

UNIVERSIDAD PRIVADA ANTENOR ORREGO

FACULTAD DE INGENIERIA

**ESCUELA PROFESIONAL DE INGENIERÍA DE COMPUTACION Y
SISTEMA**



**“SISTEMA INTELIGENTE DE BÚSQUEDA DE EXPEDIENTES
JUDICIALES BASADO EN WEB SEMANTICA PARA CONSULTORIOS
JURÍDICOS”**

**TESIS PARA OBTENER EL TÍTULO PROFESIONAL DE INGENIERO DE
COMPUTACIÓN Y SISTEMAS**

LÍNEA DE INVESTIGACIÓN: WEB SEMÁNTICA

AUTORES: Br. MARLON DAVID ARAUJO CASTAÑEDA.

Br. FRANCISCO JAVIER SUPO CORNEJO.

ASESOR: Dr. LUIS VLADIMIR URRELO HUIMAN

TRUJILLO - PERU

2015

HOJA DE FIRMAS

“SISTEMA INTELIGENTE DE BÚSQUEDA DE EXPEDIENTES JUDICIALES BASADO EN WEB SEMANTICA PARA CONSULTORIOS JURÍDICOS”

Autores:

Br. Marlon David Araujo Castañeda

Br. Francisco Javier Supo Cornejo

Aprobado por:

Ing. Jorge Huapaya Escobedo
Presidente
No. CIP: 17215

Ing. Enrique Cardenas Rengifo
Secretario
No. CIP: 148025

Ing. Heber Abanto Cabrera
Vocal
No. CIP: 106421

Dr. Luis Vladimir Urrelo Huiman
Asesor.
No. CIP: 88212

DEDICATORIAS

Hija mía, te dedico este logro, que es por ti que he alcanzado esta meta, eres tú la persona que cada día miro, y muchas veces sin entender tu media lengua, sé que me dices que puedo lograrlo. Dedico, a su vez, este logro a mis dos hermanas, para que les sea de ejemplo que todo lo que se desea se puede alcanzar con perseverancia y sacrificio. También a mi esposa, por ser la persona que decidió caminar a mi lado, tomados de la mano, superando los retos que nos pone la vida. Y a mis padres, que no tuvieron miedo de enfrentar al mundo, que no tuvieron vergüenza de trabajar en lo que fuera, para conseguir que sea quien soy hoy.

- Marlon Araujo.

Este trabajo de tesis ha sido una gran experiencia en todo sentido y te lo agradezco padre, y no cesan mis ganas de decir gracias a ti que esta meta está cumplida. A mis grandes amigos Marlon y Leonardo quienes fueron un gran apoyo emocional durante el tiempo en que escribía esta tesis. Gracias a las personas que, de una manera u otra, han sido claves en mi vida profesional, y por extensión, en la personal: a mis amigos Cesar Jaeger y Ángeles López, quienes siempre comparten sus experiencias, ¡gracias, amigos, por estar ahí! Todo esto nunca hubiera sido posible sin el amparo incondicional de mi familia, mis padres y mis hermanos. Esto es también vuestro premio.

- Francisco Supo.

AGRADECIMIENTOS

Sería muy vanidoso por mi parte comenzar esta tesis sin agradecer sinceramente a todas las personas que en menor o mayor medida han mostrado su apoyo para su finalización. Sé que estas humildes líneas no son suficientes para mostrarles mi inmenso y profundo agradecimiento por su ayuda e infinita paciencia conmigo (a veces más de la que merecía) por la cual todo esto ha sido posible. Quiero agradecerse a mis padres: por confiar en mí, por darme una oportunidad de ser y de estar en la vida. Por ofrecermme muchas veces más de lo que me merecía; por estar en los buenos y malos momentos; por ser todo y más de lo que cualquier hijo puede pedir. Les estoy eternamente agradecido. Espero que algún día os sintáis tan orgullosos de mí, como yo de vosotros. Todos mis logros son también vuestros.

- Francisco Supo.

Hoy en día, se ha perdido el hábito de agradecer los logros, porque se cree que son personales; pero yo pienso diferente. Quiero agradecer a todas aquellas personas que han influido de manera directa o indirecta en mi vida. Pero tengo que hacerlo en especial a mi hija, Lina Fabiola, que sin hablar, me dice que siga adelante con una sonrisa. A mis hermanas, Claudia y Diana, quienes me alentaron en cada momento de mi vida. A mi esposa, Alicia, que sin dudarlo me apoya cada momento. A mis padres, Quirino y Lida, que jamás dudaron en darme su apoyo y consejo, y siempre creyeron en mí. A mi tía Gloria quien siempre me brinda apoyo y cariño, por lo que la considero una segunda madre. A mis padres políticos, Francisco y Miriam, que siempre me dan la mano cuando la necesito.

- Marlon Araujo.

Un especial agradecimiento a BeeMovil SAC, empresa que nos enseñó algunas de las tecnologías empleadas en el desarrollo de esta investigación.

RESUMEN Y PALABRAS CLAVE

La mayor parte de estudios jurídicos en el Perú tiene un problema, el tiempo que emplean en localizar sus expedientes judiciales entre sus archivos, el cual tiene una duración prolongada, donde el Estudio Jurídico Díaz Pulido no es la excepción; dado que los expedientes están en 2 almacenes, y en ocasiones hay expedientes en las oficinas de los abogados, lo cual dificulta su rápida localización para su trabajo. Puesto que estos archivos (expedientes judiciales de materia civil, familiar, laboral y constitucional) contienen toda la información de cómo va avanzando el proceso judicial y son buscados de manera manual.

Para solucionar este problema fue necesario modelar los procesos principales del estudio jurídico, con el fin de entender cómo funciona realmente el desarrollo de sus actividades; a partir del cual se plantea un prototipo de sistema inteligente de búsqueda que utilice el lenguaje natural de las personas para establecer parámetros de búsqueda, apoyándose en la metodología Methontology para realizar esta conversión.

Luego de la creación del prototipo de sistema inteligente de búsqueda, se procede a tabular los tiempos obtenidos buscando el mismo expediente, lo cual permitió visualizar la diferencia exacta de tiempos entre la búsqueda con la técnica hasta ahora utilizada y el uso del sistema inteligente de búsqueda. Partiendo de lo ya expresado ¿un sistema inteligente de búsqueda basado en web semántica acortará de manera considerable el tiempo de búsqueda de los expedientes judiciales en el Estudio Jurídico Díaz Pulido? Con el objetivo de contestar esta interrogante, se hicieron las pruebas necesarias. Donde el análisis estadístico arrojó una reducción del 98.1% del tiempo medio de búsqueda. Lo cual brinda como respuesta que, en efecto, un sistema inteligente de búsqueda reduce significativamente el tiempo de búsqueda de expedientes judiciales.

Palabras clave: Web Semántica, Sistema Inteligente de Búsqueda, Expediente Judicial.

ABSTRACT Y KEYWORDS

Most law firms in Peru have a problem, the time spent in locating their case files between your files, which has a long duration, where the law firm Diaz Pulido is no exception; Since the registries are in 2 warehouses, and sometimes there are registries in the offices of lawyers, making it difficult to quickly locate them for their work. Because these files (court registries of civil, family, labor and constitutional) contain all the information of how the trial progresses and are manually searched.

To solve this problem it was necessary to model the main processes of the law firm, in order to understand how the development of its activities actually works; from which a prototype intelligent search system that uses the natural language of the people to set search parameters, relying on Methontology methodology to perform this conversion arises.

After prototyping intelligent search system, we proceed to tabulate the times obtained looking for the same file , which allowed visualize the exact time difference between the search with the technique so far used and the use of intelligent search system . From what has been said does a smart search system based on semantic web shorten considerably the time search of court registries in the law firm Diaz Pulido? In order to answer this question, the necessary tests were done. Where the statistical analysis showed a reduction of 98.1 % of the average search time. Which provides in response, in effect, an intelligent search system significantly reduces the time spent searching for court registries.

Keywords: Semantic Web, Intelligent Search System, Judicial File.

INDICE

HOJA DE FIRMAS	i
DEDICATORIAS.....	ii
AGRADECIMIENTOS	iii
RESUMEN Y PALABRAS CLAVE	iv
ABSTRACT Y KEYWORDS.....	v
INDICE	vi
INDICE DE TABLAS E LUSTRACIONES.....	x
I. INTRODUCCIÓN	1
II. Marco Teórico	4
1. Web Semántica	4
2. Búsqueda Inteligente.....	5
3. MongoDB.....	5
3.1. Características.....	6
3.2. Funcionamiento de MongoDB.....	7
3.3. Donde Utilizar MongoDB	7
3.4. Donde No Usar MongoDB.....	7
4. Migración de SQL a NoSQL.....	8
5. Zend Framework	12
5.1. Características.....	12
5.2. Estructura Zend	12
5.3. Desarrollo con Zend	12
5.4. Requerimientos	13
6. Web Services Description Language	13
7. BPMN (Business Process Model and Notation).....	14
7.1. Objetos de Flujo	17
8. Metodología de Desarrollo ICONIX.....	18
8.1. Fase 1: Análisis y Requerimientos	20
8.2. Fase 2: Diseño Preliminar	22
8.3. Fase 3: Diseño Detallado	24
8.4. Fase 4: Implementación	25
9. JavaScript Object Notation	26

10.	Methontology	28
10.1.	Actividades de Desarrollo.....	29
10.2.	Tareas de Conceptualización.....	30
11.	Expedientes Judiciales	32
12.	Técnica de Observación.....	33
III.	MATERIALES Y MÉTODOS	37
1.	Metodología de Investigación.....	37
2.	Diagramación de Procesos	37
3.	Construcción de la Sistema Inteligente de Búsqueda de Expedientes Judiciales.....	37
4.	Evaluación del Sistema Inteligente de Búsqueda	37
IV.	RESULTADOS	39
1.	Modelado de Procesos del Estudio Jurídico Díaz Pulido.....	39
1.1.	Subproceso de Búsqueda de Expediente	39
1.2.	Anexado de Documentación a Expediente	39
1.3.	Búsqueda de Expediente por Cliente.....	40
2.	Diseño de la Sistema Inteligente de Búsqueda de Expedientes Judiciales	41
2.1.	Listado de Actores.....	41
2.2.	Lista de Casos de Uso del Negocio	41
2.3.	Especificación de Casos de Uso del Negocio.....	41
2.4.	Modelo de Casos de Uso del Negocio	42
2.5.	Requerimientos	42
2.6.	Diagramas de Caso de Uso	43
2.7.	Especificación de Casos de Uso.....	44
2.8.	Diagramas de Robustez.....	49
2.9.	Diagramas de Secuencia	50
2.10.	Diagrama de Componentes	52
2.11.	Modelo de Dominio.....	53
2.12.	Diagrama de Clases	53
2.13.	Modelo de Base de Datos	55
2.14.	Algoritmo de Búsqueda	55
3.	Migración de Datos.....	56

3.1.	Estructura Relacional	57
3.2.	Estructura Plana	58
3.3.	Proceso de Migración de Datos	58
4.	Construcción de Ontologías	59
4.1.	Glosario de Términos	59
4.2.	Diagrama de Taxonomía de Conceptos.....	60
4.3.	Diagrama de Relaciones Binarias	60
4.4.	Diccionario de Conceptos	63
4.5.	Descripción de Axiomas.....	63
5.	Prototipos de Interfaz del Sistema Inteligente de Búsqueda	64
5.1.	Interfaz del Buscador.....	65
5.1.	Interfaz de Resultado de Búsqueda	65
5.1.	Interfaz de Detalle de Expediente	66
6.	Tabulación de las Encuestas	67
7.	Tabulación de los Tiempos Obtenidos	68
7.1.	Medición de Tiempos	68
V.	DISCUSIÓN.....	70
1.	Gráfico de Evaluación de Encuesta	70
2.	Procesamiento Estadístico de la Encuesta	70
2.1.	Promedio de Puntaje	70
2.2.	Análisis de Resultados Promedio.....	71
3.	Gráfico Comparativo de Tiempos.....	71
4.	Procesamiento Estadístico de los Resultados Obtenidos	71
4.1.	Hipótesis Estadística	72
4.2.	Nivel de Error.....	72
4.3.	Elección de la Prueba Estadística	72
4.4.	Lectura del P-Valor.....	73
5.	Demostración de la Hipótesis.....	74
VI.	CONCLUSIONES.....	76
VII.	RECOMENDACIONES	77
VIII.	REFERENCIAS BIBLIOGRÁFICAS.....	78

IX.	Anexos.....	81
1.	Anexo N° 1: Encuesta de Usabilidad y Eficiencia	81

INDICE DE TABLAS E LUSTRACIONES

Ilustración 1: Web 3.0 o Web Semántica (Interacciones, 2007)	4
Ilustración 2: Comparación entre la web actual y la web semántica (La web semántica, la siguiente generación de webs, 2007)	5
Ilustración 3: Diagrama de Procesos de Negocio (Sebastián Bayard Ocares , 2011)	15
Ilustración 4: Ejemplo de Modelado de Procesos de Negocio.....	15
Ilustración 5: Modelo del Proceso de Solicitud de Crédito (BIZAGI)	17
Ilustración 6: Subproceso de Verificación de Información del Cliente (BIZAGI).....	17
Ilustración 7: Resumen de la Metodología (García, 2011)	19
Ilustración 8: Ejemplo de Diagrama de Robustez	24
Ilustración 9: Ejemplo de Diagrama de Secuencia	25
Ilustración 10: Objetos JSON	27
Ilustración 11: Arreglos JSON	27
Ilustración 12: Valores JSON	27
Ilustración 13: Cadenas JSON.....	28
Ilustración 14: Números JSON.....	28
Ilustración 15: Diagrama del método Methontology.....	30
Ilustración 16: Tareas de Conceptualización de Methontology	30
Ilustración 17: Modelo de Subproceso de Búsqueda de Expediente	39
Ilustración 18: Modelo del Proceso de Anexado de Documentación a Expediente	40
Ilustración 19: Modelo del Proceso de Búsqueda de Expediente por Cliente	41
Ilustración 20: Modelo de Caso de Uso del Negocio	42
Ilustración 21: Administrar Expedientes	43
Ilustración 22: Administrar Clientes.....	44
Ilustración 23: Registro de Cliente	49
Ilustración 24: Buscar Cliente.....	49
Ilustración 25: Registrar Expediente	49
Ilustración 26: Buscar Expediente.....	49
Ilustración 27: Editar Expedientes.....	50
Ilustración 28: Registrar Cliente – Secuencia	50
Ilustración 29: Buscar Cliente – Secuencia.....	51
Ilustración 30: Registrar Expediente – Secuencia	51
Ilustración 31: Buscar Expediente – Secuencia.....	52
Ilustración 32: Editar Expedientes - Secuencia.....	52
Ilustración 33: Diagrama de Componentes Sistema Inteligente de Búsqueda.....	53
Ilustración 34: Modelo de Dominio	53
Ilustración 35: Diagrama de Clases	54
Ilustración 36: Modelo de Datos.....	55
Ilustración 37: Diagrama N-S del Algoritmo de Búsqueda.....	56
Ilustración 38: Modelo Relacional a Migrar a Plano.....	57

Ilustración 39: Modelo de Datos Plano de la Lista de Expedientes Judiciales	58
Ilustración 40: Proceso de Migración de Datos SQL a NoSQL	58
Ilustración 41: Diagrama de Taxonomía de Conceptos	60
Ilustración 42: Relación Binaria Expediente – Materia	60
Ilustración 43: Referencias de Materia – MongoDB.....	61
Ilustración 44: Relación Binaria Expediente – Juzgado.....	61
Ilustración 45: Referencias de Juzgado – MongoDB.....	62
Ilustración 46: Relación Binaria Expediente - Distrito Judicial	62
Ilustración 47: Referencias de Distrito Judicial – MongoDB.	63
Ilustración 48: Axioma del Cuaderno Principal - MongoDB	64
Ilustración 49: Prototipo de Interfaz de Búsqueda	65
Ilustración 50: Interfaz de Resultados de Búsqueda.....	66
Ilustración 51: Interfaz de Detalle de Expediente.....	67
Ilustración 52: Gráfico Descriptivo de la Encuesta.....	70
Ilustración 53: Polígono comparativo de Tiempos de Búsqueda.....	71
Ilustración 54: Nivel Alfa	72
Tabla 1: Diferencias entre SQL y MongoDB	9
Tabla 2: Estructura de datos MySQL (Relacional) y MongoDB (No Relacional) (Solvetic, 2014) 9	
Tabla 3: Sentencias básicas lenguaje de consultas MongoDB (Solvetic, 2014)	10
Tabla 4: Modelos relacional vs orientado a documentos (Manjarrez Antaño, Martínez Castro, & Cuevas Valencia, 2014)	11
Tabla 5: Tipos de Eventos BPMN	18
Tabla 6: Tipos de Actividades BPMN.....	18
Tabla 7: Tabulación de Encuesta.....	67
Tabla 8: Tiempos de Búsqueda Manual	69
Tabla 9: Tiempo de Búsqueda Automático	69
Tabla 10: Calificación promedio por Sujeto Evaluado	70
Tabla 11: Análisis Estadístico Descriptivo	71
Tabla 12: Elección de la Prueba Estadística.....	72
Tabla 13: Pruebas de Normalidad.....	73
Tabla 14: Calificaciones de Normalidad.....	73
Tabla 15: Resultados T de Student.....	74
Tabla 16: Evaluación de Resultados T de Student	74

I. INTRODUCCIÓN

El Consultorio Jurídico Díaz Pulido, actualmente realiza sus tareas de búsqueda de expediente judiciales de manera manual, proceso que se realiza con lentitud por el hecho que dichos expedientes están ubicados de manera alfabética, considerando la primera letra del apellido paterno solamente, en dos almacenes y también, de forma aleatoria y ocasional, en las seis oficinas de los abogados.

Esta desorganización en el manejo de sus expedientes judiciales conlleva a las siguientes deficiencias:

- Tiempo: actualmente el estudio jurídico presenta demoras en la búsqueda de sus documentos, ya que cada uno de estos representa a un cliente y en algunos casos existe más de un expediente porque puede llevar más de un proceso judicial a la vez, esta actividad puede tardar de veinte minutos a varias horas.
- Servicio Deficiente: al no tener los documentos de manera ordenada y de fácil acceso, eso influirá en el manejo de los servicios hacia sus clientes, ya que tienen que hacer las mencionadas búsquedas cada vez que un cliente hace una consulta sobre el progreso de su caso.

La búsqueda de información en el Estudio Jurídico Díaz Pulido es importante ya que de ello depende su desarrollo. Bajo este contexto, llegamos a la siguiente formulación:

¿Cómo reducir el tiempo de búsqueda de expedientes judiciales soportado por sistemas inteligentes en el consultorio jurídico Díaz Pulido?

Las siguientes investigaciones forman parte de los antecedentes para la presente tesis:

Autores: (Niño Zambrano, López Gómez, Andrade, Cobos Lozada, & Fabregat Gesa, 2012)

Título de investigación: Modelo Semántico de Expansión de Consultas para la Búsqueda Web, 15 Junio de 2012.

Conclusiones: La búsqueda semántica proporciona grandes ventajas en comparación con la búsqueda web tradicional puesto que la primera se enfoca en identificar relaciones existentes entre conceptos y en determinar el sentido más apropiado a una frase, a diferencia de la segunda que sólo se basa en la coincidencia de palabras clave.

Contribución: El uso de relaciones entre conceptos existentes permitirá que las consultas generadas sean las adecuadas, obteniendo de esta manera un resultado más fiable y preciso.

Autor: (Suárez Barón & Salinas Valencia, 2009)

Título de investigación: Una aproximación a la indexación semántica y la recuperación de información.

Conclusiones: Con base en este criterio para la indexación semántica, técnicas de encaminamiento y técnicas de similitud semántica se puede concluir que existe un cuerpo amplio de la investigación en cada una de estas áreas, muchas de las cuales soportarán una mayor investigación en el contexto de la Web Semántica indexación y gestión del conocimiento.

Contribución: La indexación y encaminamiento semántico de términos será útil a la hora de buscar un término en específico, obteniendo de esta manera los expedientes relacionados con la terminología buscada.

La justificación de esta tesis es la siguiente.

La mayor parte de estudios jurídicos en el Perú tiene un problema, el tiempo que emplean en localizar sus expedientes judiciales entre sus archivos, tiene una prolongada duración dado que los expedientes están en 2 almacenes, y en ocasiones hay expedientes en las oficinas de los abogados, lo cual dificulta su rápida localización para su trabajo. Puesto que estos archivos (expedientes judiciales de materia civil, familiar, laboral y constitucional) contienen toda la información de cómo va avanzando el proceso judicial. Y la construcción de un sistema inteligente de búsqueda para estos archivos en forma digital dentro de un sistema computacional, en donde se aplicará la tecnología de indexación al área judicial en un entorno de web semántica; viene a ser la solución más viable en cuestión de optimización de tiempo y recursos, el cual permitirá agilizar la labor que desempeñan los abogados en su labor. Para este propósito, el desarrollo del prototipo del mismo deberá ser desarrollado en una plataforma web ya que se adapta a las necesidades de la organización estudiada porque permite desarrollar inteligencia artificial en la web mediante motor de inferencia plano y sin la necesidad de complementos en las estaciones de los usuarios finales.

Teniendo como objetivo general:

- Desarrollar un sistema inteligente de búsqueda basado en web semántica para el rápido acceso a la información de los expedientes judiciales en el consultorio jurídico Díaz Pulido de Trujillo.

Y los objetivos específicos de esta investigación son:

- Modelar el proceso de búsqueda de expedientes judiciales del consultorio jurídico Díaz Pulido, utilizando la representación formal Business Project Management and Notes.
- Desarrollar un prototipo de buscador semántico aplicando las técnicas de traducción de consulta en lenguaje natural a lenguaje formal y desarrollo de ontologías de Rubén Darío Alvarado, con la metodología Methontology; así como la utilización la metodología de desarrollo ICONIX.
- Realiza pruebas al prototipo antes desarrollado a búsquedas de expedientes judiciales en el consultorio Díaz Pulido y obtener los resultados.

Hipótesis de trabajo (H1): Un sistema inteligente de búsqueda basado en web semántica acortará el tiempo en las búsquedas de expedientes judiciales del consultorio jurídico Díaz Pulido.

Hipótesis Nula (H0): Un sistema inteligente de búsqueda basado en web semántica no acortará el tiempo en las búsquedas de expedientes judiciales del consultorio jurídico Díaz Pulido.

Indicación de variables:

- Variable independiente (x): sistema inteligente de búsqueda basado en web semántica.
 - Indicadores:
 - Facilidad
 - Rapidez.
- Variable dependiente (y): búsquedas de expedientes judiciales del consultorio jurídico Díaz Pulido.
 - Indicadores:
 - Rapidez.
 - Eficiencia.

II. Marco Teórico

1. Web Semántica

En la definición de (Berners-Lee, Hendler, & Lassila, 2001), la web semántica es una extensión de la web tradicional donde la información recibe un significado bien definido, permitiendo a los computadores y las personas trabajar en cooperación de mejor forma.

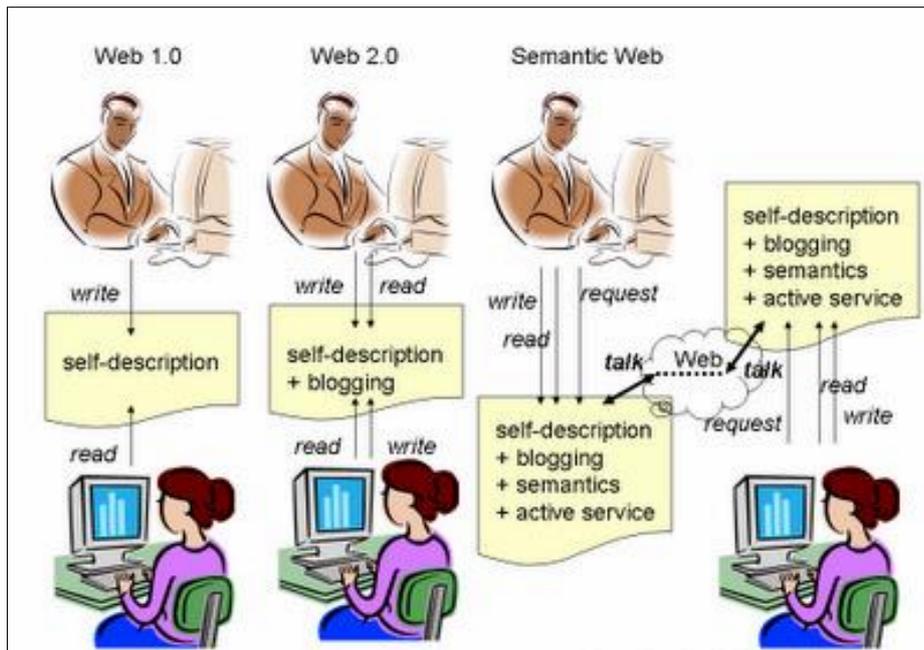


Ilustración 1: Web 3.0 o Web Semántica (Interacciones, 2007)

La web se asemeja a un grafo formado por nodos del mismo tipo, y arcos que representan hiperenlaces igualmente indiferenciados, tal y como se ve en la Ilustración 1: Web 3.0 o Web Semántica. Por el contrario, en la web semántica cada nodo (recurso) tiene un tipo, y los arcos representan relaciones explícitamente diferenciadas.

La Ilustración 2: Comparación entre la web actual y la web semántica se basa en la idea de añadir metadatos semánticos y ontológicos a la World Wide Web de manera formal, para que así sea posible evaluarlas automáticamente por máquinas de procesamiento. (La web semántica, la siguiente generación de webs, 2007)

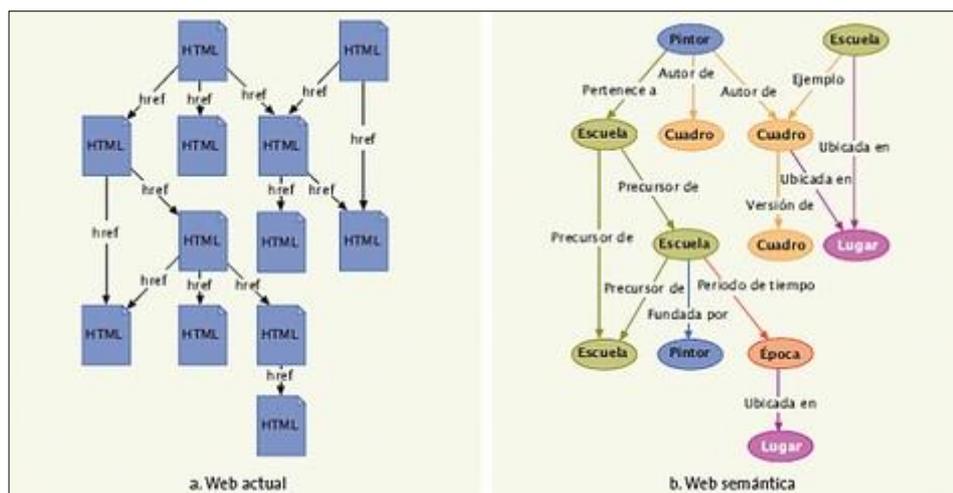


Ilustración 2: Comparación entre la web actual y la web semántica (La web semántica, la siguiente generación de webs, 2007)

2. Búsqueda Inteligente

Los buscadores semánticos han dejado los laboratorios de investigación para convertirse en aplicaciones funcionales. Uno de los primeros buscadores semánticos es SWOOGLE (Ding, y otros, 2004) el cual extrae metadatos a partir de documentos RDF y OWL. También es capaz de encontrar las relaciones existentes entre dichos documentos. En el 2006 surge SemSearch (Lei, Uren, & Motta, 2006), un buscador semántico orientado a usuarios comunes que no están familiarizados con las tecnologías de la web semántica o con el dominio específico de los datos semánticos. En (d'Aquin & Motta, 2011) se presenta Watson, un buscador semántico que proporciona un conjunto de APIs que contienen funciones de alto nivel para encontrar, explorar y consultar datos semánticos y ontologías que han sido publicadas en línea. (Hidalgo Delgado & Rodríguez Puente, 2013).

3. MongoDB

Para el desarrollo de nuestro prototipo del Sistema inteligente de búsqueda de Expedientes Judiciales decidimos usar una base de datos documental de código abierto que proporciona un alto rendimiento, alta disponibilidad y la escala automática.

Un registro de MongoDB es un documento, que es una estructura de datos compuesta por pares de campos y de valor. Documentos MongoDB son similares a los objetos *JavaScript Object Notation* (JSON). Los valores de los campos pueden incluir otros documentos, matrices y matrices de documentos. (MongoDB, 2015)

Ejemplo:

```
{ "nombre": "Mariana", "edad": 25, "estado": "A", "grupo": [ "Desarrollador", "Analista", "Programador" ] }
```

Los documentos pueden ser comparables a los registros o tuplas que poseen las bases de datos relaciones, aunque los documentos no se adaptan a ningún esquema, es decir, son *schema-less* (sin esquema). Cuando hablamos de este tipo de bases de datos podemos decir que son un tipo especial de las bases de datos clave-valor que se explicarán más adelante, con todas las ventajas que dicho tipo pueda proporcionar, pero con la excepción de que el valor se almacena de manera que el servidor pueda entenderlo y realizar operaciones sobre los datos. El formato en el que suele ser almacenado es JSON (Ecma-International, 2013), un formato ligero destinado al intercambio de datos, aunque se puede almacenar en otros tipos como puede ser XML (*Extensible Markup Language*). Otra de sus grandes características es que no solo nos permite realizar búsquedas del tipo clave-valor, si no que se pueden hacer búsquedas por contenido de los documentos a través de consultas más complejas.

Para empezar comenzaremos a describir las principales características por las que usamos MongoDB.

3.1. Características

3.1.1. Alto Rendimiento

MongoDB proporciona la persistencia de datos de alto rendimiento. En particular, el apoyo a los modelos de datos incrustados reduce la actividad de E / S (Entrada y salida) en el sistema de base de datos.

- Índices apoyan consultas más rápidas y pueden incluir teclas de documentos y arreglos embebidos.

3.1.2. Alta Disponibilidad

- Para proporcionar una alta disponibilidad, facilidad de replicación de MongoDB, llamados conjuntos de réplicas, incluidas:
 - Automatización de conmutación por error.
 - Redundancia de datos.
- Un conjunto de réplicas es un grupo de servidores MongoDB que mantienen el mismo conjunto de datos, proporcionando redundancia y aumento de la disponibilidad de datos.

3.1.3. Escalabilidad Automática

- MongoDB ofrece escalabilidad horizontal como parte de su núcleo de funcionalidad.
- Automático sharding distribuye los datos en un clúster de máquinas.

- Conjuntos de réplicas pueden proporcionar eventualmente consistente lee para implementaciones de alto rendimiento de baja latencia. (MongoDB, 2015)

3.2. Funcionamiento de MongoDB

MongoDB está escrito en C++, aunque las consultas se hacen pasando objetos JSON como parámetro. Es algo bastante lógico, dado que los propios documentos se almacenan en BSON (Fernandez, 2014).

MongoDB se ejecuta bajo consola, esta consola está construida sobre JavaScript, por lo que las consultas se realizan utilizando ese lenguaje. Además de las funciones de MongoDB, se puede utilizar muchas de las funciones propias de JavaScript.

3.3. Donde Utilizar MongoDB

Aunque suele decirse que las bases de datos NoSQL tienen un ámbito de aplicación reducido, MongoDB se puede utilizar en muchos de los proyectos que se desarrollan en la actualidad (Fernandez, 2014).

Cualquier aplicación que necesite almacenar datos semi estructurados puede usar MongoDB. Es el caso de las típicas aplicaciones CRUD o de muchos de los desarrollos web actuales.

Eso sí, aunque las colecciones de MongoDB, no necesitan definir un esquema, es importante que se diseñe en nuestra aplicación para seguir un comportamiento ordenado. Se debe tener en cuenta si se normalizar los datos, de normalizarlos o utilizar una aproximación híbrida. Estas decisiones pueden afectar al rendimiento de nuestra aplicación. En definitiva el esquema lo definen las consultas que se vayan a realizar con más frecuencia.

MongoDB es especialmente útil en entornos que requieran escalabilidad. Con sus opciones de replicación y sharding, que son muy sencillas de configurar, se puede conseguir un sistema que escale horizontalmente sin demasiados problemas (Fernandez, 2014).

3.4. Donde No Usar MongoDB

En esta base de datos no existen las transacciones. Aunque la aplicación puede utilizar alguna técnica para simular las transacciones, MongoDB no tiene esta capacidad. Solo garantiza operaciones atómicas a nivel de documento. Si las transacciones son algo indispensable en el desarrollo, se deberá pensar en otro sistema (Fernandez, 2014).

Tampoco existen los JOINS. Para consultar datos relacionados en dos o más colecciones, se tiene que hacer más de una consulta. En general, si los datos

pueden ser estructurados en tablas, y utilizar las relaciones, es mejor que optemos por un RDBMS clásico.

Para almacenar los documentos, MongoDB utiliza una serialización binaria de JSON, llamada BSON, que es una lista ordenada de elementos simples. El núcleo de la base de datos es capaz de interpretar su contenido, de modo que lo que a simple vista parece un contenido binario, realmente es un documento que contiene varios elementos. Estos datos están limitados a un tamaño máximo de 16 MB; para tamaños superiores se requiere del uso de GridFS (Requena, 2010).

MongoDB al ser una base de datos no relacional posee características principales. Es decir que las consultas son a nivel de colecciones (sin joins), los índices se generan en las colecciones, los documentos tendrán un ID única y su atomicidad es a nivel de documentos son de tipo JSON como esquema dinámico, haciendo que la integración de datos en ciertas aplicaciones sea más fácil y rápida (Aulet, 2011).

4. Migración de SQL a NoSQL

Pueden existir infinidad de motivos por los cuales se desea migrar datos, entre los más comunes se encuentran la necesidad de mover los datos de un origen a otro dada por el cambio de plataforma, las dificultades que representa el trabajar con grandes cantidades de información, la estructura actual sobre la que se soportan los datos no es lo suficientemente extensa, la infraestructura con que cuenta la organización para almacenar su información no es suficiente, inclusión de campos para almacenamiento de tipos de datos no soportados por la estructura de base de datos actual, nuevos requerimientos del software que alimenta de datos a la base de datos, nuevas necesidades de los usuarios finales, y así un sin fin de argumentos totalmente válidos para llevar a cabo esta fase llámese de mantenimiento o actualización. Como en toda migración, hay puntos importantes que necesitan analizarse, entre los cuales se pueden encontrar: el volumen de la información a migrar, los tipos de datos a migrar, se debe también conocer la estructura de la base de datos actual con el fin de empatar campos y hacer las conversiones correspondientes antes de realizar alguna acción que pueda provocar pérdida de datos o que la migración simplemente no concluir de manera correcta por la generación de errores incontrolados. Los manejadores de base de datos así como el lenguaje que utilizan son pieza clave en este proceso. Para no quedar fuera de contexto se debe conocer a lo que se enfrenta. Las sentencias SQL a las que se están acostumbrados quedaran un poco de lado, sin embargo citamos algunas comparativas para estar familiarizados con el tema, ver Tabla 1: Diferencias entre SQL y MongoDB. (Manjarrez Antaño, Martínez Castro, & Cuevas Valencia, 2014)

Diferencias Entre SQL y MongoDB	
SQL	MongoDB

Data Base	Data Base
Table	Collection
Row	Document
Column	Field
Index	Index
Join	-
Primary Key	Primary Key (_id)

Tabla 1: Diferencias entre SQL y MongoDB

MYSQL	MONGODB
	

Tabla 2: Estructura de datos MySQL (Relacional) y MongoDB (No Relacional) (Solvetic, 2014)

SQL	MONGODB
SELECT user_id, status FROM users WHERE status = "A"	db.users.find({ status: "A" }, { user_id: 1, status: 1, _id: 0 })
SELECT * FROM users WHERE status != "A"	db.users.find({ status: { \$ne: "A" } }))
SELECT * FROM users WHERE status = "A" AND age = 50	db.users.find({ status: "A", age: 50 })
SELECT * FROM users WHERE status = "A" OR age = 50	db.users.find({ \$or: [{ status: "A" }, { age: 50 }] }))
SELECT * FROM users WHERE age > 25	db.users.find({ age: { \$gt: 25 } }))
SELECT *	db.users.find(

SQL	MONGODB
FROM users WHERE age < 25	{ age: { \$lt: 25 } })
SELECT * FROM users WHERE user_id like "%bc%"	db.users.find({ user_id: /bc/ })
SELECT * FROM users WHERE user_id like "bc%"	db.users.find({ user_id: /^bc/ })
SELECT * FROM users WHERE status = "A" ORDER BY user_id ASC	db.users.find({ status: "A" }).sort({ user_id: 1 })
SELECT * FROM users WHERE status = "A" ORDER BY user_id DESC	db.users.find({ status: "A" }).sort({ user_id: -1 })
SELECT COUNT(*) FROM users	db.users.count() or db.users.find().count()
SELECT COUNT(user_id) FROM users	db.users.count({ user_id: { \$exists: true } }) or db.users.find({ user_id: { \$exists: true } }).count()
SELECT COUNT(*) FROM users WHERE age > 30	db.users.count({ age: { \$gt: 30 } }) or db.users.find({ age: { \$gt: 30 } }).count()
SELECT * FROM users LIMIT 1	db.users.findOne() or db.users.find().limit(1)
SELECT * FROM users LIMIT 5 SKIP 10	db.users.find().limit(5).skip(10)
EXPLAIN SELECT * FROM users WHERE status = "A"	db.users.find({ status: "A" }).explain()

Tabla 3: Sentencias básicas lenguaje de consultas MongoDB (Solvetic, 2014)

Una vez que se tenga claro dónde se está, hacia dónde se quiere ir y, sabiendo que los datos son la base de toda información requerida para la toma de decisiones, no

se pasará de una estructura a otra por simple vanidad, si bien las bases de datos NoSQL ofrecen gran capacidad de escalabilidad, espacio para almacenamiento, flexibilidad, entre otras, no se debe dejar de lado las posibles necesidades de tener una estructura fija que permita realizar la manipulación segura consciente de la información.

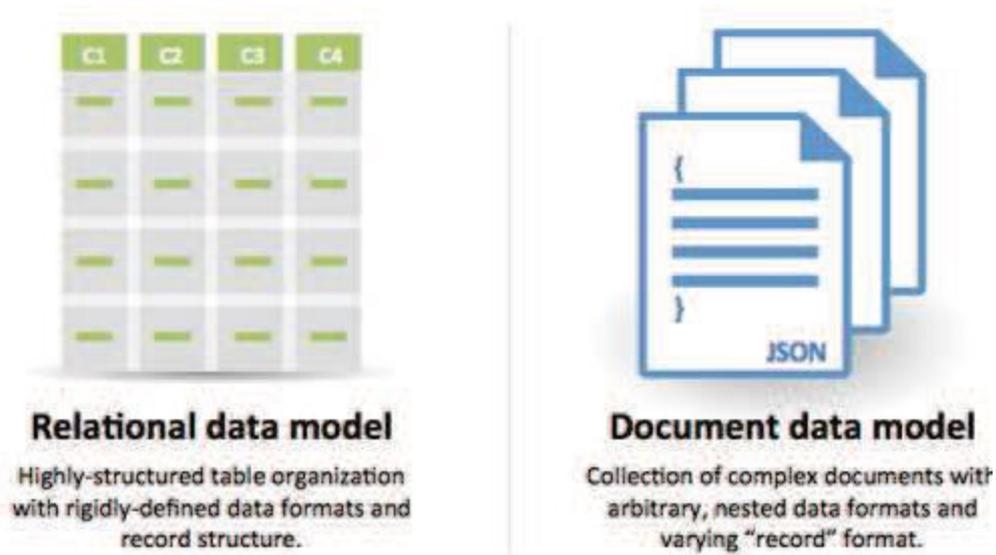


Tabla 4: Modelos relacional vs orientado a documentos (Manjarrez Antaño, Martínez Castro, & Cuevas Valencia, 2014)

Cada registro en una base de datos relacional se ajusta a un esquema - con un número fijo de campos (columnas) cada uno con un propósito especificado y tipo de datos. La ventaja es que hay menos datos duplicados en la base de datos. La desventaja es que un cambio en el esquema significa realizar varias declaraciones "alter table".

Con las bases de datos de documentos en el otro lado, cada documento puede tener una estructura completamente diferente de otros documentos.

No se requiere ninguna gestión adicional en la base de datos para controlar los cambios de esquemas de documentos.

MongoDB guarda estructuras de datos en documentos tipo JSON con un esquema dinámico (MongoDB llama ese formato BSON), haciendo que la integración de los datos en ciertas aplicaciones sea más fácil y rápida.

Entonces, teniendo una colección de datos CSV o XML pueden convertirse al formato JSON, a partir de este último se puede generar las sentencias correspondientes para realizar inserciones de datos en nuestro nuevo entorno. (Manjarrez Antaño, Martínez Castro, & Cuevas Valencia, 2014)

5. Zend Framework

Zend Framework considerado open source para PHP5 desarrollado por Zend, empresa encargada de la mayor parte de las mejoras hechas a PHP, por lo que se podría decir que es el framework "oficial".

Está diseñado para ser liviano, modular y extensible, posee un diseño minimalista a de más de permite libertad, proporcionando estructuras, convenciones y layouts de códigos reutilizables. (Espinoza Aguirre, 2012)

5.1. Características

- Totalmente orientado a objetos.
- Soporta la internalización y localización de aplicaciones.
- Facilita el setup de los proyecto mediante línea de comandos.
- Componentes con bajo acoplamiento, por lo que los puedes usar en forma independiente.
- Posee Zend_Test para facilitar el testing de nuestra aplicación.
- Programación estandarizada.
- Soporta adapters para múltiples tipos de bases de datos.
- Componentes para la autenticación y autorización de usuarios, envío de mails, cache en varios formatos, creación de web services, etc.

5.2. Estructura Zend

Zend posee una estructura de archivos, de las cuales las carpetas principales son:

- **Application:** Crea subcarpetas que contendrán los modelos, vistas y controladores de nuestra aplicación.
- **Public:** Almacena imágenes, scripts y archivos CSS.
- **Library:** Contiene archivos descargados del Zend Framework y otras librerías propias.
- **Test:** Su objetivo es hacer pruebas fáciles de leer, escribir y depurar.
- **.zfproject.xml:** Contendrá su estado de estructura de la aplicación en formato XML, al ser generadas ya sea mediante la consola.

5.3. Desarrollo con Zend

Para comenzar a utilizar:

- **htaccess:** ubicado en public, redirige todas las peticiones al index.php.
- **index.php:** Decide que página mostrar haciendo referencia a un archivo application.ini.
- **application.ini:** el cual contiene la configuración del sitio.

- **Bootstrap:** registrando en el autoloader el namespace de nuestra aplicación para poder instanciar las clases que usemos sin necesidad de hacer antes un include de dicho archivo.

5.4. Requerimientos

Los requisitos para implementar Zend Framework son los siguientes:

- PHP 5.1.4 (o más actual).
- Un servidor Web que soporte funcionalidades de mod_rewrite del apache ubicado, en el archivo httpd.conf dentro de la línea #LoadModule rewrite_module, para desactivarlo se debe borrar el numeral al inicio de dicha fila.
- Apache debe estar configurado para soportar archivos .htaccess, en el caso de no estar configurado se cambia la configuración de AllowOverride None aAllowOverride All.

6. Web Services Description Language

El Web Services Description Language (WSDL) es un formato XML para describir servicios de red como un conjunto de puntos finales que operan en los mensajes que contienen información orientado al procedimiento orientado al documento. Las operaciones y mensajes se describen de forma abstracta, y luego unidos a un formato de protocolo de red y el mensaje de hormigón para definir un punto final. Criterios de valoración concretos relacionados se combinan en los puntos finales abstractos (servicios). WSDL es extensible para permitir la descripción de los puntos finales y sus mensajes, independientemente de qué mensaje formatos o protocolos de red se utilizan para comunicarse, sin embargo, los únicos enlaces que se describen en este documento se describe cómo utilizar WSDL junto con SOAP 1.1, HTTP GET / POST, y MIME. (Christensen, Curbera, Merendith, & Weerawarana, 2001)

Como los protocolos de comunicación y formatos de mensaje están estandarizados en la comunidad web, se hace cada vez más posible e importante para ser capaz de describir las comunicaciones de alguna manera estructurada. WSDL aborda esta necesidad mediante la definición de una gramática XML para describir servicios de red como colecciones de puntos finales de comunicación capaces de intercambiar mensajes.

Un documento WSDL define **los servicios** como las colecciones de los puntos finales de la red, o puertos. En WSDL, la definición abstracta de los puntos finales y los mensajes se separa de su despliegue de red o de formato de datos consolidaciones concretas. Esto permite la reutilización de definiciones abstractas: mensajes, que son descripciones abstractas de los datos que se intercambian, y tipos de puertos que son colecciones abstractas de operaciones. Las especificaciones de los protocolos y formatos de datos concretos para un tipo de puerto en particular constituyen un reutilizable vinculante. Un puerto se define mediante la asociación de una dirección de

red con una unión reutilizable, y una colección de puertos definir un servicio. Por lo tanto, un documento WSDL utiliza los siguientes elementos en la definición de los servicios de red: (Christensen, Curbera, Merendith, & Weerawarana, 2001)

- Tipos - un recipiente para las definiciones de tipos de datos que usan algún tipo de sistema (como XSD).
- Mensaje - una definición abstracta mecanografiada de los datos que se comunica.
- Operación - una descripción abstracta de una acción apoyada por el servicio.
- Tipo de puerto - un conjunto abstracto de las operaciones de apoyo de uno o más puntos finales.
- Encuadernación - un protocolo concreto y especificación de formato de datos para un tipo determinado puerto.
- Puerto - un único punto final definido como una combinación de una unión y una dirección de red.
- Servicio - una colección de puntos finales relacionados.

7. BPMN (Business Process Model and Notation)

La notación para el modelado de procesos de negocio (Business Process Model And Notation – BPMN por sus siglas en ingles), es una forma estándar y gráfica de modelar procesos de negocios.

La meta fundamental de BPMN es proporcionar una notación estándar que sea fácilmente comprensible por todos los Stakeholders.

Provee una notación simple para los flujos, independiente del entorno de implementación. La notación se sustenta en un marco riguroso que facilita trasladar los modelos de nivel de negocio hacia modelos ejecutables que las suites de BPM y motores Workflow puedan comprender. En los últimos años, BPMN ha sido ampliamente adoptado por los productos relacionados a la Gestión de Procesos de Negocios (BPM - Business Process Management), tanto para los fabricantes de herramientas de Análisis de Procesos de Negocios (BPA - Business Process Analysis), como por los de herramientas de Modelado y Suites completas de BPM.

Business Process Model And Notation, BPMN proporciona un lenguaje común para que las partes involucradas puedan comunicar los procesos de forma clara, completa y eficiente. De esta forma BPMN define la notación y semántica de un Diagrama de Procesos de Negocio (Business Process Diagram, BPD) para luego ser capaz de automatizarlos (exportándolo en formato XPD).

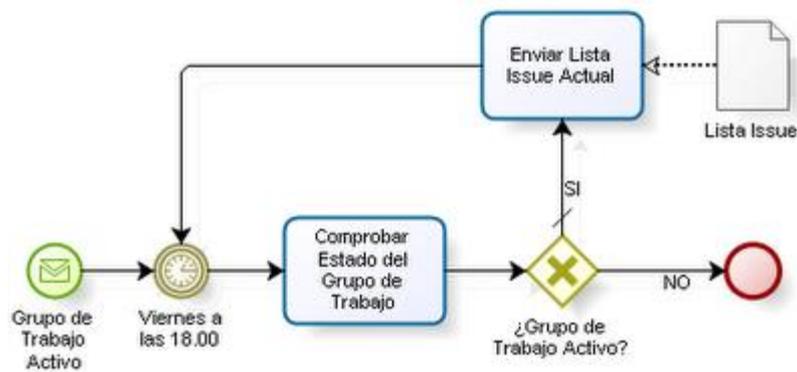


Ilustración 3: Diagrama de Procesos de Negocio (Sebastián Bayard Ocares , 2011)

Business Process Modeling And Notation (BPMN), es una notación gráfica que describe la lógica de los pasos de un proceso de negocio. Esta notación plasma gráficamente el diseño de los procesos y su implementación en la práctica. Su sintaxis está basada en elementos gráficos, pero tales elementos tienen una relación con instrucciones en el lenguaje BPEL, que permite generar código BPEL viable desde un modelo BPMN. (Sebastián Bayard Ocares , 2011)

Business Process Modeling Notation (BPMN) proporciona un lenguaje común para que las partes involucradas puedan comunicar los procesos de forma clara, completa y eficiente. De esta forma BPMN define la notación y semántica de un Diagrama de Procesos de Negocio (Business Process Diagram, BPD). BPD es un Diagrama diseñado para representar gráficamente la secuencia de todas las actividades que ocurren durante un proceso, basado en la técnica de “Flow Chart”, incluye además toda la información que se considera necesaria para el análisis (BIZAGI).

Para introducir al tema de BPMN, a lo largo de este documento el lector se encontrará con una serie de ejemplos desarrollados en torno a un proceso de Solicitud de Crédito de Consumo. Un proceso de crédito consta básicamente de un Registro de la solicitud, donde el cliente manifiesta su interés de adquirir un crédito, en esta etapa se incluye la presentación de la solicitud y documentación requerida a la entidad, luego se realiza una verificación de la información, posteriormente la etapa donde se realiza el Análisis o Estudio de la solicitud de crédito y por último encontramos las actividades referentes a hacer efectivo el crédito o informar el rechazo al cliente.

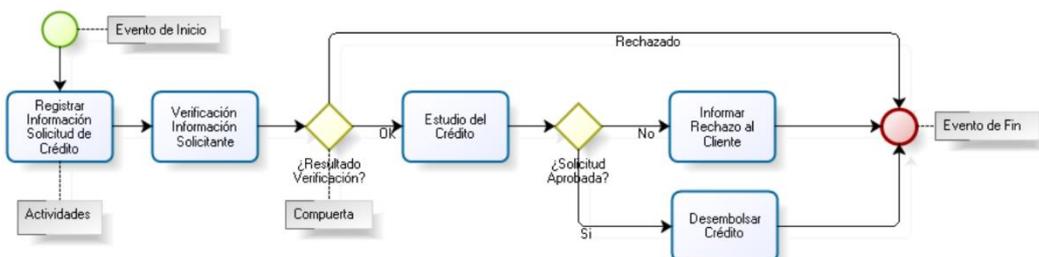


Ilustración 4: Ejemplo de Modelado de Procesos de Negocio

Como puede observar en el ejemplo anterior, dentro de un Diagrama de procesos de negocio existe un conjunto de elementos gráficos que nos permiten representar un proceso de negocio. En el ejemplo anterior se pueden visualizar diferentes tipos de elementos que describen el comportamiento del proceso, dentro de estos elementos encontramos las actividades que representan el trabajo realizado, los eventos de inicio y de fin que indican el inicio y el fin del proceso y los elementos de decisión conocidos en BPMN como Compuertas que indican una división en el camino. Dichos elementos se encuentran conectados por líneas de secuencia, que muestran cómo fluye el proceso.

Al principio del proceso de Solicitud de Crédito está graficada la figura “evento de inicio” indicando el comienzo del proceso. Los procesos pueden iniciar de distintas formas, BPMN provee diferentes tipos de eventos de inicio (simple, mensaje, señal, entre otros) Al final de este proceso se encuentra la figura “Evento de fin Terminal”, indicando la terminación del proceso, como puede observar el proceso se termina cuando el solicitante fue rechazado, la solicitud de crédito no fue aprobada o ya se realizó el desembolso del crédito.

La compuerta utilizada dentro del ejemplo anterior es la compuerta exclusiva, esta compuerta como elemento de decisión se comporta como un “XOR”, es decir, de varias alternativas presentadas solo una de ellas puede ser tomada. Dentro del proceso de solicitud de crédito podemos observar dos ejemplos del uso de la compuerta exclusiva, en el primero dependiendo del resultado de la verificación de la información del solicitante el flujo tomaría un camino o el otro, si el resultado fue “Solicitante rechazado” el proceso se terminaría y si el solicitante fue aceptado se continúa con el proceso. En el segundo ejemplo la decisión se basa en el resultado del estudio del crédito, ya que si la solicitud fue rechazada se le informa al cliente y si fue aprobada se procede a realizar el desembolso.

Si analizamos el proceso de solicitud de crédito, podemos ver que existen actividades que pueden analizarse en más detalle, una de estas actividades es la verificación de la Información suministrada por el solicitante, ya que normalmente las entidades que otorgan créditos realizan varios análisis al solicitante, por ejemplo se verifica si el solicitante ya es un cliente de la entidad, si es un cliente objetivo del banco o por el contrario se encuentra en alguna lista de clientes negativos y posteriormente consultan su situación financiera.

Por lo anterior las actividades pueden ser compuestas o no, dentro de BPMN las actividades compuestas se conocen como Subprocesos, y las actividades atómicas como tareas.

El diagrama de flujo del proceso de solicitud de Crédito quedaría de la siguiente manera al incluir la actividad de verificación de información como un subproceso.

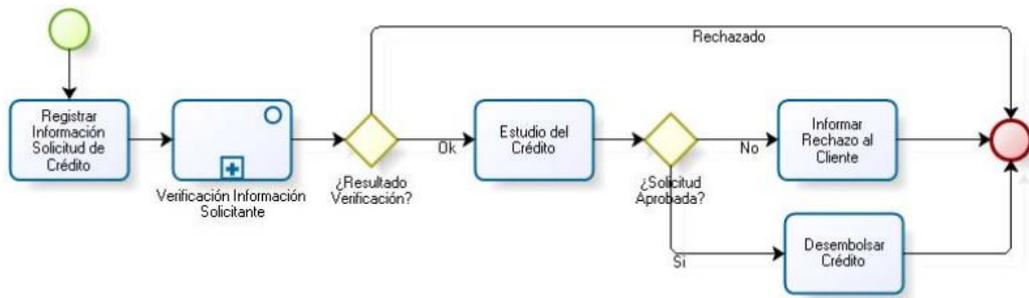


Ilustración 5: Modelo del Proceso de Solicitud de Crédito (BIZAGI)

El subproceso de verificación de Información del solicitante es el siguiente.

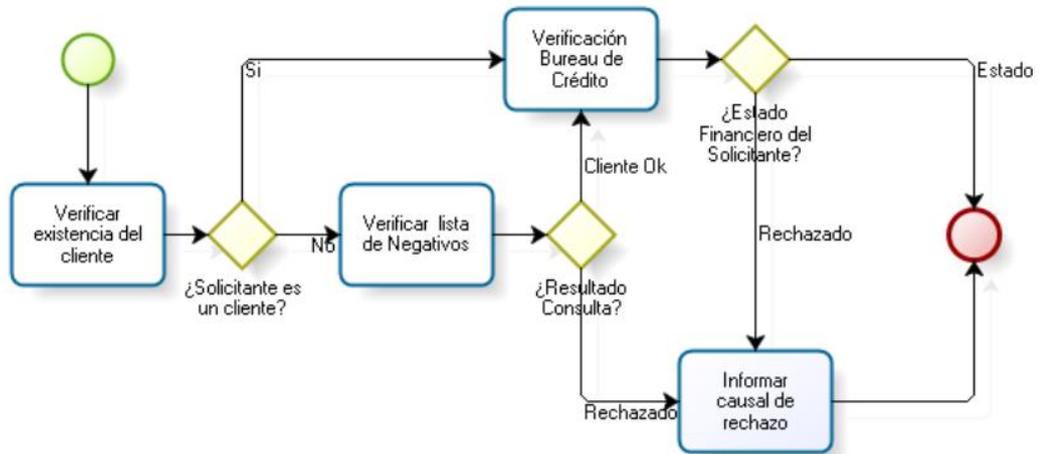


Ilustración 6: Subproceso de Verificación de Información del Cliente (BIZAGI)

Como hemos visto dentro de los ejemplos anteriores hemos utilizado algunos elementos Gráficos de BPMN, estos elementos en BPMN se encuentran clasificados dentro de 4 categorías.

7.1. Objetos de Flujo

Son los principales elementos gráficos que definen el comportamiento de los procesos. Dentro de los Objetos de Flujo encontramos:

7.1.1. Eventos

Son algo que sucede durante el curso de un proceso de negocio, afectan el flujo del proceso y usualmente tienen una causa y un resultado. Dentro de los anteriores ejemplos utilizamos inicio, fin y temporizador, estos elementos son eventos y a su vez se encuentran clasificados en 3 tipos.

Eventos de Inicio	
-------------------	--

Eventos Intermedios	
Eventos de Fin	

Tabla 5: Tipos de Eventos BPMN

7.1.2. Actividades

Estas Representan el trabajo que es ejecutado dentro de un proceso de negocio. Las actividades pueden ser compuestas o no, por lo que dentro de los ejemplos utilizamos los dos tipos de actividades existentes:

Tareas	
Subprocesos	

Tabla 6: Tipos de Actividades BPMN

7.1.3. Compuertas

Son elementos del modelado que se utilizan para controlar la divergencia y la convergencia del flujo. Existen 5 tipos de compuertas, dentro de los ejemplos desarrollados en este documento pudimos ver el uso de la compuerta Exclusiva (XOR) y de la compuerta Inclusiva. Los 5 tipos de compuertas son:

- Compuerta Exclusiva.
- Compuerta Basada en Eventos.
- Compuerta Paralela.
- Compuerta Inclusiva.
- Compuerta Compleja.

8. Metodología de Desarrollo ICONIX

Es una metodología de desarrollo de software simple y práctico, ya que se encuentra entre la complejidad de RUP (Rational Unified Process) porque está dirigido por casos de uso y la simplicidad de eXtreme Programming (XP) donde se captura requerimientos, diseña y repite estos pasos de forma iterativa. (Salazar Paredes & Sanchez Macas, 2013)

ICONIX es un proceso simplificado en comparación con otros procesos más tradicionales, que unifica un conjunto de métodos de orientación a objetos con el objetivo de abarcar todo el ciclo de vida de un proyecto. Fue elaborado por Doug Rosenberg y Kendall Scott a partir de una síntesis del proceso unificado de los “tres amigos” Booch,

Rumbaugh y Jacobson y que ha dado soporte y conocimiento a la metodología ICONIX desde 1993. Presenta claramente las actividades de cada fase y exhibe una secuencia de pasos que deben ser seguidos. Además ICONIX está adaptado a los patrones y ofrece el soporte de UML, dirigido por casos de uso y es un proceso iterativo e incremental (De San Martin Oliva, 2011).

Las tres características fundamentales de ICONIX son:

- Iterativo e incremental: varias iteraciones ocurren entre el desarrollo del modelo del dominio y la identificación de los casos de uso. El modelo estático es incrementalmente refinado por los modelos dinámicos.
- Trazabilidad: cada paso está referenciado por algún requisito. Se define trazabilidad como la capacidad de seguir una relación entre los diferentes artefactos producidos.
- Dinámica del UML: La metodología ofrece un uso “dinámico del UML” como los diagramas del caso de uso, diagramas de secuencia y de colaboración.

Rosenberg y Scoot destacan un análisis de requisitos, un análisis y diseño preliminar, un diseño y una implementación como las principales tareas (De San Martin Oliva, 2011).

ICONIX es una metodología compuesta por 4 fases:

- Fase 1: Análisis y requerimientos.
- Fase 2: Diseño Preliminar.
- Fase 3: Diseño detallado.
- Fase 4: Implementación.

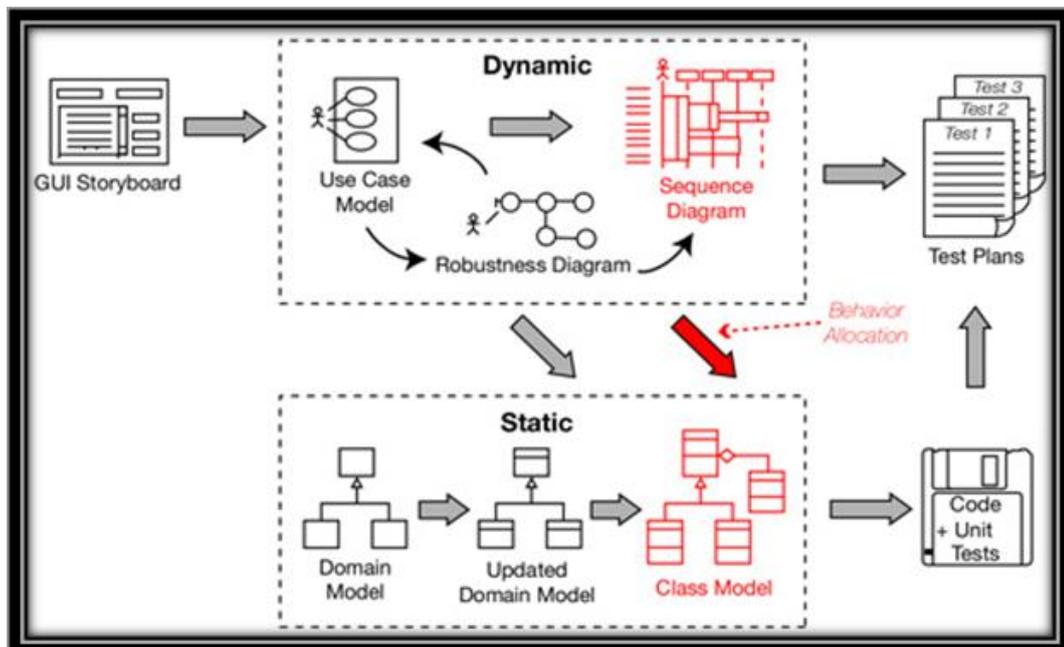


Ilustración 7: Resumen de la Metodología (García, 2011)

8.1. Fase 1: Análisis y Requerimientos

Esta fase contiene 4 subfases, que cumplen una función específica dentro de la etapa de requerimientos:

- a. Identificar en el “mundo real” los objetos y todas las relaciones de agregación y generalización entre ellos. Utilizar un diagrama de clases de alto nivel definido como modelo de dominio.

El trabajo es iniciado con un relevamiento informal de todos los requisitos que en principio deberían ser parte del sistema. Luego con los requisitos se construye el diagrama de clases, que representa las agrupaciones funcionales con que se estructura el sistema que se desarrolla.

De generarse el sistema a este nivel de especificación, se obtendría el menú principal del sistema con la interfaces iniciales de los casos o actividades de cada división funcional. Los diagramas del segundo nivel o superior, accesibles a partir de cada escenario o estado del nivel anterior, representan los casos, actividades y secuencias de interacción de cada división funcional.

En estos se pueden reutilizar interfaces ya definidas en otros diagramas, representándose con bordes tenues.

- b. Presentar, si es posible, una prototipación rápida de las interfaces del sistema, los diagramas de navegación, etc., de forma que los clientes puedan comprender mejor el sistema propuesto.

Con el prototipo se espera que las especificaciones iniciales estén incompletas. En general se necesita entre 2 y 3 reuniones para establecer las especificaciones iniciales. La rapidez con la que se genera el sistema es esencial para que no se pierda el estado de ánimo sobre el proyecto y que los usuarios puedan comenzar a evaluar la aplicación en la mayor brevedad posible (De San Martin Oliva, 2011).

Durante la evaluación se debe capturar información sobre lo que les gusta y lo que les desagrada a los usuarios, al mismo tiempo poner atención al porque reaccionan los usuarios en la forma en que lo hacen.

Los cambios al prototipo son planificados con los usuarios antes de llevarlos a cabo.

El proceso se repite varias veces y finaliza cuando los usuarios y analistas están de acuerdo en que el sistema ha evolucionado lo suficiente como para incluir todas las características necesarias o cuando es evidente que no se obtendrá mayor beneficio con una iteración adicional.

El diseño de prototipos es una técnica popular de ingeniería para desarrollar modelos a escala (o simulados) de un producto o sus componentes. Cuando se aplica al desarrollo de sistemas de información

el diseño de prototipos implica la creación de un modelo o modelos operativos de trabajo de un sistema o subsistema.

Existen cuatro tipos de prototipos:

- Prototipo de viabilidad: para probar la viabilidad de una tecnología específica aplicable a un sistema de información.
- Prototipo de Necesidades: utilizado para “descubrir” las necesidades de contenido de los usuarios con respecto a la empresa.
- Prototipo de Diseño: es el que usa Iconix. Se usa para simular el diseño del sistema de información final. Se centra en la forma y funcionamiento del sistema deseado. Cuando un analista crea un prototipo de diseño, espera que los usuarios evalúen este prototipo, como si formara parte del sistema final. Los usuarios deberían evaluar la facilidad de aprendizaje y manejo del sistema, así como el aspecto de las pantallas y los informes y los procedimientos requeridos para utilizar el sistema. Estos prototipos pueden servir como especificaciones parciales de diseño o evolucionar hacia prototipos de información.
- Prototipo de Implantación: es una extensión de los prototipos de diseño donde el prototipo evoluciona directamente hacia el sistema de producción.

Los prototipos de pantallas también proporcionan una manera de obtener las reacciones de los usuarios hacia la cantidad de información presentada sobre la pantalla de visualización. Tal vez el usuario decida que un diseño en particular es muy denso ya que existen demasiados detalles sobre la pantalla. En otros casos la información sobre la pantalla aunque no es excesiva en el sentido de causar que la pantalla se vuelva densa, tal vez sea mucho mayor que la que un individuo necesita durante todo el tiempo.

Sus ventajas son:

- Los usuarios se hacen participantes más activos en los desarrollos del sistema. Suelen mostrarse más interesados en los prototipos de trabajo que en las especificaciones de diseño.
- La definición de necesidades se simplifica por el hecho de que muchos usuarios finales no comprenden o no son capaces de enumerar detalladamente sus necesidades hasta que ven un prototipo.
- La probabilidad de que los usuarios aprueben un diseño y luego rechacen su implantación se reducirá notablemente.
- Según se dice el diseño mediante prototipos reduce el tiempo de desarrollo, aunque algunos cuestionan este ahorro.

- Los prototipos suelen pasar a las fases de análisis y diseño con demasiada rapidez. Ello empuja al analista a pasar demasiado rápido a la codificación, sin haber comprendido las necesidades y los problemas. Condición deseable en un proceso ágil.
- c. Identificar los casos de uso del sistema mostrando los actores involucrados. Utilizar para representarlo el modelo de casos de uso.
- Los casos de uso describen bajo la forma de acciones y reacciones el comportamiento de un sistema desde el punto de vista de un usuario; permiten definir los límites del sistema y las relaciones entre el sistema y el entorno.
- Un caso de uso es una manera específica de utilizar un sistema. Es la imagen de una funcionalidad del sistema, desencadenada en respuesta a la estimulación de un actor externo.
- El modelo de los casos de uso comprende los actores, el sistema y los propios casos de uso. El conjunto de funcionalidades de un sistema se determina examinando las necesidades funcionales de cada actor.
- Los casos de usos reubican la expresión de las necesidades sobre los usuarios partiendo del punto de vista muy simple que dice que un sistema se construye ante todo para sus usuarios. La estructuración del método se efectúa respecto a las interacciones de una sola categoría de usuarios a la vez; esta partición del conjunto de necesidades reduce considerablemente la complejidad de la determinación de las necesidades.
- Los casos de uso permiten a los usuarios estructurar y articular sus deseos; les obligan a definir la manera como querrían interactuar con el sistema, a precisar que informaciones quieren intercambiar y a describir lo que debe hacerse para obtener el resultado esperado. Los casos de uso concretan el futuro sistema en una formalización próxima al usuario, incluso en ausencia de un sistema a criticar.
- d. Asociar los requisitos funcionales con los casos de uso y con los objetos del dominio (trazabilidad).
- Un importante aspecto de ICONIX es que un requisito se distingue explícitamente de un caso de uso. En este sentido, un caso de uso describe un comportamiento; un requisito describe una regla para el comportamiento. Además, un caso de uso satisface uno o más requisitos funcionales; un requisito funcional puede ser satisfecho por uno o más casos de uso.

8.2. Fase 2: Diseño Preliminar

Compuesto principalmente por la descripción de los casos de uso y el diagrama de robustez con el fin de identificar los objetos y las interacciones entre tales objetos.

Los pasos para completar esta fase de ICONIX son las siguientes:

- a. Describir los casos de uso, como un flujo principal de acciones, pudiendo contener los flujos alternativos y los flujos de excepción. La principal sugerencia de ICONIX, en esta actividad, es que no se debe perder mucho tiempo con la descripción textual.

Debería usarse un estilo consistente que sea adecuado al contexto del proyecto.

- b. Realizar un diagrama de robustez. Se debe ilustrar gráficamente las interacciones entre los objetos participantes de un caso de uso. Este diagrama permite analizar el texto narrativo de cada caso de uso e identificar un conjunto inicial de objetos participantes de cada caso de uso. El análisis de robustez ayuda a identificar los objetos que participaran en cada caso de uso.

Estos objetos que forman parte de los diagramas de robustez se clasifican dentro de los tres tipos siguientes:

- Objetos de interfaz: usados por los actores para comunicarse con el sistema. Son con los que los actores interactúan con el sistema, generalmente como ventanas, pantalla, diálogos y menús.
- Objetos entidad: son objetos del modelo del dominio. Son a menudo tablas y archivos que contiene archivos para la ejecución de dicho caso de uso.
- Objetos de control: es la unión entre la interfaz y los objetos entidad. Sirven como conexión entre los usuarios y los datos. Los controles son “objetos reales” en un diseño, pero usualmente sirven como una especie de oficinista para asegurar que no se olvide ninguna funcionalidad del sistema la cual puede ser requerida por algún caso de uso.

Esta técnica tan simple pero poderosa sirve como interfaz entre el “que” y el “como” de un análisis. Además el análisis de robustez provee de una gran ayuda a saber si las especificaciones del sistema son razonables.

El análisis de robustez facilita el reconocimiento de objetos. Esto es un paso crucial ya que es casi seguro que se olvida algunos objetos durante el modelado del dominio; y de esta manera se podrán identificar antes de que esto cause problemas serios, además sirve para identificar más y mejores clases, antes del desarrollo del diagrama de secuencias.

Las reglas básicas que se deben aplicar al realizar los diagramas de análisis de robustez:

- Actores solo pueden comunicarse con objetos interfaz.
- Las interfaces solo pueden comunicarse con controles y actores.
- Los objetos entidad solo pueden comunicarse con controles.

- Los controles se comunican con interfaces, objetos identidad y con otros controles pero nunca con actores.
 - Tomando en cuenta que los objetos entidad y las interfaces son sustantivos y los controles son verbos. Se pueden enunciar de manera sencilla que los sustantivos nunca se comunican con otros sustantivos, pero los verbos, si pueden comunicarse con otros verbos y a su vez con sustantivos.
- c. Actualizar el diagrama de clases ya definido en el modelo de dominio con las nuevas clases y atributos descubiertas en los diagramas de robustez.

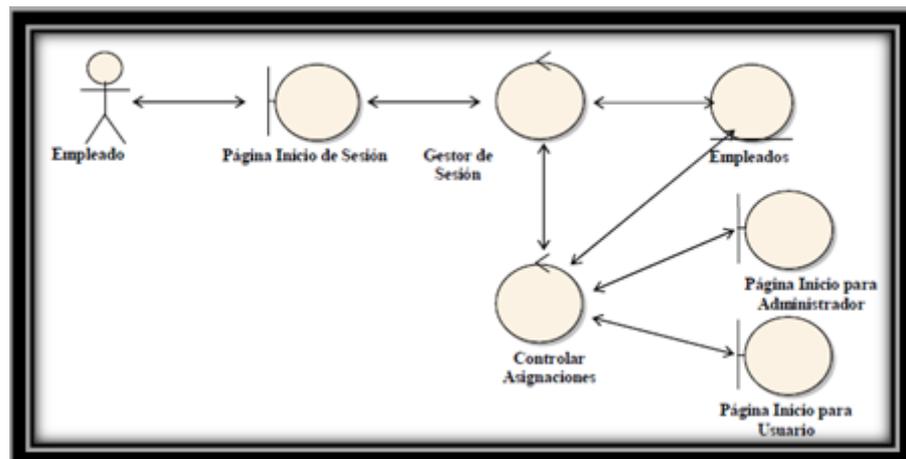


Ilustración 8: Ejemplo de Diagrama de Robustez

8.3. Fase 3: Diseño Detallado

La Fase del Diseño Detallado de ICONIX presenta como principal objetivo la realización de los diagramas de secuencia, donde se podrá identificar los mensajes entre los diferentes objetos. Además esta fase trabaja de forma iterativo sobre las etapas de la fase anterior.

- a. Especificar el comportamiento a través del diagrama de secuencia. Para cada caso de uso identificar los mensajes entre los diferentes objetos. Es necesario utilizar los diagramas de colaboración para representar la interacción entre los objetos.

El diagrama de secuencia muestra interacciones entre objetos según un punto de vista temporal. El contexto de los objetos no se representa de manera explícita como en los diagramas de colaboración. La representación se concentra sobre la expresión de las interacciones.

A pesar de que a partir de los diagramas de casos de uso y de los diagramas de robustez ya tenemos entre un 75 y 80 por ciento de atributos de nuestras clases identificados, es hasta el diagrama de secuencia donde se empiezan a ver que métodos llevarán las clases de nuestro sistema.

Esto se debe a que hasta que vemos interactuando a los objetos de nuestras clases con los actores y con otros objetos de manera dinámica, hasta ese momento tenemos suficiente información como para poder empezar a especificar los métodos de nuestras respectivas clases.

El diagrama de secuencia es el núcleo de nuestro modelo dinámico y muestra todos los cursos alternos que pueden tomar todos nuestros casos de uso. Los diagramas de secuencia se componen de 4 elementos que son: el curso de acción, los objetos, los mensajes y los métodos (operaciones).

- b. Terminar el modelo estático, adicionando los detalles del diseño en el diagrama de clases.
- c. Verificar si el diseño satisface todos los requisitos identificados.

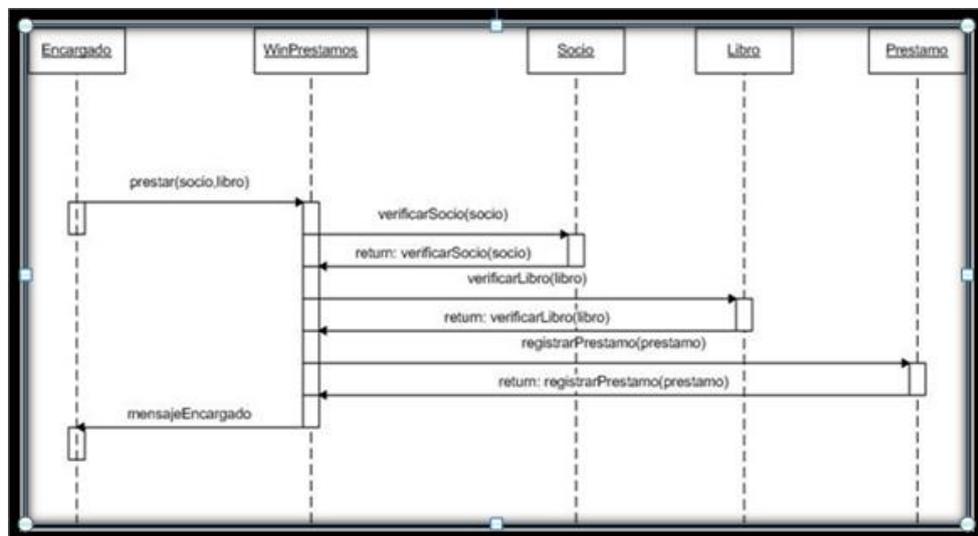


Ilustración 9: Ejemplo de Diagrama de Secuencia

8.4. Fase 4: Implementación

Es la fase final donde se pasa a la parte técnica del proyecto, aplicando el diseño previo del sistema, además de probar la solución con la realización de pruebas.

- a. Utilizar el diagrama de componentes, si fuera necesario para apoyar el desarrollo. Es decir, mostrar la distribución física de los elementos que componen la estructura interna del sistema.

El diagrama de componentes describe los elementos físicos y sus relaciones en el entorno de realización. El diagrama muestra las opciones de realización.

- b. La importancia de la interactividad, accesibilidad y navegación en el software harán que el usuario se sienta seguro y cómodo al poder hacer uso de la aplicación sin inconvenientes tales como son los problemas de

comunicación. Este y otros problemas como la realización de cambios, son factores que deben ser tenidos en cuenta.

Pero además debemos tener en cuenta factores como:

- La Reusabilidad: que es la posibilidad de hacer uso de los componente en diferentes aplicaciones.
 - La Extensibilidad: que consiste en modificar con facilidad el software.
 - La Confiabilidad: realización de sistemas descartando las posibilidades de error.
- c. Realizar pruebas. Test de unidades, de casos, datos y resultados. Test de integración con los usuarios para verificar la aceptación de los resultados.

9. JavaScript Object Notation

JavaScript Object Notation (JSON), en español Notación de Objetos de JavaScript, es un formato ligero de intercambio de datos. Leerlo y escribirlo es simple para humanos, mientras que para las máquinas es simple interpretarlo y generarlo. Está basado en un subconjunto del Lenguaje de Programación JavaScript, Standard ECMA-262 3rd Edition - Diciembre 1999. JSON es un formato de texto que es completamente independiente del lenguaje pero utiliza convenciones que son ampliamente conocidos por los programadores de la familia de lenguajes C, incluyendo C, C++, C#, Java, JavaScript, Perl, Python, y muchos otros. Estas propiedades hacen que JSON sea un lenguaje ideal para el intercambio de datos (ECMA-404 The JSON Data Interchange Standard., 2015). JSON está constituido por dos estructuras:

- Una colección de pares de nombre/valor. En varios lenguajes esto es conocidos como un *objeto*, registro, estructura, diccionario, tabla hash, lista de claves o un arreglo asociativo.
- Una lista ordenada de valores. En la mayoría de los lenguajes, esto se implementa como arreglos, vectores, listas o secuencias.

Estas son estructuras universales; virtualmente todos los lenguajes de programación las soportan de una forma u otra. Es razonable que un formato de intercambio de datos que es independiente del lenguaje de programación se base en estas estructuras.

En JSON, se presentan de estas formas:

Un objeto es un conjunto desordenado de pares nombre/valor. Un objeto comienza con “{” (llave de apertura) y termine con “}” (llave de cierre). Cada nombre es seguido por: (dos puntos) y los pares nombre/valor están separados por “,” (coma) (ECMA-404 The JSON Data Interchange Standard., 2015).

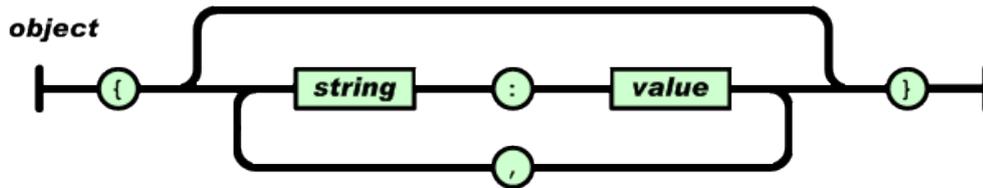


Ilustración 10: Objetos JSON

Un arreglo es una colección de valores. Un arreglo comienza con “[” (corchete izquierdo) y termina con “]” (corchete derecho). Los valores se separan por “,” (coma).

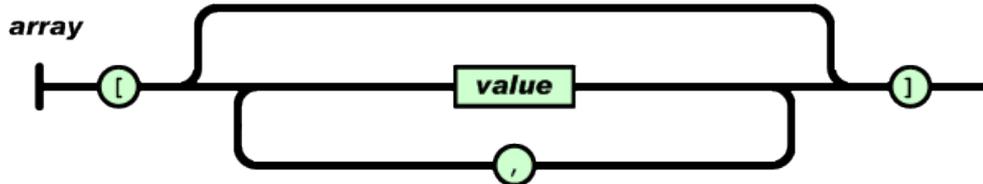


Ilustración 11: Arreglos JSON

Un valor puede ser una cadena de caracteres con comillas dobles, o un número, o true o false o NULL, o un objeto o un arreglo. Estas estructuras pueden anidarse.

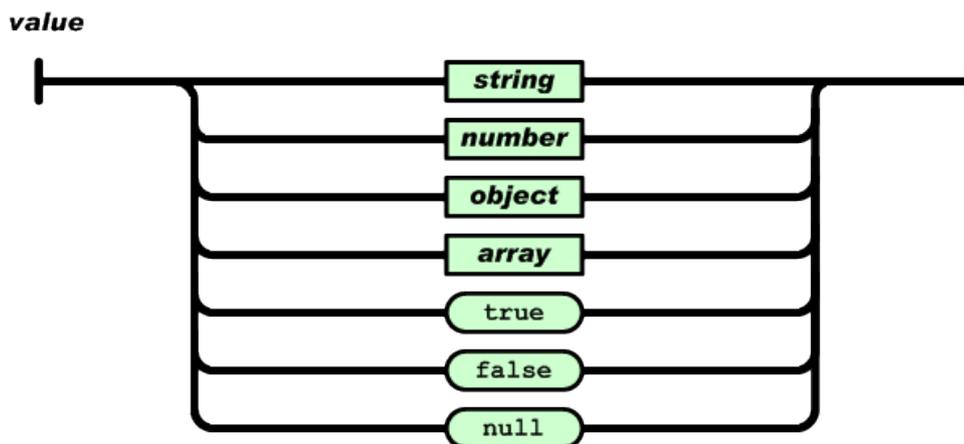


Ilustración 12: Valores JSON

Una cadena de caracteres es una colección de cero o más caracteres Unicode, encerrados entre comillas dobles, usando barras divisorias invertidas como escape. Un carácter está representado por una cadena de caracteres de un único carácter. Una cadena de caracteres es parecida a una cadena de caracteres C o Java.

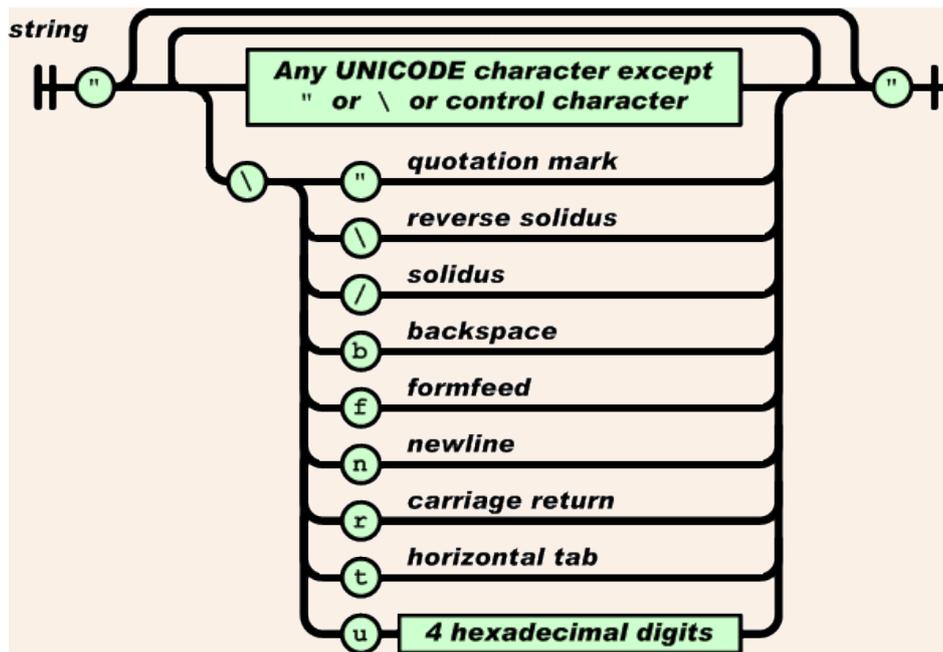


Ilustración 13: Cadenas JSON

Un número es similar a un número C o Java, excepto que no se usan los formatos octales y hexadecimales.

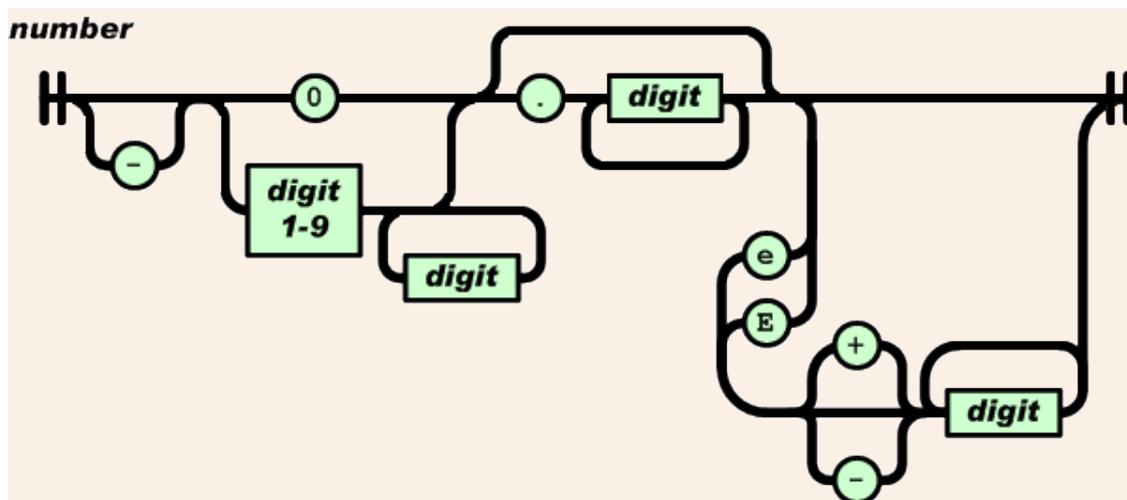


Ilustración 14: Números JSON

10. Methodology

Metodología desarrollada en el laboratorio de Inteligencia Artificial de la Universidad Politécnica de Madrid, que permite la construcción de ontologías a nivel de conocimiento incluyendo la identificación del proceso de desarrollo de la ontología, un ciclo de vida basado en la evolución de prototipos y técnicas particulares para realizar cada actividad (Gomez-Perez, Fernández-López, & Corcho-García, 2004).

Las actividades de control, aseguramiento de calidad, adquisición de conocimiento, integración, evaluación documentación y manejo de configuración se realizan simultáneamente con las actividades de desarrollo.

10.1. Actividades de Desarrollo

10.1.1. Especificación

El propósito de la ontología, sus usos, escenario de uso, usuarios finales, etc.

Nivel de formalidad de la ontología: altamente informal, semi-informal, semi-formal o formal, dependiendo de si los términos y sus significados están codificados entre un lenguaje natural y un lenguaje formal.

Ámbito de aplicación, que incluye el conjunto de términos que se representan, sus características y su granularidad.

En definitiva, un buen documento de especificación debe ser un resumen sin que haya información irrelevante y consistente.

- **Adquisición de conocimientos:** Esta actividad es independiente del proceso de desarrollo de la ontología. Ya que la mayor parte de la adquisición de conocimientos se realiza durante la fase de especificación. Consiste en realizar distintas técnicas como intercambio de ideas, entrevistas, análisis de textos, etc. Para conseguir una idea clara de los efectos de la ontología.

10.1.2. Conceptualización

Consiste en organizar y convertir una percepción informal de un dominio en una especificación semi-formal usando un conjunto de representaciones intermedias (tablas, diagramas) que puedan ser entendidas por los expertos del dominio y los desarrolladores de ontologías.

10.1.3. Formalización

Realizar la transformación del modelo conceptual en un modelo formal o semi computables.

10.1.4. Implementación

Realizar la codificación de la ontología utilizando un lenguaje formal (Ontolingua, XOL, OIL, DAML, OWL, entre otros).

10.1.5. Mantenimiento

Esta actividad permite la actualización y corrección de la ontología.

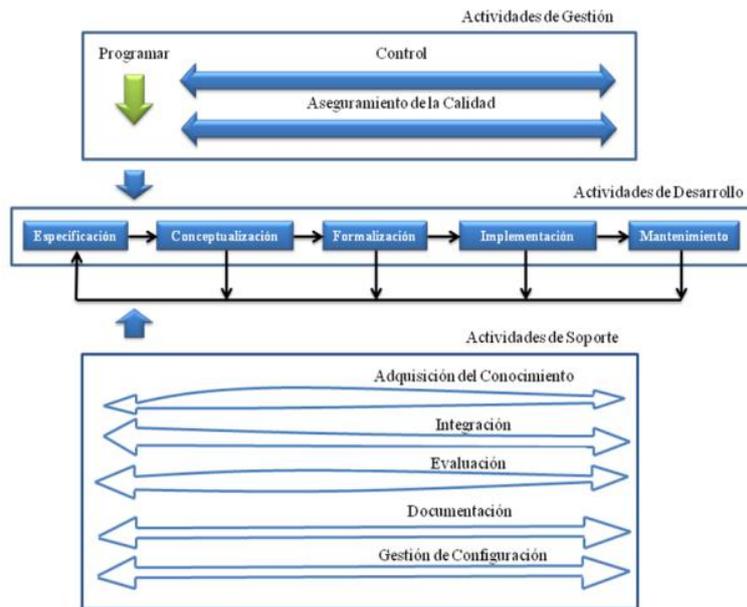


Ilustración 15: Diagrama del método Methontology

10.2. Tareas de Conceptualización

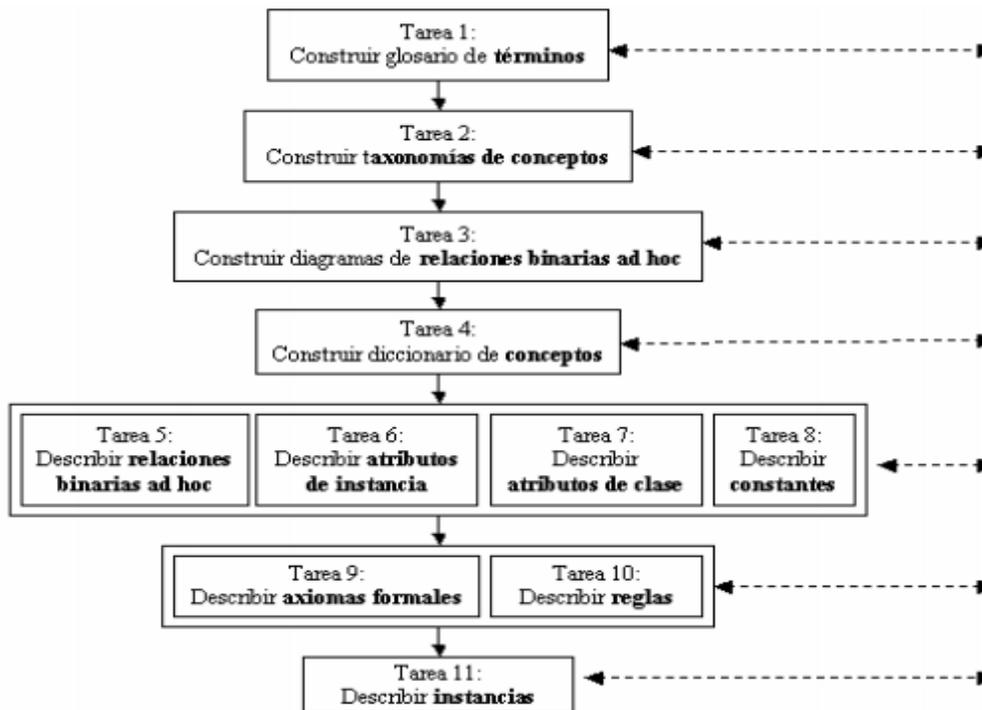


Ilustración 16: Tareas de Conceptualización de Methontology

10.2.1. Tarea 1

Construir el glosario de términos. El glosario de términos debe incluir todos los términos relevantes del dominio (conceptos, instancias, atributos, relaciones entre conceptos, etc.), sus descripciones en lenguaje natural, sus sinónimos y acrónimos. Es importante mencionar, que al empezar pueden existir varios

términos que se refieren al mismo concepto los cuales deben ser identificados y colocados como sinónimos.

10.2.2. Tarea 2

Construir la taxonomía de conceptos. Cuando el glosario de términos tenga una cantidad importante de elementos, se construye una taxonomía que defina la jerarquía entre conceptos. Para realizar dicha taxonomía, se debe seleccionar los términos clasificados como conceptos en el glosario de términos y construir la jerarquía.

10.2.3. Tarea 3

Construir un diagrama de relaciones binarias. El objetivo de este diagrama es establecer las relaciones entre los conceptos de una o más taxonomías de conceptos. Se debe evaluar que el diagrama creado no contenga errores.

10.2.4. Tarea 4

Construir el diccionario de conceptos. El diccionario de conceptos contiene los conceptos del dominio, sus relaciones, instancias, atributos de clases y atributos de instancias. Las relaciones, atributos de instancias, y atributos de clases son locales al concepto, lo que significa que sus nombres pueden repetirse en diferentes conceptos.

10.2.5. Tarea 5

Describir las relaciones binarias en detalle. Se crea la tabla de relaciones binarias en la que se describe detalladamente todas las relaciones binarias incluidas en el diccionario de conceptos. Para cada relación binaria se debe especificar: nombre, conceptos fuente y destino, cardinalidad y relación inversa.

10.2.6. Tarea 6

Describir los atributos de instancia en detalle. Se crea la tabla de atributos de instancias en la que se describe detalladamente todos los atributos de instancias incluidos en el diccionario de conceptos. Los atributos de instancias son aquellos atributos que describen las instancias de un concepto, y sus valores pueden ser diferentes para cada instancia del concepto. Para cada atributo de instancia, se debe especificar: nombre, concepto al que pertenece, tipo de valor, rango de valores (en el caso de valores numéricos) y cardinalidad.

10.2.7. Tarea 7

Describir los atributos de clases en detalle. Se crea la tabla de atributos de clases en la que se describe detalladamente todos los atributos de clases incluidos en el diccionario de conceptos. Para cada atributo de clase, se debe especificar: nombre, concepto donde es definido, tipo de valor, valor y cardinalidad.

10.2.8. Tarea 8

Describir las constantes en detalle. Se crea la tabla de constantes en la que se describe detalladamente cada una de las constantes definidas en el glosario de términos. Para cada constante, se debe especificar: nombre, tipo de valor, valor y unidad de medida (para constantes numéricas).

10.2.9. Tarea 9

Describir los axiomas formales. Se deben identificar los axiomas formales necesarios en la ontología y describirlos con precisión en una tabla. Para cada definición de axioma formal de debe especificar: nombre, descripción, expresión lógica que formalmente lo describe (preferiblemente utilizando lógica de primer grado), los conceptos, atributos y relaciones binarias a las cuales el axioma hace referencia y las variables utilizadas.

10.2.10. Tarea 10

Definir las reglas. Se deben identificar cuáles reglas son necesarias en la Ontología y describirlas en una tabla de reglas. Para cada regla, se debe especificar: nombre, descripción, expresión que formalmente la describe, los conceptos, los atributos y las relaciones a los que hace referencia y las variables usadas en la expresión. Para la especificación de las reglas se sugiere la forma: Si <condiciones> entonces <consecuencias o acciones>.

10.2.11. Tarea 11

Describir las instancias. Una vez que el modelo conceptual de la Ontología ha sido creado, se deben definir las instancias relevantes que aparecen en el diccionario de conceptos en una tabla de instancias.

11. Expedientes Judiciales

Un proceso judicial está definido por una discordancia entre dos partes: persona o institución que inicia el proceso y la persona o institución sobre la que se inicia el proceso (generalmente denominados demandante y demandado, respectivamente). Dicha causa será resuelta por un tercer integrante, llamado juez, que se encargará de discernir cuál

de las dos partes se fundamenta con mayor base en el estado de derecho y partir de ello dará un fallo.

Los procesos judiciales están clasificados por materias, entre las principales se encuentran la civil, penal, laboral, de familia y comercial. Si bien es cierto cada materia tiene sus propias características que las distinguen del resto (por ejemplo, las partes se denominan demandante - demandado en un proceso civil y agraviado - inculpado en un proceso penal), todas comparten ciertos principios comunes y se basan en la materia civil.

Aunque un proceso puede presentar diversas variantes a lo largo de su desarrollo, en general su esencia se mantiene. No obstante, se debe tener en cuenta las etapas del proceso diferenciadas por cada uno de los órganos jurisdiccionales (aquellos que tienen la misión de administrar justicia, es decir, de juzgar y hacer ejecutar lo juzgado). Dichas etapas se denominan instancias y los órganos jurisdiccionales son los siguientes: (Sedano, 2010)

- Juzgados de Paz (No Letrados y Letrados): el menor nivel jerárquico en que se encuentra organizado el Poder Judicial.
- Juzgados de Primera Instancia (Especializados o Mixtos): el segundo nivel jerárquico en que se encuentra organizado el Poder Judicial del Perú.
- Cortes Superiores: el tercer nivel jerárquico en que se organiza el Poder Judicial. Sólo se encuentran bajo la autoridad de la Corte Suprema de la República y es, en la mayoría de procesos, el último organismo que conoce un proceso.
- Corte Suprema: máximo órgano jurisdiccional del Perú. Su competencia se extiende a todo el territorio del país.
- Mediante las instancias se puede realizar una impugnación a un fallo (esto se describirá a detalle en la siguiente sección) y es la Corte Suprema la última instancia ante la cual se pueden apelar todos los procesos judiciales.
- Todo proceso tiene su equivalente expediente, el cual contiene las piezas escritas del proceso, agregadas sucesivamente y en orden de presentación, con las que se forma un solo cuerpo foliado con número y letras.

12. Técnica de Observación

Toda investigación o aplicación de carácter científico en Ciencias Sociales, como en las ciencias en general, debe implicar la utilización de procedimientos operativos rigurosos, bien definidos, transmisibles, susceptibles de ser aplicados de nuevo en las mismas condiciones, adaptados al fenómeno a discutir. Esas son las técnicas; la elección de éstas depende del objetivo perseguido, el cual está ligado al método de trabajo. Todo medio (método, técnica o instrumento) siempre debe estar subordinado al objetivo que se busca. Por lo tanto, el problema de la elección de un medio, es un problema de adecuación "medio - objetivo". (Fabbri, 2006)

"Ob-servar es ponerse delante (eso significa el prefijo ob.) de un objeto, a la vez como esclavo (que éste es el primer significado de la raíz serv.) Para serle fiel; y como maestro para poseerlo o conservarlo (que es el segundo significado de la raíz serv.). Observar a uno es poner la mirada en él, considerarlo como objeto."

Es un procedimiento empírico por excelencia, el más primitivo y a la vez el más usado.

Es el método por el cual se establece una relación concreta e intensiva entre el investigador y el hecho social o los actores sociales, de los que se obtienen datos que luego se sintetizan para desarrollar la investigación.

Este método es una "lectura lógica de las formas" y supone el ejercicio y "metodología de la mirada" (deconstrucción y producción de nueva realidad).

Desde fines del siglo pasado y comienzos de éste, se fue recurriendo a esta técnica de manera cada vez más sistemática, llegando a constituirse como el instrumento metodológico por excelencia en la antropología, si bien hoy cabe reconocer que su uso es mucho más amplio.

La observación es un proceso cuya función primera e inmediata es recoger información sobre el objeto que se toma en consideración. Esta recogida implica una actividad de codificación: la información bruta seleccionada se traduce mediante un código para ser transmitida a alguien (uno mismo u otros). Los numerosos sistemas de codificación que existen, podrían agruparse en dos categorías: los sistemas de selección, en los que la información se codifica de un modo sistematizado mediante unas cuadrículas o parrillas preestablecidas, y los sistemas de producción, en los que el observador confecciona él mismo su sistema de codificación.

En su origen la palabra "observar" significaba "ajustarse a lo que está prescripto" (ponerse en actitud de siervo ante la ley): "observar los mandamientos", "observar la Ley". En este sentido se habla de observancia: se vigila, se observa también, a cualquiera que infringe la norma para hacerle una observación o una indicación. Aunque nosotros nunca utilizaremos este significado, es interesante con todo subrayarlo para acotar el campo connotativo de la noción de observación.

Desde este punto de vista de las técnicas de investigación social, la observación es un procedimiento de recolección de datos e información que consiste en utilizar los sentidos para observar hechos y realidades sociales presentes y a la gente donde desarrolla normalmente sus actividades.

En los contextos experimentales, clínico y educativo, la observación conlleva alternativamente varias significaciones más específicas, derivadas de su primer sentido (la observación considerada como un proceso).

La observación es un objetivo que hay que conseguir o una aptitud que hay que desarrollar: aprender a observar; desarrollar el sentido de la observación.

El investigador hablará más de técnica de observación que de método de observación. Con ello pretende significar una técnica de recogida de datos sirviéndose de uno de varios observadores (sentido estricto) o de cuestionario de encuesta (sentido amplio).

El investigador hablará también de observación en oposición a experimentación. En tal caso, la observación designa esa fase de la investigación, consistente en familiarizarse con una situación o fenómeno determinado, en describirlo, en analizarlo con el fin de establecer una hipótesis coherente con el cuerpo de conocimientos anteriores ya establecidos.

La observación es el resultado codificado del acto de observar seguido del acto de interpretar. Para el investigador, ello supone la referencia a un marco teórico.

OBSERVAR ES UN PROCESO, SITUADO MAS ALLA DE LA PERCEPCION QUE, NO SOLAMENTE HACE CONSCIENTES LAS SENSACIONES, SINO QUE LAS ORGANIZA. La observación organiza las percepciones. Ello implica toda una serie de operaciones de sensibilización y de concentración de la atención, de comparación, de discernimiento, todo ello dirigido por una intención. Observar es un acto promovido a la vez por disposiciones de orden cognitivo y afectivo, cuyo producto es una creación del espíritu. Al final el producto de una observación puede ser que no tenga ninguna existencia real, es una representación interiorizada, seguida de una expresión exteriorizada en lenguajes muy distintos (verbal, gráfico, formal, etc.) de una realidad percibida. En ella juegan un papel importantísimo la intencionalidad y la atribución de un significado.

SE PUEDE DEFINIR EL PROCEDIMIENTO DE OBSERVACION: LA OBSERVACION ES UNA OPERACIÓN DE SELECCIÓN Y DE ESTRUCTURACION DE DATOS DE MODO QUE QUEDE PATENTE UNA RED DE SIGNIFICACIONES.

La selección de datos: comprende la recogida de información en la situación de que se trata. Las informaciones que hay que recoger dependen de los objetivos que uno se fije. Puede hacerse en función de una hipótesis preconcebida o de una manera más explícita, en función del objetivo que se persigue con la investigación.

La estructuración de los datos: los datos recogidos de ese modo son categorizados, o sea, clasificados según criterios de equivalencias determinados de antemano.

En la observación en un medio natural, como es el caso de las situaciones educativas, las influencias producidas por el observador, son más difíciles de someter a un estudio experimental.

El observador de una situación pedagógica puede ser una persona en formación, un formador, un maestro que participa de una experiencia pedagógica, un investigador, un inspector. Las condiciones psicológicas cambian según el estatus o el rol que le atribuyen los participantes en la situación que hay que observar. Puede tratarse de una verdadera intrusión que introduce modificaciones en las estructuras comportamentales; o bien de una presencia reactiva que enriquece algunos fenómenos habituales sin llegar a modificarlos totalmente.

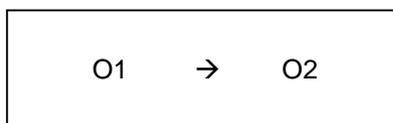
Como también intervienen las representaciones que el observador tiene de la situación educativa, provenientes de su experiencia como observador, y a veces, de este actor en este tipo de situaciones, y sus referenciales teóricos, valores personales, opiniones

ideológicas y pedagógicas, su método de observación va unido a la anticipación que él hace del acopio de un determinado tipo de datos. A veces el método, sobre todo si es género clínico, corre el riesgo de volcarse siempre sobre sí mismo. Su estado psicológico, en un momento dado, le hace más vulnerable a ciertos hechos y más apto para darse cuenta de unos comportamientos mejor que de otros.

III. MATERIALES Y MÉTODOS

1. Metodología de Investigación

Para el desarrollo de la presente investigación se optó por aplicar el diseño de investigación experimental/causal. Se utilizará el pre y post-test sin grupo de control, de la siguiente manera:



En donde: O1 es el pre-test a la variable dependiente, O2 es el post-test a la misma variable, donde → es la aplicación de un sistema inteligente de búsqueda desarrollado usando ontologías de la web semántica.

2. Diagramación de Procesos

Para lograr realizar el diagrama de procesos, que es uno de los objetivos específicos. Que a su vez ayudarán a determinar de manera más precisa los requerimientos del sistema a desarrollar.

- Uso de la técnica de la observación para comprender como realizan sus actividades dentro del estudio jurídico.
- Utilización de BPMN para diagramar cada uno de los procesos del estudio jurídico, teniendo en cuenta lo obtenido con la técnica de observación.

3. Construcción de la Sistema Inteligente de Búsqueda de Expedientes Judiciales.

Para lograr la construcción del sistema a desarrollar, segundo objetivo específico. Se tendrá en cuenta los siguientes puntos.

- Utilización de la Metodología ICONIX para el desarrollo del sistema inteligente de búsqueda.
- Uso de Methontology para la creación de la ontología de búsqueda en MongoDB.
- Construcción de Sistema inteligente de búsqueda, utilizando Zend Framework para crear el WSDL que transmite los datos desde MongoDB hacia la aplicación desarrollada en PHP.

4. Evaluación del Sistema Inteligente de Búsqueda

Para poder evaluar la efectividad del sistema inteligente de búsqueda se deberá realizar pruebas de uso.

- Tomar tiempos de búsqueda de expedientes utilizando el sistema inteligente de búsqueda y la técnica que siempre han usado (manual).
- Validar la coherencia de los datos obtenidos.
- Determinar qué tipo de evaluación estadística es la que corresponde a los datos obtenidos.
- Ejecutar el análisis estadístico de los tiempos tabulados.
- Interpretar los resultados estadísticos y establecer si se cumple o no la hipótesis de esta investigación.

IV. RESULTADOS

1. Modelado de Procesos del Estudio Jurídico Díaz Pulido

Tras realizar el estudio de la empresa con la Técnica de Observación, se ha obtenido una serie de resultados concernientes a los procesos realizados en la Empresa.

1.1. Subproceso de Búsqueda de Expediente

El subproceso está orientado exclusivamente a la búsqueda de expedientes. Subproceso que comienza al tener el nombre del patrocinado del cual se desea tener el expediente y se realiza la búsqueda manual en uno de los almacenes de expedientes, cabe resaltar que el Estudio Jurídico Díaz Pulido cuenta con dos almacenes de expedientes. De ser localizado el expediente se procede a la finalizar este subproceso.

En caso de no ser localizado en el primer almacén, se procede a la búsqueda en el segundo almacén, donde de ser encontrado se finaliza también el subproceso.

Un segundo punto de búsqueda son las oficinas de los abogados, ya que por su propia función pueden tener más de un expediente en sus escritorios. Dado que es más sencillo buscar de esta forma porque se puede preguntar a cada uno de los abogados, si tiene en su poder el expediente del patrocinado.

En caso que la búsqueda de negativo, se utilizan otros nombres para buscar, nombres de patrocinados que también están relacionados al expediente.

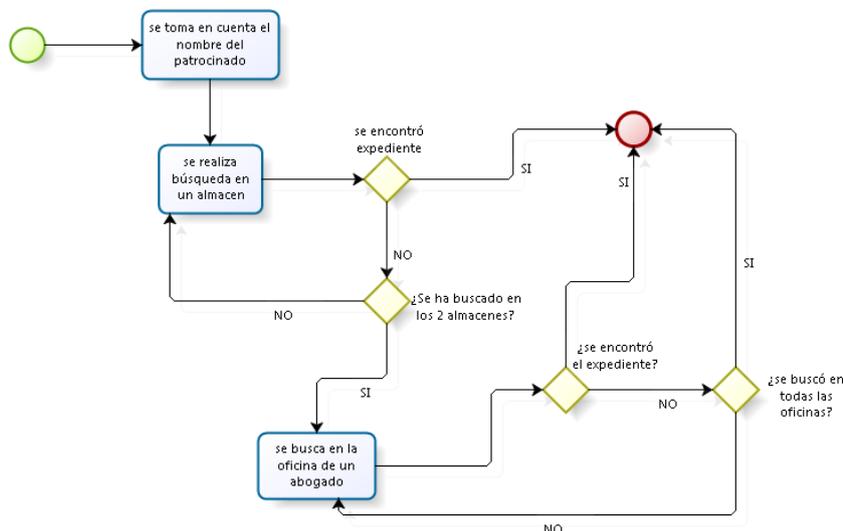


Ilustración 17: Modelo de Subproceso de Búsqueda de Expediente

1.2. Anexo de Documentación a Expediente

Este proceso inicia cuando el mensajero del Poder Judicial entrega documentación del expediente al Estudio Jurídico Díaz Pulido, donde la secretaria receptiona dicho

documento. Donde lo primero en realizarse es verificar si ésta documentación no corresponde a ningún expediente del que el estudio jurídico ya tenga un registro.

De verificarse que no corresponde a ningún expediente que se tenga almacenado, se procede a ingresarlo en un nuevo folio y a su respectivo almacenaje en una posición acorde con el nombre del patrocinado.

Pero en caso de tratarse de un expediente que ya se tiene en los almacenes, se procede a la realización de la búsqueda del mismo y una vez encontrado se le anexa la documentación que acaba de recibirse y se coloca en el folio para que vuelva a ser almacenado.

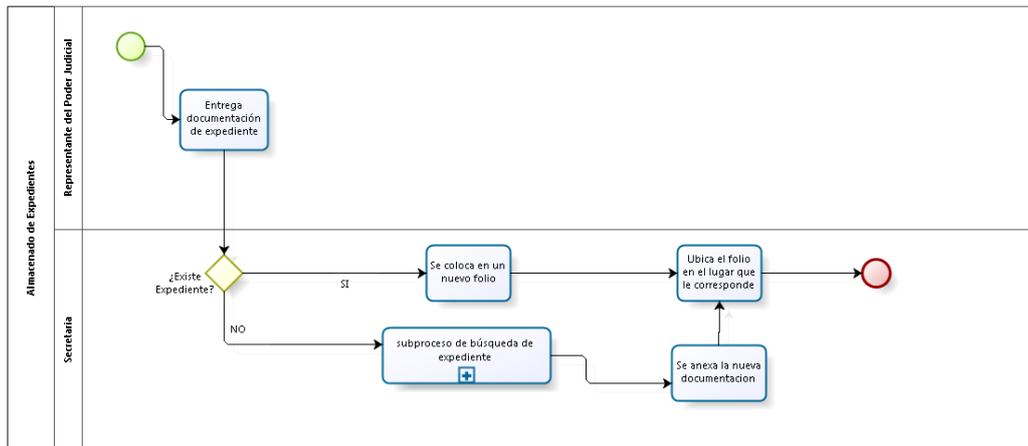


Ilustración 18: Modelo del Proceso de Anexado de Documentación a Expediente

1.3. Búsqueda de Expediente por Cliente

Este proceso depende únicamente del hecho que un abogado necesite consultar la información que este en un determinado expediente. La búsqueda se inicia cuando el abogado solicita el expediente de un cliente, donde la secretaria tendrá que efectuar el procedimiento de búsqueda de expedientes, donde una vez encontrado, le entregue la documentación al abogado.

En caso de no ser encontrado el expediente solicitado, la secretaria preguntará al abogado solicitante, si es que el expediente tiene más de un patrocinado registrado, en cuyo caso se deberá hacer una nueva búsqueda con el nombre que se le acaba de facilitar.

Básicamente esta parte será efectuada varias veces dependiendo de la cantidad de patrocinados implicados en el expediente y si se logra localizar el expediente con los nombres que paulatinamente se le den a la secretaria.

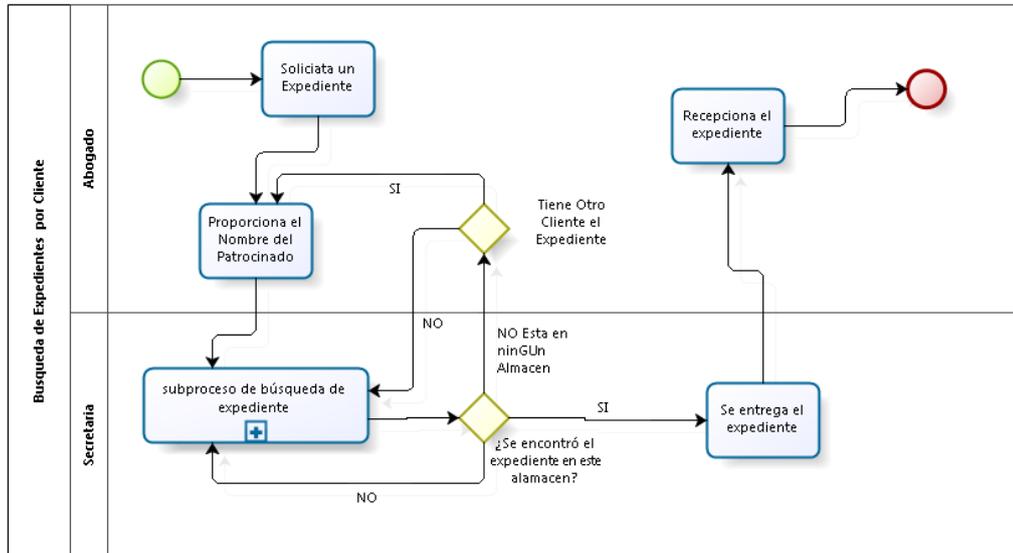


Ilustración 19: Modelo del Proceso de Búsqueda de Expediente por Cliente

2. Diseño de la Sistema Inteligente de Búsqueda de Expedientes Judiciales

A continuación se mostrará los distintos elementos que constituyen el sistema inteligente de búsqueda de expedientes judiciales.

2.1. Listado de Actores

- Abogado
- Secretaria

2.2. Lista de Casos de Uso del Negocio

- Administrar Clientes
- Administrar Expedientes

2.3. Especificación de Casos de Uso del Negocio

- Administrar Clientes: La secretaria realiza el registro de clientes, para que estos puedan ser debidamente vinculados a los Expedientes Judiciales con los que trabajan. Esta administración también incluye la búsqueda de clientes y la edición de sus datos.
- Administrar Expedientes: Los abogados, o la secretaria hasta cierto punto, realizan los registros de expedientes por cada cliente que ha sido debidamente registrado. Cada expediente tiene asignadas diversas especificaciones que lo hacen único por cada cliente tales como: el Juzgado al que es asignado, el tipo de judicatura, el proceso en que se encuentra, la

especialidad a la que se deriva (Laboral, civil, etc.), la materia en la que se encuentra y las personas implicadas en el caso.

2.4. Modelo de Casos de Uso del Negocio

Describe los procesos de un negocio, vinculados al campo de acción, y cómo se benefician e interactúan los abogados y secretaria en estos procesos.

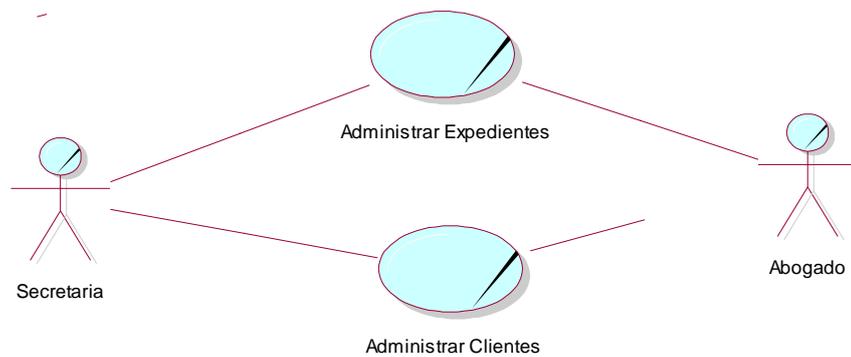


Ilustración 20: Modelo de Caso de Uso del Negocio

2.5. Requerimientos

2.5.1. Requerimientos Funcionales

- Registrar Clientes
- Buscar Clientes
- Editar Clientes
- Registrar Expedientes
- Buscar Expedientes
- Editar Expedientes
- Registrar Seguimiento
- Listar Seguimiento
- Realizar Búsquedas de Expedientes Judiciales
 - Sólo brindar los datos necesarios para hacer un listado de los expedientes relacionados con la búsqueda.
- Ordenarlos por semejanza a los parámetros proporcionados a sistema inteligente de búsqueda.

2.5.2. Requerimientos No Funcionales

- Rendimiento: el sistema no debe tardar más de 1 segundos en realizar la búsqueda de un Expediente.
- Usabilidad: las interfaces deben ser de navegación intuitiva.

- Escalabilidad: El sistema inteligente de búsqueda debe poder ser adaptable a modificaciones continuas, sin presentar mayor problema en su desarrollo.

2.6. Diagramas de Caso de Uso

Los casos de uso diseñados para cada módulo son los siguientes:

- Administrar Expedientes

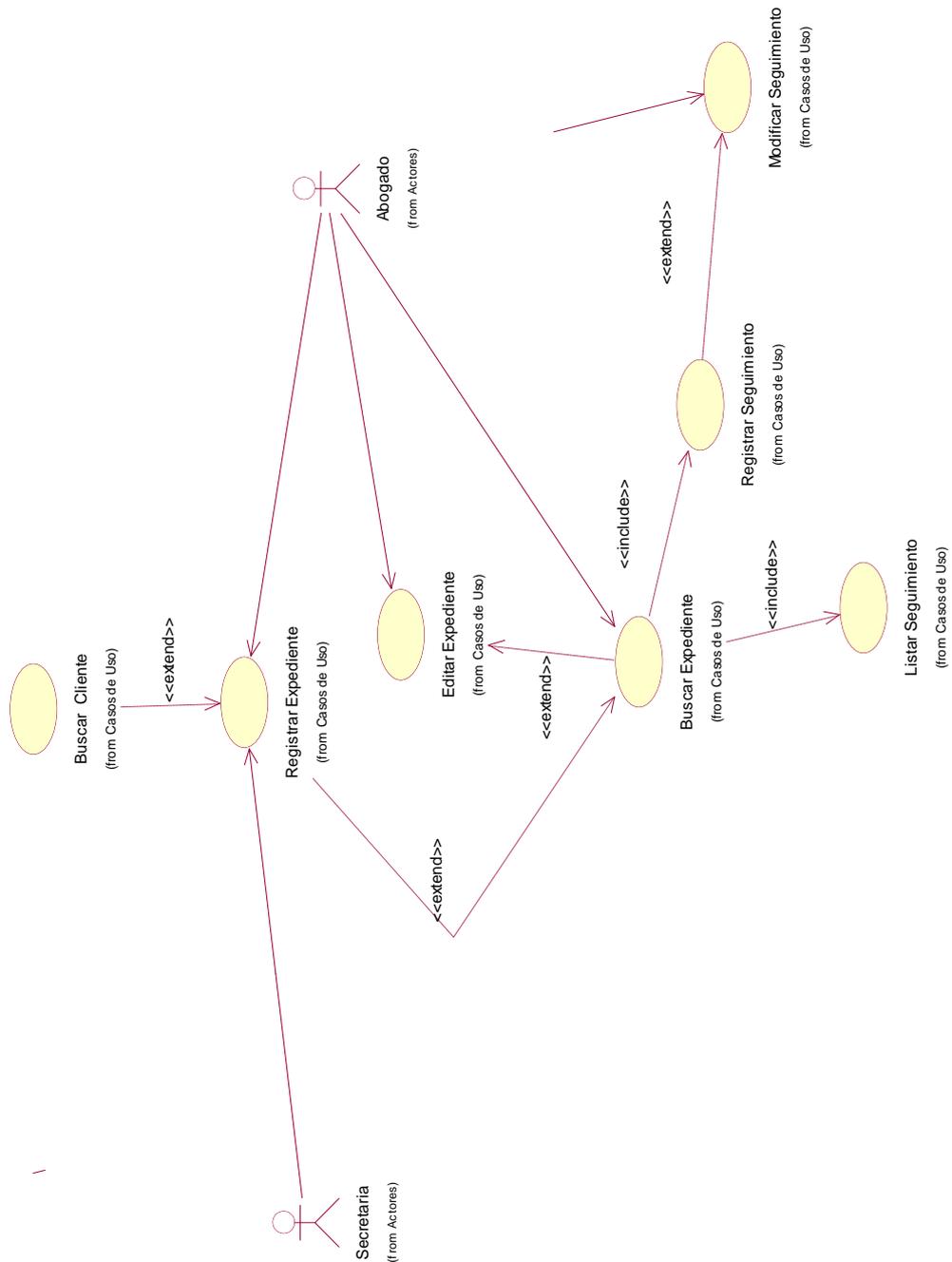


Ilustración 21: Administrar Expedientes

- Administrar Clientes

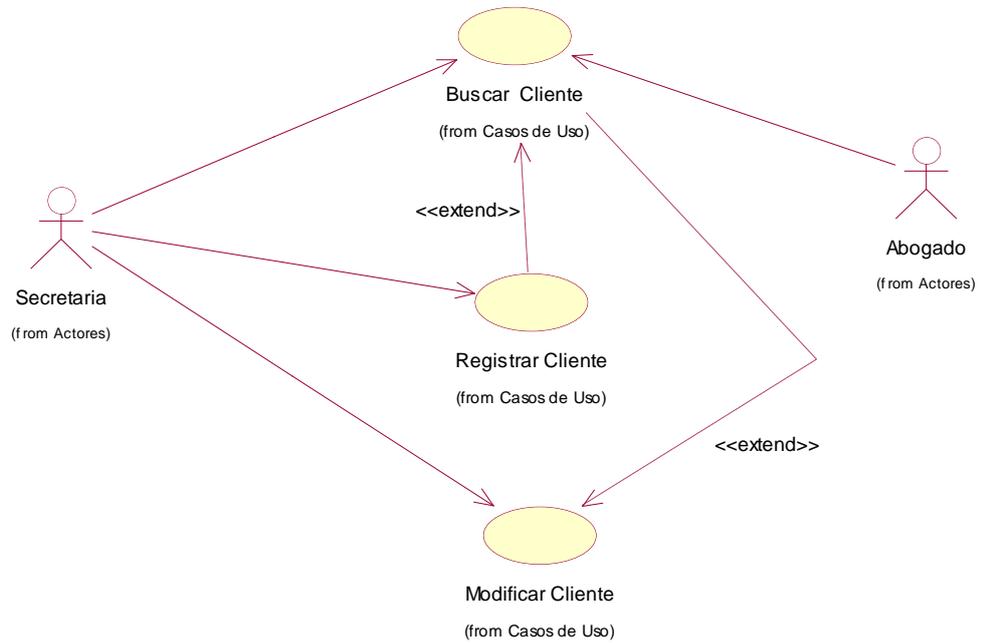


Ilustración 22: Administrar Clientes.

2.7. Especificación de Casos de Uso

A continuación se dará la especificación de cada uno de los casos de uso ya modelados en las ilustraciones Ilustración 21: Administrar y la Ilustración 22: Administrar Clientes.

Nombre:	Registrar Expediente
Autor:	Francisco Supo
Descripción:	Permite crear un expediente en la base de datos.
Actores:	Los abogados
Precondiciones:	El abogado debe haberse logeado en el sistema.
Flujo Normal:	El abogado ingresa los datos necesarios para el expediente. Los datos son registrados en la base de datos. La base de datos retorna el mensaje de “Registro Satisfactorio”

<p>Flujo Alternativo:</p> <p>El sistema comprueba la validez de los datos, si los datos no son correctos, se avisa al actor de ello permitiendole que los corrija.</p>
<p>Poscondiciones:</p> <p>El mensaje registro satisfactorio.</p>

Nombre:	Buscar Expediente
Autor:	Francisco Supo
Descripción:	Permite buscar un expediente en la base de datos.
Actores:	Los abogados
Precondiciones:	El abogado debe haberse logeado en el sistema.
Flujo Normal:	<p>Selecciona la opción Búsqueda Avanzada en el menú principal o desde la búsqueda del cliente.</p> <p>Selecciona la opción modificar expediente</p> <p>Se modifican los datos del expediente.</p> <p>En la base de datos se registran los cambios.</p> <p>Se muestra el mensaje "Modificación Satisfactorio"</p>
Flujo Alternativo:	El sistema comprueba la validez de los datos, si los datos no son correctos, se avisa al actor de ello permitiendole que los corrija.
Poscondiciones:	Muestra solo los resultados.

Nombre:	Modificar Expediente
Autor:	Francisco Supo, Marlon Araujo

<p>Descripción:</p> <p>Permite modificar un expediente en la base de datos.</p>
<p>Actores:</p> <p>Los abogados</p>
<p>Precondiciones:</p> <p>El abogado debe haberse logeado en el sistema.</p>
<p>Flujo Normal:</p> <ol style="list-style-type: none"> 1. Selecciona la opción Búsqueda Avanzada en el menú principal o desde la búsqueda del cliente. 2. Selecciona la opción modificar expediente 3. Modifica los campos del expediente. 4. El sistema modifica el expediente y muestra el mensaje "Modificación Satisfactorio"
<p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 1. El sistema comprueba la validez de los datos, si los datos no son correctos, se avisa al actor de ello permitiendole que los corrija.
<p>Poscondiciones:</p> <p>Muestra solo los resultados.</p>

Nombre:	Registrar Seguimiento
Autor:	Francisco Supo
Descripción:	Permite crear un seguimiento en la base de datos.
Actores:	Los abogados
Precondiciones:	El abogado debe haberse logeado en el sistema.
Flujo Normal:	<p>El abogado ingresa los datos necesarios para el seguimiento.</p> <p>El sistema registrara los datos.</p> <p>El sistema mostrará el mensaje "Registro Satisfactorio"</p>
Flujo Alternativo:	<p>El sistema comprueba la validez de los datos, si los datos no son correctos, se avisa al actor de ello permitiendole que los corrija.</p>
Poscondiciones:	El mensaje registro satisfactorio.

Nombre:	Registrar cliente
Autor:	Francisco Supo
Descripción: Permite crear un cliente en la base de datos.	
Actores: Secretaria	
Precondiciones: El abogado debe haberse logeado en el sistema.	
Flujo Normal: La secretaria selecciona la opcion nuevo cliente en el menú principal La secretaria llenara todos los datos personales necesairios para el registro. El sistema mostrará el mensaje "Registro Satisfactorio"	
Flujo Alternativo: El sistema comprueba la validez de los datos, si los datos no son correctos, se avisa al actor de ello permitiendole que los corrija.	
Poscondiciones: El mensaje registro satisfactorio.	

Nombre:	Buscar Cliente
Autor:	Francisco Supo
Descripción: Permite buscar un cliente en la base de datos.	
Actores: Secretaria	
Precondiciones: El abogado o la secretaria debe haberse logeado en el sistema.	
Flujo Normal: La secretaria selecciona la opción Buscar Cliente en el menú principal.	

<p>Busca por el tipo de cliente.</p> <p>El sistema mostrará los datos necesarios.</p>
<p>Flujo Alternativo:</p> <p>El sistema comprueba la validez de los datos, si los datos no son correctos, se avisa al actor de ello permitiendole que los corrija.</p>
<p>Poscondiciones:</p> <p>Muestra solo los resultados.</p>

Nombre:	Modificar Cliente
Autor:	Francisco Supo
Descripción:	Permite modificar un cliente en la base de datos.
Actores:	Secretaria
Precondiciones:	El abogado debe haberse logeado en el sistema.
Flujo Normal:	<p>La secretaria busca al cliente mediante el menú principal.</p> <p>Selecciona al cliente</p> <p>Modifica los campos del cliente.</p> <p>El sistema modifica el cliente y muestra el mensaje "Modificacion Satisfactorio"</p>
Flujo Alternativo:	<p>El sistema comprueba la validez de los datos, si los datos no son correctos, se avisa al actor de ello permitiendole que los corrija.</p>
Poscondiciones:	El mensaje modificación satisfactorio.

2.8. Diagramas de Robustez

En este punto se mostrarán los diagramas de robustez modelados para el sistema inteligente de búsqueda.

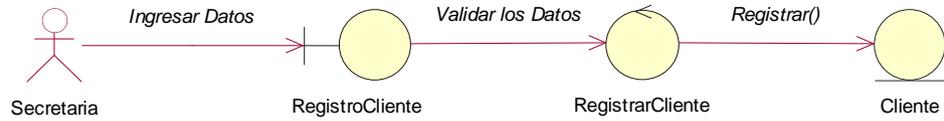


Ilustración 23: Registro de Cliente

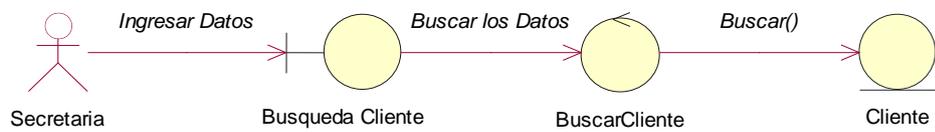


Ilustración 24: Buscar Cliente.

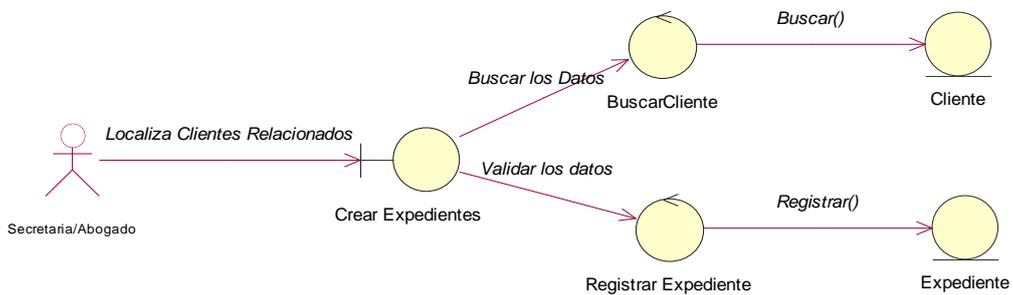


Ilustración 25: Registrar Expediente

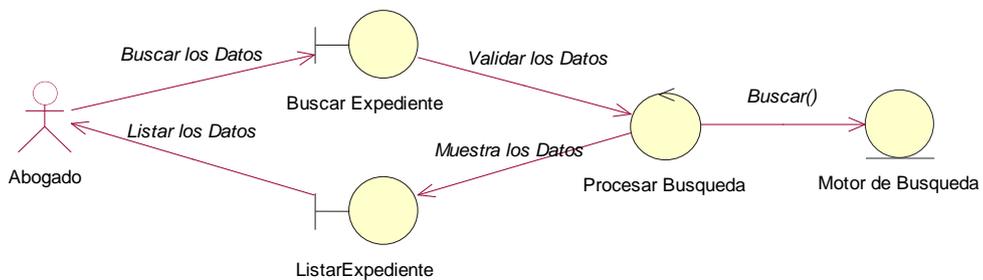


Ilustración 26: Buscar Expediente

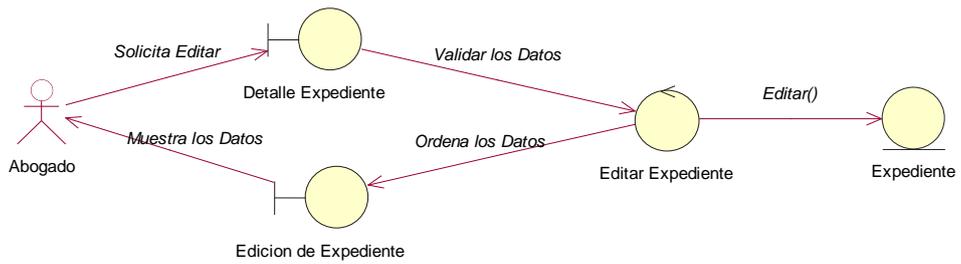


Ilustración 27: Editar Expedientes.

2.9. Diagramas de Secuencia

A continuación se mostrarán los diagramas de secuencia de los distintos procesos que se trabajarán en el sistema inteligente de búsqueda.

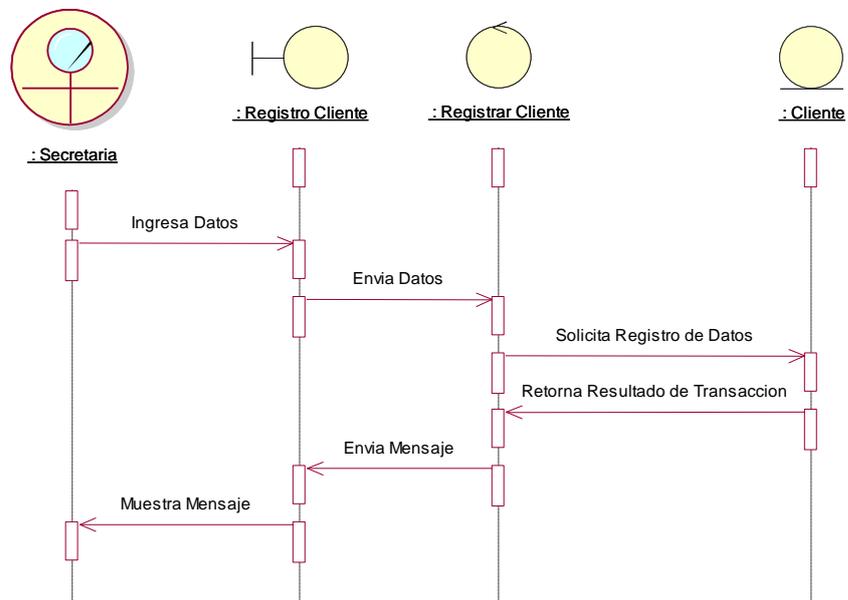


Ilustración 28: Registrar Cliente – Secuencia

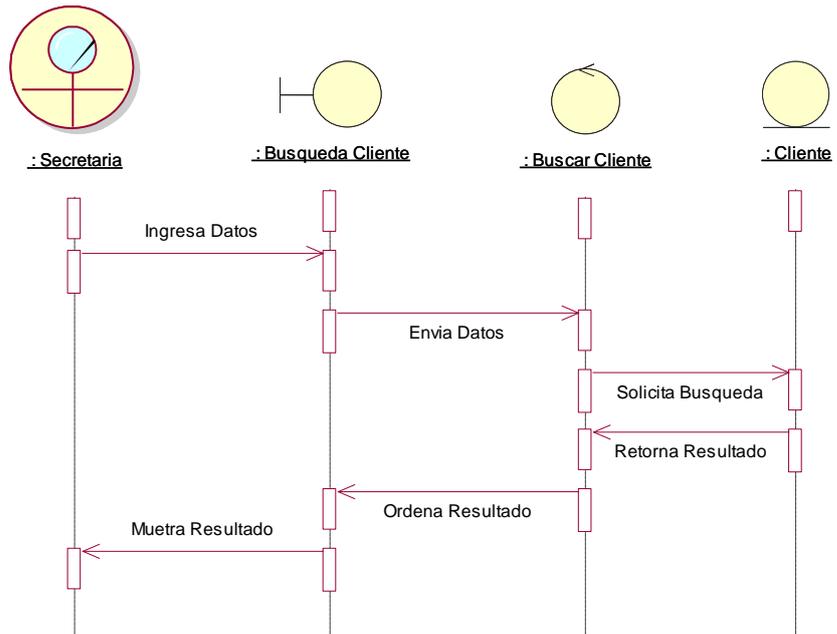


Ilustración 29: Buscar Cliente – Secuencia

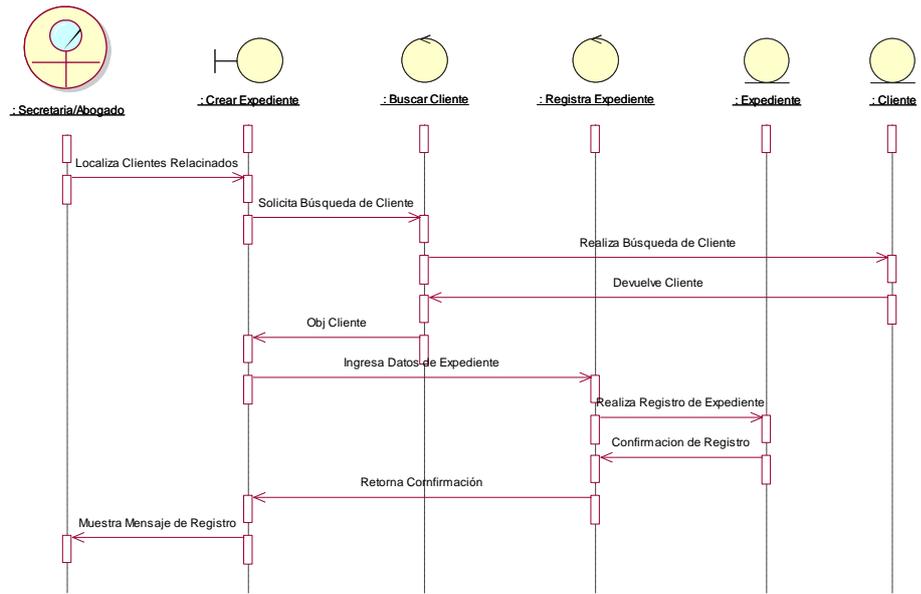


Ilustración 30: Registrar Expediente – Secuencia

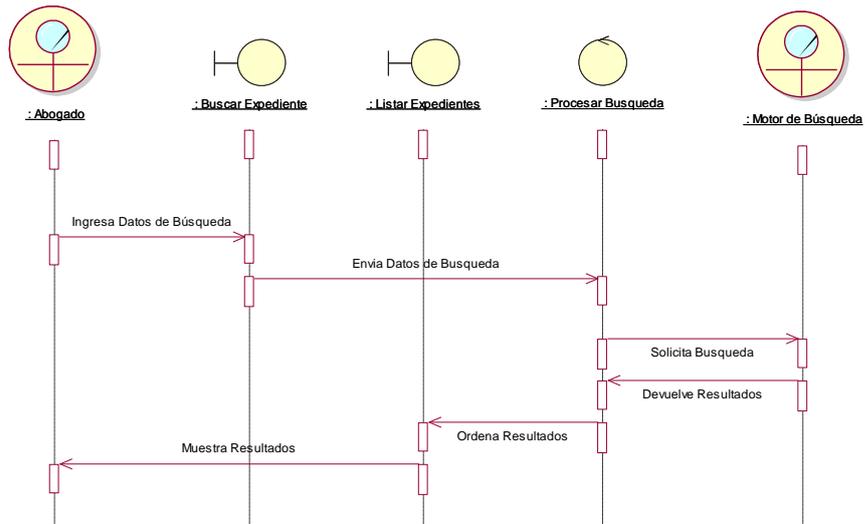


Ilustración 31: Buscar Expediente – Secuencia

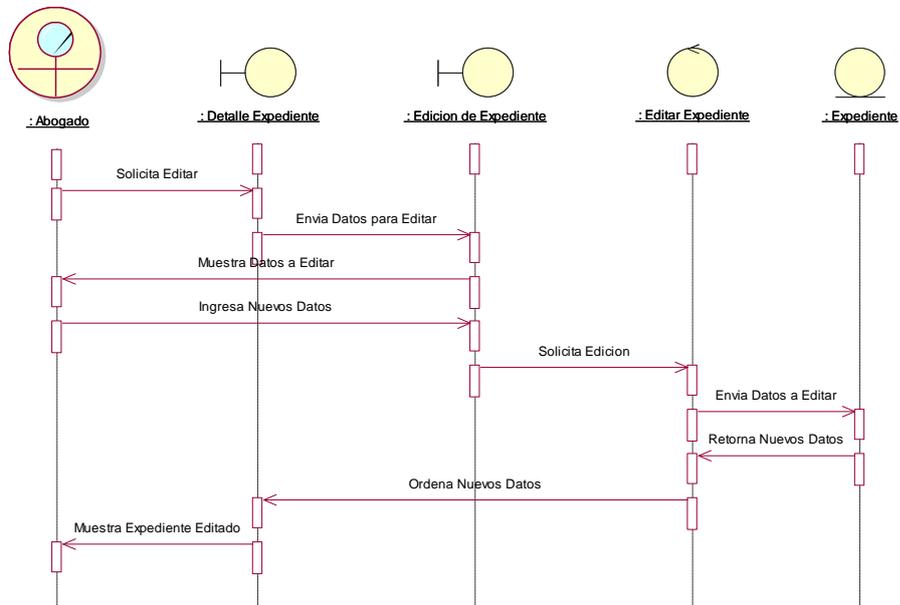


Ilustración 32: Editar Expedientes - Secuencia

2.10. Diagrama de Componentes

A continuación se mostrarán las dependencias que tienen los distintos componentes del Sistema Inteligente de Búsqueda de Expedientes Judiciales.

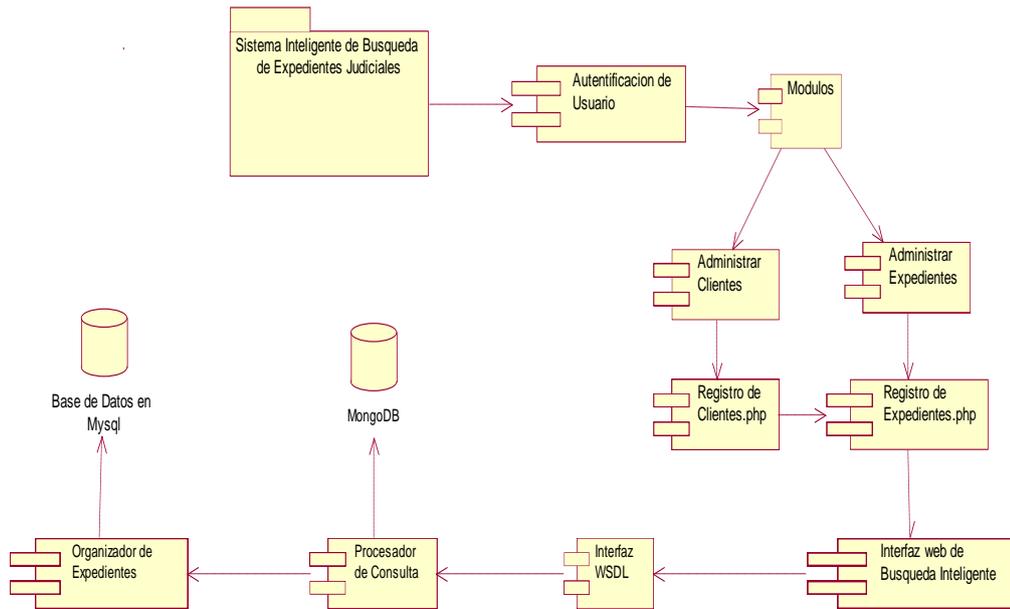


Ilustración 33: Diagrama de Componentes Sistema Inteligente de Búsqueda

2.11. Modelo de Dominio

En esta parte se representará gráficamente los conceptos clave del dominio del Sistema Inteligente de Búsqueda, y la relación que existe entre estos.

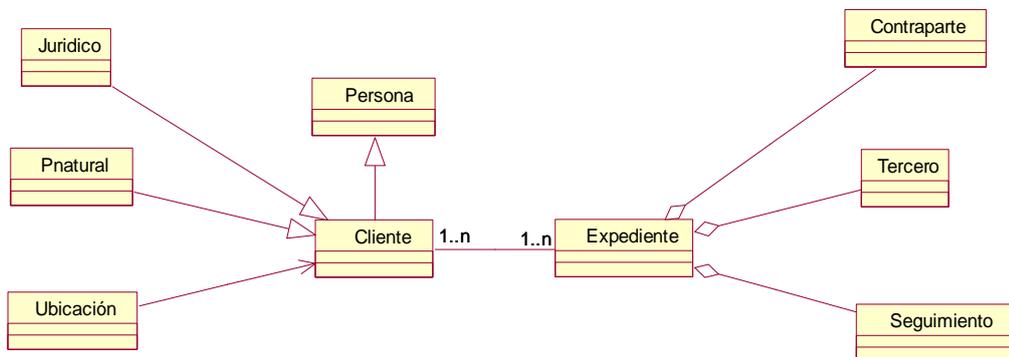


Ilustración 34: Modelo de Dominio

2.12. Diagrama de Clases

En este modelo se coloca el diagrama estático que describe la estructura del sistema mostrando sus clases, atributos y métodos. Todo orientado a objetos.

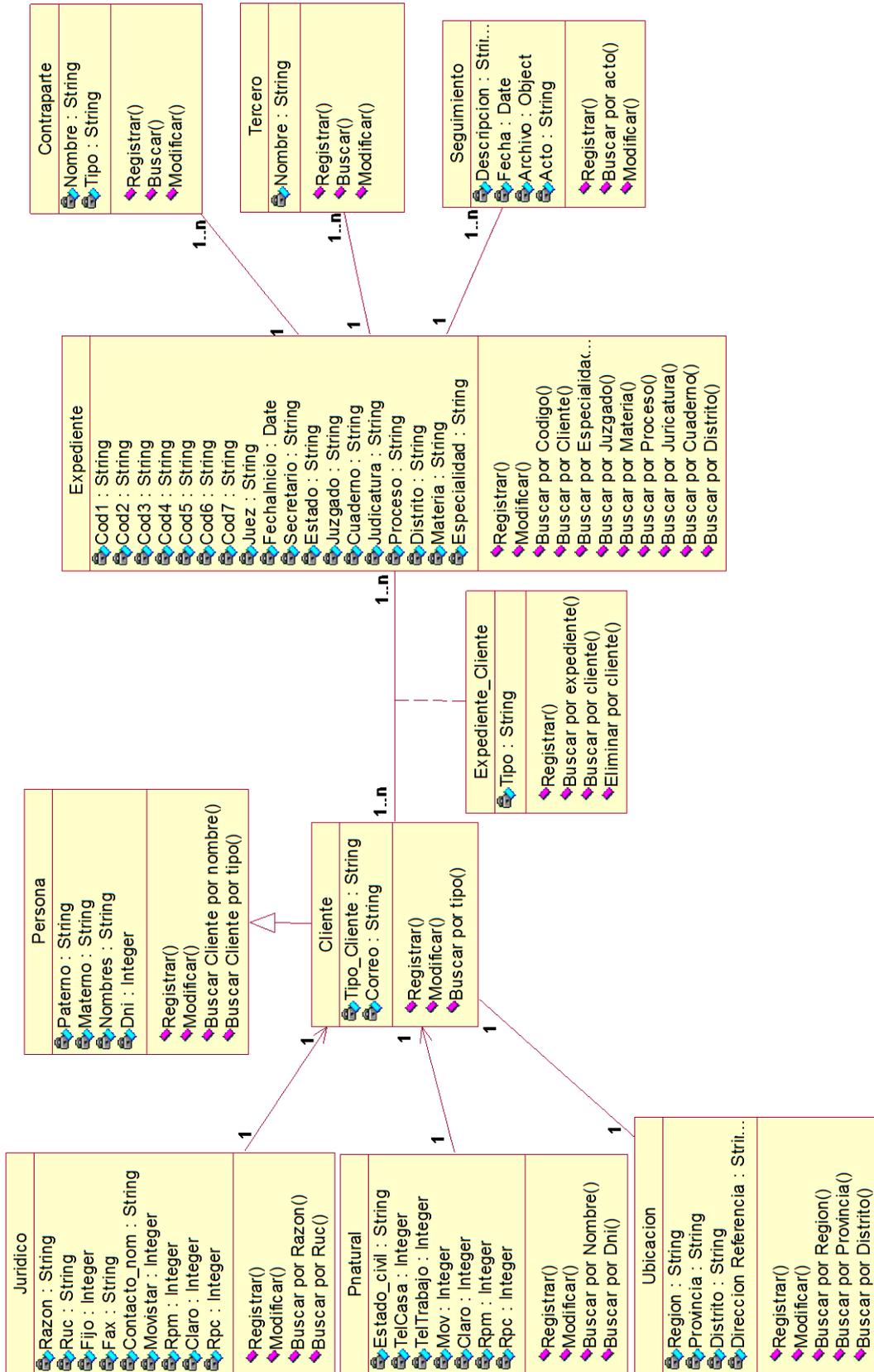


Ilustración 35: Diagrama de Clases

2.13. Modelo de Base de Datos

Aquí se mostrará el modelo de datos utilizado en el Sistema Inteligente de Búsqueda, el cual ofrece la granularidad necesaria para poder crear una ontología de búsqueda.

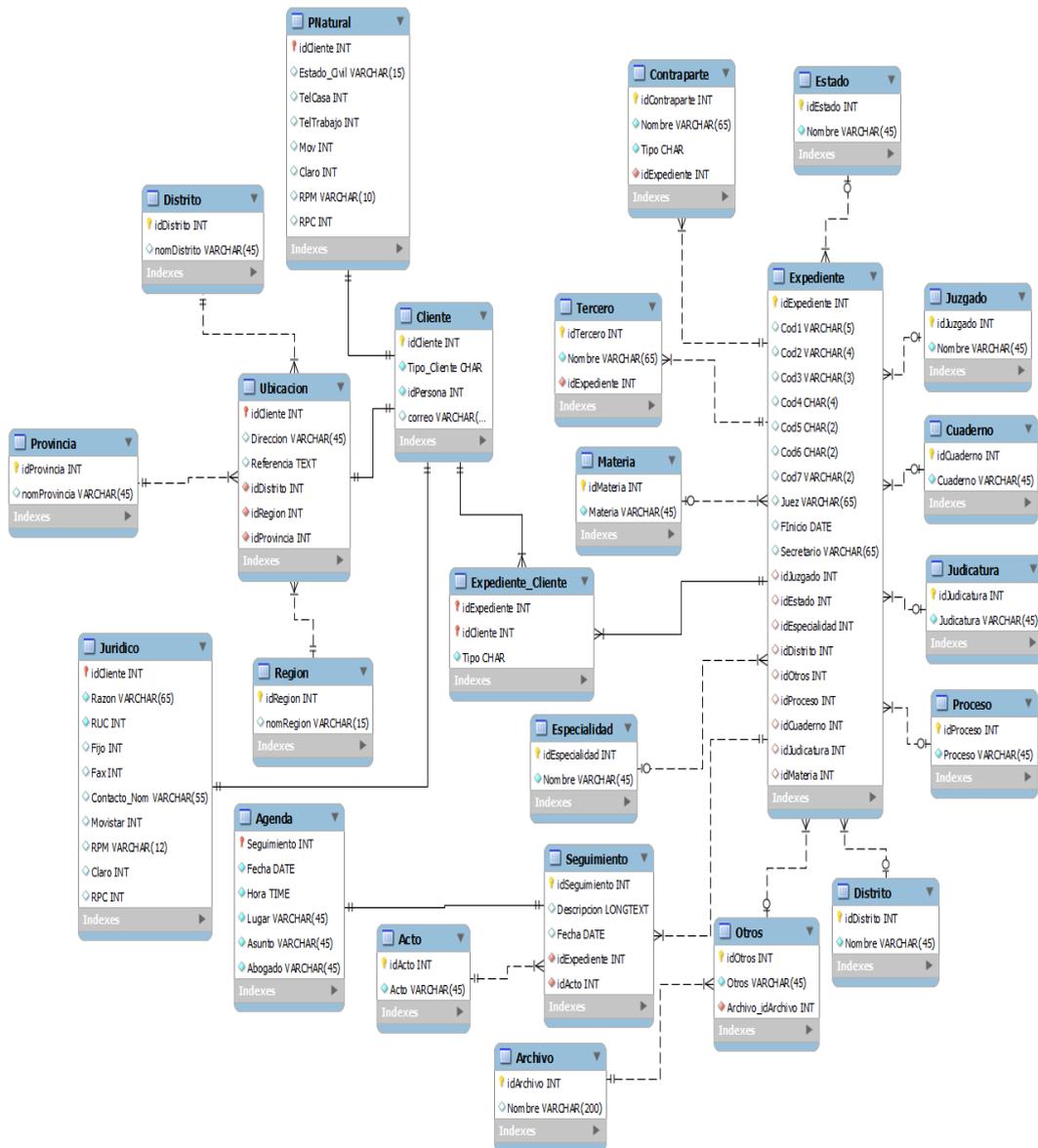


Ilustración 36: Modelo de Datos

2.14. Algoritmo de Búsqueda

En el siguiente diagrama se mostrará el proceso que realiza el sistema inteligente de búsqueda para realizar una búsqueda.

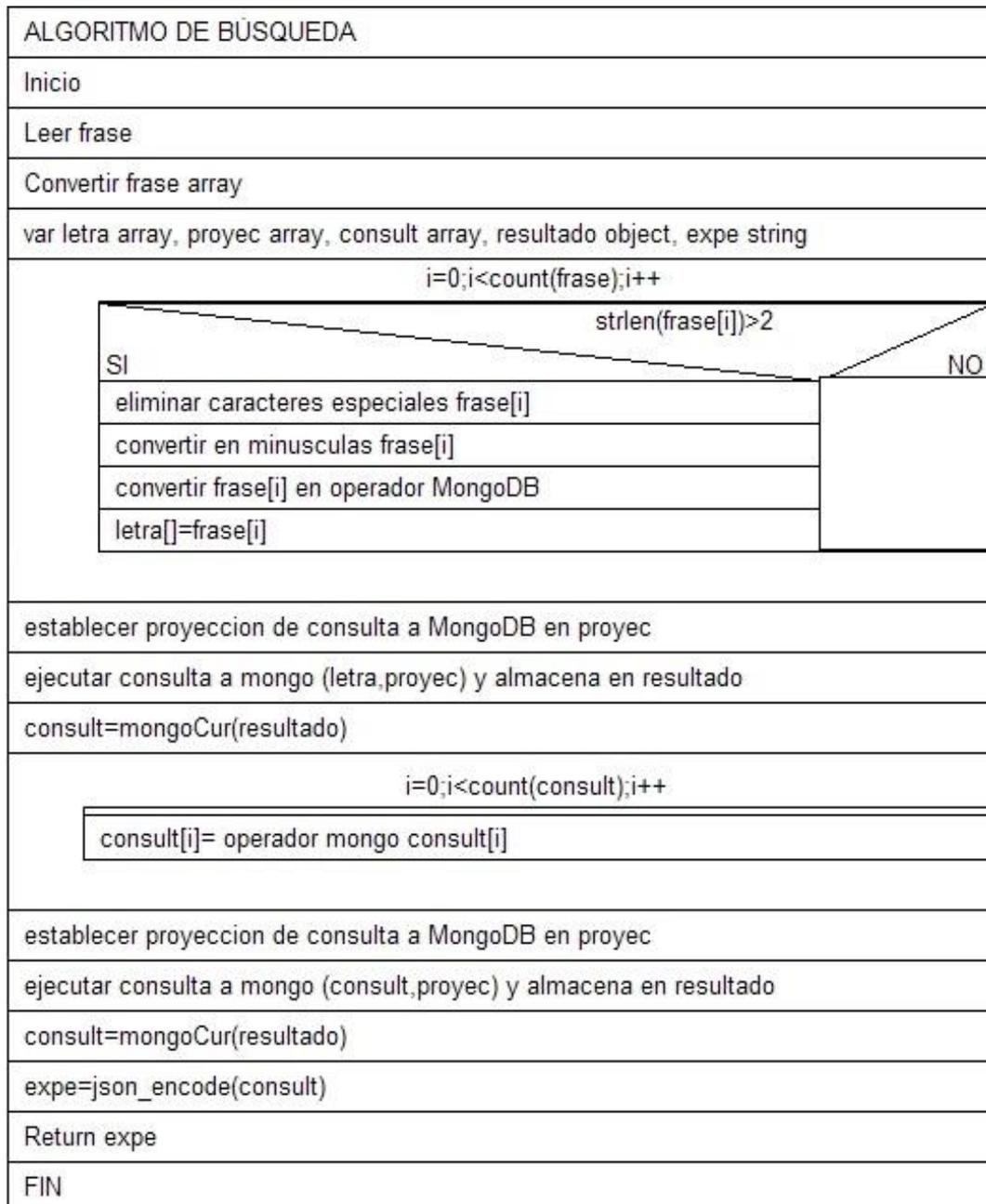


Ilustración 37: Diagrama N-S del Algoritmo de Búsqueda

3. Migración de Datos

En esta parte se mostrará la secuencia seguida para hacer la migración de los datos desde la base de datos relacionada en MySQL, hacia la base de datos plana en MongoDB. En la cual se utilizará un ordenamiento de datos en formato JSON.

3.1. Estructura Relacional

El modelo de datos que se usará para este proceso, es una parte del modelo de datos original presentado en la Ilustración 36: Modelo de Datos, para poder así respetar la vinculación entre los datos relacionales y los planos.

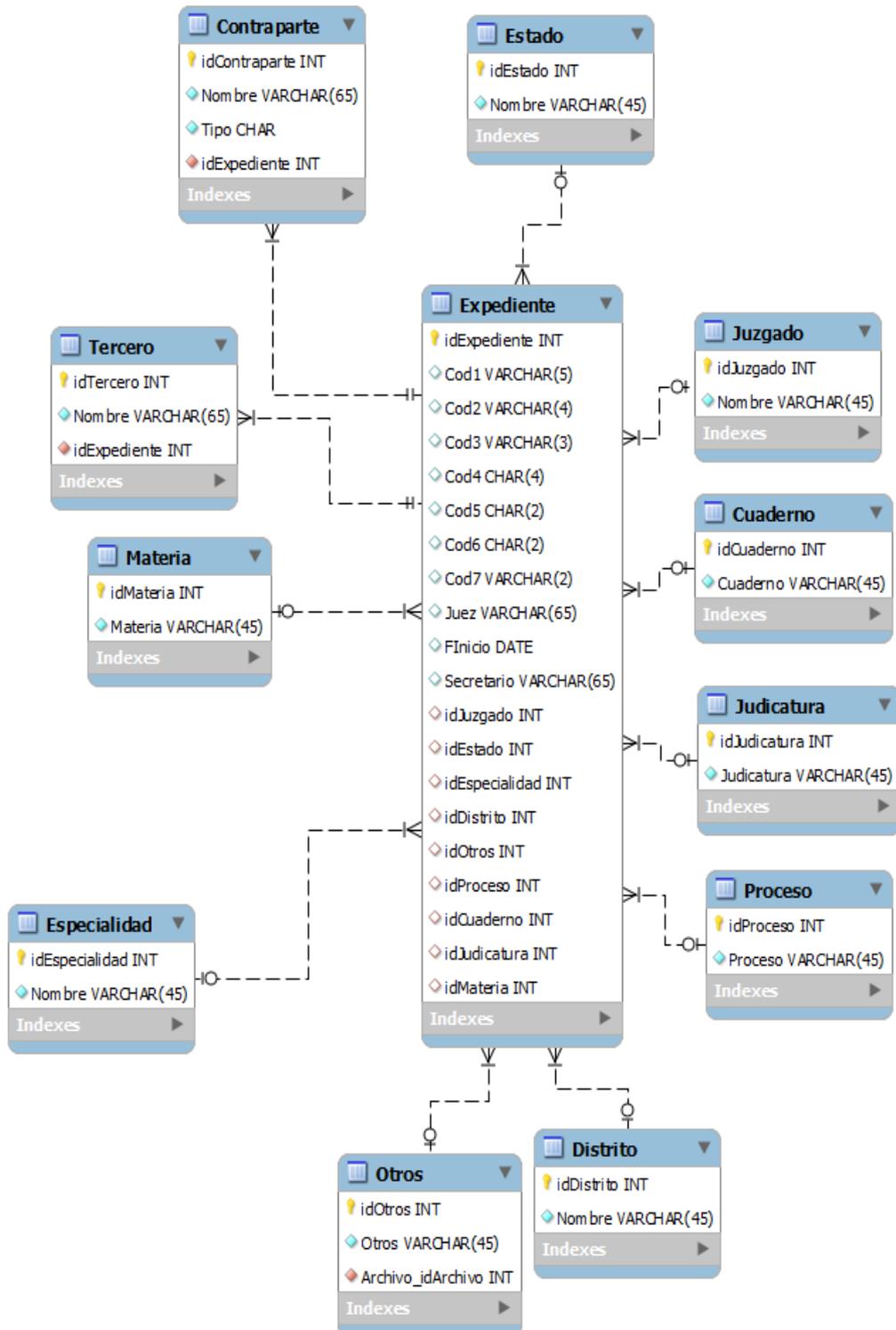


Ilustración 38: Modelo Relacional a Migrar a Plano

3.2. Estructura Plana

Modelo donde se representa los parámetros que tendrá cada documento dentro de la colección de MongoDB, el cual dependerá de los parámetros seleccionados en la Ilustración 38: Modelo Relacional a Migrar a Plano.

```
{
  "_id" : ObjectId("54dfc89c8a0c929010000029"),
  "Cod1" : "546",
  "Cod2" : "2012",
  "Cod3" : "0",
  "Cod4" : "1601",
  "Cod5" : "JP",
  "Cod6" : "CI",
  "Cod7" : "04",
  "Juzgado" : "primero",
  "Cuaderno" : "Principal",
  "Materia" : "Obligación de dar suma de dinero ",
  "Sumilla" : "Demanda de obligacion de dar suma de dinero"
}
```

Ilustración 39: Modelo de Datos Plano de la Lista de Expedientes Judiciales

3.3. Proceso de Migración de Datos

Este diagrama presenta cada uno de los pasos a seguir para conseguir el migrado de los datos desde MySQL a MongoDB.

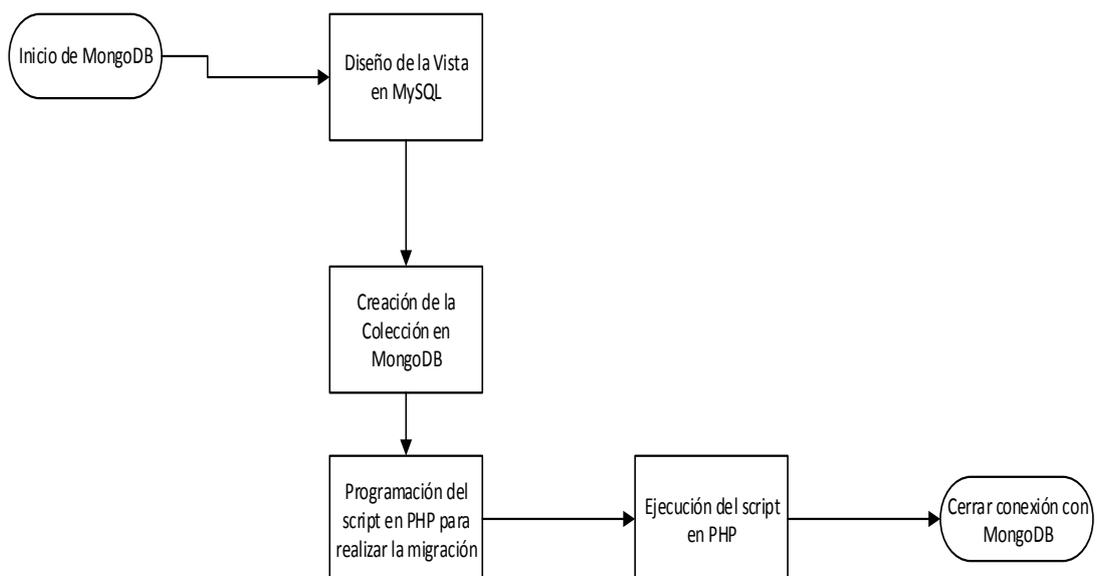


Ilustración 40: Proceso de Migración de Datos SQL a NoSQL

4. Construcción de Ontologías

A continuación se muestran los resultados obtenidos tras aplicar la Methontology para construir la ontología para.

4.1. Glosario de Términos

Esta información se obtuvo de los diversos tipos de expedientes que obtuvo en el proceso de analizar los conceptos que harán parte de nuestra ontología.

Términos	
Materia	Juzgados
<u>Civil</u>	Paz Letrado Primero
Contencioso administrativo	Paz Letrado Segundo
Obligación de dar suma de dinero	Paz Letrado Tercero
Reivindicación	Paz Letrado Cuarto
Interdicto de recobrar	Paz Letrado Quinto
Interdicto de retener	Paz Letrado Sexto
Nulidad de cosa juzgada fraudulenta	
Reivindicación y otros	Especializado Penal Primero
Acción de amparo	Especializado Penal Segundo
Divorcio por causal	Especializado Penal Tercero
Tercería	Especializado Penal Cuarto
Ejecución de laudos arbitrales	Especializado Penal Quinto
Por definir	Especializado Penal Sexto
Indemnización por daños y perjuicios	
Enriquecimiento indebido	Mixto Primero
	Mixto Segundo
<u>Laboral</u>	Mixto Tercero
Acción contenciosa administrativa	Mixto Cuarto
Desnaturalización	Mixto Quinto
	Mixto Sexto

<u>Familiar</u>	
Alimentos	
Violencia familiar"	

4.2. Diagrama de Taxonomía de Conceptos

El diagrama describe los conceptos en su orden de dependencia, de acorde con las necesidades del sistema inteligente de búsqueda del presente proyecto.

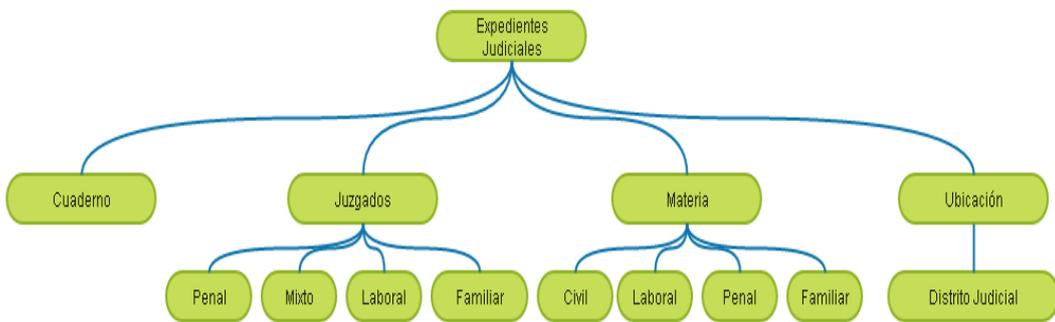


Ilustración 41: Diagrama de Taxonomía de Conceptos

4.3. Diagrama de Relaciones Binarias

Estos diagramas denotan la relación entre los elementos que el sistema inteligente de búsqueda utiliza para la búsqueda, y los elementos que componen un expediente judicial.

4.3.1. Relación de Materias

Un Expediente siempre va estar relacionado con una Materia, la cual es una parte indispensable para identificar el tipo de documento que se evaluara.

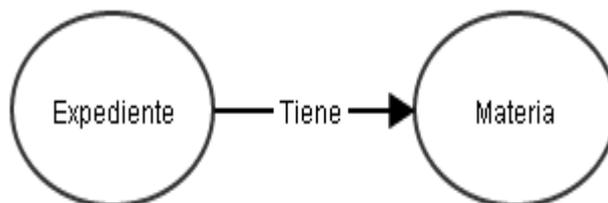


Ilustración 42: Relación Binaria Expediente – Materia

Lo que da como resultado a nivel de documentos que facilitan la búsqueda en MongoDB.

```
{
  "_id" : ObjectId("54df8e4fda02efabc5ecc707"),
  "Tabla" : "Materia",
  "Valor" : "Civil",
  "Cod" : [
    "Cod6"
  ],
  "CodVal" : [
    "CI"
  ],
  "Indices" : [
    "Civil",
    "Contencioso administrativo",
    "Obligación de dar suma de dinero",
    "Reivindicacion", |
    "Nulidad de cosa juzgada fraudalenta",
    "Reivindicacion y otros",
    "Acción de amparo",
    "Divorcio por causal",
    "Terceria",
    "Ejecución de laudos arbitrales",
    "Indemnización por danos y perjuicios",
    "Enriquecimiento indebido",
    "obligacion de dar suma de dinero",
    "ejecución de laudos arbitrales",
    "enriquecimiento indebido"
  ]
}
```

Ilustración 43: Referencias de Materia – MongoDB

4.3.2. Relación de Juzgados

Un Expediente pertenece a un Juzgado, cuyo objetivo principal es dar sentencia a estos casos.

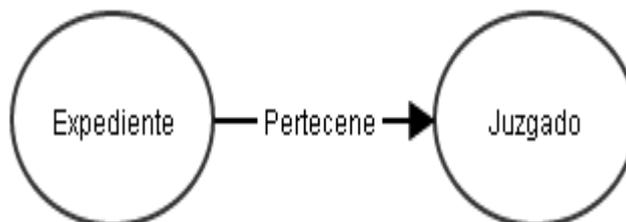


Ilustración 44: Relación Binaria Expediente – Juzgado

Que al representarse como referencia en MongoDB nos da como resultado.

```

{
  "_id" : ObjectId("54df8ec1da02efabc5ecc70c"),
  "Tabla" : "Juzgado",
  "Valor" : "Mixto",
  "Cod" : [
    "Cod5"
  ],
  "CodVal" : [
    "JM"
  ],
  "Indices" : [
    "Primero",
    "Segundo",
    "Tercero",
    "Cuarto",
    "Quinto",
    "Sexto",
    "Juzgado Mixto",
    "juzgado mixto",
    "Mixto",
    "mixto"
  ]
}

```

Ilustración 45: Referencias de Juzgado – MongoDB

4.3.3. Relación de Distrito Judicial

Un Expediente pertenece a un Distrito Judicial, son las areas de competencia determinada de una determinada corte superior establecida por ley, lo cual corresponde a todos nuestros distritos.

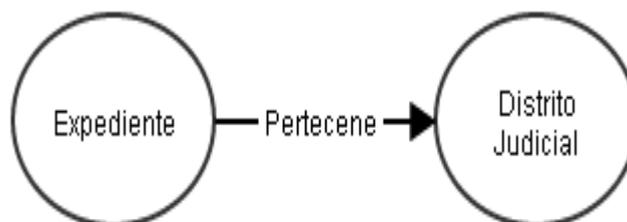


Ilustración 46: Relación Binaria Expediente - Distrito Judicial

Relación que desarrollándose en mongo, arroja este resultado.

```

{
  "_id" : ObjectId("54df8ed3da02efabc5ecc70d"),
  "Tabla" : "Distrito",
  "Valor" : "Trujillo",
  "Cod" : [
    "Cod4"
  ],
  "CodVal" : [
    "1601"
  ],
  "Indices" : [
    "Trujillo"
  ]
}

```

Ilustración 47: Referencias de Distrito Judicial – MongoDB.

4.4. Diccionario de Conceptos

Los Juzgados de Primera Instancia son el tercer nivel jerárquico en que se encuentra organizado el Poder Judicial del Perú. Cada provincia tiene, cuando menos, un juzgado de primera instancia. Aunque, por razones de carga procesal, se puede englobar varias provincias. Los juzgados de primera instancia tienen competencia sobre temas de mayor cuantía y se subdividen de acuerdo a la especialidad que conocen.

Las especialidades en que se dividen los juzgados son las siguientes:

- **Civil:** Que conoce todos los temas relacionados al Derecho Civil con excepción de lo relacionado al Derecho de Familia. Dentro de esta especialidad esta la subespecialidad Comercial, que actualmente cuenta con Salas y Juzgados Comerciales en la Corte Superior de Justicia de Lima.
- **Penal:** Que conocen de delitos y otros temas relacionados al Derecho Penal.
- **Laboral:** Que conocen de temas relacionados al Derecho Laboral.
- **Familia:** Que conocen de temas relacionados al Derecho de Familia.
- **Contencioso Administrativo:** Que conocen los casos en los que una institución del Estado vulnera un derecho de los ciudadanos.

4.5. Descripción de Axiomas

Axioma (definición): Proposición tan clara y evidente que se admite sin necesidad de demostración (Real Academia Española, 2012).

Basándonos en la definición anterior, los axiomas vendrían a ser aquellas relaciones o reglas que se cumplan siempre, lo cual arroja el siguiente resultado.

4.5.1. Cuaderno Principal

En este axioma, todo expediente, está formado por cuadernos. Donde el Tercer Código en caso de ser valor 0 (cero), es el cuaderno principal; es decir, el cuaderno con el que se abrió el proceso judicial y naturalmente, el expediente.

```
{
  "_id" : ObjectId("54df8eefda02efabc5ecc70f"),
  "Tabla" : "Cuaderno",
  "Valor" : "Principal",
  "Cod" : [
    "Cod3"
  ],
  "CodVal" : [
    "0"
  ],
  "Indices" : [
    "Principal"
  ]
}
```

Ilustración 48: Axioma del Cuaderno Principal - MongoDB

5. Prototipos de Interfaz del Sistema Inteligente de Búsqueda

Como resultado final se obtiene el prototipo del sistema inteligente de búsqueda, el cual cuenta con las siguientes interfaces.

5.1. Interfaz del Buscador

Este prototipo, Ilustración 49: Prototipo de Interfaz de Búsqueda, es el que se usará para obtener los resultados de tiempos de búsqueda. En el siguiente ejemplo se usa para tener un ejemplo de la forma de búsqueda que tiene el sistema inteligente de búsqueda de expedientes judiciales, realizando un filtro por el tipo Materia (Familiar) y por el Distrito Judicial (Trujillo).



Ilustración 49: Prototipo de Interfaz de Búsqueda

5.1. Interfaz de Resultado de Búsqueda

En la interfaz de resultados, Ilustración 50: Interfaz de Resultados de Búsqueda, se lista los resultados de lo solicitado en la interfaz de Búsqueda. En el ejemplo, el resultado de esta búsqueda ha sido logrado por el análisis respectivo por cada una partes que comprende un expediente y ordenándolas en el proceso que se encuentra cada una de estas.

Cliente	Expedientes Judiciales	Documentos	Reportes	Usuario
Atras				
N° Expediente: 1737	2014	0	1601	JP FC 02 Detalle
Sumilla: demanda de alimentos				
Juzgado: no definido				
Materia: Alimentos				
Cuaderno: Principal				
N° Expediente: 1737	2014	76	1601	JP FC 02 Detalle
Sumilla: cuaderno cautelar codigo N° 260830-2014				
Juzgado: segunda				
Materia: Alimentos				
Cuaderno: Cautelar				
N° Expediente: 3662	2014	0	1601	JR FC 03 Detalle
Sumilla: demanda de violencia familiar				
Juzgado: no definido				
Materia: violencia familiar				
Cuaderno: Principal				
N° Expediente: 822	2012	0	1601	JR FC 04 Detalle
Sumilla: demanda de Tercería de propiedad				
Juzgado: no definido				
Materia: Tercería de propiedad				
Cuaderno: Principal				
N° Expediente: 2498	2008	0	1601	JR FC 04 Detalle
Sumilla: demanda de divorcio por causal				
Juzgado: no definido				
Materia: divorcio por causal				
Cuaderno: Principal				
N° Expediente: 2399	2014	0	1601	JR FC 03 Detalle
Sumilla: demanda de separación convencional				
Juzgado: no definido				
Materia: Separación convencional				
Cuaderno: Principal				
N° Expediente: 308	2015	0	1601	JP FC 03 Detalle
Sumilla: Demanda de Alimentos				
Juzgado: no definido				
Materia: Alimentos				
Cuaderno: Principal				
N° Expediente: 788	2010	0	1601	JR FC 02 Detalle
Sumilla: demanda de separación convencional				
Juzgado: no definido				
Materia: Separación convencional				
Cuaderno: Principal				
N° Expediente: 283	2015	0	1601	JR FC 01 Detalle
Sumilla: Demanda de Nulidad de Matrimonio				
Juzgado: no definido				
Materia: Nulidad de Matrimonio				
Cuaderno: Principal				

Ilustración 50: Interfaz de Resultados de Búsqueda

5.1. Interfaz de Detalle de Expediente

En la Ilustración 51: Interfaz de Detalle de Expediente, se muestra el detalle de los expedientes seleccionados. Para el ejemplo se seleccionó el primer expediente listado en los resultados de la búsqueda.

Cliente ▾ Expedientes Judiciales ▾ Documentos ▾ Reportes ▾ Usuario ▾

Editar

Detalle Seguimiento

N° Expediente: 1737 2014 0 1601 JP FC 02

Sumilla: demanda de alimentos

Juez: victor hugo camacho haro **Especialista:** gala pelaez miller

Instancia: no definido **Cuaderno:** Principal

Judicatura: 2 juzgado **Especialidad:** Paz Letrado

Distrito Judicial: La Libertad **Materia:** Alimentos

Fecha Elaboracion/Notificación: 2014-06-10 **Proceso:** Unico

Otros: César Augusto Cueva Castillo **Estado:** Demanda admitida

Partes Procesales

Demandante : jully lastenia castillo arevalo

Demandado : becerra caceres walter estuardo

Terceros :

Seguimiento del Expediente

Acto Procesal: Demanda de alimentos **Fecha Ingreso:** 2014-06-10

Descripción:

Observación: **Archivo:** [Descargar](#)

[Eliminar](#)

Seguimiento del Expediente

Acto Procesal: Resolución N° 01 **Fecha Ingreso:** 2014-07-02

Descripción: declarar inadmisibile la demanda de alimentos

Observación: plazo tres días **Archivo:** [Descargar](#)

[Eliminar](#)

Ilustración 51: Interfaz de Detalle de Expediente

6. Tabulación de las Encuestas

En esta parte se presentará la tabulación de los resultados obtenidos al aplicar la encuesta del ANEXO N° 1 abogados del Estudio Jurídico Díaz Pulido.

	P1	P2	P3	P4	P5
Sujeto 1	5	4	5	5	5
Sujeto 2	5	4	5	5	5
Sujeto 3	4	4	5	5	4
Sujeto 4	4	5	4	5	4
Sujeto 5	5	4	3	5	4
Sujeto 6	5	5	4	5	5

Tabla 7: Tabulación de Encuesta

7. Tabulación de los Tiempos Obtenidos

A continuación se muestran los tiempos obtenidos en minutos y segundos de las Búsquedas realizadas manualmente, y las que usan el Sistema de Búsqueda Inteligente. Cabe resaltar que los tiempos de búsqueda han sido realizados a los mismos expedientes objetivos, pero por personas diferentes con las herramientas ya señaladas.

7.1. Medición de Tiempos

Aquí se muestran los tiempos obtenidos de las búsquedas de expedientes.

7.1.1. Búsqueda Manual

En esta tabla se muestran los tiempos que tardó la secretaria para encontrar los expedientes solicitados por los abogados, los cuales se encontraban en el almacén de expedientes del Consultorio Jurídico.

Búsqueda N°	Tiempo de Búsqueda		Búsqueda N°	Tiempo de Búsqueda	
	Minutos	Segundos		Minutos	Segundos
1	43.67	1444	26	29.45	2209
2	5.31	257	27	32.41	1260
3	22.54	2755	28	4.97	382
4	32.37	358	29	3.58	920
5	17.35	2594	30	3.87	2437
6	21.07	1459	31	36.69	668
7	9.02	659	32	27.02	720
8	18.16	1886	33	22.99	2363
9	34.64	1564	34	16.04	1820
10	5.61	2010	35	26.99	1699
11	42.11	1025	36	18.88	1901
12	24.07	1553	37	6.25	1408
13	40.02	475	38	41.97	2191
14	38.49	872	39	42.59	1837
15	4.45	2079	40	11.35	413
16	7.17	1902	41	30.05	2131
17	8.71	586	42	32.72	1337
18	5.08	2511	43	10.95	2309
19	13.8	670	44	31.33	1538
20	44.7	181	45	6.52	1501
21	15.62	1375	46	41.4	2526
22	27.97	605	47	25.15	1616
23	12.28	2236	48	20.34	2147
24	9.77	2425	49	25.25	837
25	12.22	1923	50	35.64	521

Tabla 8: Tiempos de Búsqueda Manual

7.1.2. Búsqueda Automática

En la siguiente tabla se muestran los tiempos registrados que tardan los abogados en localizar un expediente utilizando el sistema inteligente de búsqueda.

Búsqueda N°	Tiempo de Búsqueda		Búsqueda N°	Tiempo de Búsqueda	
	Minutos	Segundos		Minutos	Segundos
1	0.3	18	26	0.43	26
2	0.17	10	27	0.48	29
3	0.8	48	28	0.82	49
4	0.4	24	29	0.73	44
5	0.32	19	30	0.67	40
6	0.42	25	31	0.78	47
7	0.23	14	32	0.22	13
8	0.27	16	33	0.08	5
9	0.17	10	34	0.75	45
10	0.53	32	35	0.52	31
11	0.23	14	36	0.48	29
12	0.62	37	37	0.42	25
13	0.72	43	38	0.67	40
14	0.68	41	39	0.38	23
15	0.53	32	40	0.53	32
16	0.52	31	41	0.23	14
17	0.57	34	42	0.18	11
18	0.33	20	43	0.08	5
19	0.33	20	44	0.77	46
20	0.43	26	45	0.13	8
21	0.72	43	46	0.83	50
22	0.1	6	47	0.45	27
23	0.55	33	48	0.55	33
24	0.48	29	49	0.78	47
25	0.83	50	50	0.38	23

Tabla 9: Tiempo de Búsqueda Automático

V. DISCUSIÓN

1. Gráfico de Evaluación de Encuesta

En el siguiente gráfico (Gráfico de Barras) se muestra los resultados obtenidos a raíz de los datos tabulados de la encuesta.

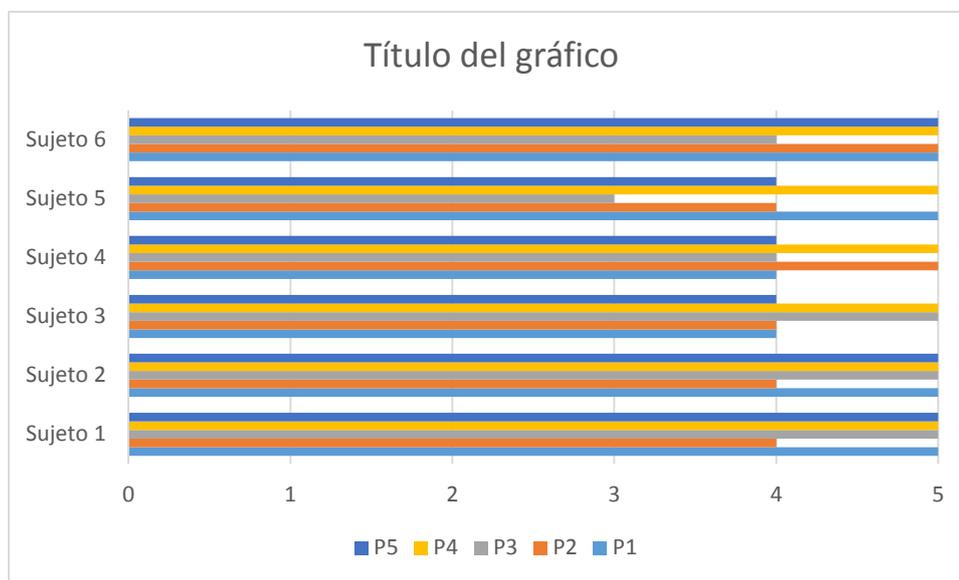


Ilustración 52: Gráfico Descriptivo de la Encuesta

La Ilustración 52: Gráfico Descriptivo de la Encuesta, se aprecia que tipo de evaluación dieron cada uno de los abogados del estudio jurídico.

2. Procesamiento Estadístico de la Encuesta

Para efectos de definir si el prototipo es de fácil uso y, a su vez, eficiente. Debemos hallar la moda y la media del puntaje general otorgado por cada abogado encuestado.

2.1. Promedio de Puntaje

En la Tabla 10: Calificación promedio por Sujeto Evaluado, se visualiza el promedio de la calificación otorgada por los sujetos evaluados. Hecho esto con el fin de establecer un punto único de evaluación para el prototipo.

Sujeto	Calificación
Sujeto 1	4.8
Sujeto 2	4.8
Sujeto 3	4.4
Sujeto 4	4.4
Sujeto 5	4.2
Sujeto 6	4.8

Tabla 10: Calificación promedio por Sujeto Evaluado

2.2. Análisis de Resultados Promedio

Dado que los datos tabulados en la Tabla 7: Tabulación de Encuesta, son datos simples, se optó por un análisis descriptivo simple. Obteniendo los siguientes resultados.

Moda	Media Aritmética
4.8	4.57

Tabla 11: Análisis Estadístico Descriptivo

3. Gráfico Comparativo de Tiempos

En el siguiente gráfico (Polígono Comparativo) se muestra la diferencia de tiempos al encontrar un Expediente Judicial, utilizando los métodos.

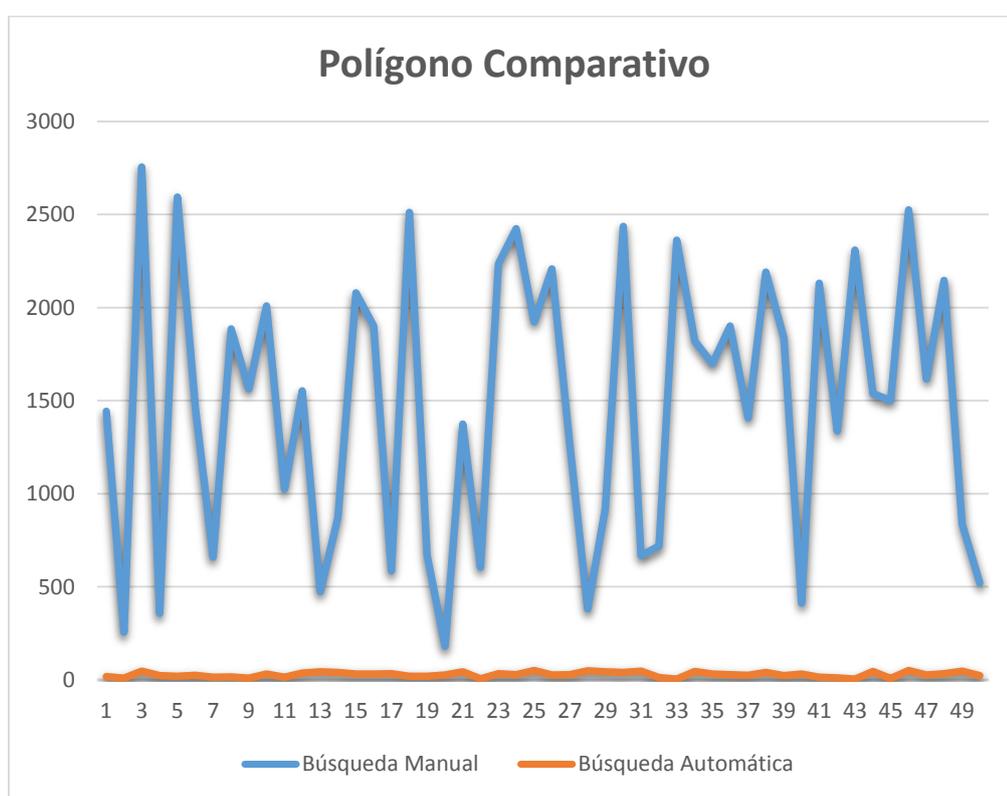


Ilustración 53: Polígono comparativo de Tiempos de Búsqueda

La Ilustración 53: Polígono comparativo de Tiempos de Búsqueda, representa una diferencia considerable entre los métodos de Búsqueda Manual y Búsqueda Automática.

4. Procesamiento Estadístico de los Resultados Obtenidos

Ahora se procede a realizar el análisis estadístico de los resultados con el objetivo de verificar si la hipótesis planteada en esta investigación es correcta.

4.1. Hipótesis Estadística

Esta hipótesis estadística se plantea en base a la hipótesis investigativa, la cual representa una manera cuantificable de demostrar la evaluación de esta última.

- **H₁: Existe** una diferencia significativa entre la media de tiempos de Búsqueda Manual y Búsqueda Automática.
- **H₀: No Existe** una diferencia significativa entre la media de tiempos de Búsqueda Manual y Búsqueda Automática.

4.2. Nivel de Error

Este es el nivel que se está dispuesto a tolerar durante el procesamiento estadístico de nuestros resultados.

$$\alpha = 5\% = 0.05$$

Ilustración 54: Nivel Alfa

4.3. Elección de la Prueba Estadística

El escenario de evaluación es: Se evalúa dos grupos de datos, Búsqueda Manual y Búsqueda Automática, que poseen datos diferentes, lo que arroja los siguientes datos.

- Tipo de trabajo: Muestras Relacionadas, puesto que resultados obtenidos buscando el mismo expediente en cada caso.
- Cantidad de Grupos: 2 grupos, por los tipos de búsqueda.
- Tipo de variables: Numéricas, los resultados son cuantificables.

Variable Aleatoria / Variable Fija		PRUEBAS NO PARAMÉTRICAS			PRUEBAS PARAMÉTRICAS
		Nominal Dicotómica	Nominal Politómica	Ordinal	Numérica
Estudio Transversal / Muestras Independientes	Un grupo	X ² Bondad de Ajuste Binomial	X ² Bondad de Ajuste	X ² Bondad de Ajuste	T de Student Una muestra
	Dos grupos	X ² Bondad de Ajuste Corrección de Yates Test de Fisher	X ² Bondad de Homogeneidad	U Mann – Withney	T de Student Muestras Independientes
	Más de dos grupos	X ² Bondad de Ajuste	X ² Bondad de Ajuste	H Kruskal - Wallis	ANOVA con un factor INTERsujetos
Estudio Longitudinal / Muestras Relacionadas	Dos medidas	Mc Nemar	Q de Cochran	Wilcoxon	T de Student Muestras relacionadas
	Más de dos medidas	Q de Cochran	Q de Cochran	Friedman	ANOVA para medidas repetidas INTRASujetos

Tabla 12: Elección de la Prueba Estadística

4.4. Lectura del P-Valor

4.4.1. Normalidad:

Se debe corroborar que la **variable aleatoria** (VA) en ambos grupos se distribuye normalmente. Para ello se utiliza la prueba de Kolmogorov-Smirnov (K-S) cuando las muestras son grandes (>30 individuos). El criterio para determinar si la VA se distribuye normalmente es:

- P-Valor $\Rightarrow \alpha$ se acepta h_0 : Los datos provienen de una distribución normal.
- P-Valor $< \alpha$ se acepta h_1 : Los datos NO provienen de una distribución normal.

Pruebas de normalidad

		Kolmogorov-Smirnov ^a		
		Estadístico	gl	Sig.
TiempoSegundos	Manual	,109	50	,191
	Automatico	,088	50	,200*

Tabla 13: Pruebas de Normalidad

NORMALIDAD Calificaciones		
P-Valor (Manual) = 0.191	>	$\alpha=0.05$
P-Valor (Automático) = 0.200	>	$\alpha=0.05$
Conclusión: La variable calificación en ambos grupos se comporta normalmente.		

Tabla 14: Calificaciones de Normalidad

De acuerdo a la conclusión obtenida del análisis de Normalidad, los datos tabulados en los resultados tienen coherencia, lo que permite seguir con el análisis estadístico.

4.4.2. Igualdad de Varianza

Se realiza la prueba de Levene, donde se corrobora la diferencia de varianza entre los grupos.

- P-Valor $\Rightarrow \alpha$ se acepta H_0 : Las varianzas son iguales.
- P-Valor $< \alpha$ se acepta H_1 : Existe diferencia significativa entre las varianzas.

Prueba t para medias de dos muestras emparejadas		
	Manual	Automático
Media	1481.90	28.31
Varianza	544828.74	179.67
Observaciones	50	50
Coefficiente de correlación de Pearson	0.044	
Diferencia hipotética de las medias	0.0	
Grados de libertad	49	
Estadístico t	13.934	
P(T<=t) una cola	5.7×10^{-19}	
Valor crítico de t (una cola)	1.6766	
P(T<=t) dos colas	1.14×10^{-18}	
Valor crítico de t (dos colas)	2.01	

Tabla 15: Resultados T de Student

Evaluación de T de Student		
P-Valor = 1.14×10^{-18}	<	$\alpha = 0.05$
Varianza de Búsqueda Manual		Varianza de Búsqueda Automática
544828.74	≠	179.67
Conclusión: SI Existe una diferencia significativa entre la media de Tiempos de Búsqueda Manual y Automático		

Tabla 16: Evaluación de Resultados T de Student

La evaluación T de Student refleja que si existe una diferencia significativa entre los tiempos de búsqueda obtenidos de cada tipo de tabulación, con los métodos que se disponen.

5. Demostración de la Hipótesis

La hipótesis de esta tesis consta de dos variables independientes:

- Variable X: si el sistema es usable y satisface las necesidades de los usuarios, da como resultado verdadero.

Ya que el resultado de la encuesta aplicada arroja como resultado 4.57 como media, en una base de evaluación de 5, se obtiene como resultado lógico de esta variable independiente el valor verdadero.

- Variable Y: si el sistema es veloz y mejora considerablemente los tiempos de búsquedas de los expedientes, la variable da como resultado verdadero.

Partiendo del hecho que la prueba T de Student, retorna que si existe una diferencia significativa entre los tiempos de búsqueda de expedientes judiciales, eso quiere decir que el valor lógico de esta variable independiente es verdadero.

La hipótesis está reflejada por la siguiente fórmula lógica:

$$X \wedge Y$$

Por lo tanto, ya que la variable X es verdadera y la variable Y es verdadera, entonces se obtiene como resultado que la hipótesis de trabajo es verdadera, como resultado de la operación lógica.

VI. CONCLUSIONES

1. De acuerdo al modelamiento de procesos realizado, se concluye que: La búsqueda manual, aparte de ser una actividad que representa un cuello de botella en las actividades, tiene un deficiente funcionamiento producto de la mala organización en los almacenes de expedientes.
2. El prototipo de buscador semántico utilizado es bastante sencillo, puesto que consta de 2 interfaces (Ilustración 44 e Ilustración 45), 1 casos de uso (Administración de Expedientes) y una clase en su procesamiento (Expedientes).
3. Donde los resultados reflejados en la Tabla 11: Análisis Estadístico Descriptivo, relejan que la calificación general dada al prototipo del sistema inteligente de búsqueda es de 4.8 en la Moda y de 4.57 y la Media, ambas en base 5 de evaluación. Se concluye que el prototipo cumple con su función, y es sencillo de utilizar.
4. De acuerdo a las pruebas realizadas, y su procesamiento estadístico con T de Student, se ha encontrado una reducción de tiempo media de 24,2 minutos, lo que representa una reducción del 98.1% del tiempo medio de búsqueda, lo cual resalta una diferencia significativa en el tiempo de Búsqueda Manual y la Búsqueda Automática.

VII. RECOMENDACIONES

1. Siempre utilizar los conversores de operadores a MongoDB, de acuerdo al lenguaje de programación utilizado.
2. Usar bases de datos planas para eventos de acceder a la información, ya que son considerablemente más rápidas en comparación con las base de datos estructuradas.
3. Utilizar una combinación fusión de ICONIX y Methontology para crear un sistema inteligente de búsqueda, da como resultado un buscador eficiente que satisface los requerimientos establecidos.
4. No forzar la realización de todas las tareas de la metodología Methontology, puesto que no todas las tareas se aplican a todos los entornos a desarrollar.
5. Utilizar WSDL para manejar la inteligencia de los sistemas web, puesto que permite manejar información de manera más fácil y rápida.

VIII. REFERENCIAS BIBLIOGRÁFICAS

1. Aulet, J. (09 de Noviembre de 2011). *SINDIKOS*. Obtenido de SINDIKOS: <http://www.sindikos.com/2011/11/introduccion-a-las-bases-de-datos-nosql-mongodb/>
2. Berners-Lee, T., Hendler, J. A., & Lassila, O. (2001). *The Semantic Web*. Scientific American.
3. BIZAGI. (s.f.). <http://www.bizagi.com/>. Recuperado el Noviembre de 2014, de BPMN: Business Process Model Notation: <http://www.bizagi.com/esp/descargas/BPMNbyExample.pdf>
4. Christensen, E., Curbera, F., Merendith, G., & Weerawarana, S. (15 de Marzo de 2001). *Web Services Description Language (WSDL) 1.1*. Obtenido de Web Services Description Language (WSDL) 1.1: <http://www.w3.org/TR/wsdl>
5. d'Aquin, M., & Motta, E. (2011). Watson, more than a Semantic Web search. *ACM DL Digital Library*, 55-63. Obtenido de <http://iospress.metapress.com/content/p4847844472v4381/fulltext.pdf>
6. De San Martin Oliva, C. (2011). Metodologia Iconix. *Portalzonda - U Revista Universidad de San Juan*, 10. Obtenido de <http://www.unsj-cuim.edu.ar/portalzonda/seminario08/archivos/MetodologiaCONIX.pdf>
7. Ding, L., Finin, T., Joshi, A., Pan, R., R. Scott, C., Peng, Y., . . . Sachs, J. (2004). *Swoogle: A Semantic Web Search and Metadata Engine*. University of Maryland Baltimore County, Baltimore MD 21250, USA: Department of Computer Science and Electronic Engineering. Obtenido de http://ebiquity.umbc.edu/_file_directory_/papers/116.pdf
8. ECMA-404 The JSON Data Interchange Standard. (2015). *Introducción a JSON*. Obtenido de <http://www.json.org/json-es.html>
9. Ecma-International. (Octubre de 2013). *Ecma-International*. Obtenido de Standard ECMA-404: <http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf>
10. Espinoza Aguirre, C. B. (2012). *Comparativa de Frameworks para el desarrollo de aplicaciones con PHP*. Cuenca, Ecuador.
11. Fabbri, M. (2006). *Técnicas de Investigación: La Observación*. (Universidad Nacional de Rosario) Recuperado el Noviembre de 2014, de <http://www.fhumyar.unr.edu.ar/>: <http://www.fhumyar.unr.edu.ar/escuelas/3/materiales%20de%20catedras/trabajo%20de%20campo/solefabri1.htm>
12. Fernandez, R. (03 de febrero de 2014). *Genbeta Desarrollo y software*. Obtenido de <http://www.genbetadev.com/bases-de-datos/mongodb-que-es-como-funciona-y-cuando-podemos-usarlo-o-no>

13. García, E. (12 de 08 de 2011). *Informática V - ICONIX*. Obtenido de Informática V - ICONIX: <http://informatica-v-iconix.blogspot.com/2011/08/normal-0-21-false-false-false-es-x-none.html>
14. Gomez-Perez, A., Fernández-López, M., & Corcho-Garcia, O. (2004). Ontological engineering. *Computing Reviews*.
15. Hidalgo Delgado, Y., & Rodriguez Puente, R. (01 de 03 de 2013). SciELO. *Revista Cubana de Ciencias Informáticas.*, 10. Obtenido de Revista Cubana de Ciencias Informáticas: http://scielo.sld.cu/scielo.php?pid=S2227-18992013000100009&script=sci_arttext
16. *Interacciones*. (19 de 12 de 2007). Recuperado el 12 de 01 de 2009, de <http://www.interacciones.com.ar/tag/web-semantic/>
17. *La web semántica, la siguiente generación de webs*. (04 de 04 de 2007). Recuperado el 15 de 12 de 2008, de <http://sociedaddelainformacion.telefonica.es/jsp/articulos/impresion.jsp?elem=4299>
18. Lei, Y., Uren, V., & Motta, E. (2006). *SemSearch: A Search Engine for the Semantic*. U.K Inglaterra: Knowledge Media Institute (KMi), The Open University, Milton Keynes. Obtenido de <http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=20E61BC376D66E18BE93188767ED9273?doi=10.1.1.98.3235&rep=rep1&type=pdf>
19. Manjarrez Antaño, A. C., Martínez Castro, J. M., & Cuevas Valencia, R. E. (2014). Migración de Bases de Datos SQL a NoSQL. *Revista Tamati Sabiduria*, 144-148.
20. MongoDB. (05 de Marzo de 2015). *MongoDB*. Obtenido de MongoDB: <http://docs.mongodb.org/manual/core/introduction/>
21. Niño Zambrano, M. A., López Gómez, I. D., Andrade, C. A., Cobos Lozada, C. A., & Fabregat Gesa, R. (15 de 06 de 2012). Modelo Semantico de Expansión de Consultas para Búsqueda Web (MSEC). *UIS Ingenierías - Revista de la Facultad de Ingenierías Fisicomecánicas*, 10.
22. Requena, C. (Ed.). (10 de Abril de 2010). Obtenido de Nosql.es: <http://www.nosql.es/blog/nosql/mongodb.html>
23. Salazar Paredes, E. F., & Sanchez Macas, F. F. (2013). *Desarrollo de una Aplicación WEB para Servicio de Entrega de Comida a Domicilio*. Quito: Escuela Politécnica Nacional.
24. Sebastián Bayard Ocares , Ó. (01 de 03 de 2011). *bpmn-bayard*. Obtenido de bpmn-bayard: <http://bpmn-bayard.blogspot.com/2011/03/1-introduccion-bpmn.html>
25. Sedano, M. C. (Marzo de 2010). *Repositorio Digital de Tesis PUCP*. Obtenido de http://tesis.pucp.edu.pe/repositorio/bitstream/handle/123456789/370/ESCOBAR_MAYR

A_ANALISIS_DISEÑO_IMPLEMENTACION_SISTEMA_APOYO_SEGUIMIEN
TO.pdf?sequence=1

26. Solvetic. (14 de Febrero de 2014). *Convertir bases de datos relacionales y SQL para MongoDB*. Obtenido de <http://www.solvetic.com/tutoriales/article/468-convertir-bases-de-datos-relacionales-y-sql-para-mongodb/>
27. Suárez Barón, M., & Salinas Valencia, K. (12 de 03 de 2009). Una aproximación a la indexación semántica y a la recuperación de información. *Revista Facultad de Ingeniería de Antioquía*, 15.

IX. Anexos

1. Anexo N° 1: Encuesta de Usabilidad y Eficiencia

Encuesta Estudio Jurídico Díaz Pulido

Marque la alternativa según su apreciación del sistema, los valores están indicados de menor a mayor.

1. ¿Cómo valorarías el funcionamiento del buscador para encontrar los expedientes?
 1 2 3 4 5
2. ¿La información solicitada se mostró de forma ordenada?
 1 2 3 4 5
3. ¿La búsqueda de los expedientes fue sencilla?
 1 2 3 4 5
4. ¿La aplicación muestra de manera rápida la información?
 1 2 3 4 5
5. ¿Fue sencillo adaptarse al sistema?
 1 2 3 4 5