscarPublicacionesCategoriaCORE (Categoria,int,string,TipoPublicacion)
Entrada del método	Categoría almacenada en el sistema, un año en el que aparezcan valoraciones, la valoración máxima de las publicaciones a devolver y el tipo de publicación que son.
Resultado esperado	El método debe devolver una lista no vacía de publicaciones del tipo pasado por parámetro.
Funcionamiento	Correcto

Tabla 25: Prueba unitaria para el método BuscarPublicacionesCategoriaCORE () dada la categoría, año, valoración y tipo de publicación

D.3.4.6 PubsCategoriaEnValoracionCORE()

Nombre test	testPubsCategoriaEnValoracionCORE()
Método a probar	List<Publicacion> PubsCategoriaEnValoracionCORE (Categoria,int,string,TipoPublicacion)
Entrada del método	Categoría almacenada en el sistema, un año en el que aparezcan valoraciones, la valoración de las publicaciones a devolver y el tipo de publicación que son.
Resultado esperado	El método debe devolver una lista no vacía de publicaciones del tipo pasado por parámetro.
Funcionamiento	Correcto

Tabla 26: Prueba unitaria para el método PubsCategoriaEnValoracionCORE () dada la categoría, año, valoración y tipo de publicación

D.3.4.7 ValoracionesEnCategoria()

Nombre test	testValoracionesEnCategoria()
Método a probar	List<string> ValoracionesEnCategoria (Categoria)
Entrada del método	Categoría almacenada en el sistema de la que obtener las

	valoraciones.
Resultado esperado	El método debe devolver una lista no vacía de cadenas que corresponden con las valoraciones posibles en la categoría.
Funcionamiento	Correcto

Tabla 27: Prueba unitaria para el método *ValoracionesEnCategoría()*

D.3.4.8 MaxAnyoCategorias()

Nombre test	testMaxAnyoCategorias()
Método a probar	int MaxAnyoCategorias ()
Entrada del método	No tiene parámetros.
Resultado esperado	El método debe devolver un año válido, mayor que cero.
Funcionamiento	Correcto

Tabla 28: Prueba unitaria para el método *MaxAnyoCategorias()*

D.3.4.9 MinAnyoCategorias()

Nombre test	testMinAnyoCategorias()
Método a probar	int MinAnyoCategorias ()
Entrada del método	No tiene parámetros.
Resultado esperado	El método debe devolver un año válido, mayor que cero.
Funcionamiento	Correcto

Tabla 29: Prueba unitaria para el método *MinAnyoCategorias()*

D.3.4.10 MaxAnyoEnFuente()

Nombre test	testMaxAnyoEnFuente()
Método a probar	int MaxAnyoEnFuente (Publicacion, string)
Entrada del método	Publicación almacenada en el sistema que tenga valoraciones y la fuente a la que pertenezca la valoración a devolver.
Resultado esperado	Valor mayor que cero.
Funcionamiento	Correcto

Tabla 30: Prueba unitaria para el método *MinAnyoCategorias()*

D.3.4.11 NumPublicacionesCategoría()

Nombre test	testNumPublicacionesCategoría()
Método a probar	int NumPublicacionesCategoría (Categoría)

Entrada del método	Categoría almacenada en el sistema.
Resultado esperado	Valor mayor que cero.
Funcionamiento	Correcto

Tabla 31: Prueba unitaria para el método NumPublicacionesCategoria ()

D.3.5 PublicadasDAO

Este apartado contiene todas las pruebas unitarias para los métodos de la clase PublicadasDAO.

D.3.5.1 ExistePublicacionEnCat()

Nombre test	testExistePublicacionEnCat()
Método a probar	boolean ExistePublicacionEnCat (Publicacion,Categoria)
Entrada del método	Publicación y categoría almacenada en el sistema y en la que dicha publicación posea una valoración.
Resultado esperado	Valor devuelto cierto.
Funcionamiento	Correcto

Tabla 32: Prueba unitaria para el método ExistePublicacionEnCat ()

D.3.5.2 BuscarDatosPublicacionEnCat()

Nombre test	TestBuscarDatosPublicacionEnCat()
Método a probar	Publicadas BuscarDatosPublicacionEnCat (Publicacion,Categoria)
Entrada del método	Publicación y categoría almacenada en el sistema y en la que dicha publicación posea una valoración.
Resultado esperado	Devuelve un objeto Publicadas válido.
Funcionamiento	Correcto

Tabla 33: Prueba unitaria para el método BuscarDatosPublicacionEnCat ()

D.3.5.3 BuscarDatosPublicacion()

Nombre test	TestBuscarDatosPublicacion()
Método a probar	List<Publicadas> BuscarDatosPublicacion (Publicacion)
Entrada del método	Publicación almacenada en el sistema.
Resultado esperado	Devuelve un listado no vacío de valoraciones.
Funcionamiento	Correcto

Tabla 34: Prueba unitaria para el método BuscarDatosPublicacion ()

D.3.6 MacrocategoriaDAO

Este apartado contiene todas las pruebas unitarias para los métodos de la clase MacrocategoriaDAO.

D.3.6.1 ObtenerMacrocategorias()

Nombre test	TestObtenerMacrocategorias()
Método a probar	List<Macrocategoria> ObtenerMacrocategorias ()
Entrada del método	No tiene parámetros.
Resultado esperado	Devuelve un listado no vacío de macrocategorias.
Funcionamiento	Correcto

Tabla 35: Prueba unitaria para el método ObtenerMacrocategorias () que obtiene todas las macrocategorias almacenadas

Nombre test	TestObtenerMacrocategoriasFuente()
Método a probar	List<Macrocategoria> ObtenerMacrocategorias (string)
Entrada del método	Fuente a la que pertenezcan macrocategorias.
Resultado esperado	Devuelve un listado no vacío de macrocategorias.
Funcionamiento	Correcto

Tabla 36: Prueba unitaria para el método ObtenerMacrocategorias() dada la fuente pasada por parámetro

D.3.6.2 ObtenerCategoriasMacrocategoria()

Nombre test	TestObtenerCategoriasMacrocategoria()
Método a probar	List<Categoria> ObtenerCategoriasMacrocategoria (Macrocategoria)
Entrada del método	Macrocategoria válida.
Resultado esperado	Devuelve un listado no vacío de categorías.
Funcionamiento	Correcto

Tabla 37: Prueba unitaria para el método ObtenerCategoriasMacrocategoria()

D.3.6.3 ObtenerMacrocategoria()

Nombre test	TestObtenerMacrocategoria()
Método a probar	Macrocategoria ObtenerMacrocategoria (string)
Entrada del método	Nombre de una macrocategoria almacenada en el sistema.
Resultado esperado	Devuelve una macrocategoria válida.

Funcionamiento	Correcto
-----------------------	----------

Tabla 38: Prueba unitaria para el método ObtenerMacrocategoria()

D.3.7 IndiceDAO

Este apartado contiene todas las pruebas unitarias para los métodos de la clase IndiceDAO.

D.3.7.1 BuscarIndice()

Nombre test	TestBuscarIndice()
Método a probar	Indice BuscarIndice (int)
Entrada del método	Identificador de un índice almacenado en el sistema.
Resultado esperado	Devuelve un índice válido.
Funcionamiento	Correcto

Tabla 39: Prueba unitaria para el método BuscarIndice()

D.3.7.2 BuscarIndicesAnyo()

Nombre test	TestBuscarIndicesAnyo()
Método a probar	List<int> BuscarIndicesAnyo (int)
Entrada del método	Año al que pertenezca un índice almacenado en el sistema.
Resultado esperado	Devuelve un listado no vacío de años.
Funcionamiento	Correcto

Tabla 40: Prueba unitaria para el método BuscarIndicesAnyo()

D.3.7.3 BuscarIndicesPublicacion()

Nombre test	TestBuscarIndicesPublicacion()
Método a probar	List<int> BuscarIndicesPublicacion (int,int)
Entrada del método	Identificador de una publicación almacenada y año del cual se tenga valoración y por tanto pertenezca a un índice.
Resultado esperado	Devuelve un listado no vacío de identificadores de los índices.
Funcionamiento	Correcto

Tabla 41: Prueba unitaria para el método BuscarIndicesPublicacion()

D.3.7.4 BuscarIndices()

Nombre test	TestBuscarIndices()
--------------------	---------------------

Método a probar	List<Indice> BuscarIndices ()
Entrada del método	No tiene parámetros.
Resultado esperado	Devuelve un listado no vacío de índices.
Funcionamiento	Correcto

Tabla 42: Prueba unitaria para el método BuscarIndices()

D.4 Pruebas de integración y sistema

Este apartado muestra las pruebas de integración y sistema que fueron realizadas para comprobar el correcto funcionamiento de la aplicación en su conjunto cada vez que se modificara o ampliara la funcionalidad.

Las pruebas están agrupadas en función de los principales subsistemas a los que pertenezcan.

D.4.1 Analizador de una publicación

A continuación se presentan el conjunto de pruebas relativas a comprobar la funcionalidad y validez del subsistema.

Identificador	PUB001
Título	Prueba de búsqueda de coincidencias
Descripción	Al introducirse el nombre o siglas de una publicación debe mostrarse un listado de las coincidencias de la búsqueda realizada.
Resultados esperados	<ul style="list-style-type: none"> • La búsqueda sólo se realizará en el caso de que se introduzca una cadena no vacía. • El listado de coincidencias deberá aparecer agrupado por tipo de publicación (revista o conferencia), ordenado alfabéticamente y estar compuesto de enlaces a la página de resultados de la publicación a la que identifica. • Cada publicación listada debe mostrar las valoraciones en el último año disponible para cada índice.

Tabla 43: Descripción de la prueba de búsqueda de coincidencias

Identificador	PUB002
Título	Prueba de búsqueda de una revista por ISSN
Descripción	Al introducirse el ISSN de una revista debe mostrar sus datos y los relativos a las valoraciones.
Resultados esperados	<ul style="list-style-type: none"> • La búsqueda sólo se realizará en el caso de que se introduzca una cadena no vacía. • Para probar la validez de los resultados, seguir prueba PUB003

--	--

Tabla 44: Descripción de la prueba de búsqueda de una revista por ISSN

Identificador	PUB003
Título	Prueba de búsqueda de valoraciones de una publicación
Descripción	Al seleccionarse un enlace del listado de coincidencias o buscar una revista por ISSN, debe redirigirse a la página que muestre los datos de la publicación y sus valoraciones en cada fuente.
Resultados esperados	<ul style="list-style-type: none"> • Deben mostrarse los datos relevantes de la publicación. • Debe aparecer una tabla en la que cada dato de ella debe presentar la valoración para la publicación en la categoría de un índice para un año en concreto. • Cada valoración es un enlace a la categoría en el año al que hace referencia. Debe indicar la posición en la categoría de la publicación, el número de publicaciones totales de ésta y el top en porcentaje en el que está incluido.

Tabla 45: Descripción de la prueba de búsqueda valoraciones de una publicación

Identificador	PUB004
Título	Prueba de búsqueda de categoría
Descripción	Al seleccionar una de las valoraciones de una publicación, se debe redirigir la página que muestre la información de la categoría en el año de la valoración.
Resultados esperados	<ul style="list-style-type: none"> • Debe mostrarse el nombre de la categoría, índice al que pertenece y año. • Debe mostrarse un listado de enlaces a las publicaciones que componen la categoría junto con su valoración en el índice y ordenado por ésta. • La información de la publicación de cuya página se ha redirigido a la categoría debe aparecer resaltado para detectarlo fácilmente.

Tabla 46: Descripción de la prueba de búsqueda de las publicaciones de una categoría

D.4.2 Analizador de un listado de publicaciones a través de citas

Identificador	PUB005
Título	Prueba de reconocimiento de citas bibliográficas
Descripción	Al introducir un listado de referencias en formato texto o BibTex se muestra el listado de publicaciones al cual hacen referencia.
Resultados esperados	<ul style="list-style-type: none"> • Debe introducirse un listado no vacío de citas bibliográficas y con el formato correcto. • Debe mostrarse un listado de las citas junto con los enlaces a las publicaciones a las que hacen referencia. Cada enlace del listado redirige a la página de visualizar los resultados de una publicación. • Para probar la validez de los resultados de una publicación, realizar prueba PUB003.

Tabla 47: Descripción de la prueba de reconocimiento de citas bibliográficas

D.4.3 Analizador de la calidad de un autor

Identificador	PUB006
Título	Prueba de búsqueda del trabajo de un autor
Descripción	Al introducir el nombre de un investigador, se muestra el listado de sus referencias bibliográficas y las publicaciones a las que hacen referencia.
Resultados esperados	<ul style="list-style-type: none"> • Debe introducirse el nombre de un autor válido y perteneciente al departamento de Sistemas de Información Distribuidos (SID). • Debe mostrarse un listado de las citas bibliográficas que han podido ser reconocidas y que representan su trabajo y los enlaces a las publicaciones a las que hacen referencia. • Para probar la validez de los resultados de una publicación, realizar prueba PUB003.

Tabla 48: Descripción de la prueba de búsqueda del trabajo de un autor

D.4.4 Mantenimiento de publicaciones

Identificador	PUB007
Título	Prueba de fusión de publicaciones
Descripción	Dentro del listado de coincidencias al realizar la búsqueda de una publicación, se selecciona la opción de fusionar una publicación de dicho listado con otra almacenada en el sistema para fusionar los datos de ambos registros en uno sólo.
Resultados esperados	<ul style="list-style-type: none"> • Se debe haber llegado al listado de coincidencias a través de la prueba PUB001. • Se selecciona la opción de fusionar una publicación concreta del listado y establecerla así como publicación origen de la fusión. • Se redirige a la página de búsqueda de la publicación destino. • Se debe introducir el criterio de búsqueda, introduciendo nombre/siglas o ISSN. • Se redirige a la página de selección de la publicación destino. • Se debe seleccionar del desplegable el nombre de la publicación en la que se quieren guardar los cambios de la fusión. • Se regresa al inicio de la aplicación. • Se debe comprobar que ahora los resultados de las dos publicaciones originales están almacenados en la seleccionada como destino y que la publicación origen ya no se encuentra almacenada. • Comprobar que se han registrado las acciones del mantenimiento en el fichero de log (FusionLog.sql).

Tabla 49: Descripción de la prueba de fusión de dos publicaciones

D.5 Pruebas de usuario

Estas pruebas permiten recoger las impresiones de aquellos usuarios reales de la aplicación y errores no controlados que pueden escaparse al desarrollador al conocer a la perfección todos los pasos que deben darse para obtener los resultados.

A continuación se muestran una serie de escenarios a ejecutar por el usuario y así comprobar cómo se comporta la aplicación en estas situaciones y si devuelve resultados correctos.

D.5.1 Escenario. Análisis de una publicación

El usuario debe ejecutar los siguientes pasos:

1. Accede a la aplicación.
2. Selecciona la opción de búsqueda de una conferencia o revista (journal).
3. Si se selecciona la opción de buscar por título o siglas:
 - 3.1. Realiza la búsqueda de todas las publicaciones que tengan en su nombre la subcadena 'ACM'.
 - 3.2. Seleccionar la publicación 'ACM Computing Surveys'.
 - 3.3. Comprobar en qué categorías del índice JCR se da valoración en el año 2008. El resultado debe ser 'Computer Science, Theory & Method'.
 - 3.4. Acceder a la categoría 'Computer Science, Theory & Method' del año 2005.
 - 3.5. Comprobar qué posición ocupa la revista 'Artificial Life'. El resultado debe ser la posición 12.
 - 3.6. Accede a la información de dicha publicación. Comprobar que el ISSN es el '1530-9185'.
 - 3.7. Anotar en qué categorías aparece información de la revista dentro del índice SJR. El resultado debe ser 'Artificial Intelligence' y 'Computational Theory and Mathematics'.
4. Si se selecciona la opción de buscar por ISSN:
 - 4.1. Introducir el ISSN '0360-0300'.
 - 4.2. Comprobar que la revista a la que se accede es 'ACM Computing Surveys'.
 - 4.3. Mismos pasos 3.3 a 3.7
5. Regresar a la página principal del buscador.
6. Se selecciona la opción de buscar por título o siglas.
7. Introducir las siglas 'WWW'.
8. Seleccionar la conferencia 'International World Wide Web Conference'.
9. Comprobar que la valoración asignada a la conferencia en CORE en el año 2008. El resultado debe ser 'A*'.
El resultado debe ser 'A*'.
10. Comprobar que las categorías en las que aparece en el año 2010 son 'Distributed Computing' e 'Information Systems'.

11. Comprobar cuántas conferencias con valoración 'A' existen en la categoría 'Information Systems'. El resultado debe ser 32.
12. Verificar que la conferencia 'WWW' aparece en ese grupo.

D.5.2 Escenario. Análisis de un listado de publicaciones

1. Acceder a la aplicación.
2. Seleccionar la opción de analizar referencias bibliográficas.
3. Introducir dos referencias bibliográficas en formato texto o BibTex del autor Eduardo Mena, copiando de la web del SID y pegando en el cuadro de texto del buscador. Anotar las citas seleccionadas y los resultados.
4. Comprobar que se accede correctamente a la página de resultados de cada publicación analizada.

D.5.3 Escenario. Análisis de la calidad de un autor

1. Acceder a la aplicación.
2. Seleccionar la opción de analizar publicaciones de una persona.
3. Introducir el nombre de 'Sergio Ilarri'.
4. Comprobar que aparecen 34 resultados.
5. Seleccionar la publicación 'Information Sciences' de la segunda cita.
6. Comprobar que la columna de las valoraciones del año que se corresponde al de la cita aparece resaltado.
7. Verificar que se accede correctamente a los datos de las valoraciones y categorías de la publicación.

D.5.4 Escenario. Mantenimiento de publicaciones

1. Acceder a la aplicación.
2. Seleccionar la opción de búsqueda de una conferencia o revista.
3. Se selecciona la opción de buscar por título o siglas.
4. Realiza la búsqueda de todas las publicaciones que tengan en su nombre la subcadena 'pervasive'.
5. Seleccionar la opción de 'Fusionar con otras publicaciones' de la publicación 'International Journal on Pervasive Computing and Communications'.
6. Introducir la subcadena 'pervasive' para buscar la publicación destino de la fusión.
7. Seleccionar del desplegable la publicación destino y continuar. Anotar cuál ha sido seleccionada.

8. Comprobar que la operación ha sido completada. Anotar en qué publicación se han guardado los resultados de la fusión y comprobar que la publicación 'International Journal on Pervasive Computing and Communications' ya no existe.
9. Anotar el contenido de las últimas 5 líneas del fichero de log 'FusionLog.sql'.

D.6 Resultados finales

Este apartado contiene las impresiones de los usuarios de la aplicación una vez que se ha finalizado la construcción y se han corregido los errores que fueron surgiendo durante las pruebas a distinto nivel. Los resultados son presentados por cada subsistema de la aplicación.

D.6.1 Analizador de una aplicación

El sistema es muy intuitivo y simple, de manera que el usuario puede consultar rápidamente la información que necesita. Las valoraciones basadas en el top en cada categoría le dan una visión al usuario mucho más clara de la calidad de las publicaciones.

Cuando la búsqueda implica una gran cantidad de resultados, el tiempo empleado en ella es elevado ya que además se van mostrando por pantalla todas las coincidencias. Una solución a este problema consiste en realizar una paginación de los resultados, de manera que sólo se muestren un número limitado de revistas y conferencias en cada página (por ejemplo, paginar cada 10 resultados).

D.6.2 Analizador de una lista de publicaciones por referencias y autor

El analizador de citas todavía presenta algunos problemas relativos a los formatos BibTex y texto a reconocer. El caso de BibTex es el más notable, ya la herramienta en la que se edita este formato, Latex, guarda ciertos caracteres de manera especial (por ejemplo, la letra ñ como $\{\{n\}\}$). No obstante, el usuario considera que su precisión es suficiente y que estos problemas podrían resolverse en futuras ampliaciones de la aplicación haciendo el analizador mucho más flexible.

El subsistema de análisis de un autor presenta un mecanismo muy sencillo y simple de búsqueda. Ya que utiliza el subsistema de referencias bibliográficas presenta la misma problemática que la expuesta anteriormente. Este subsistema podría ser ampliado en el futuro haciendo posible acceder al trabajo de otros autores que no pertenecieran al SID (por ejemplo, búsquedas por autor en DBLP).

D.6.3 Mantenimiento de publicaciones

Este subsistema permite resolver la problemática existente de publicaciones duplicadas que pueden aparecer en el sistema. El usuario considera este subsistema especialmente útil y muy intuitivo.

Sin embargo, los pasos a dar para realizar el mantenimiento son excesivos, teniendo que realizar una nueva búsqueda de la publicación destino de la fusión cuando podrían utilizarse para su selección los resultados de la búsqueda de la publicación origen. Con esta solución también podría evitarse el poder seleccionar cualquier publicación sin semejanza alguna como destino de la fusión.

D.6.4 Conclusiones finales

La aplicación satisface las necesidades del usuario cumpliendo todos los requisitos establecidos al inicio del proceso de desarrollo. El sistema se caracteriza por su flexibilidad y por centralizar toda la información relativa a los índices de calidad más importantes ofreciendo un mecanismo sencillo y rápido de análisis.

Las mejoras e impresiones negativas de los anteriores apartados no suponen problemas de grandes dimensiones para el usuario, pudiendo corregirse en futuras ampliaciones de la aplicación.

D.7 Referencias

D.7.1 Referencias web

[SID]: <http://sid.cps.unizar.es/>

Todas las referencias web han sido probadas a fecha de 25/08/2010.

D.7.2 Referencias bibliográficas

[Test]: Elfriede Dustin (2002). *Effective Software Testing*. Addison Wesley.

[JUnit]: Petar Tahchiev, Felipe Leme, Vincent Massol, Gary Gregory (2010). *JUnit in Action*. Manning.

Anexo E:

Manual de usuario

Publindex: Análisis de la calidad de
publicaciones de investigación

Tabla de contenido

E.1	INTRODUCCIÓN	193
E.2	ANALIZADOR DE UNA PUBLICACIÓN	195
E.2.1	ACCEDER AL ANALIZADOR	195
E.2.2	BÚSQUEDA DE LAS VALORACIONES DE UNA PUBLICACIÓN DADO SU NOMBRE O SIGLAS	195
E.2.3	BÚSQUEDA DE LAS VALORACIONES DE UNA REVISTA DADO SU ISSN	198
E.2.4	VER EL CONTENIDO DE UNA CATEGORÍA JCR/SJR	199
E.2.5	VER EL CONTENIDO DE UNA CATEGORÍA CORE	200
E.2.6	MANTENIMIENTO DE PUBLICACIONES.....	202
E.3	ANALIZADOR DE UN LISTADO DE PUBLICACIONES A TRAVÉS DE REFERENCIAS BIBLIOGRÁFICAS	205
E.3.1	ACCEDER AL ANALIZADOR.....	205
E.3.2	BÚSQUEDA DE UN LISTADO DE PUBLICACIONES	205
E.3.3	FORMATOS VÁLIDOS DE LAS CITAS	207
E.4	ANALIZADOR DE LA CALIDAD DEL TRABAJO DE UN AUTOR	209
E.4.1	ACCEDER AL ANALIZADOR.....	209
E.4.2	BÚSQUEDA DEL TRABAJO DE UN AUTOR	209
E.5	CARGA DEL ÍNDICE DE CALIDAD CSCR	211
E.5.1	ANTES DE EMPEZAR	211
E.5.2	INFORMACIÓN A ALMACENAR EN BASE DE DATOS	211
E.5.3	CÓMO PROCESAR LOS DATOS	212

E.1 Introducción

Este documento tiene la finalidad de servir de guía al usuario a la hora de utilizar la aplicación, mostrando todas las opciones y servicios que ofrece el sistema. Para ello, la explicación se apoya en pantallazos que servirán como guía visual para el lector.

El alcance de este documento comprende toda la fase de uso de la aplicación por parte del usuario, desde que se le entrega la primera versión estable del sistema.

Este manual desglosa la aplicación en subsistemas, de manera que puedan mostrarse las opciones que ofrece cada uno de ellos. En primer lugar, se describe paso a paso cómo poder realizar búsquedas de valoraciones de revistas y conferencias de manera individual por nombre o ISSN. Los siguientes apartados describen cómo realizar análisis de la calidad de listados de publicaciones y de autor.

Una vez presentada la guía de la aplicación, se describen los pasos a dar para poder ampliar el juego de datos del que se dispone añadiendo el índice de calidad CSCR. Estas pautas no serán del todo exactas, pero servirán al usuario como una ayuda de partida para poder mejorar la aplicación.

E.2 Analizador de una publicación

Este apartado muestra los pasos a seguir para consultar las valoraciones de calidad de una publicación concreta a través de su nombre/siglas o ISSN. Además, también se muestra la guía de uso de la fusión de publicaciones.

E.2.1 Acceder al analizador

Para acceder a este buscador, debe seleccionarse la opción de 'Buscar una conferencia o journal' del menú principal. La dirección URL para acceder al menú principal es la siguiente: <http://nacho.cps.unizar.es:8080/Publindex/jsps/index.jsp>.

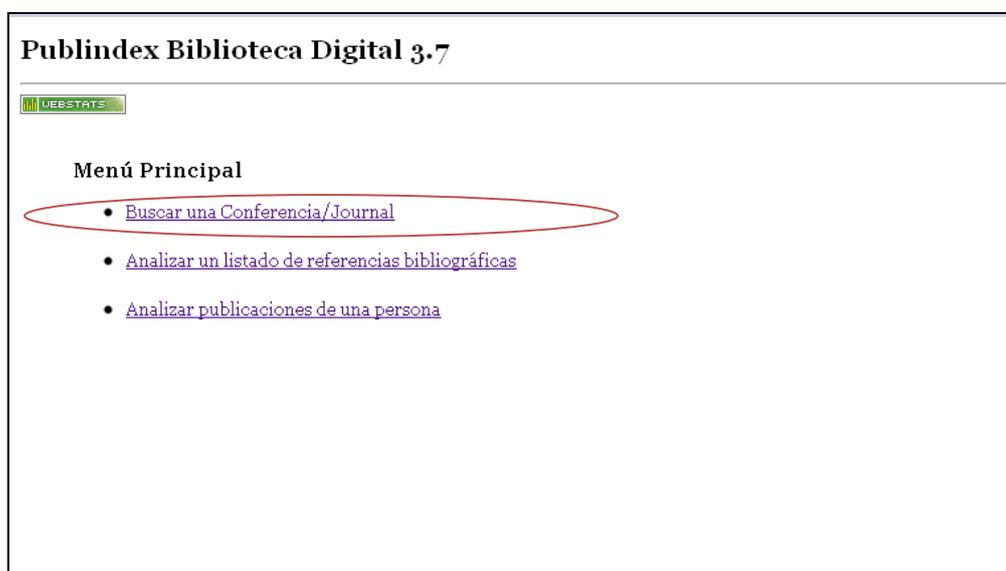


Figura 1: Página de inicio de la aplicación: *Buscar una conf./revista*

E.2.2 Búsqueda de las valoraciones de una publicación dado su nombre o siglas

En el formulario de la página de inicio del buscador de una publicación debe seleccionarse la opción de buscar por título e introducir las siglas o la cadena que contengan el título o siglas de la publicación.

Publindex: Buscador de publicaciones

WEBSTATS

Búsqueda de:

Journal/Conferencia (Búsqueda por título)

Journal/Conferencia (Búsqueda por título)
 Journal (Búsqueda por ISSN)
 Buscar en el campo indicado

Índices de JCR donde realizar la búsqueda

Unknown Category Computer Science Multidisciplinary Sciences
 Robotics Telecommunications

Índices de CORE donde realizar la búsqueda

Mathematical Sciences Physical Sciences Chemical Sciences

Figura 2: Selección del criterio de búsqueda por nombre o siglas

Además de introducir el nombre de la publicación, pueden seleccionarse las macrocategorías de cada índice en las que realizar la búsqueda. Las macrocategorías a seleccionar se presentan en formato checkbox, de manera que incluir en la búsqueda esa macrocategoría depende de si ha sido marcado o no. Por defecto aparecen unas macrocategorías seleccionadas.

Publindex: Buscador de publicaciones

WEBSTATS

Búsqueda de:

Journal/Conferencia (Búsqueda por título)

Al teclear 'artificial intelligence' buscará dicho substring en el campo indicado

Índices de JCR donde realizar la búsqueda

Unknown Category Computer Science Multidisciplinary Sciences
 Robotics Telecommunications

Índices de CORE donde realizar la búsqueda

Mathematical Sciences Physical Sciences Chemical Sciences
 Earth Sciences Environmental Sciences Biological Sciences
 Agricultural and Veterinary Sciences Information and Computing Sciences Engineering
 Technology Medical and Health Sciences Built Environment and Design
 Education Economics Commerce, Management, Tourism and Services
 Studies In Human Society Psychology and Cognitive Sciences Law and Legal Studies
 Studies In Creative Arts and Writing Language, Communication and Culture History and Archaeology
 Philosophy and Religious Studies Multidisciplinary Unknown Category

Índices de SJR donde realizar la búsqueda

Artificial Intelligence Communication Computational Mathematics
 Computational Mechanics Computational Theory and Mathematics Computer Graphics and Computer-Aided Design
 Computer Networks and Communications Computer Science (miscellaneous) Computer Science Applications
 Computer Vision and Pattern Recognition Computers in Earth Sciences Discrete Mathematics and Combinatorics
 Multidisciplinary Theoretical Computer Science Hardware and Architecture
 Human-Computer Interaction Information Systems Signal Processing
 Software

Figura 3: Macrocategorías seleccionadas por defecto en la aplicación

Una vez introducida toda la información de la búsqueda, se pulsa el botón 'Buscar' que redirige a la página de coincidencias resultados. Por ejemplo, pongamos que hemos introducido la cadena 'ACM com':

Esta página muestra todas las coincidencias, tanto de revistas y conferencias, agrupadas por tipo. En este caso, sólo se han encontrado resultados de revistas.

Notar que en las coincidencias ya se puede comprobar la calidad asignada por cada índice en el año más reciente, para podernos hacer una idea rápida de la calidad reciente de la revista o conferencia.

Se han encontrado 4 resultados para la clave 'ACM com':

Se listan a continuación las coincidencias alfabéticamente junto con sus valoraciones más recientes

Journals (4 resultados)

- [ACM COMPUTER COMMUNICATION REVIEW \(ISSN no disponible\)](#) (CORE B)
[Fusionar con otras publicaciones]
- [ACM COMPUTERS IN ENTERTAINMENT \(ISSN no disponible\)](#) (CORE B)
[Fusionar con otras publicaciones]
- [ACM COMPUTERS IN ENTERTAINMENT \(1544-3574\)](#) (CORE B) (SJR 0,034)
[Fusionar con otras publicaciones]
- [ACM COMPUTING SURVEYS \(0360-0300\)](#) (JCR 9,92) (CORE A*) (SJR 0,266)
[Fusionar con otras publicaciones]

Publindex contiene registros de los siguientes índices:

Año	Índice	Descripción
2002	JCR-Sc	JCR Science

Figura 4: Pantalla de coincidencias de la búsqueda

Pinchamos sobre el enlace del nombre la publicación de la que queremos ver la calidad, supongamos que se quiere ver 'ACM computing Surveys'.

A continuación se muestran las valoraciones de la publicación en formato tabla junto con datos importantes de ésta como son el país, ISSN, nombre y lenguaje.

Journal ACM COMPUTING SURVEYS

ISSN: 0360-0300
 País: UNITED STATES
 Lenguaje: ENGLISH

Resultados desde año 2002 a año 2010

JCR	2002	2003	2004	2005	2006	2007	2008	2009	2010
Valoración	FI 2,769	FI 7,5	FI 10,037	FI 7,4	FI 4,13	FI 5,25	FI 9,92	?	?
Categorías									
Computer Science, Theory & Method	1/69 (top 1%)	-	1/70 (top 1%)	1/71 (top 1%)	1/75 (top 1%)	1/70 (top 1%)	1/85 (top 1%)	?	?
Unknown Category	-	4/459 (top 1%)	-	-	-	-	-	?	?
CORE	2002	2003	2004	2005	2006	2007	2008	2009	2010
Valoración	?	?	?	?	?	?	A*	A*	A*
Categorías									
Computer Software	?	?	?	?	?	?	-	6/74 (top 8%)	6/92 (top 7%)
Information and Computing Sciences	?	?	?	?	?	?	2/51 (top 4%)	-	-
SJR	2002	2003	2004	2005	2006	2007	2008	2009	2010
Valoración	?	?	0,217	0,282	0,246	0,199	0,266	?	?
Categorías									
Computational Theory and Mathematics	?	?	6/124 (top 5%)	9/124 (top 2%)	6/124 (top 5%)	7/124 (top 6%)	6/124 (top 5%)	?	?

Figura 5: Datos y valoraciones de la publicación ACM Computing Surveys

Como puede verse, las valoraciones están agrupadas por índice de calidad, presentándose para cada uno la siguiente información:

- La primera fila se corresponde con la valoración asignada por el índice. Cada celda de la fila muestra la valoración asignada por el índice en el año de la columna.
- Las siguientes filas muestran las valoraciones para cada categoría presentándose por año el top % que ocupa la publicación en ella.

Puede ocurrir que para una categoría o índice no aparezca información. En esos casos, se indica lo siguiente:

- El símbolo '?' informa que para el año concreto no hay ningún dato almacenado de ese índice.
- El símbolo '-' informa que para el año concreto existen datos cargados para el índice pero no para la categoría a la que hace referencia la celda.

Por ejemplo para la revista ACM Computing Surveys, existe información de JCR en el año 2008 para la categoría 'Computer Science, Theory & Method' (con un top 1%) pero en el año 2003, no aparece información alguna de esa categoría. Por otro lado, del año 2009 y 2010 no se tiene información cargada de JCR.

E.2.3 Búsqueda de las valoraciones de una revista dado su ISSN

En el formulario de la página de inicio del buscador de una publicación debe seleccionarse la opción de buscar por ISSN, introducir el código de la revista en formato XXXX-XXXX y seleccionar las macrocategorías donde realizar la búsqueda:

The screenshot shows the 'Publindex: Buscador de publicaciones' interface. At the top, there is a 'LIBESTATS' button. Below it, the 'Búsqueda de:' section contains a dropdown menu with 'Journal (Búsqueda por ISSN)' selected, a text input field containing '0360-0300', and a 'Buscar' button. Below the dropdown, there are three options: 'Journal/Conferencia (Búsqueda por título)', 'Journal (Búsqueda por ISSN)', and 'Journal (Búsqueda por ISSN)'. The 'Journal (Búsqueda por ISSN)' option is highlighted. Below this, there are two sections: 'Índices de JCR donde realizar la búsqueda' and 'Índices de CORE donde realizar la búsqueda'. The JCR section has checkboxes for 'Unknown Category', 'Robotics', 'Computer Science', 'Telecommunications', and 'Multidisciplinary Sciences', all of which are checked. The CORE section has checkboxes for 'Mathematical Sciences', 'Earth Sciences', 'Physical Sciences', 'Environmental Sciences', 'Chemical Sciences', and 'Biological Sciences', all of which are unchecked.

Figura 6: Selección del criterio de búsqueda por ISSN

La búsqueda de la revista por ISSN es única, por lo que se redirige directamente a la pantalla de valoraciones de la publicación que se corresponde con el ISSN introducido en la búsqueda (ver figura 5).

E.2.4 Ver el contenido de una categoría JCR/SJR

Para ver el contenido de una categoría JCR/SJR en un año en concreto se accede desde la página de visualización de resultados mostrada en el apartado E.2.2. Debe seleccionarse el enlace que se encuentra en la celda cuya fila es la categoría que se desea consultar y cuya columna es el año del que se quiera ver la información de la categoría. Por ejemplo, queremos ver las publicaciones de la categoría 'Computer science, theory & method' del año 2008, accediendo desde la página de resultados de 'ACM Computing Surveys':

Journal ACM COMPUTING SURVEYS									
ISSN: 0360-0300									
País: UNITED STATES									
Lenguaje: ENGLISH									
Resultados desde año 2002 a año 2010 <input type="button" value="Buscar"/>									
JCR	2002	2003	2004	2005	2006	2007	2008	2009	2010
Valoración	FI 2,769	FI 7,5	FI 10,037	FI 7,4	FI 4,13	FI 5,25	FI 9,92	?	?
Categorías									
Computer Science, Theory & Method	1/69 (top 1%)	-	1/70 (top 1%)	1/71 (top 1%)	1/75 (top 1%)	1/79 (top 1%)	1/85 (top 1%)	?	?
Unknown Category	-	4/459 (top 1%)	-	-	-	-	-	?	?
CORE	2002	2003	2004	2005	2006	2007	2008	2009	2010
Valoración	?	?	?	?	?	?	A*	A*	A*
Categorías									
Computer Software	?	?	?	?	?	?	-	6/74 (top 8%)	6/92 (top 7%)
Information and Computing Sciences	?	?	?	?	?	?	2/51 (top 4%)	-	-
SJR	2002	2003	2004	2005	2006	2007	2008	2009	2010
Valoración	?	?	0,217	0,282	0,246	0,199	0,266	?	?
Categorías									
Computational Theory and Mathematics	?	?	6/124 (top 5%)	5/124 (top 2%)	6/124 (top 5%)	7/124 (top 6%)	6/124 (top 5%)	?	?

Figura 7: Celda a seleccionar para acceder a la categoría JCR/SJR deseada

Al clicar en el enlace seleccionado, se muestra la información de la categoría. La información mostrada es el listado de enlaces a las publicaciones que se incluyen en la categoría en ese año ordenadas de mayor a menor valoración. La publicación desde la que se ha accedido a la página de la categoría aparece resaltada para indicar que se ha accedido desde ella. En el ejemplo de esta guía aparece resaltada la revista 'ACM Computing Surveys'.

[JCR 2008]:Categoría Computer Science, Theory & Method
 Se han encontrado 85 publicaciones en esta categoría

Las publicaciones aparecen ordenadas según el factor de impacto

1. [ACM COMPUTING SURVEYS \(0360-0300\) \(9,92\)](#)
2. [IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION \(1089-778X\) \(3,736\)](#)
3. [IEEE TRANSACTIONS ON NEURAL NETWORKS \(1045-9227\) \(3,726\)](#)
4. [QUANTUM INFORMATION & COMPUTATION \(1533-7146\) \(3,379\)](#)
5. [EVOLUTIONARY COMPUTATION \(1530-9304\) \(3\)](#)
6. [HUMAN-COMPUTER INTERACTION \(1532-7051\) \(2,905\)](#)
7. [ACM TRANSACTIONS ON COMPUTATIONAL LOGIC \(1529-3785\) \(2,766\)](#)
8. [COMMUNICATIONS OF THE ACM \(0001-0782\) \(2,646\)](#)
9. [IEEE-ACM TRANSACTIONS ON NETWORKING \(1063-6692\) \(2,576\)](#)
10. [ACM TRANSACTIONS ON MULTIMEDIA COMPUTING COMMUNICATIONS AND APPLICATIONS \(1551-6857\) \(2,465\)](#)
11. [ACM TRANSACTIONS ON COMPUTER SYSTEMS \(0734-2071\) \(2,391\)](#)
12. [JOURNAL OF THE ACM \(0004-5411\) \(2,339\)](#)
13. [JOURNAL OF CRYPTOLOGY \(1432-1378\) \(2,265\)](#)

Figura 8: Categoría 'Computer Science, Theory & Method' en el año 2008

Como vemos, de cada publicación se muestra el nombre y la valoración.

Los pasos a dar y la apariencia de una categoría SJR es exactamente la misma.

E.2.5 Ver el contenido de una categoría CORE

Por ejemplo, queremos ver las publicaciones de la categoría CORE 'Information and Computing Sciences' del año 2008, accediendo desde la página de resultados de 'ACM Computing Surveys':

Journal ACM COMPUTING SURVEYS

ISSN: 0360-0300
 País: UNITED STATES
 Lenguaje: ENGLISH

Resultados desde año 2002 a año 2010

JCR	2002	2003	2004	2005	2006	2007	2008	2009	2010
Valoración	FI 2,769	FI 7,5	FI 10,037	FI 7,4	FI 4,13	FI 5,25	FI 9,92	?	?
Categorías									
Computer Science, Theory & Method	1/69 (top 1%)	-	1/70 (top 1%)	1/71 (top 1%)	1/75 (top 1%)	1/79 (top 1%)	1/85 (top 1%)	?	?
Unknown Category	-	4/459 (top 1%)	-	-	-	-	-	?	?
CORE	2002	2003	2004	2005	2006	2007	2008	2009	2010
Valoración	?	?	?	?	?	?	A*	A*	A*
Categorías									
Computer Software	?	?	?	?	?	?	-	6/74 (top 8%)	6/92 (top 7%)
Information and Computing Sciences	?	?	?	?	?	?	2/51 (top 4%)	-	-
SJR	2002	2003	2004	2005	2006	2007	2008	2009	2010
Valoración	?	?	0,217	0,282	0,246	0,199	0,266	?	?
Categorías									
Computational Theory and Mathematics	?	?	6/124 (top 5%)	3/124 (top 2%)	6/124 (top 5%)	7/124 (top 6%)	6/124 (top 5%)	?	?

Figura 9: Celda a seleccionar para acceder a la categoría CORE deseada

Al clicar en el enlace indicado en la imagen, nos aparece la información de la categoría CORE. Hay que decir que en este índice no se puede determinar si una publicación es mejor o

peor que otra si tienen la misma valoración. Por lo tanto, los resultados de la categoría se muestran ordenados por valoración decreciente (A*, A, B, C,...) y dentro de cada valoración las publicaciones ordenadas alfabéticamente pero sin indicar ningún orden. Sólo se consideran en la categoría aquellas publicaciones del mismo tipo que el de la publicación desde la que se accede.

En el ejemplo, la categoría CORE seleccionada tendría la siguiente apariencia:

[CORE 2008]:Categoría Information and Computing Sciences

Se han encontrado 51 publicaciones en esta categoría

Las publicaciones aparecen agrupadas por valoración y dentro de ésta ordenadas alfabéticamente

Valoración 'A*' (2 publicaciones)

- [ACM COMPUTING SURVEYS \(0360-0300\)](#)
- [JOURNAL OF THE ASSOCIATION FOR COMPUTING MACHINERY \(ISSN no disponible\)](#)

Valoración 'A' (9 publicaciones)

- [COMPUTER JOURNAL \(1460-2067\)](#)
- [COMPUTER RESEARCH AND DEVELOPMENT \(ISSN no disponible\)](#)
- [COMPUTING \(0010-485X\)](#)
- [CONCURRENCY AND COMPUTATION-PRACTICE & EXPERIENCE \(1532-0626\)](#)
- [DECISION SCIENCES \(0011-7315\)](#)
- [IBM JOURNAL OF RESEARCH AND DEVELOPMENT \(0018-8646\)](#)
- [IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING \(1041-4347\)](#)
- [INFORMATION AND COMPUTATION \(1090-2651\)](#)
- [INTEGRATION-THE VLSI JOURNAL \(0167-9260\)](#)

Valoración 'B' (6 publicaciones)

- [COMMUNICATIONS OF THE ACM \(0001-0782\)](#)
- [IEEE COMPUTER \(ISSN no disponible\)](#)
- [INFORMATION SCIENCES \(0020-0255\)](#)
- [JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY \(1000-9000\)](#)
- [JOURNAL OF RESEARCH AND PRACTICE IN INFORMATION TECHNOLOGY \(1443-458X\)](#)
- [KNOWLEDGE AND INFORMATION SYSTEMS \(0219-1377\)](#)

Valoración 'C' (34 publicaciones)

- [ACIS INTERNATIONAL JOURNAL OF COMPUTER AND INFORMATION SCIENCE \(ISSN no disponible\)](#)

Figura 10: Categoría Information and Computing Sciences del año 2008

Debido a que las publicaciones con una misma valoración no pueden ser comparadas unas con otras, calcular el top de una publicación en una categoría CORE no es tan sencillo. Pongamos por ejemplo que queremos calcular el top de la revista 'Computer Journal'. El top es calculado considerando el número total de revistas con valoración mayor o igual de la que se calcula el porcentaje con respecto al total de revistas de la categoría. Por tanto, el top de 'Computer Journal' es igual a 11 revistas (2 A* + 9 A) / 51 (total de revistas en la categoría). Notar que en el caso de mostrar conferencias el procedimiento es el mismo.

Al igual que para las categorías JCR/SJR, la publicación desde la cual se ha accedido a la información de la categoría aparece resaltada.

E.2.6 Mantenimiento de publicaciones

Este servicio se incluye dentro del buscador ya que se accede al subsistema de mantenimiento desde la página de resultados de la búsqueda de publicaciones. En el caso de que haya más de una publicación en el listado de coincidencias también aparece un enlace con título 'Fusionar con otras publicaciones' para cada entrada de la lista. Por ejemplo, pongamos que se han buscado las publicaciones con la subcadena 'pervasive' en su título y que el sistema encuentra los siguientes resultados para las revistas:

Se han encontrado 13 resultados para la clave 'pervasive':

Se listan a continuación las coincidencias alfabéticamente junto con sus valoraciones más recientes

Journals (4 resultados)

- [IEEE PERVASIVE COMPUTING \(1536-1268\) \(JCR 2,615\) \(CORE A\) \(SJR 0,15\)](#)
[Fusionar con otras publicaciones]
- [INTERNATIONAL JOURNAL OF PERVASIVE COMPUTING AND COMMUNICATIONS \(1742-7371\) \(CORE C\)](#)
[Fusionar con otras publicaciones]
- [INTERNATIONAL JOURNAL OF PERVASIVE COMPUTING AND COMMUNICATIONS \(ISSN no disponible\) \(CORE C\)](#)
[Fusionar con otras publicaciones]
- [PERVASIVE AND MOBILE COMPUTING \(1574-1192\) \(CORE B\) \(SJR 0,046\)](#)
[Fusionar con otras publicaciones]

Conferencias (0 resultados)

- [IEEE INTERNATIONAL CONFERENCE ON PERVASIVE COMPUTING AND COMMUNICATIONS \(PERCOM\) \(CORE A\)](#)

Figura 11: Coincidencias de revistas para la subcadena 'pervasive'

Como puede comprobarse, existen dos revistas con el mismo nombre que por error han sido almacenadas dos veces cuando en realidad hacen referencia a la misma publicación. Clicamos en el enlace de fusionar la primera revista duplicada y llegamos a la pantalla de búsqueda de la publicación destino:

Publindex Biblioteca Digital: Mantenimiento de datos

Publicación original a fusionar: INTERNATIONAL JOURNAL OF PERVASIVE COMPUTING AND COMMUNICATIONS

Publicacion resultado de la fusión:

Journal/Conferencia (Búsqueda por título)

Journal/Conferencia (Búsqueda por título)

Journal (Búsqueda por ISSN)

Figura 12: Página de búsqueda de la publicación destino de la fusión

Introducimos el mismo criterio de búsqueda para encontrar las coincidencias de revistas (sólo se permite la fusión con publicaciones del mismo tipo) y seleccionar la publicación destino

de la fusión. Una vez introducidos los datos, se pulsa el botón de búsqueda y se llega a la pantalla de selección del destino de la fusión:

Publindex Biblioteca Digital: Mantenimiento de datos

Publicación original a fusionar: INTERNATIONAL JOURNAL OF PERVASIVE COMPUTING AND COMMUNICATIONS

Publicacion resultado de la fusión:

INTERNATIONAL JOURNAL OF PERVASIVE COMPUTING AND COMMUNICATIONS ▾

Figura 13: Selección de la publicación destino de la fusión

Esta página presenta un desplegable donde se selecciona la publicación destino de la fusión. Una vez seleccionada, se pulsa guardar para realizar la fusión o cancelar para regresar al inicio. El mensaje informativo de éxito de la fusión es el siguiente:

Fusión realizada con éxito

Los cambios han sido establecidos sobre la publicación
INTERNATIONAL JOURNAL OF PERVASIVE COMPUTING AND COMMUNICATIONS

[Realizar nueva búsqueda](#)

Figura 14: Mensaje de éxito de la fusión

Los cambios realizados en la fusión se almacenan en el fichero 'FusionLog.sql'. Este fichero es en realidad un script que al ejecutarse en base de datos permite recuperar los cambios de las fusiones realizadas en el caso de restaurar al estado inicial los datos almacenados. Este fichero se encuentra en el directorio raíz donde se ejecuta la aplicación en la carpeta Publindex/FusionLog.sql. Un ejemplo de fichero de log es el siguiente:

```
Update publicadoencat set id_publicacion = 923 where id_publicacion = 935;  
  
Delete from publicacion where id_publicacion = 935;  
  
Update publicadoencat set id_publicacion = 923 where id_publicacion = 17527;
```

```
Update publicadoencat set id_publicacion = 923 where id_publicacion =  
17527;
```

```
Delete from publicacion where id_publicacion = 17527;
```

E.3 Analizador de un listado de publicaciones a través de referencias bibliográficas

Este apartado muestra los pasos a seguir para analizar un listado de citas bibliográficas y obtener el listado de publicaciones a las que hacen referencia.

E.3.1 Acceder al analizador

Para acceder a este buscador, debe seleccionarse la opción de 'Analizar un listado de referencias bibliográficas' del menú principal:

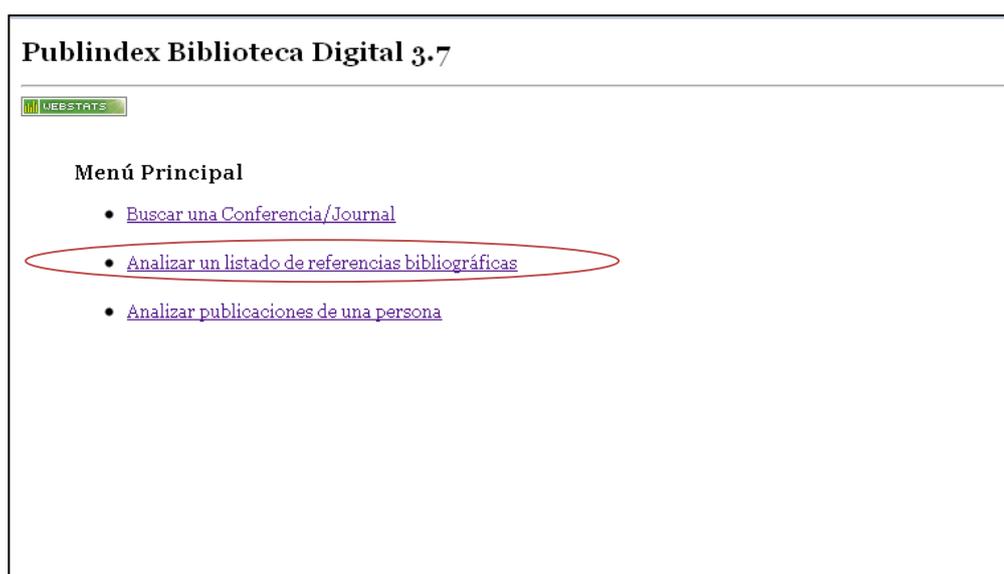


Figura 15: Página de inicio de la aplicación

E.3.2 Búsqueda de un listado de publicaciones

Debe introducirse el listado de referencias en formato texto y/o BibTex a analizar. Por ejemplo, supongamos que introducimos las siguientes referencias bibliográficas:

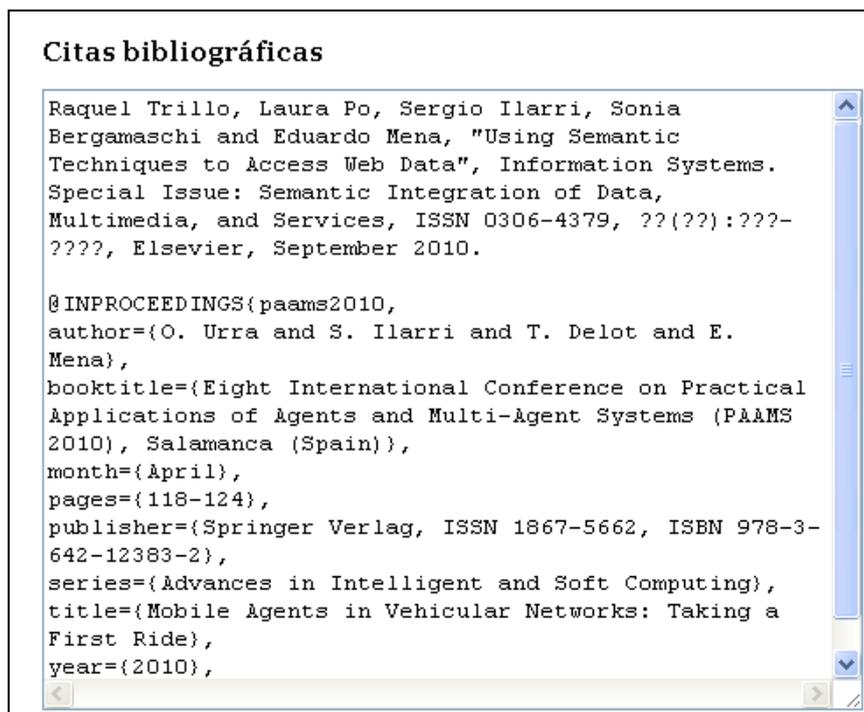


Figura 16: Página de inicio del buscador de citas

Al pulsar el botón de buscar, se procede a realizar el análisis de las referencias. En este caso se han introducido formatos válidos, por lo que se redirige a la página de resultado de la búsqueda de las publicaciones. Esta página muestra en formato tabla los resultados, indicando los datos más relevantes de las citas analizadas (título, año, nombre de la publicación extraído y autores) y las publicaciones que referencian.

Se han encontrado 2 referencias. Se muestra cada una de las citas asociadas al evento coincidente en el sistema

Datos referencia	Publicaciones resultado
Autores: Raquel Trillo, Laura Po, Sergio Ilarri, Sonia Bergamaschi and Eduardo Mena Año ref: 2010 Cita: Using Semantic Techniques to Access Web Data Journal/Conf: Information Systems. Special Issue: Semantic Integration of Data	INFORMATION SYSTEMS (0306-4379) (JCR 1,66) (CORE A*)
Autores: O. Urra and S. Ilarri and T. Delot and E. Mena Año ref: 2010 Cita: Mobile Agents in Vehicular Networks: Taking a First Ride Journal/Conf: Eight International Conference on Practical Applications of Agents and Multi-Agent Systems (PAAMS 2010)	Cannot extract the event

Figura 17: Resultado de la búsqueda de publicaciones

El resultado de la búsqueda puede no encontrar resultados, como ha ocurrido en el ejemplo con la segunda cita. En ese caso, se muestra el mensaje 'Cannot extract the event'. En el caso de que la búsqueda encuentre resultados, se presenta un enlace a la página de datos y valoraciones de la publicación y las valoraciones de los últimos años asignadas por los índices.

La página de resultados de cada publicación es la misma que la de resultados expuesta en el apartado E.2.2, con la única diferencia de que los datos pertenecientes al año de la cita aparecen resaltados en color verde. Esto es debido a que se pretende resaltar qué calidad presentaba el evento en el año que el artículo fue publicado. Veamos en el ejemplo la

aparición de la página de datos de la revista 'Information Systems' que ha sido referenciada a través de la primera cita introducida (ver figura 17).

Journal INFORMATION SYSTEMS									
ISSN: 0306-4379									
País: ENGLAND									
Lenguaje: MULTI-LANGUAGE									
Resultados desde año 2002 a año 2010 <input type="button" value="Buscar"/>									
JCR	2002	2003	2004	2005	2006	2007	2008	2009	2010
Valoración	FI 2	FI 3,327	FI 2,343	FI 1,027	FI 1,887	FI 1,681	FI 1,66	?	?
Categorías									
Computer Science, Information Systems	7/76 (top 9%)	-	10/78 (top 13%)	36/89 (top 43%)	16/87 (top 18%)	20/92 (top 22%)	33/99 (top 33%)	?	?
Unknown Category	-	18/159 (top 4%)	-	-	-	-	-	?	?
CORE	2002	2003	2004	2005	2006	2007	2008	2009	2010
Valoración	?	?	?	?	?	A	?	A*	A*
Categorías									
Unknown Category	?	?	?	?	?	121/425 (top 28%)	?	-	-
Information Systems	?	?	?	?	?	-	?	9/129 (top 7%)	18/201 (top 9%)

Figura 18: Resultados de la revista con los datos del año de la cita resaltados

E.3.3 Formatos válidos de las citas

Las citas que el sistema puede reconocer tienen un formato concreto. El formato texto reconocido es el siguiente:

[Autores], "[Título cita]", [Título publicación], [Resto de información]

En cuanto al formato BibText, hay que destacar que dependiendo del tipo de evento en el que se publique, el formato que se presenta es diferente y por tanto, el analizador deberá reconocer unos patrones concretos. En concreto, el analizador reconoce los formatos BibText @INPROCEEDINGS y @ARTICLE (Ref. [BibText]). A continuación se muestran unos ejemplos:

```
@INPROCEEDINGS{iwann09-Urra,
author={O. Urra and S. Ilari and E. Mena},
booktitle={10th International Work-Conference on Artificial Neural Networks (IWANN'09), Special Session on Practical Applications of Agents and Multi-Agent Systems, Salamanca (Spain)},
month={June},
pages={627--634},
publisher={Springer Verlag, ISSN 0302-9743, ISBN 978-3-642-02477-1},
series={Lecture Notes in Computer Science},
title={Agents Jumping in the Air: Dream or Reality?},
volume={5517},
year={2009},
idbdi={yes},
```

```
file={iwann09.pdf},
keywords={Mobile Agents},
}

@ARTICLE{is08,
author={S. Ilarri and E. Mena and A. Illarramendi},
journal={Information Sciences, ISSN 0020-0255},
month={May},
number={9},
pages={2105--2127},
publisher={Elsevier, doi:"10.1016/j.ins.2007.12.015"},
title={Using Cooperative Mobile Agents to Monitor Distributed and Dynamic Environments},
volume={178},
year={2008},
idbdi={yes},
file={is08.pdf},
keywords={Mobile Computing, ANTARCTICA, Mobile Agents, Location-dependent queries},
}
```

No obstante, puede ocurrir que aunque se sigan estos formatos se introduzcan caracteres no aceptados, símbolos reservados ('{' y '}'), etc. En ese caso se muestra un mensaje de error indicando dónde se ha encontrado el fallo de formato:



Figura 19: Mensaje de error de cita no válida

E.4 Analizador de la calidad del trabajo de un autor

Este apartado muestra los pasos a seguir para analizar la calidad del trabajo de un autor a través del listado de eventos donde ha publicado.

E.4.1 Acceder al analizador

Para acceder a este buscador, debe seleccionarse la opción de 'Analizar publicaciones de una persona' del menú principal:

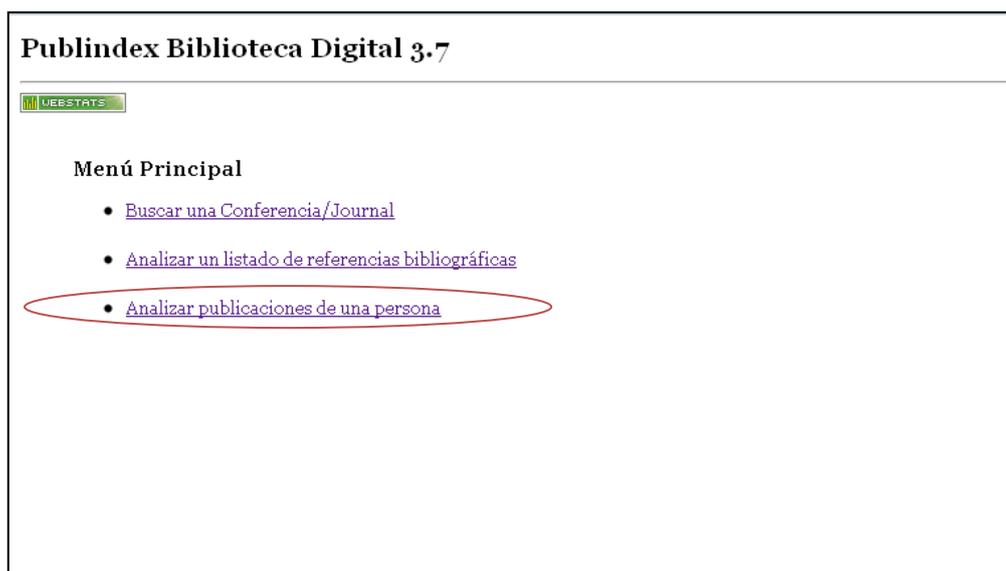


Figura 20: Página de inicio de la aplicación

E.4.2 Búsqueda del trabajo de un autor

La página de inicio del buscador contiene un formulario donde introducir el nombre del autor a consultar. Recordemos que los autores de los que se dispone de información son los componentes del grupo de investigación Sistemas de Información Distribuidos [SID]. Por ejemplo, queremos buscar el trabajo de 'Carlos Bobed':

Publindex: Búsqueda de citas bibliográficas por autor



Nombre de autor:

Buscar sólo publicaciones de miembros del SID

Introduzca el nombre de un autor

[Volver a Inicio](#)

Figura 21: Página de inicio del buscador de autores

Al realizar la búsqueda, se muestra el listado de referencias bibliográficas y las publicaciones a las que hacen referencia, de la misma manera que se presentaban en el analizador de publicaciones a través de citas (apartado E.3.2). La página de resultados es la siguiente:

Citas bibliográficas de Carlos Bobed	
<i>Se han encontrado 7 referencias. Se muestra cada una de las citas asociadas al evento coincidente en el sistema</i>	
Datos referencia	Publicaciones resultado
Autores: Carlos Bobed and Sergio Ilarri and Eduardo Mena Año ref: 2010 Cita: Exploiting the Semantics of Location Granules in Location-Dependent Queries Journal/Conf: East-European Conference on Advances in Databases and Information Systems (ADBIS'10)	SYMPOSIUM ON ADVANCES IN DB AND INFORMATION SYSTEMS (ADBIS) (CORE B)
Autores: Carlos Bobed, Sergio Ilarri, Eduardo Mena Año ref: 2010 Cita: Distributed Mobile Computing: Development of Distributed Applications Using Mobile Agents Journal/Conf: International Conference on Parallel and Distributed Computing (PDPTA'10)	INTERNATIONAL CONFERENCE ON PARALLEL AND DISTRIBUTED PROCESSING TECHNIQUES AND APPLICATIONS (PDPTA) (CORE B)
Autores: Carlos Bobed and Eduardo Mena Año ref: 2010 Cita: Enhancing the Discovery of Web Services: A Keyword-oriented Multontology Reconciliation Journal/Conf: International Conference on Parallel and Distributed Computing (PDPTA'10)	INTERNATIONAL CONFERENCE ON PARALLEL AND DISTRIBUTED PROCESSING TECHNIQUES AND APPLICATIONS (PDPTA) (CORE B)
Autores: S. Ilarri and A. Corral and C. Bobed and E. Mena Año ref: 2009 Cita: Probabilistic Granule-Based Inside and Nearest Neighbor Queries Journal/Conf: East-European Conference on Advances in Databases and Information Systems (ADBIS'09)	SYMPOSIUM ON ADVANCES IN DB AND INFORMATION SYSTEMS (ADBIS) (CORE B)
Autores: Carlos Bobed and Raquel Trillo and Eduardo Mena and Jorge Bernad Año ref: 2008 Cita: Semantic Discovery of the User Intended Query in a Selectable Target Query Language Journal/Conf: International Conference on Web Intelligence (WI 2008)	IEEE/WIC/ACM INTERNATIONAL CONFERENCE ON WEB INTELLIGENCE INTERNATIONAL (WI) (CORE B)
Autores: S. Ilarri and E. Mena and C. Bobed Año ref: 2007 Cita: Processing Location-Dependent Queries with Location Granules Journal/Conf: On the Move to Meaningful Internet Systems 2007: OTM 2007 Workshops (PerSys)	Cannot extract the event
Autores: C. Bobed and E. Mena Año ref: 2007	

Figura 22: Página de resultado de la búsqueda de publicaciones del autor

E.5 Carga del índice de calidad CSCR

Este apartado contiene los pasos a dar para incorporar el nuevo índice de calidad CSCR, que debido a plazos de tiempo, no pudo ser almacenado en la base de datos del sistema Publindex.

La única fuente de información de CSCR es su propia página web, que contiene los rankings para un conjunto de categorías. La dirección URL de esta página web es la siguiente: <http://www.cs-conference-ranking.org/conferencerankings/alltopics.html>. Por tanto, la fuente de información va a ser un fichero HTML del cual se extraerán los fragmentos que interesen.

E.5.1 Antes de empezar

Para comenzar es necesario crear un proyecto Java con los siguientes paquetes y clases:

- Paquete que contenga las clases DAO que facilitan los métodos de edición y consulta de la base de datos
- Paquete con las entidades de negocio POJO a almacenar.
- Los ficheros de configuración de Hibernate.
- Un paquete que contenga la clase principal. Esta clase posee el método de inicio *main* que será el encargado de llamar a los métodos de reconocimiento de la información CSCR.
- Un paquete que contenga las clases que implementen el reconocimiento de la información de CSCR.

Una vez creado el proyecto ya se puede comenzar a cargar la información de CSCR.

E.5.2 Información a almacenar en base de datos

En primer lugar, se realizará la carga en base de datos de los índices y categorías que es necesario que estén almacenadas antes de comenzar la carga de las valoraciones para una fecha en concreto:

- Almacenar una nueva entrada en la tabla de Índice con el valor de CSCR.
- Almacenar las macrocategorías de CSCR para el año al que pertenezcan las valoraciones. Estas macrocategorías deben ser almacenadas para poder seleccionar si se incluyen en la búsqueda en la página de inicio del buscador de una publicación. Las entradas a guardar en la tabla de macrocategoría son:
 - Databases / Knowledge and Data Management / Data Security / Web / Mining
 - Artificial Intelligence / Machine Learning / Robotics / Human Computer Interaction

- Architecture / Hardware / High-Performance Computing / Tools / Operating Systems
- Applications / Education / Software / Theory / Communications / Graphics / Bioinformatics
- Almacenar las categorías de CSCR en la tabla de base de datos con el mismo nombre. Estas categorías deben referenciar al índice y macrocategorías creadas en los puntos anteriores.

Una vez almacenados estos datos, se procede a cargar la información de las categorías CSCR.

E.5.3 Cómo procesar los datos

Una vez creadas las entradas en base de datos, es necesario extraer el texto HTML que contiene el listado de publicaciones para cada categoría. De cada categoría puede crearse un fichero de texto que contenga su listado de conferencias en código HTML. Crear estos ficheros no será muy costoso, ya que resulta muy sencillo identificar los listados al estar contenidos entre los tags y . El formato que presenta es el siguiente:

```
<ul>
...
<br/>
Siglas conferencia ':' Título conferencia '(' Valoración ')'
<br/>
...
</ul>
```

La valoración está comprendida entre paréntesis y es un número con dos cifras decimales comprendido entre los valores 0.00 y 0.99. Las siglas y el título de una conferencia van separados por dos puntos.

Debe implementarse un analizador léxico (.lex) y sintáctico (.cup) que reconozca este formato. Estos ficheros deben ir ubicados en el paquete destinado a aquellas clases que realicen el reconocimiento de las categorías CSCR en el proyecto Java creado anteriormente. A la vez que se va reconociendo la información, se irá almacenando en base de datos a través de las clases DAO. Un ejemplo de gramática escrita en pseudocódigo y que reconoce esta información es la siguiente:

```
categoria:  ABRE_UL listado CIERRA_UL;

listado:   listado conf BR |
          conf BR;

conf:     CADENA: siglas ':' CADENA: titulo '(' CADENA: valor ')'
          { /* Guardar información de conferencia y valor */ }
```

Los datos a almacenar serán:

- La conferencia, de la que se conocen su título y siglas. Antes de almacenarla, es necesario comprobar si ya existe en la base de datos. En caso de que no sea así, debe ser almacenada a través de los métodos de inserción que facilitan las clases ConferenciaDAO y PublicaciónDAO. Si estos métodos no se ajustan, será necesario modificarlos o añadir nuevos que permitan insertar conferencias a través de la información facilitada por CSCR.
- Además de la conferencia, es necesario guardar la valoración a través de la clase PublicadasDAO. Para ello, tiene que tenerse en cuenta qué categoría está siendo procesada. Los datos que deben guardarse de cada valoración son:
 - Id de la conferencia que ha sido reconocida por el analizador.
 - Id de la categoría que está siendo procesada actualmente.
 - Valoración y factor de impacto: valoración reconocida por el analizador.
 - Posición: posición de la publicación en la categoría.

El método que deberá llamarse para realizar el reconocimiento de cada fichero es el método *Parser* del analizador sintáctico creado. Este método será llamado desde la clase de inicio *main* tantas veces como ficheros de categorías haya.

