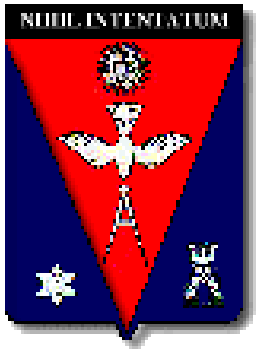



# Proyecto de grado

Esteban David Zamar

Ingeniería en Informática

Facultad de Ingeniería

Universidad Católica de Salta

Año 2016

Título del trabajo: “BUSCADOR SEMÁNTICO”

Profesor Guía: Dra. Alicia Pérez / Lic. Jorge Perdiguero

Tribunal evaluador:

Presidente: Ing. Beatriz Parra de Gallo.

I Miembro: Lic. Carolina Cardoso.

II Miembro: Lic. Patricio Arredes.

Fecha de presentación: ___/___/___

# Agradecimientos

En primer lugar a la Dra. Alicia Pérez que sin su criterio, apoyo, comprensión y participación no podría haber finalizado este trabajo. Además también a la Lic. Carolina Cardozo que me apoyó en algunas actividades de este trabajo

A mis padres y familiares por su confianza, inversión y compañía en mi carrera.

A mis amigos, compañeros y profesores de carrera de los que me llevo gratos recuerdos.

A mi excompañero el Ing. Sebastián García por asesorarme en la implementación del prototipo.

A la empresa JARNet Ingeniería en la cual trabajo, por brindarme su apoyo.

Al Lic. Jorge Perdiguero por su acompañamiento al final de este proyecto y por brindarme toda la ayuda complementaria para finalizar este trabajo dándole un enfoque de proyecto de aplicación real.

# Índice

# Índice de figuras



# Índice de tablas

## Resumen

Este trabajo consiste en la construcción de un Prototipo de Buscador Semántico para Resoluciones Rectorales de la Universidad Católica de Salta. Conforma una parte de un proyecto de investigación sobre Minería de Textos a cargo de la Dra. Alicia Pérez y la Licenciada Carolina Cardoso. Entonces, el trabajo en cuestión intenta asociar la idea de que a partir de la minería de textos se puede desarrollar un buscador semántico con herramientas de software libre cumpliendo características de usabilidad que en primera instancia integre el proyecto de investigación ya nombrado y en segunda instancia muestre la apertura de nuevos caminos que deben encausar las ciencias informáticas generando soluciones cada vez más eficientes, eficaces y satisfactorias para los usuarios.



Palabras claves: Buscador Semántico, Usabilidad, Minería de textos

# Capítulo 1 - Introducción

Este capítulo explicará el problema que se desea resolver con la realización de este trabajo detallando principalmente su contexto, importancia y la motivación para abordarlo.

## 1.1 Introducción al problema: búsqueda de información semántica

En el ámbito institucional y organizacional se maneja un caudal de información que incrementa con el paso del tiempo debido al crecimiento de la cantidad de gestiones realizadas y registradas en los sistemas utilizados en nuestro entorno. La mayoría de esos registros son o se sustentan con documentos de textos digitales y tienen funciones de información específicas.

Una de las funciones más importantes que tienen estos archivos es las de responder a requerimientos de búsqueda de información que demandan los usuarios en el desempeño de sus actividades. Para ello existen los buscadores de archivos de texto. Estos buscadores normalmente son aplicaciones embebidas o integradas en otras aplicaciones y sistemas operativos, y sus funciones están limitadas a procesos de búsqueda de palabras claves. Es decir que son buscadores que funcionan solo en base a la coincidencia de palabras entre las ingresadas por un usuario como un parámetro de búsqueda y las contenidas en los archivos de texto. Estos buscadores presentan ciertas limitaciones que pueden ser percibidas en los siguientes ejemplos [Hampp & Lang, 2005]:

- Se debe llamar a un cliente. ¿Cómo puedo encontrar el e-mail con el número de teléfono de ese cliente? Tal vez nunca se utilizó una cadena "Número de teléfono" en el contenido de un correo, y se utilizó algo como "Puede contactarme al teléfono 0387-4223344"

- Se debe crear un informe de calidad. ¿Cómo encuentro los reportes de reparación de automóviles que informan sobre problemas de frenos, en el norte de la ciudad de Salta?. Las descripciones por los talleres del norte de la provincia especifican cosas como "Pastillas de freno ajustadas debido a fugas en el sistema hidráulico". Y además, sólo contienen la dirección de los talleres de reparación.

- Quiero investigar acerca de un nuevo fármaco. ¿Cómo puedo encontrar documentos que hablen de una determinada proteína y del o los tipos de enfermedades curables con ese fármaco en el mismo párrafo?. Hay veinte nombres diferentes de la proteína en los prospectos. Además, en los documentos no puede existir el término "enfermedad", ya que sólo se nombra la enfermedad que cura dicho fármaco.

Estos ejemplos muestran principalmente que las organizaciones manejan datos desestructurados con un alto valor de información y no los aprovechan. Por otro lado, tampoco saben cómo estructurar esos datos para implementar funciones y características de búsqueda que

beneficien concretamente necesidades de información específicas de los usuarios, tales como las presentadas en los ejemplos.

## 1.2 Descripción del problema

En este contexto el problema radica principalmente en la construcción de un Buscador para el trabajo previo realizado en el proyecto de Minería de Textos para la categorización de Documentos a cargo de la Doctora Alicia Pérez y la Lic. Carolina Cardoso.

La arquitectura del Sistema construido en el proyecto nombrado se muestra en la Figura 1 donde se presentan y definen los componentes y procesos más importantes del trabajo realizado [Pérez y Cardoso, 2011]:

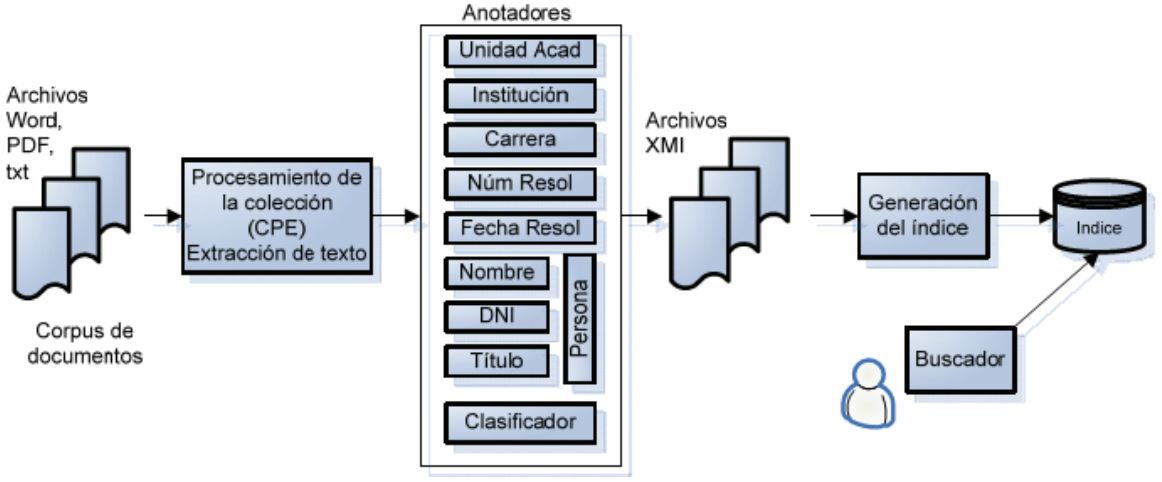



Figura 1: Arquitectura del sistema de minería de textos para la categorización automática de documentos

A continuación se explicarán los procesos indicados en la Figura 1 para comprender específicamente el contexto del problema.

En primera instancia se utilizaron librerías de extracción de texto para obtener textos de documentos con distintos formatos (Word,PDF,txt) de resoluciones rectorales de la Universidad Católica de Salta con el objetivo de someterlos a un proceso de análisis para obtener finalmente documentos XMI (*XML Metadata Interchange*), que son documentos con estructura y contenido lo que permite realizar búsquedas puntuales en base a parámetros o anotadores predeterminados sobre estos documentos.

En este dominio se procesaron alrededor de 8 mil resoluciones rectorales. Para la extracción de textos se utilizaron las librerías "POI" (poi.apache.org) y tm-extractors (www.textmining.org). En esta etapa se remplazaron los caracteres especiales por el carácter ASCII correspondiente y además se separó el cuerpo del encabezado en cada archivo.

La fase de análisis incluye tokenización y detección de entidades en documentos individuales tales como personas, fechas, organizaciones, unidades académicas y datos sobre la resolución (fecha y número). Además con la ayuda de un clasificador aprendido automáticamente del corpus de resoluciones se anota cada documento con una categoría. Existen 21 categorías que fueron obtenidas del personal especializado en la elaboración de resoluciones. Algunos ejemplos son: designación de planta docente, convenio de pasantías, convenio de colaboración, llamado a concurso docente, o designación de tribunal de concurso. [Pérez y Cardoso, 2011]

La fase de análisis generó como resultado un conjunto de archivos XMI. Estos archivos, a su vez, fueron convertidos a una nueva estructura tipo "Document" compuesta por "Fields" o "campos" donde cada campo representa un anotador. Finalmente se indexaron los archivos con estructura "Document" y fueron almacenarlos en un archivo de índices.

El proceso final de la arquitectura consiste en la creación de un prototipo de buscador que realice búsquedas sobre los archivos de índices y muestre los resultados en una interfaz comprensible y amigable para el usuario, lo cual es el desafío de este proyecto.

## 1.3 Importancia del problema

Para comprender la importancia del problema se presentarán nuevamente algunos ejemplos concretos en los que los usuarios de una organización o institución realizan búsquedas en los sistemas de información o en los sistemas operativos de sus PC' s donde los datos no se encuentran estructurados ni ordenados y los resultados obtenidos no están relacionados con lo que están buscando. Además se analizarán las consecuencias o pérdidas de oportunidades que generan estas limitaciones para ponderar la incidencia cuando acontezcan.

Ejemplo 1:"El usuario Daniel Corbalán, encargado de Ventas de una empresa de seguridad biométrica, requiere consultar la cantidad de lectores dactilares de un determinado modelo que fueron cotizadas a un cliente en particular en un periodo de tiempo para evaluar luego la fidelidad de ese cliente respecto a las compras realizadas del mismo lector. Las cotizaciones son realizadas en una hoja de cálculo (Excel) y exportadas a PDF. El usuario procede a ejecutar el buscador del Sistema Operativo de su PC seleccionando la carpeta donde desea buscar (ordenadas por clientes) y el nombre del modelo del lector de huellas. Al ejecutar la búsqueda se generan resultados de archivos *.xls y *.pdf con fechas de modificación distintas. El nombre del modelo puede ser cargado como "OA-99" o "u-bio". Eventualmente, dependiendo del usuario que realizó la cotización los guiones medios se omiten y selecciona el nombre del modelo que prefiera usar. La variedad de resultados generados no permite que el usuario pueda identificar con precisión las

cotizaciones realizadas de un tipo de lector dactilar a un cliente determinado, por ende, no encuentra lo que está buscando y no es posible realizar una evaluación certera de la fidelidad del cliente.

Ejemplo 2: Una empresa productora de cortometrajes tiene la innovadora idea de automatizar el proceso de identificación de personajes, actores, equipo técnico y vestuario a partir de una memoria técnica y un guion presentados al finalizar la planificación de cada cortometraje. Esto para tener a modo de resumen los gastos de equipamiento, viajes, alquileres y personal antes de emprender cada producción. El guion y la memoria técnica son realizadas en archivos *.doc. ¿Cómo se podría encontrar cada uno de estos conceptos en esos documentos si no se encuentran estructurados por anotadores? Es imposible.

Todo esto clarifica la idea de que los buscadores tradicionales de archivos presentan una limitación muy grande donde las organizaciones, para superarla, implementan sistemas de bases de datos relacionales que representan además de costos para su creación, redundancia de datos e incrementan los riesgos que podrían afectar la consistencia e integridad de esas bases de datos ya que habría que transcribir los datos de interés desde los archivos de texto a los registros del sistema de Base de Datos relacional.

Se llega a la conclusión de que la "Búsqueda de palabras claves" o "Tradicional" no permite buscar información en base a conceptos de alto nivel y tampoco puede interpretar relaciones entre las palabras a buscar. Debido a esta limitación y necesitad surge la denominada "Búsqueda Semántica" o "Búsqueda Inteligente" cuyo funcionamiento se basa en la estructuración o descripción semántica de los archivos de texto como donde existe un trabajo previo a la realización de consultas de información. Esa "estructuración" es desarrollada por la "Minería de Texto" la cual se encarga de definir relaciones y conceptos de un dominio expresado en forma textual, procesando y estructurando su contenido para identificar, acceder y mostrar información del dominio correspondiente mediante herramientas y aplicaciones que lo permitan, como por ejemplo un Buscador.

Un buscador semántico, entonces, tiene como objetivo comprender conceptos y relaciones que son ingresados como parámetros y mostrar resultados acordes a lo que el usuario desea encontrar. Dicho objetivo es la principal limitación de los buscadores "tradicionales" o sintácticos". Sin embargo los buscadores tradicionales evolucionaron durante los últimos años y hoy presentan numerosas ventajas y características en cuanto a la Usabilidad, y deberían ser adoptadas para la construcción de los Buscadores Semánticos proyectando así un enfoque evolutivo hacia la creación de sistemas que implementen búsquedas sobre bases de conocimiento generadas por ontologías, minería de texto, minería de datos, etc., que respondan de la mejor manera a los nuevos requerimientos de información existentes en nuestra sociedad.

## 1.4 Motivación para abordarlo

Los motivos que justifican abordar este problema están centrados en la innovación de las Tecnologías de Información utilizadas e implementadas actualmente en Sistemas de Búsquedas de nuestro entorno. Se mostrarán entonces las ventajas el uso de los buscadores semánticos mediante la superación a los límites que hoy imponen los buscadores tradicionales.

Por otro lado el oficio de un Profesional Informático se centra en resolver problemas y limitaciones de información que deben abordarse utilizando la Ciencia y las Tecnologías de Información. De esta manera se definen retos y actividades que den respuestas y soluciones a esos problemas y limitaciones de información de un dominio determinado. En este proyecto los problemas y retos que motivan abordar el tema de la Búsqueda Semántica son los siguientes:

- Las limitaciones de los buscadores tradicionales para comprender conceptos de alto nivel y relaciones sobre un dominio en particular.
- La reutilización del alto grado de Usabilidad que poseen los buscadores tradicionales actualmente centrando la atención en desarrollar un buscador eficaz, eficiente y que satisfaga los requerimientos de información de los usuarios de manera óptima.
- Demostrar la utilidad que podrían tener los Buscadores Semánticos en los Sistemas Informáticos de nuestro entorno mediante la utilización de la Minería de Textos para estructurar información digital que actualmente no se encuentra estructurada.
- Incentivar a los desarrolladores a construir sistemas fáciles de usar teniendo en cuenta que un alto grado de sencillez en el diseño de una Interfaz de usuario conlleva numerosos beneficios en cuanto a la competencia y rentabilidad de sus trabajos.

## 1.5 Pasos a realizar

Este trabajo estará estructurado de la siguiente manera:

Capítulo 2: “Estado de la cuestión”, en el cual se reflejará la situación actual de los buscadores semánticos respecto a la utilidad y la evolución que han tenido en los últimos años. Se definirán las tendencias y los trabajos de investigación que justifican y promueven la utilización de los buscadores semánticos con la usabilidad en su construcción.

Capítulo 3: “Definición del problema”, se definirá el problema en un contexto específico detallando los puntos y pautas que servirán de guía para encarar la solución del mismo.

Capítulo 4: "Solución Propuesta", básicamente se documentará todo el desarrollo de la construcción de un buscador que solucione el problema en concreto, justificando cada proceso y actividad desempeñada.

Capítulo 5: "Resultados", donde se detallarán y analizarán los valores resultantes de los procesos documentados en el capítulo anterior resaltando y justificando las elecciones antes realizadas para solucionar el problema.

## 1.6 Criterios de éxito

Los criterios que determinan la utilidad de la solución propuesta al problema en cuestión son los siguientes:

1. Cumplimiento de los objetivos preestablecidos en el desarrollo de este proyecto.
2. Seguimiento adecuado y documentado de las metodologías a utilizarse para la realización del trabajo
3. Construcción de prototipo de un buscador semántico que sea fácil de usar utilizando herramientas de software libre, centrando la atención en la recuperación y visualización de los resultados.

# Capítulo 2 - Estado de la cuestión

En la fase inicial de este proyecto se realizaron investigaciones teóricas y prácticas sobre la "Búsqueda Semántica" para entender sus conceptos e implementar ejemplos que permitan mostrar lo aprendido y conocer las herramientas que se utilizan para el desarrollo de este tipo de buscadores. El capítulo en cuestión describirá las fases que conformaron las actividades de investigación previa a la realización del prototipo del buscador semántico.

Actividades realizadas:

- Relevamiento y recolección de información
- Investigación de herramientas de búsqueda y procesamiento de texto
- Investigación de herramientas para desarrollo web.
- Presentación del tema y anteproyecto
- Investigación de aplicaciones y ejemplos prácticos
- Implementación de ejemplo

## 2.1 Relevamiento y recolección de información

Esta actividad fue la que encabezó el proyecto. Consistió en entender el contexto del proyecto, mediante una conversación o entrevista con la Dra. Alicia Pérez.

El contexto es el ámbito académico. Consiste en implementar un prototipo de un buscador semántico para la búsqueda de resoluciones rectorales de la Universidad Católica de Salta mediante el uso de herramientas diseñadas para éste propósito.

En primera instancia las herramientas fueron: "*Semantic Search*" de IBM y "*Lucene*" de Apache que serán descriptas al final de esta sección.

Se decidió desarrollar el trabajo con *Lucene* ya que presenta un funcionamiento flexible frente a otras herramientas de búsqueda semántica y utiliza Java, lo cual contribuye a facilitar el uso y la integración de las librerías y herramientas libres de extracción de texto. Además Semantic Search dejó de estar disponible públicamente al pasar a formar parte de la plataforma comercial de búsqueda semántica de IBM OmniSearch.

Luego de estudiar algunos artículos, tutoriales y otros documentos se asimiló el auge y la importancia que tiene la implementación de este tipo de búsqueda en los archivos de texto, bases de datos, páginas web y otras estructuras de datos dentro de una organización.

Luego de haber investigado lo suficiente como para entender y definir las ventajas que brinda el uso de este tipo de búsqueda se procedió a determinar de manera informal y limitada un

análisis abstracto del prototipo definiendo los tipos de archivos con los que se trabajará y cuáles serán los medios de acceso o búsqueda sobre los mismos.

Se trata de archivos anotados como entrada y accedidos de alguna manera vía web. Dichas anotaciones corresponden a la identificación de los metadatos los cuales ya fueron definidos en el trabajo de Minería de Texto a cargo de la Dra. Alicia Pérez.

Ambos aspectos serán tenidos en cuenta en la actividad del Análisis y Diseño del prototipo.

### 2.1.1 Semantic Search

"Semantic Search" es una herramienta creada por IBM que permite realizar búsquedas semánticas. Específicamente añade a UIMA un motor de búsqueda semántica. Incluye un CAS Consumer que llena un índice con el contenido del documento así como las anotaciones generadas añadidas por los anotadores implementados en UIMA. [Perez & Cardoso, 2011]

La herramienta dentro del proyecto fue estudiada por la Dra. Alicia Perez y la Lic. Carolina Cardoso quienes desarrollaron un ejemplo funcional sobre los anotadores definidos para el contexto de este mismo problema.

### 2.1.2 Lucene

Lucene es una herramienta de software libre escrita en Java desarrollada como un proyecto Apache que funciona como un potente motor de búsqueda. Esta herramienta se integra fácilmente en cualquier tipo de proyecto en el que se pretenda implementar distintas variantes de búsquedas sobre archivos digitales. El funcionamiento consiste básicamente en la creación de índices a partir de cadenas de texto ingresadas o extraídas de algún documento (Proceso de Análisis e Indexación) y en la búsqueda sobre esos índices (Proceso de Búsqueda). En la figura 2 se muestran los componentes típicos de una aplicación de búsqueda, donde los componentes con fondo gris pertenecen a Lucene.

No es el objeto de este trabajo brindar un tutorial del uso de la herramienta Lucene para el lector, por ello es que solo se explicarán las funciones principales de las librerías para comprender el funcionamiento de la herramienta y los tipos de datos con los que trabaja.

En principio Lucene adopta un tipo de dato denominado "Document". Este tipo de dato está estructurado por "Fields" o campos que deberían representar atributos del documento, como por ej: nombre, fecha de creación, autor, contenido, etc. También podrían representar algún otro atributo creado por el desarrollador que considere identificar en el documento.

En la figura 3 se muestra el proceso de Análisis e Indexación. El Análisis consiste principalmente en el parseo, preparación y filtro de la cadena de caracteres con la que se deberá trabajar para luego indexarla o no. Esta cadena normalmente es extraída de un archivo de texto que puede estar en distintos formatos. La extracción de estos textos no es realizada por un componente de Lucene, en otra sección se detallarán algunas de estas herramientas. El proceso de Indexación consiste en registrar una cadena filtrada o analizada en un "Archivo de índices" (representado como Indices en la Figura 2 y como Archivo en la Figura 3). Una vez que un documento se encuentra indexado en un archivo de índices se pueden realizar búsquedas sobre los "Fields" o campos que se indexaron de cada documento. Los resultados generados se pueden ordenar, contar y ponderar (entre las funciones principales de búsqueda) para mostrar los resultados de una manera mucho más completa. Es importarte decir que los valores de los campos de un documento pueden ser almacenados o no. La diferencia radica principalmente en que los valores almacenados pueden luego ser obtenidos para ser mostrados. Es recomendable almacenar valores que correspondan a palabras aisladas o textos cortos (el nombre de un archivo por ejemplo). En cambio si se trata de un valor que corresponda a un texto largo no es recomendable que se lo almacene ya que ocuparía demasiado espacio de almacenamiento en disco cuando los índices se generen (tal es el caso del cuerpo de una página web por ejemplo).

A continuación se muestra un ejemplo para realizar el indexado de un conjunto de archivos *.txt y luego otro ejemplo para realizar una búsqueda sobre el archivo de índices generado, a los fines de comprender el uso y los conceptos de Lucene.

## Indexación

```
public class Indexar  {
    // Se definen las variables locales a utilizar
    IndexWriter  writer  = null;
    File        dir     = new File("c:/lucene_dir_to_index");
    Document    doc     = null;

    try {
       // Se crea un indice del tipo StandardAnalyzer almacenandolo en un directorio del HDD
       writer = new IndexWriter(new File("c:/lucene_output_dir_index"), new StandardAnalyzer(), true);

       // Recorremos todos los archivos del directorio (No sus subdirectorios)
       File[] files = dir.listFiles();
       for (File file : files){
          if (file.isFile() && file.canRead() && file.getName().endsWith(".txt")) {
             System.out.println("Indexando el archivo: " + file.getAbsolutePath());

             // Se crea e inicializa el Document con los datos del archivo que queremos guardar
             doc = new Document();
             doc.add(new Field("contents", new FileReader(file)));
             doc.add(new Field("path",    file.getPath(), Field.Store.YES, Field.Index.UN_TOKENIZED));
             // Se añade el documento al indice.
             writer.addDocument(doc);
          }
       }
       // Se organizan y guardan los datos.
       writer.optimize();
       writer.close();
       System.out.println("OK");
    } catch (Exception ex){
      System.out.println(ex);
    }
   }
}
```

## Búsqueda

```
public class Buscar {

  //Vble donde se almacenarán los resultados
  private Hits resultados;

  //Gets y Sets
  public Hits getResultados() {
    return resultados;
  }

  public void setResultados(Hits resultados) {
    this.resultados = resultados;
  }
  //Método para buscar sobre un archivo de indice a partir de un query de búsqueda,
  public void BuscarQuery(String queryBusqueda, String analizador){

      try {
          // Ubicación del índice
          FSDirectory    directory = FSDirectory.getDirectory(new File("D:/Esritorio/Proyecto de grado/Index Lucene"));

          // Creación de un IndexSearcher  a partir de los archivos de índices dentro del directory
          IndexSearcher   searcher  = new IndexSearcher(directory);

          // String de búsqueda"
          String        textToSearch = queryBusqueda;

            // Se crea la consulta y se realiza la búsqueda. Los resultados son almacenados en la vble. "resultados"
           QueryParser    parser   = new QueryParser("content" ,new StandardAnalyzer());
           Query         query    = parser.parse(textToSearch);
           Hits          hits     = searcher.search(query);
           this.resultados=hits;
           }

       } catch (Exception ex){
          System.out.println(ex);
       }
    }
}
```

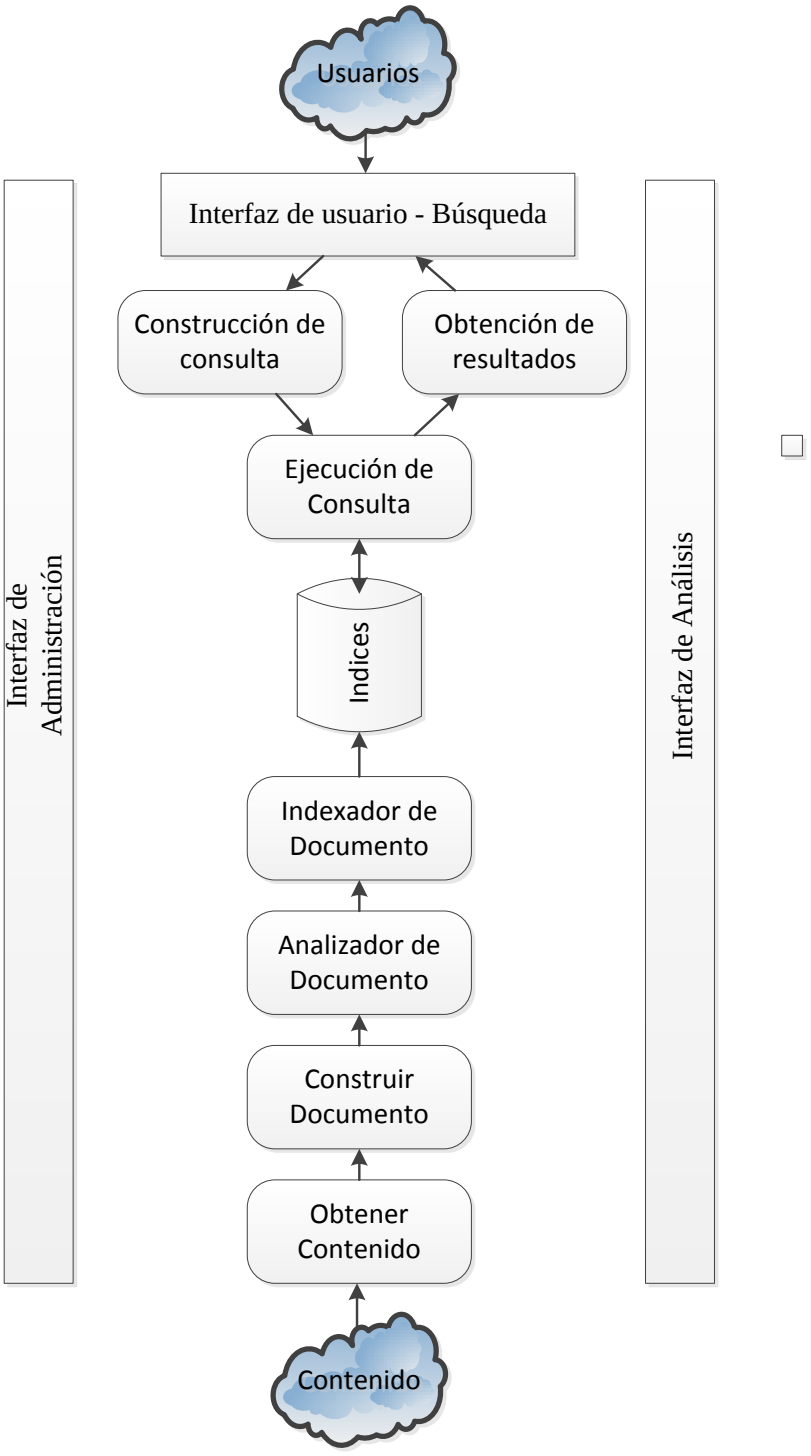



Figura 2: Componentes de una aplicación de Búsqueda con Lucene [Hatcher, Gospodnetic & McCandless, 2009]

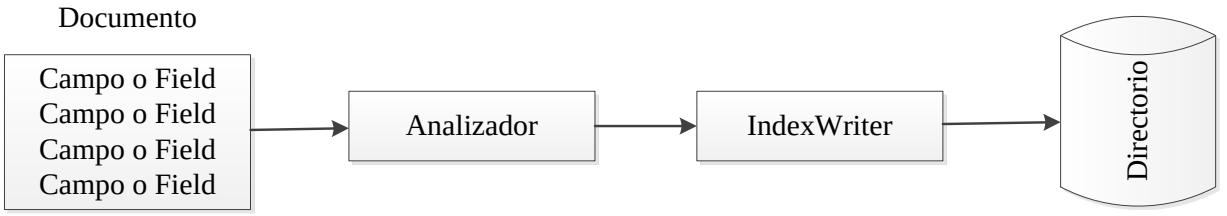



Figura 3: Proceso de Análisis e Indexación de Lucene [Hatcher, Gospodnetic & McCandless, 2009]

## 2.2 Investigación de herramientas de búsqueda y procesamiento de texto

Luego de entender a grandes rasgos el funcionamiento de este tipo de buscadores se interiorizó la herramienta Lucene y otras herramientas de procesamiento de texto. Todas son librerías de clases escritas en *Java* por lo que se utilizó el IDE (*Integrated Development Environment* - Entorno integrado de desarrollo) Eclipse con el fin de estudiar sus características y funciones.

Las librerías, aplicaciones y componentes estudiados fueron los siguientes:

- Framework para realizar Búsqueda: Apache Lucene: https://lucene.apache.org
- Librería de clases para extracción de texto en formato .doc, .docx y.xls: TextMining y POI. (TextMining hace uso de POI). https://poi.apache.org
- Librería de clases para extracción de texto en formato pdf: PDFBox**.** https://pdfbox.apache.org
- Framework para el desarrollo WEB**:** Java Server Faces (JSF)
- Librería complementaria JSF**:** PrimeFaces. http://primefaces.org
- Servidor de aplicaciones**:** Apache Tomcat. https://tomcat.apache.org
- Librerías para el manejo de archivos XML y XMI.

Cabe aclarar que todas las herramientas y librerías utilizadas son de licencia libre cuya documentación se encuentra en las páginas web de sus creadores.

## 2.3 Investigación de herramientas para el desarrollo web.

En base a lo descrito anteriormente ya es posible realizar búsquedas sobre distintos documentos y se verifica efectivamente que Lucene funciona tanto para indexar como para buscar contenido. Se deben integrar las funciones de Lucene en un proyecto que sea orientado a web. Para esto se investigó acerca de las herramientas que ofrece Eclipse y de algunas otras herramientas de licencia libre que sirvan para el desarrollo web.

Al principio se utilizó lo que proporciona Eclipse: JSP (Java Server Pages), o en español Páginas de Servidor Java. Es una tecnología orientada a crear páginas Web con programación en Java. [Alvarez, 2002].

Luego se investigó sobre JSF (Java Server Faces), lo cual es un framework de desarrollo basado en el patrón MVC (Modelo Vista Controlador). [Pérez García, 2006].

A partir de JSF se utilizaron dos tipos de Frameworks: IceFaces y PrimeFaces, que son tal como JSF, pero permiten al programador incluir una serie de tags Ajax en sus JSP o xhtml de tal manera que el Ajax es generado por el framework correspondiente de manera automática y además ofrecían un entorno mucho más amigable para el desarrollo de aplicaciones web.

## 2.4 Presentación del tema y anteproyecto

En esta actividad se definieron y presentaron el tema y el anteproyecto correspondientes al proyecto de grado. [Zamar, 2013].

## 2.5 Investigación de aplicaciones y ejemplos prácticos e implementación de ejemplo

Esta actividad consistió en investigar acerca de la utilización de librerías para extraer texto de distintos tipos de archivos y realizar búsquedas para luego implementar un ejemplo que realice búsquedas sintácticas. Por otro lado investigar sobre un factor de calidad de software importante para el diseño de los buscadores en general (la "Usabilidad"), para evaluar y comparar los componentes y características presentes en los Frameworks *PrimeFaces y IceFaces* con determinados criterios y principios que aseguren la calidad del software desde la usabilidad.

Para el ejemplo en concreto se utilizaron las siguientes versiones de componentes y librerías:

- POI v. 3.0: Librería para extraer texto de documentos "doc,docx,xls,xlsx,txt"
- Lucene v 2.4: Framework para crear los archivos índices a partir del texto extraído y realizar búsquedas sobre esos archivos.

- IDE: Eclipse LUNA
- Textmining v. 1.0: Librería para extraer texto de documentos “doc y txt”. Hace uso de POI.
- PDFBox: Librería para extraer texto de documentos “PDF”.
- JSF 2.0
- PrimeFaces 6
- Java 8

## 2.6 Antecedente: Proyecto de Minería de Textos para la categorización automática de documentos.

El principal antecedente para destacar en el desarrollo de este proyecto es la utilización de la arquitectura UIMA para transformar la información no estructurada (archivos de textos) en información estructurada (archivos XMI en el marco del proyecto). Esta transformación se realiza en una tarea de Análisis y sigue dos fases. La primera es de entrenamiento donde se utilizan un conjunto de algoritmos de aprendizaje automático aplicados a un conjunto de documentos con el fin de crear anotadores que representen entidades y relaciones de los documentos procesados para luego categorizar o clasificar los nuevos documentos en la segunda fase haciendo uso de los anotadores creados.

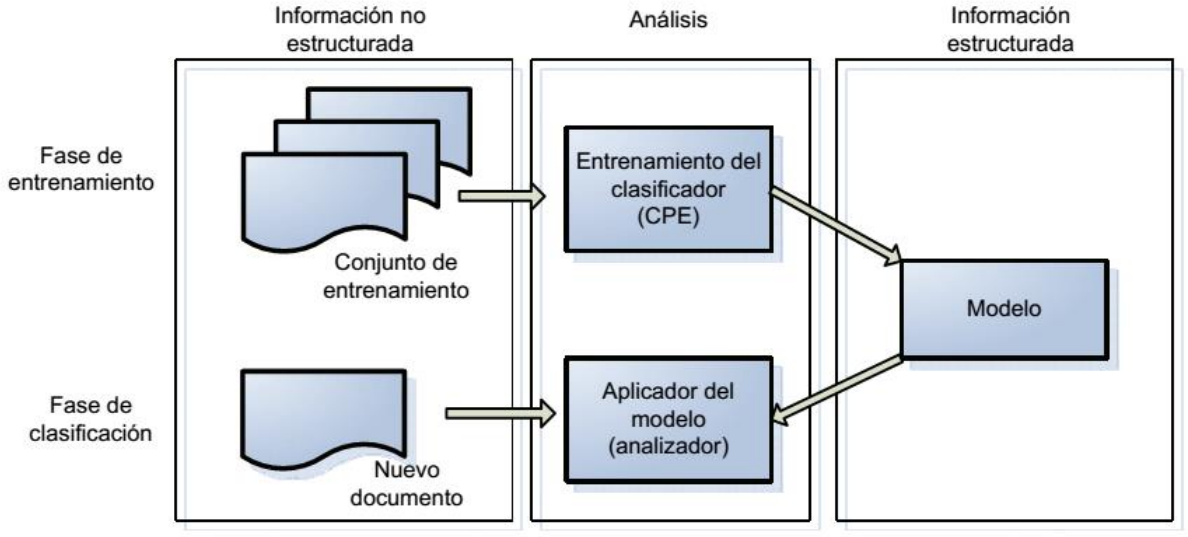



Figura 4: Implementación del aprendizaje automático en la arquitectura UIMA. [Pérez y Cardoso, 2011]

Esto permite generar los archivos anotados o clasificados (archivos XMI) a partir de un conjunto de documentos NO estructurados de Resoluciones Rectorales de la Universidad Católica de Salta. Tales archivos XMI luego son indexados por un generador de índices (como el que posee Lucene por ejemplo) en función de los anotadores definidos en la fase de entrenamiento y es posible la creación de un motor de búsqueda sobre este dominio, lo cual es específicamente el objeto de este proyecto.

# Capítulo 3 – Definición del Problema

En este capítulo se definirán los aspectos y características principales del problema a resolver identificando los objetivos y alcances del proyecto, analizando el contexto del problema y evaluando las alternativas existentes para resolverlo.

## 3.1 Definición de objetivos y alcances

*Objetivo General:*

Investigar y analizar información relacionada a la "Búsqueda Semántica" y aplicarla a la construcción de un prototipo para un problema concreto.

*Objetivos Específicos:*

- Analizar conceptos y fundamentos de las herramientas informáticas utilizadas para realizar "Búsqueda Semántica"
- Analizar tecnologías que incorporen herramientas de búsqueda semántica, para luego evaluar y catalogar sus aplicaciones en el contexto informático y social actual.
- Explorar temas de usabilidad aplicados a la construcción de interfaces para un buscador semántico.
- Diseñar e implementar un prototipo de buscador semántico para el corpus de resoluciones rectorales de la UCASAL.

*Alcances del proyecto*

El proyecto estará conformado por las siguientes tareas generales:

- Reunir y analizar información relevante relacionada a la "Búsqueda Semántica" en el marco informático.
- Investigar acerca de las herramientas y tecnologías que automatizan los procesos de búsqueda semántica y considerar la Usabilidad en la construcción de los buscadores semánticos.
- Analizar, diseñar y construir un prototipo de búsqueda de información en documentos digitales de resoluciones de la Universidad Católica de Salta implementando el concepto de "Búsqueda Semántica" con herramientas y aplicaciones de licencia libre.

## 3.2 Cronograma del proyecto

En esta sección se detallaran las tareas o actividades que conforman el proyecto del buscador semántico clasificando dichas tareas en fases. La primera fase corresponde a las tareas de inicio conformada por la investigación de las herramientas necesarias para el desarrollo del prototipo y por el estudio de los conceptos relacionados al contexto del proyecto sobre el que se venía trabajando para entender y aplicar las herramientas en la construcción de ejemplos y del prototipo.

La segunda fase corresponde al ordenamiento de las actividades que conforman el proyecto, la realización de un análisis del contexto, un análisis de riesgos y un estudio de factibilidad para determinar, entre otras cuestiones, las ventajas de invertir en él.

En la tercera fase se plantean las actividades del desarrollo del prototipo y en la cuarta fase las tareas de cierre y entrega también del prototipo.

A continuación se presenta el cronograma. Los hitos se resaltan con color azul. El diagrama de Gantt completo se presenta en el Anexo 4.

| Nombre de tarea | Duración | Comienzo | Fin |
|---|---|---|---|
| **Proyecto Buscador Semántico** | **103 días** | **mié 01/07/15** | vie 20/11/15 |
| **1-Inicio** | **50 días** | **mié 01/07/15** | mar 08/09/15 |
| **1.1- Relevamiento y recolección de información** | 8 días | mié 01/07/15 | vie 10/07/15 |
| 1.1.1- Reunión de especificaciones de líneas de trabajo | 1 día | mié 01/07/15 | mié 01/07/15 |
| 1.1.2- Estudio del trabajo de Minería de Textos | 6 días | jue 02/07/15 | jue 09/07/15 |
| 1.1.3- Adquisición de elementos de oficina, bibliografía y archivos para realizar investigación | 1 día | vie 10/07/15 | vie 10/07/15 |
| **1.2- Investigación de herramientas de búsqueda y procesamiento de texto** | 20 días | **vie 10/07/15** | jue 06/08/15 |
| 1.2.1- Preparación del entorno de desarrollo | 1 día | vie 10/07/15 | vie 10/07/15 |
| 1.2.2- Estudio de Lucene | 12 días | lun 13/07/15 | mar 28/07/15 |
| 1.2.3- Estudio de herramientas para extracción de textos de distintos documentos | 4 días | mié 29/07/15 | lun 03/08/15 |
| 1.2.4- Creación de ejemplos de indexación y búsqueda con Lucene | 3 días | mar 04/08/15 | jue 06/08/15 |
| 1.2.5- Validación de ejemplos | 0 días | jue 06/08/15 | jue 06/08/15 |
| **1.3- Investigación de herramientas para desarrollo WEB.** | 24 días | jue 06/08/15 | mar 08/09/15 |
| 1.3.1- Reunión de asesoramiento con Ing. Sebastián García | 1 día | jue 06/08/15 | jue 06/08/15 |
| 1.3.2- Estudio de Java Server Faces | 5 días | vie 07/08/15 | jue 13/08/15 |
| 1.3.3- Estudio de herramienta Ice Faces y creación de ejemplos | 3 días | vie 14/08/15 | mar 18/08/15 |
| 1.3.4- Estudio de herramienta Prime Faces y creación de ejemplos | 10 días | mié 19/08/15 | mar 01/09/15 |
| 1.3.5- Integración de Java Server Faces, Prime Faces y Lucene en la creación de un ejemplo | 4 días | mié 02/09/15 | lun 07/09/15 |
| 1.3.6- Validación de ejemplo de búsqueda e indexación WEB | 0 días | mar 08/09/15 | mar 08/09/15 |
| **2- Planeamiento del proyecto del Buscador Semántico** | **10 días** | **mié 09/09/15** | mar 22/09/15 |

| | | | |
|---|---|---|---|
| 2.1- Definición de objetivos y alcance del proyecto | 0 días | **mié 09/09/15** | mié 09/09/15 |
| 2.2- Definición del cronograma | 0 días | **mié 09/09/15** | mié 09/09/15 |
| 2.3- Análisis del proyecto | 7 días | **mié 09/09/15** | jue 17/09/15 |
| 2.4- Análisis de riesgos y definición de estrategias | 2 días | vie 18/09/15 | lun 21/09/15 |
| 2.5- Validación de actividades desarrolladas en Inicio | 0 días | mar 22/09/15 | mar 22/09/15 |
| 2.6- Validación del análisis del proyecto | 0 días | mar 22/09/15 | mar 22/09/15 |
| 2.7- Validación del análisis de riesgos del proyecto y estrategias | 0 días | mar 22/09/15 | mar 22/09/15 |
| **3- Ejecución** | **36 días** | mié 23/09/15 | mié 11/11/15 |
| 3.1- Definición del análisis y diseño del prototipo | 10 días | mié 23/09/15 | mar 06/10/15 |
| 3.2- Presentación del análisis y diseño del prototipo | 0 días | mié 07/10/15 | mié 07/10/15 |
| 3.3- Codificación del prototipo | 20 días | mié 07/10/15 | mar 03/11/15 |
| 3.4- Planificación de pruebas | 1 día | mié 04/11/15 | mié 04/11/15 |
| 3.5- Pruebas de prototipo | 2 días | jue 05/11/15 | vie 06/11/15 |
| 3.6- Correcciones del prototipo | 1 día | lun 09/11/15 | lun 09/11/15 |
| 3.7-Validacion de informes de realización de pruebas | 1 día | mar 10/11/15 | mar 10/11/15 |
| 3.8- Implementación del prototipo | 1 día | mié 11/11/15 | mié 11/11/15 |
| **4- Cierre** | **7 días** | jue 12/11/15 | vie 20/11/15 |
| 4.1- Control de objetivos, alcances y cronograma | 1 día | jue 12/11/15 | jue 12/11/15 |
| 4.2- Preparación de la presentación del prototipo | 5 días | vie 13/11/15 | jue 19/11/15 |
| 4.3- Validación y entrega del prototipo | 1 día | vie 20/11/15 | vie 20/11/15 |

Tabla 1: Cronograma de trabajo

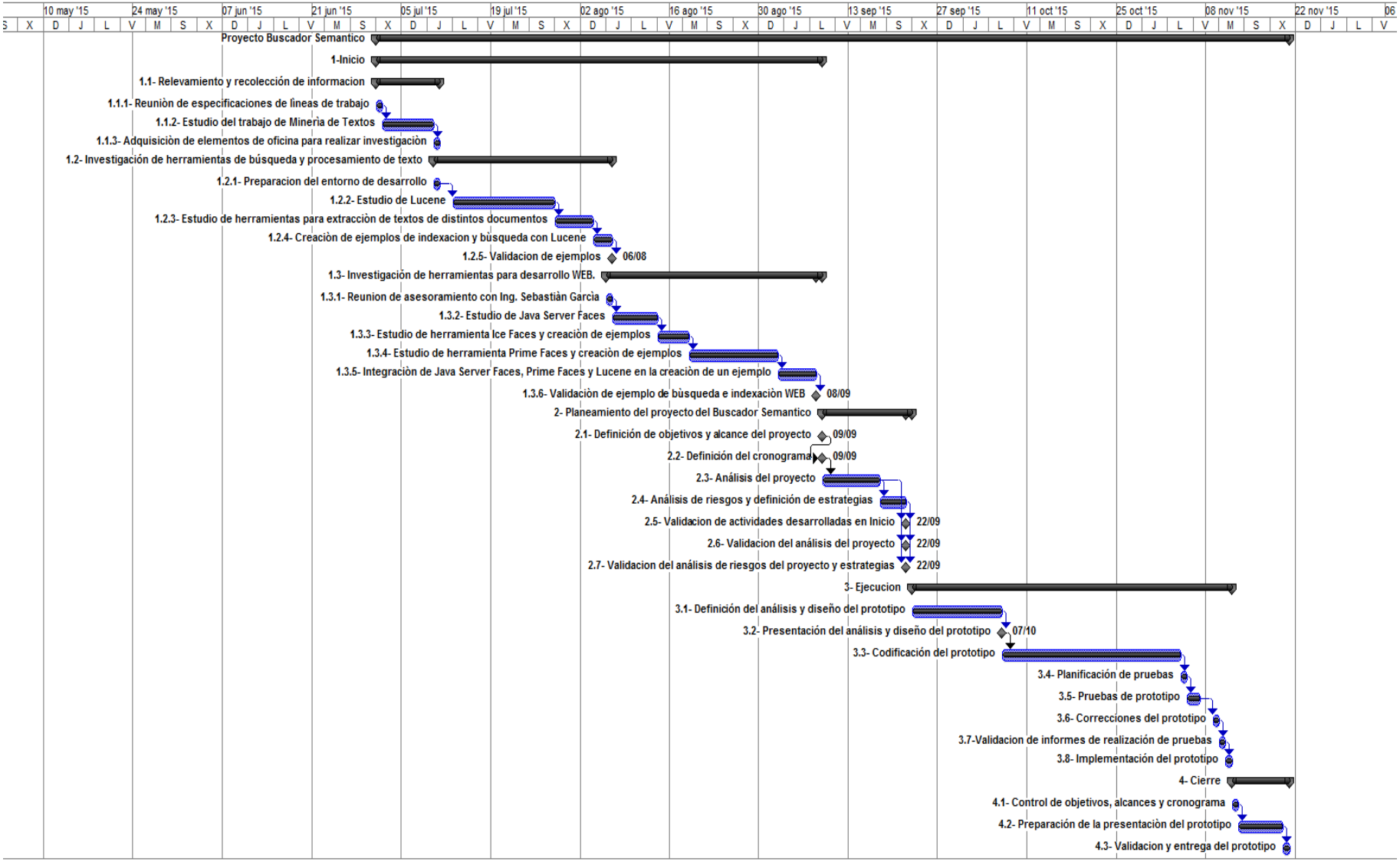



Figura 5: Diagrama de Gantt

## 3.3 Análisis de contexto

En esta sección se describirá el entorno del problema y se realizará un análisis del contexto para determinar fortalezas, oportunidades, debilidades y amenazas a tener en cuenta a la hora de plantear una estrategia de solución.

### 3.3.1 Descripción de la Organización

El prototipo del buscador semántico se desarrollará en función de una de las actividades que conformaron el alcance del proyecto de investigación "Minería de Textos para la categorización automática de documentos" dirigido por la Dra. Alicia Pérez en el departamento de investigación de la Facultad de Ingeniería, proyecto que concluyó en el año 2009.

A continuación se transcribe el alcance del proyecto nombrado remarcando en color **azul** la actividad la cual es objeto de este proyecto:

- *Fase 1: Preparación de los datos - corpus de resoluciones rectorales disponibles: formatos de los documentos, conversión si es necesaria, estructura de los documentos*
- *Fase 2: Estudio y evaluación de plataformas de software libre para el lenguaje natural y minería de textos (GATE, UIMA) y de su interacción con la herramienta de software libre para aprendizaje automático WEKA.*
- *Fase 3: Revisión de la literatura relevante, con reuniones periódicas del equipo (al menos quincenales) y/o un seminario, posiblemente abierto a otros interesados, para trabajar en grupo los materiales más relevantes.*
- *Fase 4: Desarrollo de un sistema base de categorización de documentos que incluye:*
    - *extracción de características (bag of words inicialmente)*
    - *exploración de técnicas de clustering para detectar en automáticamente en lo posible las categorías de los documentos. Dependiendo de la evaluación por un experto humano de los clusters obtenidos, se continuará con*
    - *aprendizaje automático de clasificadores (utilizando Máquinas de Vectores Soporte o Naïve Bayes en primera instancia), que requiere un conjunto de entrenamiento en que los documentos han sido etiquetados manualmente.*
- *Fase 5: Evaluación del sistema base en el corpus de resoluciones rectorales. Se evaluará:*
    - *el funcionamiento del sistema en base a la clasificación de los documentos comparado con la clasificación ideal (gold standard) creada a mano por un usuario experto utilizando métricas de precisión, cobertura (recall) y su combinación (F-measure).*

- *el funcionamiento del sistema como ayuda a la búsqueda de documentos mediante pruebas de usabilidad y evaluación por los usuarios de las sugerencias del sistema*

- *Fase 6: Mejora del prototipo y evaluación de las mejoras mediante la implementación y evaluación de otras técnicas de extracción de características y de aprendizaje automático.*

- ***Fase 7: Exploración de opciones de visualización de los resultados de la categorización con énfasis en la usabilidad y desarrollo de una interfaz para el sistema de resoluciones.***

- *Fase 8: Análisis de la transferencia de las técnicas y métodos desarrollados a otras áreas de la UCS con vistas a un aporte a la gestión del conocimiento.*

### 3.3.2 Análisis F.O.D.A.

Factores Internos:

**Fortalezas:**

1. El nivel académico de las personas que conforman el equipo del proyecto es muy bueno.
2. Se cuenta con toda la información relacionada a búsqueda semántica, que requiera este proyecto.
3. El personal del departamento de investigación posee computadoras personales y conexión a Internet en todo momento.
4. Se posee conocimiento y dominio del lenguaje Java.
5. Las personas que integran el departamento, son además profesores y alumnos de la carrera de Ingeniería en Informática.
6. Existe muy buena relación entre las personas del departamento.

**Debilidades:**

1. La mayoría del material bibliográfico se encuentra en inglés.
2. La mayoría del personal del departamento posee y desempeña otros trabajos.

Factores Externos:

**Oportunidades**

1. La búsqueda semántica es una nueva forma de búsqueda que las empresas pretenden implementar.
2. Existencia de herramientas de software libre para desarrollar el prototipo en su integridad.
3. Apoyo del personal de la Universidad, ya que la organización central donde se desarrolla el proyecto es en el departamento de investigación de la Facultad de Ingeniería.
4. Surgimiento de nuevas herramientas y tecnologías para el uso e implementación de este tipo de buscador.
5. La búsqueda que se implementa es muy ágil.

**Amenazas**

1. Nueva forma de búsqueda, no implementada en gran escala en procesos operativos dentro del mercado.
2. Al ser un tipo de búsqueda donde su implementación está todavía en proceso de investigación, las empresas asumen riesgos significativos a la hora de evaluar la implementación de este tipo de buscador.
3. Las librerías utilizadas en la codificación del prototipo se actualizan rápidamente y existen muchas versiones de las mismas lo cual podría generar incompatibilidades entre librerías de ser actualizadas.

### 3.3.3 Matriz de Estrategias

| **Factores Internos**<br><br>**Factores Externos** | **FORTALEZAS** | **DEBILIDADES** |
|---|---|---|
| **OPORTUNIDADES** | Contribuir al proyecto de investigación de "Minería de datos para la categorización automática de documentos", utilizando información y herramientas proporcionadas en su mayoría por el departamento de investigación. | Establecer un cronograma de encuentros para desarrollar este proyecto con el departamento de investigación, en función de los horarios disponibles de cada uno. |
| **AMENAZAS** | Actualizarse periódicamente acerca las nuevas librerías y herramientas que contemplan y utilizan esta forma de búsqueda. | Incentivar el uso e implementación de este tipo de búsqueda en los procesos operativos, desarrollando una interfaz con un alto nivel de usabilidad, contribuyendo de esta manera, a incrementar la calidad del prototipo. |

Tabla 2: Matriz FODA

## 3.4 Análisis Técnico-Operativo

A continuación se definirán las características que debería tener un prototipo que brinde una solución óptima al problema que se está analizando y se enumerarán las posibles alternativas tecnológicas para el desarrollo del prototipo.

### 3.4.1 Prototipo Óptimo

El prototipo óptimo deberá contar con las siguientes funcionalidades y características principales:

- Procesamiento de documentos XMI.
- Interfaz diseñada contemplando la usabilidad del buscador. Los conceptos a tener en cuenta respecto a este factor de calidad de software se detallan en el Anexo 2 de este trabajo.
- El prototipo debe funcionar en un entorno web y debe ser independiente del sistema operativo en donde se ejecute.
- Uso de un servidor dedicado para este buscador.

### 3.4.2 Alternativas tecnológicas

Algunas de las alternativas tecnológicas para el desarrollo del prototipo óptimo son:

- Lenguajes de Programación orientadas a objetos que interactúan con Internet
  - Visual.net
  - Java
- Frameworks, herramientas y librerías para realizar búsquedas:
  - UIMA (IBM).
  - Semantic Search (IBM).
  - Lucene (Apache).
- Herramientas, ide ‘s y librerías para desarrollar aplicaciones en entorno web.
  - ASP.Net
  - IceFaces
  - Visual Studio 2008
  - Eclipse “Luna”

- PrimeFaces

- Sistemas Operativos
  - Windows 7, 8 o 10.
  - Windows Server 2012
  - Linux (Cualquier distribución).

- Servidores de aplicaciones
  - Apache TomCat.
  - Glashfish
  - BEA Weblogic Server
  - Borland AppServer
  - Allaire ColdFusion
  - Lotus Domino
  - Netscape application server
  - Oracle application server
  - Sybase Enterprise Server
  - IBM WebSpher
  - WildFly

# Capítulo 4 – Solución Propuesta

A continuación se presentarán los aspectos y características de la solución propuesta al problema analizado en el capítulo anterior definiendo la metodología a utilizar, las herramientas y librerías para construir el prototipo y los recursos necesarios para su implementación. Además se realizará un estudio de factibilidad del proyecto, un análisis de riesgos y una definición de actividades estratégicas en las etapas más importantes de este proyecto.

## 4.1 Definición del prototipo a desarrollar

El prototipo está construido con las siguientes librerías y herramientas. Se justifica además la elección de cada herramienta:

- Lenguaje de Programación orientadas a objetos que interactúan con Internet
    - Java: Ya que permite independizar la ejecución de los Sistemas Operativos en donde se instale la aplicación, siempre y cuando la máquina virtual de Java esté instalada en el Sistema Operativo.
- Librerías para realizar búsquedas:
    - Lucene (Apache): Desde el principio del proyecto la herramienta asignada y estudiada para desarrollar este prototipo fue Lucene. Además se encuentra escrita en Java y es de software libre.
- Herramientas y librerías para desarrollar aplicaciones en entornos web.
    - PrimeFaces: Este framework presenta más flexibilidad y soporte para desarrollar aplicaciones en JSF en comparación a los demás. Además los componentes prediseñados que ofrece son muy versátiles y fáciles de implementar.
    - Eclipse “Luna”: Eclipse en general es un entorno muy flexible y los componentes que se deben utilizar en este proyecto se agregan y adaptan fácilmente.
- Sistema Operativo
    - Windows 7 Ultimate x64: El desarrollo y la implementación de este prototipo se efectuó en un sistema operativo Windows 7 Ultimate x64 desde el principio.

- Servidor de aplicaciones

    - Apache TomCat 7.0: Se integraba a Eclipse casi automáticamente y la compatibilidad con PrimeFaces y JSF era adecuada. Como alternativa también se configuro el servidor de aplicaciones WildFly ya que posee herramientas que permiten empaquetar los proyectos de páginas web dinámicas de Java junto con todos los servicios que consume dicho proyecto en un archivo *.jar. Principalmente a los fines de presentarlo al tribunal evaluador sin que requerían desplegar la aplicación Web en forma manual.

Para la implementación se propone la utilización de los siguientes recursos de hardware:

| Equipo | Especificaciones | Sistema Operativo | Estado |
|---|---|---|---|
| PC-Buscador | PC HP 600 SFF i5 4570 500GB 4GB | Windows 7 Ultimate x64 | Nuevo |
| Monitor | MONITOR LG 19" LED 19EN33S | | Nuevo |
| UPS | Ups Apc Bx1100ci 1100VA | | Nuevo |

Tabla 3: Requerimientos de Hardware y Software

Recursos Humanos

La estructura de trabajo planteada para el desarrollo de este proyecto es la siguiente:

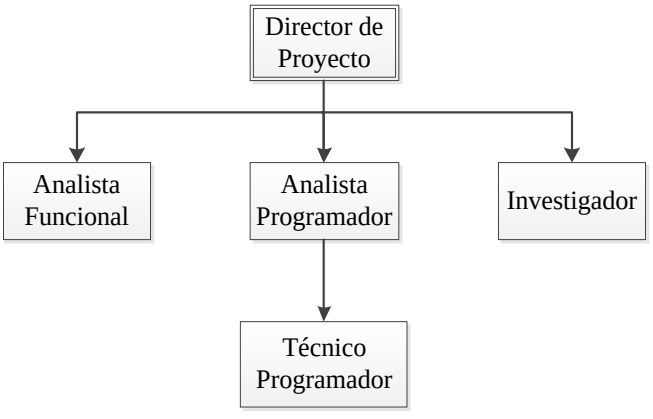



Figura 6: Recursos humanos

Los profesionales son independientes y deberán presentar facturas por los servicios que presten en la realización de sus actividades. El responsable general del proyecto frente a la Universidad es el Director de proyecto.

Matriz de responsabilidades

Las responsabilidades que tendrá cada profesional en las actividades planificadas y definidas se clasificaran de la siguiente manera: S: Supervisa; E: Encargado; P: Participa; O: Opinión requerida

| **Involucrado** | **Tarea** | 1.1.1 | 1.1.2 | 1.1.3 | 1.2.1 | 1.2.2 | 1.2.3 | 1.2.4 | 1.2.5 | 1.3.1 | 1.3.2 | 1.3.3 | 1.3.4 | 1.3.5 | 1.3.6 | 2.1 | 2.2 | 2.3 | 2.4 | 2.5 | 2.6 | 2.7 | 3.1 | 3.2 | 3.3 | 3.4 | 3.5 | 3.6 | 3.7 | 3.8 | 4.1 | 4.2 | 4.3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Director de Proyecto | | E | S | S | S | S | S | S | S | S | S | S | S | S | S | E | E | E | E | E | E | E | S | S | S | S | S | S | S | S | E | E | E |
| Analista funcional | | P | | | P | P | P | P | P | P | P | P | P | P | P | P | P | P | P | P | P | P | E | E | P | P | P | P | P | P | P | P | P |
| Analista programador | | P | | | O | O | O | O | O | O | O | O | O | O | O | P | P | P | P | P | P | P | P | P | E | E | E | E | E | E | P | P | P |
| Técnico programador | | | | | | | | | | | | | | | | | | | | | | | P | P | P | P | P | P | P | P | P | P | |
| Investigador | | P | E | E | E | E | E | E | E | E | E | E | E | E | E | P | P | P | P | P | P | P | O | O | O | O | O | O | O | O | P | P | P |

Tabla 4: Matriz de responsabilidades del proyecto

## 4.2 Metodología de Diseño

La metodología del análisis y diseño del prototipo será orientado a objetos.

Es decir, que se realizará un diseño definiendo clases, objetos, métodos, atributos y mensajes, los cuales serán los componentes que detallarán junto con otras herramientas los procesos funcionales de este prototipo.

Para realizar este análisis y diseño se utilizará el lenguaje de modelado unificado (UML) definiendo de ésta manera casos de uso, diagramas de actividades, diagramas de clases, diagramas de interacción y diseño de interfaces.

## 4.3 Análisis de factibilidad

En esta sección se analizara la factibilidad del proyecto en el contexto de 3 variables: económicas, técnicas y operativas. Se determinara finalmente si es conveniente la realización del proyecto desde distintos contextos.

### 4.3.1 Factibilidad Económica

A continuación se presentará el análisis de costos del proyecto y luego se evaluará la factibilidad económica de realizarlo con respecto a valores contables/administrativos calculados teniendo en cuenta que se trata de un equipo de trabajo independiente y el que asume la responsabilidad total frente al cliente es el Director del Proyecto en condición de Responsable Inscripto en el marco impositivo-contable.

Costos de recursos humanos(*):

| Profesional | $ / hs | hs / día | Días | Subtotal |
|---|---|---|---|---|
| Director de Proyecto | 270,00 | 4 | 103 | 111240,00 |
| Analista funcional | 180,00 | 4 | 96 | 69120,00 |
| Analista programador | 180,00 | 4 | 54 | 38880,00 |
| Técnico programador | 120,00 | 4 | 42 | 20160,00 |
| Investigador | 150,00 | 4 | 67 | 40200,00 |
| Total | | | | 279600,00 |

Tabla 5: Costos de Recursos Humanos

(*): Estos valores fueron obtenidos de la tabla de honorarios indicativos de profesionales informáticos de Salta: http://www.copaipa.org.ar/informatica/

Costos de Hardware:

| Renglón | Componente | Unidades[ud] | Precio unitario[$] | Costo total[$] |
|---|---|---|---|---|
| 1 | PC HP 600 SFF i5 4570 500GB 4GB | 1 | 9911,00 | 9911,00 |
| 2 | Monitor LG 19" LED 19EN33S | 1 | 4065,60 | 4065,60 |
| 3 | Notebook Toshiba Intel 4gb 500gb 15,6 Led Hdmi Wifi | 5 | 9499,00 | 47495,00 |
| 4 | Modem 4G Personal | 5 | 900,00 | 4500,00 |
| 5 | Ups Apc Bx1100ci 1100VA | 1 | 3200,00 | 3200,00 |
| **Total** | | | | 69171,60 |

Tabla 6: Costos de Hardware

Costos de Software:

| Renglón | Componente | Unidades [ud] | Precio unitario[$] | Costo total[$] |
|---|---|---|---|---|
| 1 | Licencia Windows 10 (Con downgrade a Windows 8 o Windows 7) – Licencia gratuita por el programa académico de Microsoft | 9 | 0,00 | **0,00** |

Tabla 7: Costos de Software

Costos varios:

| Renglón | Componente | Unidades[ud] | Precio unitario[$] | Costo total[$] |
|---|---|---|---|---|
| 1 | Elementos de Limpieza | 103 días | 980,00 | 980,00 |
| 2 | Gastos de electricidad | 103 días | 1090,00 | 1090,00 |
| 3 | Gastos de internet | 103 días | 1308,00 | 1308,00 |
| 4 | Gastos de oficina (café, papel higiénico, cuadernos, lapiceras, etc) | Para 103 días | 900,00 | 900,00 |
| Total | | | | 4278,00 |

Tabla 8: Costos varios

**Servicio de Consultoría y comisión por venta de hardware**

Este costo corresponde al 20% del costo total de RRHH, 1 sueldo mensual del director de proyecto para la preparación del proyecto y el 20% del costo total de otros insumos. Además se incluye un 20% del costo total de los costos de hardware en concepto de comisión por venta de equipos de hardware.

| Renglón | Componente | Unidades[ud] | Precio unitario[$] | Costo total[$] |
|---|---|---|---|---|
| 1 | Servicio de Consultaría | 1 | 97150,80 | 97150,80 |
| 2 | Comisión por venta de Hardware | 1 | 13834,32 | 13834,32 |
| Total | | | | 110985,12 |

Tabla 9: Costos de consultoría y comisión por venta de equipos

**Costo total:**

| Renglón | Tipo de costo | Subtotal [$] |
|---|---|---|
| 1 | Costos de recursos humanos | 279600,00 |
| 2 | Costos de hardware | 69171,60 |
| 3 | Costos de software | 0,00 |
| 4 | Otros | 4278,00 |
| 5 | Costos de Consultoría y Comisión por venta de hardware | 110985,12 |
| **Total** | | **464034,72** |

Tabla 10: Costo total

Todos los valores son en pesos Argentinos con IVA del 21%.

Para definir la financiación del proyecto se dividirán las instancias de pago en 4 partes iguales como se detalla en el siguiente cuadro estableciendo el último pago en el momento en que se termina el proyecto.

| Parte | Monto**[$]** | Fecha de pago límite |
|---|---|---|
| 1 | 116008,68 | 01/07/2015 |
| 2 | 116008,68 | 01/08/2015 |
| 3 | 116008,68 | 01/09/2015 |
| 4 | 116008,68 | 22/10/2015 |
| **Total** | **464034,72** | |

Tabla 11: Financiación del proyecto

**Análisis Costo-Beneficio**

A continuación se presenta el flujo de caja de los ingresos y egresos relacionados al equipo de trabajo que desarrollara el prototipo. Para este análisis se tienen en cuenta los costos de Recursos Humanos y los costos varios antes detallados para determinar la factibilidad y los beneficios cuantitativos que tiene para el equipo de desarrollo realizar este proyecto. El periodo 0 es el lapso en el que se realizan los trabajos de preparación del proyecto antes de presentarlo al cliente, es por ello que se considera como inversión inicial en el periodo 0 un egreso correspondiente al sueldo mensual del Director del Proyecto.

| **Concepto \ Período [mes]** | **0** | **1** | **2** | **3** | **4** |
|---|---|---|---|---|---|
| **Ingresos** | | | | | |
| Pagos del proyecto | 0,00 | 116008,68 | 116008,68 | 116008,68 | 116008,68 |
| **Total de ingresos** | **0,00** | **116008,68** | **116008,68** | **116008,68** | **116008,68** |
| **Egresos** | | | | | |
| Inversión inicial | 33646,00 | 0,00 | 0,00 | 0,00 | 0,00 |
| Costos de RRHH | 0,00 | 69900,00 | 69900,00 | 69900,00 | 69900,00 |
| Costos varios | 0,00 | 1246,00 | 1246,00 | 1246,00 | 540,00 |
| Costos de Hardware | 0,00 | 17292,90 | 17292,90 | 17292,90 | 17292,90 |
| **Total de egresos** | **33646,00** | **88438,90** | **88438,90** | **88438,90** | **87732,90** |
| **Flujo de caja [$]** | **-33646,00** | **27569,78** | **27569,78** | **27569,78** | **28275,78** |

Tabla 12: Flujo de caja – Equipo de desarrollo

A continuación se calculan la Tasa Interna de Retorno (TIR) y el Valor Actual Neto (VAN) del flujo de caja realizada para determinar si es conveniente invertir en el proyecto.

| | |
|---|---|
| TIR | 72,74% |
| VAN | $ 37724,84 |

Tabla 13: TIR-VAN de equipo de desarrollo

La TIR indica que la inversión del proyecto es rentable al tener un valor que supera a la tasa de corte propuesta del 20%.

El VAN indica un valor positivo, es decir que la inversión permitirá ganar dinero.

Por otra parte se calculara la ganancia Bruta y la Ganancia Neta del proyecto en base a la facturación del Director del Proyecto a la Universidad en el marco impositivo y legal de registración contable de Argentina y de la Provincia de Salta.

| Costos | Monto ($) |
|---|---|
| Costo total del proyecto (que debe pagar el cliente) | 464034,72 |
| Costo total de RRHH, Costos Varios y Costos de Hardware( que debe pagar el Director de Proyecto) | 353049,60 |
| Ganancia Bruta | 110985,12 |
| | |
| Impuestos | Monto ($) |
| Aportes (por 4 meses) | 33047,48 |
| Impuestos a las ganancias | 28306,64 |
| Devolución de IVA del 21% en compra de Hardware | -14526,036 |
| Total de impuestos | 46828,084 |
| | |
| Ganancia Neta | **64157,04** |

Tabla 14: Ganancias Bruta-Neta del equipo de desarrollo

Para finalizar se detallará el retorno de la inversión:

| Periodo | Flujo de Caja | Flujo de caja acumulado |
|---|---|---|
| 0 | -33646 | -33646 |
| 1 | 116008,68 | 82362,68 |
| 2 | 116008,68 | 198371,36 |
| 3 | 116008,68 | 314380,04 |
| 4 | 116008,68 | 430388,72 |

Tabla 15: Retorno de la inversión – Equipo de desarrollo

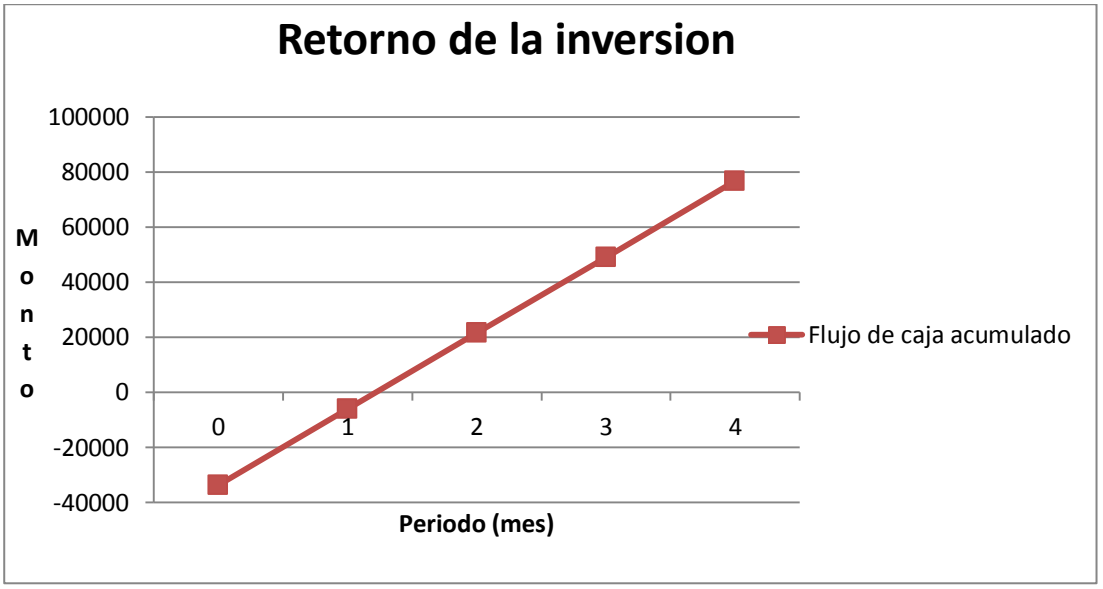



Figura 7: Retorno de la inversión – Equipo de desarrollo

Según el cálculo de intersección de la función del flujo de caja acumulado con el eje X, se puede determinar que el periodo de recupero de la inversión inicial será de 1,22 meses.

Ahora se realizará el análisis del flujo de caja para determinar los valores del TIR y VAN que garanticen que la inversión de la Universidad podrá ser solventada y le permitirá generar beneficios económicos.

Primeramente se deberán identificar los ahorros económicos que generara la implementación del prototipo. Este ahorro será principalmente en cuestiones operativas ya que se consideran 2 personas trabajando en la búsqueda de resoluciones rectorales en base a peticiones o solicitudes específicas de búsqueda, por ejemplo, obtener todas las resoluciones en las que el Ingeniero Juan Pérez aparezca. Considerando este ejemplo sin la aplicación del buscador las personas encargadas de realizar la búsqueda deberían realizarla de manera manual y no aseguraría eso eficiencia en la obtención de resultados. En este caso ambas personas tardarían 60 min en lograr obtener los resultados de forma manual. Con el buscador obtendrían los resultados en por lo menos 5 min. A continuación se mostrara el ahorro operativo que generaría la aplicación del buscador para este caso.

| **Concepto** | **Valor** |
|---|---|
| Costo por hora (empleado administrativo) ($) | 40,00 |
| Horas de operación ahorradas al mes | 176 |
| Total por mes ($) | 7040,00 |
| Total por año ($) | 84480,00 |
| Cantidad de operarios | 2 |
| **Total ($)** | **168960,00** |

Tabla 16: Ahorro operativo

Además es altamente probable que otros sectores de la universidad requieran adquirir el buscador para otras funciones que contribuyan a agilizar sus trabajos y brindar respuestas más eficientes a sus empleados y/o alumnos, es por ello que se identificara como un tipo de ahorro la captación de nuevos alumnos o sectores de la Universidad.

Para analizar entonces la rentabilidad de la inversión por parte de la Universidad se detallara el flujo de caja teniendo en cuenta los ahorros señalados anteriormente y a partir de allí se calcularan la TIR y el VAN. Cabe aclarar que el periodo 1 corresponde al momento en el que buscador se encuentra implementado y funcionando, es decir, cuando el proyecto termina.

| **Concepto \ Período [año]** | **0** | **1** | **2** | **3** | **4** |
|---|---|---|---|---|---|
| **Ingresos** | | | | | |
| Captación de nuevos clientes/sectores de la Universidad | 0,00 | 25000,00 | 50000,00 | 75000,00 | 100000,00 |
| Ahorro en costos de operación | 0,00 | 168960,00 | 168960,00 | 168960,00 | 168960,00 |
| **Total de ingresos** | **0,00** | **193960,00** | **218960,00** | **243960,00** | **268960,00** |
| **Egresos** | | | | | |
| Inversión inicial | 464034,72 | 0,00 | 0,00 | 0,00 | 0,00 |
| **Total de egresos** | **464034,72** | **0,00** | **0,00** | **0,00** | **0,00** |
| **Flujo de caja [$]** | **-464034,72** | **193960,00** | **218960,00** | **243960,00** | **268960,00** |

Tabla 17: Flujo de caja – Universidad

| | |
|---|---|
| TIR | 32,4 % |
| VAN | $ 120541,51 |

Tabla 18: TIR-VAN Universidad

Se puede observar que la TIR supera la tasa de corte propuesta de 20 % y esto determina que es rentable la inversión. Por otra parte el VAN es positivo y la inversión permitirá entonces generar dinero.

Por último se analizara el retorno de la inversión de la Universidad para determinar el periodo de recupero de la inversión.

| Periodo | Flujo de Caja | Flujo de caja acumulado |
|---|---|---|
| 0 | -464034,72 | -464034,72 |
| 1 | 193960,00 | -270074,72 |
| 2 | 218960,00 | -51114,72 |
| 3 | 243960,00 | 192845,28 |
| 4 | 268960,00 | 461805,28 |

Tabla 19: Retorno de la inversión - Universidad

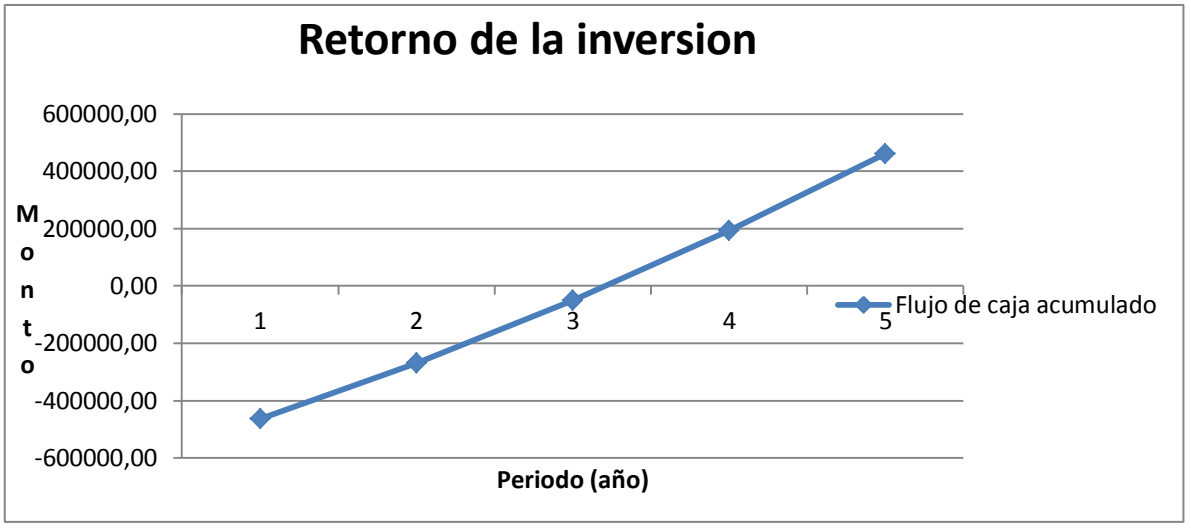



Figura 8: Retorno de la inversión - Universidad

Periodo de recupero de la inversión: 2,2 años.

Con la aplicación de este sistema existen además otros beneficios que no se ven reflejados en variables cuantitativas y financieras como ser la inmediatez y disponibilidad de la información que se desea encontrar. Esto favorece de forma relevante a la agilización de los tiempos en la toma de decisiones y es un factor determinante en cuestiones de ahorro económico, financiero y operativo de la universidad.

*Factibilidad Técnica*

Es técnicamente factible ya que se dispone de todas las herramientas antes mencionadas y también de información pertinente al desarrollo del prototipo gracias al aporte del departamento de investigación.

*Factibilidad Operativa*

El recurso humano disponible está capacitado para trabajar con dichas herramientas y dominar el inglés para avanzar con las investigaciones relacionadas a la búsqueda semántica. Desde el punto de vista de los usuarios de la universidad que usarían el sistema se debe remarcar que uno de los criterios relevantes para desarrollar el prototipo es la usabilidad, lo que contribuye a que las interfaces sean amigables para esos usuarios y la complejidad de entender su funcionamiento sea reducida a un alto grado.

## 4.4 Análisis y Diseño del prototipo

A continuación se presenta el funcionamiento general del prototipo orientado a un proyecto de software, siguiendo la metodología definida en la sección 4.2

Para modelar en UML se utilizó la herramienta en versión de evaluación de Visual Paradigm.

### 4.4.1 Diagrama de casos de usos

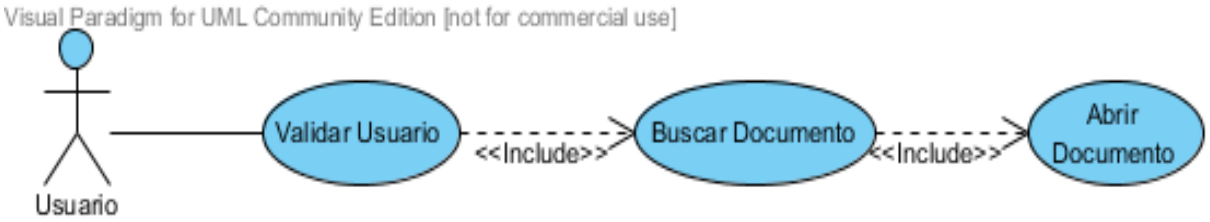



Figura 9: Diagrama de casos de usos

### 4.4.2 Especificación de casos de uso

**CU1-Validar Usuario.**

| Total | | | |
|---|---|---|---|
| **ID de Caso de Uso** | 1 | | |
| **Nombre de CU** | Validar Usuario | | |
| **Actores principales** | Usuario | | |
| **Actores secundarios** | | | |
| **Descripción** | El usuario es validado en el sistema para poder usarlo | | |
| **Pre-condiciones** | Usuario y contraseña de usuario conocidos | | |
| **Flujo de eventos** | | Acción | Respuesta |
| | 1 | Usuario Ingresa Usuario y contraseña | |
| | 2 | | Se validan los datos ingresados |
| | 3 | | Se muestra interfaz de |

| | |
|---|---|
| | búsqueda |
| **Post-condiciones** | Usuario logueado listo para operar. |
| **Excepciones** | 2: Datos inválidos: Se muestra un mensaje de error, diciendo que el usuario ingresó incorrectamente sus datos. Si sucede 3 veces consecutivas esto, se bloquea el acceso al sistema. |
| **Autor** | David Zamar |
| **Fecha** | 25/09/2015 |

**CU2- BuscarDoc**

| Total | | | |
|---|---|---|---|
| **ID de Caso de Uso** | 2 | | |
| **Nombre de CU** | Buscar Documento | | |
| **Actores principales** | Usuario | | |
| **Descripción** | El usuario ingresa parámetros de búsqueda y en base a los mismos se muestran los resultados. | | |
| **Pre-condiciones** | El usuario debe estar logueado. Se deben poseer parámetros de búsqueda. Deben existir documentos indexados | | |
| **Flujo de eventos** | | Acción | Respuesta |
| | 1 | El usuario carga e ingresa los parámetros de búsqueda de una resolución. | |
| | 2 | | Se procesan los datos de ingreso. |
| | 3 | | Se realiza una búsqueda de |

| | | |
|---|---|---|
| | | dichos parámetros en los índices creados. |
| | 4 | Se muestran los resultados. |
| **Post-condiciones** | Mostrar los resultados encontrados | |
| **Excepciones** | 3 - No existen archivos relacionados a la búsqueda: Aparece un mensaje diciendo que la búsqueda no tuvo éxito. No hay archivos relacionados con la misma. | |
| **Autor** | David Zamar | |
| **Fecha** | 25/09/2015 | |

**CU3- AbrirDoc**

| Total | | | |
|---|---|---|---|
| **ID de Caso de Uso** | 3 | | |
| **Nombre de CU** | Abrir Documento | | |
| **Actores principales** | Usuario | | |
| **Descripción** | El usuario selecciona un resultado encontrado, y abre el archivo. | | |
| **Pre-condiciones** | Debe haberse realizado una búsqueda exitosa | | |
| **Flujo de eventos** | | Acción | Respuesta |
| | 1 | El usuario selecciona un resultado encontrado | |
| | 2 | | Se procesa la selección del usuario. |
| | 3 | | Se abre o se |

| | |
|---|---|
| | descarga la resolución en cuestión. |
| **Post-condiciones** | Resolución abierta para leer. |
| **Autor** | David Zamar |
| **Fecha** | 25/09/2015 |

### 4.4.3 Diagrama de clases

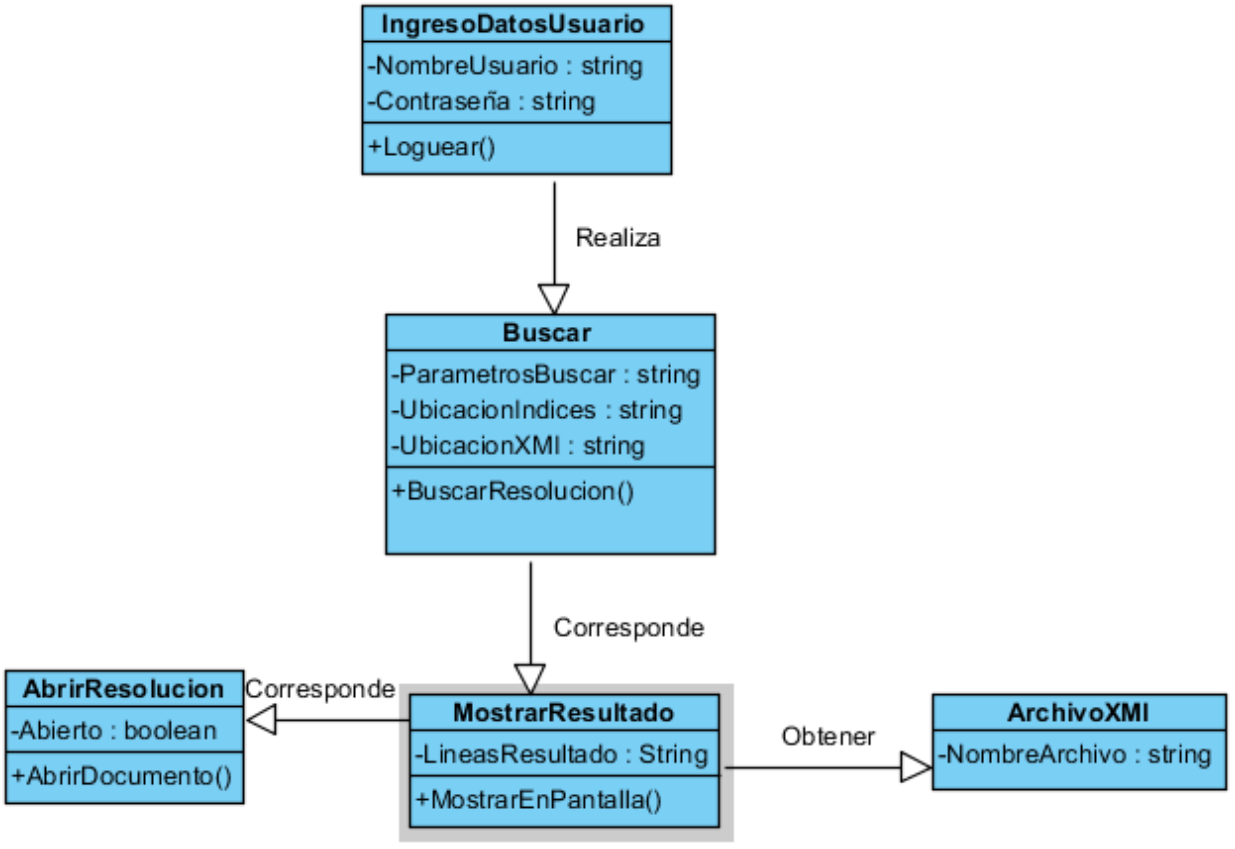



Figura 10: Diagrama de clases

### 4.4.4 Diagramas de secuencia

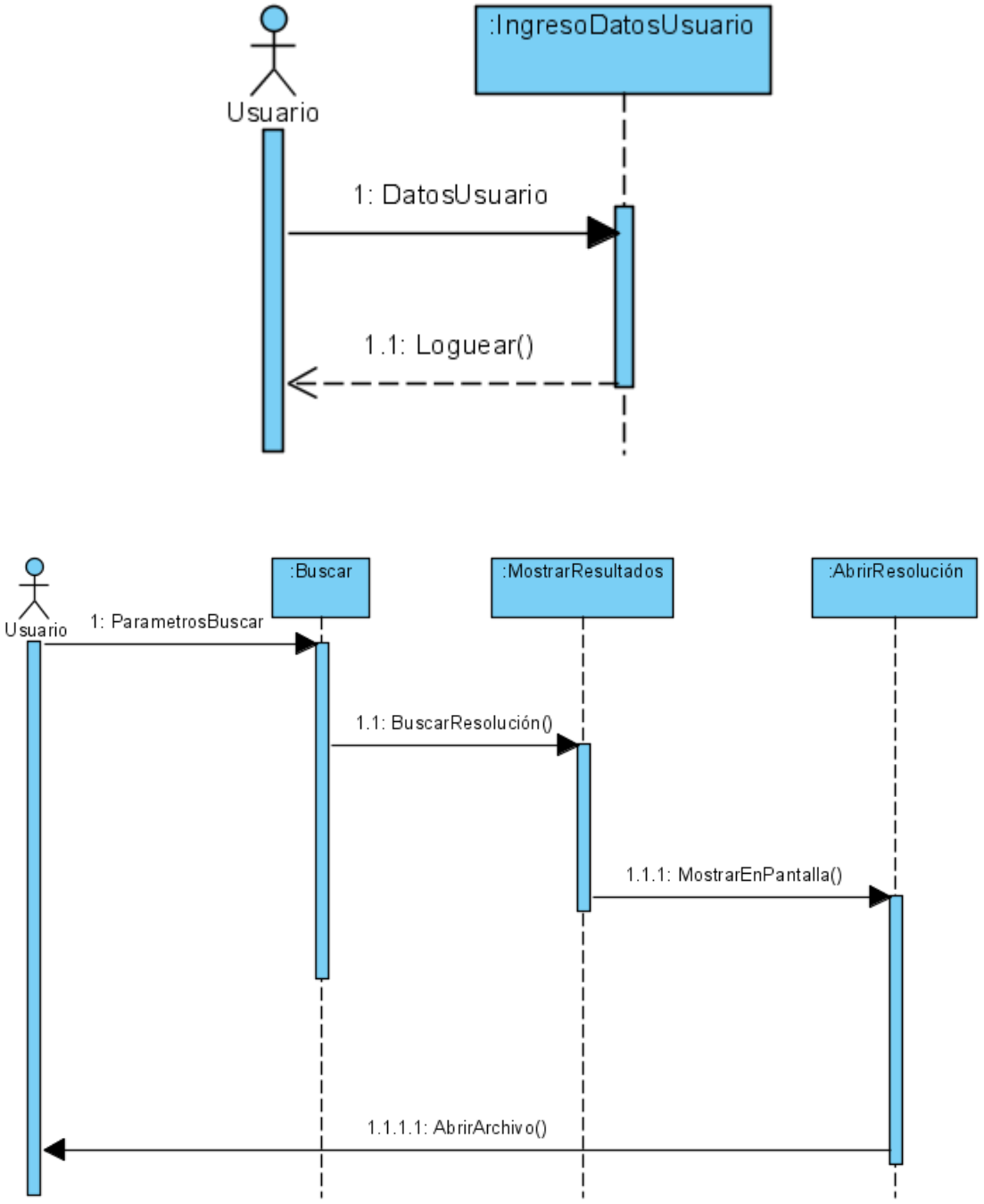



Figura 11: Diagramas de secuencia

## 4.5 Análisis de Riesgos

Para implementar este proyecto es necesario evaluar una serie de riesgos a los cuales está sujeto el desarrollo del mismo. A continuación se van a enumerar y definir los riesgos para luego ponderarlos y analizarlos según la probabilidad de ocurrencia y el impacto que genere en los resultados de los objetivos de este proyecto. Cabe aclarar que este análisis forma parte de una de las actividades de la metodología para el desarrollo de proyectos (PMI) descripta en el anexo 2 de este escrito.

### 4.5.1 Clasificación de los riesgos

Los riesgos del proyecto van a clasificarse de acuerdo al tipo que correspondan y luego se realizará un análisis para evaluar, valorar y ponderar ordenadamente cada uno de ellos. Los riesgos ponderados serán encausados para asociar un plan de contingencia en caso de que el evento riesgoso ocurra.

Los riesgos son:

*Técnicos:*

1. Actualizaciones en las versiones de las librerías utilizadas en el desarrollo del prototipo que generen incompatibilidades con las demás herramientas y la codificación.
2. Cortes de energía eléctrica en lugar en donde se instale la PC que ejecuta el prototipo.
3. Aparición de errores que pueda tener el Sistema Operativo donde se ejecuta el prototipo y ocasione una caída del servidor de aplicaciones.
4. Apariciones de fallas en el hardware utilizado para el desarrollo del prototipo.

*Del negocio:*

5. Alguna o varias herramientas o librerías utilizadas en la implementación del prototipo podrían licenciarse y dejar de ser libre.

*Del proyecto:*

6. Extraviar o perder información relacionada con el proyecto debido a fallas del hardware o errores del usuario.

7. Fracaso o disolución del proyecto de "Minería de datos para la categorización automática de documentos", en el cual está inmerso este proyecto.

8. El prototipo no responde al cumplimiento de los objetivos del proyecto.

### 4.5.2 Matriz de evaluación de riesgos

Para que el lector pueda entender la "matriz de evaluación de riesgos", se explicarán los conceptos a continuación:

Los valores de probabilidad de ocurrencia de que suceda el evento que pone en riesgo al proyecto se los cataloga de la siguiente manera:

1. Improbable
2. Moderado
3. Probable
4. Muy probable

Los valores de impacto en el proyecto en caso de que alguno de los riesgos se presente se los cataloga de la siguiente manera:

1. Insignificante
2. Moderado
3. Mayor
4. Catastrófico

Los rangos de valores de severidad del riesgo indican la importancia de encausar un plan de contingencia para contra restar los inconvenientes que pueda generar el suceso del riesgo. Es el resultado de multiplicar la Probabilidad de ocurrencia con el Valor de Impacto. En base a dichos resultados se clasificará la severidad de la siguiente manera:

[1-3]: Insignificante

(3-6]: Moderado

(6-9]: Alto

(9-12]: Muy alto

| Número de riesgo | Probabilidad de ocurrencia | | Impacto | | Severidad | |
|---|---|---|---|---|---|---|
| | **Clasificación** | **Valor** | **Clasificación** | **Valor** | **Clasificación** | **Valor** |
| 1 | Muy probable | 4 | Mayor | 3 | Muy alto | **12** |
| 2 | Moderado | 2 | Mayor | 3 | Moderado | **6** |
| 3 | Moderado | 2 | Mayor | 3 | Moderado | **6** |
| 4 | Moderado | 2 | Mayor | 3 | Moderado | **6** |
| 5 | Improbable | 1 | Moderado | 2 | Insignificante | **2** |
| 6 | Moderado | 2 | Catastrófico | 4 | Alto | **8** |
| 7 | Improbable | 1 | Moderado | 2 | Insignificante | **2** |
| 8 | Moderado | 2 | Catastrófico | 4 | Alto | **8** |

Tabla 20: Matriz de evaluación de riesgos

### 4.5.3 Plan de contingencia.

Se puede observar que los riesgos de mayor severidad son los riesgos 1, 6 y 8, por lo que se determinará un plan de contingencia para cada uno de ellos

P.C.R.1: Se revisará el código periódicamente y se evaluarán las herramientas actualizadas con las existentes. De esta manera, si existen conflictos, se los resolverán modificando ya sea las versiones de las herramientas existentes o incorporar otra versión de la herramienta actualizada. Al final definir las versiones con las que el prototipo se desarrollará.

P.C.R.6: Se realizarán copias de seguridad del código fuente en forma periódica, y además la información deberá estar almacenada no sólo en el equipo de desarrollo, sino también en algún dispositivo de almacenamiento portable.

P.C.R.8: Se deberá efectuar un control de cumplimiento de objetivos cada vez que finalice una fase del proyecto, lo cual ya está definido en el cronograma.

## 4.6 Estrategias del proyecto

En esta sección se definirán las estrategias del proyecto en función de las actividades propuestas en el cronograma presentado al inicio de esta fase, por lo cual se dividirán las estrategias en tres partes. La primera será plantear la estrategia de análisis y diseño del prototipo, la segunda la estrategia de codificación y pruebas, y la tercera la estrategia de implementación.

**Estrategia de análisis y diseño.**

Para realizar el análisis y diseño del prototipo se utilizará un enfoque orientado a objetos, con el uso de UML para modelar este sistema.

Como recurso principal se utilizará el libro titulado UML y Patrones [Larman, 2002], como referencia para trabajar, y además el software Visual Paradigm como herramienta para modelar dicho prototipo.

El tiempo de desarrollo de esta actividad será de una semana.

**Estrategia de codificación y pruebas.**

Para codificar el prototipo se utilizarán las herramientas y recursos definidos en la Solución Propuesta.

Para realizar las pruebas se realizará un plan de pruebas luego del análisis y diseño del prototipo, el cuál contemplará pruebas unitarias, funcionales y de sistema, y los métodos y herramientas necesarias para realizarlas.

Se seguirá un modelo formal de ejecución de las pruebas definido en la sección posterior.

**Estrategia de implementación.**

Si bien no se trata de un proyecto en el que la implementación sea una actividad indispensable (ya que sólo debería funcionar en el marco en el que fue definido), hipotéticamente se plantea un caso que consiste en que el prototipo esté instalado en una máquina servidor con las características ya planteadas.

En esta actividad también se capacitará a la persona que utilice dicho prototipo en el departamento que corresponda para que entienda y aplique el mismo como una solución o contribución a sus tareas laborales.

## 4.7 Sintaxis de consultas en Lucene

En esta sección se explicará brevemente la sintaxis de consultas para la librería de Lucene lo cual servirá en la siguiente sección para entender cómo se armaron las consultas de búsqueda en este prototipo. Una consulta en Lucene se compone principalmente de términos y operadores. Los términos pueden ser palabras o frases. Las palabras pueden ser por ejemplo: hola, sol, luna, etc. Las frases son una concatenación de palabras, por ejemplo: “hola Juan”, “mañana es lunes”, etc. Estas últimas son representadas entre comillas dobles.

Los operadores representan operaciones de conjuntos entre palabras tales como unión (OR), intersección (AND) y diferencia (NOT). Existen además 2 caracteres comodines que se utilizan como herramientas alternativas dentro de una palabra y/o frase los cuales son:

"?": Representa un carácter cualquiera.

"*": Representa uno o varios caracteres cuales quiera.

Es importante aclarar que Lucene busca en campos. Si en la consulta el campo no está definido la búsqueda se realizara en el campo predeterminado (normalmente "content").

En el caso de este prototipo se definieron tantos campos como anotadores mostrados en la Figura 1.

Un ejemplo de una consulta en la que se desea encontrar todos aquellos documentos que contengan el apellido "Acosta" y hayan sido escritos en el año 2007, sería:

(apellido:"acosta" AND anioResol:"2007")

## 4.8 Funcionamiento del buscador

El proceso de realización de una búsqueda se divide en 2 partes generales: ingreso de parámetros de búsqueda y visualización de resultados.

En cuanto al ingreso de los parámetros de búsqueda se presentan alternativas de búsqueda sobre los índices de Lucene generados en el proyecto de Minería de textos para la categorización de documentos. Las búsquedas sobre los índices pueden ser realizadas por token, por anotación o por categorías. Las búsquedas por token son búsquedas por coincidencia de caracteres, es decir que se trata de búsquedas sintácticas. Las búsquedas por anotación son aquellas búsquedas conceptuales o semánticas sobre los documentos de índices generados en los procesos de minería de textos donde una anotación corresponde a la identificación de una entidad dentro de un documento, por ejemplo: apellido, unidad académica, facultad, año, etc. Las búsquedas por categoría son búsquedas realizadas en una clasificación de documentos también efectuada en los procesos de minería de textos. A continuación se definen las alternativas de búsqueda del prototipo y se especifica el tipo de búsqueda al que corresponde cada alternativa.

1. Con la frase exacta (por tokens)
2. Con algunas de las palabras (por tokens)
3. Con todas las palabras (por tokens)
4. Sin ninguna de las palabras (por tokens)
5. Nombre (por anotación)

6. Apellido (por anotación)
7. Institución (por anotación)
8. Unidad académica (por anotación)
9. DNI (por anotación)
10. Año de resolución (por anotación)
11. Carrera (por anotación)
12. Categoría (por categoría)
    a. Designación de autoridades
    b. Designación de docentes – Extraordinario
    c. Designación de docentes – Planta docente
    d. Otras designaciones – Otras actividades
    e. Designación de representantes
    f. Designación honoraria
    g. Licencia o renuncia docente o autoridad
    h. Auspicio/aval/declaración de interés académico
    i. Convenio-Colaboración
    j. Convenio- Pasantía
    k. Convenio – PPS
    l. Curso/jornada/seminario/taller
    m. Carrera nueva – Plan de estudios
    n. Carrera modificación
    o. Llamado a concurso, designación de jurado
    p. Dictamen concurso
    q. Proyecto de investigación
    r. Juramento privado
    s. Reglamentos

t. Designación de Tribunal Evaluador

u. Becas – Reducción arancelaria

v. Otra categoría.

Cabe aclarar que es posible realizar una búsqueda con más de un criterio relacionándolos mediante operaciones lógicas (AND,OR y NOT). A continuación se presenta la interfaz de búsqueda que permite el ingreso de estos criterios:

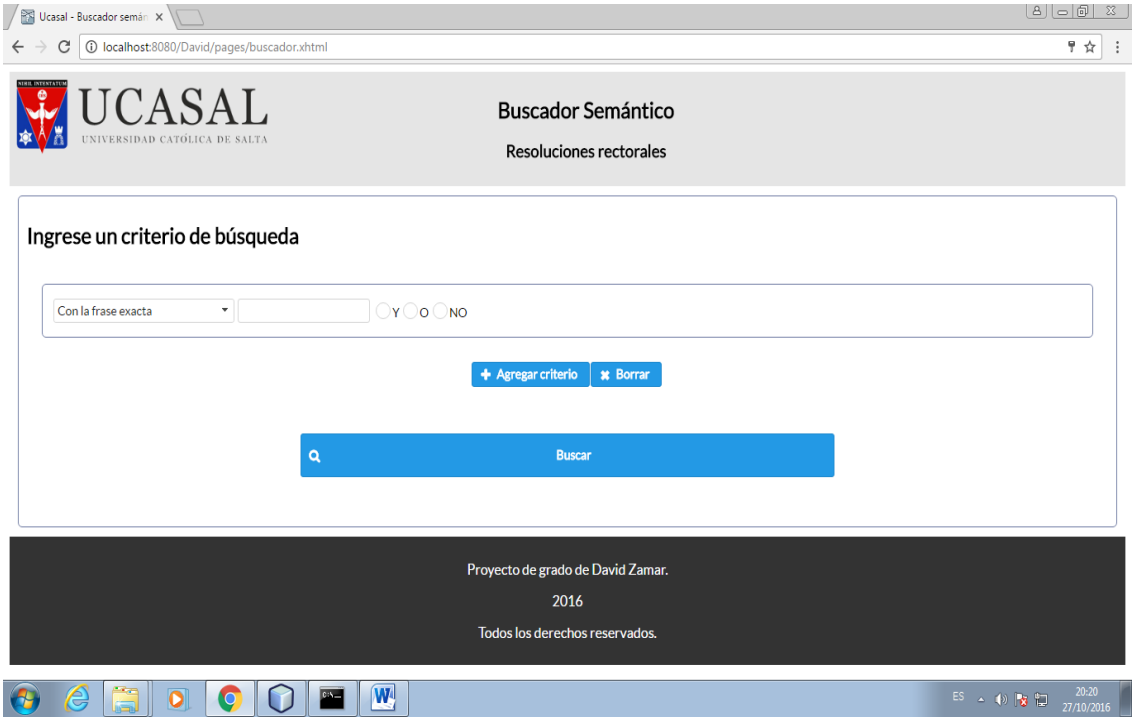



Figura 12: Captura de pantalla – Interfaz de búsqueda

Se realizará una búsqueda de ejemplo donde se ingresarán 3 criterios:

- Apellido: Perez
- Y (AND)
- Año: 2007
- Y (AND)
- Unidad académica: Ingenieria.

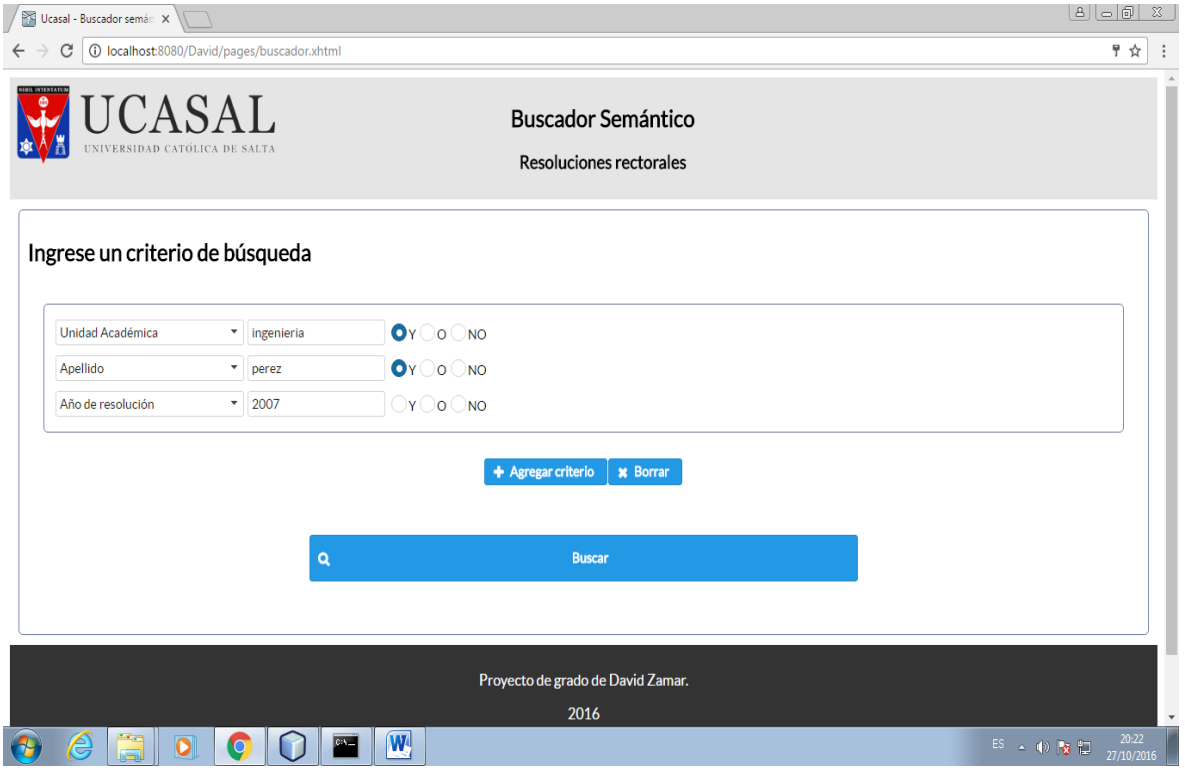



Figura 13: Captura de pantalla – Ingreso de parámetros de búsqueda – Ejemplo 1

Una vez ingresados los parámetros correctamente, el sistema se encargará de generar las consultas reconocidas por Lucene. El código para generar estas consultas está definido en el bean "Consulta.java":

```
//Clase para generar nombres y consultas necesarios para usar lucene y generar las búsquedas

public class Consulta {

  //Función para crear una línea de consulta de 1 criterio.
  public String CrearLineaConsulta(String criteria , String value ){
      int cbo;
      if (criteria=="" || value==""){return "";}
        else
          cbo = Integer.parseInt(criteria);
        {
          switch (cbo){

        //con la frase exacta
          case 1: {String linea= new String("content:\"" + value + "\"");
                  return linea;}

        //con alguna /s  de las palabras
          case 2: { String linea= new String("content:" + value);
            return linea;}

        //con todas las palabras
          case 3: {

                String linea= new String ("");
```

```
            String [] arrayLinea = value.split(" ");

            //Array de palabras AND
            for (int i = 0; i < arrayLinea.length; i++) {

                if (i < arrayLinea.length - 1){
                 linea=linea + "content:" + arrayLinea[i] + " AND ";}
                else
                {
                   linea=linea + "content:" + arrayLinea[i];
                }
            } //Fin del for

return linea;

        }//Fin del caso 3


      //  Se necesita de un valor antes para usar el NOT. Por ej content:ingenieria -content:zamar

        case 4:{
           return null;
        }

      //Por nombre
        case 5:{
           String linea= new String("nombre:\"" +value + "\"");
             return linea;}

      //Por apellido
        case 6:{
           String linea= new String("apellido:\"" +value + "\"");
           return linea;}

      //Por institucion
        case 7:{
           String linea= new String("institucion:\"" +value + "\"");
           return linea;}

      //Por unidad academica
        case 8:{
           String linea= new String("UA:\"" +value + "\"");
           return linea;}

      //Por DNI
        case 9:{
           String linea= new String("dni:\"" +value+ "\"");
           return linea;}

      //Por año de resolucion
        case 10: {
           String linea= new String("anioResol:\"" +value+ "\"");
           return linea;}

      //Por numero de resolucion
        case 11: {
```

```
                Boolean e=true;
                String valor=new String (value);
                    //.substring(0, value.length()-3)); En caso de que el ingreso sea con .../NN

                while (e){
                  if (valor.startsWith("0")){

                    valor=valor.substring(1, valor.length());
                    System.out.println(valor);

                  }
                  else e=false;
                }
                String linea= new String("numResol:\"" +valor + "\"");
                System.out.println(linea);
                return linea;}

            //Por Carrera
              case 12:{
                String linea= new String("Carrera:\"" +value + "\"");
                return linea;}

            //Por categoria
              case 13:{
                String linea= new String("categoria:\"" +value + "\"");
                return linea;}

            } //Fin del Switch
          }//Fin del else
          return null;
}

    //Función para generar la consulta de lucene para todos los criterios dispuestos en una lista con la estructura <criterio,valor>

    public String generarConsulta(List<BuscadorBean> list) {
      Boolean flag=true;
      String lineActual=new String("");
      String lineAnterior=new String("");
      for(BuscadorBean b:list){
        if (b.getUserCriteria().equals("13")){
          flag=false;
          return "";
        }
       }
      if (flag){

      //Variables para guardar valores anteriores
      String cond=new String("");
      int cont=0;
      //Crea la consulta final
       for(BuscadorBean b:list){
       cont++;
       if (cont!=1){
        lineAnterior=lineActual;
        lineActual=this.CrearLineaConsulta(b.getUserCriteria(), b.getUserValue());
        lineActual="(" + lineAnterior + " " + cond + " " + lineActual + ")";
```

```
        cond=b.getUserCondicion();
      }
      else
      {
        lineActual=this.CrearLineaConsulta(b.getUserCriteria(),b.getUserValue());
        cond=b.getUserCondicion();
      }
     }
    }
     return lineActual;
  }
```

Una vez generada la consulta de Lucene, se realiza la búsqueda sobre los índices correspondientes. Esto da como resultado un conjunto de nombres de archivos que cumplen con los criterios de búsqueda ingresados. Para mostrar los resultados se utilizan los archivos XMI que son referenciados por los nombres encontrados en el proceso anterior. Esto es porque los campos que fueron indexados con Lucene de las resoluciones rectorales no se encuentran almacenados para reducir espacio de almacenamiento (ver sección 2.1.2). De los archivos XMI se extraen los valores de los siguientes campos para ser mostrados en la pantalla de Resultados:

- Número de resolución
- Categoría
- Contenido
- Fecha de resolución.

Para la lectura y extracción de contenido de los archivos XMI se utilizó una API denominada DOM (Document Object Model) que brinda las herramientas necesarias para parsear archivos estructurados tales como los XMI. A modo de ejemplo se presenta un fragmento de un archivo XMI utilizado en este prototipo para comprender su estructura y mostrar los anotadores generados en las etapas anteriores de este proyecto:

```
<?xml version="1.0" encoding="UTF-8"?>
...
<cas:Sofa xmi:id="1" sofaNum="2" sofaID="encabezado" mimeType="text" sofaString="RESOLUCION  N
001/07&#13;&#10;&#13;&#10;&#13;&#10;En el Campo Castanares, sito en la Ciudad de Salta, Capital de la
Provincia del mismo nombre, Republica Argentina, sede de la Universidad Catolica de Salta, a un dia del mes de
febrero  del ano dos mil siete:"/><cas:Sofa xmi:id="13" sofaNum="1" sofaID="_InitialView"
mimeType="text" sofaString="la presentacion efectuada por las autoridades de la SECRETARIA DE
POSTGRADO Y PERFECCIONAMIENTO DOCENTE; y &#13;&#10;&#13;&#10;CONSIDERANDO:&#9;que se trata
de la presentacion del Tema de Tesis:  LAS COMPETENCIAS PROFESIONALES EN INGENIERIA-
IMPLEMENTACION DE UN MODELO CURRICULAR BASADA EN COMPETENCIAS EN LA CARRERA DE
INGENIERIA CIVIL, perteneciente al alumno, NESTOR EUGENIO LESSER; Directora: DRA. MARIA CELIA
ILVENTO; &#13;&#10;que es necesario dictar el instrumento legal que apruebe el Tema y el
Director;&#13;&#10;                    por todo ello;&#13;&#10;&#13;&#10;&#13;&#10;&#13;&#10;EL RECTOR  DE LA
UNIVERSIDAD CATOLICA DE SALTA&#13;&#10;&#13;&#10;RESUELVE&#13;&#10;&#13;&#10;Art.1.-
&#9;APROBAR el TEMA de TESIS: LAS COMPETENCIAS PROFESIONALES EN INGENIERIA- IMPLEMENTACION
DE UN MODELO CURRICULAR BASADA EN COMPETENCIAS EN LA CARRERA DE INGENIERIA CIVIL,
perteneciente al alumno, NESTOR EUGENIO LESSER, de la Maestria en Educacion.&#13;&#10;Art.2.-
```

	APROBAR como Directora de Tesis a la DRA. MARIA CELIA ILVENTO."/><tcas:DocumentAnnotation xmi:id="20" sofa="13" begin="0" end="934" language="x-unspecified"/><examples:SourceDocumentInformation xmi:id="25" sofa="13" begin="0" end="0" uri="file:/D:/Resols/Resoluciones%20AÑO%202007/RES.%20%20Nº0001-07.doc" offsetInSource="0" documentSize="23040" lastSegment="false"/><mio:NumeroResol xmi:id="40" sofa="13" begin="0" end="20" nroResol="1/07" numero="1" anio="2007"/><mio:FechaResol xmi:id="55" sofa="13" begin="152" end="233" anio="2007" mes="FEBRERO" dia="1" fechaResolCompleta="UN DE FEBRERO DE DOS MIL SIETE"/>
...
<mio:UA xmi:id="63" sofa="13" begin="52" end="105" confidence="20.0" componentId="de.julielab.jules.lingpipegazetteer.GazetteerAnnotator" specificType="UnidadAcademica"/><mio:Carrera xmi:id="75" sofa="13" begin="767" end="783" confidence="0.0" componentId="de.julielab.jules.lingpipegazetteer.GazetteerAnnotator" specificType="Carrera"/><mio:Carrera xmi:id="87" sofa="13" begin="840" end="861" confidence="0.0" componentId="de.julielab.jules.lingpipegazetteer.GazetteerAnnotator" specificType="Carrera"/><mio:Carrera xmi:id="99" sofa="13" begin="304" end="320" confidence="0.0" componentId="de.julielab.jules.lingpipegazetteer.GazetteerAnnotator" specificType="Carrera"/><mio:Clase xmi:id="111" sofa="13" begin="0" end="0" valor="Otra"/><mio:Nombre xmi:id="116" sofa="13" begin="347" end="353" confidence="1.0" componentId="de.julielab.jules.lingpipegazetteer.GazetteerAnnotator"
....
<mio:NumeroResol xmi:id="33" sofa="1" begin="0" end="20" nroResol="1/07" numero="1" anio="2007"/><mio:FechaResol xmi:id="47" sofa="1" begin="152" end="233" anio="2007" mes="FEBRERO" dia="1" fechaResolCompleta="UN DE FEBRERO DE DOS MIL SIETE"/><cas:View sofa="13" members="20 25 40 55 63 75 87 99 111 116 128 140 152 164 176 188 200 212 224 236 248 260 272 281 290 299 308 312 316 320 324 328 332 336 340 344 348 352 356 360 364 368 372 376 380 384 388 392 396 400 404 408 412 416 420 424 428 432 436 440 444 448 452 456 460 464 468 472 476 480 484 488 492 496 500 504 508 512 516 520 524 528 532 536 540 544 548 552 556 560 564 568 572 576 580 584 588 592 596 600 604 608 612 616 620 624 628 632 636 640 644 648 652 656 660 664 668 672 676 680 684 688 692 696 700 704 708 712 716 720 724 728 732 736 740 744 748 752 756 760 764 768 772 776 780 784 788 792 796 800 804 808 812 816 820 824 828 832 836 840 844 848 852 856 860 864 868 872 876 880 884 888 892 896 900 904 908 912 916 920 924 928 932 936 940 944"/><cas:View sofa="1" members="8 33 47"/></xmi:XMI>

En el anexo 3 se transcribe el código desarrollado para recorrer, extraer y presentar la información de los campos de los archivos XMI utilizados en este prototipo.

En el caso de la consulta definida al principio de la sección, donde se pretendía encontrar aquellas resoluciones del año 2007, de la unidad académica de Ingeniería y donde aparezca el apellido Perez, resultó coincidente una sola resolución que se muestra en la figura siguiente:

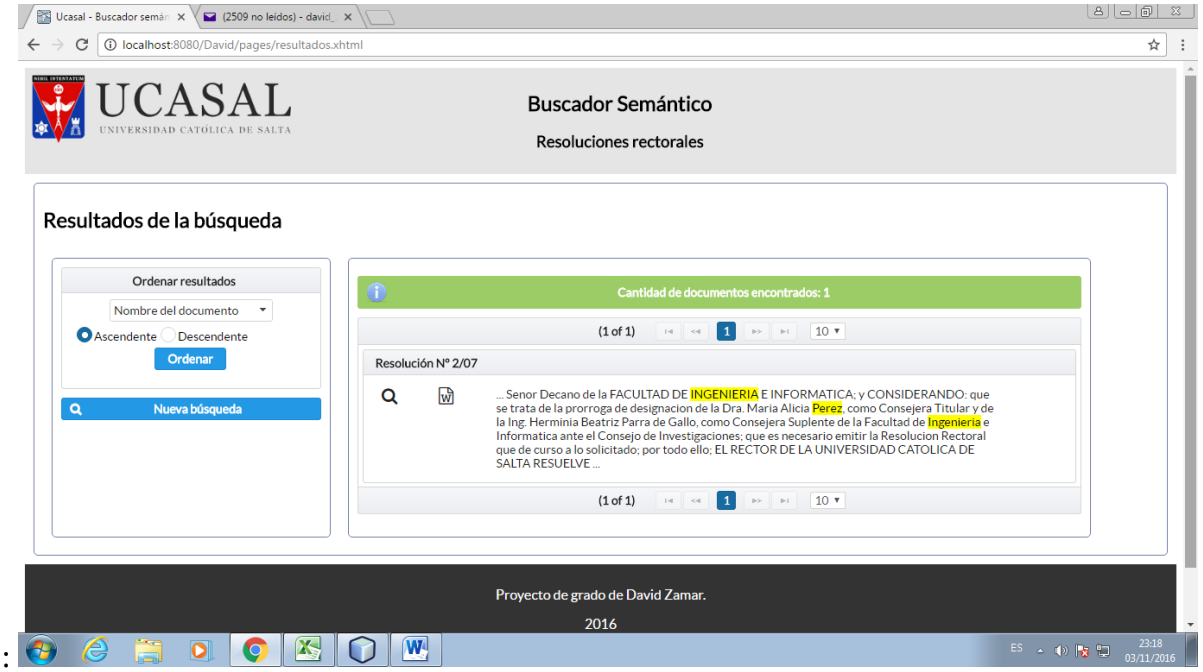



Figura 14: Captura de pantalla – Resultados – Ejemplo 1

Por otro lado se realizó otra consulta para poder visualizar mejor la paginación de los resultados encontrados que corresponde a aquellas resoluciones donde figure el apellido Perez. A continuación se muestran los resultados paginados y la explicación de las partes de la interfaz de resultados:

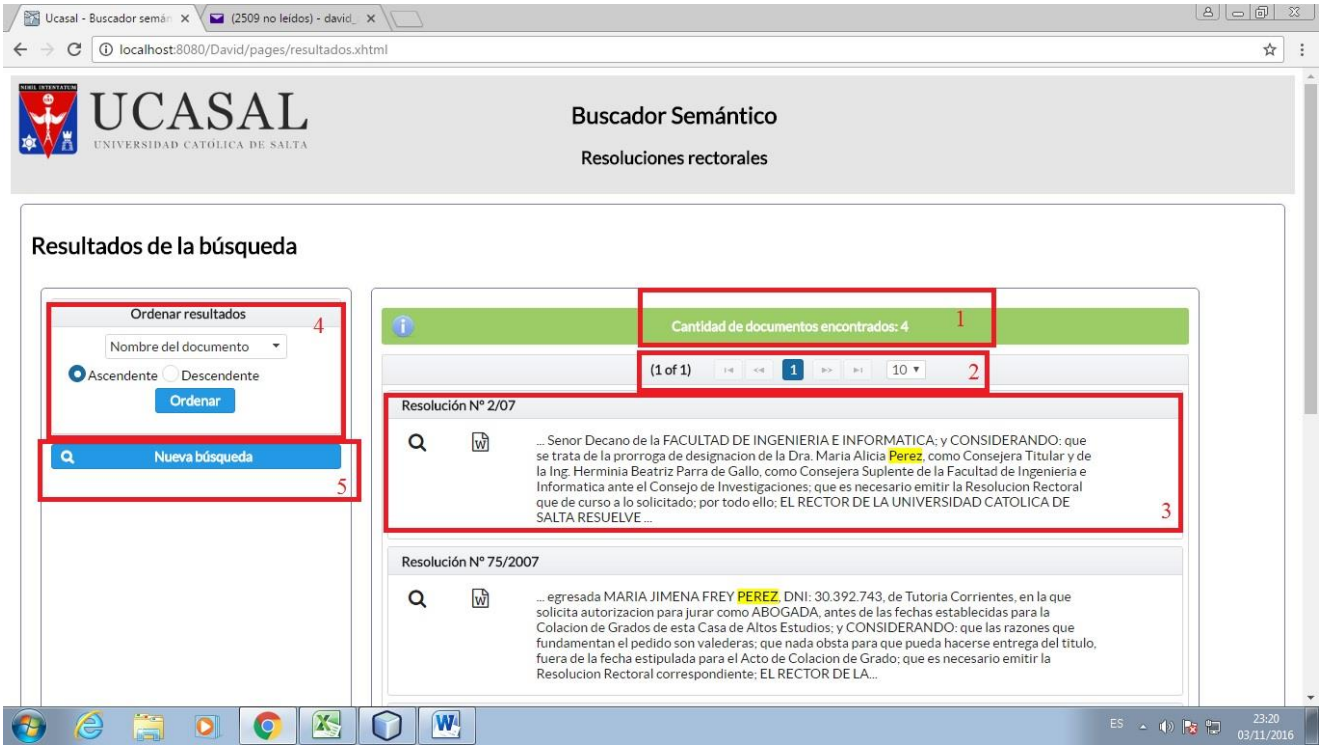



Figura 15: Captura de pantalla - Resultados

En la captura de pantalla se puede identificar lo siguiente:

1- Cantidad de documentos que fueron encontrados según el ingreso de esos criterios.

2- Paginación de resultados

3- Resultado, con las alternativas de ver el contenido del archivo XMI y descargar la resolución en Word. Además se muestran algunas líneas del contenido de la resolución.

4- Se puede seleccionar el ordenamiento ascendente o descendente de los resultados: Por fecha, número de resolución y por nombre de la resolución.

5- Botón para regresar a la pantalla principal de búsqueda para realizar una nueva búsqueda.

Para finalizar se mostrará otro ejemplo con la intención de mostrarle al lector que el buscador realiza concretamente una búsqueda semántica sobre las resoluciones rectorales. En el ejemplo se consultarán todas aquellas resoluciones que sean del año 2007, unidad académica: ingeniería y categoría: designación de autoridades – docentes.

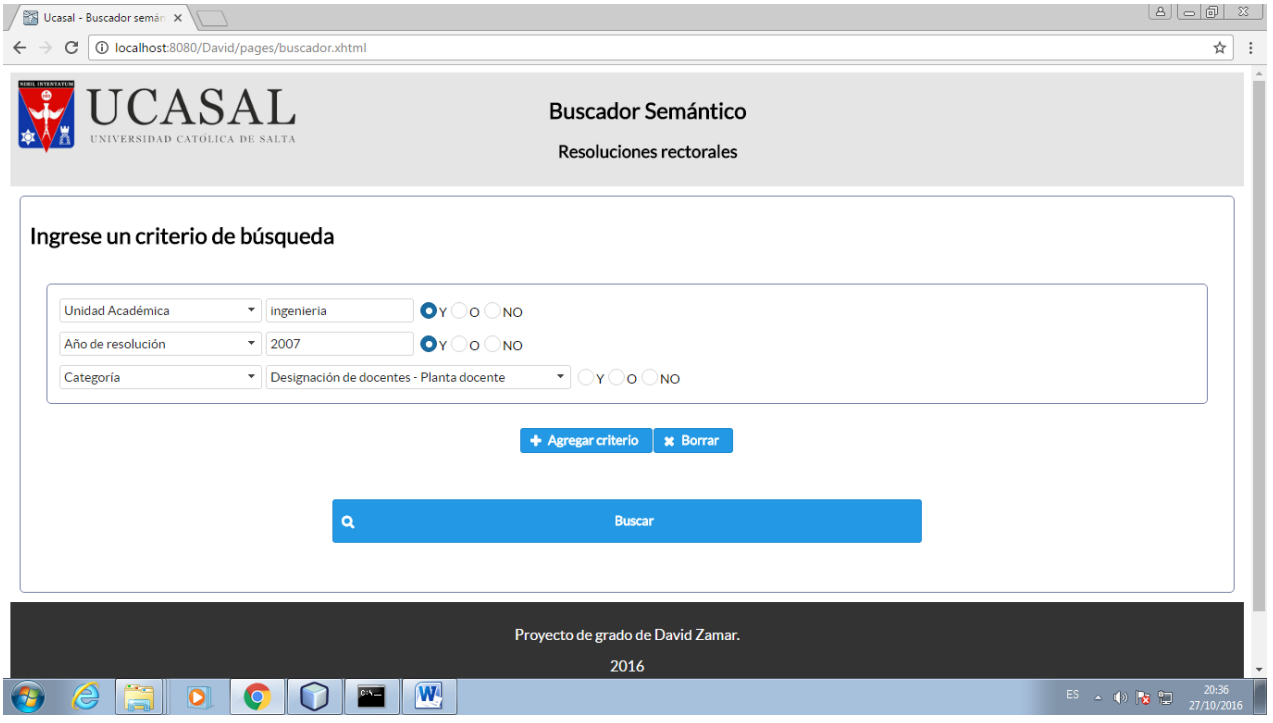



Figura 16: Captura de pantalla – Ingreso de parámetros de búsqueda – Ejemplo 2

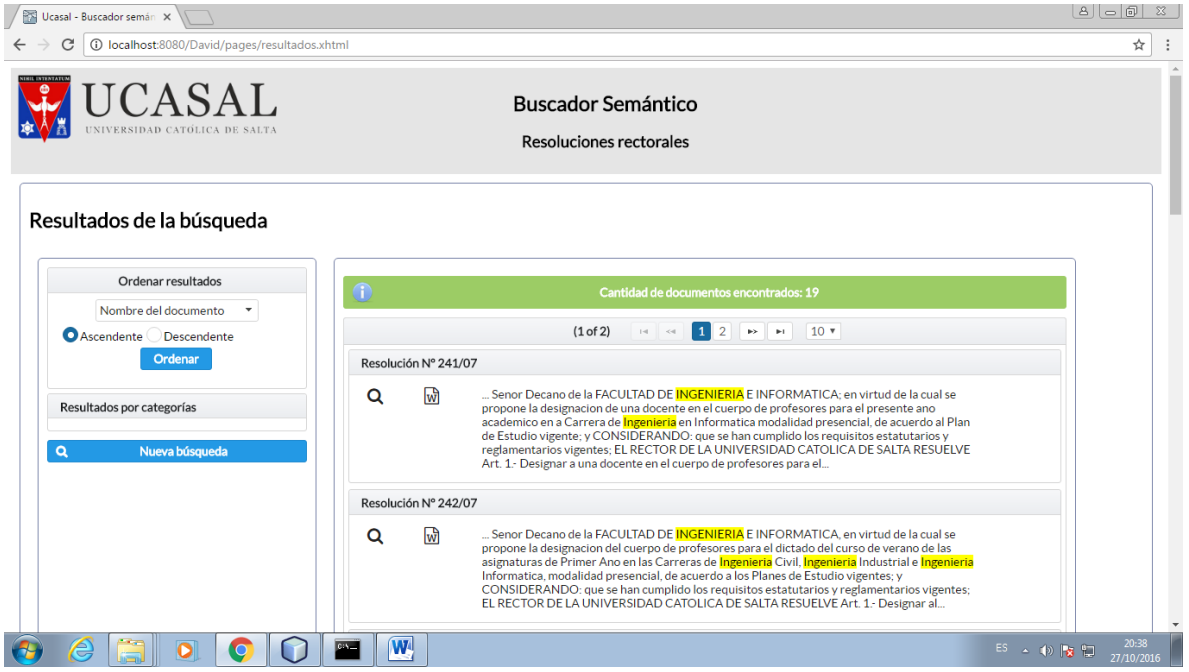



Figura 17: Captura de pantalla – Resultados – Ejemplo

# Capítulo 5 – Resultados

A continuación se presentarán las pruebas realizadas al prototipo desarrollado que verifican el correcto funcionamiento del mismo. Las pruebas son identificadas en una matriz de trazabilidad para contar con un historial versionado de las pruebas realizadas y los resultados obtenidos siguiendo además una metodología establecida para desarrollarlas. Esta matriz generaliza y define los casos de prueba que se realizarán. Los casos de prueba pueden o no responder al requerimiento especificado en la tabla y los requerimientos a probar son definidos por el Analista Programador en la función de Testing. Se tratan de pruebas de sistema y son de caja negra, es decir, que no se realiza una depuración del código fuente o un análisis de camino crítico del código para ejecutarlas. Son definidas según la complejidad y criticidad que el analista programador considera evaluar.

Trazabilidad:

| No. | Requerimiento | Fuente del Requerimiento | Módulo del Sistema | Caso Prueba |
|---|---|---|---|---|
| 1 | Selección de parámetros de Búsqueda | Diseño de Buscador Semántico | BUSCADOR | CP1 |
| 2 | Búsqueda según parámetros seleccionados(relación entre archivos XMI e índices de Lucene | Diseño de Buscador Semántico | BUSCADOR | CP2 |
| 3 | Mostrar y descargar resultado seleccionado | Diseño de Buscador Semántico | BUSCADOR | CP3 |
| 4 | Mostrar resultados encontrados por página, con contenido y nombre. | Diseño de Buscador Semántico | BUSCADOR | CP4 |
| 5 | Ordenar resultados encontrados | Diseño de Buscador Semántico | BUSCADOR | CP5 |

Tabla 21: Matriz de trazabilidad de pruebas

| Solicitud / Proyecto: | Buscador Semántico | Fecha: | 06/11/2015 | Hora: | 10:00 |
|---|---|---|---|---|---|
| Lugar: | Oficina de investigación | Duración: | | 2 min | |
| Caso Prueba: | CP1 | Versión: | 1 | Iteración | 1 |

| *Participantes* | | | |
|---|---|---|---|
| **Nombre** | **Área u Organismo** | **Datos complementarios** | **Firma** |
| *Analista programador* | *Testing* | | |

| | |
|---|---|
| *Aplicación / Modulo:* | *Buscador Semántico* |
| *Pre-requisitos:* | *Contar con una idea para realizar una búsqueda* |
| *Scripts y herramientas:* | *No existen* |

| *Secuencia de la prueba* | | | |
|---|---|---|---|
| *No.* | *Paso a Ejecutar* | *Resultado Esperado* | *Resultado Obtenido* |
| *1* | *Loguearse en la aplicación* | *Acceso concedido* | *Acceso concedido* |
| *2* | *Seleccionar Primer criterio y agregar primer valor* | *Comportamiento de caja de texto correcto, de acuerdo al tipo de dato correspondiente al tipo de criterio.* | *Comportamiento de caja de texto correcto, de acuerdo al tipo de dato correspondiente al tipo de criterio.* |
| *3* | *Seleccionar "Agregar Criterio"* | *Se crea una nueva entrada de criterio* | *Se crea una nueva entrada de criterio* |
| *4* | *Seleccionar "Agregar Criterio" con criterio anterior vacío* | *Aparece una ventana informando que se debe ingresar un valor de criterio.* | *Aparece una ventana informando que se debe ingresar un valor de criterio.* |

| *Dictamen:* | *A* | *Evaluación de la Operación:* | *J* |
|---|---|---|---|
| *A=Aprobada R=Rechazada B=Bloqueador C=Critico M=Mayor N=Menor T=Trivial J=Mejorable E=Estético* | | | |

| Solicitud / Proyecto: | Buscador Semántico | Fecha: | 06/11/2015 | Hora: | 10:05 |
|---|---|---|---|---|---|
| Lugar: | Oficina de investigación | Duración: | | 5 min | |
| Caso Prueba: | CP2 | Versión: | 1 | Iteración | 1 |

| *Participantes* | | | |
|---|---|---|---|
| **Nombre** | **Área u Organismo** | **Datos complementarios** | **Firma** |
| *Analista programador* | *Testing* | | |

| | |
|---|---|
| *Aplicación / Modulo:* | *Buscador Semántico* |
| *Pre-requisitos:* | *Contar con una idea para realizar una búsqueda* |
| *Scripts y herramientas:* | *No existen* |

| *Secuencia de la prueba* | | | |
|---|---|---|---|
| *No.* | *Paso a Ejecutar* | *Resultado Esperado* | *Resultado Obtenido* |
| *1* | *Loguearse en la aplicación* | *Acceso concedido* | *Acceso concedido* |
| *2* | *Seleccionar Primer criterio y agregar primer valor* | *Comportamiento de caja de texto correcto, de acuerdo al tipo de dato correspondiente al tipo de criterio.* | *Comportamiento de caja de texto correcto, de acuerdo al tipo de dato correspondiente al tipo de criterio.* |
| *3* | *Seleccionar "Buscar"* | *Aparece pantalla de Resultados con los resultados encontrados y verificar que tengan coherencia con los parámetros ingresados y que exista coherencia entre el conjunto de archivos XMI y los índices de Lucene.* | *Concuerda con el esperado.* |

| *Dictamen:* | *A* | *Evaluación de la Operación:* | *J* |
|---|---|---|---|
| *A=Aprobada R=Rechazada B=Bloqueador C=Critico M=Mayor N=Menor T=Trivial J=Mejorable E=Estético* | | | |

| Solicitud / Proyecto: | Buscador Semántico | Fecha: | 06/11/2015 | Hora: | 10:10 |
|---|---|---|---|---|---|
| Lugar: | Oficina de investigación | Duración: | | 5 min | |
| Caso Prueba: | CP3 | Versión: | 1 | Iteración | 1 |

| *Participantes* | | | |
|---|---|---|---|
| **Nombre** | **Área u Organismo** | **Datos complementarios** | **Firma** |
| *Analista programador* | *Testing* | | |

| | |
|---|---|
| *Aplicación / Modulo:* | *Buscador Semántico* |
| *Pre-requisitos:* | *Haber realizado una búsqueda exitosa y contar con resultados.* |
| *Scripts y herramientas:* | *No existen* |

| *Secuencia de la prueba* | | | |
|---|---|---|---|
| *No.* | *Paso a Ejecutar* | *Resultado Esperado* | *Resultado Obtenido* |
| *1* | *Verificar la pantalla de resultados* | *Resultados clasificados por página, mostrando una parte del contenido de cada uno (no más de 2 renglones), el nombre y las opciones de descarga y visualización online.* | *Coherente con lo esperado.* |
| *2* | *Seleccionar botón de ver en línea de un resultado.* | *Se abre una nueva página con el contenido del archivo.* | *Coherente con lo esperado.* |
| *3* | *Cerrar la última página y seleccionar la opción de descargar el documento en un formato determinado (.doc)* | *Se abre asistente para guardar el archivo en el formato seleccionado.* | *El botón no posee funcionalidad.* |

| *Dictamen:* | *R* | *Evaluación de la Operación:* | *C* |
|---|---|---|---|
| *A=Aprobada R=Rechazada B=Bloqueador C=Critico M=Mayor N=Menor T=Trivial J=Mejorable E=Estético* | | | |

| Solicitud / Proyecto: | Buscador Semántico | Fecha: | 06/11/2015 | Hora: | 10:15 |
|---|---|---|---|---|---|
| Lugar: | Oficina de investigación | Duración: | | 5 min | |
| Caso Prueba: | CP4 | Versión: | 1 | Iteración | 1 |

| *Participantes* | | | |
|---|---|---|---|
| **Nombre** | **Área u Organismo** | **Datos complementarios** | **Firma** |
| *Analista programador* | *Testing* | | |

| | |
|---|---|
| *Aplicación / Modulo:* | *Buscador Semántico* |
| *Pre-requisitos:* | *Haber realizado una búsqueda exitosa y contar con resultados.* |
| *Scripts y herramientas:* | *No existen* |

| *Secuencia de la prueba* | | | |
|---|---|---|---|
| *No.* | *Paso a Ejecutar* | *Resultado Esperado* | *Resultado Obtenido* |
| *1* | *Verificar que por cada resultado se muestren: 2 renglones de contenido; el nombre; opciones de descarga y visualización.* | *2 renglones de contenido; el nombre; opciones de descarga y visualización en la ventana de resultados por cada archivo* | *Coherente con lo esperado.* |

| *Dictamen:* | *A* | *Evaluación de la Operación:* | *J* |
|---|---|---|---|
| *A=Aprobada R=Rechazada B=Bloqueador C=Critico M=Mayor N=Menor T=Trivial J=Mejorable E=Estético* | | | |

| Solicitud / Proyecto: | Buscador Semántico | Fecha: | 06/11/2015 | Hora: | 10:20 |
|---|---|---|---|---|---|
| Lugar: | Oficina de investigación | Duración: | | 5 min | |
| Caso Prueba: | CP5 | Versión: | 1 | Iteración | 1 |

| *Participantes* | | | |
|---|---|---|---|
| **Nombre** | **Área u Organismo** | **Datos complementarios** | **Firma** |
| *Analista programador* | *Testing* | | |

| | |
|---|---|
| *Aplicación / Modulo:* | *Buscador Semántico* |
| *Pre-requisitos:* | *Haber realizado una búsqueda exitosa y contar con resultados.* |
| *Scripts y herramientas:* | *No existen* |

| *Secuencia de la prueba* | | | |
|---|---|---|---|
| *No.* | *Paso a Ejecutar* | *Resultado Esperado* | *Resultado Obtenido* |
| *1* | *Seleccionar el criterio de ordenamiento y hacer clic sobre el botón Ordenar.* | *Resultados paginados y ordenados mostrados en la pantalla de Resultados.* | *Coherente con lo esperado.* |

| *Dictamen:* | *A* | *Evaluación de la Operación:* | *J* |
|---|---|---|---|
| *A=Aprobada R=Rechazada B=Bloqueador C=Critico M=Mayor N=Menor T=Trivial J=Mejorable E=Estético* | | | |

# Capítulo 6 – Conclusiones

Al inicio de este proyecto se estudiaron herramientas aisladas y con características específicas que sirvieron de base para comprender y conocer procesos de extracción de texto, recorrido de archivos y ficheros, análisis y recorrido de palabras (parsers) y la forma de indexar contenido para luego poder realizar búsquedas. Lucene resultó ser una potente herramienta que permite realizar la mayoría de estas tareas y es el componente principal del desarrollo de este prototipo. Las tareas realizadas para el desarrollo de este proyecto en forma detallada fueron las siguientes:

- Estudio e investigación de herramientas, librerías y otros proyectos relacionados al procesamiento de textos, minería de textos, usabilidad en buscadores, procesamiento de archivos XMI y buscadores en general.
- Adecuación e integración de librerías y herramientas para que el prototipo desarrollado funcione sin inconvenientes relacionados a las versiones de los componentes que lo conforman, como ser servidor de aplicaciones, librería de Lucene, librerías de JSF, framework Primefaces, librerías de extracción de textos de distintos tipos de archivos (PDF,TXT,DOC), librerías de procesamiento de archivos XMI y JAVA.
- Asesoramiento, reuniones y validaciones con profesionales especialistas en distintas áreas que cubrieron y contribuyeron al avance del desarrollo de este proyecto.
- Exploración e investigación acerca de los conceptos de usabilidad en los buscadores para adoptar especificaciones de diseño que contribuyeron a construir un prototipo fácil de usar lo cual era uno de los objetivos de este proyecto.

Por otra parte el proyecto realizado promueve la idea de que la búsqueda semántica es un paradigma que se está incorporando de a poco en la web y en los sistemas informáticos en general. El motivo por el cual todavía los buscadores tradicionales y populares todavía no integran este tipo de búsqueda es porque resulta necesario que la información indexada que se encuentra en sus servidores debe estar estructurada y organizada. Para ello debe someterse a diversos procesos de análisis y aprendizaje para brindar resultados que respondan a criterios conceptuales y resulta sumamente costoso y difícil de implementar con la cantidad y diversidad de información almacenada en esos servidores.

Es importante aclarar que el principal beneficiado en una búsqueda semántica es el usuario pero no basta con lograr que pueda encontrar lo que está buscando, sino que debería poder encontrar lo que busca de la manera más sencilla y eficiente posible. Es por ello que la usabilidad debe tenerse en cuenta no sólo en un proyecto de estas características, sino que también debe estar presente en cualquier otro sistema que se desarrolle porque en definitiva la interfaz es lo que atrapa la atención principal del cliente final.

Los objetivos de este proyecto fueron cumplidos. De ahora en adelante el horizonte de trabajo es muy claro: la minería de textos como solución a problemas y necesidades de información. Las instituciones de hoy en día cuentan con muchos documentos de textos que pueden ser analizados y categorizados con herramientas de la minería de textos y es posible, como se dijo antes, brindar información conceptual o semántica en respuesta a las consultas que un usuario o cliente realice. Esto conlleva un beneficio estratégico para cualquier empresa o institución ya que los empleados podrían contar con información estratégica sin tener que estar analizando manualmente todo el conjunto de archivos de textos con el que cuenta.

# Bibliografía

Miguel Angel Alvarez. 2002. Artículo "¿Qué es JSP?"
http://www.desarrolloweb.com/articulos/831.php
Página vigente al 18/02/16

Thomas Hampp & Alexander Lang, 2005. Semantic search in WebSphere Information Integrator OmniFind Edition: The case for semantic search. http://www.ibm.com/developerworks/data/library/techarticle/dm-0508lang/.
Página vigente al 18/02/16.

Erick Hatcher, Otis Gospodnetic & Michael McCandless, 2009. Lucene in action. Segunda edición. 399 páginas. Manning Publications.

ISO. 1993. ISO 9241: Ergonomic requirements for office work with visual display terminals. International Organization for Standardization.

ISO. 1999. ISO 13407: "User centred design process for interactive systems. International Organization for Standardization".

ISO. 2011. ISO/IEC 25010. Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE).

Cristhy Nataly Jiménez Granizo. 2008. Usabilidad en los buscadores semánticos. Tesis de grado. Pontifica Universidad Católica de Valparaíso. 204 páginas.

Craig Larman. 2002. UML y patrones, segunda edición. Pearson Educación. Madrid 2003. ISBN: 84-205-3438-2. 624 páginas.

Jakob Nielsen, 1993. "Usability engineering". Academic Press. San Diego, CA. USA.

Alicia Pérez y Carolina Cardoso. 2011. Categorización automática de documentos. Simposio Argentino de Inteligencia Artificial, 40 Jornadas Argentinas de Informática (JAIIO), Córdoba, ISSN 1850-2776.

Alejandro Perez García. 2006. Artículo "JSF – Java Server Faces".
http://www.desarrolloweb.com/articulos/2380.php
Página vigente al 18/02/16.

Roger S. Pressman, 1991. Ingeniería de Software: Un enfoque práctico. 7ma Edición.

Project Managment Institute.
http://www.pmvalue.com.ar/Home/Que_es_Adm.htm
http://www.pmi.org/Pages/default.aspx

Página vigente al 18/02/16

PMI Standards Committee. 1996. A Guide to the Project Management Body of Knowledge. s.l.: Project Management Institute. 182 págs. ISBN: 1880410125.

Zamar, Esteban David. 2013. Anteproyecto de buscador semántico. 13 páginas.

# Glosario

**Apache POI**: es una librería de código abierto escrita en Java para manipular formatos de archivo de Microsoft. POI corresponde al acrónimo de "Poor Obfuscation Implementation" (Implementación de ofuscación mala o pobre) en relación a que los formatos de los archivos que eran objeto del trabajo del proyecto de Apache POI parecían deliberadamente ofuscados.

**CAS Consumer**: *Common Analysis Structure Consumer*. Consumidor de estructura de análisis común. Estructura con la cual se comunican lo componentes en UIMA.

**Parsear**: Proceso de analizar una secuencia de símbolos a fin de determinar su estructura gramatical.

**Tokenizar**: Proceso de separar una cadena de texto en palabras, frases, símbolos u otros elementos significativos llamados token

**UIMA**: *Unstructured Information Management Architecture*. Arquitectura de manejo de información NO estructurada.

**XML**: eXtensible Markup Language. Lenguaje de Marcas Extensible

**XMI**: *XML Metadata Interchange*. XML de Intercambio de Metadatos

# Anexo 1 – Descripción de metodologías del Proyecto y del Diseño

Tal como se referenció en la sección 4.2 las metodologías a utilizar para el desarrollo de este proyecto y prototipo son las siguientes:

**Metodología de Diseño del prototipo: Ciclo de vida clásico o en cascada.**

Se eligió esta metodología principalmente porque los requisitos a los que debe responder el prototipo son claros y están especificados en los casos de uso desarrollados en la sección 4.4. A continuación se presentará el modelo de la metodología seleccionada en un esquema [Pressman, 1991].

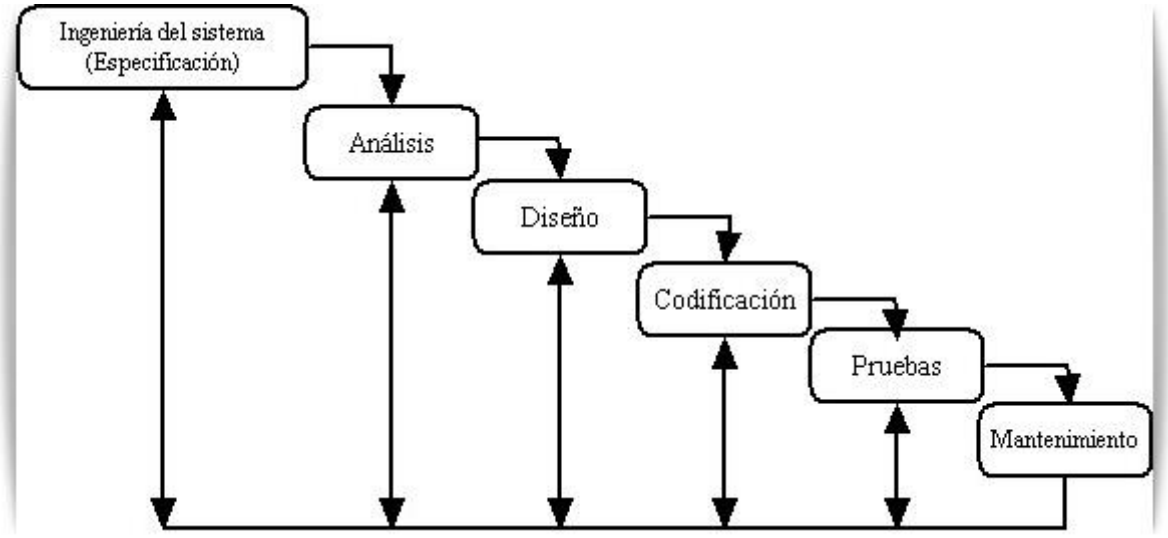



Figura 18: Ciclo de vida del desarrollo de software en cascada

Ingeniería del Sistema (Especificación): Consiste principalmente en la realización de las actividades de relevamiento y adquisición de información lo que permite especificar entre otras cosas el alcance de las funciones del prototipo. Estas actividades fueron descritas y especificadas en el capítulo 2 de este informe (estado de la cuestión) donde se investigaron los temas y herramientas a utilizar que pertenecen al contexto del proyecto en el que se está desarrollando este.

Análisis: Esta actividad consiste en determinar en forma detallada las funciones y características que el prototipo debería tener para responder a los requerimientos especificados en el proceso anterior. El análisis está desarrollado en las secciones 4.4.1 y 4.4.2 de este documento.

Diseño: Es la arquitectura del prototipo a implementar. Se detallará la construcción de las funciones y características definidas en el proceso de Análisis mediante un conjunto de herramientas UML. El diseño está desarrollado en las secciones 4.4.3 y 4.4.4 de este documento.

Codificación: Es la interpretación y escritura de las partes del diseño en un lenguaje y entorno determinados.

Pruebas: Actividades que corresponden a la evaluación y verificación de las funciones y características codificadas en el proceso anterior. Las pruebas están especificadas en el capítulo 5 de este trabajo. Esta actividad se describe en el capítulo 5 de este informe.

Mantenimiento: Consiste en la implementación y mantenimiento del prototipo instalado y ejecutándose en la infraestructura real y final. En el alcance de este proyecto el mantenimiento no es una actividad a considerar.

**Metodología del proyecto: PMI**

Para el desarrollo de este proyecto se utilizaron algunas herramientas que conforman procesos y áreas de conocimiento de una metodología para la gestión de proyectos denominada PMI. Las herramientas y áreas que fueron desarrolladas en el proyecto son referenciadas al final o al principio de cada descripción teórica de cada herramienta, proceso o área.

Los procesos y áreas de conocimiento se encuentran definidos y estandarizados por la institución denominada PMI (Project Management Institute) en la que están determinados cinco procesos y nueve áreas de conocimiento. [PM Book, 1996]

Cada proceso cuenta con un conjunto de actividades definidas que tienen por objetivo asegurar el éxito del proyecto cumpliendo con los plazos estipulados para actividad con los requisitos planteados al inicio del proyecto y con el presupuesto estimado. En el contexto de este proyecto se reflejan aplicadas las actividades principales de esta metodología en el cronograma de trabajo y diagrama de Gantt detallados en la sección 3.2.

Las actividades principales son las siguientes:

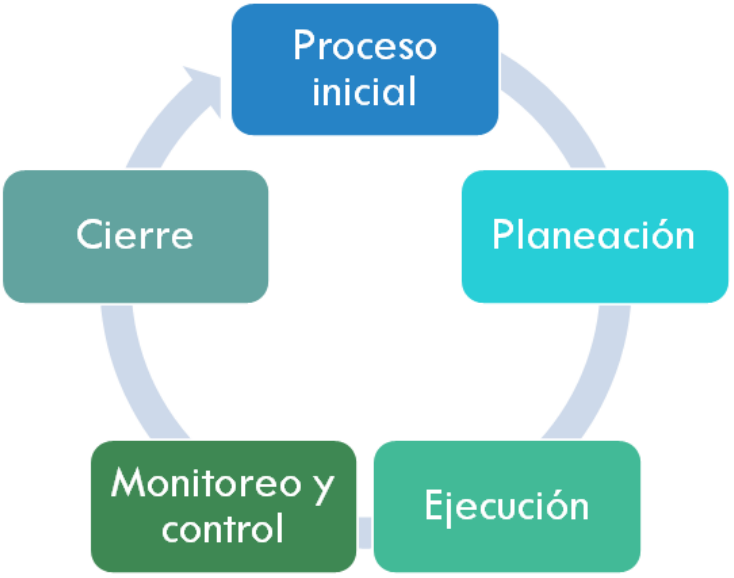



Figura 19: Actividades principales de la gestión de proyectos – PMI

1-Proceso Inicial: Se conforma principalmente por actividades que permiten determinar la factibilidad, dirección y estrategias que debería abordar el proyecto para ser ejecutado.

2-Planeación: Consiste en diseñar un esquema metódico y estructurado que describa el uso de los recursos humanos y materiales del proyecto a lo largo del tiempo.

3-Ejecución: Corresponde a llevar a cabo las tareas definidas en la planeación.

4-Monitoreo y Control: Son las actividades que permiten determinar si existe correspondencia entre el esquema planteado en la Planeación y las actividades realizadas en la Ejecución. Este proceso permite dar valores indicativos de la situación global y detallada de un proyecto en función del nivel de planificación con que cuente.

5-Cierre: Es el proceso o etapa final del proyecto, en el que se verifica que se cumplieron los objetivos, alcance, cronograma y presupuesto del proyecto para concluirlo.

Las áreas de conocimiento o gestiones recomendadas a realizar en un proyecto por la institución del PMI son las siguientes:


Figura 20: Areas de conocimiento - PMI

Gestión de integración: Conjunto de procesos que permiten asegurar la interacción y coordinación de todos los elementos que conforman el proyecto.

Gestión de Alcance: Conjunto de procesos que permiten asegurar y controlar que se cumplan metas y objetivos específicos del proyecto.

Gestión del Tiempo: Conjunto de procesos que permiten establecer un control sobre el inicio, duración y terminación de cada actividad del proyecto. Se ven implementados algunos de ellos en la sección 3.2 de este informe.

Gestión del Costo: Procesos que permiten asegurar que el proyecto cumpla con el presupuesto que tiene asignado. Se ven implementados algunos de ellos en la sección 4.3.1 de este informe.

Gestión de Calidad: Conjunto de procesos que aseguran que el producto y los objetivos del proyecto cumplan ciertos estándares de eficiencia y eficacia.

Gestión de Recursos Humanos: Procesos que permiten organizar y controlar todas las actividades relacionadas al manejo del personal involucrado en el proyecto. Se ven implementados algunos de ellos en la sección 4.1 de este informe.

Gestión de Comunicaciones: Conjunto de procesos que permiten instrumentar las actividades relacionadas a los flujos de información entre las personas involucradas en el proyecto.

Gestión de Riesgos: Conjunto de procesos que permiten analizar y diseñar contingencias ante la ocurrencia de posibles riesgos que afecten el proyecto. Se ven implementados algunos de ellos en la sección 4.5 de este informe.

Gestión de Adquisiciones: Conjunto de procesos que aseguran el correcto y eficiente aprovisionamiento de los recursos necesarios para llevar a cabo el proyecto.

# Anexo 2 – La usabilidad en los buscadores semánticos

La usabilidad es un concepto que se aplica a varios contextos y disciplinas para identificar el grado de facilidad de uso que posee un componente o la ejecución de un proceso. En el contexto de este proyecto se asocia la usabilidad principalmente a las características y procesos relacionados con la interacción hombre-máquina para la búsqueda de información. Se integran entonces disciplinas como la informática, psicología, ciencias sociales y ergonomía que permiten proponer un diseño de interfaz de usuario para este prototipo.

## Conceptos de usabilidad

Existen varias definiciones de usabilidad en función del contexto en que se la analice:

La usabilidad de productos en general se define como [ISO, 1993: Part 11]:

Medida en la que un producto puede ser usado por un grupo de usuarios determinados, para conseguir objetivos específicos con efectividad, eficiencia y satisfacción en un contexto de uso especificado"

Donde:

- **Efectividad:** exactitud con la que usuarios específicos logran objetivos específicos en un ambiente en particular.
- **Eficiencia:** recursos gastados con relación a la exactitud de los objetivos logrados.
- **Satisfacción:** comodidad y aceptabilidad del sistema de trabajo por parte de los usuarios y de las demás personas que se ven afectadas por el uso de este sistema.

Desde el punto de vista del Software de acuerdo también al estándar ISO [ISO, 2011] la usabilidad se define como:

"La capacidad de un software de ser comprendido, aprendido, usado y ser atractivo para el usuario, en condiciones específicas de uso"

Este enunciado, corresponde a la definición de usabilidad como parte de la calidad del software, donde la calidad se especifica en el estándar como: "Un conjunto de atributos de software que se sostienen en el esfuerzo necesitado para el uso y en la valoración individual de tal uso por un conjunto de usuarios declarados o implicados".

La usabilidad es analizada en términos de su comprensibilidad, aprendizaje, operatividad, atractividad y conformidad, tal como se describe a continuación:

- **Comprensibilidad:** capacidad del producto software para permitir al usuario entender si el software es adecuado, y como puede ser usado para tareas y condiciones de uso particulares.
- **Aprendizaje:** capacidad del producto software para permitir a los usuarios aprender a usar sus aplicaciones.
- **Operatividad:** capacidad del producto software para permitir al usuario operarlo y controlarlo.
- **Atractivo:** capacidad del producto software para ser atractivo al usuario. Se refiere principalmente al uso de colores y diseño gráfico del producto.
- **Conformidad a estándares y pautas**: capacidad del producto software para adherirse a estándares, convenciones, guías de estilo o regulaciones relacionadas con la usabilidad.

**Definición de usabilidad en buscadores semánticos.**

El concepto de usabilidad es complejo y multidimensional ya que abarca diversos aspectos que deben ser enfocados respecto al sistema u objeto de análisis. Dentro del contexto del presente trabajo, se ha elegido definir a la usabilidad de buscadores semánticos, en base a la definición proveniente del estándar ISO/IEC 9241, de la siguiente manera:

*"La Usabilidad de un sistema de búsqueda semántico, es la medida en la que éste puede ser usado por un grupo de usuarios novatos o avanzados en el uso de sistemas de búsqueda, para llevar a cabo tareas de descubrimiento de información definidas en función de sus necesidades de conocimiento y el tipo de búsqueda a realizar; de tal forma que encuentren resultados precisos y relevantes en el menor tiempo posible, con un mínimo esfuerzo invertido, y con comodidad y satisfacción en la visualización y/o manipulación de resultados; en un contexto de búsqueda de información en la web."*[Jiménez Granizo, 2008]

Así mismo los conceptos de efectividad, eficiencia y satisfacción en el contexto de los buscadores web se determinan de la siguiente manera:

- **Efectividad:** el usuario encuentra resultados precisos y relevantes que concuerdan con sus necesidades de información.
- **Eficiencia:** el usuario encuentra resultados precisos y relevantes en el menor tiempo posible y con una inversión mínima de esfuerzo.
- **Satisfacción:** el usuario se siente cómodo utilizando el buscador, durante la consulta, visualización y manipulación de resultados.


Figura 21: Esquema de conceptos de efectividad, eficiencia y satisfacción en la búsqueda semántica

# Anexo 3 – Procesamiento de archivos XMI

A continuación se presentan las funciones utilizadas para recorrer los archivos XMI de una carpeta y mostrar los resultados utilizando la API DOM:

```
public class Extractor {

//Clase para extraer los datos para mostrar de un conjunto de archivos XMI determinados

private static String directoryXMI=new
String("C:/Users/David/workspace/PrototipoBuscadorSemanticoLucene/WebContent/resources/xmiFiles/"
);

   //Función para recorrer los archivos XMI dentro de una carpeta
   public static void recorrerDir(String path){
      File directorio=new File(path);
      String[] ficheros=directorio.list();
      String line;
      for (int i = 0; i < ficheros.length; i++) {
         try {
            BufferedReader br = new BufferedReader(new FileReader(path + ficheros[i]));
            System.out.println("Contenido del fichero " + ficheros[i]);
            while ((line = br.readLine()) != null) {
               System.out.println(line);
            }
         } catch (IOException e) {
            e.printStackTrace();
         }
      }
   }

   public static void main(String[] args){
      recorrerDir(directoryXMI);

   }



   public String getDirectoryXMI() {
      return directoryXMI;
   }

   //Función para obtener los valores de cada archivo XMI
   public Resultado getXMIValues(String nombreLuceneFile){
      String nombreXMIFile=nombreLuceneFile+".xmi";
      LecturaXMIFile cmd=new LecturaXMIFile();
      return cmd.getResultadoXMI(nombreXMIFile);
   }
}
```

La lectura de los valores de cada archivo XMI se realiza con las siguientes funciones y procedimientos:

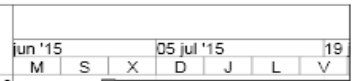
```
//Funciones y procedimientos para leer contenido de un archivo XMI con la API DOM

//Funcion para obtener todos los atributos de un archivo XMI
  public Resultado getResultadoXMI(String nombreDocXMI){
        Consulta query=new Consulta();
        Resultado r=new Resultado();
        r.setPath(nombreDocXMI);
        r.setNombre(query.generarNombreDocOriginal(nombreDocXMI));
        r.setTipoDoc(query.generarTipoDoc(nombreDocXMI));
        r.setFile(this.descargarFile(nombreDocXMI));
        DOMParser parser = new DOMParser();
        try {
           parser.parse(this.directoryXMI + nombreDocXMI);
        } catch (SAXException e) {
           // TODO Auto-generated catch block
           e.printStackTrace();
        } catch (IOException e) {
           // TODO Auto-generated catch block
           e.printStackTrace();
        }
        Document d = parser.getDocument();
        DocumentTraversal dt = (DocumentTraversal) d;

        //Se crea un iterador de nodo para parsear el archivo XMI
        NodeIterator it = dt.createNodeIterator(d.getDocumentElement(),
        NodeFilter.SHOW_ALL,
        new ObjectFilter(),
        true);
        Node n =  it.nextNode();
        while (n != null) {  //Recorro los nodos del XMI

           if(n.getNodeName()=="mio:NumeroResol"){
              r.setNumResol(this.getNumResolXMI(n)); //Obtener el valor del atributo que corresponda al
numero de resolucion
              }
           if(n.getNodeName()=="cas:Sofa"){


              String top=this.getContenidoCabeceraXMI(n);
              String body=this.getContenido_initialViewXMI(n);


              Contenido cont=new Contenido();
              cont.setEncabezado(top);
              cont.set_initialView(body);
              r.setContenido(cont);

              }
           if(n.getNodeName()=="mio:FechaResol"){
              r.setFecha(this.getFechaXMI(n));
           }
           if(n.getNodeName()=="mio:Clase"){
              r.setCategoria(this.getCategoria(n));
              System.out.println(this.getCategoria(n));
           }
           n = it.nextNode();
        }
```

```
        return r;
        }



    public NumeroResol getNumResolXMI(Node object) {

        NumeroResol nr=new NumeroResol();

         NamedNodeMap attribs = object.getAttributes(); //Obtengo los atributos
         for (int j = 0; j < attribs.getLength(); ++j){
            if (attribs.item(j).getNodeName()=="nroResol"){
               nr.setNroResol("Resoluci    "+ attribs.item(j).getNodeValue());
            }
            if (attribs.item(j).getNodeName()=="numero"){
               nr.setNumero(attribs.item(j).getNodeValue());
            }
            if (attribs.item(j).getNodeName()=="anio"){
               nr.setAnio(attribs.item(j).getNodeValue());
            }
         }
         return nr;
    }

    public String getCategoria(Node object){
       String c=new String();
       NamedNodeMap attribs = object.getAttributes();
       for (int j = 0; j < attribs.getLength(); ++j){
          if (attribs.item(j).getNodeName()=="valor"){
             c= attribs.item(j).getNodeValue();
          }
    }
       return c;
    }

    public String getContenidoCabeceraXMI(Node object) {

        String c=new String();
          NamedNodeMap attribs = object.getAttributes(); //Obtengo los atributos
          for (int j = 0; j < attribs.getLength(); ++j){
             if (attribs.item(j).getNodeName()=="sofaID"){
                if(attribs.item(j).getNodeValue()=="encabezado");
                  for (int i = 0; i < attribs.getLength(); ++i){
                     if(attribs.item(i).getNodeName()=="sofaString"){
                        c=attribs.item(i).getNodeValue();
                     }
                  }

                  }
               }
          String [] arrayLinea = c.split(" "); //Para no obtener palabras cortadas
          String texto=new String("");
          if(arrayLinea.length<=100){
             for (int i = 0; i < arrayLinea.length; i++) {
              texto=texto+ " " + arrayLinea[i];}
          }
          else {
```

```
        for (int i = 5; i <80 ; i++) {
            texto=texto+ " " + arrayLinea[i];
        }
    }

    return "..." + texto + "...";
}

public String getContenido_initialViewXMI(Node object) {
   String c=new String();
   NamedNodeMap attribs = object.getAttributes(); //Obtengo los atributos
   for (int j = 0; j < attribs.getLength(); ++j){
      if (attribs.item(j).getNodeName()=="sofaID"){
         if(attribs.item(j).getNodeValue()=="_InitialView");
           for (int i = 0; i < attribs.getLength(); ++i){
              if(attribs.item(i).getNodeName()=="sofaString"){
                 c=attribs.item(i).getNodeValue();
              }
           }

           }
      }
   return c;
     }

public Fecha getFechaXMI(Node object) {

    Fecha f=new Fecha();

     NamedNodeMap attribs = object.getAttributes(); //Obtengo los atributos
     for (int j = 0; j < attribs.getLength(); ++j){
        if (attribs.item(j).getNodeName()=="anio"){
           f.setAnio(attribs.item(j).getNodeValue());
        }
        if (attribs.item(j).getNodeName()=="mes"){
           f.setMes(attribs.item(j).getNodeValue());
        }
        if (attribs.item(j).getNodeName()=="dia"){
           f.setDia(attribs.item(j).getNodeValue());
        }

     if (attribs.item(j).getNodeName()=="fechaResolCompleta"){
           f.setFechaCompleta(attribs.item(j).getNodeValue());
        }
     }
     f.setFechaCompletaNumero(f.getDia()+ " de " + f.getMes() + " de " + f.getAnio());
     return f;
}
```

# Anexo 4 – Diagrama de Gantt completo

| Id | Nombre de tarea | Duración | Comienzo | Fin |
|---|---|---|---|---|
| 1 | **Proyecto Buscador Semantico** | **103 días** | **mié 01/07/15** | vie 20/11/15 |
| 2 | **1-Inicio** | **50 días** | **mié 01/07/15** | mar 08/09/15 |
| 3 | **1.1- Relevamiento y recolección de informacion** | 8 días? | mié 01/07/15 | vie 10/07/15 |
| 4 | 1.1.1- Reuniòn de especificaciones de lineas de trabajo | 1 día | mié 01/07/15 | mié 01/07/15 |
| 5 | 1.1.2- Estudio del trabajo de Mineria de Textos | 6 días | jue 02/07/15 | jue 09/07/15 |
| 6 | 1.1.3- Adquisiciòn de elementos de oficina para realizar investigaciòn | 1 día? | vie 10/07/15 | vie 10/07/15 |
| 7 | **1.2- Investigación de herramientas de búsqueda y procesamiento de texto** | 20 días | **vie 10/07/15** | jue 06/08/15 |
| 8 | 1.2.1- Preparacion del entorno de desarrollo | 1 día | vie 10/07/15 | vie 10/07/15 |
| 9 | 1.2.2- Estudio de Lucene | 12 días | lun 13/07/15 | mar 28/07/15 |
| 10 | 1.2.3- Estudio de herramientas para extracciòn de textos de distintos docume | 4 días | mié 29/07/15 | lun 03/08/15 |
| 11 | 1.2.4- Creaciòn de ejemplos de indexacion y bùsqueda con Lucene | 3 días | mar 04/08/15 | jue 06/08/15 |
| 12 | 1.2.5- Validacion de ejemplos | 0 días | jue 06/08/15 | jue 06/08/15 |
| 13 | **1.3- Investigación de herramientas para desarrollo WEB.** | 24 días | jue 06/08/15 | mar 08/09/15 |
| 14 | 1.3.1- Reunion de asesoramiento con Ing. Sebastiàn Garcia | 1 día | jue 06/08/15 | jue 06/08/15 |
| 15 | 1.3.2- Estudio de Java Server Faces | 5 días | vie 07/08/15 | jue 13/08/15 |
| 16 | 1.3.3- Estudio de herramienta Ice Faces y creaciòn de ejemplos | 3 días | vie 14/08/15 | mar 18/08/15 |
| 17 | 1.3.4- Estudio de herramienta Prime Faces y creaciòn de ejemplos | 10 días | mié 19/08/15 | mar 01/09/15 |
| 18 | 1.3.5- Integraciòn de Java Server Faces, Prime Faces y Lucene en la creaciòn | 4 días | mié 02/09/15 | lun 07/09/15 |
| 19 | 1.3.6- Validaciòn de ejemplo de bùsqueda e indexaciòn WEB | 0 días | mar 08/09/15 | mar 08/09/15 |
| 20 | **2- Planeamiento del proyecto del Buscador Semantico** | **10 días** | **mié 09/09/15** | mar 22/09/15 |
| 21 | 2.1- Definición de objetivos y alcance del proyecto | 0 días | **mié 09/09/15** | mié 09/09/15 |
| 22 | 2.2- Definición del cronograma | 0 días | **mié 09/09/15** | mié 09/09/15 |
| 23 | 2.3- Análisis del proyecto | 7 días | **mié 09/09/15** | jue 17/09/15 |
| 24 | 2.4- Análisis de riesgos y definición de estrategias | 2 días | vie 18/09/15 | lun 21/09/15 |
| 25 | 2.5- Validacion de actividades desarrolladas en Inicio | 0 días | mar 22/09/15 | mar 22/09/15 |
| 26 | 2.6- Validacion del análisis del proyecto | 0 días | mar 22/09/15 | mar 22/09/15 |
| 27 | 2.7- Validacion del análisis de riesgos del proyecto y estrategias | 0 días | mar 22/09/15 | mar 22/09/15 |

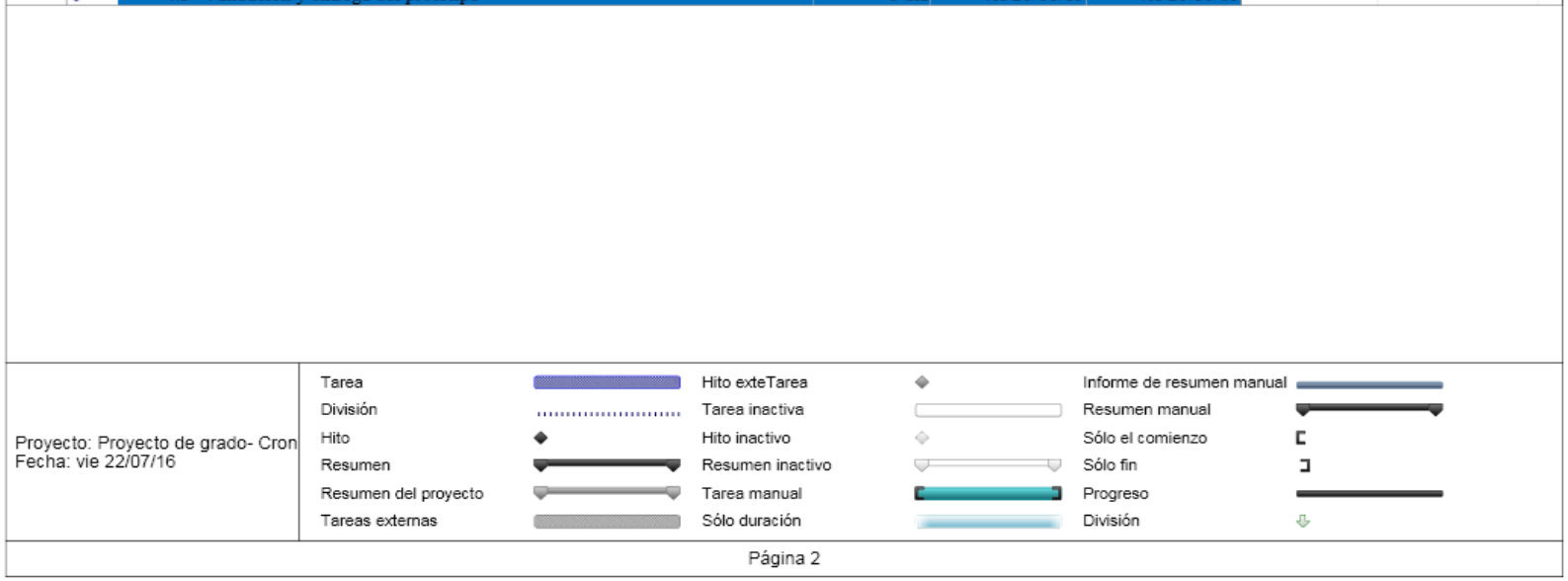



Proyecto: Proyecto de grado- Cron
Fecha: vie 22/07/16

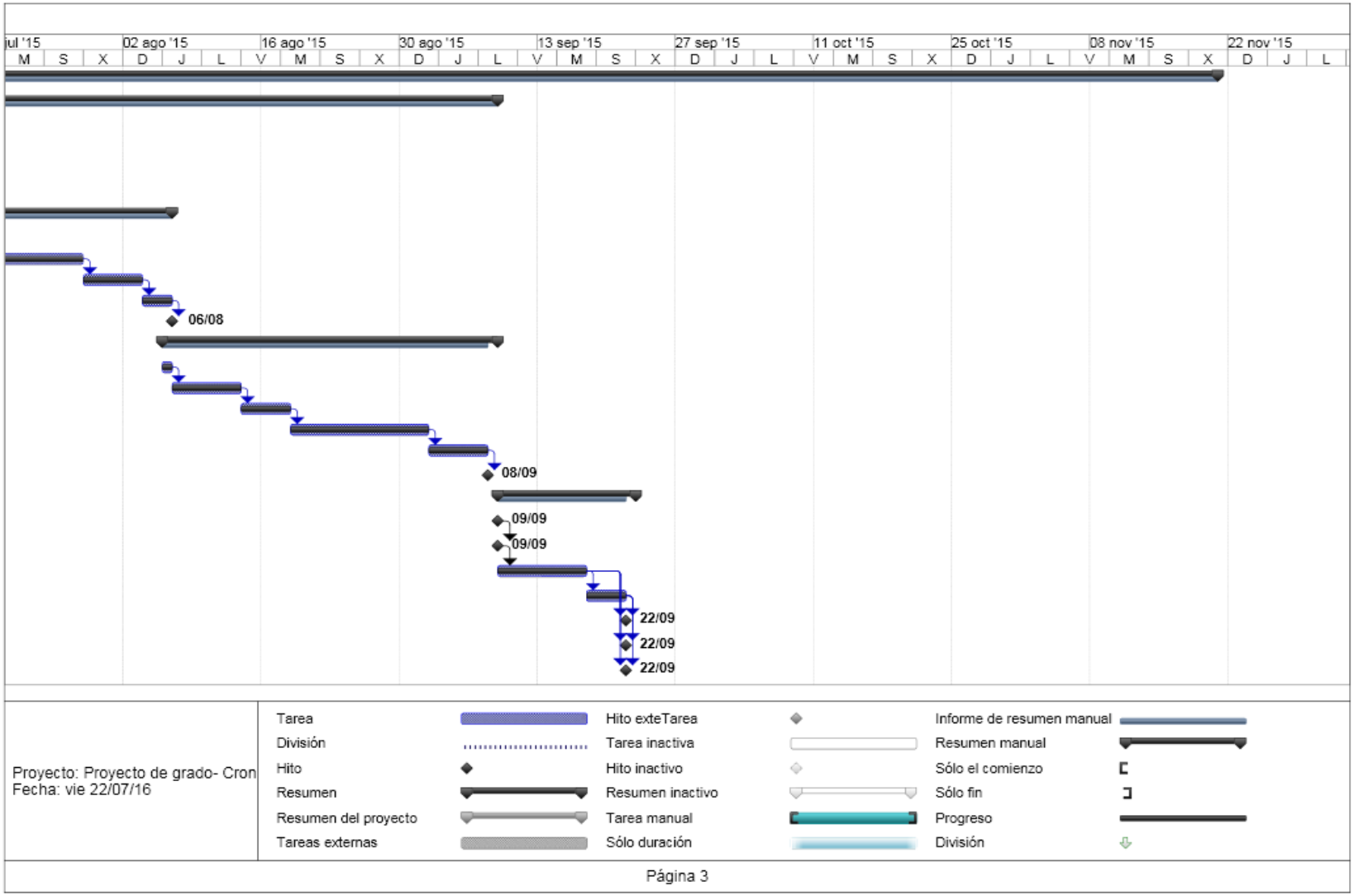
| | | |
|---|---|---|
| Tarea | Hito exteTarea | Informe de resumen manual |
| División | Tarea inactiva | Resumen manual |
| Hito | Hito inactivo | Sólo el comienzo |
| Resumen | Resumen inactivo | Sólo fin |
| Resumen del proyecto | Tarea manual | Progreso |
| Tareas externas | Sólo duración | División |

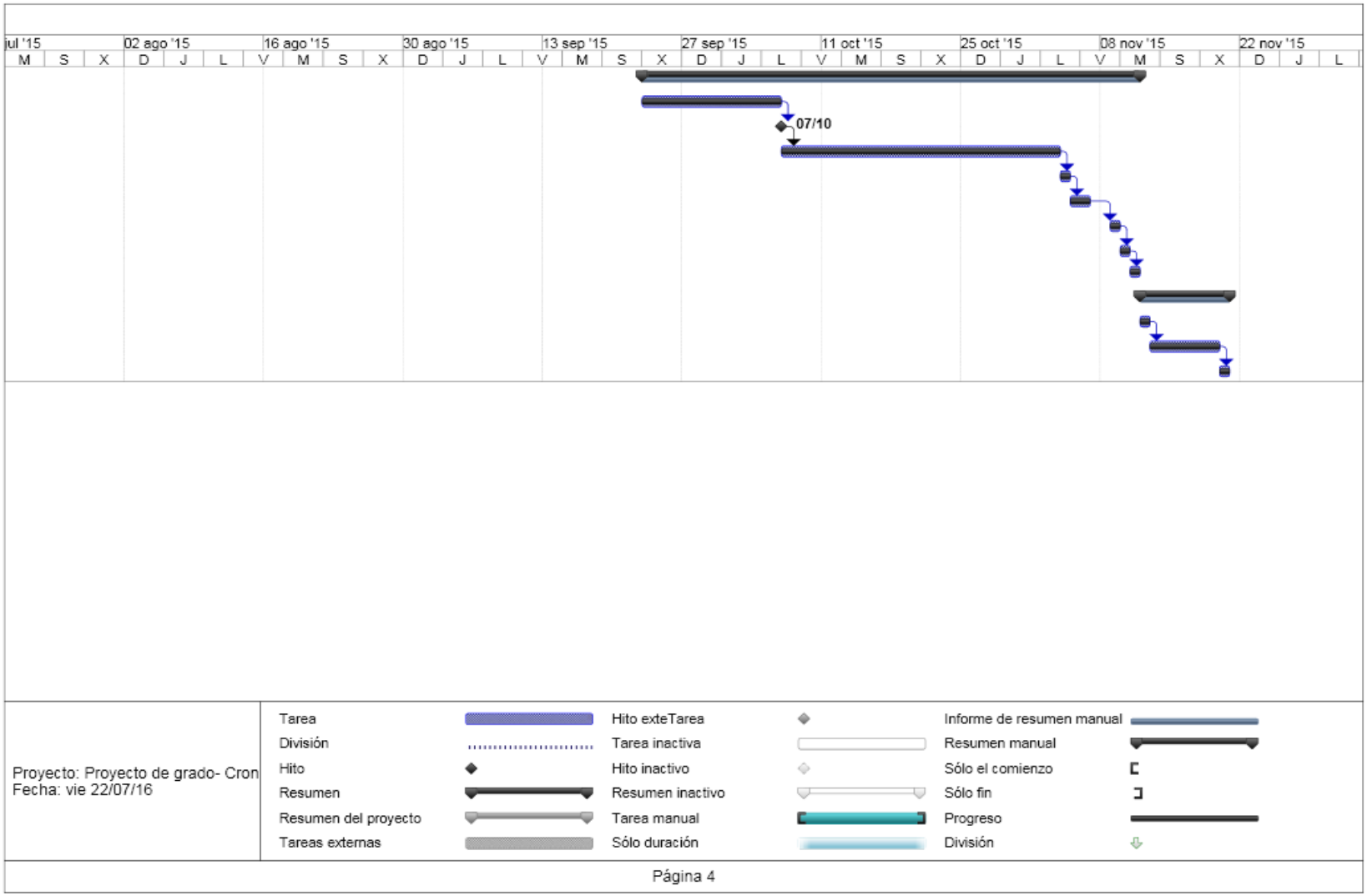
Página 1

| Id | Nombre de tarea | Duración | Comienzo | Fin |
|---|---|---|---|---|
| 28 | **3- Ejecucion** | **36 días** | mié 23/09/15 | mié 11/11/15 |
| 29 | 3.1- Definición del análisis y diseño del prototipo | 10 días | mié 23/09/15 | mar 06/10/15 |
| 30 | 3.2- Presentación del análisis y diseño del prototipo | 0 días | mié 07/10/15 | mié 07/10/15 |
| 31 | 3.3- Codificación del prototipo | 20 días | mié 07/10/15 | mar 03/11/15 |
| 32 | 3.4- Planificación de pruebas | 1 día | mié 04/11/15 | mié 04/11/15 |
| 33 | 3.5- Pruebas de prototipo | 2 días | jue 05/11/15 | vie 06/11/15 |
| 34 | 3.6- Correcciones del prototipo | 1 día | lun 09/11/15 | lun 09/11/15 |
| 35 | 3.7-Validacion de informes de realización de pruebas | 1 día | mar 10/11/15 | mar 10/11/15 |
| 36 | 3.8- Implementación del prototipo | 1 día | mié 11/11/15 | mié 11/11/15 |
| 37 | **4- Cierre** | **7 días** | jue 12/11/15 | vie 20/11/15 |
| 38 | 4.1- Control de objetivos, alcances y cronograma | 1 día | jue 12/11/15 | jue 12/11/15 |
| 39 | 4.2- Preparación de la presentación del prototipo | 5 días | vie 13/11/15 | jue 19/11/15 |
| 40 | 4.3- Validacion y entrega del prototipo | 1 día | vie 20/11/15 | vie 20/11/15 |




Página 2


jul '15
02 ago '15
16 ago '15
30 ago '15
13 sep '15
27 sep '15
11 oct '15
25 oct '15
08 nov '15
22 nov '15
06/08
08/09
09/09
09/09
22/09
22/09
22/09
Proyecto: Proyecto de grado- Cron
Fecha: vie 22/07/16
Tarea
División
Hito
Resumen
Resumen del proyecto
Tareas externas
Hito exteTarea
Tarea inactiva
Hito inactivo
Resumen inactivo
Tarea manual
Sólo duración
Informe de resumen manual
Resumen manual
Sólo el comienzo
Sólo fin
Progreso
División
Página 3


jul '15
02 ago '15
16 ago '15
30 ago '15
13 sep '15
27 sep '15
11 oct '15
25 oct '15
08 nov '15
22 nov '15
M S X D J L V M S X D J L V M S X D J L V M S X D J L V M S X D J L V M S X D J L
07/10
Proyecto: Proyecto de grado- Cron
Fecha: vie 22/07/16
Tarea
División
Hito
Resumen
Resumen del proyecto
Tareas externas
Hito exteTarea
Tarea inactiva
Hito inactivo
Resumen inactivo
Tarea manual
Sólo duración
Informe de resumen manual
Resumen manual
Sólo el comienzo
Sólo fin
Progreso
División
Página 4