



Universidad Central de Venezuela
Facultad de Ciencias
Escuela de Computación
Centro de Ingeniería de Software y Sistemas
Laboratorio de Inteligencia Artificial
Opción Profesional: Inteligencia Artificial

DESARROLLO DE UNA APLICACIÓN EN AMBIENTE WEB PARA LA VISUALIZACIÓN DE LA ONTOLOGÍA DE INSECTOS ACUÁTICOS

**Trabajo Especial de Grado Presentado ante la Ilustre Universidad Central de
Venezuela, por la Bachiller Yessika Labrador, para optar al Título de Licenciado en
Computación**

Tutora: Esmeralda Ramos

Tutora: Haydemar Núñez

Caracas, Octubre 2007

A mis padres, a mis hermanas y a mi novio.

AGRADECIMIENTOS

Quiero agradecer primeramente a Dios, por darme la dicha de vivir, de tener salud y de tener una familia maravillosa.

A mis padres, Argenis y Yudimar, por cuidarme y amarme, por apoyarme y enseñarme desde pequeña que puedo lograr las cosas que de verdad quiero. Gracias por su apoyo.

A mis hermanas, Yelimar y Yessire, por siempre hacerme reír, hasta en los malos momentos, y por recordarme en cada instante que tengo una familia maravillosa.

A Alvaro, por ser mi apoyo incondicional, gracias por estar a mi lado en toda la carrera, por ayudarme a superar mis miedos y por hacerme una mejor persona. No sabes cuanto agradezco el que hayas llegado a mi vida.

A mis amigos, por apoyarme y por quererme, por estar conmigo en todo momento y por poder escucharme cuando de verdad lo necesito.

A mis tutoras, Haydemar y Esmeralda, por guiarme en este camino tan largo de la tesis, por la paciencia y dedicación que tuvieron para conmigo y mi trabajo. Me encantó trabajar con ustedes.

En fin, gracias a todas las personas que han intervenido de alguna u otra forma en mi trabajo y en mi desempeño, de verdad quiero darle gracias a todos.

RESUMEN

En los estudios de los ecosistemas de aguas corrientes los macroinvertebrados acuáticos han recibido una gran atención, principalmente por su importancia como indicadores biológicos. La razón es que estos insectos presentan una alta sensibilidad a los cambios ambientales, por lo que se utilizan como indicadores de la calidad del agua de los sistemas acuáticos continentales (ríos y lagos). El proceso de identificación de estas especies está limitado principalmente por la falta de claves adecuadas a la fauna acuática de los trópicos, lo cual a su vez refleja la carencia de expertos en el área. Adicionalmente, la información disponible se encuentra dispersa y en medios de difícil acceso, haciendo que a muchos de los investigadores se les dificulte el proceso de estudio e investigación.

Como una forma de solventar estas dificultades, en un trabajo previo, se desarrolló una Ontología en el dominio de los insectos acuáticos. Para acceder a este conocimiento de manera sencilla y desde cualquier ubicación geográfica, se presenta en este trabajo una aplicación de visualización para ambiente Web, la cual le ofrece a los usuarios funcionalidades como: navegar en la Ontología y consultar el contenido de la misma de forma gráfica y textual, visualizar imágenes asociadas a instancias específicas, descargar documentos, y acceder a sitios de interés asociados con el dominio. La aplicación también cuenta con un módulo de administración de contenido para mantener actualizada la información en la aplicación, siendo ésta los enlaces, documentos y la Ontología.

Palabras Claves

Ontología, insectos acuáticos, aplicación Web, visualización de ontología, OWL, Protégé, Jena.

ÍNDICE GENERAL

INTRODUCCIÓN.....	1
CAPÍTULO I. MARCO TEÓRICO.....	3
1.1 Aspectos Básicos de las Ontologías	3
1.1.1 Tipos de Ontologías.....	4
1.1.2 Componentes de una Ontología.....	6
1.1.3 Utilidad de las Ontologías	6
1.2 Lenguajes de Ontologías	7
1.2.1 RDF (Resource Description Framework).....	7
1.2.2 DAML+OIL (DARPA Agent Markup Language) y OIL (Ontology Inference Layer)	8
1.2.3 OWL (Web Ontology Language).....	8
1.2.4 Otros lenguajes	10
1.3 Editores de Ontologías	11
1.3.1 Protégé.....	11
1.3.2 Ontolingua	12
1.3.3 WSMO Studio	12
1.3.4 Otros	13
1.4 Implantación de Ontologías en ambiente Web.....	13
1.4.1 La Web Semántica.....	14
1.4.2 Funcionamiento de la Web Semántica	14
1.4.3 Arquitectura de la Web Semántica	17
1.4.4 Utilidad de las Ontologías para la Web	19
1.4.5 Visualización de Ontologías.....	21
CAPÍTULO II. MARCO APLICATIVO.....	23
2.1 Planteamiento del Problema	23
2.2 Solución Propuesta	24

2.3	Planteamiento del Problema	25
2.3.1	Objetivo General.....	25
2.3.2	Objetivos Específicos	25
2.4	Desarrollo de la Aplicación	25
2.4.1	Identificación y Análisis de los Requerimientos Funcionales.....	26
2.4.2	Establecimiento de los aspectos Técnicos	49
2.4.3	Establecimiento de tecnologías a usar	50
2.4.4	Arquitectura de la Aplicación.....	53
2.4.5	Diseño del modelo de navegación entre páginas.....	55
2.4.6	Definición de la Base de Datos.....	55
2.4.7	Estructura lógica de desarrollo de la Aplicación	58
2.4.8	Diseño de la Interfaz de Usuario	59
2.4.9	Implementación de las funcionalidades.....	63
	CONCLUSIONES Y RECOMENDACIONES	76
	REFERENCIAS	78
	ANEXO 1: GLOSARIO DE TÉRMINOS	83
	ANEXO 2: DIAGRAMA DE ROBUSTEZ	85
	ANEXO 3: INSTALACIÓN DE SOFTWARE	90

ÍNDICE DE FIGURAS

Figura 1: Flujo general del conocimiento. (www.semanticweb.org)	15
Figura 2: Arquitectura de la Web Semántica. (Berners-Lee, 2001).	17
Figura 3: Mapa conceptual de la Web Semántica. (Rodríguez y Ronda, 2005).....	19
Figura 4: Diagrama de CU – Módulo de Administración.	28
Figura 5: Diagrama de CU – Sub Módulo Administrar Ontología.	31
Figura 6: Diagrama de CU – Sub Módulo Gestionar Mensajes.	32
Figura 7: Diagrama de CU – Sub Módulo Administrar Documentos.	33
Figura 8: Diagrama de CU – Sub Módulo Administrar Enlaces.	36
Figura 9: Diagrama de CU – Sub Módulo Administrar Contenido Actual.	38
Figura 10: Diagrama de CU – Sub Módulo Administrar Historial de Ontologías.	42
Figura 11: Diagrama de CU – Sub Módulo Administrar Modificar Instancia.	43
Figura 12: Diagrama de CU –Módulo de Acceso Público.	45
Figura 13: Arquitectura de la Aplicación.	54
Figura 14: E/R Entidad Administrador.....	55
Figura 15: E/R Entidad Documentos.....	55
Figura 16: E/R Entidad Enlaces.	56
Figura 17: E/R Entidad Imágenes.....	56
Figura 18: E/R Entidad Mensajes.	56
Figura 19: E/R Relación Administrar Documentos.....	57
Figura 20: E/R Relación Administrar Enlaces	57
Figura 21: E/R Relación Administrar Imágenes.....	57
Figura 22: E/R Relación Responder Mensajes	57
Figura 23: Diagrama Entidad – Relación.	58
Figura 24: Ejemplo de Interfaz de la Aplicación.....	62
Figura 25: Ejemplo de iconos de la Aplicación.....	63
Figura 26: Ejemplo de iconos de la Aplicación: Eliminar, Editar y Descargar.....	63
Figura 27: Página de inicio.....	64
Figura 28: Página para la visualización de la Ontología de Insectos Acuáticos	65

Figura 29: Ejemplos de las formas de visualización: Gráfica y Textual	65
Figura 30: Ejemplo de información mostrada al usuario referente a la Instancia Anacroneuria Bari	66
Figura 31: Ejemplo de una consulta sobre la Ontología.....	67
Figura 32: Pantalla de Recursos	67
Figura 33: Pantalla Contáctenos.....	68
Figura 34: Pantalla de inicio para el módulo de Administración	69
Figura 35: Pantalla de para la modificación de datos del Administrador.....	70
Figura 36: Pantalla de entrada para administrar enlaces.	71
Figura 37: Pantalla de entrada para administrar documentos.....	72
Figura 38: Marco de trabajo para modificar y/o agregar instancias.	73
Figura 39: Formulario para modificar y/o agregar instancias.	73
Figura 40: Pantalla de carga de Ontología.....	74
Figura 41: Pantalla de consulta de mensajes recibidos	75

ÍNDICE DE ANEXOS

ANEXO 1: GLOSARIO DE TÉRMINOS	83
ANEXO 2: DIAGRAMA DE ROBUSTEZ	85
ANEXO 3: INSTALACIÓN DE SOFTWARE	90

INTRODUCCIÓN

La conservación de los ecosistemas y su biodiversidad han recibido mucho interés en los últimos años y actualmente es considerado en el ámbito mundial un tema prioritario. En general, la variación en las características físico-químicas del agua han sido utilizadas para determinar su calidad. Sin embargo, en los últimos años se ha propiciado el uso de indicadores biológicos por presentar una serie de ventajas en relación a las mediciones químicas. Entre estos indicadores resaltan los insectos acuáticos principalmente por poseer un ciclo de vida corto que permite cuantificar los cambios que se suceden en dichos ecosistemas (Ramírez A. 2006, Wetzel 2000, Rincón y Cressa 2000, Cressa 1994, Resh y Rosenberg 1984).

A pesar de la importancia de los insectos acuáticos, la información existente en nuestro país, es escasa y se encuentra dispersa, por lo que se planteo en un trabajo anterior (Chaudary y Scarioni, 2005) la creación de una Ontología en este dominio, con la finalidad de que el conocimiento pudiera ser reutilizado y compartido por la comunidad interesada.

El tener una Ontología como repositorio central del conocimiento nos ofrece diversos beneficios, sin embargo surge un nuevo problema, el cual consiste en cómo hacer que la información llegue hasta los expertos y personas interesadas en el área de una manera fácil de visualizar para ellos y sin ninguna limitante geográfica.

El presente trabajo busca resolver esta inquietud mediante el diseño de una aplicación Web, la cual permita a las personas interesadas en la información contenida en la Ontología poder visualizarla de manera sencilla.

Este trabajo está dividido en dos Capítulos, el primero de ellos tiene como finalidad definir y explicar el contexto en el cuál se desarrolló la aplicación, las ventajas que nos aportan las Ontologías y la importancia de las mismas dentro del ambiente Web.

El segundo capítulo, comprende la estructuración de la aplicación, presentandose los diversos requerimientos obtenidos durante la fase de levantamiento de información y, el diagrama de

casos de uso que refleja los mismos. Igualmente, se describe la arquitectura del sistema, cómo está estructurada la base de datos, una breve explicación sobre la interfaz definida y se muestran imágenes de la aplicación de visualización explicando las funcionalidades del sistema.

Al final del trabajo se presentan las conclusiones alcanzadas y las recomendaciones que se hacen para trabajos futuros. Se anexa el diagrama de robustez (también denominado diagrama de análisis del dominio), el cual es importante para definir la interacción de componentes y páginas con el usuario y la base de datos. Por último, se anexa un glosario de términos con definiciones referentes al contexto del presente trabajo.

CAPÍTULO I

MARCO TEÓRICO

1.1 Aspectos Básicos de las Ontologías

La palabra Ontología esta compuesta por los términos griegos ontos “el ser” y logos “estudio de”, por lo cual se puede definir como el estudio del ser en general y de sus partes trascendentales. Algunas autores la definen como:

- La parte de la filosofía que investiga en qué consiste el ser y cuáles son los ámbitos o regiones fundamentales del ser (Echegoyen, 2006).
- El estudio del ente en cuanto a tal, por ello es llamada la teoría del ser, es decir, el estudio de las cosas: qué es, cómo es y cómo es posible. La Ontología se ocupa de establecer las categorías fundamentales o modos generales de ser de las cosas (Wikipedia, Ontología).

El concepto de Ontología en el área de computación tiene un enfoque diferente, algunas de sus definiciones se presentan a continuación.

- La Ontología es una especificación de una conceptualización de un dominio del conocimiento. Es un vocabulario controlado que describe objetos y las relaciones entre ellos de una manera formal, y tiene una gramática para usar los términos del vocabulario para expresar algo significativo dentro de un dominio de interés. El vocabulario se utiliza para hacer preguntas y aserciones (Lamarca, 2006).
- Una Ontología es una especificación explícita y formal, que sigue unas reglas lógicas y semánticas, y que se supone contiene el significado informativo y las relaciones existentes en la conceptualización; conceptualización que corresponde a una parte del mundo o universo que es objeto de tratamiento (Tramullas, 1999).
- El objetivo de las Ontologías en la Inteligencia Artificial es representar procesos del tipo no algorítmicos (procesos que no usan algoritmos lógicos para su desarrollo) con el fin de dotar a los sistemas inteligentes de la suficiente flexibilidad que les permita reproducir, en alguna medida, los procesos cognitivos que se llevan a cabo en la mente humana (Ramírez Z., 2006).

Algunas características de las Ontologías se presentan a continuación:

- **Pueden existir Ontologías múltiples:** el propósito de una Ontología es hacer explícito algún punto de vista, por lo que, a veces, será preciso combinar dos o más Ontologías. Cada Ontología introducirá conceptos específicos.
- **Podemos identificar niveles de abstracción de las Ontologías:** estos niveles de generalización o abstracción ofrecen una topología de Ontologías. La idea es caracterizar una red de Ontologías usando multiplicidad y abstracción, y puesto que no se puede aspirar a tener una descripción completa del mundo, podemos pensar en una estrategia de construcción gradual de abajo hacia arriba.
- **Multiplicidad de la representación:** un concepto puede ser representado de muchas formas, por lo que pueden coexistir múltiples representaciones de un mismo concepto.
- **Mapeo de Ontologías:** establecer relaciones entre los elementos de una o más Ontologías, para establecer conexiones, especializaciones, generalizaciones, etc.

Noy y McGuinness (2002), plantean que las Ontologías tienen como objetivos principales los siguientes:

- Compartir la comprensión común de la estructura de información entre personas o agentes de software, lo que debe revertirse de forma positiva y casi necesaria en la extracción y recuperación de información, por ejemplo en páginas Web de contenidos conectados temáticamente.
- Permitir la reutilización del conocimiento perteneciente a un dominio.
- Hacer explícitos los supuestos de un dominio.
- Separar el conocimiento de un dominio del conocimiento que se puede denominar operacional.
- Hacer posible el análisis del conocimiento de un campo, en lo que se refiere al estudio de los términos y relaciones que lo integran ya sea formalmente o no.

1.1.1 Tipos de Ontologías

Steve et al. (1997) distinguen tres tipos fundamentales de Ontologías:

- **Ontologías de un dominio**, en las que se representa el conocimiento especializado pertinente de un dominio o subdominio, como la medicina, las aplicaciones militares, la cardiología, etc.
- **Ontologías genéricas**, en las que se representan conceptos generales y funcionales del conocimiento, como las estructuras parte/todo, la cuantificación, los procesos o los tipos de objetos.
- **Ontologías representacionales**, en las que se especifican las conceptualizaciones que subyacen a los formalismos de representación del conocimiento, por lo que también se denominan *meta-ontologías* (*meta-level* o *top-level ontologies*).

A estos tipos de Ontologías se les añaden las que han sido creadas para una actividad o tarea específica denominadas Ontologías de tarea (*task ontologies*), como por ejemplo, la venta de productos o el diagnóstico de una enfermedad y las Ontologías creadas para una aplicación específica.

Sin embargo es posible encontrar otras clasificaciones según diversos aspectos (Lamarca, 2006), como por ejemplo:

- Según el tipo de agente al que vayan destinadas las Ontologías pueden ser:
 - **Ontologías lingüísticas**: se vinculan a aspectos lingüísticos, tales como gramáticos, semánticos y sintácticos, destinados a su utilización por los seres humanos.
 - **Ontologías no lingüísticas**: destinadas a ser utilizadas por robots y agentes inteligentes.
 - **Ontologías mixtas**: combinan las características de las anteriores.
- Según el grado o nivel de abstracción y razonamiento lógico que permitan las Ontologías se pueden dividir en:
 - **Ontologías descriptivas**: incluyen descripciones, taxonomías de conceptos, relaciones entre los conceptos y propiedades, pero no permiten inferencias lógicas.
 - **Ontologías lógicas**: permiten inferencias lógicas mediante la utilización de una serie de componentes como la inclusión de axiomas, etc.

1.1.2 Componentes de una Ontología

Según Gómez et al. (2003) los componentes de una Ontología son los siguientes:

- **Clases:** son las ideas a formalizar y representan los conceptos en el sentido más amplio. Engloban a un conjunto de objetos que se pueden describir especificando las propiedades que éstos deben satisfacer para pertenecer a dicha clase. Las clases en una Ontología se suelen organizar en taxonomías a las que se les pueden aplicar los mecanismos de herencia.
- **Relaciones:** representan las interacciones entre clases, es decir, entre conceptos. Las relaciones más habituales son binarias.
- **Funciones:** son casos especiales de relaciones donde se identifican elementos mediante el cálculo de una función.
- **Instancias:** se usan para representar elementos o individuos en una Ontología, simbolizan objetos determinados de un concepto.
- **Axiomas:** sirven para modelar sentencias que son siempre ciertas. Normalmente se usan para representar conocimiento que no puede ser formalmente definido por los componentes descritos anteriormente. También son usados para verificar la consistencia de la propia Ontología.

1.1.3 Utilidad de las Ontologías

Según Tramullas (1999) las Ontologías bajo un concepto unificador de “*knowledge sharing*” pueden desempeñar varios papeles o roles, los cuales se presentan a continuación:

- Como repositorios para la organización de conocimiento e información, tanto de tipo corporativo como científico.
- Como herramienta para la adquisición de información, en situaciones en las que un equipo de trabajo la utiliza como soporte común para la organización del dominio.
- Como herramienta de referencia en la construcción de sistemas basados en conocimiento.
- Para permitir la reutilización del conocimiento ya existente, en la creación de nuevas aplicaciones.
- Como base para la construcción de lenguajes de representación del conocimiento, acompañada de la formalización del cálculo que tenga lugar entre los términos.

1.2 Lenguajes de Ontologías

Los Lenguajes de codificación de Ontologías son lenguajes que permiten representar las Ontologías y posibilitan la utilización de las mismas por parte de entes computarizados. Es importante conocer los diferentes lenguajes que existen y cuáles son los ámbitos de aplicación de cada uno de ellos, para poder usar el que mejor se adapte a nuestras necesidades. Algunos de los lenguajes usados para la representación de Ontologías son:

1.2.1 RDF (Resource Description Framework)

Se trata de un modelo para definir relaciones semánticas entre distintas URIs (*Uniform Resource Identifier*). Está basado en la sintaxis XML y permite describir semánticamente una URI asociándole un conjunto de propiedades y valores. Los modelos RDF se construyen como grafos dirigidos especificando tripletas: URI, propiedad, valor. Los metadatos especificados con RDF son comprensibles por las máquinas, y por lo tanto se pueden procesar de manera automática.

Este lenguaje está construido en base a las siguientes reglas:

- Un recurso es cualquier cosa que puede tener un URI, esto incluye todas las páginas Web, todos los elementos individuales de cada documento XML y mucho más.
- Una propiedad es un recurso que tiene un nombre y que a su vez puede usarse como una propiedad, por ejemplo autor o título.
- Una sentencia consiste en la combinación de un recurso, una propiedad y un valor. Estas partes son conocidas como el sujeto, predicado y el objeto de la sentencia.

Entre las características del lenguaje RDF podemos mencionar las siguientes (Bray, 2001):

- **Independencia:** dado que una propiedad es un recurso, todo ente independiente puede inventarlas. Podemos inventar una propiedad llamada "Autor" y otros pueden inventar una propiedad llamada "Director" que podría aplicarse, por ejemplo, a recursos asociados con películas. Esta libertad es necesaria porque no hay un ente central que se encargue de definir cuales son las propiedades a usar en todo el mundo.

- **Intercambio:** dado que las sentencias se escriben en XML pueden ser fácilmente usadas para intercambiar información.
- **Simplicidad:** las sentencias son simples (recurso, propiedad, valor) por lo que son fáciles de manejar y de usar para buscar objetos aún en volúmenes realmente grandes.

1.2.2 DAML+OIL (DARPA Agent Markup Language) y OIL (Ontology Inference Layer)

Este lenguaje nació de la cooperación entre OIL y DARPA y unifica los lenguajes DAML (DARPA's Agent Markup Language) y OIL (Ontology Inference Layer). Se basa en estándares del W3C.

El lenguaje DAML se desarrolló como una extensión del lenguaje XML y de *Resource Description Framework* (RDF), con la finalidad de extender el nivel de expresividad de RDFS. DAML- OIL hereda muchas de las características de OIL, pero se aleja del modelo basado en clases (frames) y potencia la lógica descriptiva. Es más potente que RDFS para expresar Ontologías.

La última revisión del lenguaje ofrece un gran conjunto de elementos con los cuales se pueden crear Ontologías y marcar la información para que sea legible y comprensible por las máquinas. También funciona como formato de intercambio. Sin embargo, este lenguaje presenta algunas carencias debido a su complejidad conceptual y de uso, complejidad que se intentó solventar con el desarrollo de OWL.

1.2.3 OWL (Web Ontology Language)

OWL o Lenguaje de Ontologías para la Web, es un lenguaje de etiquetado semántico para publicar y compartir Ontologías en la Web. Se trata de una recomendación del W3C, y puede usarse para representar Ontologías de forma explícita, es decir, permite definir el significado de términos en vocabularios y las relaciones entre estos términos. Es un lenguaje diseñado para usarse cuando la información contenida en los documentos necesita ser procesada por programas o aplicaciones, en oposición a situaciones donde el contenido solamente necesita ser presentado a los seres humanos (Lamarca, 2006).

OWL surge como una revisión al lenguaje DAML-OIL y es mucho más potente que éste. Al igual que OIL, OWL se estructura en capas que difieren en la complejidad y puede ser adaptado a las necesidades de cada usuario, al nivel de expresividad que se precise y a los distintos tipos de aplicaciones existentes (motores de búsqueda, agentes, etc.).

Alguna de las características del lenguaje OWL podemos encontrar (Barrón):

- Es un lenguaje Web para Ontologías compatible con la World Wide Web en general y con la Web Semántica en particular.
- Se usa cuando la información de los documentos requiere ser procesada por aplicaciones, no para presentarlos.
- Puede representar el significado de los términos en vocabularios y las relaciones entre ellos.
- La Web semántica será construida sobre la ventaja de XML de definir esquemas de etiquetado y el uso de RDF para representar datos.
- Se requiere sobre RDF un lenguaje para Ontologías capaz de describir formalmente la semántica de las terminologías usadas en documentos Web

Según el grupo de la www.w3.org el lenguaje OWL tiene tres (3) sub-lenguajes que incrementan su expresión, estos son:

- **OWL Lite:** da soporte a aquellos usuarios que primordialmente necesitan una clasificación jerárquica y restricciones simples. Por ejemplo, soporta restricciones cardinales, pero solamente permite valores cardinales de 0 ó 1. En resumen, tiene una complejidad formal más baja que OWL DL.
- **OWL DL:** OWL DL da soporte a aquellos usuarios que quieren la máxima expresividad mientras conservan completamente la computacionalidad (todas las conclusiones son garantizadas para ser computables) y resolubilidad (todas las computaciones terminarán en tiempo finito). OWL DL incluye todos los constructores del lenguaje OWL, pero pueden usarse solamente bajo ciertas restricciones (por ejemplo, mientras una clase puede usarse por una subclase de muchas clases, una clase no puede ser una instancia de otra clase).

Se denomina así debido a su correspondencia con las descripciones lógicas (DL), un campo de investigación que han estudiado los lógicos para la fundación formal de OWL.

- **OWL Full:** OWL Full da soporte a usuarios que requieren el máximo de expresividad y la libertad sintáctica de RDF sin garantías computacionales. Por ejemplo, en OWL Full una clase puede ser tratada simultáneamente como una colección de individuos y como un individuo por derecho propio. OWL Full permite a una Ontología aumentar el significado del vocabulario predefinido (RDF ó OWL). Es poco probable que algún software racional pueda soportar por completo el razonamiento para cada característica de OWL Full.

Entre estos lenguajes existen las siguientes relaciones las cuales van en un solo sentido y se presentan a continuación:

- Cada ontología bien formada de OWL Lite es una ontología bien formada de OWL DL.
- Cada ontología bien formada de OWL DL es una ontología bien formada de OWL Full.
- Cada conclusión válida OWL Lite es una conclusión válida OWL DL.
- Cada conclusión válida OWL DL es una conclusión válida OWL Full.

1.2.4 Otros lenguajes

Algunos de los muchos lenguajes que podemos encontrar a la hora de implantar una Ontología son (Lamarca, 2006):

- **KIF** (*Knowledge Interchange Format*): lenguaje para el intercambio de conocimiento entre agentes. Incluye definición de objetos, lógica de representación y semántica.
- **FLogic** (*Frame Logic*): lenguaje orientado a objetos y sistema de base de datos deductivo.
- **OCML** (*Operational Conceptual Modelling Language*): este lenguaje de modelado sostiene la construcción de modelos de conocimiento por significados de varios tipos de construcciones. Permite la especificación y operación de funciones, relaciones, clases, instancias y reglas. También incluye mecanismos para definir Ontologías y métodos para resolver problemas.

- **SHOE** (*Simple HTML Ontology Extensions*): se trata de un lenguaje de representación del conocimiento para la Web compatible con XML. Permite a los autores de las páginas anotar sus documentos Web.
- **CLIPS** (*C Language Integrated Production System*): entre otras características, permite la representación del conocimiento. CLIPS provee una herramienta para gestionar una extensa variedad de conocimiento con el soporte de tres paradigmas de programación diferentes: basado en reglas, orientado a objetos y procesal.
- **LOOM**: es un lenguaje y un entorno para construir aplicaciones inteligentes. La base de LOOM es un sistema de representación del conocimiento que es utilizado para proveer un soporte deductivo para la porción declarativa del lenguaje LOOM. El conocimiento declarativo en LOOM consiste en definiciones, reglas, preguntas y herramientas por defecto.

1.3 Editores de Ontologías

Los editores de Ontologías son herramientas que permiten la codificación de Ontologías. Su uso facilita la tarea de definición, unión y desarrollo distribuido de Ontologías. Algunas herramientas son:

1.3.1 Protégé

Protégé es un editor para construir Ontologías y un marco general para representar el conocimiento. Con la ayuda de este software se pueden construir aplicaciones que representen la información de forma ontológica (Esquivias et al., 2005).

Protégé es libre y open source, y soporta dos maneras principales de modelado ontológico: vía editores de Protégé-Frames y del Protégé-OWL. Las Ontologías se pueden exportar en una variedad de formatos incluyendo RDF(S), OWL y XML Schema. Está basado en Java, es extensible, y proporciona un ambiente de *plug-in-play* el cual lo hace un editor flexible para el desarrollo rápido.

Protégé esta apoyado por una amplia comunidad de desarrolladores y académicos, gubernaciones y usuarios corporativos, los cuales están utilizándolo para soluciones del conocimiento en áreas tan diversas como la biomedicina y modelado corporativo.

1.3.2 Ontolingua

Ontolingua es un software realizado por el *Knowledge Software Laboratory* (KSL) de la Universidad de Stanford, que sirve para crear, modificar y usar Ontologías (Sierra, 2003). Provee formas para definir clases, relaciones, funciones, objetos y teorías. Traduce definiciones escritas de manera estándar en salidas para una variedad de implementaciones de sistemas.

La sintaxis y semántica de las definiciones de Ontolingua se basan en una notación y una semántica para una versión extendida del cálculo de primer orden del predicado llamado Knowledge Interchange Format (KIF) (Genesereth y Fikes, 1992). KIF es conocido como un lenguaje para la publicación y comunicación del conocimiento.

1.3.3 WSMO Studio

WSMO Studio es un entorno que permite el modelado de una Ontología para el ambiente Web, el cual está disponible como un conjunto de *plug-ins* para Eclipse.

Entre las características de WSMO Studio se encuentran:

- Editor de elementos WSMO (ontologías, servicios, metas, mediadores).
- Diseño de Coreografías para WSMO.
- SAWSDL editor para añadir anotaciones semánticas a documentos WSDL.
- Importa y exporta a:
 - WSML
 - Un subconjunto de OWL-DL
 - RDF
 - Representación XML de WSML.
- Integra validador WSML.

- Integra WSML Reasoner para verificar la consistencia de Ontologías.
- Entre otras.

WSMO Studio esta disponible bajo una licencia LGPL (*Lesser General Public License*).

1.3.4 Otros

Algunos editores disponibles en la Web son (Lamarca, 2006):

- **Apollo**: aplicación amigable de modelado de conocimiento el cual está basado en los principios básicos tales como clases, instancias, funciones, relaciones, etc. La interface de usuario tiene una arquitectura abierta y está escrito en lenguaje de programación JAVA.
- **LinkFactory**: se trata de una herramienta utilizada para construir complejos sistemas de terminología corporativa, capaz de extraer valor significativo de una gran cantidad de datos no estructurados.
- **OntoEdit Free and Professional versions**: esta herramienta permite crear, navegar y modificar Ontologías. Está basado en los estándares del W3C y ofrece muchas interfaces exportables a la mayor parte de lenguajes de representación de Ontologías.
- **OpenKnoME**: es la piedra angular de la aplicación utilizada por los motores de conocimiento *topThing*. Es una herramienta de gestión del conocimiento y un motor de Ontologías. Desde el 2001 el código fuente está abierto para la comunidad académica y la clínica sin ánimo de lucro.
- **WebODE**: herramienta para modelar el conocimiento utilizando Ontologías. Facilita la máxima flexibilidad e interoperabilidad con otras aplicaciones de negocios necesarias para las empresas de hoy en día.

1.4 Implantación de Ontologías en ambiente Web

Las Ontologías forman la base de la Web semántica, debido a que definen teorías sobre dominios compartidos y comunes, y permiten que las personas y los computadores se comuniquen de forma más eficaz. A continuación se presenta una introducción a la Web Semántica y las ventajas que ofrecen las Ontologías para la misma.

1.4.1 La Web Semántica

Según la W3C la Web Semántica es una Web extendida, dotada de mayor significado en la que cualquier usuario en Internet podrá encontrar respuestas a sus preguntas de forma más rápida y sencilla gracias a una información mejor definida. Al dotar a la Web de más significado y, por lo tanto, de más semántica, se pueden obtener soluciones a problemas habituales en la búsqueda de información gracias a la utilización de una infraestructura común, mediante la cual, es posible compartir, procesar y transferir información de forma sencilla. Esta Web extendida y basada en el significado, se apoya en lenguajes universales que resuelven los problemas ocasionados por una Web carente de semántica en la que, en ocasiones, el acceso a la información se convierte en una tarea difícil y frustrante.

Por lo tanto la idea de la Web Semántica se basa en añadir metadatos semánticos a la World Wide Web. Esas informaciones adicionales deben estar relacionadas de manera formal, para que sea posible evaluarlas automáticamente por máquinas. El objetivo es mejorar la World Wide Web para ampliar la interoperabilidad entre los sistemas informáticos y reducir la mediación de operadores humanos necesaria.

1.4.2 Funcionamiento de la Web Semántica

Una gran mayoría de los sitios Web, están contruidos o basados en HTML, el cual no es más que un lenguaje de marcado de etiquetas que permite visualizar contenido a través de un *browser*. Las etiquetas informan al *browser* cómo desplegar la información contenida en ellas. A pesar de que este lenguaje es un estándar, no es suficiente en cuanto a la información que permite almacenar con respecto a la página, así mismo es considerado un lenguaje estático, por lo cual surgen otros lenguajes para tratar de dar dinamismo a las páginas.

Estos lenguajes deben permitir modelar la información de forma sencilla y mucho más flexible. Es aquí cuando surge XML, un metalenguaje que permite definir tags relacionados a la información que se desea desplegar. A XML se une RDF, con el cual se puede incluir a una página mucha más información, que puede ser útil para realizar búsquedas, índices, categorizaciones, etc.

Pero, solo con estos lenguajes no basta, es necesario tener lenguajes semánticos y Ontologías para permitir un mejor desenvolvimiento de los entes inteligentes en el ambiente Web. Es así que surgen diversos lenguajes, entre ellos uno de los más conocidos, y del cual se hizo referencia anteriormente, denominado OWL.

En la Web Semántica se maneja un flujo de conocimiento, el cual el grupo de investigación de semanticweb.org representa de la siguiente manera:

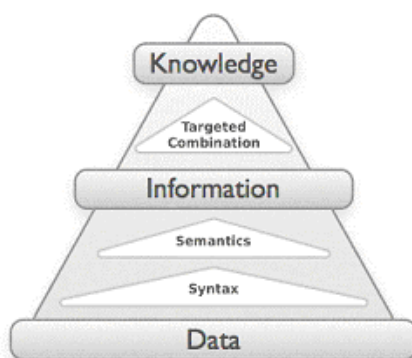


Figura 1: Flujo general del conocimiento. (www.semanticweb.org)

- **Datos:** son las señales individuales, en bruto y sin ningún significado que manipulan las computadoras para producir la información. En tal sentido, un dato puede ser un simple código ASCII.
- **Sintaxis:** la sintaxis significa que los datos se conformen según reglas o estructuras definidas (gramática). En la Web, la sintaxis puede estar definida por un documento DTD o un Schema.
- **Semántica:** el analizar datos es un requisito muy importante, pero no es suficiente para conseguir la información codificada. Por lo cual hace falta el significado real de los elementos en estructuras: “semántica supuesta”. Hay diversas maneras de expresar la semántica, por ejemplo en lengua natural, o en maneras más formales. La Web semántica se centra en el último que abre la puerta en el tratamiento de la información automatizada.
- **Información:** la información son datos que ha adquirido significado por formas de conexión relacional. Por ejemplo, de una base de datos relacional se puede inferir información de los datos almacenados debido a las diversas relaciones que en ella se establecen.
- **Conocimiento:** el conocimiento es una colección apropiada de información, y es un proceso determinístico. Cuando alguien almacena información adquiere conocimiento, que puede ser

útil para el ente que lo adquiere pero no produce una integración de tal manera que pueda generar conocimiento adicional. Por ejemplo, un niño que aprende a sumar, incorpora las entradas de los números en una tabla de suma, pero al preguntar algún registro que no se encuentra en ella no tiene la habilidad para responder. Esto solo se logra cuando se pasa al siguiente nivel. En el ámbito computacional, la mayoría de los sistemas con los que se trabaja para modelar, simular, etc., requieren de cierto tipo de conocimiento almacenado.

- **Entendimiento:** el entendimiento es un proceso cognitivo y analítico. Es el proceso por el cual se toma conocimiento y se transforma, con la ayuda del mismo, en nuevo conocimiento. La diferencia entre conocer y entender, viene dada por la diferencia entre aprender y memorizar. Los sistemas en IA poseen entendimiento, en el sentido de que ellos pueden procesar nuevo conocimiento basándose en el ya establecido y almacenado.
- **Sabiduría:** engloba todos los niveles anteriores del sentido, específicamente sobre aquellos que están relacionados con el ser humano (los códigos morales, éticos, etc.).

En resumen, según Bellinger y otros (2004):

Los **datos** representan un hecho o una declaración del acontecimiento sin relación a otras cosas.

e.g.: Está lloviendo.

La **información** incorpora la comprensión de una relación de cierta clase, posiblemente la causa y el efecto.

e.g.: La temperatura cayó 15 grados y entonces comenzó a llover.

El **conocimiento** representa un patrón que conecta y proporciona generalmente un alto nivel de previsibilidad en cuanto a lo que se describe o lo que sucederá después.

e.g.: Si la humedad está muy alta y la temperatura cae substancialmente es poco probable que la atmósfera pueda contener la lluvia.

La **sabiduría** incorpora más de una comprensión de los principios fundamentales que se encuentran dentro del conocimiento humano.

e.g.: Llueve porque llueve. Y esto abarca una comprensión de todas las interacciones que sucedan entre llover, la evaporación, corrientes de aire, gradientes de la temperatura, los cambios, y llover.

1.4.3 Arquitectura de la Web Semántica

Según Berners y otros (2001), la arquitectura de la Web Semántica podría estar representada tal como se muestra en la Figura 2.

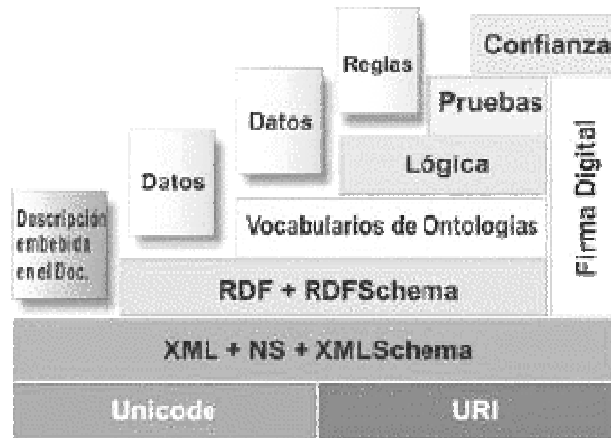


Figura 2: Arquitectura de la Web Semántica. (Berners-Lee, 2001).

- **Unicode:** es una norma de codificación de caracteres cuyo objetivo es asignar a cada posible carácter de cada posible lenguaje un número y nombre único.
- **URI (*Uniform Resource Identifier*):** texto corto que identifica unívocamente cualquier recurso (servicio, página, documento, enciclopedia, etc.), accesible en una red. Normalmente consta de dos partes: un identificador del método de acceso o protocolo al recurso, por ejemplo http:, mailto:, ftp:, y el nombre del recurso.
- **XML + NS + *xm*lschema:** es una capa técnica, la cual permite la comunicación entre los agentes, debido a que está enmarcada por conceptos que son estándares.
 - **XML:** aporta una sintaxis para los documentos estructurados, pero no les aporta ningún tipo de restricción sobre el significado.
 - **NS (*NameSpaces*):** se encargan de asociar elementos y atributos de nombres usados en XML con espacios de nombre identificados por referencias URI. Con esto se logra que los elementos cuya denominación es la misma puedan ser diferenciados por el NS al cual están asociados.
 - **XMLSchema:** es lenguaje el cual define la estructura del XML. Al tener un documento XML asociado a un XMLSchema podemos saber si el mismo es un XML válido y bien formado.

- **RDF + rdfschema:** es una capa que permite dotar de mayor significado al modelo, debido a que esta basado en un lenguaje que permite expresar y definir diferentes ideas en forma de Tripleta (sujeto, predicado, objeto).
 - **RDF:** es un modelo de datos para objetos ("recursos") y para las relaciones entre ellos, provee una semántica simple para este modelo de datos, a la vez que este modelo de datos puede ser representado en sintaxis XML. Es un modelo común (*Framework*) que permite hacer afirmaciones sobre los recursos (*Description*) y que hace posible que estos recursos pueden ser nombrados por URIs (*Resource*).
 - **RDF Schema:** es un vocabulario para describir propiedades y clases de recursos RDF, con una semántica para generalización de jerarquías de aquellas propiedades y clases.
- **Lenguaje de Ontologías:** son los lenguajes que permiten representar las Ontologías y posibilitan la utilización de las mismas por parte de entes computarizados. Esta capa permite agregar mucha más metainformación, por ejemplo, cardinalidades, transitividad, etc.
- **Lógica:** además de las Ontologías, se necesitan reglas de inferencia, que puedan ayudar al ser humano a sacar conclusiones desde ciertos valores dados. En este aspecto la máquina no entiende nada de lo que esta procesando, solo da respuestas asociadas a sus reglas y a la Ontología con la cual se esta trabajando. Generalmente, estas reglas pueden ser exportadas desde un sistema, pero no pueden ser importadas.
- **Pruebas:** Es necesario realizar pruebas para conocer si las reglas que se han definido son del todo correctas, para así ver el funcionamiento de la Web.
- **Confianza:** los entes encargados de recolectar información de la Web, deben ser muy escépticos en cuanto a lo encontrado en la Web, hasta que no confirmen las fuentes de la información recolectada.
- **Firma Digital:** es un conjunto de datos asociados a un mensaje digital que permite garantizar la identidad del firmante y la integridad del mensaje.

La interacción de todos los factores anteriores dentro de la Web semántica se observa en la Figura 3, en donde se tienen conceptos como Metadatos, Taxonomía, Web Services, Agentes Inteligentes, entre otros, los cuales están relacionados entre sí.

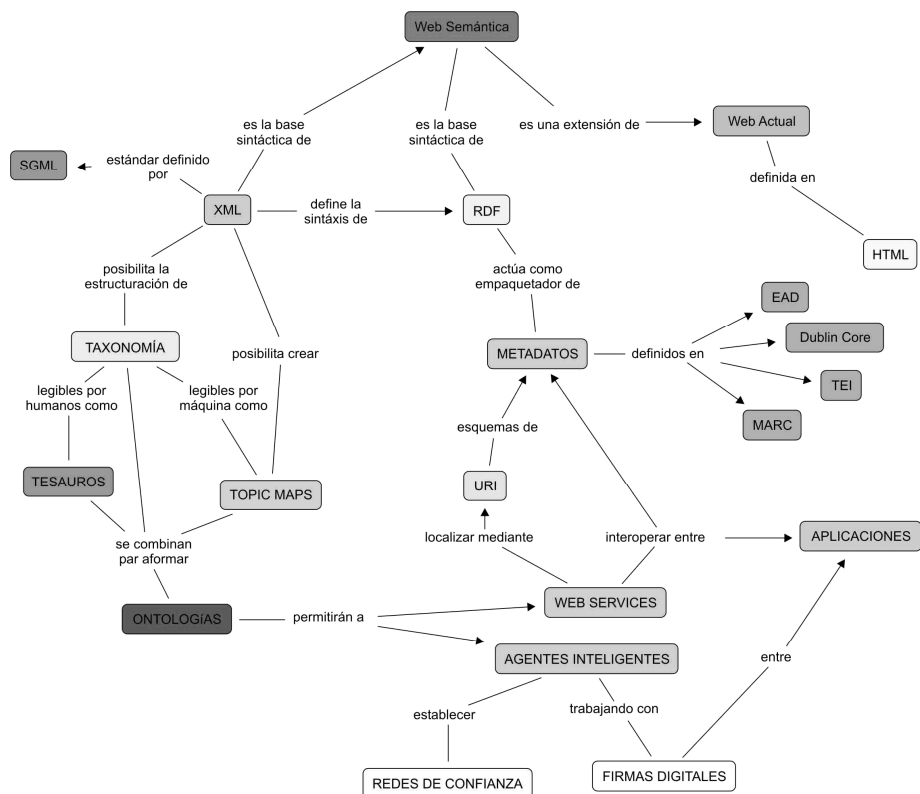


Figura 3: Mapa conceptual de la Web Semántica. (Rodríguez y Ronda, 2005).

1.4.4 Utilidad de las Ontologías para la Web

En la actualidad las Ontologías son usadas para satisfacer el agrupamiento, la búsqueda y la recolección de información en la Web, entre otras funciones. Estudios en este sentido han agrupado su utilidad en las siguientes sub-áreas (Heflin et. al, 2002):

- **Portales Web**

Un portal Web es un "sitio" o dirección Web que ofrece un conjunto de servicios sobre algún tema concreto o sobre diversos temas. Permite tener enlaces a temas de interés para las personas que se sienten personalizadas o familiarizadas con el tipo de información que es publicada en el mismo. Para que un portal sea exitoso debe tener sitios para localizar contenido interesante, los cuales la mayoría de las veces, son enviados por las personas adscritas al portal.

La información a mostrar es categorizada en tópicos y subtópicos, para lo cual a los documentos se le añaden simples tags o metadata para su clasificación. Muchos portales hacen únicamente un

índice de categorías para desplegar la información, sin embargo para realizar búsquedas de documentos en un portal grande esto no es suficiente.

En un portal Web se puede definir una Ontología para el tema que se trata en el sitio, la cual puede proveer terminología adecuada para describir contenido de los documentos, tener reglas y axiomas que permitan realizar las inferencias necesarias para dar resultados inteligentes ante búsquedas determinadas y/o personalizadas, lo cual sería difícil de realizar teniendo un sistema de búsqueda convencional.

- **Colecciones Multimedia**

Las Ontologías pueden ser usadas para dar sentido semántica a documentos multimedia (sonido, imágenes, videos, etc.), los cuales carecen de algún tipo de información o datos que pudieran ayudar en alguna búsqueda, debido a que están sumamente ligados al tratamiento humano y no al computacional. Sin embargo, aunque cada persona puede tener una forma distinta de representar la información ésta debe contener palabras claves que puedan ayudar a la búsqueda.

- **Administración de Sitios Web Corporativos**

Las grandes corporaciones poseen considerables cantidades de información concerniente a procesos de la empresa, páginas Web internas y externas, documentos de evaluación y educación del personal, etc., la cual puede ser organizada y estructurada de manera efectiva en una Ontología, con lo cual se puede lograr una organización taxonómica automatizada de datos y documentos y realizar fusiones entre la información de diferentes sectores corporativos.

- **Documentación de Diseño**

Al llevar a cabo un proyecto se genera una gran cantidad de documentación referente al proceso de desarrollo del mismo. Sin embargo, el formato de esta documentación puede ser distinto y depender de la etapa de desarrollo en la cual se esté. Una Ontología permite que los documentos puedan ser clasificados según las etapas y tener relaciones entre las mismas. Se pueden definir términos claves en cuanto a vocabulario común y enlaces externos a explicaciones detalladas de ciertos procesos.

- **Agentes Inteligentes**

En la Web Semántica los agentes tienen la habilidad de “entender” e integrar información proveniente de diversos recursos. Este proceso lo hacen usando Ontologías que pueden definir los términos asociados a su especialidad. Por ejemplo, un agente que recomienda libros a una persona, debe tener una Ontología de dominio que represente los términos para librerías, bibliotecas, etc., la cual le permitirá al agente capturar la información necesaria y discriminar entre las preferencias del usuario.

1.4.5 Visualización de Ontologías

A pesar de las ventajas que las Ontologías presentan a la hora de recuperar y dar información referente a un contexto, uno de los problemas de estos modelos es su presentación legible y comprensible al usuario final. En la fase de diseño y construcción de una Ontología se valora su poder expresivo y no se toman en cuenta aspectos visuales o estéticos del mismo.

En este sentido algunos trabajos se han realizado anteriormente y se presentan a continuación:

- **Aplicación de visualización de una Ontología para el dominio del análisis del semen humano**

Es un proyecto desarrollado por el Laboratorio de Inteligencia Artificial, de la Universidad Central de Venezuela, para la visualización de la Ontología referente al dominio del análisis del semen humano. En este proyecto se desarrolló una aplicación en ambiente Web, integrada por un módulo de administración y otro de acceso público. A través del primero se mantiene el sitio Web de la Ontología, mientras que el segundo permite a los usuarios acceder al conocimiento almacenado y a un conjunto de recursos, tales como imágenes, videos, artículos relativos al dominio, manuales y protocolos de laboratorio.

La arquitectura propuesta facilita la observación y recuperación de las complejas estructuras de conocimiento, así como la navegación y administración de la información representada en la Ontología. El enfoque utilizado en el diseño de los mecanismos de recuperación de información

está dirigido tanto a usuarios poco familiarizados con el vocabulario del dominio, como a aquellos que ya lo conocen (Pereira et. al, 2006).

- **RICOTERM-2**

Es un proyecto coordinado entre las Universidades de Santiago de Compostela y la Universitat Pompeu Fabra de Barcelona, en el que también participan investigadores de la Universidad del País Vasco y de la Universidad de Amberes. El objetivo principal de este proyecto de investigación, de carácter aplicado, es el diseño y el desarrollo del prototipo de un reelaborador de consultas para búsquedas en Internet (RECCI). Esta aplicación consiste en una interfaz, ubicada en un portal Web especializado en economía, que permitirá transformar una consulta simple monolingüe en un conjunto de consultas multilingües expandidas hacia otros conceptos relacionados con la consulta inicial. Las lenguas de trabajo son, además del inglés, el castellano, el catalán, el gallego y el vasco (Lorente, 2005). Se puede acceder a ella a través de la siguiente dirección: <http://genoma.iula.upf.edu:8080/genoma/index.jsp>.

Muchos han sido los trabajos orientados a búsquedas en la Web semántica, basándose en Ontologías para clasificar resultados, sin embargo, debido a que éstas generalmente no se utilizan para mostrar la información contenida a los usuarios, pocos son los visualizadores en ambiente Web que podemos encontrar.

Debido a la importancia que actualmente están tomando las Ontologías, se hace necesario contar con más herramientas para su visualización y edición, las cuales no sean aplicaciones de escritorio. Dichas herramientas no sólo deberían permitir obtener vistas útiles de la Ontología sobre la que se trabaja, sino que también deberían facilitar su manipulación y validación (Escarza et. al, 2005).

CAPÍTULO II

MARCO APLICATIVO

2.1 Planteamiento del Problema

Actualmente la información concerniente al dominio de los insectos acuáticos es muy escasa y de difícil acceso debido a que se encuentra dispersa y no publicada, existen pocos expertos en el área y los mismos a veces difieren en cuanto a la definición de conceptos. Es por esta razón que el Laboratorio de Inteligencia Artificial conjuntamente con el Laboratorio de Ecología de Sistemas Acuáticos del Instituto de Biología Experimental, ambos de la Facultad de Ciencias, de la Universidad Central de Venezuela, deciden desarrollar una Ontología en el dominio de los insectos acuáticos (Chaudary y Scarioni, 2005), usando Methontology como metodología para el proceso de levantamiento y recolección de la información y Protégé, como herramienta de formalización en el proceso de modelado del conocimiento.

Esta Ontología tiene como finalidad estandarizar y organizar el vocabulario referente a los insectos acuáticos, solventando el problema de tener información dispersa debido a que los conceptos del dominio y las relaciones entre los mismos se encuentran centralizados en la Ontología.

Como se mencionó anteriormente, la Ontología fue formalizada con la herramienta de modelado de conocimiento Protégé, la cuál permite visualizarla de forma detallada, sin embargo, la forma de representación de la misma puede ser difícil de entender por parte de las personas con poca experiencia, ya que inserta nodos que sirven como categorización de clases y muestra ciertas características que fuera del contexto computacional no aportan información y pueden constituir un elemento de perturbación.

La persona interesada en el contenido de la Ontología debería tener cierto conocimiento sobre el manejo de la herramienta Protégé, debido a que no solo muestra la Ontología en forma de árbol, sino también categorías, instancias, valores, y otras propiedades en pestañas diferentes, lo cual

resulta sumamente provechoso a la hora de editarla, mas no a la hora de ser visualizada por personas que poseen escasos conocimientos y/o habilidades para manejar la herramienta.

El poder visualizar la Ontología sólo a través de la herramienta Protégé lo convierte en una limitante, debido a que ésta debe estar instalada cuando exista la necesidad de visualizarla en cualquier máquina, lo cual puede ser desalentador para las personas que sólo deseen visualizar de manera inmediata la información o se encuentren trabajando en computadoras con cierto tipo de restricciones en cuanto a la instalación de nuevas aplicaciones.

Por lo planteado anteriormente, surge el interés de encontrar una manera de publicar el contenido de la Ontología para que sea visualizada de una forma sencilla por los expertos y personas interesadas en el área, la cual permita no solo visualizar su contenido, sino también ver imágenes asociadas a instancias y realizar búsquedas, la cuál no este limitada a una sola computadora sino que pueda ser accedida desde cualquier lugar geográfico.

2.2 Solución Propuesta

Teniendo como base el planteamiento anterior se propone crear una aplicación Web, que permita a las personas interesadas visualizar la Ontología de Insectos Acuáticos. Debido a que la aplicación a crear estará dirigida a la consulta a través de Internet se puede entonces aprovechar la ventaja de poder acceder a la información de manera rápida y sin ninguna limitación geográfica.

Es conveniente que la aplicación muestre la Ontología de tal forma que se haga natural para el usuario el poder navegar entre la información, asimismo que permita realizar varios tipos de búsqueda sobre el conocimiento y los recursos existentes en la aplicación, pudiendo el usuario filtrar sus resultados y orientándolos hacia lo que él necesita.

También resulta beneficioso el tener alguna forma de interactuar con el usuario de la aplicación, para poder obtener nuevos conocimientos, sugerencias, aportes y cualquier tipo de información que ayude a mantener actualizada la Ontología.

La capacidad de poder visualizar las imágenes de las especies a través de la aplicación es importante para los expertos, debido a que permite brindar una mejor comprensión acerca de las especies y sus características.

La aplicación deberá brindar la posibilidad de cargar y editar la Ontología, así como otros recursos, debido a que cada día se está en un continuo descubrimiento y aprendizaje, por lo cual se hace necesario el poder tener actualizada la información y así ofrecer un contenido completo en la aplicación, evitando que la Ontología quede desactualizada en un futuro.

2.3 Planteamiento del Problema

2.3.1 Objetivo General

Desarrollar un sitio Web para la visualización de la Ontología de Insectos Acuáticos.

2.3.2 Objetivos Específicos

- Identificar los requerimientos funcionales de la aplicación Web para la visualización de la Ontología.
- Diseñar la interfaz de la aplicación.
- Implementar un módulo de visualización para la información contenida en la Ontología.
- Implementar un módulo de actualización de instancias, el cual comprende las acciones de editar y agregar nuevas instancias, así como el asociar imágenes a las mismas.
- Implementar un módulo de búsqueda sobre la Ontología, así como todas las funcionalidades identificadas en procesos anteriores.
- Realizar pruebas para verificar el correcto funcionamiento de la aplicación.

2.4 Desarrollo de la Aplicación

Para el desarrollo de la aplicación de Visualización de la Ontología de Insectos Acuáticos se llevarán a cabo las siguientes actividades:

- Identificación y Análisis de los Requerimientos Funcionales: definición de las funcionalidades que debe satisfacer la aplicación.
- Establecimiento de los aspectos Técnicos: definición de las herramientas y tecnologías a usar durante el desarrollo de la aplicación.
- Definición de la Arquitectura: identificación de los módulos que conformarán la aplicación, así como los componentes de la misma y sus respectivas relaciones.
- Diseño del modelo de navegación entre páginas: establecimiento de reglas de navegación a través de un diagrama de despliegue.
- Definición de la Base de Datos: definición de tablas y campos requeridos para guardar los datos que necesitará la aplicación en diversos procesos.
- Diseño de la Interfaz de Usuario: creación de la interfaz a usar como pantalla de presentación de la información del sistema.
- Implementación de las funcionalidades: programación de los requerimientos definidos en actividades anteriores.
- Evaluación de la Aplicación: valoración de la aplicación por parte del experto, equipo de trabajo y personas que pudiesen ser usuarios de la aplicación.
- Optimización de la Aplicación: modificar la aplicación según resultados obtenidos en el proceso de evaluación.

A continuación, se presenta de forma detallada las actividades realizadas en el desarrollo de la aplicación, las cuales fueron definidas previamente en el documento.

2.4.1 Identificación y Análisis de los Requerimientos Funcionales

Los requerimientos identificados durante la fase de levantamiento de información fueron los siguientes:

- La aplicación de visualización estará formado por dos (2) módulos: un módulo de acceso público y un módulo de administración. El módulo de acceso público permitirá la visualización de la Ontología. El módulo de administración tiene como objetivo dar mantenimiento a la aplicación mediante la manipulación de los datos e información contenida en ella.

- La aplicación debe permitir visualizar la Ontología de Insectos Acuáticos de una manera sencilla según los expertos.
- El módulo de acceso público debe permitir al usuario:
 - Realizar búsqueda de términos no solo dentro de la ontología, sino también sobre todos los recursos que se encuentran en el sitio.
 - Ver y descargar documentos de interés presentes en la aplicación.
 - Acceder a sitios de interés a través de enlaces.
 - Consultar la sección de preguntas frecuentes.
 - Enviar consultas y/o sugerencias al administrador o experto.
 - Consultar la sección de ayuda del sistema.
- Se debe contar con un módulo de control de acceso para el administrador del sistema.
- El módulo administrativo debe permitir al administrador:
 - Editar la Ontología de Insectos Acuáticos para añadir imágenes a las instancias, así como realizar modificaciones al contenido de las mismas: agregar y actualizar.
 - Cargar nuevos archivos de extensión OWL los cuales contienen la ontología a ser desplegada en el sistema.
 - Cargar documentos así como sitios de interés relacionados con la aplicación, los cuales serán mostrados a los usuarios.
 - Consultar los mensajes enviados por parte de los usuarios a través de la aplicación.

Diagrama de Casos de Uso

Actores del sistema: al evaluar los requerimientos funcionales del sistema se pudieron identificar los siguientes actores:

- Administrador.
- Usuario.

A continuación se presentan los casos de uso identificados, los cuales para una mayor comprensión están divididos según el módulo al cual pertenezcan.

Módulo Administrativo

Nivel 0 - Módulo Administrativo

El diagrama que representa las funcionalidades del Módulo Administrativo se encuentran reflejadas en la Figura 4 que se presenta a continuación.

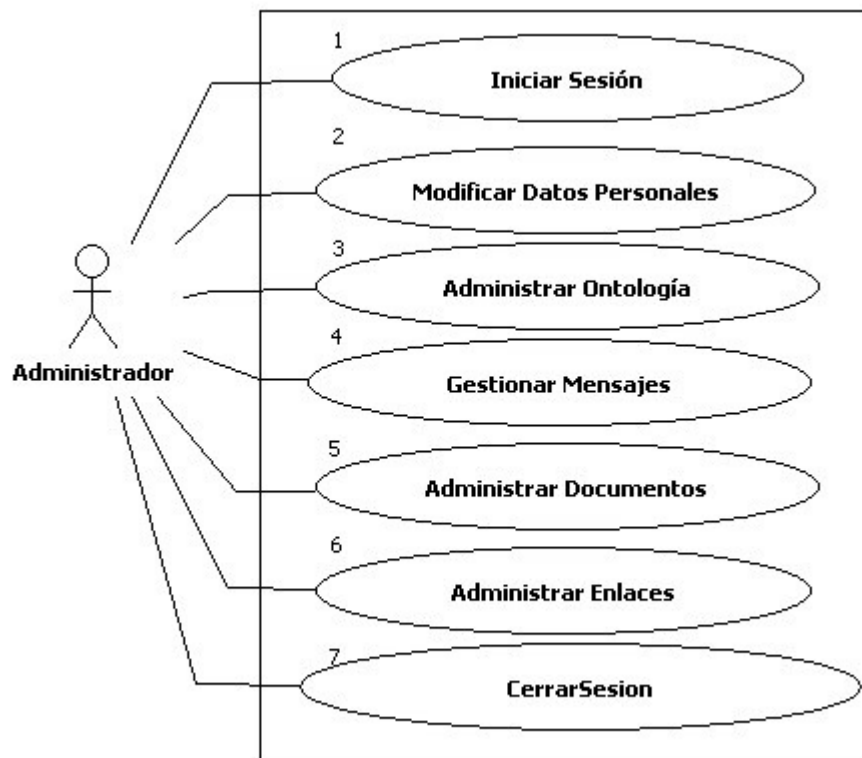


Figura 4: Diagrama de CU – Módulo de Administración.

Descripción de los Casos de Usos.

Nombre: Iniciar Sesión.

Identificador: CU 1.

Descripción: permite autenticar al usuario como administrador e iniciar una nueva sesión.

Pre-Condición: el administrador esta registrado en el sistema.

Post-Condición: el administrador ha sido reconocido por el sistema y se inicia una sesión.

Curso Básico de Acción:

1. El caso de uso comienza cuando el usuario selecciona la opción de iniciar sesión.

2. El usuario ingresa su login y password en los campos correspondientes y hace clic en el botón aceptar.
3. El sistema recibe estos datos y los procesa, realizando una validación contra la Base de Datos.
4. Si los datos son correctos se crea una nueva sesión para el administrador. [Curso Alt A].
5. Se redirecciona a la página de bienvenida del módulo de administración.
6. Finaliza el caso de uso.

Cursos Alternos:

- Curso Alt A: Los datos introducidos por el usuario no son correctos.
 - A.1 El sistema determina que el login introducido por el usuario no concuerda con la contraseña.
 - A.2 El sistema informa al usuario sobre este error vía una pantalla informativa.
 - A.3 El sistema le da la oportunidad al usuario de volver a introducir los datos.
 - A.4 El control de flujo regresa al punto 2 del curso básico de acción.

Nombre: Modificar Datos Personales.

Identificador: CU 2.

Descripción: modifica la información relacionada al administrador actual que está haciendo uso de la aplicación de visualización.

Pre-Condición: los datos del administrador han sido previamente cargados en la Base de Datos del sistema.

Post-Condición: los datos relacionados al administrador son cambiados en la Base de Datos del sistema, en la tabla Administrador.

Curso Básico de Acción:

1. El administrador selecciona la opción de modificar sus datos.
2. El sistema busca en la tabla Administrador el registro asociado al administrador que ha iniciado la sesión.
3. Los datos son desplegados en una pantalla sobre la cual el administrador puede realizar los cambios de valores que considere necesarios incluyendo la contraseña actual.
4. El administrador hace clic sobre el botón aceptar. [Curso Alt A].
5. El sistema efectúa la modificación de los datos en la tabla administrador de la B.D.

6. El sistema informa al administrador que la modificación fue realizada con éxito mediante un mensaje.
 7. Finaliza el caso de uso.
- Curso Alt A: No es válido al menos uno de los datos suministrados por el administrador.
 - A.1. El sistema determina que al menos uno de los datos suministrados por el administrador no cumple con las especificaciones del mismo.
 - A.2. El sistema informa al administrador que datos no pueden ser procesados mediante un mensaje de notificación, y le da la oportunidad de corregirlos.
 - A.3 El control de flujo regresa al punto 3 del curso básico de acción.

Nombre: Administrar Ontología.

Identificador: CU 3.

Descripción: permite al administrador realizar acciones de modificación y consulta sobre la Ontología actual, así como la carga de un nuevo archivo referente a la Ontología.

Nombre: Gestionar Mensajes.

Identificador: CU 4.

Descripción: permite al administrador realizar acciones de consulta y eliminación de mensajes emitidos por los usuarios del sistema.

Nombre: Administrar Documentos.

Identificador: CU 5.

Descripción: permite al administrador realizar diversas acciones sobre los documentos almacenados en el servidor, tales como consultar, agregar, modificar y eliminar documentos.

Nombre: Administrar Enlaces.

Identificador: CU 6.

Descripción: permite al administrador realizar diversas acciones sobre los enlaces de interés que son mostrados al público en general, tales como consultar, agregar, modificar y eliminar enlaces.

Nombre: Cerrar Sesión.

Identificador: CU 7.

Descripción: permite al administrador cerrar la sesión actual.

Pre-Condición: ninguna.

Post-Condición: el administrador ha cerrado la sesión y visualiza la pantalla de inicio del sistema.

Curso Básico de Acción:

1. El caso de uso comienza cuando el usuario selecciona la opción de cerrar sesión.
2. El sistema toma la petición y cancela la sesión actual.
3. El sistema redirecciona al usuario hacia la página de iniciar sesión.
4. Finaliza el caso de uso.

A continuación se muestra el primer nivel del diagrama de Casos de Uso correspondiente al módulo de administración.

Nivel I - Módulo Administrativo

La Figura 5 muestra el desglose del caso de uso referente a Administrar Ontología.

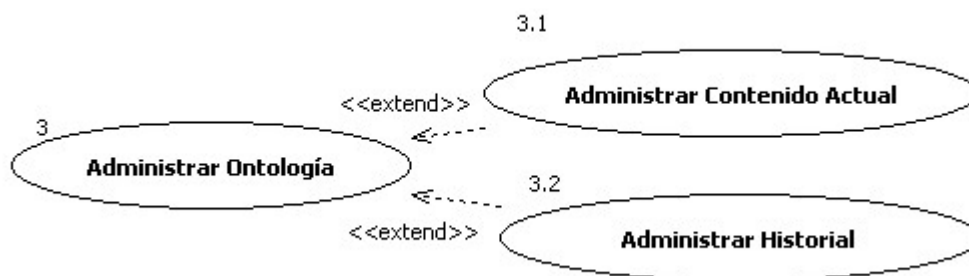


Figura 5: Diagrama de CU – Sub Módulo Administrar Ontología.

Nombre: Administrar Contenido Actual.

Identificador: CU 3.1.

Descripción: permite al administrador realizar diversas acciones sobre la Ontología actual, en especial sobre las instancias, tales como agregar y modificar instancias.

Nombre: Administrar Historial.

Identificador: CU 03.2.

Descripción: permite al administrador realizar diversas acciones sobre el historial de versiones de Ontologías, el cual se encuentra almacenado en el servidor, tales como consultar y eliminar versiones.

La Figura 6 muestra el desglose del caso de uso referente a la gestión de los mensajes enviados por los usuarios.

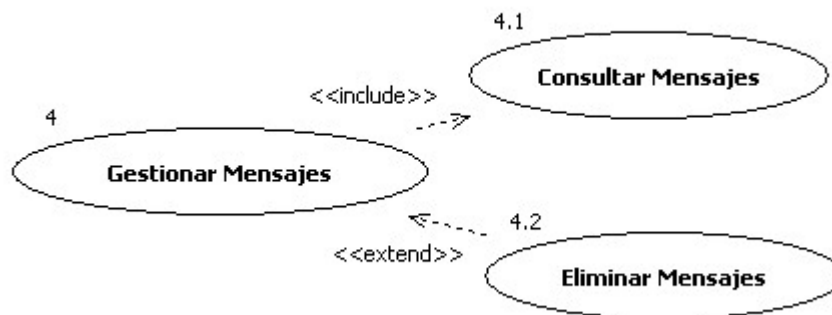


Figura 6: Diagrama de CU – Sub Módulo Gestionar Mensajes.

Nombre: Consultar Mensajes.

Identificador: CU 4.1

Descripción: muestra al administrador los mensajes que han sido enviados por los usuarios.

Pre-Condición: ninguna.

Post-Condición: el administrador visualiza los mensajes almacenados en el servidor.

Curso Básico de Acción:

1. El administrador selecciona la opción de consultar mensajes.
2. Se muestra al administrador una interfaz de búsqueda de mensajes.
3. El administrador selecciona la opción de búsqueda e introduce los parámetros de la misma.
4. Hace clic en el botón aceptar.
5. El sistema busca los mensajes que cumplan con las características dadas.
6. Se muestran estos mensajes al administrador, ordenados por fecha de emisión.
7. Finaliza el caso de uso.

Nombre: Eliminar Mensajes.

Identificador: CU 4.2

Descripción: eliminar un mensaje del sistema.

Pre-Condición: debe existir al menos un mensaje en el buzón de mensajes recibidos.

Post-Condición: el registro del mensaje ha sido eliminado de la tabla Mensajes de la Base de Datos.

Curso Básico de Acción:

1. El administrador selecciona la opción de consultar mensajes.
2. El administrador selecciona el mensaje que desea eliminar.
3. Presiona el botón eliminar.
4. El sistema elimina el registro relacionado a este mensaje de la tabla Mensajes.
5. Finaliza el caso de uso.

La Figura 7 muestra el desglose del caso de uso referente a Administrar Documentos.

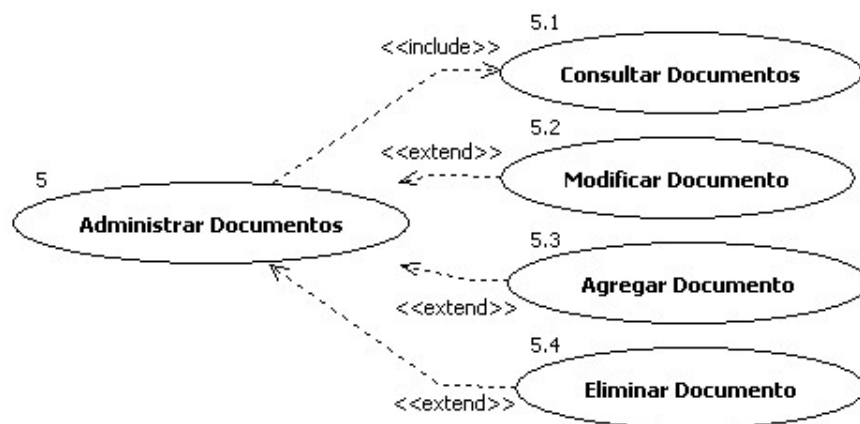


Figura 7: Diagrama de CU – Sub Módulo Administrar Documentos.

Nombre: Consultar Documentos.

Identificador: CU 5.1

Descripción: muestra la lista de documentos que se encuentran almacenados en el servidor y que van a ser mostrados al usuario como documentos de interés.

Pre-Condición: ninguna.

Post-Condición: se mostraron todos los documentos almacenados con sus respectivos atributos.

Curso Básico de Acción:

1. El administrador selecciona la opción consultar documentos.
2. Ingresa algún término de búsqueda.
3. Hace clic sobre el botón aceptar.
4. El sistema realiza una búsqueda sobre la lista de documentos almacenados.
5. Se muestra el resultado de la búsqueda.
6. Finaliza el caso de uso.

Nombre: Modificar Documento.

Identificador: CU 5.2

Descripción: modificar la información correspondiente a un documento específico.

Pre-Condición: debe existir al menos un documento en el servidor.

Post-Condición: la información del documento es actualizada en la Base de Datos del sistema, en la tabla Documentos.

Curso Básico de Acción:

1. El administrador selecciona la opción consultar documentos.
2. Elige el documento que desea modificar.
3. Se despliega una pantalla para editar el nombre y la descripción del documento.
4. El administrador introduce los nuevos datos y presiona el botón aceptar.
5. El sistema actualiza el registro correspondiente de la tabla Documentos en la BD.
6. Se redirecciona a la pantalla de consulta de documentos.
7. Finaliza el caso de uso.

Nombre: Agregar Documento.

Identificador: CU 5.3

Descripción: carga un nuevo documento a la aplicación.

Pre-Condición: ninguna.

Post-Condición: el nuevo documento ha sido cargado en el servidor y se actualiza la Base de Datos con un nuevo registro en la tabla Documentos.

Curso Básico de Acción:

1. El administrador selecciona la opción de cargar documento.

2. Se despliega una interfaz para realizar la carga del archivo y los datos asociados al documento.
 3. En el campo cargar se introduce la ruta local del archivo.
 4. El administrador hace clic en el botón aceptar.
 5. El sistema toma el archivo y lo almacena en una ruta específica en el servidor denominada recursos. [Curso Alt A].
 6. En la tabla Documentos de la Base de Datos se guardan los datos asociados a este nuevo documento.
 7. Se redirecciona a la pantalla de consulta de documentos.
 8. Finaliza el caso de uso.
- Curso Alt A: Ocurre un error en la carga del documento.
 - A.1. El sistema determina que ocurrió un error durante la carga del documento al servidor.
 - A.2. El sistema informa al administrador que el documento no ha podido ser cargado mediante un mensaje de error: carga fallida.
 - A.3 El control de flujo regresa al punto 2 del curso básico de acción.

Nombre: Eliminar Documento.

Identificador: CU 5.4

Descripción: elimina un documento del sistema.

Pre-Condición: debe existir al menos un documento en el servidor.

Post-Condición: el documento ya no existe en el servidor ni su registro en la tabla Documentos de la Base de Datos.

Curso Básico de Acción:

1. El administrador selecciona la opción consultar documentos.
2. El administrador selecciona el documento que desea eliminar.
3. Hace clic sobre el botón eliminar.
4. El sistema borra el archivo del servidor.
5. Se elimina el registro relacionado a este documento de la Base de Datos del sistema
6. Finaliza el caso de uso.

La Figura 8 muestra el desglose del caso de uso referente a Administrar Enlaces.

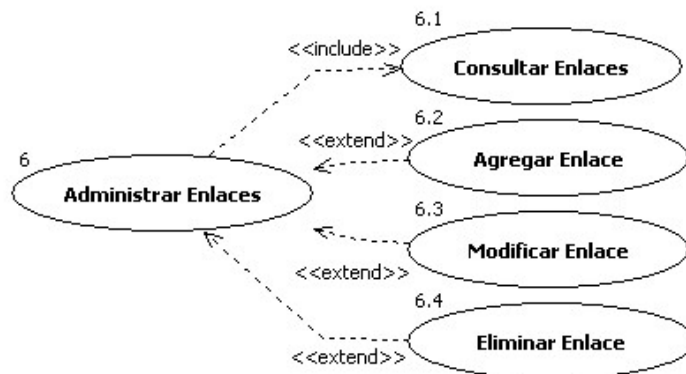


Figura 8: Diagrama de CU – Sub Módulo Administrar Enlaces.

Nombre: Consultar Enlaces.

Identificador: CU 6.1

Descripción: muestra la lista de enlaces que se encuentran almacenados en la Base de Datos y que van a ser mostrados al usuario como enlaces de interés.

Pre-Condición: ninguna.

Post-Condición: se mostraron todos los enlaces con sus respectivos atributos.

Curso Básico de Acción:

1. El administrador selecciona la opción consultar enlaces.
2. Se ingresa algún término de búsqueda.
3. El administrador hace clic sobre el botón aceptar.
4. El sistema realiza una búsqueda sobre la lista de enlaces almacenados en la tabla Enlaces, de la Base de Datos del sistema.
5. Se muestra en la interfaz una tabla con el resultado de la búsqueda.
6. Finaliza el caso de uso.

Nombre: Modificar Enlace.

Identificador: CU 6.2

Descripción: modifica los datos correspondientes a un enlace específico.

Pre-Condición: debe existir al menos un enlace almacenado en la tabla Enlaces.

Post-Condición: los datos asociados al enlace son actualizados en la Base de Datos del sistema.

Curso Básico de Acción:

1. El administrador selecciona la opción consultar enlaces
2. El administrador elige el enlace que desea modificar.
3. El sistema carga los datos asociados a ese enlace.
4. Se despliega una pantalla para editar los datos.
5. El administrador introduce los nuevos datos y hace clic sobre el botón aceptar. [Curso Alt A].
6. El sistema actualiza los datos en el registro correspondiente de la tabla enlaces de la Base de Datos.
7. Se redirecciona a la pantalla de consulta de enlaces.
8. Finaliza el caso de uso.

Cursos Alternos:

- Curso Alt A: No es válido al menos uno de los datos suministrados por el administrador.
 - A.1. El sistema determina que al menos uno de los datos suministrados por el administrador no cumple con las especificaciones del mismo.
 - A.2. El sistema informa al administrador que datos no pueden ser procesados mediante una pantalla de notificación, y le da la oportunidad de corregirlos.
 - A.3 El control de flujo regresa al punto 5 del curso básico de acción.

Nombre: Agregar Enlace.

Identificador: CU 6.3

Descripción: agrega un enlace de interés al sistema.

Pre-Condición: ninguno.

Post-Condición: el nuevo enlace ha sido guardado en la tabla Enlaces de la Base de Datos.

Curso Básico de Acción:

1. El administrador selecciona la opción de agregar enlace.
2. Se despliega una interfaz para realizar la carga de los datos del enlace a agregar.
3. El administrador introduce los datos asociados al nuevo registro.
4. El administrador hace clic en el botón aceptar.
5. El sistema guarda los datos asociados a este nuevo enlace en la tabla Enlaces de la Base de Datos del sistema.

6. Se redirecciona a la pantalla de consulta de enlaces.
7. Finaliza el caso de uso.

Nombre: Eliminar Enlace.

Identificador: CU 6.4

Descripción: elimina un enlace del sistema.

Pre-Condición: debe existir al menos un enlace.

Post-Condición: el registro del enlace ha sido eliminado de la tabla enlaces de la Base de Datos.

Curso Básico de Acción:

1. El administrador selecciona la opción consultar enlaces.
2. El administrador selecciona el enlace que desea eliminar.
3. Presiona el botón eliminar.
4. El sistema elimina el registro relacionado a este enlace de la tabla Enlaces.
5. Finaliza el caso de uso.

A continuación se muestra el segundo nivel del diagrama de Casos de Uso correspondiente al módulo de administración.

Nivel II - Módulo Administrativo

La Figura 9 muestra el desglose del caso de uso referente a la administración del contenido de la ontología actual.

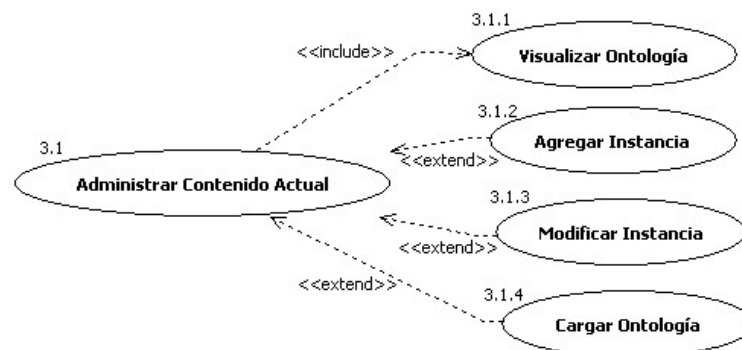


Figura 9: Diagrama de CU – Sub Módulo Administrar Contenido Actual.

Nombre: Visualizar Ontología.

Identificador: CU 3.1.1

Descripción: permite consultar la información contenida en la Ontología actual.

Pre-Condición: existe una Ontología disponible.

Post-Condición: el administrador ha visualizado la información contenida en la Ontología actual y puede navegar a través de ella.

Curso Básico de Acción:

1. El caso de uso comienza cuando el administrador selecciona la opción de Administrar Ontología.
2. El sistema busca el archivo que contiene la Ontología, la procesa a diversos objetos para que pueda ser mostrada.
3. Se muestra la información a través de una interfaz.
4. Finaliza el caso de uso.

Nombre: Agregar Instancia.

Identificador: CU 3.1.2

Descripción: agrega una nueva instancia a la Ontología con la que se está trabajando.

Pre-Condición: existe una Ontología actual.

Post-Condición: se crea una nueva instancia en el modelo y se almacena en el archivo OWL.

Curso Básico de Acción:

1. Se despliega una página en la cual se visualizan las clases de la Ontología y sus instancias.
2. El administrador selecciona la opción de agregar una instancia.
3. Se despliega un formulario con los atributos correspondientes a la clase asociada o clase padre de la instancia.
4. El administrador inserta los valores de los campos que desee.
5. Hace clic en el botón aceptar.
6. El sistema crea una nueva instancia, asocia los valores tomados desde la interfaz y la agrega al archivo OWL.
7. Se redirecciona a la página inicial de Administración de Ontología.
8. Finaliza el caso de uso.

Nombre: Modificar Instancia.

Identificador: CU 3.1.3

Descripción: cambia los valores de los atributos de una instancia específica.

Pre-Condición: existe una Ontología actual con la cual se está trabajando.

Post-Condición: los datos de la instancia son cambiados y guardados.

Curso Básico de Acción:

1. Se despliega una página en la cual se visualizan las clases de la Ontología y sus instancias.
2. El administrador selecciona una instancia para editar.
3. Se despliega un formulario con los atributos correspondientes a la clase asociada o clase padre de la instancia.
4. El administrador escribe o sobrescribe los valores de los campos que desea cambiar.
5. Hace clic en el botón aceptar.
6. El sistema actualiza los valores de los atributos en los archivos correspondientes.
7. Se redirecciona a la página inicial de Administración de Ontología.
8. Finaliza el caso de uso.

Nombre: Cargar Ontología.

Identificador: CU 3.1.4

Descripción: se carga una Ontología desde la máquina local al servidor, la cual será considerada como actual y podrá ser visualizada por los usuarios.

Pre-Condición: ninguna.

Post-Condición: la nueva Ontología ha sido almacenada en el servidor.

Curso Básico de Acción:

1. El administrador selecciona la opción de cargar Ontología.
2. Se despliega una interfaz para realizar la carga.
3. En el campo cargar se introduce la ruta local del archivo.
4. Se introduce el número de la versión que se desea cargar.
5. El administrador hace clic en el botón aceptar.
6. El sistema toma el archivo y lo almacena en una ruta específica en el servidor y la guarda con el nuevo número de versión. [Curso Alt A].

7. De existir una Ontología almacenada como actual al momento de realizar la carga de la nueva versión, ésta pasa a ser parte del historial de versiones.
 8. En la tabla Documentos de la Base de Datos se guardan los datos asociados a este nuevo archivo.
 9. Se redirecciona a la página inicial de Administración de Ontología.
 10. Finaliza el caso de uso.
- Curso Alt A: Ocurre un error en la carga de la Ontología.
 - A.1. El sistema determina que ocurrió un error durante la carga de la Ontología al servidor.
 - A.2. El sistema informa al administrador que la Ontología no ha podido ser cargado mediante un mensaje de error: carga fallida.
 - A.3 El control de flujo regresa al punto 2 del curso básico de acción.

Antes de pasar al siguiente caso de uso, es importante resaltar una característica de la aplicación a desarrollar, la cual se basa en la independencia de la carga del archivo OWL del contexto de la aplicación, es decir, todo archivo OWL desarrollado en Protégé y que cumpla con dos (2) reglas básicas en la Ontología, puede ser desplegado por la aplicación si necesidad de hacer referencia al contexto de los insectos acuáticos.

Las dos reglas que hay que tener en cuenta son las siguientes:

- El nombre de la relación entre una clase y su(s) clase(s) padre(s) debe estar representado con un comentario en el archivo OWL de la Ontología con el prefijo: Hijo de.
- El nombre de la relación entre una clase y su(s) clase(s) hija(s) debe estar representado con un comentario en el archivo OWL de la Ontología con el prefijo: Padre de.

De no tener alguno de estos comentarios, la aplicación no mostrara el nombre de la relación omitida.

A continuación se muestra el desglose del caso de uso referente a la administración del historial de ontologías.

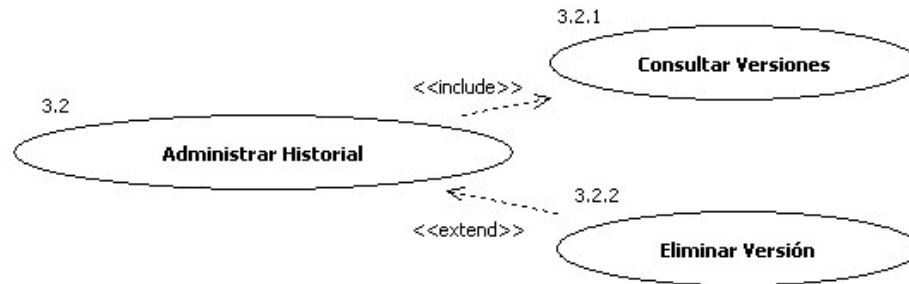


Figura 10: Diagrama de CU – Sub Módulo Administrar Historial de Ontologías.

Nombre: Consultar Versiones.

Identificador: CU 3.2.1

Descripción: permite al administrador consultar cuales versiones de la Ontología están almacenadas en el servidor y los datos asociadas a las mismas.

Pre-Condición: ninguna.

Post-Condición: el administrador visualiza las Ontologías que se encuentran almacenadas en el historial.

Curso Básico de Acción:

1. El administrador selecciona la opción de Cargar Ontología.
2. El sistema busca el nombre de las versiones existentes en la Base de Datos, así como los datos relacionados a ellas.
3. El sistema despliega en la pantalla la lista de Ontologías anteriores
4. Finaliza el caso de uso.

Nombre: Eliminar Versión.

Identificador: CU 3.2.2

Descripción: permite al administrador eliminar una versión de la Ontología en el servidor.

Pre-Condición: existe al menos una Ontología en el Historial de Ontologías.

Post-Condición: la versión es eliminada del servidor y por consiguiente no aparecerá de nuevo en la lista de versiones al momento de consultarlas.

Curso Básico de Acción:

1. El administrador consulta la sección de administración de documentos.
2. Selecciona la versión que desea eliminar.

3. Presiona el botón eliminar.
4. El sistema borra el archivo de la versión en el servidor.
5. Se emite un mensaje de éxito al administrador.
6. Finaliza el caso de uso.

A continuación se muestra el tercer nivel del diagrama de Casos de Uso correspondiente al módulo de administración.

Nivel III - Módulo Administrativo

En este nivel se detalla el caso de uso que hace referencia a la modificación de instancias en el módulo de administración y que se presenta en la Figura 11.

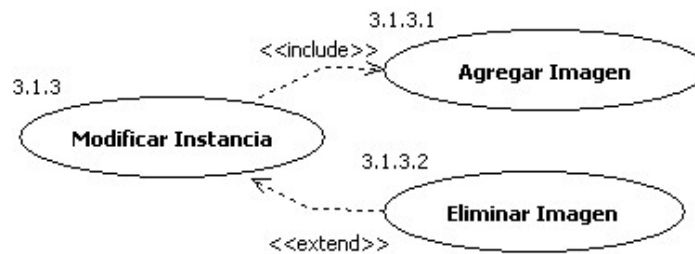


Figura 11: Diagrama de CU – Sub Módulo Administrar Modificar Instancia.

Nombre: Agregar Imagen.

Identificador: CU 3.1.3.1

Descripción: carga imágenes en el servidor las cuales son asociadas a una instancia específica de la Ontología.

Pre-Condición: existe una Ontología actual con la cual se está trabajando.

Post-Condición: la imagen es guardada en el servidor y relacionada a una instancia a través de la tabla Imágenes de la Base de Datos del sistema.

Curso Básico de Acción:

1. Se despliega una página en la cual se visualiza la Ontología y sus propiedades.
2. El administrador selecciona una instancia para editar.
3. Se selecciona la opción de edita imágenes.
4. Se despliega una interfaz de carga de imágenes.

5. En el campo cargar se introduce la ruta local de la imagen a cargar.
 6. El administrador hace clic en el botón aceptar.
 7. El sistema toma el archivo y lo almacena en el servidor en una carpeta imágenes. [Curso Alt A].
 8. El sistema relaciona el nombre de la imagen con la instancia en la tabla imágenes de la Base de Datos.
 9. Se emite un mensaje de carga exitosa al administrador.
 10. Finaliza el caso de uso.
- Curso Alt A: Ocurre un error durante la carga de la imagen.
 - A.1. El sistema determina que ocurrió un error durante la carga de la imagen al servidor.
 - A.2. El sistema informa al administrador que la imagen no ha podido ser cargado mediante un mensaje de error: carga fallida.
 - A.3 El control de flujo regresa al punto 4 del curso básico de acción.

Nombre: Eliminar Imagen.

Identificador: CU 3.1.2.2.

Descripción: permite al administrador eliminar una imagen asociada a una instancia específica la Ontología.

Pre-Condición: existe una Ontología actual con la cual se está trabajando.

Post-Condición: la imagen es borrada del servidor, así como su registro en la tabla Imágenes de la Base de Datos.

Curso Básico de Acción:

1. Se despliega una página en la cual se visualiza la Ontología y sus propiedades.
2. El administrador selecciona una instancia para editar.
3. Se selecciona la opción de edita imágenes.
4. El administrador selecciona la imagen a eliminar.
5. El administrador presiona el botón eliminar.
6. El sistema borra la imagen en el servidor y su registro en la tabla Imágenes.
7. Se emite un mensaje de éxito al administrador.
8. Finaliza el caso de uso.

Módulo de Acceso Público

Nivel 0 - Módulo de Acceso Público:

El diagrama que representa las funcionalidades del Módulo de Acceso Público se encuentran reflejadas en la Figura 12 que se presenta a continuación.

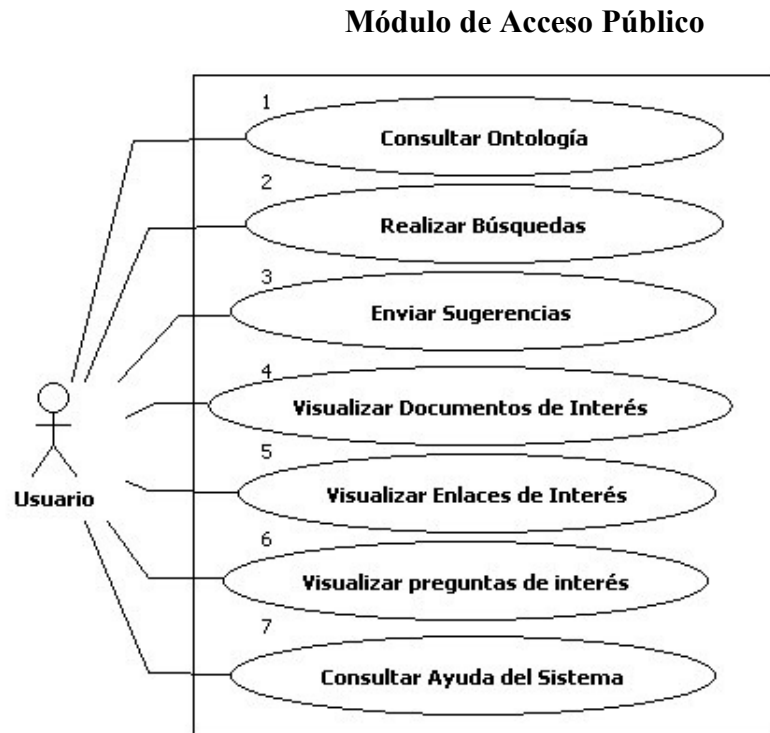


Figura 12: Diagrama de CU –Módulo de Acceso Público.

Descripción de los Casos de Usos.

Nombre: Consultar Ontología.

Identificador: CU 1.

Descripción: permite consultar la información contenida en la Ontología actual.

Pre-Condición: ninguna.

Post-Condición: el usuario ha visualizado la información contenida en la Ontología actual y puede navegar a través de ella.

Curso Básico de Acción:

1. El caso de uso comienza cuando el usuario selecciona la opción de Visualizar Ontología.
2. El sistema busca el archivo que contiene la Ontología, la procesa a diversos objetos para que pueda ser mostrada.

3. Se muestra la información a través de una interfaz.
4. Finaliza el caso de uso.

Nombre: Realizar Búsquedas.

Identificador: CU 2.

Descripción: permite consultar la información contenida en la aplicación.

Pre-Condición: ninguna.

Post-Condición: el usuario visualiza los recursos y/o términos relacionados a los parámetros de su búsqueda.

Curso Básico de Acción:

1. El caso de uso comienza cuando el usuario selecciona la opción Buscar.
2. El usuario introduce su parámetro de búsqueda más el contexto de la misma, en la Ontología ó el sitio.
3. El sistema toma estos parámetros y realiza la búsqueda sobre el contexto seleccionado usando algoritmos de búsquedas para hacerla mucho más eficiente.
4. Se arma una lista con los resultados obtenidos.
5. Se despliega la lista resultante en una pantalla para que el usuario pueda visualizarla.
6. Finaliza el caso de uso.

Nombre: Enviar Sugerencias.

Identificador: CU 3.

Descripción: permite al usuario enviar un mensaje relacionado a una pregunta y/o sugerencia al administrador del sistema.

Pre-Condición: ninguna.

Post-Condición: el mensaje se ha almacenado como un nuevo registro en la Base de Datos del sistema.

Curso Básico de Acción:

1. El caso de uso comienza cuando el usuario selecciona la opción de Buzón de Sugerencias, en el menú de contactos.
2. El usuario introduce los datos necesarios para enviar el mensaje, tales como su nombre, apellido, correo electrónico y texto del mensaje.

3. El usuario hace clic sobre el botón enviar [Curso Alt A].
4. El sistema toma todos los datos y los introduce en la Base de Datos del sistema como un nuevo registro, añadiendo la hora y fecha de emisión del mensaje.
5. El sistema informa al usuario sobre el éxito del envío.
6. Finaliza el caso de uso.

Cursos Alternos:

- Curso Alt A: No es válido al menos uno de los datos suministrados para enviar el mensaje.
 - A.1. El sistema determina que al menos uno de los datos suministrados por el usuario no cumple con las especificaciones establecidas.
 - A.2. El sistema informa al usuario que datos no pueden ser procesados mediante una pantalla de notificación, y le da la oportunidad de corregirlos.
 - A.3 El control de flujo regresa al punto 2 del curso básico de acción.

Nombre: Visualizar Documentos de Interés.

Identificador: CU 4.

Descripción: permite al usuario visualizar los documentos almacenados en el servidor.

Pre-Condición: ninguna.

Post-Condición: el usuario visualiza la lista de documentos almacenados con su descripción.

Curso Básico de Acción:

1. El caso de uso comienza cuando el usuario selecciona la opción Documentos de Interés.
2. El sistema carga de la tabla Documentos de la Base de Datos el nombre y descripción de los documentos almacenados.
3. Se despliega la interfaz correspondiente para visualizar los documentos.
4. Finaliza el caso de uso.

Nombre: Visualizar Enlaces de Interés.

Identificador: CU 5.

Descripción: permite al usuario visualizar los enlaces almacenados en la tabla Enlaces de la Base de Datos del sistema.

Pre-Condición: ninguna.

Post-Condición: el usuario visualiza la lista de enlaces de interés.

Curso Básico de Acción:

1. El caso de uso comienza cuando el usuario selecciona la opción Enlaces de Interés.
2. El sistema carga de la tabla Enlaces de la Base de Datos el nombre de los enlaces, así como la información relacionada a ellos.
3. Se despliega la interfaz correspondiente para visualizar los enlaces.
4. Finaliza el caso de uso.

Nombre: Visualizar Preguntas de Interés.

Identificador: CU 6.

Descripción: permite al usuario visualizar las preguntas de interés que se encuentran publicadas en la aplicación. Es importante resaltar que estas preguntas fueron suministradas por la experta en el dominio de la aplicación y hacen referencia a las preguntas que se realizan de manera frecuente, al momento de trabajar o estudiar a los insectos acuáticos.

Pre-Condición: ninguna.

Post-Condición: el usuario visualiza las preguntas y su correspondiente respuesta.

Curso Básico de Acción:

1. El caso de uso comienza cuando el usuario selecciona la opción de Visualizar Preguntas de Interés.
2. El sistema carga de la Base de Datos las diversas preguntas que se encuentran en el sistema, así como su respuesta.
3. Se despliega la interfaz correspondiente en donde se publican estas preguntas.
4. Al hacer clic sobre alguna pregunta, se despliega una ventana emergente con la pregunta seleccionada y su correspondiente respuesta.
5. Finaliza el caso de uso.

Nombre: Consultar Ayuda del Sistema.

Identificador: CU 7.

Descripción: permite al usuario visualizar la ayuda del sistema.

Pre-Condición: ninguna.

Post-Condición: el usuario visualiza la ayuda.

Curso Básico de Acción:

1. El caso de uso comienza cuando el usuario selecciona la opción Ayuda.
2. Se despliega la ventana de ayuda.
3. Finaliza el caso de uso.

2.4.2 Establecimiento de los aspectos Técnicos

La especificación de la plataforma de desarrollo de la aplicación de Visualización de La Ontología de Insectos Acuáticos se presenta a continuación.

- **Especificaciones de Hardware:** el hardware que fue utilizado para la implementación de la aplicación es el siguiente:
 - Procesador Intel Pentium IV.
 - 1 GB de Memoria RAM.
 - Disco Duro de 40GB.
- **Sistema Operativo:** la aplicación fue desarrollada bajo el sistema operativo Windows XP.
- **Servidor Apache Tomcat:** Tomcat es un contenedor Web, el cual permite compilar y desplegar archivos Java Server Pages (JSP) y administrar el ciclo de vida de los servlets para el control de la aplicación. En el desarrollo de la Aplicación de Visualización se utilizó Apache Tomcat versión 5.5.9. Para la instalación y configuración de Tomcat ver Anexo C.
- **Herramientas de Software:** luego de evaluar diversas herramientas para el desarrollo de la aplicación se seleccionaron las siguientes herramientas:
 - **Kit de Desarrollo Java (JDK):** incluye la Máquina Virtual de Java (*Java Runtime Environment*) versión 1.5.0_06, que es el entorno de ejecución de Java. El JDK usado es el jsdk j2sdk1.4.2_11. Para la instalación del JDK ver Anexo C.
 - **Editor de Ontologías Protégé versión 3.2.1:** es una herramienta para el modelado de conocimiento en forma de ontología.
 - **Entorno de desarrollo Eclipse.**

- **Editor de archivos Notepad++:** editor de gran ayuda a la hora de trabajar con cualquier archivo de texto, en especial archivos XML.
- **StarUML:** editor para la realización de diagramas.
- **Macromedia Dreamweaver MX 2004:** herramienta usada para el diseño y programación de las páginas Web, debido a que facilita la generación de código.
- **Macromedia Fireworks MX 2004:** programa usado para editar y manipular las imágenes necesarias para la interfaz de la aplicación.
- **Servidor de Base de Datos MySql 4.1.14.**
- **Navegador Web:** debido a la que la aplicación desarrollada se encuentra en un ambiente Web, es necesario tener un navegador (*browser*) para poder visualizar las páginas html y jsp a desarrollar. Los navegadores usados durante la fase de desarrollo son Mozilla Firefox 5.0 e Internet Explorer 6.
- **JavaScript:** se utilizó en la aplicación para realizar validaciones de formularios y para dar dinamismo a la interfaz.
- **Java Server Pages (JSP) y Servlets:** debido a que se escogió como entorno de desarrollo el ambiente que proporciona Java, se usaron páginas JSP, ya que son páginas Web dinámicas cuyo procesamiento de código se realiza en el servidor. Por otro lado, los Servlet son usados para implementar la lógica que deben tener cada página.
- **eXtensible Markup Language (XML):** los archivos XML en la aplicación son usados para la configuración del sistema.
- **Paquete Jakarta Commons Fileupload:** paquete muy útil en procesos que involucren carga de archivo al servidor. Paquete libre Jakarta Commons Fileupload desarrollado por Apache Software Foundation.
- **Mail:** paquetes para el envío de mensajes a correos electrónicos.

2.4.3 Establecimiento de tecnologías a usar

La especificación de las tecnologías de desarrollo de la aplicación de Visualización de La Ontología de Insectos Acuáticos se presenta a continuación.

- **Jena**

Jena es un framework de código abierto desarrollado en Java, el cual provee un Api para extraer y guardar datos en grafos RDF, los cuales son representados como un modelo abstracto. El modelo puede tener como fuente datos archivos, bases de datos, urls o una combinación de éstos. También se pueden hacer preguntas sobre el modelo a través con SPARQL.

A través de la librería que ofrece Jena se pueden manipular ontologías implementadas en OWL, es decir, tomar y cambiar valores tanto de clases como de instancias almacenadas en la Ontología. Para esto, el API proporciona diversos métodos a usar, todos ellos basados en la construcción de un modelo lógico, sobre el cual se realizan las búsquedas de los elementos correspondientes.

En el API de Jena se encuentran las siguientes clases fundamentales:

- **Model:** es el modelo lógico que representa la Ontología.
- **OntClass:** representan las clases que conforman la Ontología, éstas tienen nombres, comentarios y atributos propios.
- **Instance,** son las instancias asociadas a una clase en particular, contienen valores relacionados a los atributos establecidos en la clase de la cual derivan.
- **Property:** son los atributos que contiene una clase, y que se dividen en tipos de datos simples, tales como string, entero, real, etc. y tipos de datos cuyo rango son objetos (otras instancias).

Este API es muy útil para manipular la información contenida en el archivo OWL de la aplicación, debido a que ofrece ventajas que permiten no tener que recorrer el archivo cada vez que se necesite consultar alguna información, sino solo se hace uso de los métodos que Jena ofrece.

- **Hibernate**

Es un framework de persistencia para los objetos, el cual permite mapear clases de java a tablas en la Base de Datos con la cual se esté trabajando. Para esto se deben crear los objetos de negocio

y sus relaciones y luego una serie de archivos xml que indican a Hibernate qué objeto se guarda en qué tabla, indicando la relación entre propiedades del objeto y columnas de la tabla.

Entre las características más resaltantes se pueden mencionar:

- Hibernate se adapta al proceso de desarrollo de software, sin importar si se parte de una base de datos en blanco o de una ya existente.
- Mapeo de clases en Java a tablas de una base de datos.
- Ofrece consulta de datos y facilidades de recuperación: Hibernate genera las sentencias SQL y libera al desarrollador del manejo manual de los datos que resultan de la ejecución de dichas sentencias, manteniendo la portabilidad entre todas las bases de datos con un ligero incremento en el tiempo de ejecución.
- Hibernate puede ser utilizado en aplicaciones Java independientes o bajo aplicaciones Java EE haciendo uso de servlets o EJB beans de sesión.

- **Struts**

Struts es un framework que implementa el patrón de arquitectura MVC (*Model-view-controller*), en él se implementa un solo controlador (*ActionServlet*) que evalúa las peticiones del usuario mediante un archivo configurable (*struts-config.xml*).

Entre las características de Struts se pueden mencionar:

- Configuración del control centralizada.
- Las relaciones entre acciones y página se especifican por tablas XML en lugar de codificarlas en los servlets o páginas.
- Existen librerías de entidades para facilitar la mayoría de las operaciones que generalmente realizan las páginas JSP: recorrido de lista, condicionales, elementos html, etc.
- Struts contiene herramientas para validación de campos de plantillas bajo varios esquemas que van desde validaciones locales en la página (en JavaScript) hasta las validaciones de fondo hechas a nivel de las acciones.

Struts permite que el desarrollador se concentre en el diseño de aplicaciones complejas como una serie simple de componentes del modelo y de la vista intercomunicados por un control centralizado. Diseñando de esta manera puede obtenerse una aplicación más consistente y más fácil de mantener.

- **JSON-RPC-Java**

Es un protocolo simple que ha sido diseñado para ser fácil de usar. El mecanismo general consiste en dos pares que establecen una conexión de datos. Durante la vida de la conexión, los pares pueden invocar los métodos proporcionados por el otro par. Para invocar un método se hace una petición y a menos que la petición sea una notificación esta debe ser contestada con una respuesta.

JSONRPC-Java es considerado una pieza fundamental en las aplicaciones Web que quieren llamar métodos de aplicaciones en el servidor sin necesidad de recargar la página.

Entre sus características podemos destacar:

- Llamadas dinámicas a métodos en el servidor desde aplicaciones Web dinámicas, en el JavaScript sin que la página sea recargada.
- Comunicación asíncrona.
- Transparencia de los objetos en java a los objetos del JavaScript.
- Protocolo ligero similar a XML-RPC aunque mucho más rápido.
- Asegura el modelo de la seguridad de J2EE con la exportación específica de la sesión de objetos.
- Esta soportado por Internet Explorer, Mozilla, Firefox, Safari, Opera y Konqueror.

2.4.4 Arquitectura de la Aplicación

Una vez especificados los requerimientos, herramientas y tecnologías a utilizar, se define la estructura lógica de la aplicación, es decir, los componentes que la aplicación debe tener y las relaciones entre los mismos.

En la Figura 13 se muestra la arquitectura de la aplicación de Visualización de la Ontología de Insectos Acuáticos:

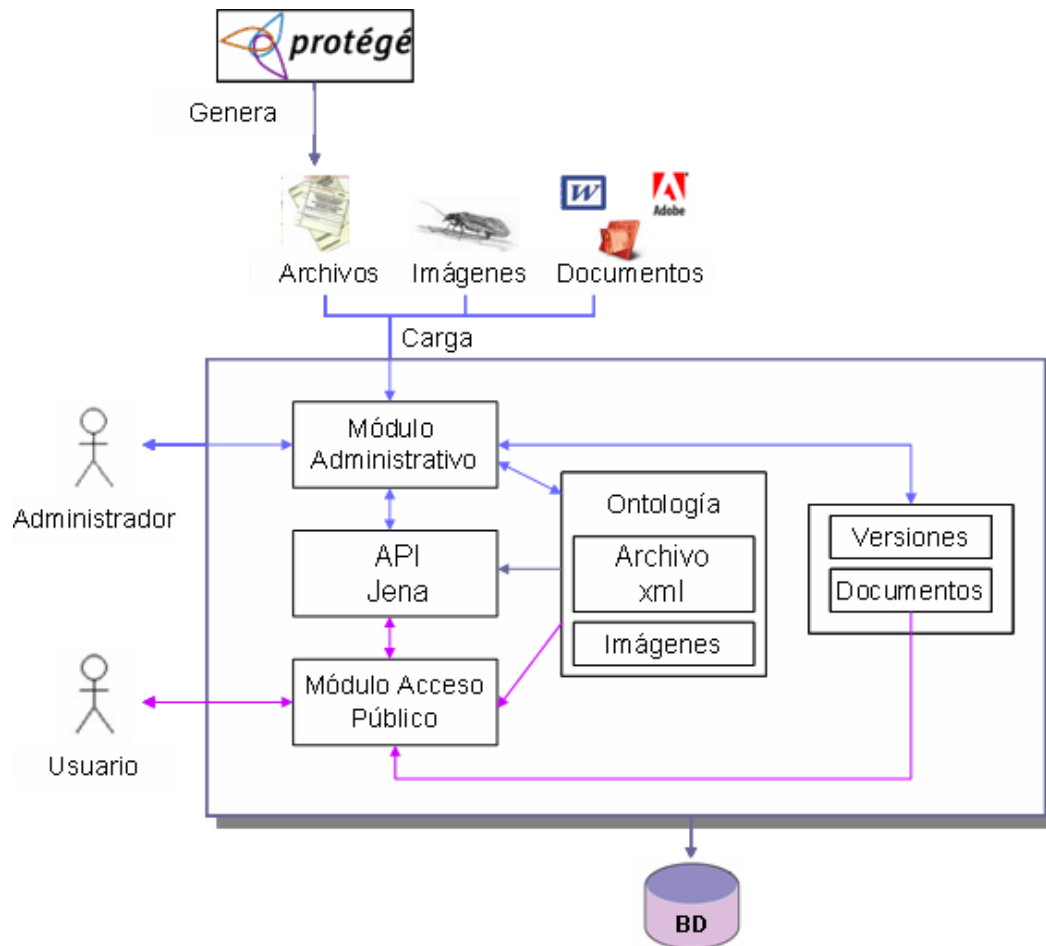


Figura 13: Arquitectura de la Aplicación.

La Ontología de Insectos Acuáticos es codificada usando la herramienta Protégé, esta Ontología es almacenada en un archivo de extensión OWL, que sirve como entrada para la aplicación. También se incluyen fotos tomadas por los expertos y otras personas, que sirven para la identificación visual de algún insecto en particular. Junto con los documentos, estos archivos son suministrados al sistema únicamente a través del módulo de administración. Internamente la aplicación los guarda en el servidor y los procesa de diversas formas dependiendo de las peticiones que realice el usuario a través de la interfaz de acceso público. El API de Jena se usa para manipular la Ontología.

2.4.5 Diseño del modelo de navegación entre páginas

Antes de implementar la aplicación, se hizo necesario conocer el flujo básico de navegación entre las páginas, éste se encuentra modelado en un diagrama de robustez el cual puede consultar en el Anexo 2.

2.4.6 Definición de la Base de Datos

Para la aplicación se han identificado las siguientes entidades:

- **Administradores:** tabla que contiene registros asociados a cada uno de los administradores que tienen acceso a la manipulación de otros datos a través de la aplicación. La definición de esta tabla y sus atributos se presenta en la Figura 14.

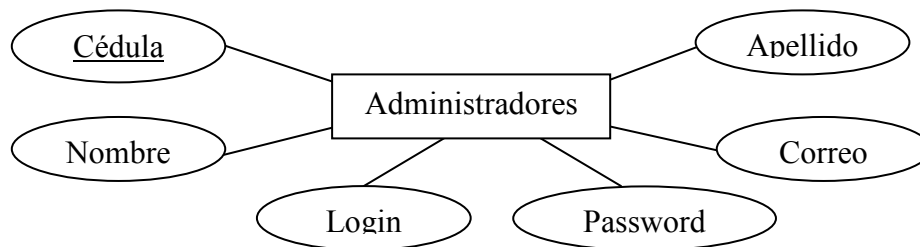


Figura 14: E/R Entidad Administrador.

- **Documentos:** tabla que posee los registros que hacen referencia a documentos almacenados en el servidor en donde se encuentra alojada la aplicación. El diseño de la tabla y sus atributos se presenta en la Figura 15.

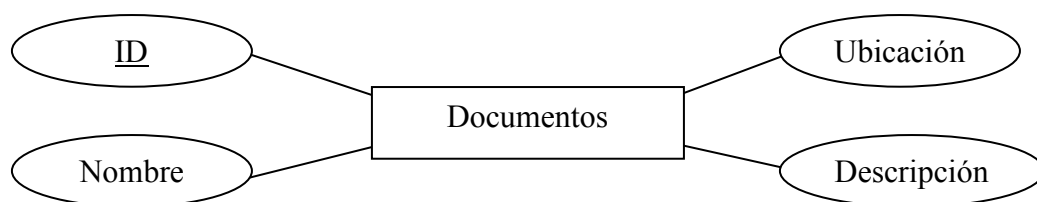


Figura 15: E/R Entidad Documentos.

- **Enlaces:** tabla que posee los registros asociados a enlaces, los cuales pueden ser de interés para el usuario que consulta la aplicación. La definición de esta tabla y sus atributos se presenta en la Figura 16.

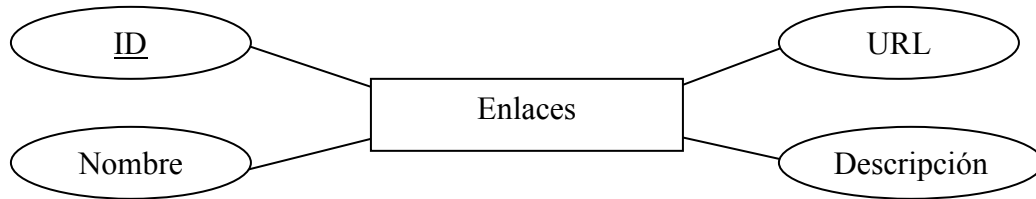


Figura 16: E/R Entidad Enlaces.

- **Imágenes:** tabla que posee los registros en donde se asocia una imagen a una instancia determinada. La estructura de la misma se presenta a continuación:



Figura 17: E/R Entidad Imágenes.

- **Mensajes:** tabla que contiene registros asociados a cada uno de los mensajes emitidos por los usuarios del sistema. La definición de esta tabla y sus atributos se presenta en la Figura 18.

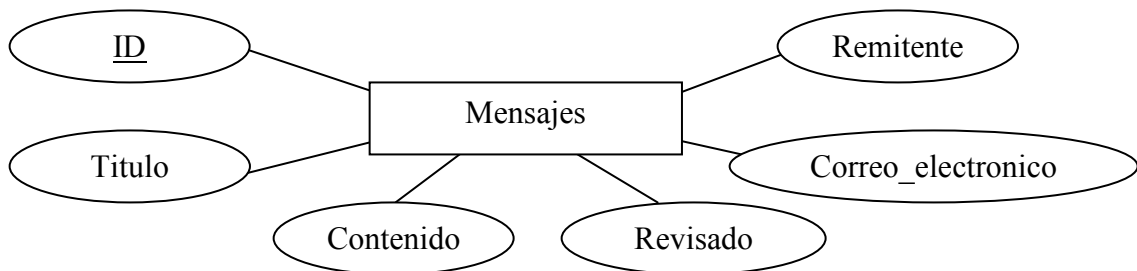


Figura 18: E/R Entidad Mensajes.

Para la aplicación se han identificado las siguientes relaciones:

- **Administrar_Documentos:**

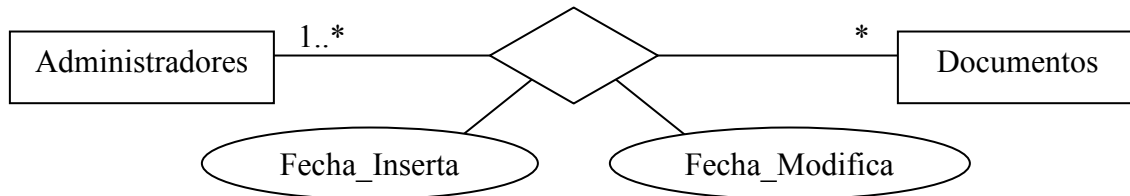


Figura 19: E/R Relación Administrar Documentos

- **Administrar_Enlaces:**

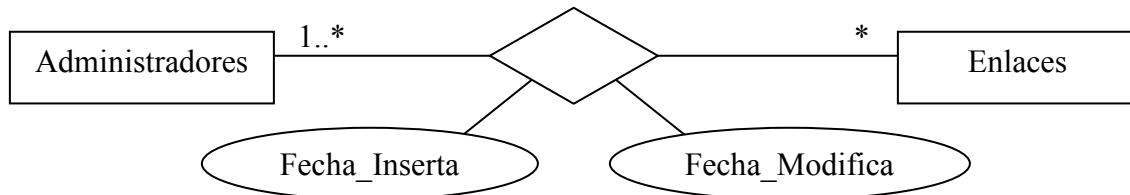


Figura 20: E/R Relación Administrar Enlaces

- **Administrar_Imágenes:**

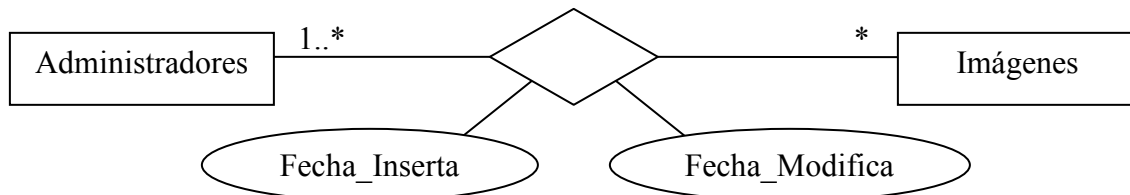


Figura 21: E/R Relación Administrar Imágenes

- **Revisar_Mensajes:**

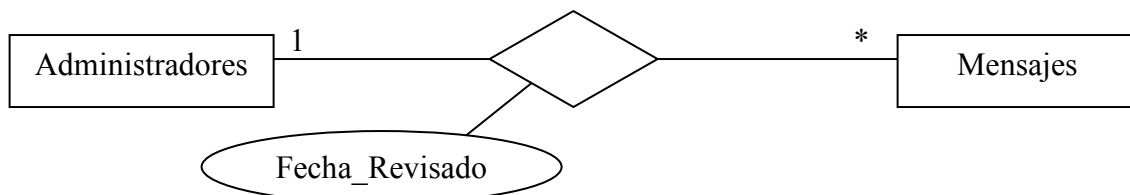


Figura 22: E/R Relación Responder Mensajes

Modelo Final E/R: a continuación se muestra el modelo de entidad relación definido para la base de datos del sistema:

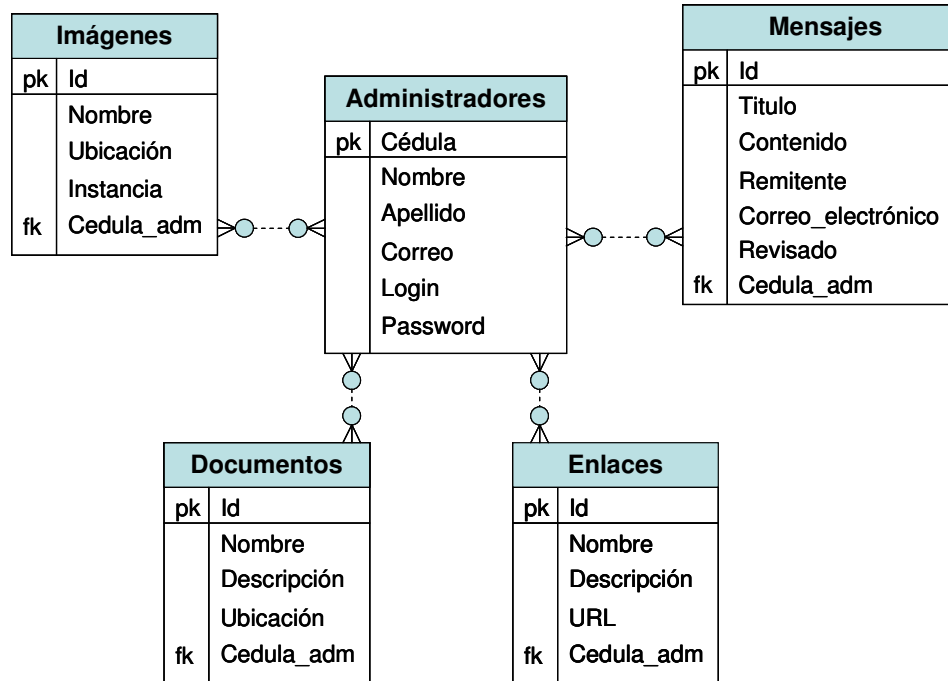


Figura 23: Diagrama Entidad – Relación.

2.4.7 Estructura lógica de desarrollo de la Aplicación

Para el desarrollo de la aplicación el código se estructuró en tres capas, en las cuales se usan las tecnologías que fueron explicadas anteriormente. Estas capas o niveles son:

- **Capa de modelo:** es la representación específica del dominio de la información sobre la cual funciona la aplicación. El modelo es otra forma de llamar a la capa de dominio. En esta capa podemos encontrar los siguientes paquetes:
 - com.ontologia.insectosacuaticos.model: en este paquete se encuentran las clases que son mapeadas a las tablas de la Base de Datos a través de Hibernate.
 - com.ontologia.insectosacuaticos.dao: se encuentran las interfaces con los métodos necesarios para acceder a la Base de Datos y manipular los registros que allí se encuentran.
 - com.ontologia.insectosacuaticos.dao.impl: se encuentran las clases que implementan las interfaces del dao.

- **Capa de negocios:** en ella se encuentran las clases que sirven de puente entre las clases controladoras y del modelo. También implementan lógica que no necesariamente son propias del acceso a la Base de Datos.
- **Capa Controladora:** es quien atiende las peticiones o acciones del usuario, e invoca métodos para cambiar el modelo y probablemente la vista.
 - `com.ontologia.insectosacuaticos.web.struts.action`: almacena las clases o componentes de control, encargados de coordinar las actividades de la aplicación, que van desde la recepción de datos del usuario hasta las verificaciones de forma y selección de un componente del modelo a ser llamado.
 - `com.ontologia.insectosacuaticos.web.struts.form`: son clases que representan a un formulario en particular, el cual debe estar mapeado en un archivo de configuración, y que permite dejar los datos que un usuario introduce en el bean de manera automática.
- **Módulo de Vista:** presenta el modelo en un formato adecuado para que el usuario interactúe con la aplicación (interfaz de usuario). En él se encuentran las páginas jsp desarrolladas las cuales van a ser vistas por el usuario.

2.4.8 Diseño de la Interfaz de Usuario

El diseño de la interfaz es uno de los elementos principales a la hora de realizar cualquier sistema, debido a que es a través de ella que el usuario interacciona con la aplicación y dependiendo de la misma, será grato o no el uso de ésta. Según Dorado (1997), la interfaz se define como el conjunto de trabajos y pasos que seguirá el usuario, durante todo el tiempo que se relacione con el programa, detallando lo que verá y escuchará en cada momento, y las acciones que realizará, así como las respuestas que el sistema le proporcionará.

Debido a la importancia de la interfaz, muchos han sido los autores que han tratado el tema y han propuesto diversos criterios a seguir a la hora de diseñar una interfaz, todos ellos toman en cuenta varios puntos de vista referentes a cómo el usuario se sentirá al momento de interactuar con el sistema, y cómo hacer que esta experiencia sea grata para él. Estos criterios los podemos agrupar

en los siguientes tópicos que tocan de manera general los puntos a tener en cuenta y que pueden condicionar la usabilidad de las páginas Web:

- **La estructura del sitio:** se refiere a que el sitio debe tener una estructura bien definida, en donde cada sección solo trate un tema o tópico en particular, que tenga un nivel de profundidad adecuado para la comprensión de la información que se presenta y que siempre le indique al usuario en donde está ubicado, así como la forma de ir a cualquier otra sección del sitio.
- **La prevención de los errores:** el sitio debe procurar que el usuario cometa el mínimo de errores posibles y los pueda corregir.
- **La consistencia estilística:** se debe ser consistente en cuanto al diseño de la interfaz en todo el sitio, se deben evitar saltos o rompimientos en el diseño, ya sea de color, forma, tamaño, letra, etc., debido a que esto puede desorientar al usuario.

Nielsen (1990), enumeró diez heurísticas de usabilidad. En ellas se hacen una serie de recomendaciones para verificar puntos críticos en interfaces de usuario, para asegurar que esta tuviera un alto nivel de usabilidad. Estas heurísticas se presentan a continuación:

1. **Visibilidad:** el sistema debe transmitir en todo momento su estado. El usuario debe saber si se está produciendo la carga de una página o si esta función se ha interrumpido por algún problema en la red. De no ser así, el usuario no sabrá a qué atenerse.
2. **Correspondencia entre el sistema y el mundo real:** la información presente en las páginas debe considerar la realidad de su usuario ideal.
3. **Libertad y posibilidad de control del sistema por parte del usuario:** el sistema debe hacer visible la organización de sus contenidos desde la primera página.
4. **Coherencia interna y externa:** y conformidad con los estándares comúnmente aceptados. Para evitar que el usuario se desoriente, todas las páginas de un sitio deben ser homogéneas, estilística y conceptualmente.
5. **Cuidado en la prevención de los errores.**
6. **Reconocer mejor que recordar:** la interfaz debe permitir una fácil comprensión de sus funciones, en lugar de obligar al usuario a memorizar una serie de procedimientos.
7. **Flexibilidad y eficiencia:** la interfaz debe respetar también las necesidades del usuario experto poniendo a su disposición herramientas para un uso más avanzado del sitio.

8. **Diseño minimalista:** para evitar tiempos de carga largos sin renunciar a calidades de tipo estético.
9. **Proporcionar al usuario los medios para corregir los errores.**
10. **Proporcionar los instrumentos de ayuda** y las instrucciones de uso del sitio.

Así mismo Shneiderman (1998) define los siguientes cinco atributos para la usabilidad:

1. **Facilidad de aprendizaje:** se refiere a que tan rápido el usuario va a aprender a usar un sistema con el cual no había tenido contacto previamente. Este punto se refiere a la consecución de tareas básicas por parte de un usuario novato.
2. **Velocidad de desempeño:** una vez que el usuario ha aprendido a utilizar el sistema, se va a ponderar el lograr la velocidad con que puede completar una tarea específica.
3. **Tasas de error por parte de los usuarios:** este atributo se refiere a aquellos errores que comete el usuario al utilizar el sistema. Una aplicación ideal evitaría que el usuario cometiera errores y funcionaría de manera óptima a cualquier petición por parte del usuario. Es vital que una vez que se produzca un error, el sistema se lo haga saber rápida y claramente al usuario, le advierta sobre la severidad del mismo y le provea de algún mecanismo para recuperarse de ese error.
4. **Retención sobre el tiempo:** cuando un usuario ha utilizado un sistema tiempo atrás, y tiene la necesidad de utilizarlo de nuevo la curva de aprendizaje debe ser significativamente menor que el caso del usuario que nunca haya utilizado dicho sistema. Esto es de primordial importancia para aplicaciones usadas intermitentemente.
5. **Satisfacción subjetiva:** este atributo se refiere a la impresión subjetiva del usuario respecto al sistema.

Aunque no se hizo una evaluación a la usabilidad de la aplicación, al momento del desarrollo e implementación de la misma, se tuvo en cuenta lo especificado en los puntos tratados anteriormente, los cuales se especifican a continuación:

- La consistencia de una aplicación se basa en seguir una uniformidad en las entradas y salidas del sistema. Para la aplicación de Visualización de Insectos Acuáticos se es consistente en varios aspectos, por ejemplo todas las páginas cuenta con un encabezado el cual tiene un *banner* con el logo de la aplicación, esta imagen es siempre la misma tanto para la interfaz

correspondiente al módulo de acceso público como la del módulo de administración, lo cual da un sentido de pertenencia al usuario dentro de la aplicación así se encuentre ubicado en cualquier sección de la misma. Debajo de este *banner* se encuentra el menú, de fácil acceso y visualización, en él están contenidas las funcionalidades básicas del módulo en el que se encuentre el usuario.

- Debajo del menú se encuentra una barra de navegación, la cual siempre indica al usuario en donde está ubicado por si quiere regresar al menú anterior y para que aprenda la ruta de cómo llegar a esa sección de interés. Estos componentes no varían entre las páginas.
- El contenido se presenta en la parte inferior del *banner* y de un tamaño adecuado para que no sea visible la barra de desplazamiento horizontal. Esta información está distribuida sobre el espacio de manera armónica, sin el uso de colores fuertes que puedan afectar la visión del usuario. Los colores adoptados para esta aplicación son el verde oliva, el beige y el gris claro, los cuales dan un ambiente sobrio a la aplicación sin recargar la misma. La página principal de la aplicación se observa en la Figura 24.



Figura 24: Ejemplo de Interfaz de la Aplicación

- Al momento de usar iconos en la aplicación, es conveniente que se conozca la importancia de los mismos y se asocie de una forma conveniente la funcionalidad a la cual se desea acceder a través de ellos. Los iconos están sujetos a la interpretación individual y subjetiva de cada persona a partir de su experiencia, nunca son totalmente claros e inequívocos y existe riesgo de malentenderlos. Es por esto que en gran parte de la aplicación son usados juntos a un texto alusivo a la funcionalidad que representan, para que el usuario se sienta más cómodo y sea más agradable su interacción con el sistema. Algunos de estos iconos se muestran a continuación en la Figura 25.



Figura 25: Ejemplo de iconos de la Aplicación

También existen iconos que no presentan un texto alusivo, sin embargo estos tiene imágenes que se asocian fácilmente a la funcionalidad que representan. Ejemplos de estos iconos se presentan en la Figura 26.



Figura 26: Ejemplo de iconos de la Aplicación: Eliminar, Editar y Descargar

- La aplicación tiene dos menus bien definidos, uno para la interfaz pública y otro para la interfaz del administrador, con esto se evita ofrecer opciones al usuario que no están disponibles para él mismo, evitando los molestos mensajes de acceso denegado.

2.4.9 Implementación de las funcionalidades

En este proceso se procede a codificar los requerimientos adquiridos en las etapas anteriores teniendo en cuenta todo lo ya definido hasta el momento: tablas, tecnologías, herramientas, estructura lógica.

A continuación se muestran las funcionalidades desarrolladas para el Módulo de Acceso Público del sistema:

- **Página de inicio:** esta interfaz es la entrada a la aplicación, en ella se encuentra una introducción al sitio, así como breves descripciones de los dos laboratorios involucrados en el desarrollo de la ontología a visualizar: Laboratorio de Inteligencia Artificial y el Laboratorio de Ecología de Aguas Continentales, con sus respectivos enlaces.

En esta pantalla se encuentra ubicado el formulario de acceso al módulo de administración del sistema. (Figura 27).



Figura 27: Página de inicio

- **Visualizador de Ontología:** esta sección del sitio muestra la Ontología que está cargada en el servidor de una forma sencilla para el usuario. El área que muestra el contenido tiene una región dividida en tres paneles, cada uno con una función en particular y que se muestran en la Figura 28. El primer panel muestra un árbol con todas las clases que contiene la Ontología, al momento de presionar alguna de estas clases se despliega en otro panel información

referente a la clase y las instancias asociadas a la misma. La información se muestra de dos maneras: una gráfica y una textual.

- De manera Gráfica: se muestra un grafo que contiene la clase actual con las relaciones hacia otras clases (Figura 29).
- De manera Textual: se muestra la descripción de la clase, así como los nombres de las relaciones existentes en la misma. También se muestran los atributos de la clase y la descripción de cada uno de ellos (Figura 29).

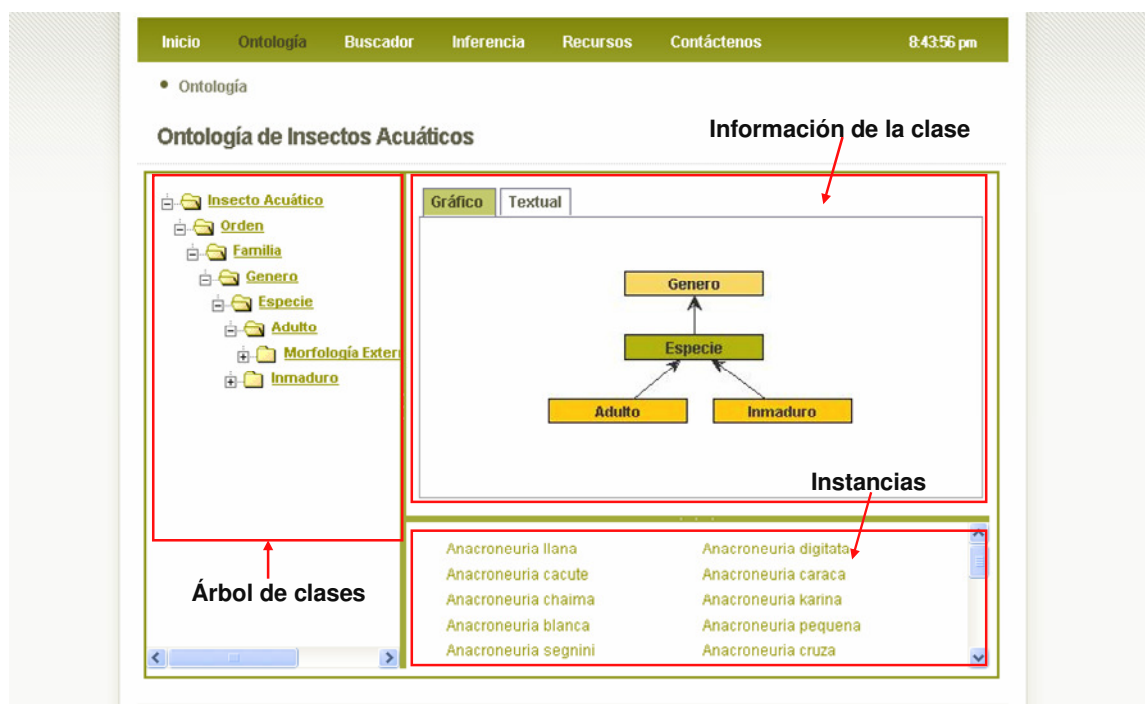


Figura 28: Página para la visualización de la Ontología de Insectos Acuáticos



Figura 29: Ejemplos de las formas de visualización: Gráfica y Textual

En el panel de instancias aparece el nombre de las instancias asociadas a la clase seleccionada, al hacer clic sobre ella se despliega una nueva ventana que contiene la información concerniente a la instancia, esta información está conformada por el nombre, la descripción y el valor de los atributos que posee. Debajo de la información se despliega una banda con las imágenes asociadas a la instancia (Figura 30).



Figura 30: Ejemplo de información mostrada al usuario referente a la Instancia Anacroneuria Bari

- **Buscador:** la aplicación cuenta con un buscador sencillo en donde se introduce la palabra que se desea buscar y el ámbito de la búsqueda: en la Ontología o en el sitio. Cuando se escoge Ontología el resultado de la búsqueda pueden ser clases e instancias, si se desea buscar sobre el sitio el resultado pueden ser enlaces o documentos (Figura 31).

Si el resultado es una clase se puede hacer ser clic sobre la misma y se mostrará una ventana con la información concerniente a esa clase, de la misma forma si se hace clic sobre una instancia. Los documentos que resultan de la búsqueda pueden ser descargados y lo enlaces visitados.

- **Inferencia:** en esta sección se muestra un enlace a la aplicación de Inferencia desarrollada por el Laboratorio de Inteligencia Artificial.
- **Recursos:** en recursos del sistema se encuentran documentos, enlaces y preguntas que pueden resultar de interés para el usuario. La forma de acceder a cada una de estas secciones puede ser de dos maneras: a través de la página inicial de recursos o mediante botones de acceso

ubicados en el lado derecho de la pantalla. Al momento de acceder a una sección de recursos el botón asociado a ella se presentará encendido (Figura 32).

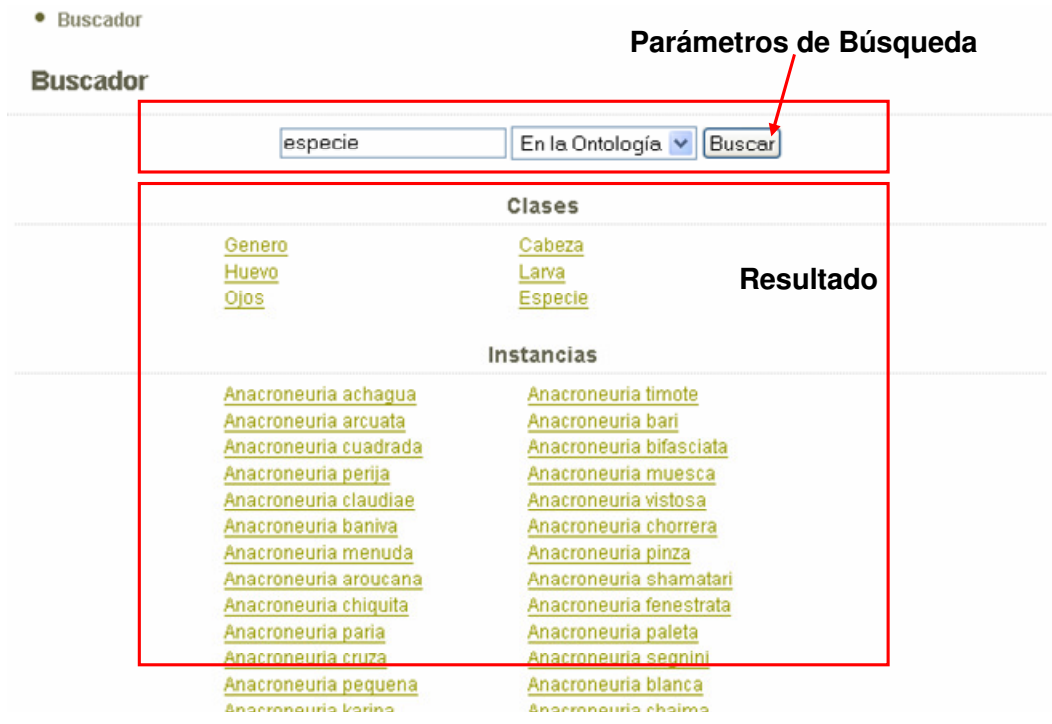


Figura 31: Ejemplo de una consulta sobre la Ontología



Figura 32: Pantalla de Recursos

Los documentos de interés pueden ser descargados al hacer clic sobre el nombre y los enlaces visitados al realizar el mismo procedimiento. Ambos se presentan en un número de seis (6) elementos por páginas y de existir más, se muestra la paginación para navegar por ella.

Las preguntas de interés también se listan en la sección de recursos, al hacer clic sobre alguna pregunta se abre una ventana con la respuesta correspondiente y los enlaces para ver la información de cada una de las instancias que se tienen como respuesta. Estas preguntas junto con sus respectivas respuestas fueron suministradas por el experto.

- **Contáctenos:** en esta sección se encuentra la página de contactos de la aplicación, en la que se tiene cada uno de los correos de las personas que han sido parte del proyecto. También se incluye una sección de buzón de sugerencias, con la finalidad de que el usuario pueda enviar algún mensaje al administrador del sitio con alguna inquietud o sugerencia. Por último se tiene la ayuda, la cual despliega en una nueva ventana una pequeña ayuda sobre la aplicación y la explicación de los módulos de acceso público. La pantalla de contáctenos se observa en la figura 33.



Figura 33: Pantalla Contáctenos

A continuación se muestran las funcionalidades desarrolladas para el Módulo de Administración del sistema:

- **Página de Inicio:** al autenticarse el administrador accede a la pantalla de Bienvenida al módulo (Figura 34), que posee el menú correspondiente al área de administración del sitio y un enlace para modificar sus datos personales.



Figura 34: Pantalla de inicio para el módulo de Administración

- **Modificar Perfil:** en esta sección se muestra un formulario con los datos asociados al administrador actual, si al momento de cambiarlos alguno de ellos es incorrecto, se muestra un mensaje de error para que se corrija y se pueda terminar la transacción de manera exitosa (Figura 35).

Figura 35: Pantalla de para la modificación de datos del Administrador

- **Administrar Enlaces de Interés:** al momento de ingresar en esta sección, se muestran los enlaces actuales en una tabla, y se muestran las tres transacciones que el administrador puede realizar:
 - **Agregar Enlace:** en la página de consulta se muestra un botón al lado derecho de la pantalla con la opción de agregar. Al hacer clic sobre ella se redirecciona a una pantalla que muestra un formulario con los campos requeridos a la hora de ingresar un nuevo enlace.
 - **Editar Enlace:** Al hacer clic sobre el icono editar de algún enlace existente en la tabla se redirecciona a una pantalla con un formulario para cambiar los valores asociados al enlace actual.
 - **Eliminar Enlace:** Un enlace se elimina al hacer clic sobre el icono eliminar que se tiene al lado del registro en la tabla de enlaces actuales de la aplicación.



Figura 36: Pantalla de entrada para administrar enlaces.

Aparte de las transacciones anteriores se tiene en esta interfaz un módulo de búsqueda que discrimina los enlaces a mostrar en la tabla de enlaces actuales. Por defecto la primera vez lista todos los enlaces existentes en la Base de Datos. Esta interfaz se observa en la Figura 36.

- **Administrar Documentos de Interés:** al ingresar a esta sección, se muestra una tabla con los documentos que se encuentran en el sistema y se muestran las tres transacciones que el administrador puede realizar:
 - **Agregar Documento:** muestra un botón al lado derecho de la pantalla con la opción de agregar. Al hacer clic sobre el botón se va a una pantalla que muestra un formulario, en él existe un campo para la carga del documento.
 - **Editar Documento:** Al hacer clic sobre el icono editar de algún documento existente en la tabla, se redirecciona a una pantalla con un formulario para cambiar los valores asociados al documento, sin embargo, no se muestra el campo de carga del archivo.
 - **Eliminar Documento:** Un documento se elimina al hacer clic sobre el icono eliminar que se tiene al lado del registro. Al eliminar se borra el documento del servidor.

Al igual que la sección de administrar enlaces, aquí también se cuenta con un módulo de búsqueda sobre los documentos actuales en el sistema. Por defecto la primera vez lista todos los

documentos existentes, entre ellos también se encuentran las versiones de la Ontología. Esta interfaz la podemos observar en la Figura 37.



Figura 37: Pantalla de entrada para administrar documentos.

- **Administrar Ontología:** al ingresar a esta sección se muestra en la pantalla un marco de trabajo dividido en dos, en el primer panel se muestra el árbol de clases asociado a la ontología actual que ha sido cargada en el servidor, al hacer clic sobre alguna de las clases se muestra en el otro panel la lista de instancias asociadas a esa clase junto con la opción agregar instancia (Figura 38).

Al hacer clic sobre alguna de las instancias se redirecciona hacia una página que posee un formulario, en donde se piden los datos asociados a la instancia y se muestran los valores que ésta posee, el usuario puede modificar o agregar nuevos valores. Aquí también se puede hacer la carga de imágenes para la instancia actual. Al hacer clic sobre el botón de agregar se muestra el formulario vacío para que el usuario introduzca los datos, sin embargo no se ofrece la opción de cargar imágenes ya que esto solo ocurre cuando la instancia ha sido agregada en la Ontología. Un ejemplo del formulario se muestra en la Figura 39.



Figura 38: Marco de trabajo para modificar y/o agregar instancias.



Figura 39: Formulario para modificar y/o agregar instancias.

En esta sección también se da la opción de cargar una nueva versión de la Ontología, en donde se muestra una pantalla con un formulario para cargar un nuevo documento OWL y

colocar el número de versión. También se muestran las versiones de la Ontología, para que el Administrador pueda descargarlas en caso de que lo crea conveniente (Figura 40).

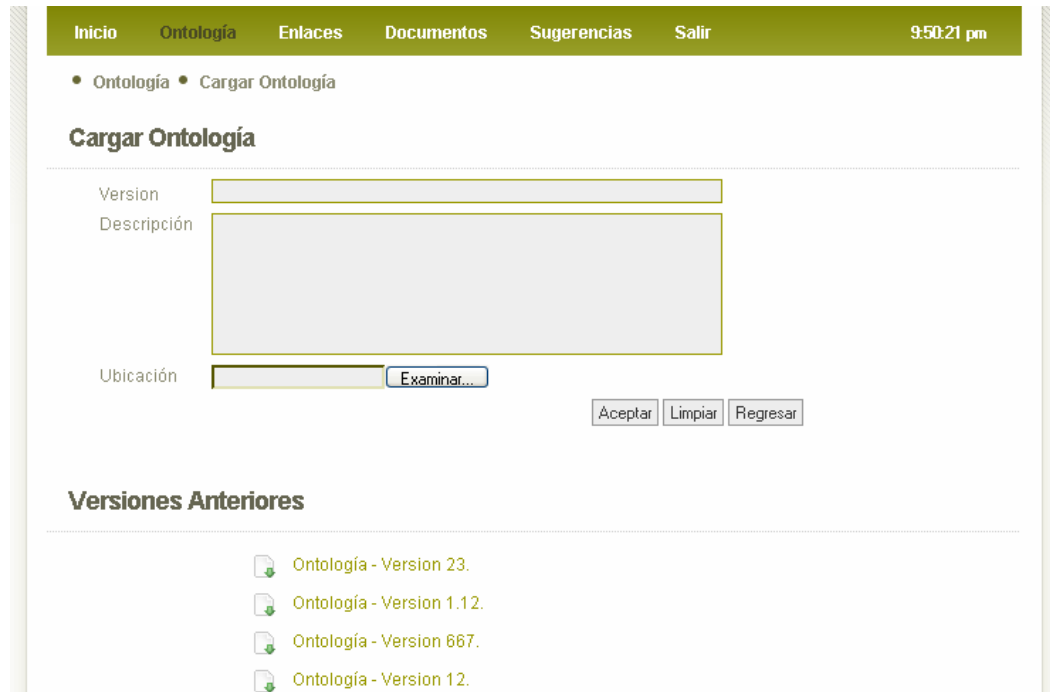


Figura 40: Pantalla de carga de Ontología

- **Visualizar Sugerencias:** esta sección muestra las sugerencias enviadas por los usuarios del sistema (Figura 41). Se muestra una lista con todos los mensajes recibidos, en donde los que ya han sido leídos por el administrador poseen un color de letra blanco y los no leídos se encuentran de color negro y en negrita. Acá también se cuenta con el módulo de búsqueda sobre mensajes.

Si se desea visualizar el contenido del mismo, solo hace falta hacer clic sobre el mensaje. Para eliminar se debe hacer clic sobre el icono eliminar que se encuentra al lado del mensaje.



Figura 41: Pantalla de consulta de mensajes recibidos

CONSLUSIONES Y RECOMENDACIONES

Al culminar la aplicación de visualización para la Ontología de los Insectos Acuáticos se puede concluir lo siguiente:

La manera de visualizar la Ontología es muy cómoda y sumamente simple, es natural el navegar en la misma y la información se muestra de manera gráfica y textual. La información gráfica está referida al detalle de las clases que componen la Ontología, mientras que para las instancias se muestra la información textual de cada una de ellas, así como las imágenes asociadas.

Se implementaron los dos módulos definidos al inicio de la aplicación, de tal forma que pudieran cumplir con cada uno de los requerimientos tomados al evaluar las necesidades del usuario.

La aplicación no sólo está centrada en la visualización de la Ontología, sino también permite la carga de nuevas versiones para mantener la información actualizada, así como un editor de la misma, a través del cual se pueden agregar nuevas instancias y modificar el contenido de las instancias existentes, esto incluye la carga de imágenes. También cuenta con módulos de búsqueda, recursos y enlaces a sitios de interés.

Entre las dificultades encontradas a la hora de desarrollar la aplicación, se tiene el generar las imágenes dinámicas de acuerdo a la distribución de las clases contenidas en la Ontología, así como la poca información referente al uso de los procesos en Jena, tales como la modificación del archivo OWL para la carga y edición de instancias. Cabe destacar que se tuvo que hacer un proceso manual de traducción de la Ontología de archivos en formato Protégé a formato OWL para que la misma pudiese ser usada por la aplicación.

Es importante resaltar que la aplicación a pesar de estar destinada a visualizar la Ontología de los Insectos Acuáticos no está limitada sólo a este dominio, es decir, la aplicación fue desarrollada sin tener reglas que la limiten a un contexto en particular, si se crea otra Ontología siguiendo algunas normas definidas previamente en el documento, esta podrá ser desplegada por el visualizador y modificada por el editor de manera correcta, sin tener que modificar el código, lo

único que se debería hacer es variar son los títulos y textos que hagan alusión al contexto con el cual se esté trabajando.

Es por esto que si se desea crear una nueva aplicación de visualización para otro contexto, solo basta con modificar archivos a nivel de interfaz, no a nivel de procedimientos complejos de código, por lo cual se puede decir que es una aplicación no dependiente del contexto. Otra ventaja que ofrece la aplicación es que los títulos, texto y cualquier otro contenido estático, se encuentran centralizados en un archivo de configuración, el cual sería el único a modificar a la hora de cambiar el contenido de la aplicación.

Como trabajos futuros se recomiendan los siguientes:

- Diseñar e implementar un generador de aplicaciones para visualizar ontologías, el cual utilice el código de esta aplicación como base y tome por parámetros títulos y textos alusivos al contexto de la ontología que se desea visualizar.
- Al iniciar el desarrollo la Ontología de Insectos Acuáticos se presentaron problemas en cuanto a la migración automática de la Ontología de archivos Protégé a OWL, es por esto que se recomienda crear una aplicación que convierta de forma correcta y confiable, ontologías desarrolladas y guardadas en archivos de Protégé a ontologías definidas en formato OWL.

REFERENCIAS

ATELA S.; GENTA N.; GONZÁLEZ D. *Tutorial de jena*. [en línea] [Consulta: 11 jul. 2007]. Disponible en <http://www.fing.edu.uy/~lsilva/JenaTutorial.ppt>

BARRÓN A. *Ontologías en acción: Protégé, Owl*. [en línea] [Consulta: 09 nov. 2006]. Disponible en <http://www.matem.unam.mx/~rajsbaum/cursos/web/ontologias2.pdf>

BELLINGER G.; CASTRO D.; MILLS A. (2004) *Data, Information, Knowledge, and Wisdom*. [en línea] [Consulta: 08 nov. 2006]. Disponible en <http://www.systems-thinking.org/dikw/dikw.htm>

BERNERS T.; HENDLER J.; LASSILA O (2001). *The Semantic Web: a new form of Web content that is meaningful to computers will unleash a revolution of new possibilities*. Revista Scientific American. [en línea] [Consulta: 09 nov. 2006]. Disponible en <http://www.sciam.com/article.cfm>.

BRAY T. (2001) *What is RDF?*. [en línea] [Consulta: 06 dic. 2006]. Disponible en <http://www.xml.com/pub/a/2001/01/24/rdf1.html>.

CHAUDARY Y.; SCARIONI C. (2005) *Ontología de Insectos Acuáticos*. Trabajo Especial de Grado. Escuela de Computación. Facultad de Ciencias. Universidad Central de Venezuela.

CRESSA, C. (1994) *Community composition and structure of macroinvertebrates of the river Camurí Grande, Venezuela*. Verhandlungen der Internationalen Vereinigung für theoretische und angewandte Limnologie 26: 1008-1011.

DORADO C. (1997) *El diseño del interfaz y la navegación*. [en línea]. [Consulta: 15 ago. 2007]. Disponible en <http://www.xtec.es/~cdorado/cdora1/esp/disseny.htm>.

ECHEGOYEN, J. (1995) *Historia de la Filosofía. Volumen 1: Filosofía Griega* [en línea]. [Consulta: 23 jul. 2006]. Disponible en <http://www.e-torredebabel.com/Historia-de-la-filosofia/FilosofiaGriega/Presocraticos/Ontologia.htm>

ESCARZA S.; CASTRO S.; MARTIG S. (2005) *Visualización de Ontologías*. [en línea] [Consulta: 1 sep. 2006]. Disponible en <http://dc.exa.unrc.edu.ar/wicc/papers/Otros/41.pdf>

ESQUIVIAS J.; ZURITA J.; GOMEZ M.; VILLAR E.; SANCHEZ M.; ALONSO J. (2005). *Representación del Conocimiento mediante Ontologías: Ensayo sobre Citología Tiroidea*. [en línea] [Consulta: 05 dic. 2006]. Disponible en http://www.conganat.org/7congreso/final/vistaImpresion.asp?id_trabajo=211

GENESERETH, M.; FIKES, R. (1992). *Knowledge Interchange Format, Version 3.0 Reference Manual*. Technical Report Logic-92-1, Computer Science Department, Stanford University.

GÓMEZ A.; FERNÁNDEZ M.; CORCHO O. (2003) *Ontological engineering*. 1era ed, New York: Springer-Verlag, LLC. 420 p. ISBN-13: 9781852335519.

HEFLIN J.; VOLZ R.; DALE J. (2002) *Requirements for a Web Ontology Language*. Technical Report, Marzo 2002. [en línea] [Consulta: 05 dic. 2006]. Disponible en <http://www.w3.org/TR/2002/WD-webont-req-20020307/#usecase-portal>

LAMARCA M. (2006) *Hipertexto: el nuevo concepto de documento en la cultura de la imagen*. Tesis doctoral. Universidad Complutense de Madrid. [en línea] [Consulta: 01 sep. 2006]. Disponible en <http://www.hipertexto.info/>

LORENTE M. (2005) *Ontología sobre economía y recuperación de Información*. [en línea] [Consulta: 1 sep. 2007]. Disponible en <http://www.hipertext.net/web/pag259.htm>.

MAHESH, K. (1996). *Ontology Development for Machine Translation: Ideology and Methodology*. NMSU. Computing Research Laboratory. New Mexico. Technical Report MCCS-96-292.

NIELSEN J. (1990) *Ten Usability Heuristic.s* [en línea]. [Consulta: 15 ago. 2007]. Disponible en http://www.useit.com/papers/heuristic/heuristic_list.html.

NOY N, MCGUINNESS D. (2000) *Ontology Development 101: A Guide to Creating Your First Ontology*. [en línea] [Consultado: 09 nov. 2006]. Disponible en <http://www.cs.man.ac.uk/~carole/old/GGF%20Tutorial%20Stuff/ontology101.pdf>

PEREIRA Y.; RAMOS E.; NÚÑEZ H.; CASTRO M.; CASAÑAS R. (2006) *Aplicación de visualización de una Ontología para el dominio del análisis del semen humano*. [en línea]. [Consulta: 1 sep. 2006]. Disponible en <http://www.eafit.edu.co/ingciencia/revista5/OntologiaALS.pdf>

RAMÍREZ A. (2006) *Insectos Acuáticos Neotropicales - Una guía para aprender sobre insectos acuáticos Neotropicales*. [en línea] [Consultado: 03 de enero del 2007]. Disponible en <http://ites.upr.edu/ramirez/acuaticos.htm>

RAMÍREZ Z. (2006) *Las ontologías como herramienta en la Gestión del Conocimiento*. [en línea]. [Consulta: 18 oct. 2006]. Disponible en <http://www.congreso-info.cu/UserFiles/File/Info/Info2006/Ponencias/208.pdf>

RESH, V. H.; ROSENBERG, D. M. (1984) *The ecology of aquatic insects*. Praeger Publishers, New York.

RINCÓN, J.A.; CRESSA, C. (2000) *Temporal variability of macroinvertebrates assemblages in a neotropical intermittent stream, Venezuela*. *Archiv für Hydrobiologie* 148:421-432.

ROCHA R.; ALVARENGA L. (2004) *A Web Semântica e suas contribuições para a ciência da informação*. [en línea] [Consulta: 08 nov. 2006]. Disponible en <http://www.ibict.br/cionline/viewarticle.php?id=71&layout=html>

RODRÍGUEZ K.; RONDA R. (2005). *Web Semántica: un nuevo enfoque para la organización y recuperación de información en la web*. . Technical Report Acimed, vol. 13, núm. 6, November-December 2005.

RUSSEL, S.t; NORVING, P. (2004) *Inteligencia Artificial: Un Enfoque Moderno*. Corchado, Juan (trad.). 2da ed. Madrid: Pearson educación S. A. 1240 p. ISBN 84-205-4003-X.

SHNEIDERMAN B. (1998) *Designing the user interface*. Addison-Wesley.

SIERRA C. (2003) *Ontolingua*. [en línea]. [Consulta: 01 sep. 2006] Disponible en <http://www.loa-cnr.it/Papers/onions97.pdf>

STEVE, G.; GANGEMI, A.; PISANELLI, D. (1997). *Integrating Medical Terminologies with ONIONS Methodology* [en línea]. [Consulta: 01 sep. 2006] Disponible en <http://www.loa-cnr.it/Papers/onions97.pdf>

SERRANO C. (2007). *La Contabilidad en la Era del Conocimiento*, [en línea] [Consulta 19 ene. 2007]. Disponible en [5campus.org](http://www.5campus.org), Sistemas Informativos Contables. <http://www.5campus.org/leccion/introduc>

TRAMULLAS, J. (1999) *Agentes y Ontologías para el tratamiento de información: clasificación y recuperación en Internet* [en línea]. [Consulta: 17 oct. 2006] Disponible en <http://www.tramullas.com/papers/isko99.pdf>

WETZEL, R. (2000) *Limnology, Lake and river Ecosystems*. Third Edition. Academic Press.

Data, Information, Knowledge. [en línea] [Consulta: 08 nov. 2006]. Disponible en

<http://www.semanticweb.org>.

Hibernate [en línea] [Consulta: 11 jul. 2007]. Disponible en
<http://es.wikipedia.org/wiki/Hibernate>

Ontología [en línea] [Consulta: 23 jul. 2006]. Disponible en
<http://es.wikipedia.org/wiki/Ontología>

WSMO Studio. [en línea] [Consulta: 01 sep. 2006]. Disponible en <http://www.wsmostudio.org/>

ANEXO 1

GLOSARIO DE TÉRMINOS

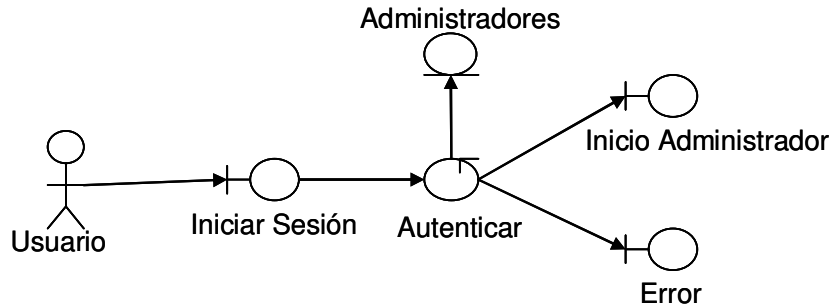
- **Agentes Inteligentes:** un tipo de programa informático que, por encargo de un usuario u otro programa, realiza de forma autónoma tareas que requieren cierto grado de inteligencia y aprendizaje. La mayoría de los agentes poseen las siguientes tres características: comunicación, inteligencia y autonomía.
- **Base de conocimiento:** término informal para referirse a una colección de información que incluye una ontología como un componente. Además de una ontología, una base de conocimiento debe contener información especificada en un lenguaje declarativo tal como reglas lógicas o sistemas expertos, aunque también incluye información no estructurada o formalizada expresada en lenguaje natural o en lenguaje de procesado.
- **Cardinalidad:** Define cuántos valores puede tener un slot individual (máximo y mínimo).
- **Casos o Instancias (instances):** Ejemplos específicos pertenecientes a alguna clase, esto es, objetos de una clase.
- **Clase:** Es un objeto que define una categoría. Describe conceptos en el dominio del discurso.
- **Clase jerárquica:** La compuesta por una colección de clases conectadas por relaciones "es un tipo de" (class hierarchy).
- **Editores de ontologías:** son las herramientas que permiten la codificación de una determinada ontología en base a un determinado lenguaje.
- **Facetas:** Se utilizan para definir qué tipo de valor puede contener un slot particular, valores permitidos, número de valores, etc. También se denominan restricciones de roles.
- **Herencia (inherence):** Es el proceso por el cual las subclases e instances de alguna clase heredan propiedades y valores definidos más arriba en la jerarquía.
- **Lenguajes de codificación de ontologías:** son los lenguajes utilizados para codificar las ontologías, permiten representar las ontologías y posibilitan su utilización por parte de las herramientas de edición de ontologías.
- **Metadatos:** son datos altamente estructurados que describen información, contenido, calidad, condición y otras características de los datos. Son datos sobre los datos.

- **RDF**: es un modelo de datos para objetos ("recursos") y para las relaciones entre ellos, provee una semántica simple para este modelo de datos, a la vez que este modelo de datos puede ser representado en sintaxis XML.
- **RDF Schema**: es un vocabulario para describir propiedades y clases de recursos RDF, con una semántica para generalización de jerarquías de aquellas propiedades y clases.
- **Relación**: Nuevo conocimiento que se obtiene por deducción, partiendo del conocimiento que se encuentra en la ontología, las relaciones utilizan variables.
- **Repositorio de ontologías**:
- **Roles o Propiedades (slots)**: Propiedades de cada concepto que describen varias características y atributos del concepto. Ayudan a definir las características de las clases.
- **Struts**:
- **Subclase**: Es en sí misma una clase, pero que es hija de alguna otra clase.
- **Taxonomía**: Una taxonomía es una organización jerárquica basada en un conjunto de atributos de los elementos clasificados. Dicha organización tiene la estructura de un árbol dirigido, por lo que no sólo no tiene ciclos, sino que un elemento no puede pertenecer al mismo tiempo a más de una clasificación (rama) dentro de la organización.
- **Tipo**: Define el tipo de valor (como cadena de caracteres, número, booleano, etc.)
- **Valor**: Describe una propiedad que se aplica a alguna clase o instance.
- **Variable**: Espacio vacío que puede llenarse preguntando a clases e instances. Cada variable comienza con un valor por defecto.
- **Web Semántica**: es una ampliación de la Web actual, se trata de un espacio donde la información tiene un significado bien definido, de manera que pueda ser interpretada tanto por agentes humanos como por agentes computerizados.
- **Web Services (Servicios Web)**: es una colección de protocolos y estándares que sirven para intercambiar datos entre aplicaciones, las cuales pueden estar desarrolladas en lenguajes de programación diferentes y ejecutadas sobre cualquier plataforma.
- **XML**: provee una sintaxis superficial para documentos estructurados, pero no impone restricciones semánticas en el significado de estos documentos.
- **XML Schema**: es un lenguaje para restringir la estructura de los documentos XML y también extiende XML con tipos de datos.

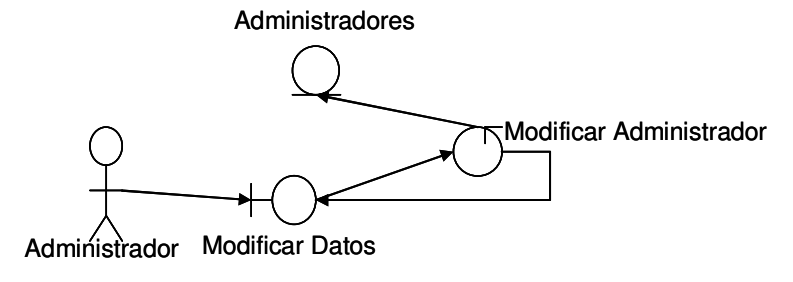
ANEXO 2
DIAGRAMA DE ROBUSTEZ

Módulo Administrativo

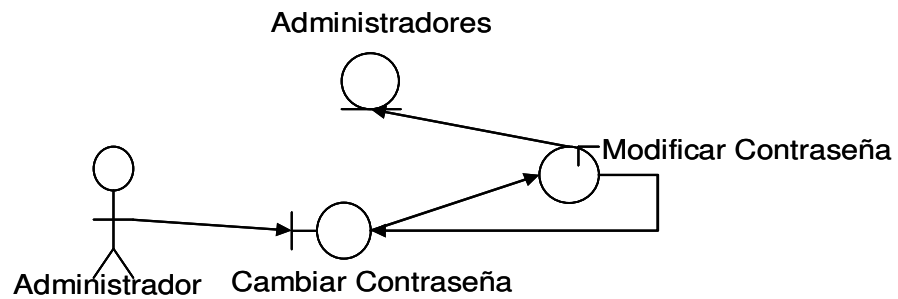
1. Iniciar Sesión



2. Modificar Datos Personales



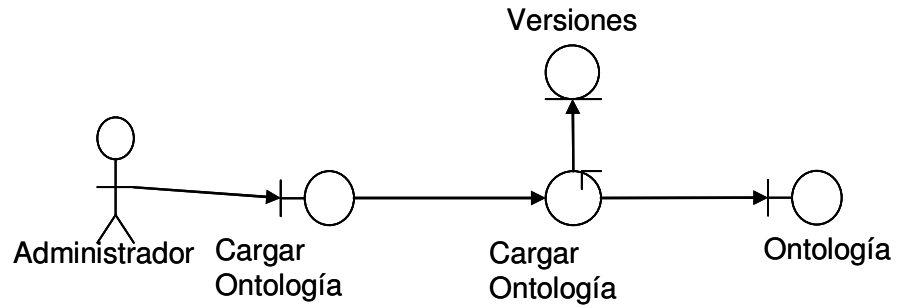
2.1. Cambiar Contraseña



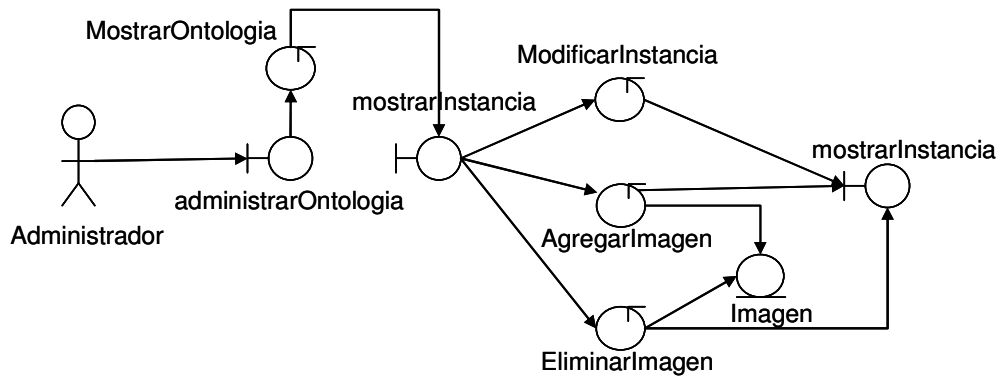
3. Administrar Ontología

3.1. Administrar Contenido Actual

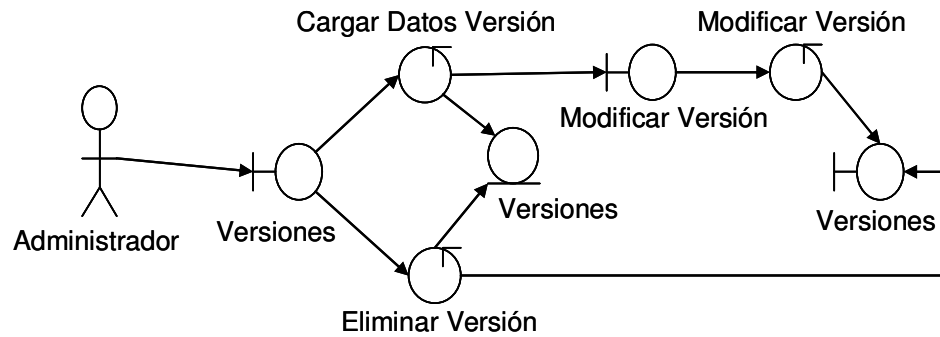
3.1.1. Cargar Ontología



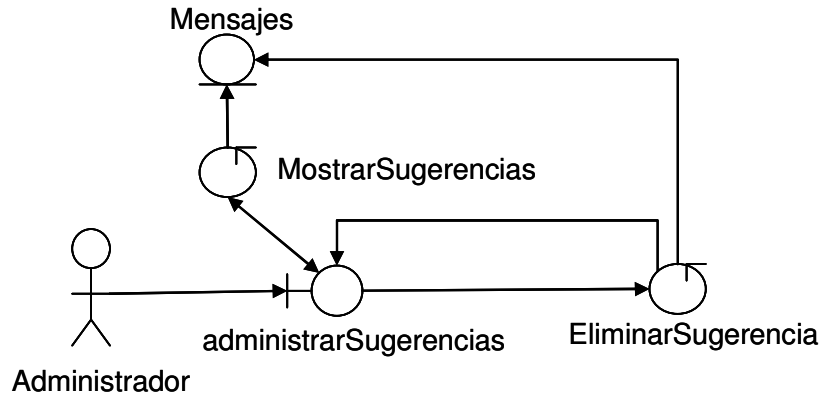
3.1.2. Modificar Instancia



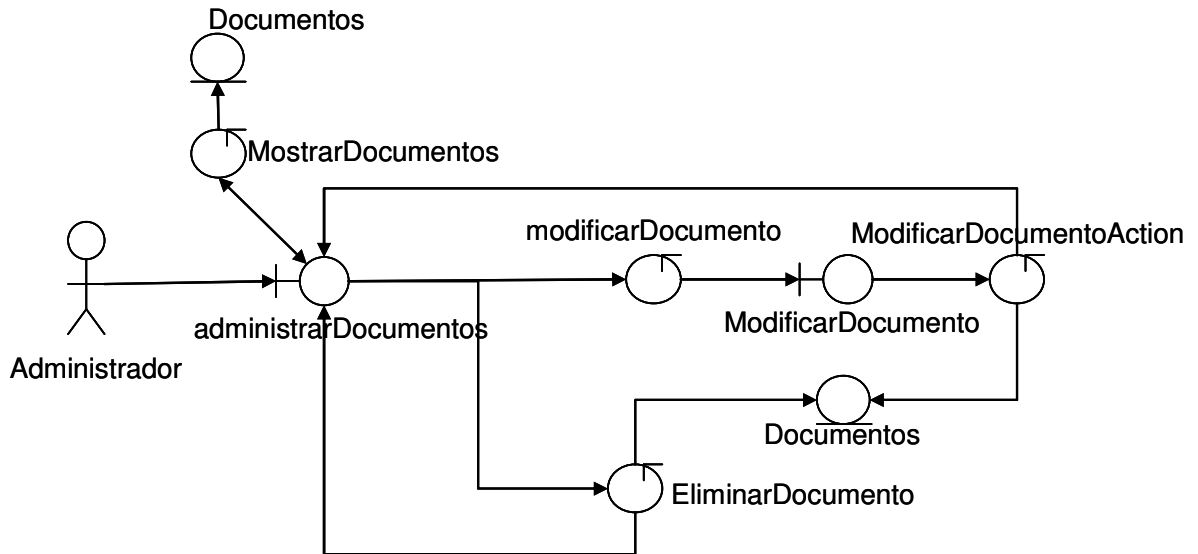
3.2. Administrar Historial



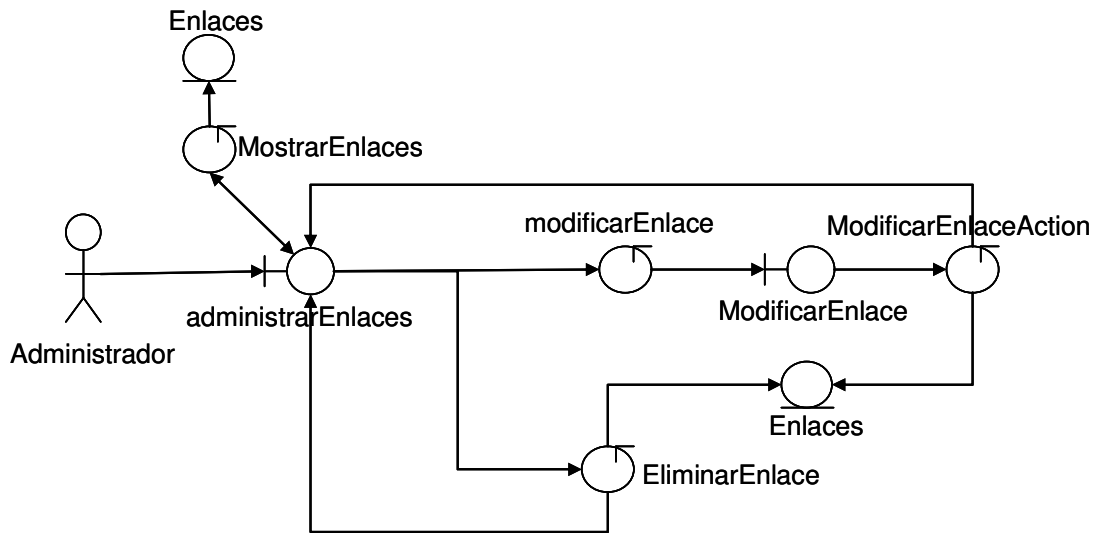
3.3. Gestionar Mensajes.



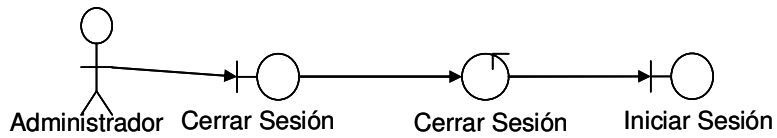
3.4. Administrar Documentos.



3.5. Administrar Enlaces.

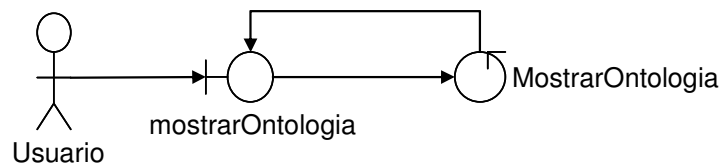


3.6. Cerrar Sesión.

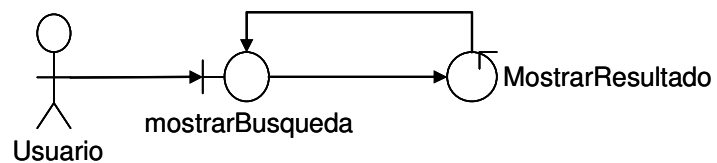


Módulo Acceso Público

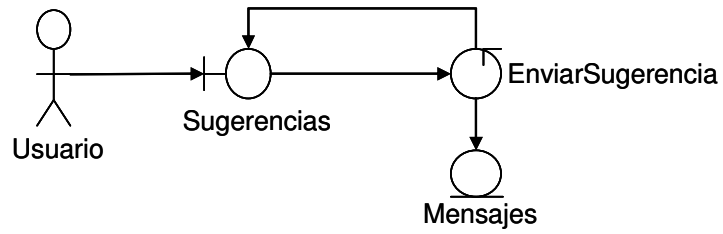
1. Consultar Ontología.



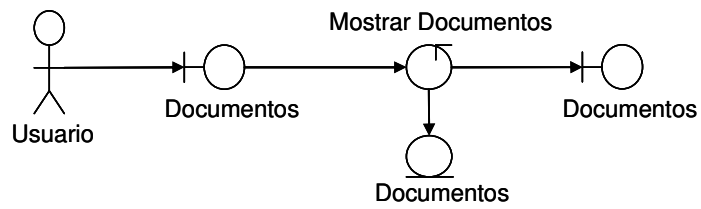
2. Realizar Búsquedas.



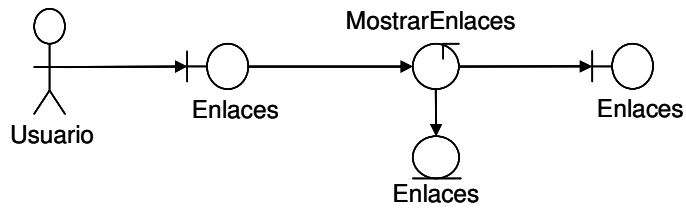
3. Enviar Sugerencias.



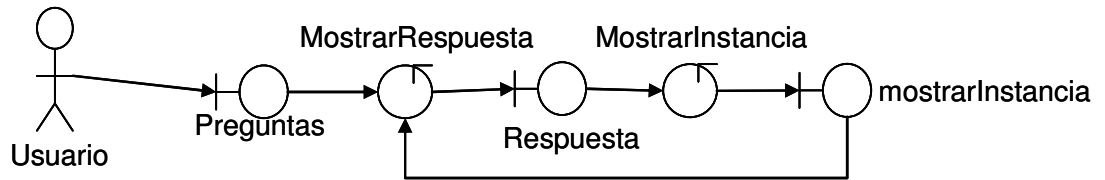
4. Visualizar Documentos de Interés.



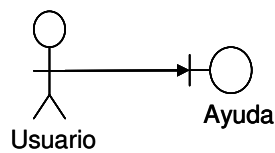
5. Visualizar Enlaces de Interés.



6. Visualizar Preguntas de Interés.



7. Consultar Ayuda del Sistema.



ANEXO 3 INSTALACIÓN DE SOFTWARE

1) Servidor Apache Tomcat 5.5.9

Los pasos necesarios para instalar, iniciar, detener y configurar el Tomcat se muestran a continuación:

Instalación del Servidor Tomcat

Windows:

- Instalar JDK, preferiblemente la versión j2sdk 1.4.2.
- Descargar la distribución binaria de Tomcat, preferiblemente la versión jakarta-tomcat-5.5.9.zip.
- Descomprimir e instalar el archivo a través del wizard instalador de Windows.
- Crear la variable de entorno CATALINA_HOME con el valor correspondiente al path del directorio en el cual está el Tomcat. Las variables de entorno se crean mediante la opción Sistema del Panel de Control.

Linux:

- Instalar JDK, preferiblemente la versión j2sdk 1.4.2.
- Descargar la distribución binaria de Tomcat, preferiblemente la versión jakarta-tomcat-5.5.9.tar.gz.
- Descomprimir el archivo mediante el comando:

```
tar xvfz jakarta-tomcat-5.5.9.tar.gz.
```
- Mover la carpeta dentro de /usr/local/ mediante el comando:

```
mv jakarta-tomcat-5.5.9 /usr/local/tomcat5.5.9.
```
- Crear la variable de entorno CATALINA_HOME con el valor correspondiente al path del directorio en el cual está el Tomcat dentro de /etc/profile. Las variables de entorno se crean mediante el comando:

```
export CATALINA_HOME=/usr/local/tomcat5.5.9
```

Iniciar y Detener el Servidor Tomcat

Windows:

Para iniciar el servidor Tomcat se debe ejecutar el archivo startup.bat y para detenerlo se debe ejecutar el archivo shutdown.bat.

Linux:

Para iniciar el servidor Tomcat se debe ejecutar el archivo startup.sh y para detenerlo se debe ejecutar el archivo shutdown.sh.

Para ambas plataformas dichos archivos se encuentran en el directorio bin de Tomcat. Para verificar que el servidor ha arrancado correctamente podemos acceder con el navegador al URL: <http://localhost:8080> (8080 es el puerto por defecto con el que se arranca Tomcat).

Configuración de Tomcat

La configuración de Tomcat se realiza a través del archivo server.xml ubicado en el directorio conf de Tomcat y del archivo web.xml el cual debe estar en la carpeta WEB-INF del contexto de la aplicación.

Es importante mencionar que Tomcat requiere que este instalado el Kit de Desarrollo Java (JDK).

2) Kit de Desarrollo Java (JDK)

Los pasos necesarios para instalar el JDK son los siguientes:

Instalación de JDK 1.4.2

Windows:

- Descargar el archivo de instalación del j2sdk, preferiblemente la versión 1.4.2.
- Ejecutar el archivo j2sdk versión 1.4.2, lo cual generará un directorio que contendrá el kit de desarrollo Java.
- Crear la variable de entorno JAVA_HOME con el valor correspondiente al path del directorio en el cual está el JDK. Las variables de entorno se crean mediante la opción Sistema del Panel de Control.

- Configurar la variable de entorno CLASSPATH, la cual indica dónde se encuentran las clases utilizadas para compilar el programa. Y es a partir de aquí que todo se ejecuta sobre la máquina virtual de Java y es de esta manera que se logra la independencia de la plataforma.

Linux:

- Descargar el archivo de instalación del j2sdk versión para Linux en binario, preferiblemente la versión `jdk-1_4_2-linux-i586.bin`.
- Dar los permisos de ejecución mediante el comando:
`chmod +x jdk-1_4_2-linux-i586.bin`
- Ejecutar el archivo mediante el comando:
`./jdk-1_4_2-linux-i586.bin`
- Mover la carpeta dentro de `/usr/local/` mediante el comando:
`mv jdk1.4.2/ /usr/local/jdk`
- Crear las siguientes variables de entorno:
`export JAVA_HOME=/usr/local/jdk`
`export JRE_HOME=${JAVA_HOME}/jre`
`export PATH=$PATH:${JAVA_HOME}/bin`