

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
 Ese apoyo que siempre quiso tener...

1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
 HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
 Gestor de contenidos (Alfresco)
 Aplicaciones híbridas

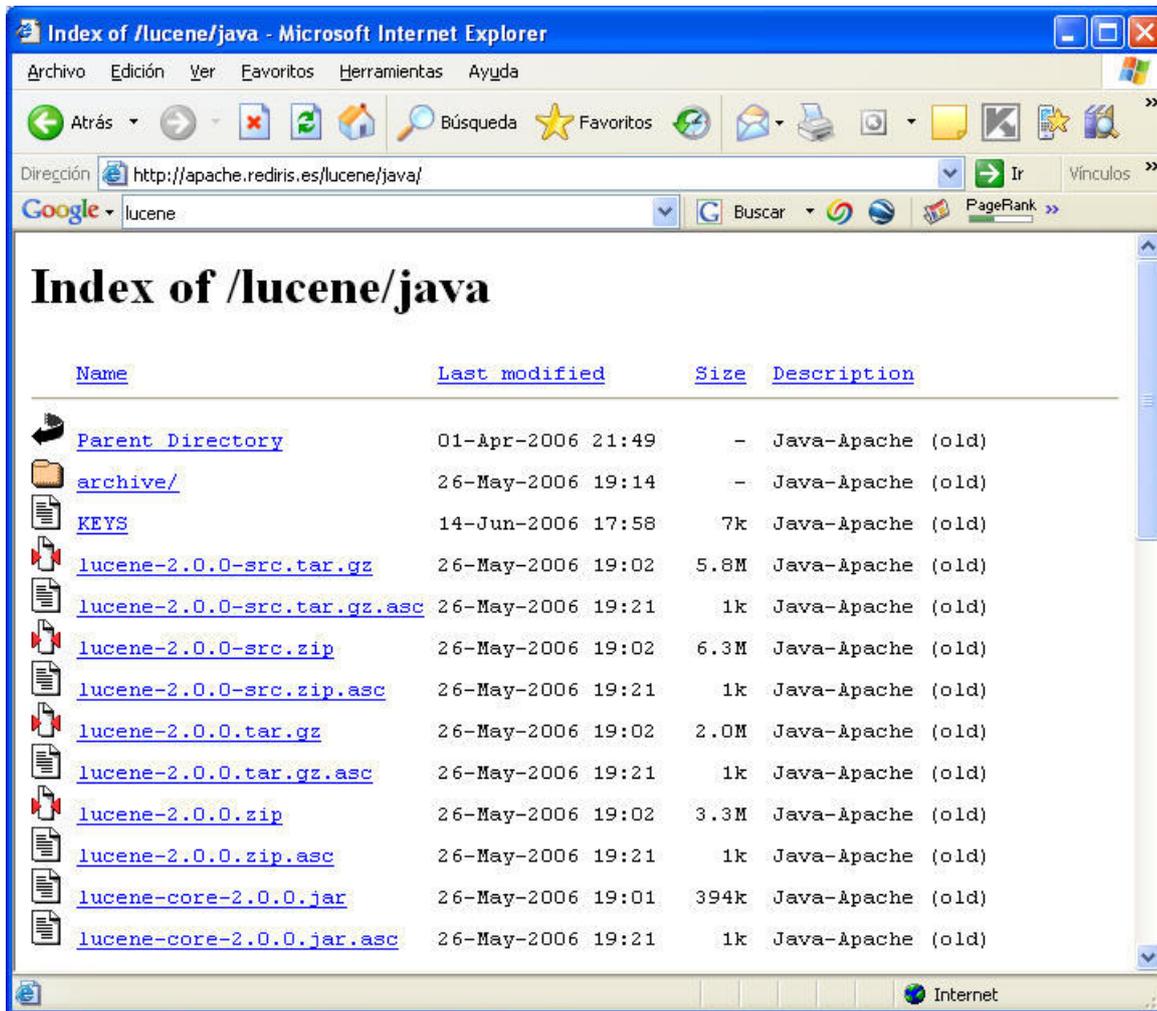
Tareas programadas (Quartz)
 Gestor documental (Alfresco)
 Inversión de control (Spring)

Control de autenticación y
 acceso (Spring Security)
 UDDI
 Web Services
 Rest Services
 Social SSO
 SSO (Cas)

JPA-Hibernate, MyBatis
 Motor de búsqueda empresarial (Solr)
 ETL (Talend)

Dirección de Proyectos Informáticos.
 Metodologías ágiles
 Patrones de diseño
 TDD

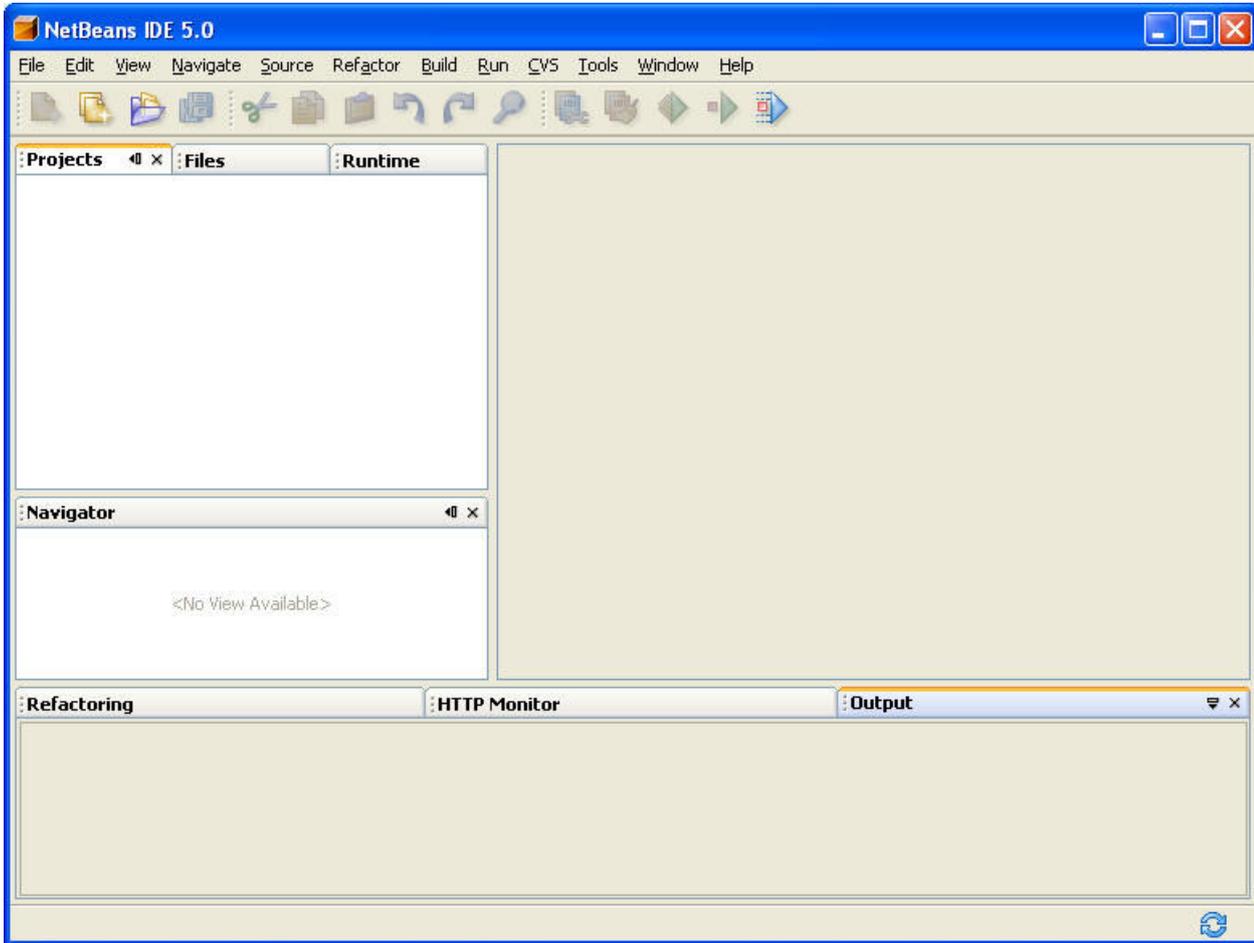
BPM (jBPM o Bonita)
 Generación de informes (JasperReport)
 ESB (Open ESB)



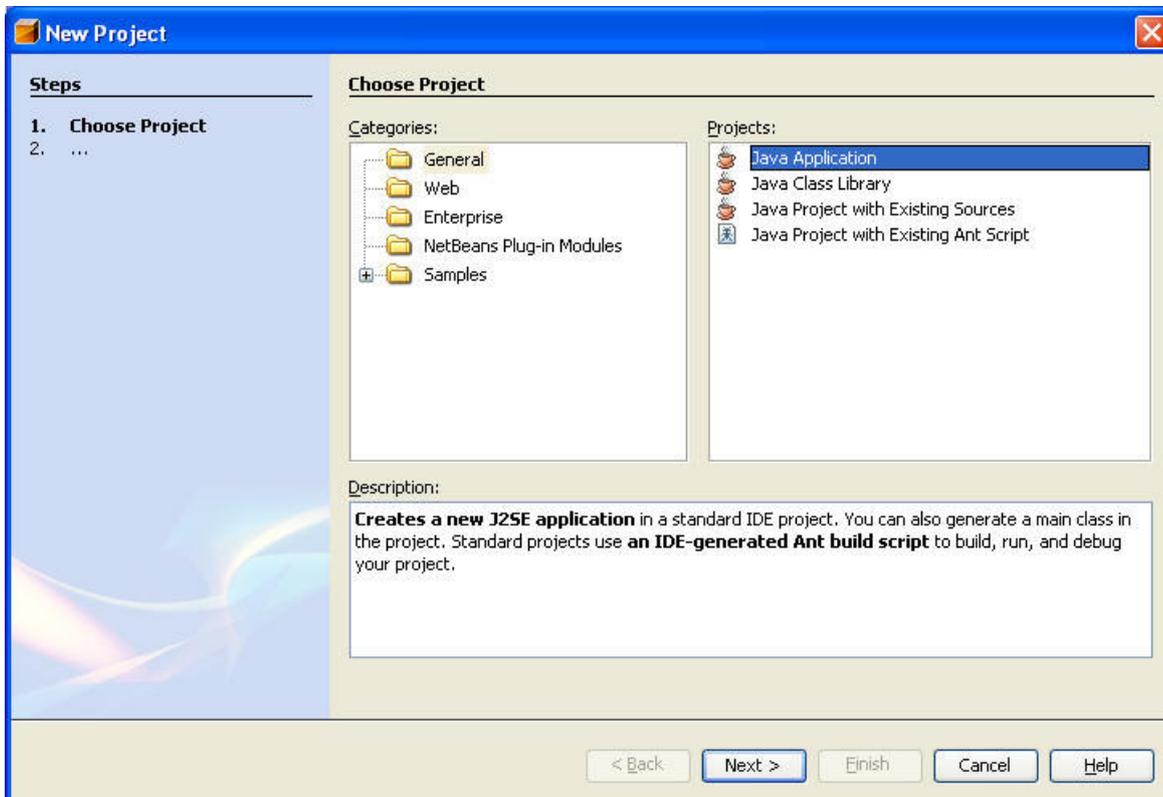
Elegimos el fichero deseado.



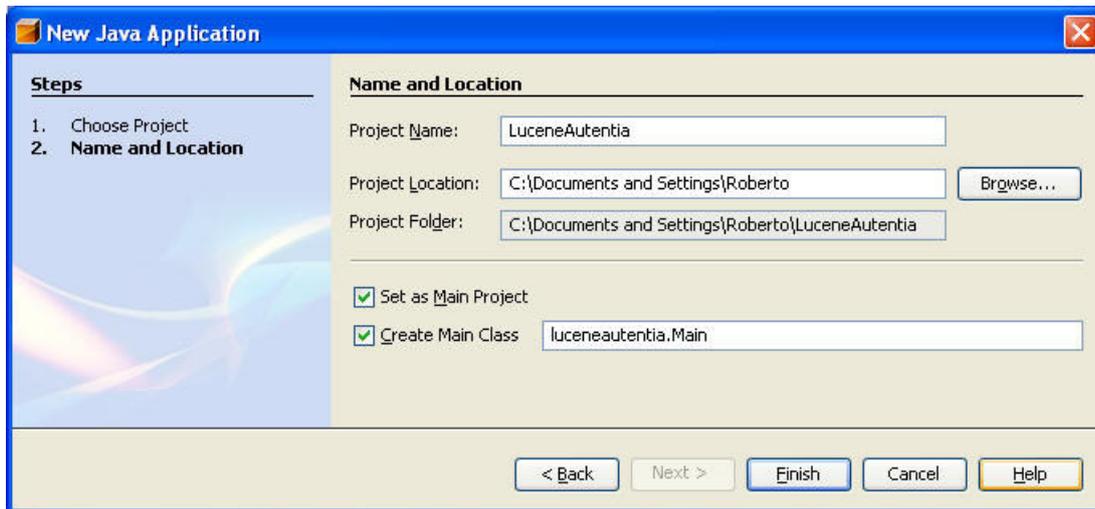
Arrancamos NetBeans 5 y creamos un nuevo proyecto en el que realizaremos las pruebas básicas.



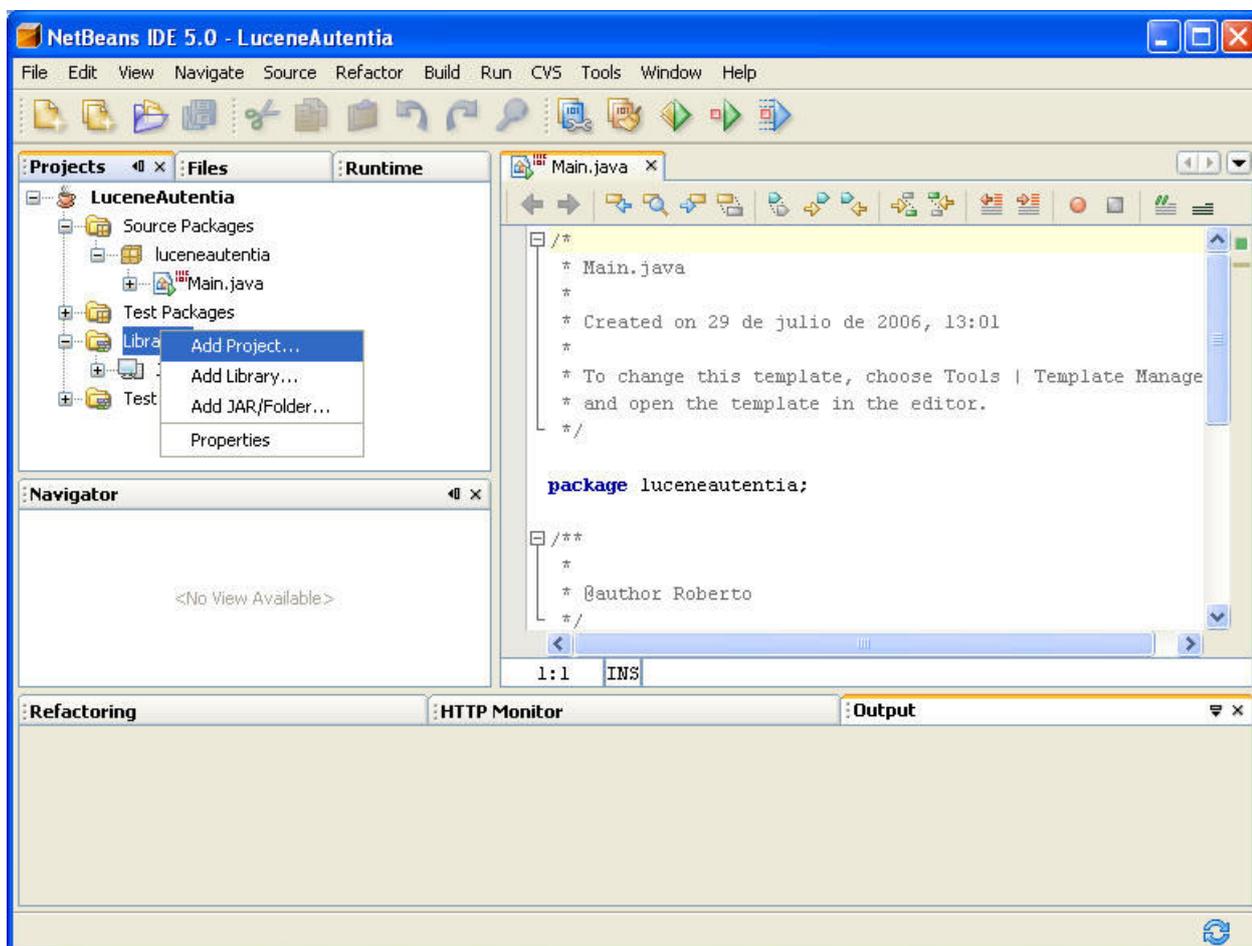
Elegimos una aplicación Java en en Wizard de NetBeans



Asignamos el nombre de proyecto: LuceneAutentia



Añadimos los ficheros jar de lucene al proyecto. Ojo, que casi todas las consultas que me hace la gente en www.adictosaltrabajo.com, son por esta misma causa: problemas con el classpath



Creamos un directorio que servirá de repositorio para la indexación



Y ahora escribimos nuestra aplicación (ojo que es para Lucene 2.0 y os puede chocar un poco la parte en azul)

```

/*
 * Created on 29 de julio de 2006, 13:01
 *
 * Roberto Canales Mora
 * www.adictosaltrabajo.com
 */
package luceneautentia;

import java.io.IOException;
import java.util.*;
import org.apache.lucene.document.*;
import org.apache.lucene.document.Field.*;
import org.apache.lucene.analysis.*;
import org.apache.lucene.index.*;
import org.apache.lucene.analysis.standard.*;
import org.apache.lucene.store.Directory;
import org.apache.lucene.store.IndexInput;
import org.apache.lucene.store.IndexOutput;
import org.apache.lucene.store.Lock;

/**
 *
 * @author Roberto
 */
public class Main {

public Main() {
}

public static void main(String[] args) {

gestorLucene ejemplo = new gestorLucene();

Document persona1 = ejemplo.crearDocumento
("Carlos","Autentia","www.autentia.com","655991172");
Document persona2 = ejemplo.crearDocumento("Jose
Maria","Autentia","www.autentia.com","918040181");

try
{
ejemplo.indexar(persona1);
ejemplo.indexar(persona2);
}
catch (Exception e)
{
System.out.println("Error en la aplicación " + e.toString());
e.printStackTrace();
}
}
}

class gestorLucene
{
String directorioIndexacion = "lucene-index-autentia";

void indexar(Document documento) throws Exception
{
// org.apache.lucene.index.IndexReader.unlock(new );

Analyzer analizador = new StandardAnalyzer();
IndexWriter writer = new IndexWriter(directorioIndexacion,
analizador, true); // ojo
writer.setUseCompoundFile(false);
writer.addDocument(documento);
}
}
}

```

```

writer.optimize();
writer.close();
}

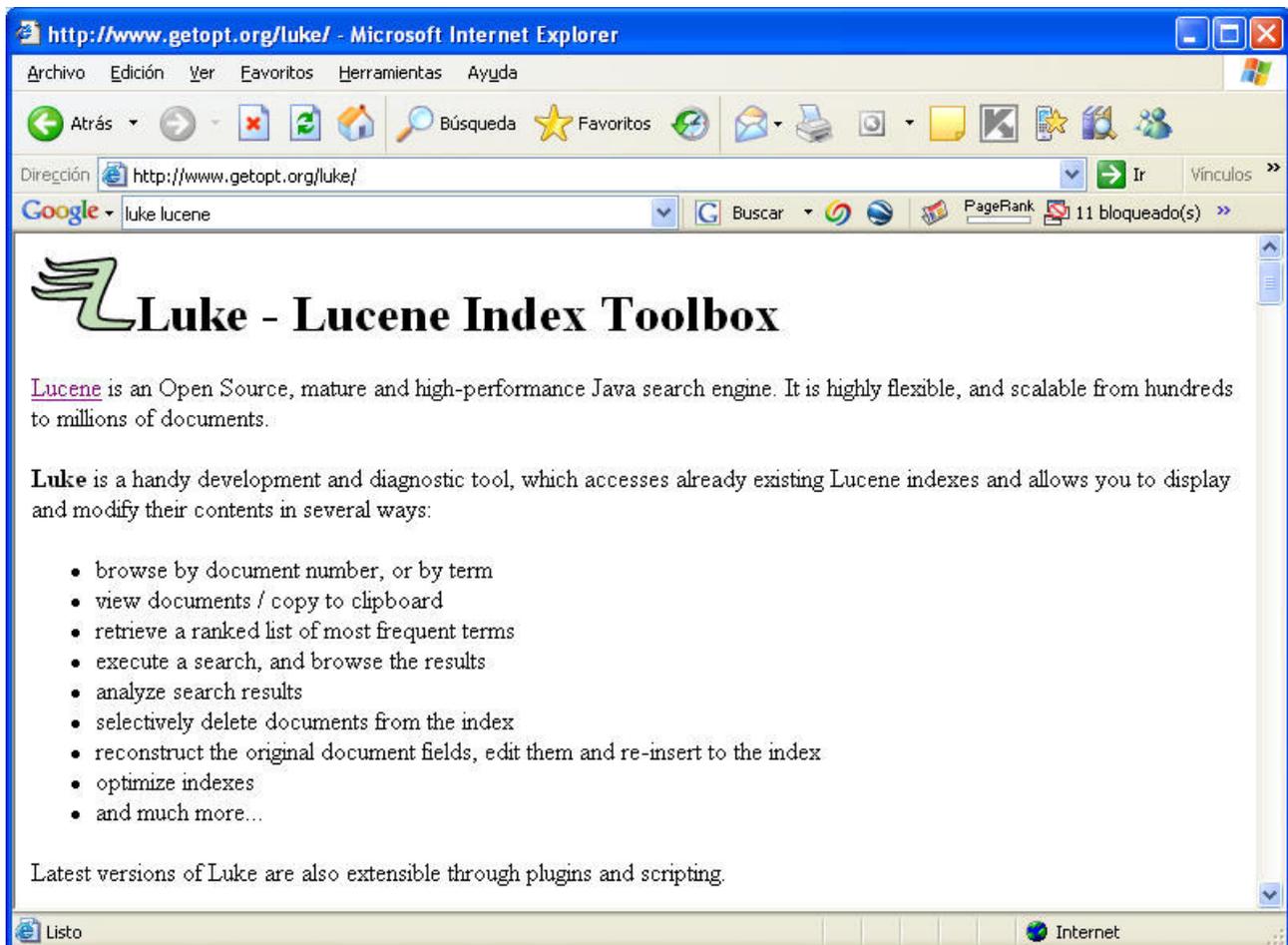
Document crearDocumento( String nombre, String empresa,
String web,String telefono )
{
Document document = new Document();
document.add(new Field("nombre", nombre, Store.NO,
Index.TOKENIZED));
document.add(new Field("empresa", empresa, Store.NO,
Index.TOKENIZED));
document.add(new Field("web", web, Store.NO,
Index.UN_TOKENIZED));
document.add(new Field("telefono", telefono, Store.NO,
Index.UN_TOKENIZED));

return document;
}
}

```

Usaremos Luke - Lucene Index Toolkit

Para asegurarnos si ha funcionado o no, vamos a descargarnos un monitor que nos permita navegar (e incluso consultar y modificar) la información indexada: Luke - Lucene Index Toolkit

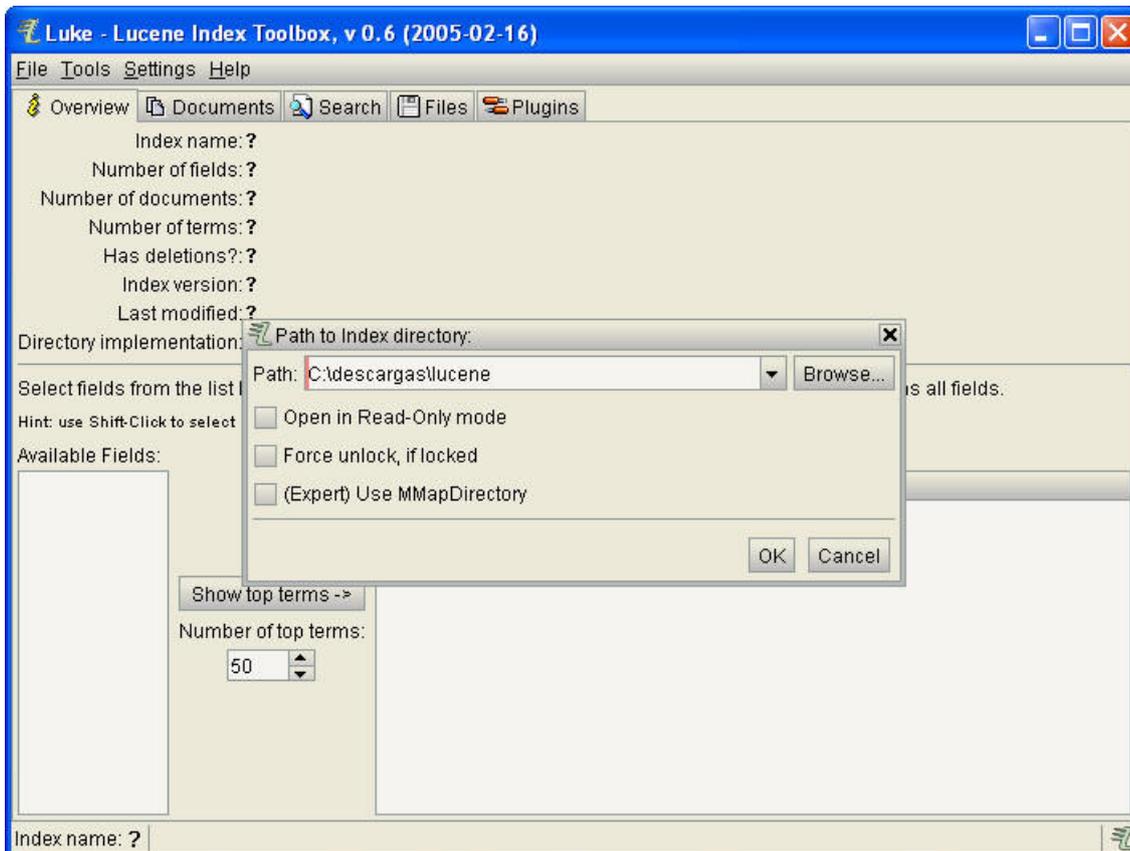


Para arrancarlo (una vez descargado), hacemos doble-click en el jar (si estamos en entorno Windows) o ejecutamos la siguiente línea de parámetros.

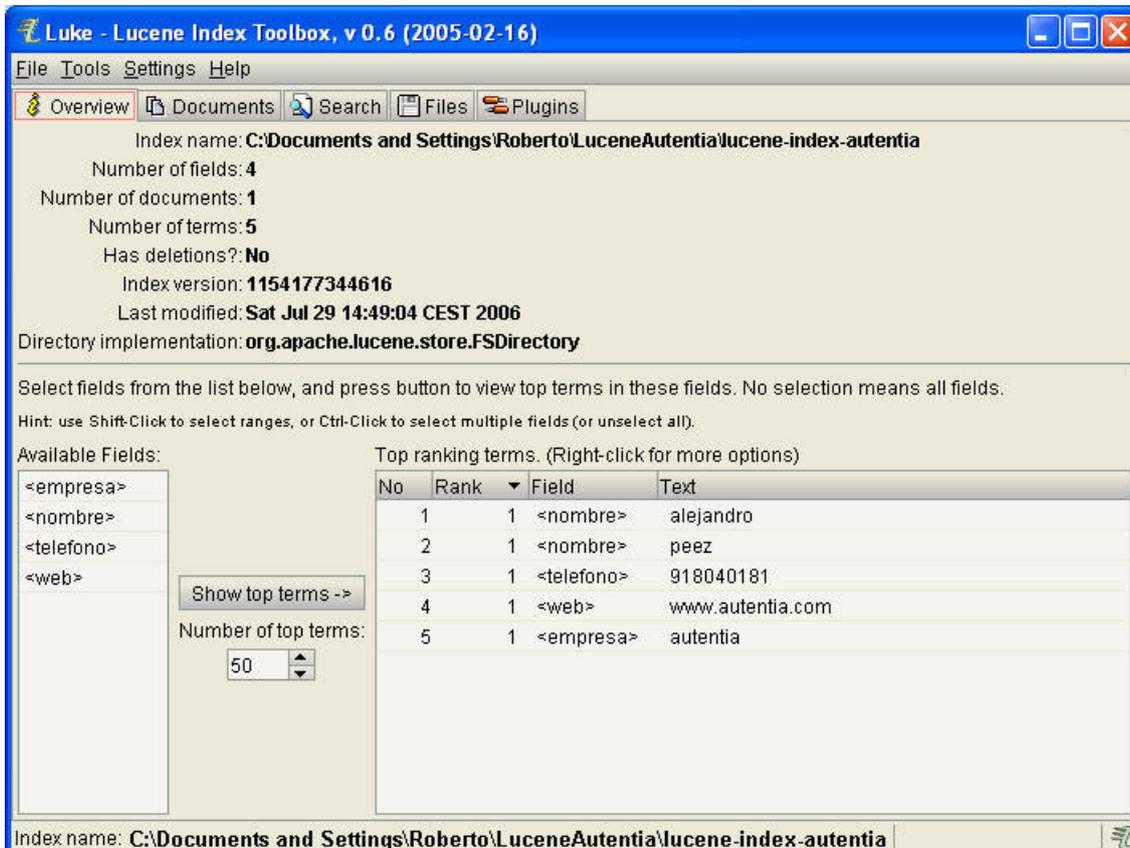
```
java -jar lukeall.jar
```

El aspecto es bastante intuitivo, si estamos acostumbrados a distintos productos de administración de bases de datos.

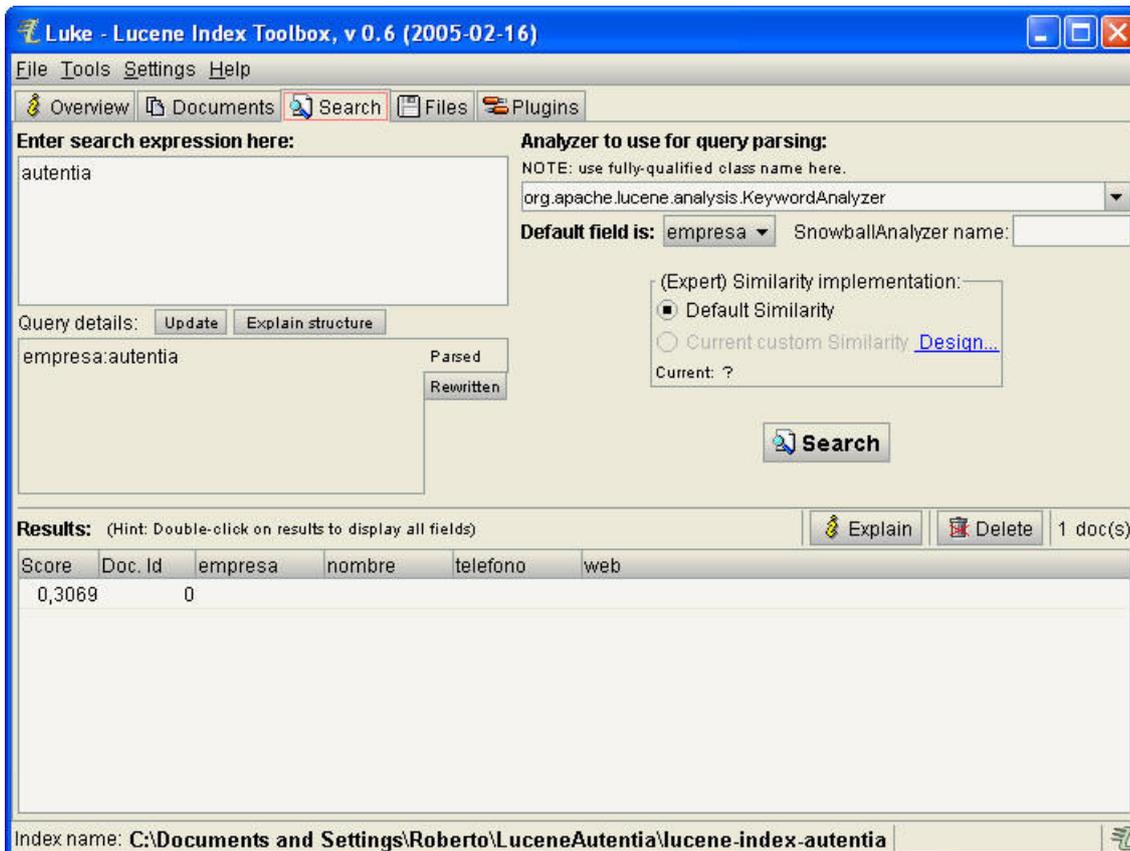
Elegimos el directorio de indexación:



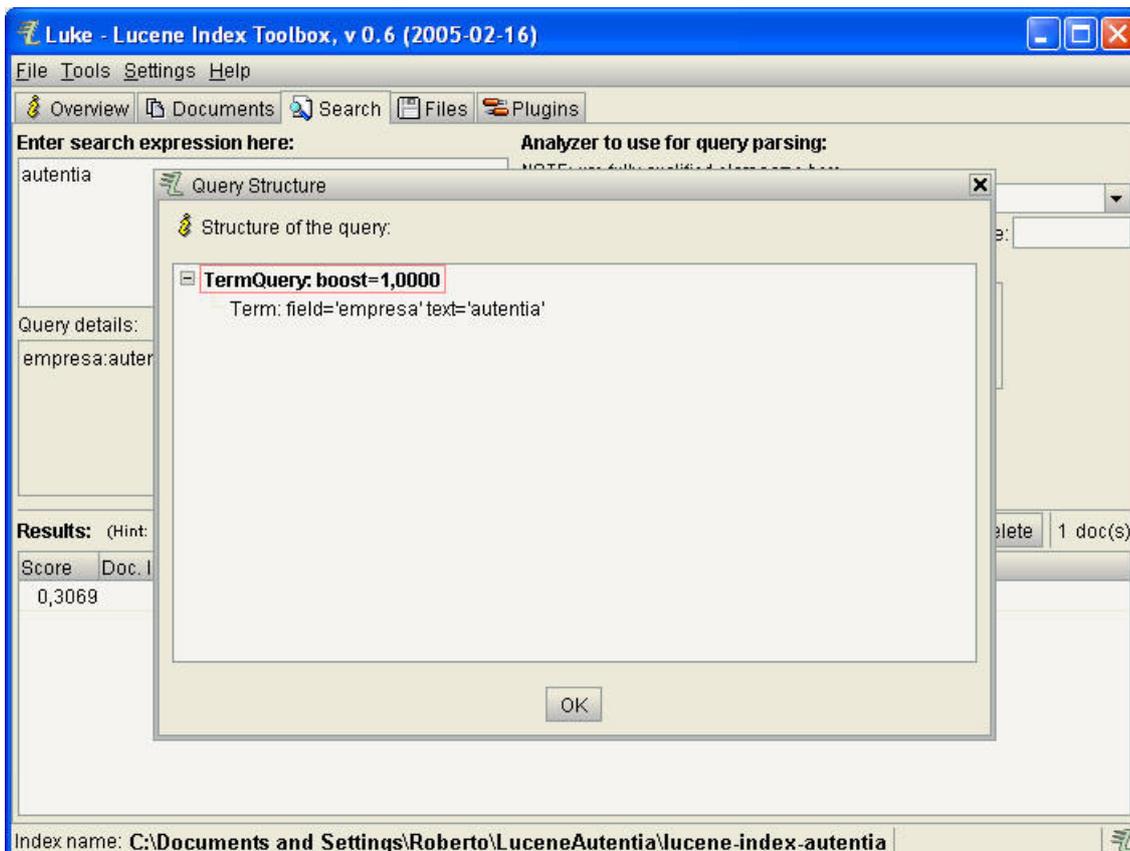
Y comprobamos que los datos se han indexado correctamente. Luke es una herramienta estupenda para entender (a través de prueba y error) los distintos tipos de almacenamiento, indexación y descomposición de campos.



Realizamos una sencilla consulta para ver si recuperamos los elementos deseados (abajo)



E incluso podemos examinar la estructura de la consulta, fundamental para escribir nuestro código Java cliente.



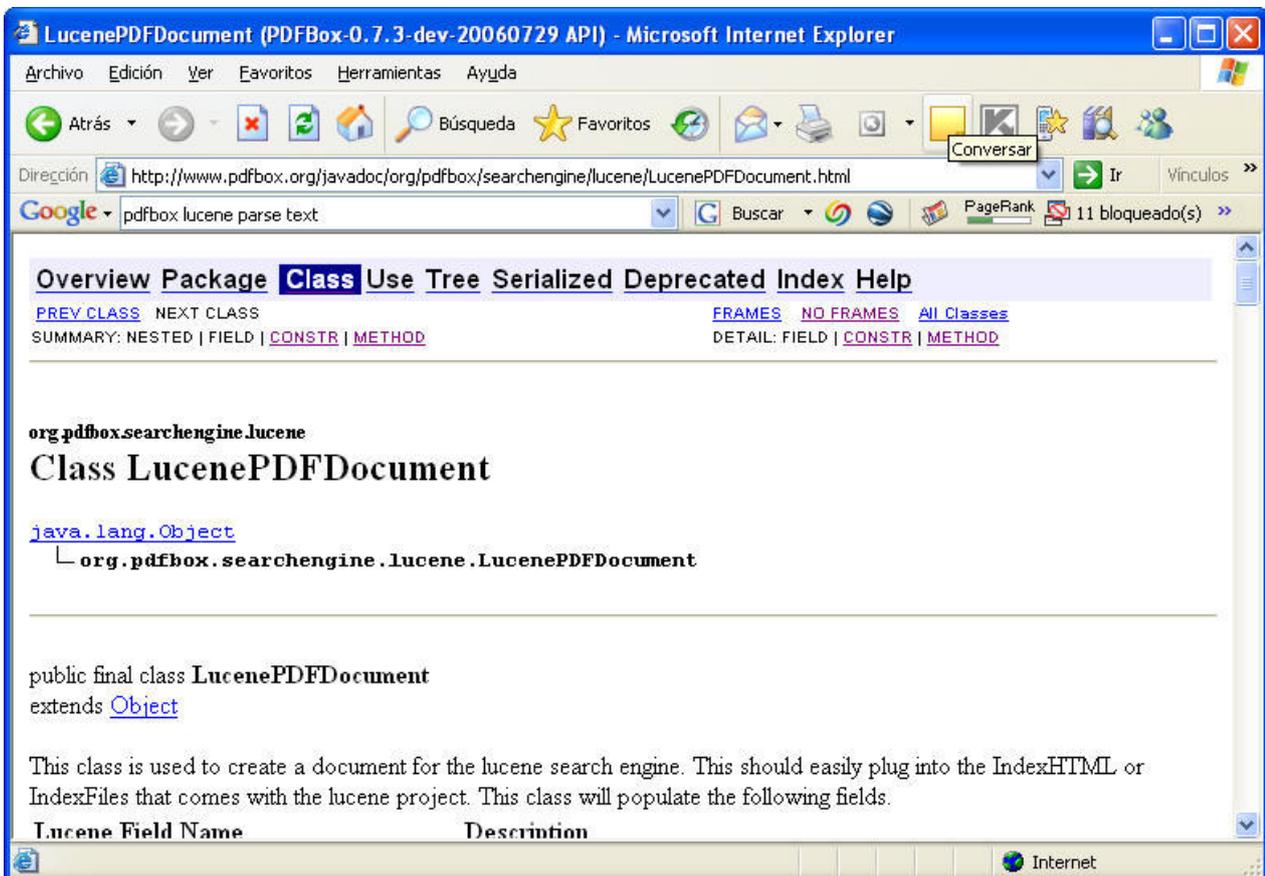
Indexación de un PDF

Ahora, vamos a hacer una pequeña mejora, que consiste en indexar, junto a nuestra información viva, el contenido de un fichero PDF (que no sea una imagen escaneada de texto, claro).

Para ello usaremos una de las APIs más extendidas PDFBOX ([recordar que ya tenemos otros tutoriales que hablan del tema](#))



Veremos que hay unas clases, que ya incorporan la integración con Lucene, es sorprendente lo que se lo curra la gente.



Si intentamos integrar con nuestro programa que usa la versión 2.0 de Lucene.



```

try
{
document = LucenePDFDocument.getDocument(new File("c:\\chumano.pdf"));

//Document document = new Document();
document.add(new Field("nombre", nombre, Store.YES, Index.NO));
document.add(new Field("empresa", empresa, Store.NO, Index.TOKENIZED));
document.add(new Field("web", web, Store.COMPRESS, Index.UN_TOKENIZED));
document.add(new Field("telefono", telefono, Store.YES, Index.UN_TOKENIZED));
}

```

Nos aparece el siguiente error, claramente predecible, ya que si elegimos estar a la última, hay algunos problemas que tenemos que pagar ... Los productos tardan un tiempo en dar el salto de una versión a otra. PDFBox está utilizando internamente una versión anterior del jar de Lucene.

```

Exception in thread "main" java.lang.NoSuchMethodError:
org.apache.lucene.document.Field.UnIndexed(Ljava/lang/String;Ljava/lang/String;)
Lorg/apache/lucene/document/Field;
at org.pdfbox.searchengine.lucene.LucenePDFDocument.getDocument
(LucenePDFDocument.java:167)
at luceneautentia.gestorLucene.crearDocumento(Main.java:82)
at luceneautentia.Main.main(Main.java:39)
Java Result: 1
BUILD SUCCESSFUL (total time: 3 seconds)

```

Los cambios son mínimos (en nuestro caso) y se producen a la hora de dar de alta objetos. En versiones más antiguas de lucene tenemos que usar los siguiente métodos:

```

Keyword: El dato es almacenado e indexado pero no particionado. Esto
interesa cuando el dato es relevante pero no interesa particionarlo, por
ejemplo, una fecha.
Text: El campo es guardado, indexado y particionado. Este campo no debe
ser utilizado con textos muy grandes porque se guardará el original y las
partes.
UnStored: El dato no es almacenado pero si indexado y particionado.
Adecuado para grandes textos.
UnIndexed: El dato es almacenado pero no indexado ni particionado. Este
es el caso de una URL, nos interesa tenerla a mano pero no indexarla o
partirla

```

Nos bajamos un jar anterior 1.4.x y rehacemos un poquito el código

```

/*
 * Roberto Canales Mora
 * www.adictosaltrabajo.com
 */

package luceneautentia;

import java.io.IOException;
import java.util.*;
import org.apache.lucene.document.*;
import org.apache.lucene.document.Field.*;
import org.apache.lucene.analysis.*;
import org.apache.lucene.index.*;
import org.apache.lucene.analysis.standard.*;
import org.apache.lucene.store.Directory;
import org.apache.lucene.store.Lock;
import org.pdfbox.searchengine.lucene.*;
import java.io.*;

/**
 * @author Roberto
 */
public class Main {

public Main() {
}

public static void main(String[] args) {

gestorLucene ejemplo = new gestorLucene();

Document persona1 = ejemplo.crearDocumento("Carlos
Garcia","Autentia","www.autentia.com","655991172");
Document persona2 = ejemplo.crearDocumento
("Roberto","Autentia","www.autentia.com","918040181");

try
{
ejemplo.indexar(persona1);
ejemplo.indexar(persona2);
}
catch (Exception e)
{
System.out.println("Error en la aplicación " + e.toString());
}
}

```

```

e.printStackTrace();
}
}
}
class gestorLucene
{
String directorioIndexacion = "lucene-index-autentia14";

void indexar(Document documento) throws Exception
{
// org.apache.lucene.index.IndexReader.unlock(new );

Analyzer analizador = new StandardAnalyzer();
IndexWriter writer = new IndexWriter(directorioIndexacion, analizador,
true);
writer.setUseCompoundFile(false);
writer.addDocument(documento);
writer.optimize();
writer.close();
}

Document crearDocumento( String nombre, String empresa, String
web,String telefono )
{
Document document = null;

try
{
document = LucenePDFDocument.getDocument(new File
("c:\\humano.pdf"));

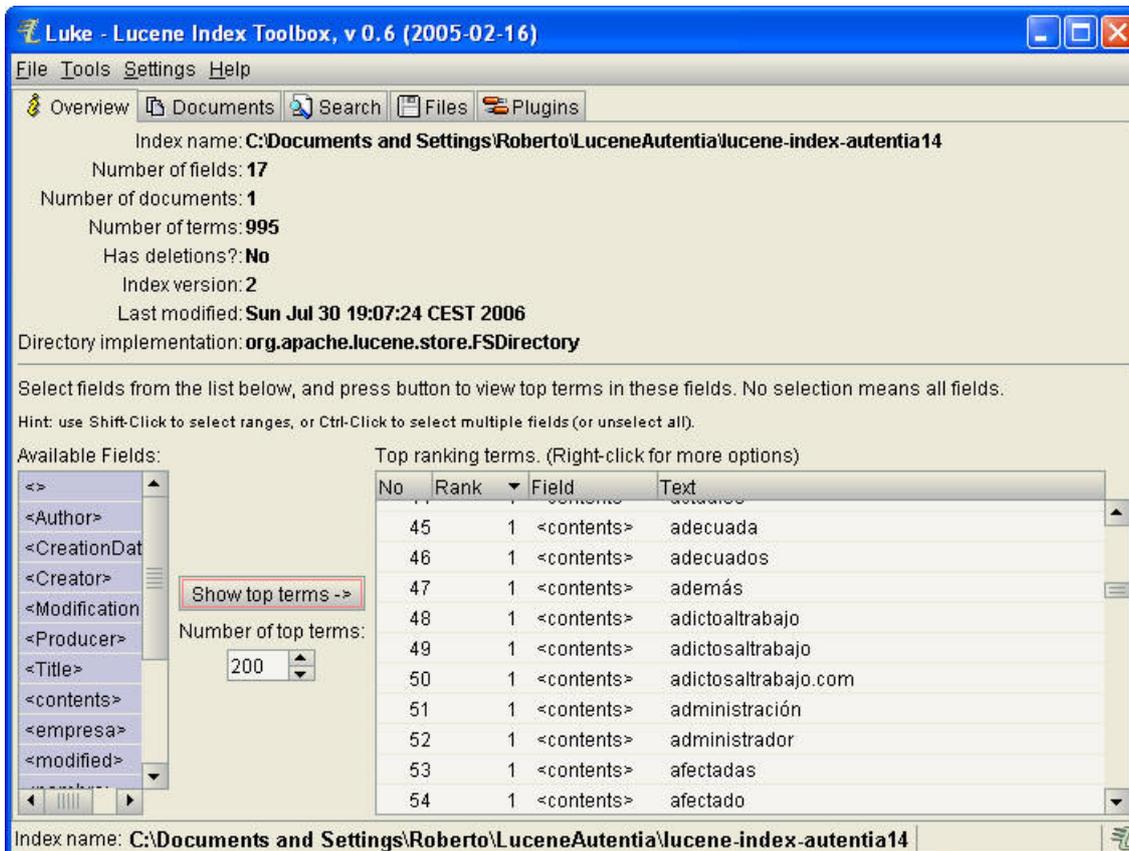
//Document document = new Document();
document.add(Field.Text("nombre",nombre));
document.add(Field.Text("empresa",empresa));
document.add(Field.UnIndexed("web",web));
document.add(Field.Keyword("telefono",telefono));

}
catch (Exception e)
{
System.out.println("Error en tratamiento pdf " + e.toString());
e.printStackTrace();
}

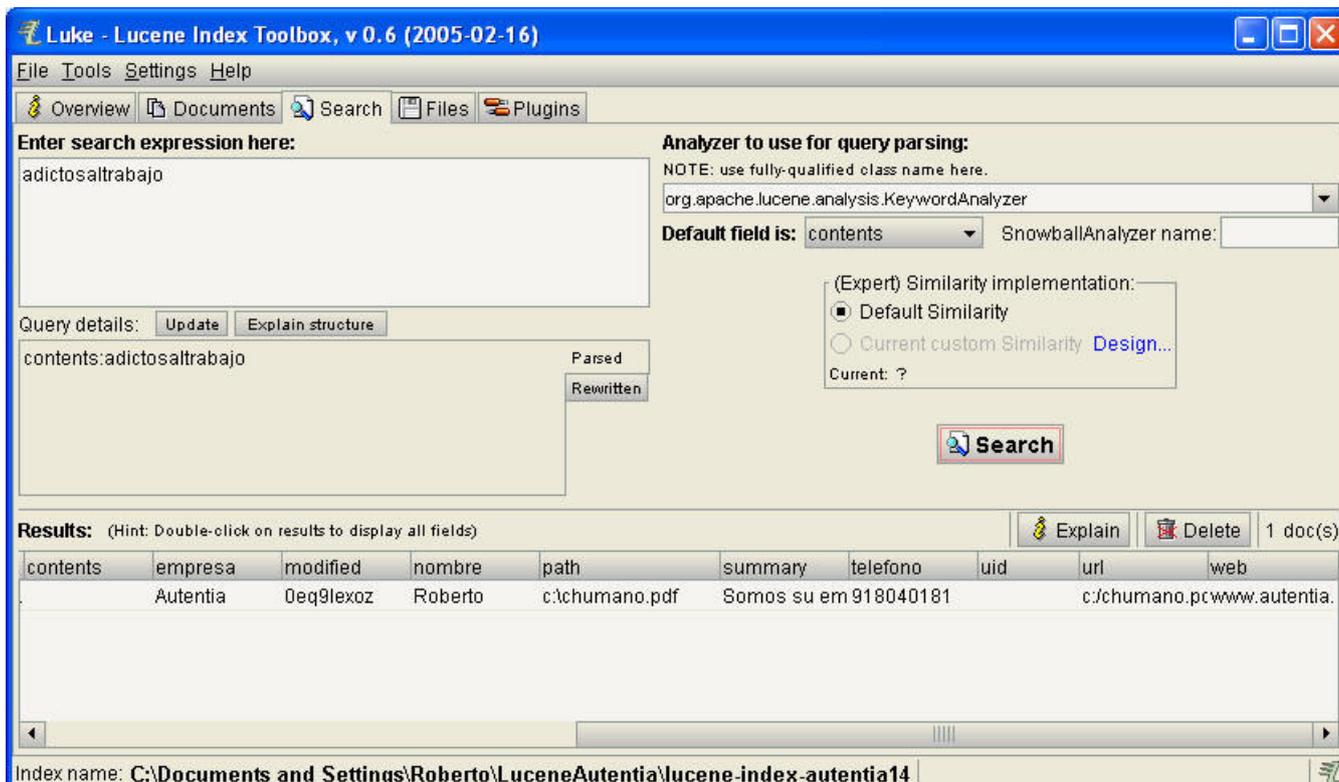
return document;
}
}
}

```

Volvemos a arranca Luke y veremos que, efectivamente el PDF está indexado.



Y podemos recuperar los documentos por los textos que contienen los pdfs, con una consulta simple (la misma que haríamos desde Java)



Deberes avanzados

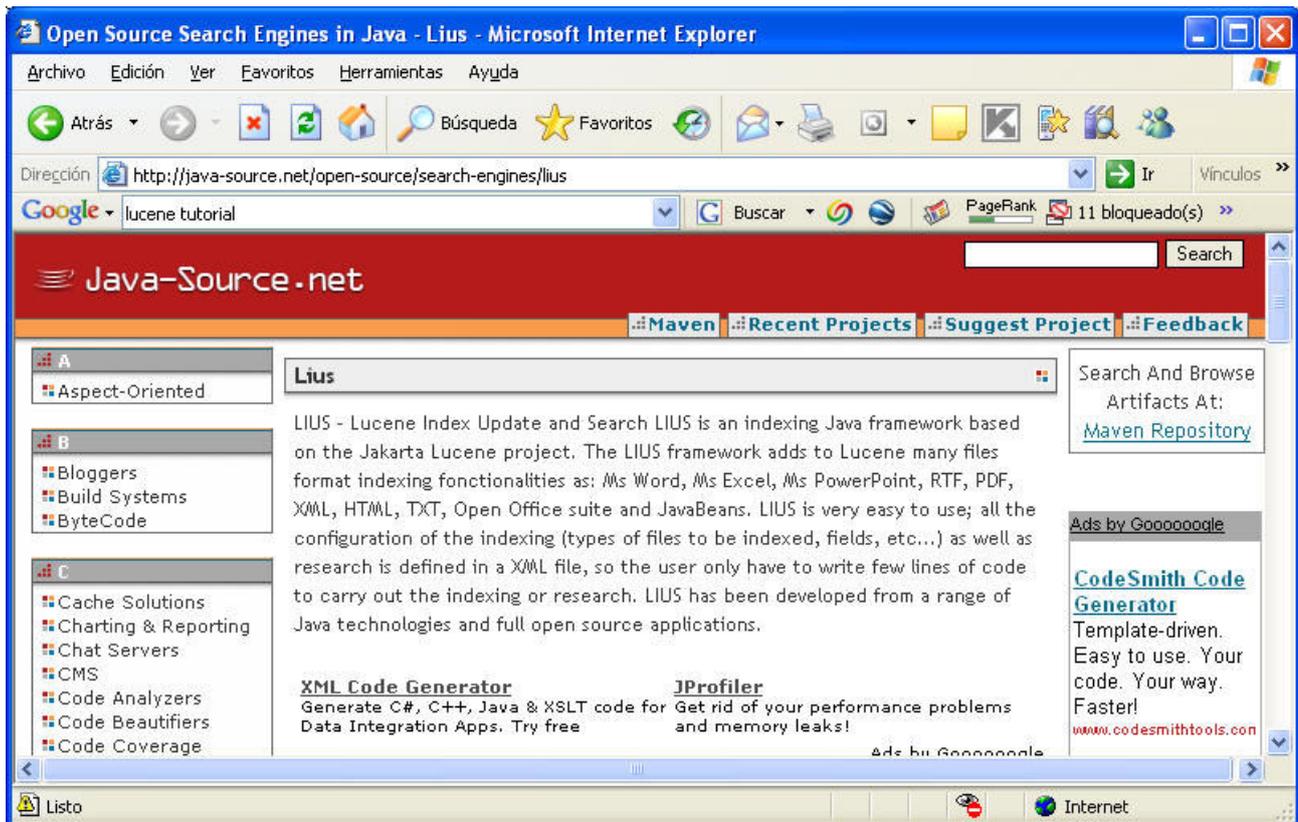
Para realizar una buena indexación, es necesario ser consciente del lenguaje de los documento. En nuestro caso es el Español. Os dejo un par de enlaces a seguir:

<http://www.mail-archive.com/lucene-user@jakarta.apache.org/msg09151.html>

[package spanishfilestemmer; import org ...](#)

Lius

Bueno, es bastante posible que muchas de las cosas que queramos hacer, ya estén inventadas, os recomiendo que os reviséis algunos otros proyectos relacionados con el mundo de los buscadores ...



Conclusiones

Cada día es mayor el número de datos a manejar. Cuanto mayor es la información, más difícil es encontrarla (no confundir información con conocimiento). Integrar un motor de búsqueda en vuestras vidas (desarrollos).

Recordad una cosa: El que conoce como funciona un producto, posee un valor táctico. El que además entiende por qué funcionan los productos, posee un valor estratégico. El personal de www.autentia.com compartimos el conocimiento táctico pensando en que un buen estratega considerará que es más barato llamarnos para temas avanzados que invertir las mismas horas y cometer los mismos errores que nosotros despistando los objetivos de negocio (necesidades de cliente).



[Puedes opinar sobre este tutorial aquí](#)

Recuerda

que el personal de [Autentia](http://www.autentia.com) te regala la mayoría del conocimiento aquí compartido ([Ver todos los tutoriales](#))

¿Nos vas a tener en cuenta cuando necesites consultoría o formación en tu empresa?

¿Vas a ser tan generoso con nosotros como lo tratamos de ser con vosotros?

info@autentia.com

Somos pocos, somos buenos, estamos motivados y nos gusta lo que hacemos

Autentia = Soporte a Desarrollo & Formación

Creatividad Internet

Nuevo servicio de notificaciones

Si deseas que te enviemos un correo electrónico cuando introduzcamos nuevos tutoriales, inserta tu dirección de correo en el siguiente formulario.

Subscribirse a Novedades	
e-mail	
	<input type="button" value="Enviar"/>

Otros Tutoriales Recomendados ([También ver todos](#))

Nombre Corto	Descripción
Introducción a OCR	En este tutorial os mostramos los fundamentos de la tecnología OCR (Optical Character Recognition) y cómo utilizar dos herramientas relacionadas: las librerías de Asprise y GOOCR.
Rotar imágenes TIFF	En este tutorial os mostramos un ejemplo de manipulación de imágenes en formato TIFF utilizando el api de programación JAI
Apache Commons Configuration	En este tutorial os vamos a enseñar a utilizar una API de Apache para gestionar las configuraciones de vuestras aplicaciones de manera avanzada
El comportamiento humano y el trabajo	En el presente tutorial os mostramos algunos principios sobre el comportamiento y actitudes de trabajadores y empresas
PMD, Eclipse y NetBeans	Tutorial que describe la instalación y uso de PMD en los entornos de desarrollo Eclipse y NetBeans
Callisto, nunca antes resultó tan fácil desarrollar con Eclipse	En este tutorial os enseñamos a instalar y utilizar Callipso: una aplicación que permite instalar de manera fácil y cómoda plugins y sus dependencias en Eclipse
JSF en Java Studio Creator 2	En este tutorial os mostramos como realizar una aplicación JSF utilizando la herramienta Java Studio Creator en su segunda versión
Librería PDFBOX de Java	En este tutorial os mostramos como utilizar algunas de las utilidades de línea de comandos que incorpora la librería Java PDFBOX, para manejar documentos en formato pdf
Integrar Google Maps en tu web	En este tutorial os mostramos con un sencillo ejemplo cómo integrar los mapas de localización de Google en tu propia web o portal

Nota: Los tutoriales mostrados en este Web tienen como objetivo la difusión del conocimiento.

Los contenidos y comentarios de los tutoriales son responsabilidad de sus respectivos autores.

En algún caso se puede hacer referencia a marcas o nombres cuya propiedad y derechos es de sus respectivos dueños. Si algún afectado desea que incorporemos alguna reseña específica, no tiene más que solicitarlo.

Si alguien encuentra algún problema con la información publicada en este Web, rogamos que informe al administrador rcanales@adictosaltrabajo.com para su resolución.

[Patrocinados por enredados.com Hosting en Castellano con soporte Java/J2EE](#)

