



**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 UN GENERADOR DE SITIOS WEB PARA LA VISUALIZACIÓN DE ONTOLOGÍAS**

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

**Tutora: Haydemar Núñez**

**Tutora: Esmeralda Ramos**

**Caracas, Octubre de 2008**

## ACTA

Quienes suscriben, miembros del jurado designado por el Consejo de la Escuela de Computación, para dictaminar sobre el Trabajo Especial de Grado titulado: “Desarrollo de un Generador de Sitios Web para la Visualización de Ontologías” y presentado por el bachiller Eduardo Camacho, de cédula de identidad V – 16.357.202, para optar al título de Licenciado en Computación, dejan constancia de lo estipulado a continuación:

Leído como fue, dicho trabajo por cada uno de los miembros del jurado, se fijó el día 28 de Octubre de 2008 a las 8:00 a.m., para que su autor lo defendiera en forma pública, lo que hizo en la Sala de Seminarios del Centro ISYS de la Escuela de Computación, mediante una presentación oral del contenido del Trabajo Especial de Grado, luego de lo cual respondió a las preguntas formuladas. Finalizada la defensa pública del Trabajo Especial de Grado, el jurado decidió APROBARLO.

En fé de lo cual se levanta la presente Acta, en la Ciudad Universitaria de Caracas a los veintiocho días del mes de octubre del año dos mil ocho dejándose también constancia de que actuó como Coordinadora del Jurado la Profesora Haydemar Núñez.

### Jurado Principal

---

**Profesora Nora Montaña**

---

**Profesor Iván Flores**

---

**Profesora Esmeralda Ramos**  
**(Tutora)**

---

**Profesora Haydemar Núñez**  
**(Tutora)**

*A mis padres, a mis hermanos y a mis amigos.*

## **AGRADECIMIENTOS**

*A Dios, por brindarme la vida, salud, seguridad, destrezas y fuerzas para enfrentar adversidades.*

*A mis padres, Miriam y Jesús por brindarme su apoyo incondicional en todo momento y por su enseñanza que hoy constituye la base de mi ser.*

*A mis tutoras Haydemar y Esmeralda, por brindarme esta oportunidad, su tiempo, disposición y apoyo, por su paciencia y por la confianza depositada en mí.*

*A la Universidad Central de Venezuela por haberme formado a lo largo de mi carrera.*

*A todas aquellas personas que de alguna forma contribuyeron a mi formación.*

## RESUMEN

En el presente trabajo de investigación se propone el diseño, desarrollo e implementación de una herramienta que permita la generación de Sitios Web para la visualización de ontologías y contenidos relacionados a la misma. Dicha herramienta estará basada en una arquitectura cliente/servidor.

La herramienta propuesta tiene como base o punto de partida los siguientes trabajos de investigación: “Ontología en ambiente Web para el dominio del Análisis de Líquido Seminal Humano” (Pereira, 2006), “Visualizador de una Ontología de Insectos Acuáticos” (Labrador, 2007) y “Desarrollo de una Aplicación en Ambiente Web para la visualización de una Ontología para el dominio del Análisis de Semen Humano” (Reyes, 2008); los trabajos mencionados anteriormente, permiten visualizar una ontología bajo un esquema cliente/servidor. Sin embargo, estos visualizadores tienen la limitante de trabajar con un dominio y ontología específica. El generador de Sitios Web para la visualización de ontologías permite solucionar dicha limitante a través de un módulo visualizador generalizado para ontologías OWL-DL.

## Palabras Claves

Ontología, Sitios Web, visualización de ontologías, OWL-DL, visualizador generalizado.

## ÍNDICE GENERAL

<b>INTRODUCCIÓN .....</b>	<b>1</b>
<b>CAPÍTULO 1. MARCO TEÓRICO .....</b>	<b>3</b>
1.1. ASPECTOS BÁSICOS DE LAS ONTOLOGÍAS .....	3
1.1.1. <i>Definición de Ontología</i> .....	3
1.1.2. <i>Componentes de una Ontología</i> .....	3
1.1.3. <i>Características de una Ontología</i> .....	4
1.1.4. <i>Tipos de Ontologías</i> .....	5
1.1.5. <i>Visualización de Ontologías</i> .....	6
1.2. LENGUAJES DE DESARROLLO ONTOLÓGICOS.....	6
1.2.1. <i>Lenguaje de Ontologías Web (OWL)</i> .....	7
1.2.1.1. <i>Definición</i> .....	7
1.2.1.2. <i>Origen</i> .....	8
1.2.1.3. <i>Objetivos</i> .....	8
1.2.1.4. <i>Sub-lenguajes de OWL</i> .....	9
1.2.1.5. <i>Descripción del lenguaje incremental de OWL DL</i> .....	10
1.2.1.6. <i>OWL y otros lenguajes de ontología</i> .....	11
1.3. HERRAMIENTAS PARA CREACIÓN, EDICIÓN Y VISUALIZACIÓN DE ONTOLOGÍAS.....	13
1.3.1. <i>Altova SemanticWork</i> .....	14
1.3.2. <i>OntoStudio</i> .....	15
1.3.3. <i>Protégé</i> .....	18
1.3.4. <i>SWeDE</i> .....	20
1.4. GENERADORES DE CONTENIDOS WEB.....	21
1.4.1. <i>Aplicaciones Web</i> .....	22
1.4.2. <i>Generador de Contenidos Web</i> .....	23
<b>CAPÍTULO 2. MARCO APLICATIVO.....</b>	<b>26</b>
2.1. PLANTEAMIENTO DEL PROBLEMA .....	26
2.2. SOLUCIÓN PROPUESTA .....	28
2.3. OBJETIVOS.....	28
2.3.1. <i>Objetivo General</i> .....	28
2.3.2. <i>Objetivos Específicos</i> .....	28
2.4. BASES PARA DETERMINAR LA ESTRUCTURA DE LOS SITIOS WEB A GENERAR .....	29
2.5. DESARROLLO DE LA APLICACIÓN.....	33
2.5.1. <i>Identificación y Análisis de los Requerimientos Funcionales</i> .....	34
2.5.1.1. <i>Diagrama de casos de uso</i> .....	34

2.5.2. Aspectos Técnicos.....	49
2.5.3. Arquitectura de la Aplicación .....	51
2.5.4. Diseño del Modelo de Navegación entre Páginas .....	52
2.5.5. Definición de la Base de Datos.....	52
2.5.6. Estructura Lógica de Desarrollo de la Aplicación.....	57
2.5.7. Diseño de la Interfaz de Usuario.....	59
2.5.8. Implementación de las Funcionalidades.....	65
<b>CONCLUSIONES Y RECOMENDACIONES .....</b>	<b>76</b>
<b>REFERENCIAS.....</b>	<b>78</b>
<b>ANEXO 1: CASOS DE USO DE NIVEL 2.....</b>	<b>84</b>
<b>ANEXO 2: CASOS DE USO DE NIVEL 3.....</b>	<b>87</b>
<b>ANEXO 3: DIAGRAMA DE ROBUSTEZ. ....</b>	<b>110</b>

## ÍNDICE DE FIGURAS

<b>Figura 1.</b> Altova SemanticWork ( <a href="http://www.altova.com/">http://www.altova.com/</a> ).....	15
<b>Figura 2.</b> OntoEstudio ( <a href="http://www.ontoprise.de/">http://www.ontoprise.de/</a> ).....	17
<b>Figura 3.</b> OntoStudio ( <a href="http://www.ontoprise.de/">http://www.ontoprise.de/</a> ).....	17
<b>Figura 4.</b> Documentación de Ontología – Protégé ( <a href="http://protege.stanford.edu/">http://protege.stanford.edu/</a> ).....	19
<b>Figura 5.</b> Protégé ( <a href="http://protege.stanford.edu/">http://protege.stanford.edu/</a> ).....	20
<b>Figura 6.</b> SweDE ( <a href="http://owl-eclipse.projects.semwebcentral.org/">http://owl-eclipse.projects.semwebcentral.org/</a> ).....	21
<b>Figura 7.</b> Sitio Web para la visualización de Ontologías de líquido seminal humano. ...	29
<b>Figura 8.</b> Sitio Web para la visualización de una Ontología de Insectos Acuáticos. ....	30
<b>Figura 9.</b> Plantilla de página del Sitio Web a ser generado. ....	31
<b>Figura 10.</b> Diagrama de casos de uso general – Sitio Web generado.....	32
<b>Figura 11.</b> Diagrama de CU – nivel 0. ....	35
<b>Figura 12.</b> Diagrama de CU – nivel 1 – sub-módulo iniciar sesión. ....	39
<b>Figura 13.</b> Diagrama de CU – nivel 1 – sub-módulo modificar datos personales. ....	42
<b>Figura 14.</b> Diagrama de CU – nivel 1 – sub-módulo administrar sitio Web. ....	45
<b>Figura 15.</b> Arquitectura de la herramienta. ....	51
<b>Figura 16.</b> E/R entidad administrador. ....	52
<b>Figura 17.</b> E/R entidad sitio Web. ....	53
<b>Figura 18.</b> E/R entidad página principal.....	53
<b>Figura 19.</b> E/R entidad página de documento. ....	54
<b>Figura 20.</b> E/R entidad página de enlace. ....	54
<b>Figura 21.</b> E/R entidad imagen. ....	54
<b>Figura 22.</b> E/R relación administrador_sitioweb. ....	55
<b>Figura 23.</b> E/R relación sitioweb_página-principal.....	55
<b>Figura 24.</b> E/R relación sitioweb_página-documento. ....	56
<b>Figura 25.</b> E/R relación sitioweb_página-enlace.....	56
<b>Figura 26.</b> E/R relación sitioweb_página-imagenes.....	56
<b>Figura 27.</b> Diagrama entidad – relación.....	57
<b>Figura 28.</b> Plantilla general de interfaz de usuario.....	63
<b>Figura 29.</b> Cuadro de dialogo de confirmación para eliminar un Sitio Web. ....	64
<b>Figura 30.</b> Página de inicio de sesión de la herramienta .....	65
<b>Figura 31.</b> Página de registro de nuevo usuario de la herramienta. ....	66
<b>Figura 32.</b> Mensaje enviado al usuario con una nueva contraseña.....	67
<b>Figura 33.</b> Opciones para administrar un Sitio Web. ....	68
<b>Figura 34.</b> Notificación exitosa de actualización de datos personales.....	69
<b>Figura 35.</b> Página para modificar página principal de Sitio Web. ....	71
<b>Figura 36.</b> Página administrar ontología.....	72
<b>Figura 37.</b> Generar Sitio Web.....	73
<b>Figura 38.</b> Diagrama de CU – nivel 2 – sub-módulo modificar Sitio Web. ....	84

<b>Figura 39.</b> Diagrama de CU – nivel 3 – sub-módulo administrar página principal. ....	87
<b>Figura 40.</b> Diagrama de CU – nivel 3 – sub-módulo administrar página de documentos. .....	91
<b>Figura 41.</b> Diagrama de CU – nivel 3 – sub-módulo administrar enlaces.....	95
<b>Figura 42.</b> Diagrama de CU – nivel 3 – sub-módulo administrar pagina de imágenes. .	99
<b>Figura 43.</b> Diagrama de CU – nivel 3 – sub-módulo administrar ontología. ....	103
<b>Figura 44.</b> Diagrama de CU – nivel 3 – sub-módulo publicar sitio Web.....	108

## INTRODUCCIÓN

El uso de las ontologías es común en diversas áreas de la sociedad, como por ejemplo: negocios, finanzas, internet, medicina, industrias, entre otras. En estos dominios, las ontologías son usadas para formalizar capitales intelectuales, integrar fuentes de datos, representar y organizar conocimiento, entre otras. Por lo tanto, las mismas son una alternativa de representación formal del conocimiento (Núñez y Ramos, 2007).

Gracias a las características con que cuentan las ontologías, en los últimos años han cobrado gran importancia en el ámbito de la informática y en la representación del conocimiento en general, así como también el intercambio de dicho conocimiento. En la actualidad, las ontologías tienen un papel protagónico en el concepto enmarcado dentro de la Web Semántica, cuya finalidad es dotar de significado al conocimiento distribuido a lo largo de World Wide Web y proveer un mecanismo de intercambio universal para el mismo.

La definición de una ontología puede estar representada por una gran cantidad de líneas de texto plano que dificulten su comprensión, por lo cual, es de gran importancia aplicaciones que permitan la visualización de dicha información.

El presente trabajo de investigación tiene como objetivo diseñar una herramienta que permita a los usuarios crear, de una manera sencilla, una aplicación Web para la fácil visualización de ontologías, sin importar el dominio del conocimiento representado en ella.

Este trabajo de investigación se encuentra estructurado en dos capítulos, el primero representan el marco teórico en el cual está sustentado el diseño de la herramienta. El último capítulo presenta todo lo concerniente al levantamiento de la información para el diseño e implementación de la herramienta. A continuación se presenta una breve descripción de los capítulos que conforman este trabajo.

El capítulo 1, presenta una breve introducción a las ontologías, abarcando los aspectos básicos de la misma, se estudia los lenguajes ontológicos más utilizados en la

actualidad, definiendo sus características y sintaxis, siendo esta última de gran importancia para el diseño e implementación futura de la herramienta antes mencionada. En este capítulo se describen algunas de las herramientas (propietarias o de código abierto) utilizadas para la creación, edición y visualización de ontología, así como también, las aplicaciones Web y los generadores de contenido Web.

El capítulo 2, presenta la metodología utilizada para diseñar la herramienta que permite generar páginas Web para la visualización de una ontología y publicación de toda la información relacionada con la misma. Este capítulo contempla el levantamiento de la información necesaria para el diseño e implementación de la herramienta, se describen los requerimientos funcionales que satisface la misma, se realiza una descripción de los casos de uso así como sus correspondientes diagramas, se describe la arquitectura de la herramienta como también el modelo relacional de la base de datos que esta última utiliza. De igual forma, se detallan los aspectos técnicos relacionados con la implantación de la herramienta.

Finalmente, 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.

## **CAPÍTULO 1. MARCO TEÓRICO**

### **1.1. Aspectos Básicos de las Ontologías**

#### **1.1.1. Definición de Ontología**

La palabra ontología está 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.

En la década de los noventa, las ontologías cobran gran importancia en el área de la inteligencia Artificial, así como también, en la representación del conocimiento y la lingüística computacional. Existen diversas definiciones del término Ontología en el contexto de la computación, una de las más utilizadas es la combinación de las definiciones planteadas por Gruber (1993) y Borst (1997): “*una especificación formal explícita de una conceptualización compartida*”, donde la palabra *conceptualización* hace referencia a un modelo abstracto de un fenómeno en el cual se identifican conceptos relevantes enmarcados en un dominio particular: entidades, atributos, procesos y las relaciones entre éstos. *Formal* implica una organización teórica de términos y relaciones usados como herramienta para el análisis de los conceptos dentro del dominio. Por último, el término *compartida* se refiere al conocimiento consensual que es aceptado por una comunidad considerable.

Se podría decir que una ontología (en computación) es un sistema de representación del conocimiento que resulta de seleccionar un dominio o ámbito del conocimiento, y aplicar sobre el mismo una metodología, con el fin de obtener una representación formal de los conceptos que contiene y las relaciones que existen entre dichos conceptos.

#### **1.1.2. Componentes de una Ontología**

Los componentes de una ontología pueden variar según el dominio de conocimiento que abarque, sin embargo, a través del tiempo se ha logrado identificar cinco componentes fundamentales en cualquier ontología:

- **Clases:** son las ideas básicas que se intentan formalizar. Las clases son la base de la descripción del conocimiento en las ontologías ya que describen los conceptos del dominio. Una clase puede ser dividida en subclases, las cuales representarán conceptos más específicos que la clase a la que pertenecen. Las clases en una ontología se suelen organizar en taxonomías a las que se les pueden aplicar los mecanismos de herencia.
- **Relaciones:** se establecen entre clases (conceptos) de una ontología para representar la interacción entre éstas.
- **Funciones:** son un tipo concreto de relación donde se identifica un elemento mediante el cálculo de una función en la cual intervienen elementos de la ontología.
- **Instancias:** se utilizan para representar objetos de un concepto.
- **Axiomas:** elementos que permiten el modelado de sentencias que son siempre ciertas dentro del dominio de la ontología. Los axiomas pueden ser estructurales y/o no estructurales. Un axioma estructural establece condiciones relacionadas con la jerarquía de la ontología, conceptos y atributos definidos. Un axioma no estructural establece relaciones entre atributos de un concepto y son específicos de un dominio.

### 1.1.3. Características de una Ontología

Según Lamarca (2007a) algunas de las características más representativas de una ontología son mencionadas a continuación:

- **Coexistencia:** el propósito de una ontología es hacer explícito algún punto de vista, por lo que, a veces, será preciso combinar dos o más ontologías en donde cada una introducirá conceptualizaciones específicas.
- **Distintos niveles de abstracción:** estos niveles de generalización o abstracción permiten establecer una topología de ontologías. La idea es caracterizar una red de ontologías usando multiplicidad y abstracción. La representación del

conocimiento en su totalidad no es una tarea sencilla, sin embargo, se puede pensar en una estrategia de construcción gradual de abajo hacia arriba con la finalidad de obtener un todo.

- **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:** se pueden establecer las relaciones entre los elementos de una o más ontologías con la finalidad de obtener conexiones, especializaciones, generalizaciones, etc.

#### 1.1.4. Tipos de Ontologías

Steve et al. (1998) distinguen tres tipos fundamentales de ontologías:

- **Ontologías de un dominio:** expresan conceptualizaciones que son específicas a un dominio o subdominio particular.
- **Ontologías genéricas:** son las ontologías de nivel más alto ya que describen conceptos generales y funcionales del conocimiento como las estructuras parte/todo, la cuantificación, los procesos o tipos de objetos. Proporcionan términos genéricos reutilizables en otros dominios.
- **Ontologías representacionales:** especifican las conceptualizaciones que subyacen a los formalismos de representación del conocimiento, por lo que también se denominan meta-ontologías. Proporcionan el vocabulario necesario para modelar otras ontologías.

A estos tipos de ontologías se les añaden las que han sido creadas para una actividad o tarea específica denominadas ontologías de tarea o específicas y abarcan un campo limitado del conocimiento.

Tomando en cuenta diversos aspectos, se ha logrado obtener a través del tiempo otros tipos de ontologías:

- Según el tipo de agente al que vayan destinadas:
  - **Ontologías lingüísticas:** se vinculan a aspectos lingüísticos: 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:
  - **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.

### 1.1.5. Visualización de Ontologías

Los criterios que se utilizan para el diseño de los mecanismos de visualización de ontologías dependerán de las características que éstas exhiban, como por ejemplo, la complejidad, el tipo y la cantidad de clases e instancias. En trabajos anteriores a este (Labrador, 2007; Reyes 2008), se realizaron visualizadores de Ontologías orientados a la Web que permiten la exploración de la información desde la jerarquía de conceptos, permitiendo así, que el usuario final logre recuperar la información contenida en la Ontología.

## 1.2. Lenguajes de Desarrollo Ontológicos.

A través de la historia de las ontologías, se han creado y estandarizado muchos lenguajes que han cumplido un rol muy importante en la evolución de la Web. Actualmente, algunos lenguajes permiten representar algunos aspectos sobre los

conceptos de un dominio y permiten mediante relaciones taxonómicas, crear una jerarquía de conceptos. Sin embargo, son necesarios lenguajes con mayor expresividad y capacidad de razonamiento para representar los conocimientos que contienen las ontologías.

Son muchos los lenguajes para ontologías existentes, se pueden destacar lenguajes ontológicos tradicionales como: *Ontolingua*, *KIF*, *OCML*, *Flogic*, *LOOM*; así como también lenguajes para la Web estandarizados por el consorcio de la W3C como; *XML*, *XOL*, *SHOE*, *DAML + OIL* e indudablemente *OWL* (Hurtado, 2006).

Es importante destacar a *OWL* (Lenguajes de Ontologías para la Web) debido a que actualmente es el lenguaje con más importancia para el desarrollo de las ontologías. *OWL* es compatible con la arquitectura de la World Wide Web en general, y especialmente con la Web Semántica (Hurtado, 2006).

Este trabajo de investigación, abarca el lenguaje antes mencionado así como *RDF*, por ser este último la base en donde se construyen las sentencias de *OWL*.

### **1.2.1. Lenguaje de Ontologías Web (OWL)**

#### **1.2.1.1. Definición**

*OWL* 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 permitiendo definir el significado de términos en vocabularios y las relaciones entre estos términos. Está diseñado para ser utilizado cuando la información contenida en los documentos necesita ser procesada por aplicaciones además de ser presentada a los seres humanos (Lamarca, 2007c).

*OWL* posee más funcionalidades para expresar el significado y semántica que *XML* y *RDF(S)*, pero *OWL* va más allá que estos lenguajes pues ofrece la posibilidad de representar contenido de la Web interpretable por aplicaciones. *OWL* extiende *RDF* para la definición de ontologías y posibilita un nivel avanzado de inferencia para la Web

semántica mediante definiciones y restricciones de clases, propiedades y relaciones. OWL cuenta con las siguientes características:

- Semántica bien definida.
- Algoritmos de razonamiento conocidos.
- Puede ser implementado a través de varias aplicaciones.
- Incluye tipos de datos primitivos de XMLS.

#### **1.2.1.2. Origen**

OWL tiene sus bases en el lenguaje DAML+OIL. El lenguaje para intercambio de ontologías OIL, permite interoperabilidad semántica entre recursos Web. DAML+OIL, fue desarrollado por un comité conjunto de Europa y Estados Unidos en el contexto del proyecto DARPA, para permitir interoperabilidad semántica en XML y comparte el mismo objetivo buscado con OIL (Duque, 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.). DAML+OIL añade a OWL la capacidad de extender a RDFS, es decir que toda construcción de OWL, esta edificada sobre RDF (Duque, 2006).

#### **1.2.1.3. Objetivos**

OWL tiene por finalidad, facilitar un modelo de marcado, construido sobre RDF y codificado en XML que permita representar ontologías a partir de un vocabulario más amplio y una sintaxis más fuerte que la que permite RDF. OWL fue diseñado con la finalidad de satisfacer los siguientes objetivos: compatible, modificable en el tiempo, interoperable, con detección de inconsistencias, balance entre expresividad y complejidad, facilidad de uso.

#### 1.2.1.4. Sub-lenguajes de OWL

OWL proporciona tres lenguajes, cada uno con mayor nivel de expresividad que el anterior, diseñados para ser usados por comunidades específicas de programadores y usuarios; estos son listados a continuación:

- **OWL Lite:** diseñado para aquellos usuarios que necesitan principalmente una clasificación jerárquica y restricciones simples. **OWL Lite** soporta restricciones cardinales aunque muy limitadas, solo permite valores cardinales de 0 ó 1.
- **OWL DL:** es mucho más expresivo que OWL Lite y está basado en lógica descriptiva. Permite a los usuarios disponer de la máxima expresividad conservando la completitud computacional (todas las conclusiones pueden ser deducidas), y la resolución (todos los cálculos se resolverán en un tiempo finito). **OWL DL** incluye todas las construcciones del lenguaje OWL, pero pueden ser usados sólo bajo ciertas restricciones (por ejemplo, una clase no puede ser una instancia de otra).
- **OWL Full:** dirigido a usuarios que quieren máxima expresividad y libertad sintáctica de RDF sin garantías computacionales. Por ejemplo, en **OWL Full** una clase puede ser considerada simultáneamente como una colección de clases individuales y como una clase individual propiamente dicha. **OWL Full** permite a una ontología aumentar el significado del vocabulario predefinido en RDF ó OWL. Es poco probable que algún software racional pueda soportar por completo el razonamiento para cada característica de **OWL Full**.

Cada uno de estos sub-lenguajes son una extensión de su predecesor más simple, respecto a lo que puede ser expresado legalmente y a la validación de sus conclusiones. El siguiente grupo de relaciones se mantienen, pero las relaciones inversas no representan una afirmación y por lo tanto son consideradas como no válidas:

- Una ontología OWL Lite válida, es una ontología OWL DL válida.
- Una ontología OWL DL válida, es una ontología OWL Full válida.

- Cualquier conclusión válida en una ontología OWL Lite, será una conclusión válida en OWL DL.
- Cualquier conclusión válida en una ontología OWL DL será una conclusión válida en OWL Full.

A continuación se presenta una descripción del sub-lenguaje OWL DL y de los elementos que lo componen ya que el presente trabajo de investigación tendrá como base el sub-lenguaje antes mencionado.

#### 1.2.1.5. Descripción del lenguaje incremental de OWL DL

OWL DL y OWL Full utilizan el mismo vocabulario aunque OWL DL está sujeto a algunas restricciones. De forma general, OWL DL requiere separación de tipos (una clase no puede ser un individuo o una propiedad, una propiedad no puede ser tampoco un individuo o una clase). Esto implica que no se pueden aplicar restricciones a elementos del lenguaje de OWL (algo que se permite en OWL Full). Además, OWL DL requiere que las propiedades sean del tipo *ObjectProperties* o del tipo *DatatypeProperties*, este último representa relaciones entre las instancias de clases y literales de RDF y tipos de datos de esquema XML, mientras que *ObjectProperties* son relaciones entre instancias pertenecientes a dos clases.

- **oneOf:** las clases se pueden describir mediante la enumeración de los individuos que la componen. Los miembros de la clase son exactamente el grupo de los individuos enumerados. Por ejemplo, la clase días de la Semana puede describirse simplemente enumerando los individuos: Lunes, Martes, Miércoles, Jueves, Viernes, Sábado, Domingo.
- **hasValue:** una propiedad puede ser necesaria para tener un determinado individuo como un valor. Por ejemplo, instancias de la clase *ciudadanosHolandeses* pueden ser caracterizadas como las personas que viven en *losPaísesBajos* como valor de su nacionalidad que es una instancia de la clase *Nacionalidades*.

- ***disjointWith***: es posible establecer que las clases sean disjuntas unas de otras. Por ejemplo, *Hombre* y *Mujer* pueden definirse como clases disjuntas. A partir de esta declaración de *disjointWith*, un razonador puede deducir una inconsistencia cuando un individuo se indica como instancia de ambas clases y de igual manera un razonador puede deducir que si A es una instancia de *Hombre*, entonces A no es una instancia de *Mujer*.
- ***unionOf*, *complementOf*, *intersectionOf***: OWL DL y OWL Full permiten combinaciones booleanas arbitrarias de clases y de restricciones. Por ejemplo, usando *unionOf*, podemos indicar que una clase contiene elementos que son *ciudadanosdeEE.UU* o *ciudadanosHolandeses*. Usando *complementOf*, podríamos indicar que los niños no son *PersonasMayores* (es decir, la clase Niños es una subclase de complemento de *PersonasMayores*).
- ***minCardinality*, *maxCardinality*, *cardinality***: mientras que en OWL Lite, las cardinalidades se ciñen a como mínimo, como máximo o exáctamente 1 ó 0, OWL Full permite realizar declaraciones de cardinalidad para números enteros no negativos arbitrarios. Por ejemplo la clase de *MatrimonioconDobleSueldoSinHijos* restringiría la cardinalidad de la propiedad *tieneSueldo* a una cardinalidad mínima de 2.
- ***complex classes***: en muchas construcciones, OWL Lite restringe la sintaxis a nombres de clase simples (por ejemplo, en declaraciones *subClassOf* o *equivalentClass*). OWL Full amplía esta restricción para permitir arbitrariamente descripciones de clase complejas, que consisten en clases enumeradas, restricciones de propiedad, y combinaciones booleanas, también permite que las clases sean utilizadas como instancias (OWL DL y OWL Lite no lo permiten).

#### 1.2.1.6. OWL y otros lenguajes de ontología

Los lenguajes anteriores a OWL han sido utilizados para desarrollar herramientas y ontologías destinadas a comunidades específicas (especialmente ciencias y

aplicaciones de comercio electrónico), no fueron definidos para ser compatibles con la arquitectura de la Web en general, y la Web Semántica en particular.

La **Tabla 1** muestra una comparación entre los lenguajes de ontologías basados en Web más utilizados. En dicha tabla, se puede observar que el lenguaje OWL presenta, en la mayoría de los casos, todas las bondades o ventajas con las que cuenta el lenguaje ontológico DAML+OIL, así como también con los beneficios de RDFS; lo que permite a OWL ser compatible con la Web semántica.

**Tabla 1.** Comparación de lenguajes ontológicos.

	XML	XMLS	RDF(S)	DAML+OIL	RDF(S) 2002	OWL
Listas limitadas				✓	✓	✓
Restricciones de cardinalidad	✓	✓		✓		✓
Expresiones de clases				✓		✓
Datatypes		✓		✓	✓	✓
Clases definidas				✓		✓
Enumeraciones	✓	✓		✓		✓
Equivalencias				✓		✓
Extensibilidad			✓	✓	✓	✓
Semántica formal				✓	✓	✓
Herencia			✓	✓	✓	✓
Inferencia				✓		✓
Restricciones locales				✓		✓
Restricciones cualificadas				✓		
Reificación			✓	✓	✓	✓

### 1.3. Herramientas para Creación, Edición y Visualización de Ontologías.

En la actualidad, las ontologías se han convertido en un medio de representación del conocimiento de gran importancia. Existen muchas aplicaciones para la construcción de las mismas, por lo general permiten además de su creación la posibilidad de visualizar la ontología en algún formato.

Sin embargo, el archivo donde es almacenada la información de la ontología no es de fácil entendimiento para los expertos, a pesar de que la misma está codificada, por lo general, en un formato estándar que facilita su comprensión como lo es XML (Lenguaje de Marcas Extensible). La definición de una ontología puede estar representada por una gran cantidad de líneas de texto plano que dificulten su comprensión, por lo cual, es de gran importancia aplicaciones que permitan la visualización de dicha información en formatos de fácil y rápido entendimiento.

Las aplicaciones o herramientas utilizadas para la creación de ontologías presentan una gran limitante para el usuario común de hoy. Dichas aplicaciones deben estar instaladas localmente en un computador, por lo tanto, las mismas suelen ser dependientes (con algunas excepciones) del Sistema Operativo. Esto representa una gran desventaja ya que las tendencias actuales son las aplicaciones multiplataformas y las que mejor se adaptan a este concepto son las aplicaciones Web, en las cuales sólo es necesario un navegador Web y una conexión a Internet.

Aplicaciones Web que permitan la visualización de ontologías, es un concepto interesante que traería ventajas a las personas que necesiten trabajar con éstas:

- Una aplicación Web para la visualización de ontologías, al estar alojada en un servidor Web, puede ser utilizada por cualquier persona que disponga de una conexión a Internet, sin importar el Sistema Operativo instalado localmente en la maquina cliente.
- Si un grupo de personas trabaja con una misma ontología, los cambios que puedan ocurrir a la misma pueden estar disponibles para todos una vez se haya

actualizado el recurso Web. En este sentido, se asemeja a los datos almacenados en una Base de Datos remota.

A continuación se describen algunas de las herramientas disponibles para la creación y edición de ontologías en los formatos más utilizados actualmente, estas herramientas también permiten la visualización de ontologías.

### **1.3.1. Altova SemanticWork**

Altova, es una empresa pionera en herramientas de desarrollo e integración de datos y es miembro activo de la World Wide Web Consortium (W3C) y Object Management Group [ALTOVA, s/n].

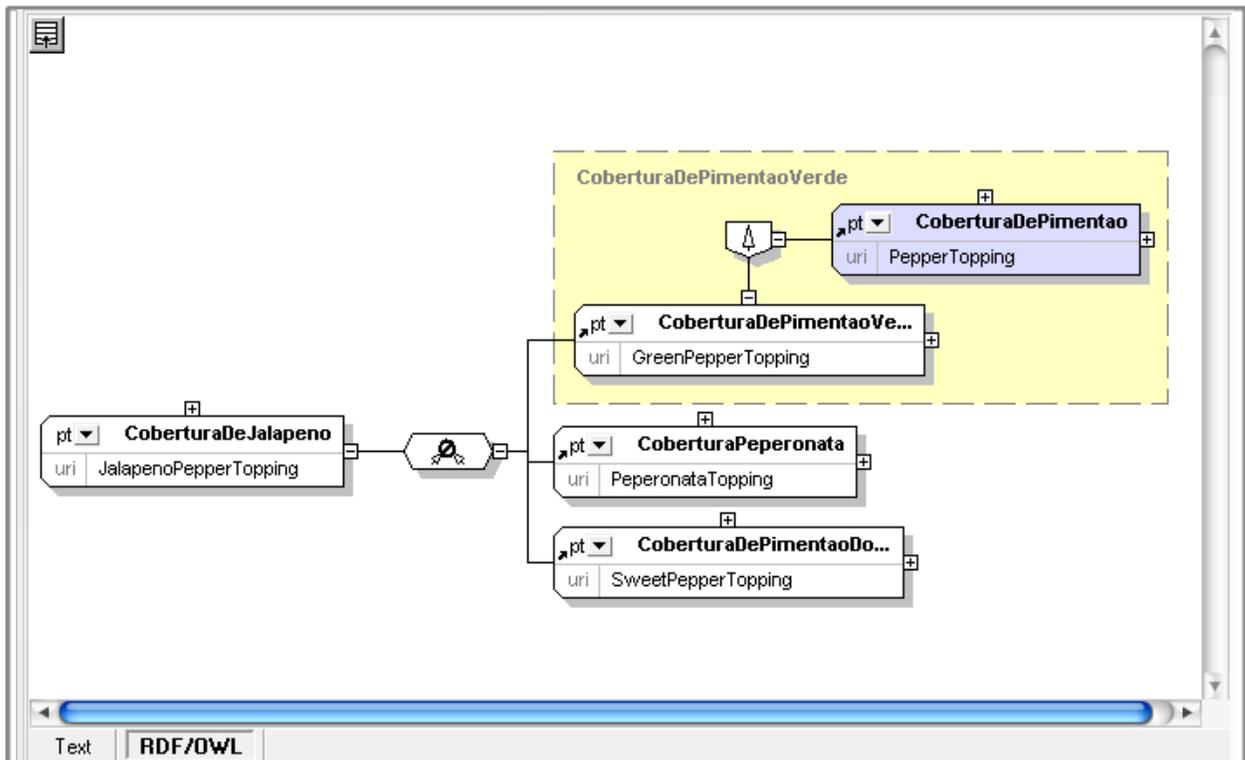
Siguiendo su tradición de proporcionar herramientas visuales de desarrollo, Altova ha creado SemanticWorks que permite aprender y trabajar con las nuevas tecnologías de la Web Semántica, en forma fácil e intuitiva. SemanticWorks separa los distintos componentes de una ontología en pestañas, como es el caso de las clases, propiedades e instancias.

Incluye un editor de RDF y OWL, con resaltado de texto, para el diseño visual de de aplicaciones basadas en Web semántica y proporciona las siguientes funcionalidades:

- Diseño de RDF(S) y comprobación de sintaxis basadas en la especificación RDF/XML.
- Diseño de OWL y soporte para sus tres sub-lenguajes, permite seleccionar el sub-lenguaje en que se desea visualizar la ontología.
- Autogeneración y edición de RDF/XML y N-triples.
- Otras.

El modulo de visualización de ontologías es lo más destacable de dicha herramienta, permite visualizar sólo los elementos seleccionados y expandirlos según sea el caso, manteniendo en un área limitada por un recuadro de color amarillo dicha expansión, el modulo de visualización permite la edición directa. SemanticWorks

permite exportar la visualización grafica a formato PNG. En la **Figura 1** se muestra la visualización de clase o concepto (Cobertura de Jalapeño) juntos con las sub-classes correspondiente de una ontología de pizza.



**Figura 1.** Altova SemanticWork (<http://www.altova.com/>).

Actualmente, esta herramienta es propietaria y sólo es posible utilizar la versión demo por un lapso no mayor a treinta días, su activación al momento de instalarla es a través de un código solicitado por medio de un formulario. Altova ya ha liberado algunos de sus otros productos como es el caso de la librería *AltovaXML* que integra *Altova XML Validator*, *Altova XSLT 1.0 / 2.0 Engine* y *Altova XQuery 1.0*.

### 1.3.2. OntoStudio

Es un entorno de desarrollo profesional para soluciones ontológicas diseñado por la compañía **Ontoprise** que cuenta con un conjunto de productos que apoyan el despliegue semántico de las tecnologías en la empresa [ONTOPRISE, s/n].

OntoStudio, a través de su diseño modular, permite agregar nuevas funcionalidades y personalizar las ya existentes con la finalidad de satisfacer las necesidades de cada usuario. Su interfaz modular se debe a que está basada en el entorno de desarrollo Eclipse. Esta herramienta soporta los lenguajes ontológicos estándares auspiciados por la W3C: OWL, RDF(S) y F-Logic.

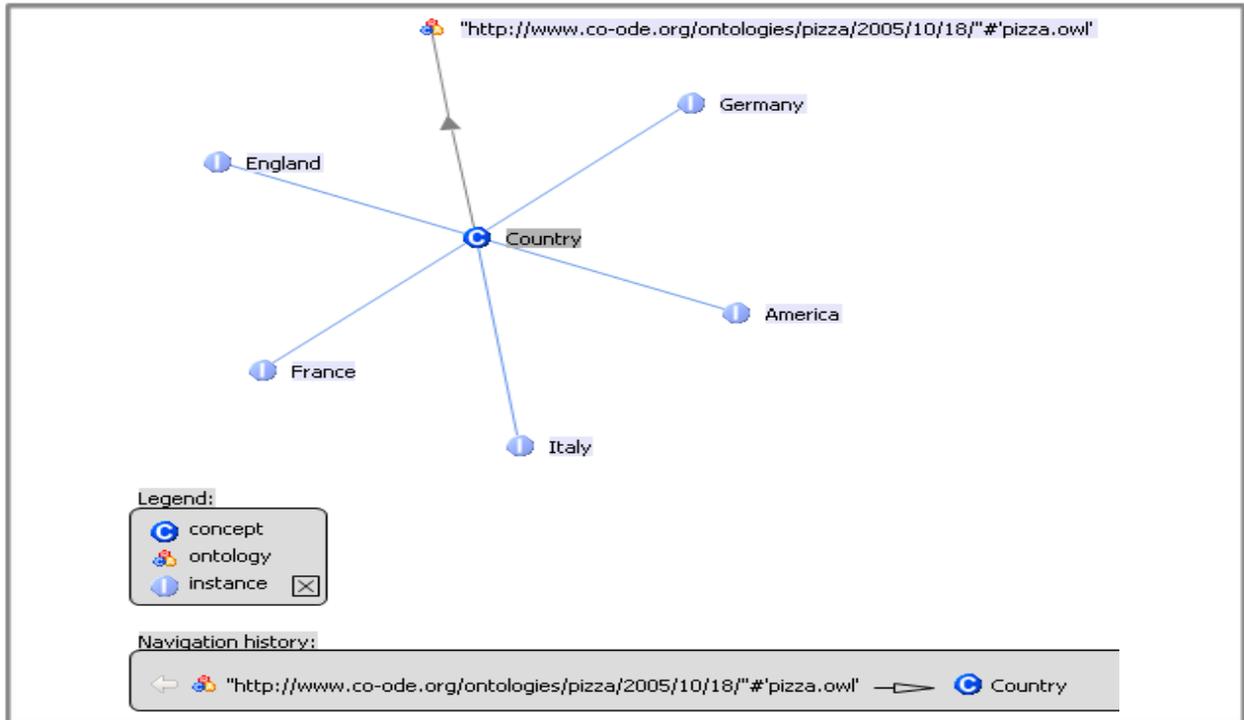
Gracias a su motor de inferencia integrado OntoBroker, OntoStudio permite el procesamiento de ontologías así como de la lógica que ellas representan. OntoBroker ha sido desarrollado para hacer frente, sin ningún inconveniente, al procesamiento de consultas rápidas y constantes. Tiene como predecesor OntoBroker OWL y KAON2. Este motor de inferencia soporta los estándares OWL, RDF(S) y F-Logic.

OntoStudio permite la integración de distintas fuentes de información como bases de datos, documentos, sistemas de archivos, aplicaciones y servicios Web. Permite, a través de reglas y filtros, resolver conflictos de nombres, de valores y entre las estructuras de las distintas fuentes. OntoStudio, por medio de una ontología, provee una vista única y en tiempo real para todas las fuentes de información.

Esta herramienta permite importar elementos de Outlook directamente a una ontología como es el caso de e-mails, contactos, tareas, entre otros. Además permite importar ontologías en formatos OWL, RDF(S), NT, N3, esquemas de bases de datos relacionales para representación de ontologías sencillas, entre otros.

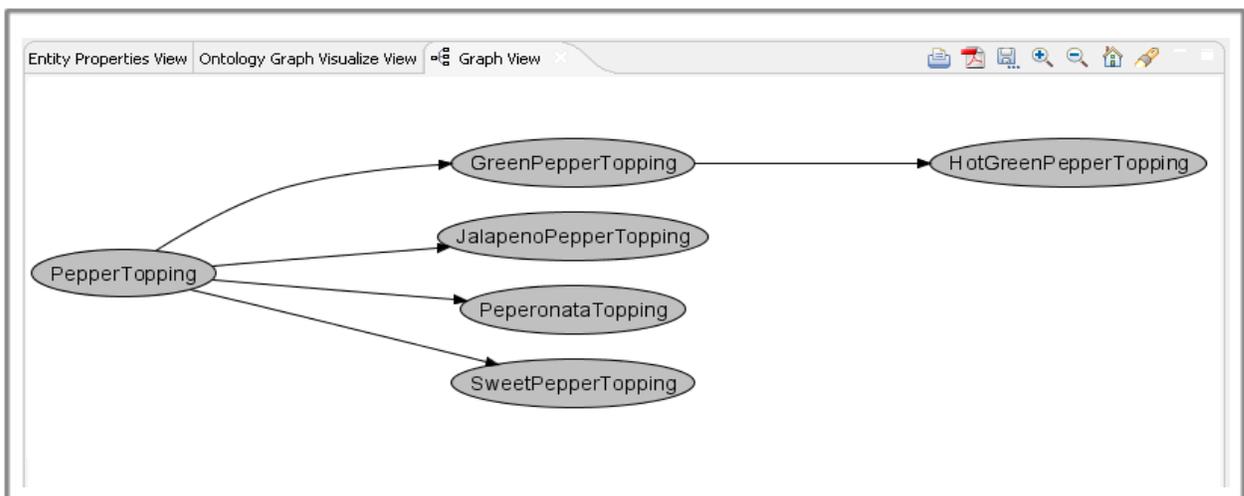
Las ontologías pueden ser exportadas a los formatos antes mencionados, en los cuales puede haber pérdida de información. Por ejemplo, en el caso de exportar la ontología a RDF, las reglas establecidas en la ontología se perderían. También permite exportar las ontologías a modelos UML 2.0 (Lenguaje Unificado de Modelado).

La estructura de la ontología es mostrada en el modulo de navegación de ontologías, y de este se puede acceder a los valores de cada uno de los elementos presentes en la misma, lo que facilita la edición. Existen dos módulos de visualización; el primero, representa el elemento seleccionado a través de nodos e incluye los sub-elementos que contenga. Ambos permiten la búsqueda de palabras claves tomando en cuenta los elementos visualizado. En la **Figura 2** se muestra dicho módulo.



**Figura 2.** OntoEstudio (<http://www.ontoprise.de/>).

El segundo módulo de visualización, permite crear grafos dirigidos a partir del elemento de la ontología seleccionado, incluyendo sus sub-elementos. Esta modalidad de visualización permite exportar el grafo generado a formatos PDF y PNG, así como imprimirlo. En la **Figura 3** se muestra dicho módulo.



**Figura 3.** OntoStudio (<http://www.ontoprise.de/>).

La herramienta OntoStudio a pesar de estar basada en Eclipse no es de libre distribución, ya que es una adaptación a través de plug-in propietarios de la compañía **Ontoprise**. OntoStudio cuenta con una licencia de prueba durante 3 meses para uso no comercial.

### 1.3.3. 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 basadas en conocimiento que representen la información de forma ontológica [PROTÉGÉ, s/n].

Protégé es libre y de código abierto, 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 y es extensible, proporcionando un ambiente de *plug-in-play* el cual lo hace un editor flexible para el desarrollo rápido.

Protégé provee funcionalidades que lo catalogan como el editor para creación y edición de ontologías más completo. Unas de las funcionalidades de mayor utilidad es la de poder generar documentación OWL al estilo JavaDoc, en la cual es posible escoger una serie de opciones, entre ellas: el lenguaje de la documentación. La documentación es creada en formato HTML lo que permite su intercambio sin inconvenientes. En la **Figura 4** se muestra parte de la documentación generada a partir de una ontología de pizza.

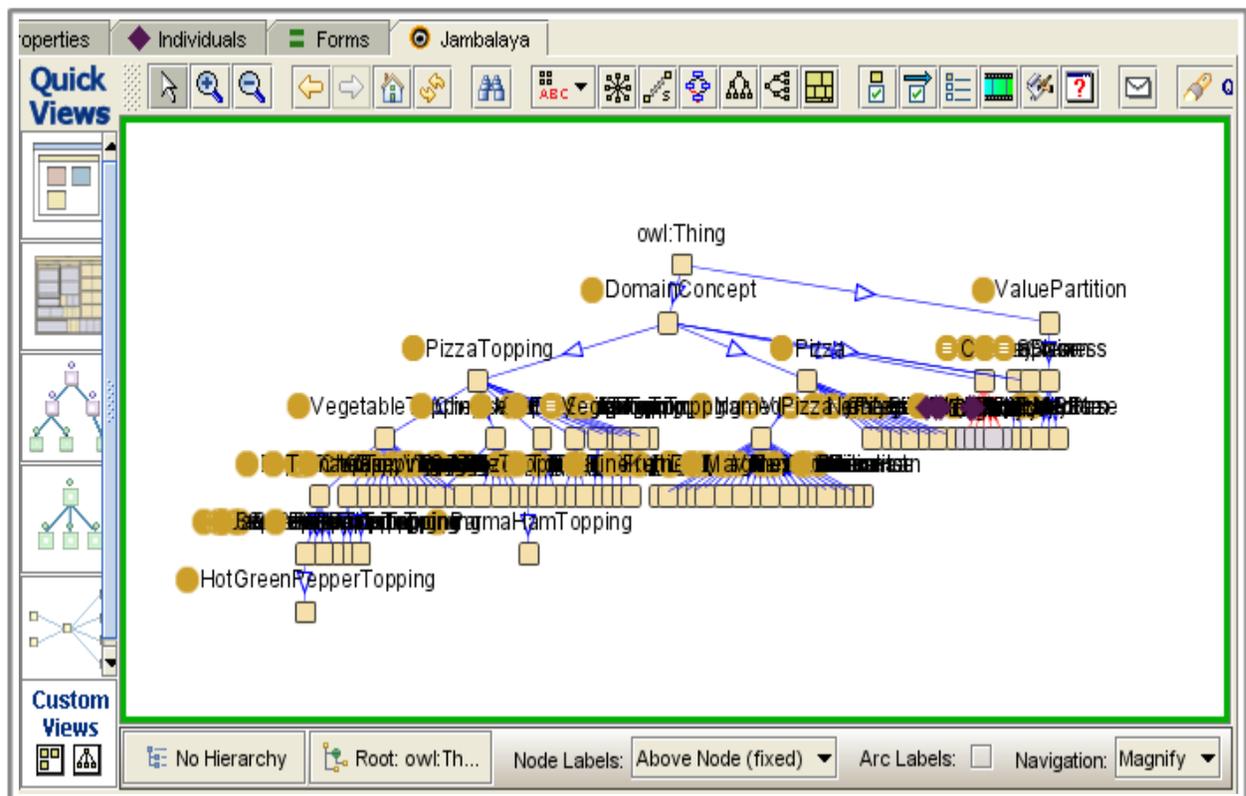


**Figura 4.** Documentación de Ontología – Protégé (<http://protege.stanford.edu/>).

Otra de las funcionalidades provista por Protégé es generar código JAVA para permitir el rápido desarrollo de aplicaciones basadas en dicho lenguaje para la manipulación automatizada de la ontología. También ofrece un módulo para chequear la consistencia de la ontología y permite determinar el sub-lenguaje OWL en la que está codificada la misma. A pesar de contar con un editor de texto, el mismo es sencillo y muestra la codificación de la ontología en formato RDF(S) sin ningún tipo de resaltado visual, dificultando la tarea de edición a través del mismo.

La herramienta Protégé tiene incorporado un visualizador conocido como Jambalaya que es un plug-in desarrollado por *Computer Human Interaction & Software Engineering Lab* (CHISEL) de la Universidad de Victoria en Canadá. Utiliza *Piccolo zooming library* desarrollado por *the Human-Computer Interaction Lab* (HCIL) en la Universidad de Maryland.

Jambalaya ofrece un editor de ontologías visual de fácil uso para la tarea de edición. Permite la visualización de ontologías a través de distintas vistas que organizan la misma bajo diferentes esquemas de representación los cuales pueden ser configurados agregando o quitando los elementos deseados en el área de visualización. Dicha configuración puede ser exportada y posteriormente importada. En la **Figura 5** se muestra la estructura general de una ontología de pizza [CO-ODE, s/n].



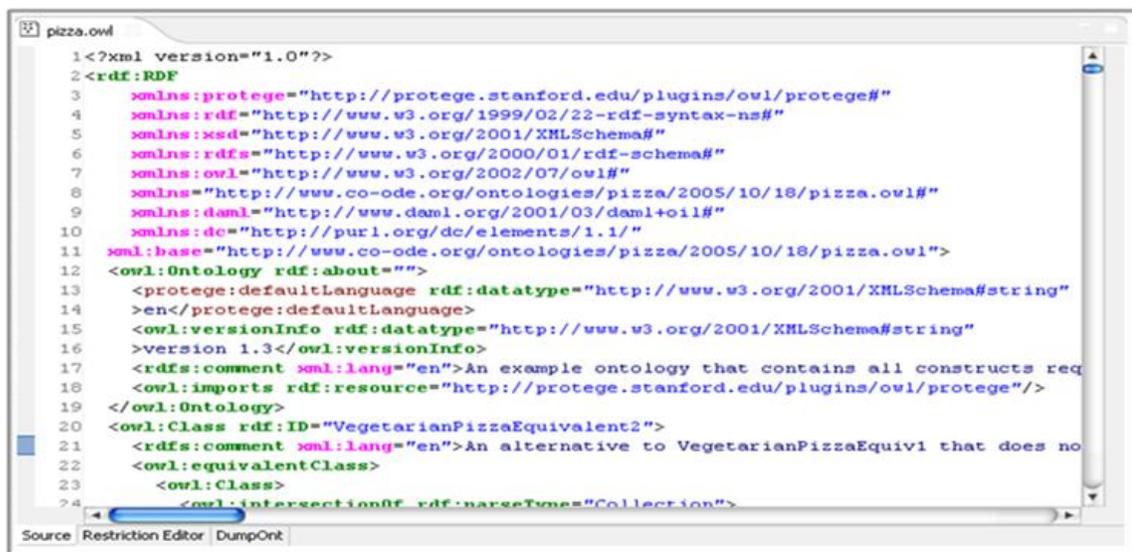
**Figura 5.** Protégé (<http://protege.stanford.edu/>).

#### 1.3.4. SWeDE

Es un marco de trabajo extensible para la integración de nuevas herramientas con las ya existentes para la Web Semántica. Es un plug-in para el entorno de desarrollo integrado Eclipse, incluye un editor de OWL con útiles funciones, como resaltado de sintaxis, órdenes, y detección de errores. Integra instrumentos existentes, como el OWL Validator, Kazuki (generador de código OWL para Java), y DumpOnt para la visualización de ontologías [BBN TECHNOLOGIES].

SweDE presenta una interfaz común para los desarrolladores acostumbrados a Eclipse. El panel de edición de ontologías cuenta con tres sub-paneles que permiten ver el código del archivo (.owl) con sintaxis resaltada a través de distintos colores, un editor de restricciones y la visualización de la ontología utilizando DumpOnt que muestra la jerarquía de clases y propiedades.

En adelante, SWeDE integrará más herramientas de edición y visualización de documentos de la Web Semántica, y las normas, así como puntos de extensión para proporcionar herramientas de traducción, editores visuales ligeros, y modelos RDF. En la **Figura 6** se muestra la representación en texto plano de una ontología de pizza codificada en OWL.



```

1<?xml version="1.0"?>
2<rdf:RDF
3  >xmlns:protege="http://protege.stanford.edu/plugins/owl/protege#"
4  >xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
5  >xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
6  >xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
7  >xmlns:owl="http://www.w3.org/2002/07/owl#"
8  >xmlns="http://www.co-ode.org/ontologies/pizza/2005/10/18/pizza.owl#"
9  >xmlns:daml="http://www.daml.org/2001/03/daml+oil#"
10 >xmlns:dc="http://purl.org/dc/elements/1.1/"
11 >owl:base="http://www.co-ode.org/ontologies/pizza/2005/10/18/pizza.owl">
12 <owl:Ontology rdf:about="">
13   <protege:defaultLanguage rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
14   >en</protege:defaultLanguage>
15   <owl:versionInfo rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
16   >version 1.3</owl:versionInfo>
17   <rdfs:comment xml:lang="en">An example ontology that contains all constructs req
18   <owl:imports rdf:resource="http://protege.stanford.edu/plugins/owl/protege"/>
19 </owl:Ontology>
20 <owl:Class rdf:ID="VegetarianPizzaEquivalent2">
21   <rdfs:comment xml:lang="en">An alternative to VegetarianPizzaEquiv1 that does no
22   <owl:equivalentClass>
23   <owl:Class>
24   <owl:intersectionOf rdf:resourceType="Collection">

```

**Figura 6.** SweDE (<http://owl-eclipse.projects.semwebcentral.org/>).

#### 1.4. Generadores de Contenidos Web.

Con la introducción del Internet y del Web en concreto, se han abierto infinidad de posibilidades en cuanto al acceso y uso de información desde cualquier parte del mundo. El avance de las nuevas tecnologías, proporciona un innegable crecimiento en la presencia de individuos u organizaciones en Internet.

En la actualidad, generar contenido para la Web, sin ningún conocimiento informático, se ha convertido en algo imprescindible para el intercambio de información y presencia en Internet. Existen diversas aplicaciones que permiten generar contenidos compatibles con la Web; algunas de ellas, presentadas como aplicaciones de escritorios y otros bajo una filosofía Web.

#### 1.4.1. Aplicaciones Web

También conocidas como aplicaciones basadas en navegadores, son programas que los usuarios utilizan a través de un navegador Web y una conexión a Internet o a una Intranet. La aplicación en sí, se encuentra alojada o instalada en una máquina que toma el rol de Servidor Web y permite atender las solicitudes de los distintos clientes o usuarios, esto es conocido como arquitectura cliente/servidor. La ubicación geográfica entre el servidor y la máquina cliente no es relevante.

Las aplicaciones Web ofrecen grandes ventajas que pueden ser aprovechadas por muchas organizaciones o individuos, sobre todo ahora que la globalización es una realidad. Entre las ventajas que se pueden mencionar están:

- **No requieren instalación:** por lo general, solo con el navegador Web será suficiente para acceder a la aplicación.
- **Son fáciles de usar:** el conocimiento para la utilización de las aplicaciones Web suele ser mínimo ya que las mismas tienden a hacer sencillas.
- **Ligeras:** los requisitos de Hardware/Software no suelen afectar el rendimiento de las aplicaciones, con tener un navegador Web y una conexión a Internet es suficiente.
- **Multiplataformas:** la ejecución de aplicaciones Web depende del servidor y no directamente del Sistema Operativo instalado en la máquina donde resida el mismo.

- **Costos de desarrollo reducidos:** el desarrollo de una aplicación Web por lo general suele ser rápida en comparación a las aplicaciones de escritorio. No se desarrollan versiones de la aplicación para cada Sistema Operativo.

#### 1.4.2. Generador de Contenidos Web

Son aplicaciones que permiten generar Sitios Web en un determinado dominio, aunque existen algunos que no abarcan ninguno en específico. El elemento diferenciador de estas aplicaciones respecto a otras similares, es que están orientada a usuarios no expertos en informática, que tan solo posean destrezas básicas de navegación en Internet y de herramientas ofimáticas, y por lo tanto, puedan realizar el proceso de creación, modificación o actualización del Sitio Web, con una participación protagónica en ello.

Suelen ser aplicaciones de fácil uso e intuitivas, y permiten brindar apoyo a las instituciones, organismo e individuos para su presencia en la Web. La estructura de estas aplicaciones suele estar dividida en módulos, los más comunes son:

- **Módulo de inicio de sesión:** verifica a través de un nombre de usuario y su respectiva contraseña el acceso al módulo de administración.
- **Módulo de contactos:** para registrar y mostrar personas de contacto de la organización con sus datos más relevantes.
- **Módulo de noticias:** para mostrar noticias importantes relacionadas con el ámbito de la organización, suelen estar listadas por orden cronológico.
- **Módulo de eventos:** muestra los eventos importantes relacionados con la organización, tienen un periodo de tiempo finito y suelen eliminarse una vez caducado el mismo.
- **Módulo de documentos:** listados de documentos relevantes para la organización, disponibles para su descarga.

- **Módulo de menús:** listado de opciones disponible en la página Web a generar, con estilos que los diferencian entre sus diferentes posibles estados.
- **Módulo de administración:** permite la creación, edición o eliminación del contenido Web reflejado en las distintas páginas que conforman el sitio Web. De ser el caso, permitirá la administración de usuarios con ciertas permisologías dentro de la aplicación a generar.
- **Modulo upload:** permite colocar ficheros en el servidor donde estará alojada la aplicación.

El Centro de Ingeniería de Software y Sistemas de la escuela de Computación de la Universidad Central de Venezuela, en el marco del Proyecto “*Acercamiento de la Tecnología Informática al Ciudadano*”; ha desarrollado aplicaciones que facilitan el acceso y uso de las tecnologías informáticas por parte de diversas comunidades y organizaciones, que contribuyen en el proceso de democratización de la información a través de las Tecnologías de Información y Comunicación (TIC). Dichas aplicaciones están desarrolladas siguiendo una arquitectura cliente/servidor bajo un dominio específico, algunas de estas son nombradas a continuación:

- **e-Alcaldía:** Generador de Alcaldías Virtuales, es un proyecto que surge con la finalidad de dotar a las alcaldías, particularmente a aquellas que no cuentan con centros informáticos sofisticados, de una herramienta que les permita crear y administrar fácilmente sitios Web, a efectos de aprovechar estas tecnologías en la comunicación con los ciudadanos. La administración del sitio Web, generado a través de dicha herramienta, será realizada por empleados de la alcaldía con un mínimo conocimiento en aplicaciones ofimáticas y navegación en Internet, esto representa una de las ventajas que aporta dicha aplicación (Fossi y Oliveira, 2006).
- **GenAsig:** Generador de Sitios Web de Asignaturas, es una aplicación que permite a docentes no expertos en el diseño Web construir sitios Web de asignaturas de manera rápida y sencilla, con la finalidad de mantener una comunicación entre el grupo docente de una asignatura y sus estudiantes, el sitio

Web resultante ofrecerá información concerniente a la asignatura como es el caso de cronograma de actividades, cartelera informativa, sección de descarga, enlaces de interés y noticias. GenAsig en su primera versión, toma como base la Facultad de Ciencias de la Universidad Central de Venezuela, pero podrá ser extendido en versiones posteriores para abarcar cualquier asignatura de cualquier universidad (Pedraza, 2003).

- **Genci:** Generador de Sitios Web de Centros de Investigación, es una herramienta que facilita la creación, publicación y mantenimiento de Sitios Web de Centros de Investigación de manera rápida y sencilla. Está dirigida a usuarios no expertos en el área de desarrollo de páginas Web. Con GENCI solo es necesario agregar contenido, seleccionar un logo y seleccionar una plantilla que proporcionara estilo al sitio, para obtener como resultado un Sitio Web de Centro de Investigación. Esta herramienta apoya la iniciativa de difusión de conocimiento científico, facilitando la promoción de la investigación en los Centros de investigación y la comunicación entre los mismos en Venezuela (Borges y Rivero, 2006).
- **G-revist@:** Generador de Revistas Electrónicas, es una herramienta portable, de fácil uso que permite generar sitios Web de revistas electrónicas y administrar todo lo concerniente a las mismas. Orientada a usuarios no expertos en el diseño de este tipo de aplicaciones, permite a los mismos la transmisión y comunicación del conocimiento de manera rápida y sencilla (Monzerratt y Rodríguez, 2006).
- **GenTur:** Generador de Sitios Web Turísticos Regionales, aplicación que permite la construcción de Sitios Web turísticos que proyectan las bellezas naturales de las regiones que conforman Venezuela, identificada por su ubicación geográfica, permite promocionar su región, su cultura, etc. Los dota de los mecanismos necesarios para el desarrollo endógeno del turismo, el fomento de una cultura turística y la participación protagónica de los actores que conforman el sistema turístico nacional [ISYS, s/n].

## **CAPÍTULO 2. MARCO APLICATIVO.**

### **2.1. Planteamiento del Problema**

En los últimos años, se ha comenzado a experimentar el uso de técnicas de representación del conocimiento que permitan el intercambio del mismo con mayor facilidad, así como también extender la web actual con metadatos, en la que los ordenadores no sólo serán capaces de presentar toda la información contenida en ella, sino que además podrán entenderla y gestionarla de forma inteligente.

Los datos por si solos, sin tener una semántica asociada, no son de utilidad ya que pueden resultar ambiguos. Por lo tanto, hay que introducir esa semántica añadiendo datos que describen a su vez los propios datos o metadatos. Sin embargo, para dar una descripción formal de contenidos, hace falta una herramienta más potente que los metadatos: las ontologías. Mientras los metadatos estructuran contenidos, las ontologías estructuran la semántica de un recurso.

La utilización de los metadatos es necesaria para poder definir en las ontologías qué partes pueden ser compartidas o reutilizadas. Sin embargo, las ontologías van más allá, pues son capaces de almacenar otro tipo de conocimiento, la semántica, a través de los conceptos y las relaciones entre ellos, permitiendo realizar una representación declarativa del conocimiento de un dominio, que puede ser comunicado entre personas y máquinas (García, 2005).

A finales de la década de los 90, surge la visión de lo que sería la evolución de la Web, conocida como la Web Semántica. Se trata de una corriente, promovida por el propio inventor de la Web y presidente del consorcio W3C Tim Berners-Lee, cuyo último fin es lograr que los computadores puedan entender, y por tanto utilizar, lo que la Web contiene. Para conseguir esta meta, la Web Semántica propone describir los recursos de la Web con representaciones procesables (es decir, entendibles) no sólo por personas, sino por programas que puedan asistir, representar, o reemplazar a las personas en tareas rutinarias o inabarcables para un humano. Las tecnologías de la Web semántica buscan desarrollar una Web más cohesionada, donde sea aún más

fácil localizar, compartir e integrar información y servicios, para sacar un partido todavía mayor de los recursos disponibles en la Web. En conclusión, la Web Semántica propone dotar de significado a la información.

Las ontologías es uno de los mecanismos o tecnologías para alcanzar el objetivo antes mencionado y permite representar el conocimiento a través de estándares bien definidos, así como también, proporcionar un lenguaje comprensible por todos los involucrados. En la Web Semántica, las ontologías representan unos de los pilares fundamentales, por lo tanto han cobrado gran importancia.

A pesar de su importancia, las ontologías son un concepto relativamente nuevo y en crecimiento, y las herramientas actuales para trabajar con las mismas no se adaptan a las necesidades de los usuarios. Las herramientas existentes, como Protégé, son ideales para la creación o edición de una ontología, pero a la hora de querer solo visualizar la misma no representa la mejor alternativa, ya que se presentan una serie de módulos o paneles con información que puede lograr a confundir a la persona que intenta visualizar la ontología, sobre todo si la misma no es experta en la herramienta. Dicha situación se presenta en las herramientas similares a Protégé.

Otra de las limitaciones presentes en la mayoría de las herramientas existentes, es que son aplicaciones de escritorio, lo que involucrara la instalación de la misma y crea una dependencia con el Sistema Operativo.

Por lo tanto, surge la necesidad de crear una aplicación que permita visualizar una ontología de manera sencilla permitiendo a las personas un fácil acceso a la representación del conocimiento. En trabajos o proyectos anteriores (Pereira, 2006; Labrador, 2007; Reyes, 2008) se han realizados visualizadores Web de ontologías, con lo cual se han conseguidos beneficios en este tópico. Estos visualizadores tienen la limitante de trabajar con un dominio y ontología específica, con lo cual se podría pensar en una aplicación que permita generar visualizadores de ontologías y así generalizar proyectos anteriores.

## **2.2. Solución Propuesta**

Teniendo en cuenta el planteamiento anterior, se propone crear una herramienta Web que permita a las personas interesadas, diseñar y elaborar una aplicación, bajo una filosofía cliente/servidor, que permita la visualización de una ontología sin limitantes geográficos ni de Hardware/Software.

La aplicación propuesta tendrá como bases o puntos de partidas los siguientes trabajos de tesis: “Ontología en ambiente Web para el dominio del Análisis de Líquido Seminal Humano” (Pereira, 2006), “Visualizador de una Ontología de Insectos Acuáticos” (Labrador, 2007) y “Desarrollo de una Aplicación en Ambiente Web para la visualización de una Ontología para el dominio del Análisis de Semen Humano” (Reyes, 2008); trabajos que permiten visualizar una ontología bajo un esquema cliente/servidor.

La herramienta propuesta (generador Web) deberá contar con los módulos más comunes en este tipo de aplicación, un módulo encargado de autenticar e identificar el usuario del sistema; el mismo deberá contar con un sub-módulo que permita el registro de nuevos usuarios.

De igual forma, la aplicación deberá contar con un módulo de administración de sitios Web, que permitirá crear, modificar y eliminar el contenido Web asociado a este último, como por ejemplo: enlaces de interés, versión de la ontología, publicación de documentos, entre otros.

## **2.3. Objetivos**

### **2.3.1. Objetivo General**

Diseñar e implementar una herramienta para la creación y publicación de una aplicación Web que permita la visualización de una ontología codificada en OWL-DL.

### **2.3.2. Objetivos Específicos**

- Analizar los antecedentes a dicho trabajo de investigación.

- Determinar la estructura de los Sitios Web a generar.
- Identificar los requerimientos funcionales del generador Web.
- Diseñar los módulos funcionales de la herramienta.
- Diseñar la interfaz de la herramienta.
- Desarrollar la herramienta.
- Estudiar la posibilidad de internacionalización del generador Web o herramienta.
- Realizar las pruebas pertinentes para verificar el correcto funcionamiento de la herramienta.

## 2.4. Bases para Determinar la Estructura de los Sitios Web a Generar

En trabajos de investigación anteriores (Labrador, 2007 y Reyes, 2008) fueron implementados visualizadores de Ontologías orientados a un dominio específico del conocimiento. En la **Figura 7** se muestra la página de inicio de la aplicación Web para “La visualización del Líquido Seminal Humano”.



**Figura 7.** Sitio Web para la visualización de Ontologías de líquido seminal humano.

El visualizador de Ontologías incluido en el Sitio Web mencionado anteriormente es una optimización de un trabajo previo orientado a la visualización de una Ontología de Insectos Acuáticos. En la **Figura 8** se muestra la página de inicio de dicha aplicación Web.

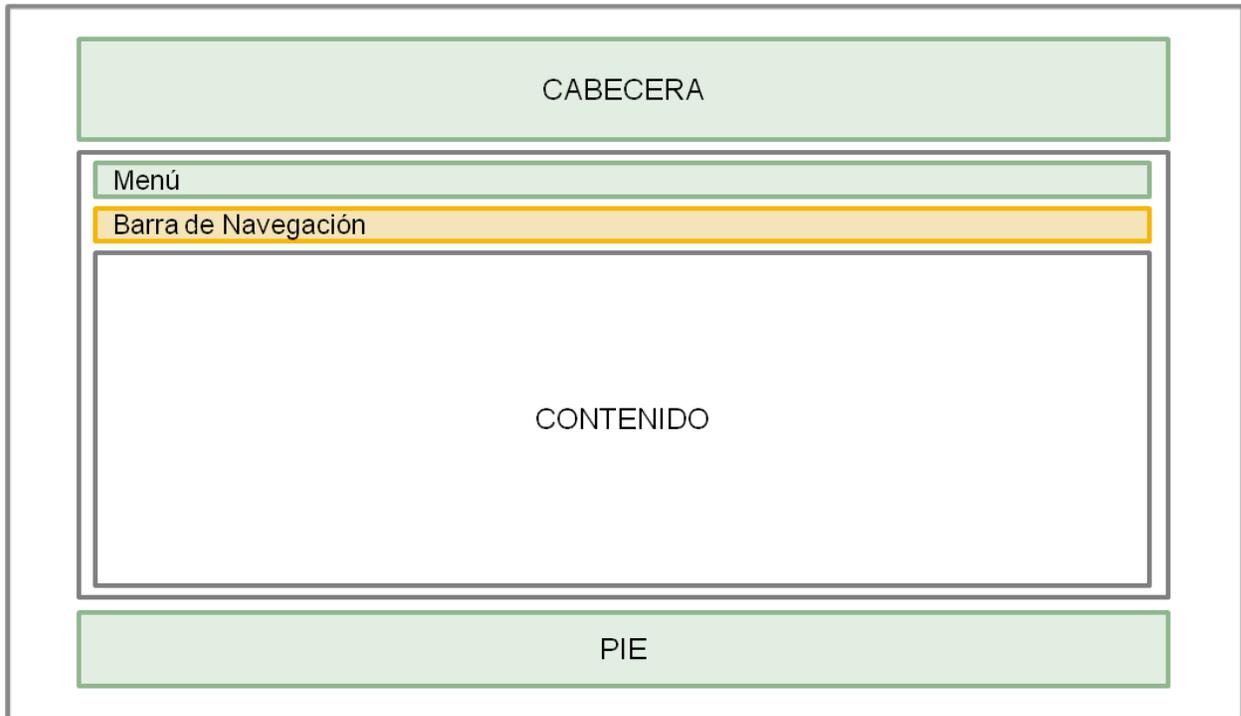


**Figura 8.** Sitio Web para la visualización de una Ontología de Insectos Acuáticos.

Los Sitios Web mencionado anteriormente presentan estructuras y funcionalidades iguales, con la salvedad de estar orientadas a un dominio del conocimiento distinto. Ambos sitios presentan un módulo administrativo que permite cambiar parte del contenido mostrado en el mismo. De los Sitios Web mencionados, se toma en cuenta los siguientes apartados:

- **La estructura de las páginas:** los Sitios Web mencionados, presentan una plantilla bien definida que se mantiene en todas las páginas que conforman a los mismos. Esta plantilla, se considera como base para definir la distribución que tendrá la información en las páginas del Sitio Web a generar. En la **Figura 9** se

muestra la plantilla general en la cual estarán basadas las páginas del Sitio Web a ser generado por la herramienta.

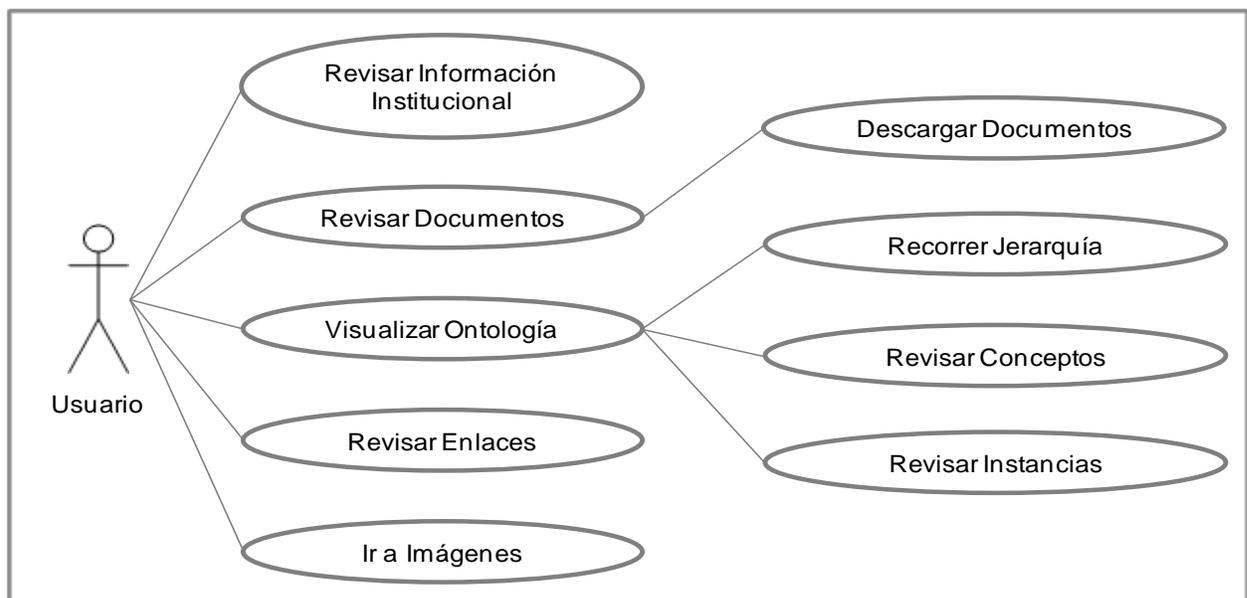


**Figura 9.** Plantilla de página del Sitio Web a ser generado.

- **Páginas que conformarán al Sitio Web:** de los Sitios Web mencionados para la visualización de Ontologías se logra identificar las páginas que conformarán al Sitio Web a ser generado, las mismas son mencionadas a continuación.
  - **Página de inicio:** será la página principal del Sitio Web. En ella podrá mostrarse una breve descripción del Sitio Web, información institucional, o información que el usuario de la herramienta considere pertinente.
  - **Página para la visualización de la Ontología:** en ella se podrá visualizar la Ontología.
  - **Página de documentos:** en esta página podrán publicarse los documentos relacionados con el dominio del conocimiento de la Ontología. Estos documentos podrán ser descargados.

- **Página de enlaces:** para la publicación de enlaces de interés.
- **Página de imágenes:** para la publicación de imágenes de interés.
- **Aspectos no considerados:** algunas de las funcionalidades de los Sitios Web para la visualización de Ontologías mencionados anteriormente no fueron considerados, éstas son mencionadas a continuación:
  - **Módulo administrativo:** el mismo no es considerado para los Sitios Web a ser generados ya que la funcionalidad está desarrollada en la herramienta o generador y permite la actualización del contenido mostrado en las distintas páginas del Sitio Web.
  - **Módulo para inferencias y búsqueda en la Ontología:** los Sitios Web a ser generados tienen como objetivo la visualización de una Ontología, por lo tanto esta funcionalidad no es considerada para los Sitios Web a ser generados.

En la **Figura 10** se muestra un diagrama de casos de uso general correspondiente a los Sitios Web ya generados.



**Figura 10.** Diagrama de casos de uso general – Sitio Web generado.

## 2.5. Desarrollo de la Aplicación

Para el desarrollo de la herramienta que permite generar una aplicación Web para la visualización de una ontología, se llevó a cabo las siguientes actividades:

- **Identificación y análisis de los requerimientos funcionales:** definición de las funcionalidades que satisface la aplicación.
- **Establecimiento de los aspectos técnicos:** definición de las herramientas y tecnologías a utilizar 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 almacenar los datos que necesita la aplicación en diversos procesos.
- **Estructura Lógica de Desarrollo de la Aplicación:** estructuración del código fuente.
- **Diseño de la Interfaz de Usuario:** creación de la interfaz a usar como pantalla de presentación de la información de la aplicación.
- **Implementación de las funcionalidades:** programación de los requerimientos definidos en actividades anteriores.
- **Evaluación de la Aplicación:** valoración de la aplicación por parte de expertos, 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 el análisis de los requerimientos funcionales de la herramienta, a través de casos de uso que permiten la descripción de las funcionalidades del sistema, independientes de la implementación; y por medio de diagramas que modelen las estructuras de datos necesarias para implementar la herramienta.

### 2.5.1. Identificación y Análisis de los Requerimientos Funcionales

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

- La herramienta propuesta estará formada por un módulo administrativo que permitirá al usuario:
  - Registrarse como usuario de la herramienta y modificar sus datos personales.
  - Crear sitios Web para su posterior publicación.
  - Crear páginas de documentos, enlaces de interés e imágenes en el contexto del Sitio Web a generar, así como también una página principal.
  - Cargar una ontología codificada en OWL en el contexto del sitio Web a generar. La herramienta permitirá cargar distintas versiones de la ontología y el usuario debe indicar cuál será la ontología que se visualizará en el sitio Web ya generado.
  - Modificar y/o eliminar cualquier página de documentos, enlaces de interés, imágenes creadas previamente de igual forma las ontologías cargadas.
  - Publicar el sitio Web. Esta opción permitirá al usuario visualizar el sitio Web construido a través de la herramienta y obtener el mismo en un formato comprimido que ocupe poco espacio.

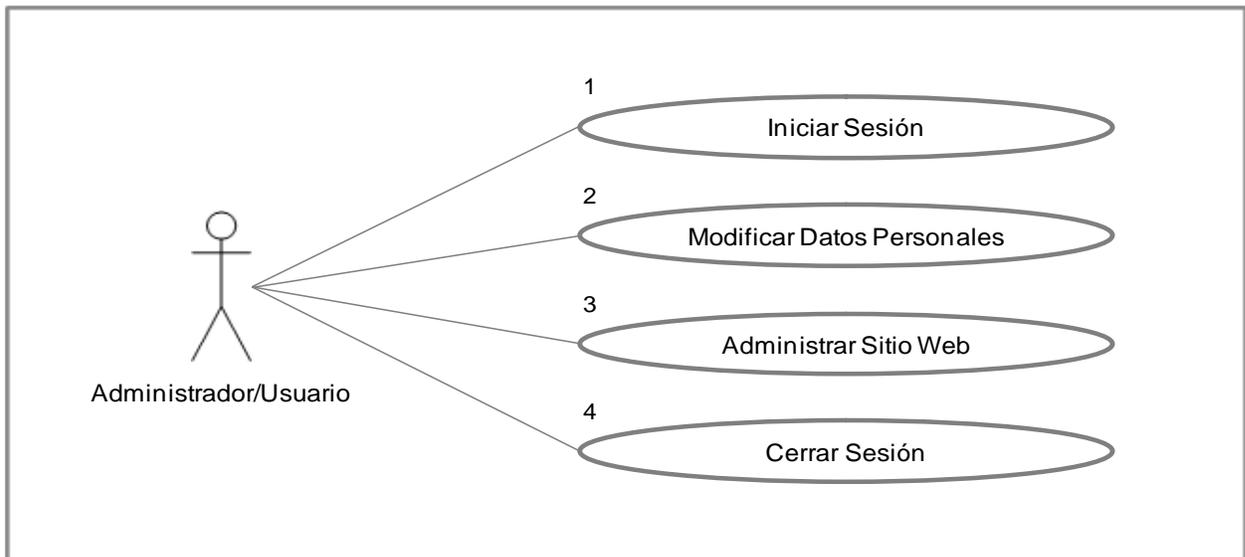
#### 2.5.1.1. Diagrama de casos de uso

A continuación se describen los casos de uso identificados para la implementación de la herramienta. Se muestran los casos de uso de nivel 0 y 1. En el **Anexo 1** y **Anexo 2** se describen los casos de uso de nivel 2 y de nivel 3 respectivamente.

**a) Actores del sistema: Administrador/Usuario.**

**b) Casos de uso de nivel 0**

En la **Figura 11** se muestra un diagrama general de las funciones con las que contará la aplicación propuesta.



**Figura 11.** Diagrama de CU – nivel 0.

- **Descripción de los casos de uso**

- **Nombre:** *Iniciar Sesión.*

- **Identificador:** CU 1.

- **Descripción:** permite autenticar al usuario dentro del sistema.

- **Pre-condición:** el usuario debe estar registrado en el sistema.

- **Pos-condición:** el usuario ha sido reconocido por el sistema y accede a la aplicación.

- **Curso básico de acción:**

1. El usuario ingresa su identificador de usuario y contraseña en los campos correspondientes, y hace clic en el botón *Iniciar Sesión*.
2. El sistema realiza una primera validación de los datos introducidos para determinar que el usuario no deje campos en blanco. [Curso alternativo A].

3. El sistema recibe los datos introducidos por el usuario, realiza una validación de los mismos contra los almacenados en la Base de Datos.
  4. Si los datos concuerdan, le es mostrada la página principal de la aplicación al usuario. [Curso alternativo B].
  5. Finaliza el caso de uso.
- **Cursos alternos:**
    - **Curso alternativo A:** el usuario no introdujo datos en alguno de los campos (*Identificador de Usuario, Contraseña*).
      - **A.1:** el sistema determina que el usuario no introdujo ningún valor en alguno de los campos necesarios para iniciar sesión.
      - **A.2:** el sistema informa del error ocurrido a través de un mensaje.
      - **A.3:** el control de flujo regresa al punto 1 del curso básico de acción.
    - **Curso alternativo B:** los datos introducidos por el usuario no coinciden con los almacenados en la Base de Datos.
      - **B.1:** el sistema determina que el valor de *Identificador de Usuario* no concuerda con el valor de la *Contraseña*.
      - **B.2:** el sistema informa del error ocurrido a través de un mensaje.
      - **B.3:** el control de flujo regresa al punto 1 del curso básico de acción.
  - **Nombre:** *Modificar Datos Personales*.
  - **Identificador:** CU 2.
  - **Descripción:** permite modificar los datos del usuario del sistema.
  - **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema.

- **Pos-condición:** el usuario logra modificar sus datos, los cambios son reflejados en la tabla correspondiente en la Base de Datos.
- **Curso básico de acción:**
  1. El usuario selecciona la opción *Modificar Datos Personales*.
  2. El sistema realiza una búsqueda en la tabla correspondiente de la Base de Datos.
  3. Los datos obtenidos, son mostrados a través de un formato que permita la visualización y actualización de los datos al mismo tiempo.
  4. El usuario cambia el valor del campo deseado y hace clic en el botón *Actualizar Datos*.
  5. El sistema realiza una primera validación de los nuevos valores introducidos para determinar que el usuario no dejará campos en blanco y que el formato de los mismos sea correcto. [Curso alternativo A].
  6. El sistema recibe los nuevos valores introducidos por el usuario, realiza una validación para determinar si el formato de los datos es correcto.
  7. En caso de que los valores posean los formatos adecuados, el sistema realiza la actualización de los mismos en la Base de Datos y le es notificado al usuario que la operación solicitada se realizó con éxito. [Curso alternativo A].
  8. Finaliza el caso de uso.
- **Cursos alternos:**
  - **Curso alternativo A:** algunos de los campos tiene un valor nulo o no respeta los formatos establecidos.
    - **A.1:** el sistema determina que el usuario dejó en blanco algunos de los campos correspondientes o no concuerdan con el formato preestablecido.
    - **A.2:** el sistema informa del error ocurrido a través de un mensaje.

- **A.3:** el control de flujo regresa al punto 3 del curso básico de acción.
  
- **Nombre: Administrar Sitio Web.**
  - **Identificador:** CU 3.
  - **Descripción:** permite al usuario realizar todas las actividades concernientes a la elaboración del Sitio Web a generar; como creación, modificación y eliminación de un Sitio Web.
  - **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema.
  - **Pos-condición:** el sistema muestra al usuario todos los sitios Web creados.
  
- **Nombre: Cerrar Sesión.**
  - **Identificador:** CU 4.
  - **Descripción:** permite al usuario cerrar una sesión previamente iniciada.
  - **Pre-condición:** ninguna.
  - **Pos-condición:** el usuario ha cerrado la sesión, previamente iniciada, y visualiza la página para iniciar sesión.
  - **Curso básico de acción:**
    1. El usuario selecciona la opción *Cerrar Sesión*.
    2. El sistema inhabilita la sesión iniciada.
    3. El sistema redirecciona el contexto de la aplicación a la página de inicio de sesión.
    4. Finaliza el caso de uso.

c) Casos de uso de nivel 1

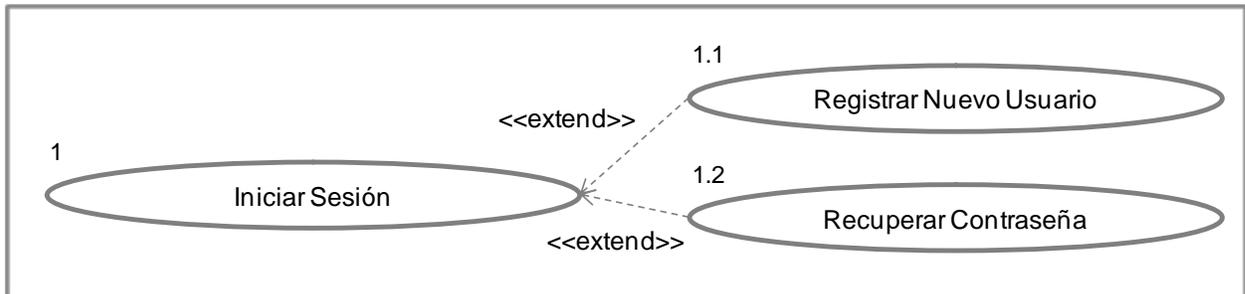


Figura 12. Diagrama de CU – nivel 1 – sub-módulo iniciar sesión.

- Descripción de los casos de uso

- **Nombre:** *Registrar Nuevo Usuario.*

- **Identificador:** CU 1.1.

- **Descripción:** permite registrar un nuevo usuario en el sistema.

- **Pre-condición:** ninguna.

- **Pos-condición:** el usuario ha sido registrado en el sistema y es posible acceder al mismo.

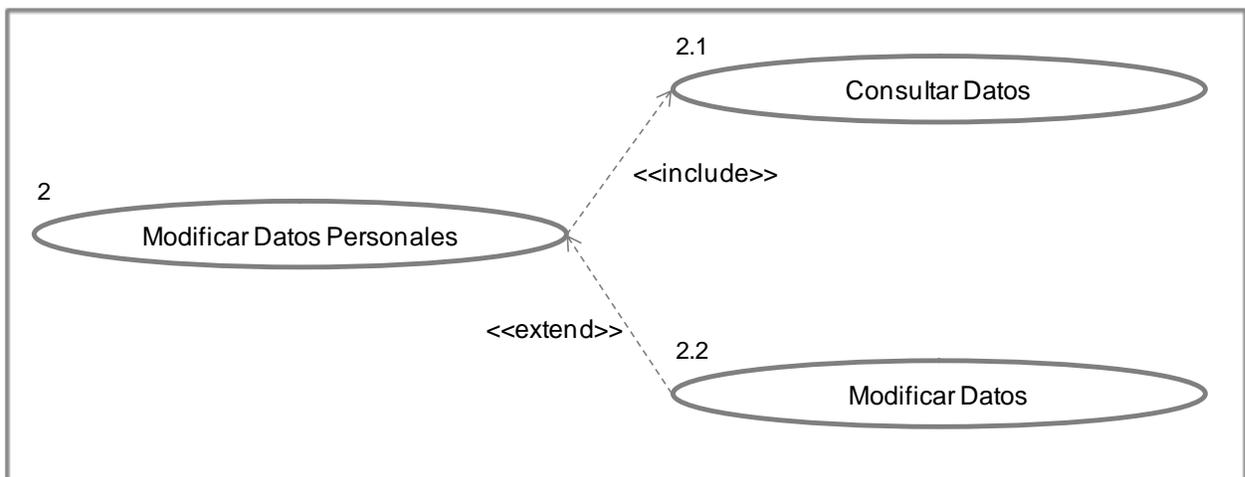
- **Curso básico de acción:**

1. El usuario selecciona la opción *Registrar Nuevo Usuario*.
2. El sistema le muestra al usuario un formulario para introducir sus datos.
3. El usuario introduce los datos solicitados por el sistema y hace clic en el botón *Registrarme*.
4. El sistema realiza una primera validación de los valores introducidos para determinar que el usuario no dejará campos en blanco y que el formato de los mismos sea correcto. [Curso alternativo A].
5. El sistema recibe los datos introducidos por el usuario y valida que los mismo sean correctos.

6. Si los datos son correctos, el sistema realiza la creación del usuario en el sistema, actualiza los campos correspondientes en la Base de Datos y le es notificado al usuario que la operación solicitada fue realizada con éxito. [Curso alternativo B].
  7. Finaliza el caso de uso.
- **Cursos alternos:**
    - **Curso alternativo A:** el usuario no introdujo datos en alguno de los campos correspondientes o al menos uno no coincide con el formato preestablecido.
      - **A.1:** el sistema determina que el usuario no introdujo ningún valor en alguno de los campos necesarios o que al menos uno de los mismos presenta valores erróneos.
      - **A.2:** el sistema informa del error ocurrido a través de un mensaje.
      - **A.3:** el control de flujo regresa al punto 2 del curso básico de acción.
    - **Curso alternativo B:** algunos de los campos tiene un valor nulo o no respeta los formatos establecidos.
      - **B.1:** el sistema determina que el usuario dejó en blanco algunos de los campos correspondientes o no concuerdan con el formato preestablecido.
      - **B.2:** el sistema informa del error ocurrido a través de un mensaje.
      - **B.3:** el control de flujo regresa al punto 2 del curso básico de acción.
  - **Nombre:** *Recuperar Contraseña.*
  - **Identificador:** CU 1.2.
  - **Descripción:** permite al usuario recuperar su contraseña en caso de no recordarla.
  - **Pre-condición:** el usuario debe estar registrado en el sistema.

- **Pos-condición:** el sistema cambia la contraseña del usuario y envía la nueva contraseña a su dirección de correo electrónico.
- **Curso básico de acción:**
  1. El usuario selecciona la opción *Olvido de Contraseña*.
  2. El sistema le muestra al usuario un formulario para introducir su *Identificador de Usuario y Correo Electrónico*.
  3. El usuario introduce los datos solicitados por el sistema y hace clic en el botón *Siguiente*.
  4. El sistema realiza una primera validación de los valores introducidos para determinar que el usuario no dejará campos en blanco y que el formato de los mismos sea correcto. [Curso alterno A].
  5. El sistema recibe los datos introducidos por el usuario y valida que los mismo sean correctos.
  6. Si los datos son correctos, el sistema muestra al usuario un nuevo formulario con la pregunta secreta introducida por el usuario al momento de registrarse y un campo para la respuesta a dicha pregunta.
  7. El usuario introduce su respuesta y presiona sobre el botón *Siguiente*.
  8. El sistema realiza una primera validación de los valores introducidos para determinar que el usuario no dejará campos en blanco y que el formato de los mismos sea correcto. [Curso alterno B].
  9. El sistema determina si los datos introducidos por el sistema coinciden con los almacenados en la Base de Datos, en caso afirmativo procede a generar de forma aleatoria una nueva contraseña, almacenarla en el campo correspondiente a la Base de Datos y enviarla al correo electrónico del usuario. En caso contrario muestra un mensaje de error al usuario.
  10. Finaliza el caso de uso.

- **Cursos alternos:**
  - **Curso alternativo A:** el usuario no introdujo datos en alguno de los campos correspondientes.
    - **A.1:** el sistema determina que el usuario no introdujo ningún valor en alguno de los campos necesarios o que al menos uno de los mismos presenta valores erróneos.
    - **A.2:** el sistema informa del error ocurrido a través de un mensaje.
    - **A.3:** el control de flujo regresa al punto 2 del curso básico de acción.
  - **Curso alternativo B:** algunos de los campos tiene un valor nulo o no respeta los formatos establecidos.
    - **B.1:** el sistema determina que el usuario dejó en blanco algunos de los campos correspondientes o no concuerdan con el formato preestablecido e informa del error ocurrido a través de un mensaje.
    - **B.3:** el control de flujo regresa al punto 6 del curso básico de acción.



**Figura 13.** Diagrama de CU – nivel 1 – sub-módulo modificar datos personales.

- **Descripción de los casos de uso**

- **Nombre:** *Consultar Datos.*
- **Identificador:** CU 2.1.
- **Descripción:** permite mostrar los datos asociados al usuario.
- **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema.
- **Pos-condición:** el usuario logra visualizar sus datos.
  
- **Nombre:** *Modificar Datos.*
- **Identificador:** CU 2.2.
- **Descripción:** permite actualizar los datos de un usuario.
- **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Modificar Datos Personales.*
- **Pos-condición:** los datos actualizados son almacenados en la Base de Datos.
- **Curso básico de acción:**
  1. El usuario selecciona la opción *Actualizar Datos.*
  2. El sistema realiza una primera validación de los valores introducidos, con la finalidad de determinar que el usuario no dejará campos en blanco y que el formato de los mismos sea correcto. [Curso alternativo A].
  3. El sistema recibe los datos introducidos por el usuario y valida que los mismo sean correctos.
  4. Si los datos son correctos, el sistema actualiza los campos correspondientes en la Base de Datos y le es notificado al usuario que la operación solicitada fue realizada con éxito. [Curso alternativo B].
  5. Finaliza el caso de uso.
- **Cursos alternos:**

- **Curso alterno A:** el usuario no introdujo datos en alguno de los campos correspondientes o al menos uno no coincide con el formato preestablecido.
  - **A.1:** el sistema determina que el usuario no introdujo ningún valor en alguno de los campos necesarios o que al menos uno de los mismos presenta valores erróneos.
  - **A.2:** el sistema informa del error ocurrido a través de un mensaje.
  - **A.3:** el control de flujo regresa al punto 1 del curso básico de acción.
- **Curso alterno B:** algunos de los campos tiene un valor nulo o no respeta los formatos establecidos.
  - **B.1:** el sistema determina que el usuario dejó en blanco algunos de los campos correspondientes o no concuerdan con el formato preestablecido.
  - **B.2:** el sistema informa del error ocurrido a través de un mensaje.
  - **B.3:** el control de flujo regresa al punto 1 del curso básico de acción.

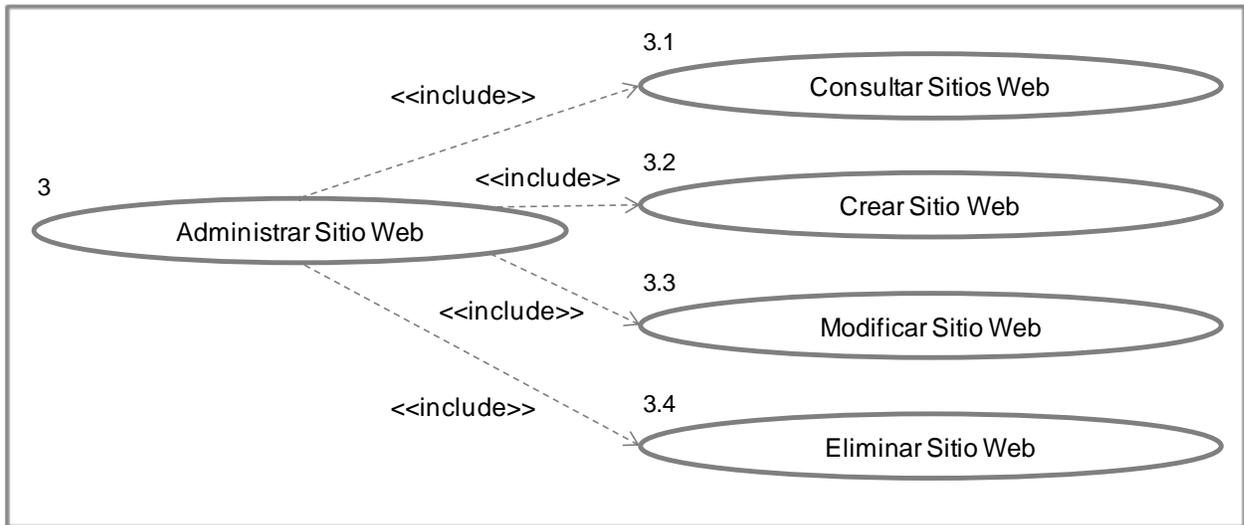


Figura 14. Diagrama de CU – nivel 1 – sub-módulo administrar sitio Web.

- **Descripción de los casos de uso**

- **Nombre:** *Consultar Sitios Web.*

- **Identificador:** CU 3.1.

- **Descripción:** permite listar los sitios Web creados por el usuario.

- **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Sitio Web*.

- **Pos-condición:** se le muestra al usuario un listado de los sitios Web creados con anterioridad, en caso contrario, el sistema indica dicha situación a través de un mensaje.

- **Nombre:** *Crear Sitio Web.*

- **Identificador:** CU 3.2.

- **Descripción:** permite al usuario crear un nuevo sitio Web.

- **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Sitio Web* y seguidamente la opción *Crear Sitio Web*.

- **Pos-condición:** es mostrado al usuario un formulario con los valores requeridos por el sistema para crear un sitio Web.
- **Curso básico de acción:**
  1. El usuario selecciona la opción *Crear Sitio Web*.
  2. El sistema le muestra al usuario un formulario para que se introduzcan los valores correspondientes.
  3. El sistema realiza una primera validación de los valores introducidos, con la finalidad de determinar que el usuario no dejará campos en blanco y que el formato de los mismos sea correcto. [Curso alternativo A].
  4. El sistema recibe los datos introducidos por el usuario y valida que los mismo sean correctos.
  5. Si los datos son correctos, el sistema actualiza los campos correspondientes en la Base de Datos y le es notificado al usuario que la operación solicitada fue realizada con éxito. [Curso alternativo B].
  6. Finaliza el caso de uso.
- **Cursos alternos:**
  - **Curso alternativo A:** el usuario no introdujo datos en alguno de los campos correspondientes o al menos uno no coincide con el formato preestablecido.
    - **A.1:** el sistema determina que el usuario no introdujo ningún valor en alguno de los campos necesarios o que al menos uno de los mismos presenta valores erróneos.
    - **A.2:** el sistema informa del error ocurrido a través de un mensaje.
    - **A.3:** el control de flujo regresa al punto 2 del curso básico de acción.
  - **Curso alternativo B:** algunos de los campos tiene un valor nulo o no respeta los formatos establecidos.

- **B.1:** el sistema determina que el usuario dejó en blanco algunos de los campos correspondientes o no concuerdan con el formato preestablecido.
  - **B.2:** el sistema informa del error ocurrido a través de un mensaje.
  - **B.3:** el control de flujo regresa al punto 2 del curso básico de acción.
- 
- **Nombre: *Modificar Sitio Web.***
  - **Identificador:** CU 3.3.
  - **Descripción:** permite actualizar los datos de un sitio Web.
  - **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Modificar Sitio Web.*
  - **Pos-condición:** los datos actualizados son almacenados en la Base de Datos.
  - **Curso básico de acción:**
    1. El usuario selecciona la opción *Actualizar Datos.*
    2. El sistema realiza una primera validación de los valores introducidos, con la finalidad de determinar que el usuario no dejará campos en blanco y que el formato de los mismos sea correcto. [Curso alternativo A].
    3. El sistema recibe los datos introducidos por el usuario y valida que los mismo sean correctos.
    4. Si los datos son correctos, el sistema actualiza los campos correspondientes en la Base de Datos y le es notificado al usuario que la operación solicitada fue realizada con éxito. [Curso alternativo B].
    5. Finaliza el caso de uso.
  - **Cursos alternos:**

- **Curso alternativo A:** el usuario no introdujo datos en alguno de los campos correspondientes o al menos uno no coincide con el formato preestablecido.
    - **A.1:** el sistema determina que el usuario no introdujo ningún valor en alguno de los campos necesarios o que al menos uno de los mismos presenta valores erróneos.
    - **A.2:** el sistema informa del error ocurrido a través de un mensaje.
    - **A.3:** el control de flujo regresa al punto 1 del curso básico de acción.
  - **Curso alternativo B:** algunos de los campos tiene un valor nulo o no respeta los formatos establecidos.
    - **B.1:** el sistema determina que el usuario dejó en blanco algunos de los campos correspondientes o no concuerdan con el formato preestablecido.
    - **B.2:** el sistema informa del error ocurrido a través de un mensaje.
    - **B.3:** el control de flujo regresa al punto 1 del curso básico de acción.
- 
- **Nombre:** *Eliminar Sitio Web.*
  - **Identificador:** CU 3.4.
  - **Descripción:** permite eliminar el sitio Web seleccionado por el usuario.
  - **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Sitio Web.*
  - **Pos-condición:** el sitio Web es eliminado, se actualizan los datos correspondientes en la Base de Datos y es mostrado al usuario un listado de los sitios Web creados con anterioridad, en caso contrario, el sistema indica dicha situación a través de un mensaje.

#### d) Casos de uso de nivel 2

Los casos de uso de nivel 2 pueden ser consultados en el Anexo 1 de este documento.

#### e) Casos de uso de nivel 3

Los casos de uso de nivel 3 pueden ser consultados en el Anexo 2 de este documento.

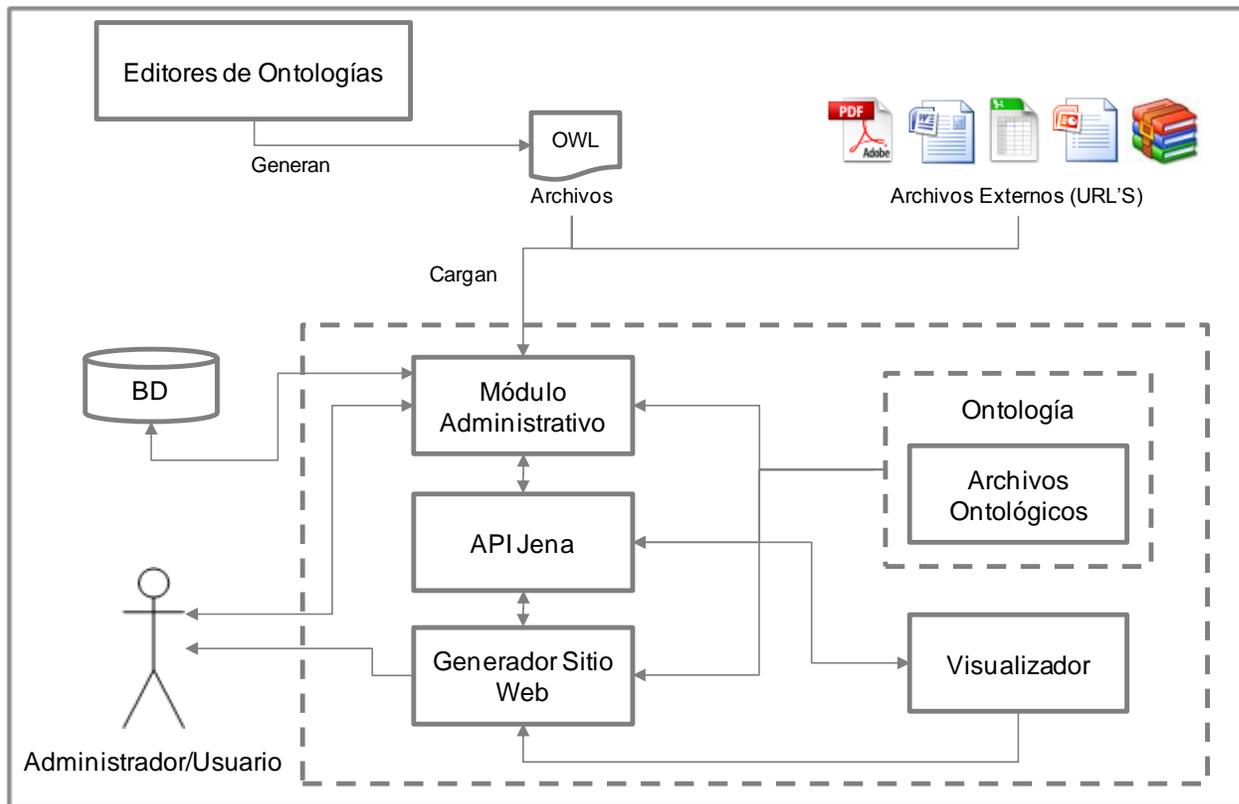
### 2.5.2. Aspectos Técnicos

La especificación de la plataforma de desarrollo de la aplicación propuesta se presenta a continuación:

- **Especificaciones de hardware:** el hardware utilizado durante la implementación de la herramienta y considerado como mínimo para el buen funcionamiento de la misma es el siguiente:
  - Procesador AMD Athlon 64 2500+.
  - 1024 MB de memoria RAM.
  - Disco Duro de 80 GB.
- **Sistema operativo:** la aplicación ha sido desarrollada bajo el Sistema Operativo Windows XP y Linux Ubuntu 8.04, sin embargo, el lenguaje de programación en el cual fue desarrollada la aplicación permite a la misma ejecutarse bajo cualquier Sistema Operativo que tenga instalado el kit de desarrollo Java junto con un contenedor de aplicaciones Web.
- **Contenedor de aplicaciones Web:** durante el desarrollo de la aplicación ha sido utilizado el contenedor Web Apache Tomcat 5.5.9, el cual permite compilar y desplegar archivos JSP (Java Server Pages) y administrar el ciclo de vida de los Servlets para el control de la aplicación. También fue utilizado Resin 3.1.5.
- **Herramientas de Software:** a continuación se presentan las herramientas a utilizar durante el proceso de desarrollo de la aplicación:

- **Kit de desarrollo Java (JDK):** conjunto de herramientas, utilidades, documentación y ejemplos para desarrollar aplicaciones en el lenguaje de programación Java. Se elige dicho lenguaje por el beneficio de multiplataforma explícito en las aplicaciones desarrolladas con el mismo.
- **Editor de Ontologías Protégé versión 3.2.1:** es una herramienta para el modelado de conocimiento en forma de ontología.
- **Eclipse:** es un entorno de desarrollo integrado (IDE), de código abierto y multiplataforma para el desarrollo de aplicaciones en distintos lenguajes de programación, entre ellos Java.
- **Editor de archivos Notepad++.**
- **Servidor de Base de Datos MySql 5.0.22.**
- **Navegador Web:** la aplicación a desarrollar se encuentra en un ambiente Web, es necesario tener un navegador para poder visualizar las páginas html y jsp desarrolladas. Los navegadores usados durante la fase de desarrollo son: Mozilla Firefox 2.0.0.13+ (basado en Netscape), Internet Explorer 7 y Safari 3.1.1.
- **Java Server Pages (JSP) y Servlets:** debido a que se ha escogido como entorno de desarrollo el ambiente que nos proporciona Java, se usaran páginas JSP, ya que son páginas web dinámicas cuyo procesamiento de código se realiza en el servidor. Por otro lado, los Servlet han sido usados para implementar la lógica que debe tener cada página.
- **Lenguaje de marcado extensible (XML):** los archivos XML en la aplicación fueron utilizados para la configuración del sistema.
- **Paquete Jakarta Commons Fileupload:** paquete de gran utilidad en procesos que involucren carga de archivo al servidor.
- **Mail:** paquetes para el envío de mensajes a correos electrónicos.

### 2.5.3. Arquitectura de la Aplicación



**Figura 15.** Arquitectura de la herramienta.

En la **Figura 15** se muestra la arquitectura de la herramienta implementada, la cual está conformada por 5 módulos y una base de datos. El módulo administrativo permite al usuario interactuar con la herramienta. El API de Jena es la encargada de realizar el procesamiento de los archivos ontológicos almacenados con la finalidad de que el módulo de visualización muestre los mismos al usuario en un formato de fácil entendimiento. El módulo generar Sitio Web se encarga de realizar una recopilación de los datos introducidos por el usuario para generar las páginas correspondientes, este módulo se encarga también de la integración del Sitio Web resultante con el módulo visualizador de ontologías, finalmente permite al usuario realizar la descarga del contenido generado.

### 2.5.4. Diseño del Modelo de Navegación entre Páginas

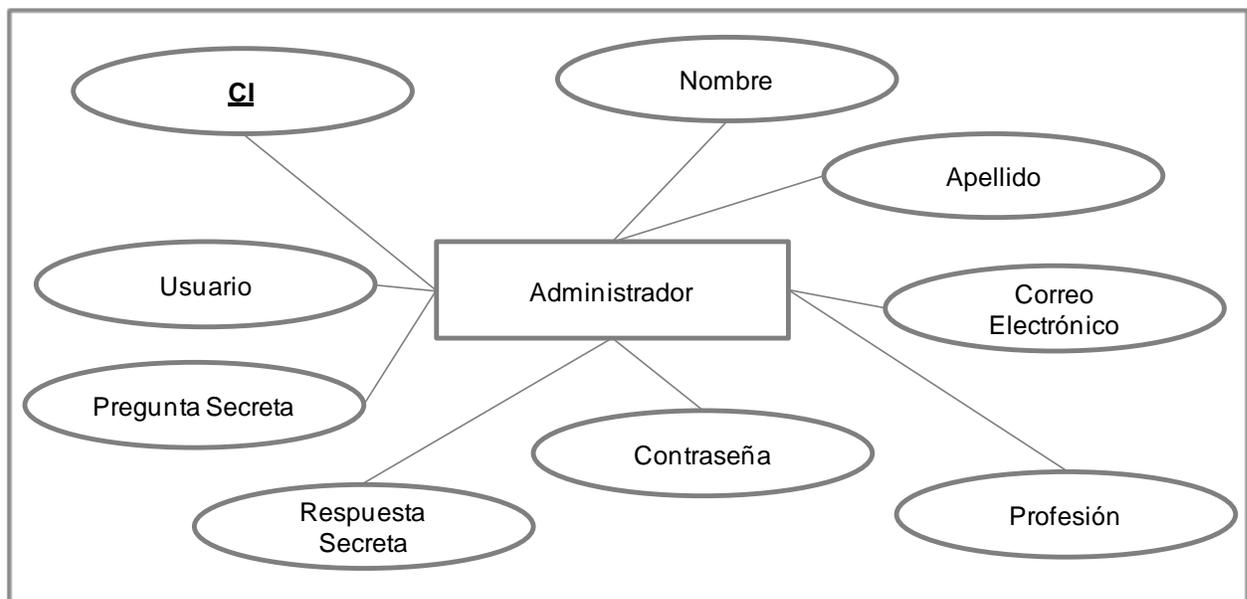
Para la implementación de la herramienta, se hizo necesario conocer el flujo básico de navegación entre las páginas, éste se encuentra modelado en un diagrama de robustez el cual es importante para definir la interacción de componentes y páginas con el usuario y la base de datos, dicho diagrama puede ser consultado en el **Anexo 3**.

### 2.5.5. Definición de la Base de Datos

#### a) Entidades

Para el desarrollo de la aplicación propuesta se han identificado las siguientes entidades:

- **Administrador/Usuario:** contiene los campos que identifican a una persona dentro del sistema.



**Figura 16.** E/R entidad administrador.

- **Sitio Web:** contiene los campos necesarios para almacenar la información concerniente al sitio Web a generar.

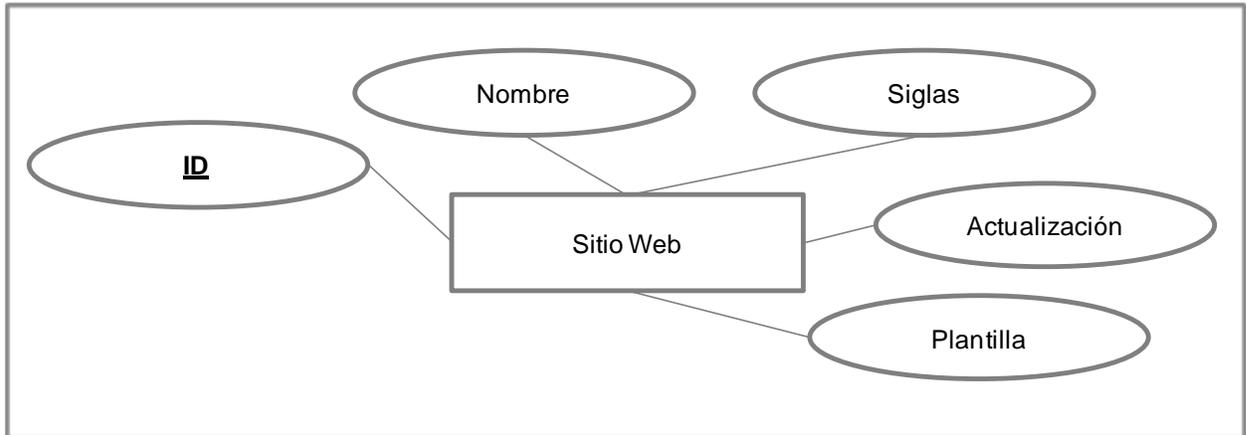


Figura 17. E/R entidad sitio Web.

- **Página Principal:** contiene los campos necesarios para identificar la página principal almacenada en el servidor donde se alojará la aplicación generada.

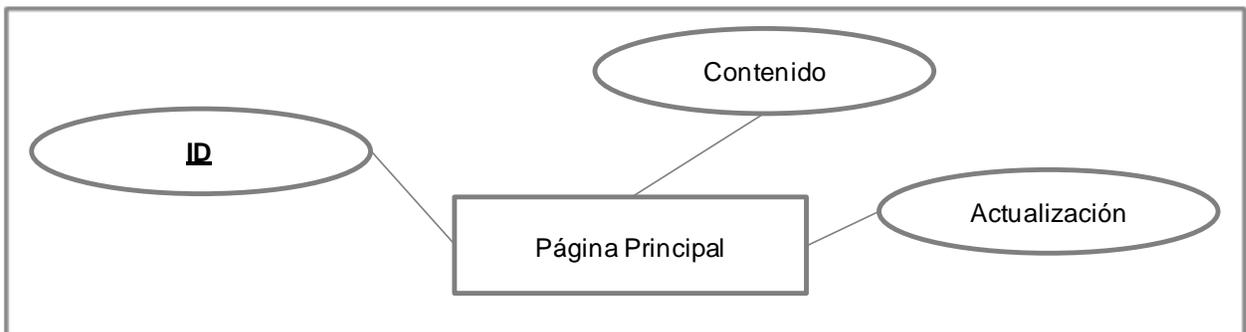


Figura 18. E/R entidad página principal.

- **Página de Documento:** contiene los campos necesarios para identificar la página de documentos almacenada en el servidor donde se alojará la aplicación generada.

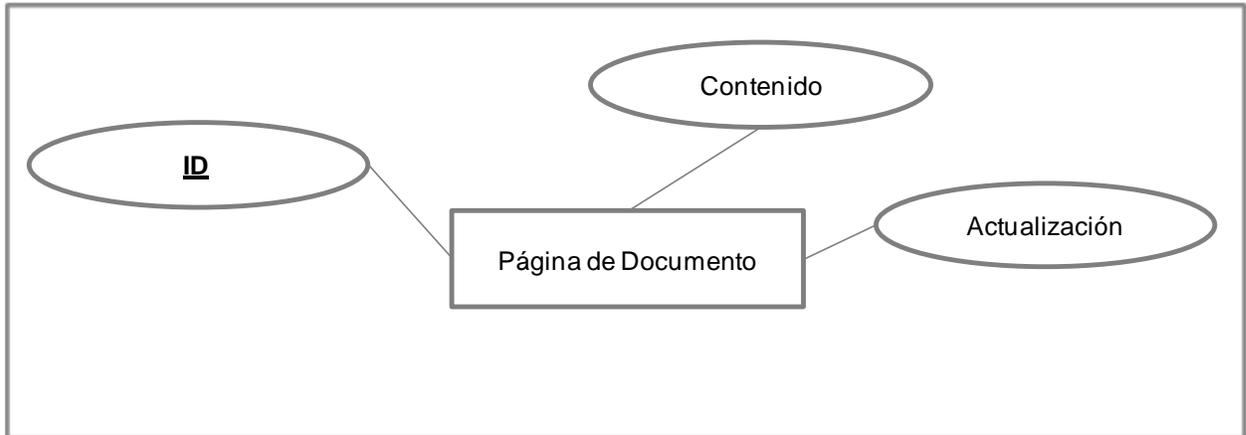


Figura 19. E/R entidad página de documento.

- **Página de Enlace:** contiene los campos para identificar la página de enlaces que será publicada en el sitio Web a generar.

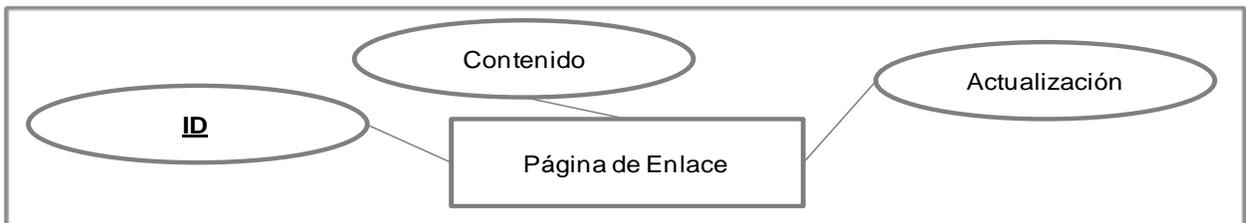


Figura 20. E/R entidad página de enlace.

- **Página de Imágenes:** entidad que posee los campos que permiten identificar la página de imágenes de la aplicación generada.

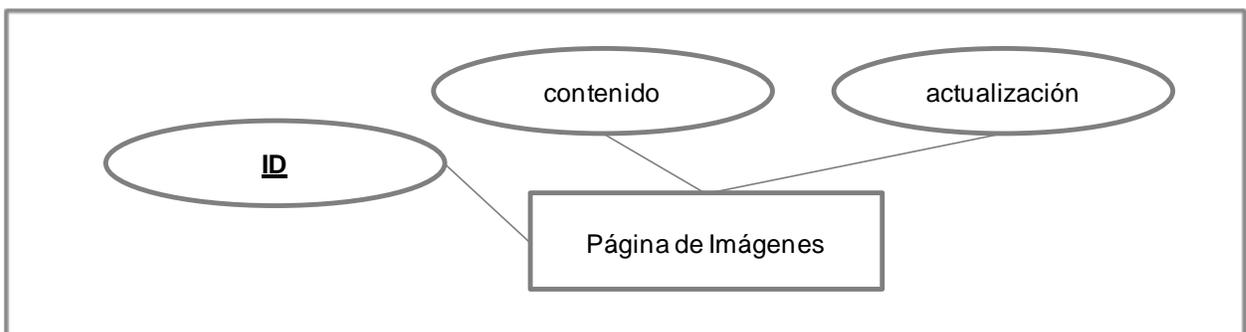
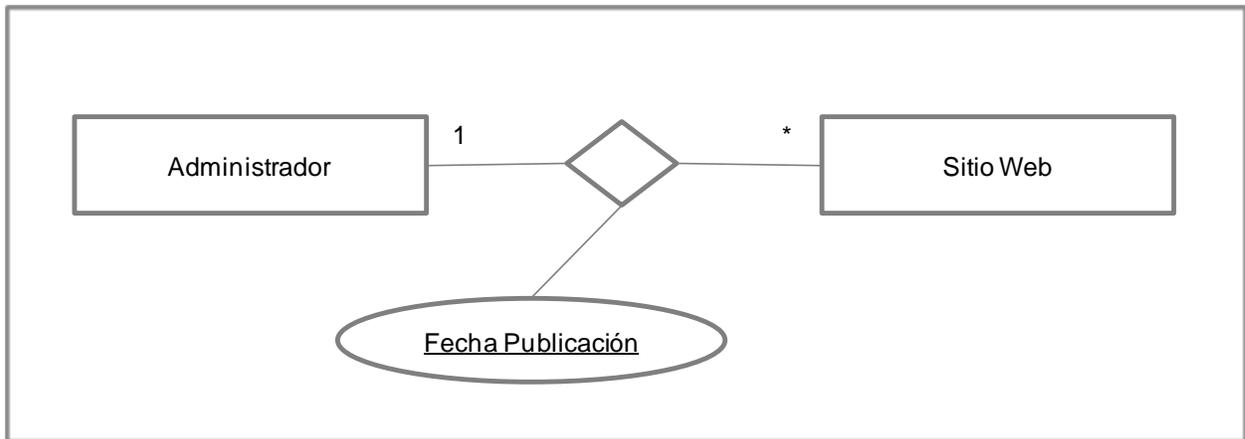


Figura 21. E/R entidad imagen.

**b) Relaciones**

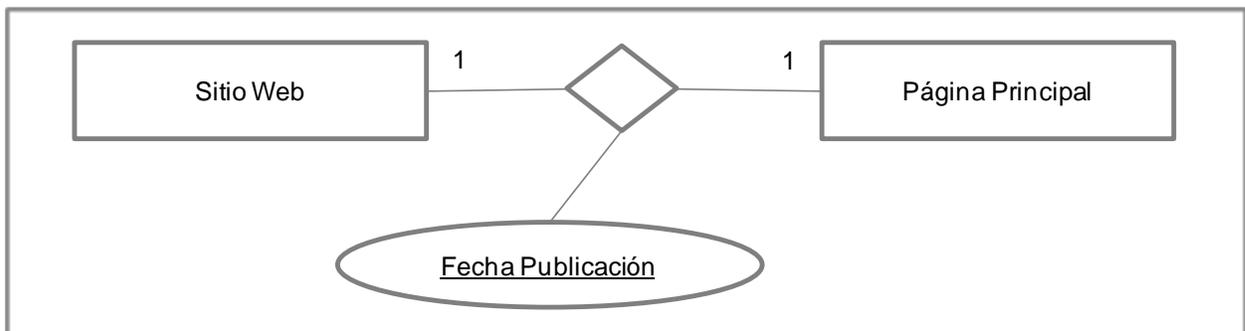
Para el desarrollo de la aplicación propuesta se han identificado las siguientes relaciones:

- **Administrador\_SitioWeb:**



**Figura 22.** E/R relación administrador\_sitioweb.

- **SitioWeb\_Página-Principal:**



**Figura 23.** E/R relación sitioweb\_página-principal

- **SitioWeb\_Página-Documento:**

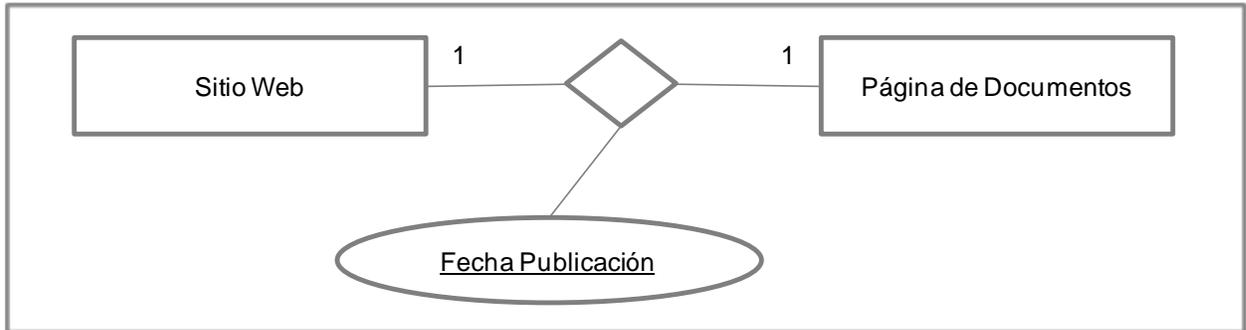


Figura 24. E/R relación sitioweb\_página-documento.

- **SitioWeb\_Página-Enlace:**

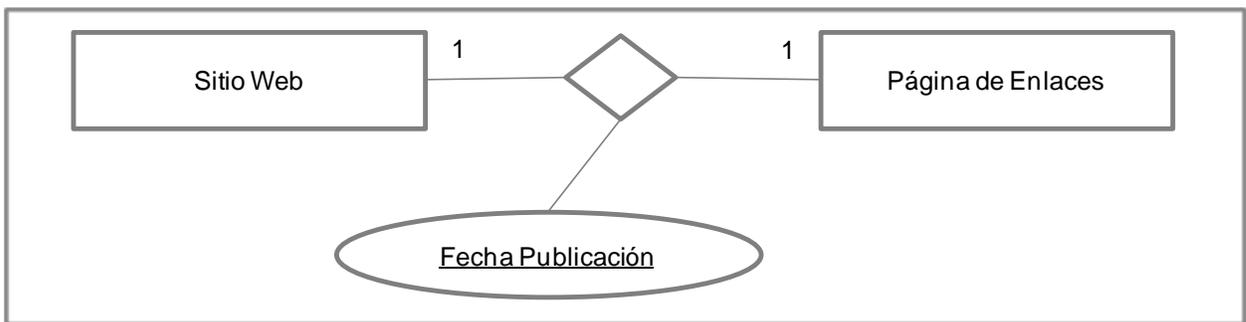


Figura 25. E/R relación sitioweb\_página-enlace.

- **SitioWeb\_Página-Imágenes:**

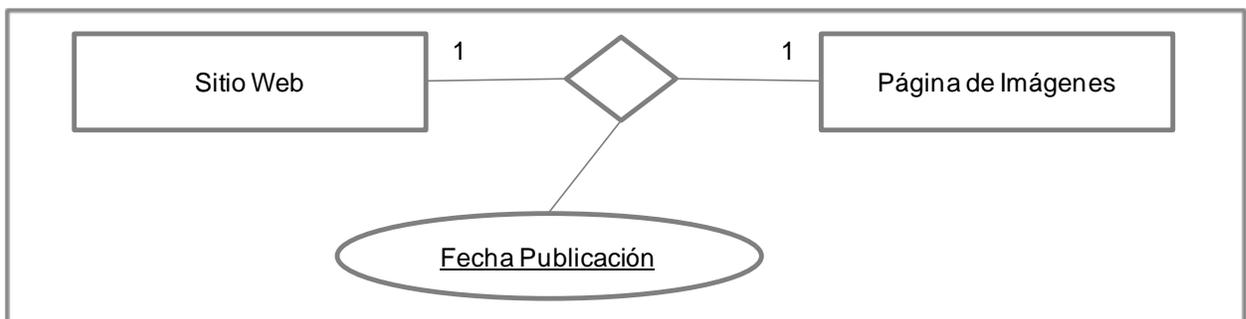


Figura 26. E/R relación sitioweb\_página-imágenes.

### c) Modelo Final Entidad – Relación

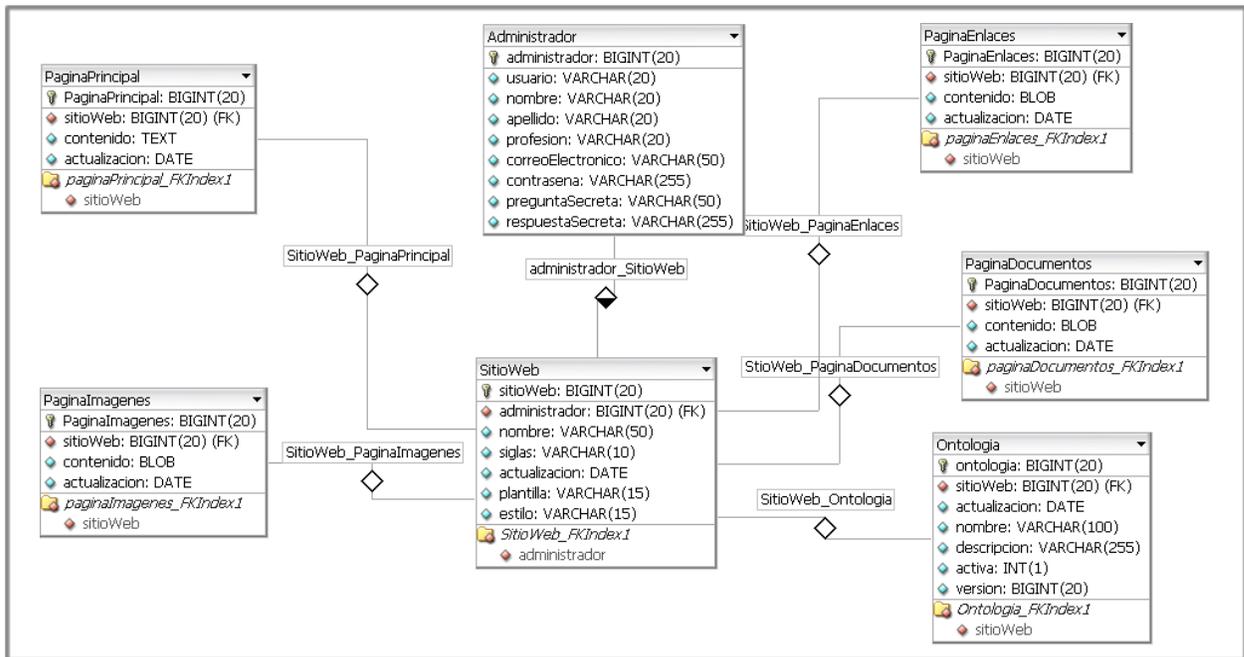


Figura 27. Diagrama entidad – relación.

#### 2.5.6. Estructura Lógica de Desarrollo de la Aplicación

Para la implementación de la aplicación se desarrolla el código siguiendo el patrón arquitectónico Web MVC (Modelo – Vista – Controlador) el cual se estructura en capas, en ellas se usan las tecnologías que fueron explicadas anteriormente. Estas capas o niveles se describen a continuación:

- **Capa de modelo:** conformada por las clases que representan los objetos del dominio de la información a ser utilizada por la aplicación. El modelo es otra forma de llamar a la capa de dominio.
- **Capa controladora:** contiene las clases acciones que atienden las peticiones del usuario, e invoca métodos para cambiar el modelo y probablemente la vista.
- **Capa de vista:** incluye las páginas dinámicas jsp con las cuales va a interactuar el usuario. Presenta el modelo en un formato adecuado para que el usuario interactúe con la aplicación (interfaz de usuario).

En cuanto a la estructura física de archivos, el código de la aplicación está estructurado en paquetes, los mismos permiten una mejor organización de las clases y archivos que conforman la herramienta, mejorando notablemente el fácil entendimiento para futuros mantenimientos. A continuación se listan dichos paquetes:

- **lia.generador.hibernate:** contiene la clase encargada de establecer la conexión con la Base de Datos a través del entorno de trabajo Hibernate.
- **lia.generador.hibernate.mapeo:** este paquete contiene los archivos necesarios para que el entorno de trabajo Hibernate realice el mapeo de las entidades de la Base de Datos con las clases de java que representan dichas entidades. Estos archivos están codificados en XML (Lenguaje de Marcas Extensible).
- **lia.generador.struts.control.action:** empaqueta las clases controladoras de la herramienta. Estas clases son las encargadas de procesar las solicitudes realizadas por los usuarios y mostrar una respuesta a través de una interfaz.
- **lia.generador.struts.control.action.actionform:** encapsula los formularios utilizados en las diferentes operaciones en las que el usuario debe ingresar datos que serán almacenados en la Base de Datos a través del entorno de trabajo Hibernate, como por ejemplo: registro de un SitioWeb. Estos archivos son necesarios ya que el entorno de trabajo Struts toma los datos introducidos por el usuario a través de estos formularios.
- **lia.generador.struts.control.action.filtro:** este paquete consta de una sola clase que permite capturar las peticiones del usuario y darle formato siguiendo una codificación de caracteres UTF-8. Esto es necesario para asegurar que sean almacenados exactamente los datos proporcionado por el usuario. Esta codificación de caracteres es una de las más utilizadas.
- **lia.generador.struts.modelo:** contiene las clases necesarias para administrar los objetos de dominio. Permite obtener los datos almacenados en la Base de Datos a través del entorno de trabajo Hibernate, este último se encarga de la

conexión entre la herramienta y la Base de Datos, también del mapeo entre los datos y los objetos de dominio.

- **lia.generador.struts.modelo.estructura:** dicho paquete consta de los objetos de dominio. Los mismos mantienen una relación estrecha con las entidades presentes en la Base de Datos. Razón por la cual el entorno de trabajo Hibernate logra realizar el mapeo entre los objetos de dominio y las entidades de la Base de Datos.
- **lia.generador.struts.recursos:** este paquete contiene el archivo que almacena todos los mensajes mostrados en la herramienta. Struts utiliza este archivo para generar las distintas páginas mostradas a los usuarios así como también cualquier mensaje presentado por la herramienta.
- **lia.generador.utilidades:** este paquete cuenta con una serie de clases que implementan funcionalidades que complementan las de otras clases. Por ejemplo: manejo de fechas, generación aleatoria de contraseñas, obtención de rutas absolutas en el servidor donde se aloja la herramienta, entre otras.

### 2.5.7. 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 o herramienta y dependiendo de la misma, será grato o no el uso de ésta. Según Myers (1996) una interfaz es un conjunto de comandos o menús a través de los cuales el usuario se comunica con el programa, por lo que la interfaz establece un vínculo entre el usuario y el programa.

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. Los mismos toman en cuenta varios puntos de vista referentes a cómo el usuario se sentirá al momento de interactuar con el sistema o herramienta, 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:

- **Visibilidad del estatus del sistema:** el sistema debe transmitir en todo momento su estado. El usuario debe saber si se está produciendo la carga de una página o esta acción ha sido interrumpida por algún problema.
- **Correspondencia entre el sistema y el mundo real:** la información presente en las páginas debe considerar la realidad de su usuario ideal.
- **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.

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

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

- **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.
- **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.
- **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

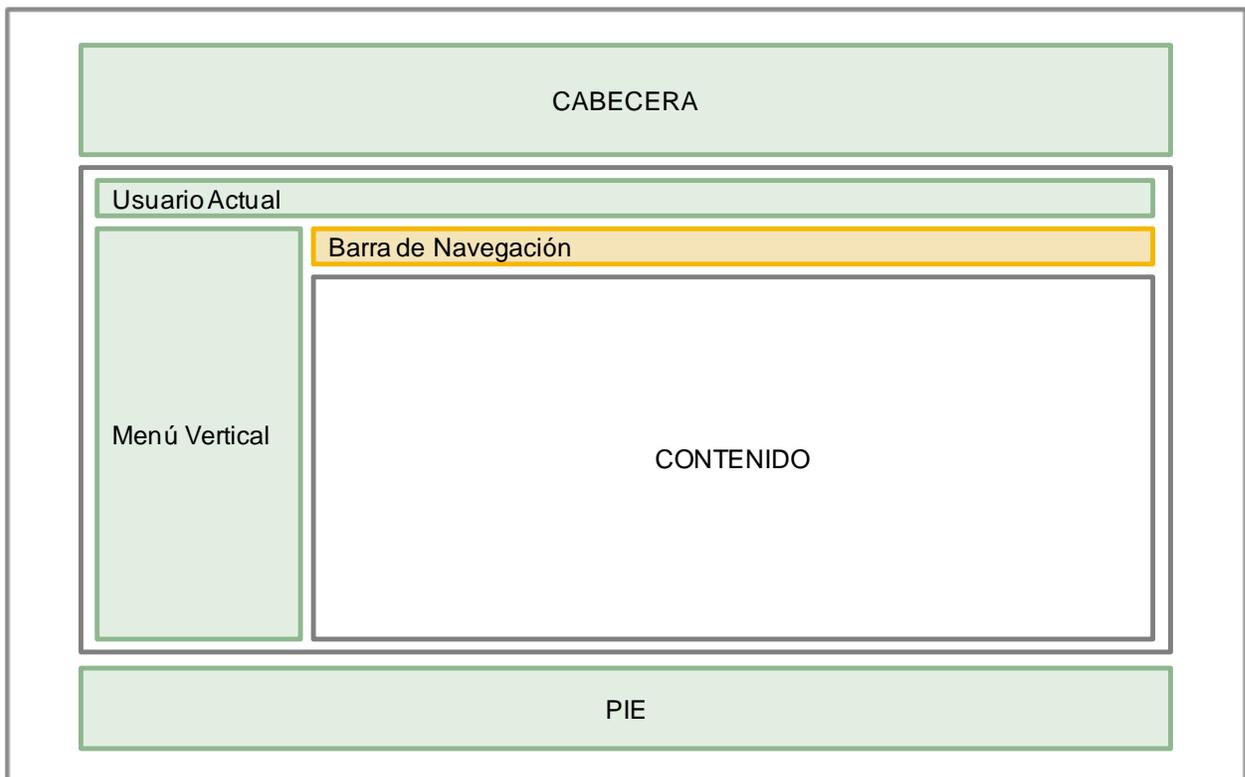
petición por parte del usuario. Es vital que una vez que se produzca un error, el sistema se lo haga saber rápida y claramente al usuario, le advierta sobre la severidad del mismo y le provea de algún mecanismo para recuperarse de ese error.

- **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.
- **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 herramienta, durante el desarrollo e implementación de la misma, se tuvo en cuenta varios de los principios, heurísticas y atributos especificados con anterioridad. A continuación se muestra una descripción de los mismos:

- El generador de visualizadores de ontologías codificadas en OWL presenta una interfaz consistente, la misma es presentada a través de una plantilla que logra identificarse en toda la herramienta. Dicha plantilla, se compone de una cabecera, un contenido y finalmente un pie. En la cabecera de la plantilla se visualiza el nombre de la herramienta, el contenido mostrado al usuario logra visualizarse por debajo de ésta y por último el pie, que presenta al usuario un listado de los navegadores compatibles con la herramienta, también presenta información de los derechos de autor de la misma.
- Una vez que el usuario ha iniciado sesión, la parte correspondiente al contenido se divide en bloques, siguiendo una nueva plantilla que consta de un panel que muestra el usuario actual de la herramienta, un panel horizontal correspondiente a la barra de navegación, que permite al usuario saber en todo momento donde se encuentra dentro del contexto de la herramienta, así como también el curso de navegación para llegar a la opción deseada, esta barra de navegación

permite al usuario regresar fácilmente a la página anterior. La sub-plantilla mencionada anteriormente, presenta un panel vertical correspondiente al menú de opciones, el cual permite al usuario acceder rápidamente a las funcionalidades ofrecida por la herramienta. Por último, el panel de contenido que muestra los mensajes, opciones, captura de datos y resultados generados por la herramienta durante la interacción de esta con el usuario. En la **Figura 28** se muestra la plantilla general de la interfaz una vez que el usuario ha iniciado sesión en la herramienta.

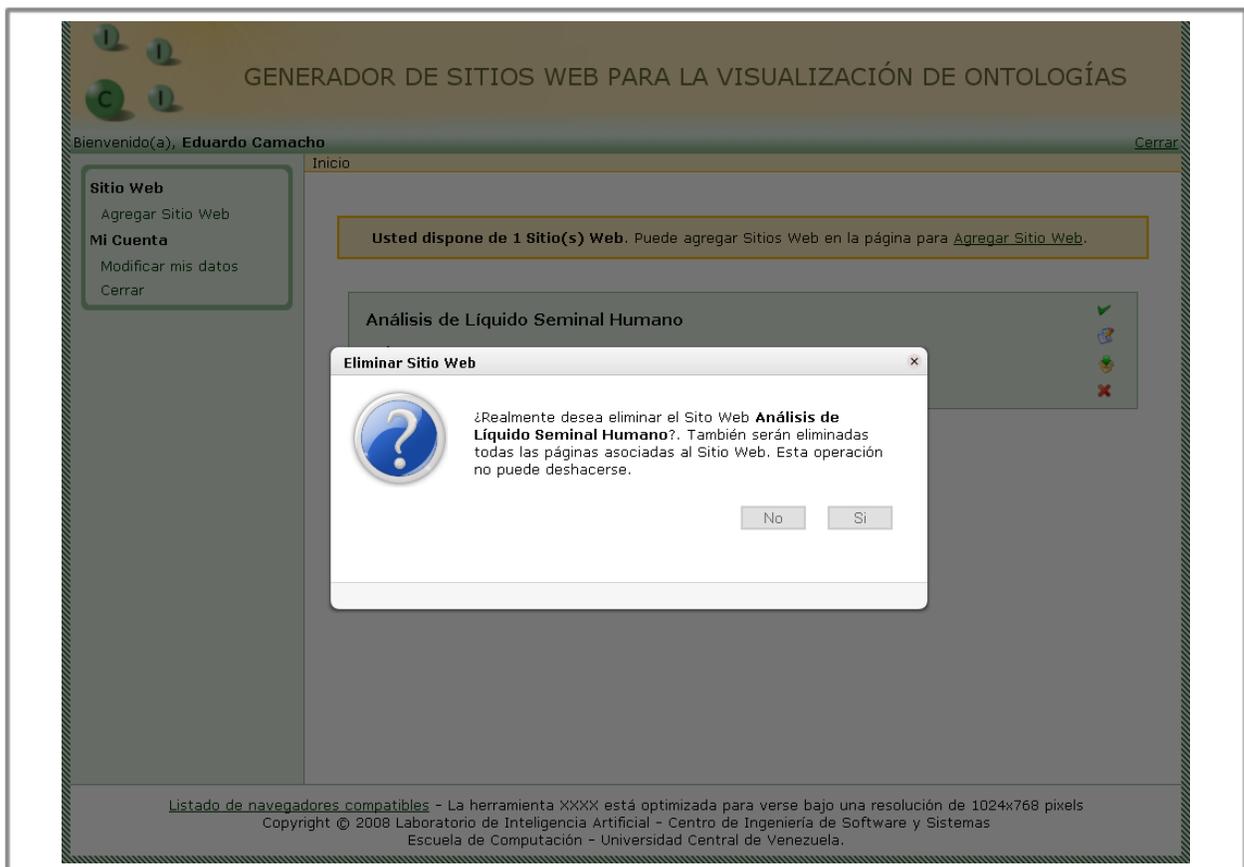


**Figura 28.** Plantilla general de interfaz de usuario.

- Al momento de usar iconos en la herramienta, es importante conocer la importancia de los mismos y que éstos sean asociados de forma conveniente con 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 malinterpretarlos. Por esta razón, al momento de utilizarlos se asocia una breve

descripción de la acción que representan, esta descripción logra visualizarse al momento de sobreponer el cursor del ratón encima de los iconos.

- Cuando el usuario se dispone a realizar una acción importante e irreversible, la herramienta por medio de un cuadro de diálogo le pregunta al usuario si está seguro de realizar dicha acción. En la **Figura 29** se muestra el cuadro de dialogo al momento en que el usuario desea eliminar un Sitio Web.



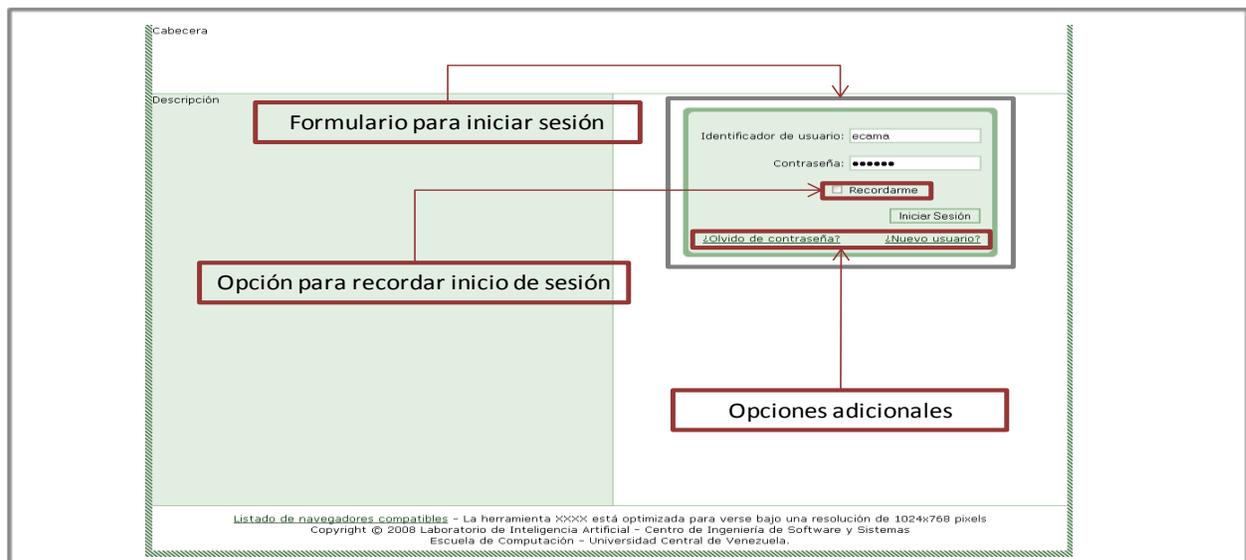
**Figura 29.** Cuadro de dialogo de confirmación para eliminar un Sitio Web.

- Se tuvo como objetivo mantener un patrón de diseño homogéneo en todas las páginas de la aplicación, en cuanto a estructura, colores utilizados, tamaños y tipos de letras.

### 2.5.8. Implementación de las Funcionalidades

En este proceso se implementan los requerimientos identificados en etapas anteriores teniendo en cuenta todo lo definido hasta el momento: tablas, tecnologías, herramientas, estructura lógica. A continuación se describen la implementación de las funcionalidades más importantes:

- **Chequear cookie:** esta acción permite verificar si el usuario en un inicio de sesión anterior, eligió almacenar una cookie para el inicio de sesión sin introducir los datos correspondientes, esta opción es conocida comúnmente como *recordarme en este equipo*. Esta acción cuenta con un parámetro configurable a través del archivo de configuración *config.properties* que es el número de días que será almacenada la cookie en el computador cliente. Si la cookie existe se muestra la página principal de la herramienta bajo el contexto del usuario. En la **Figura 30** se muestra la página de inicio de sesión.



**Figura 30.** Página de inicio de sesión de la herramienta

- **Iniciar de sesión:** la página de inicio de sesión permite que un usuario se autentique en la herramienta para comenzar a utilizar todas sus opciones. Esta funcionalidad permite validar los datos ingresados y en caso de ser correctos muestra al usuario la página principal de la herramienta bajo el contexto del

usuario. La página de inicio de sesión permite también mostrar las páginas que permiten registrar un nuevo usuario y obtener una nueva contraseña en caso de que el usuario no la recuerde.

- **Registrar nuevo usuario:** la página de registrar nuevo usuario presenta un formulario con los datos necesarios para que un usuario nuevo se registre en la herramienta. Los datos introducidos por el usuario son validados a nivel local o maquina cliente así como también del lado del servidor, estas acciones son necesarias para evitar almacenar datos erróneos en la Base de Datos de la herramienta. Luego de procesados los datos, la herramienta indica al usuario si la operación fue exitosa o no. Esta acción se realiza siguiendo el patrón arquitectónico Web MVC (modelo - vista - controlador). En la **Figura 31** se muestra la página de registro de nuevo usuario.

regresar a Inicio de sesión

**Antes de completar sus datos personales tenga en cuenta las siguientes indicaciones:**

- Los campos mostrados en el formulario no pueden quedar en blanco.
- El campo **Identificador de usuario** servirá para que usted inicie sesión en la herramienta luego de registrarse, por lo tanto, no existen dos usuarios con el mismo **Identificador de Usuario**.
- Una vez introducido todos los valores, presione el botón **Enviar Información**. Si algunos de los datos es erróneo el sistema se lo indicará para que pueda corregirlo, si este es el caso presione nuevamente el botón mencionado anteriormente.
- Luego de registrarse, inicie sesión para disfrutar de la herramienta.

Identificador de usuario:

Nombre:

Apellido:

Profesión:

Correo electrónico:

Contraseña:

Confirmación contraseña:

Pregunta secreta:

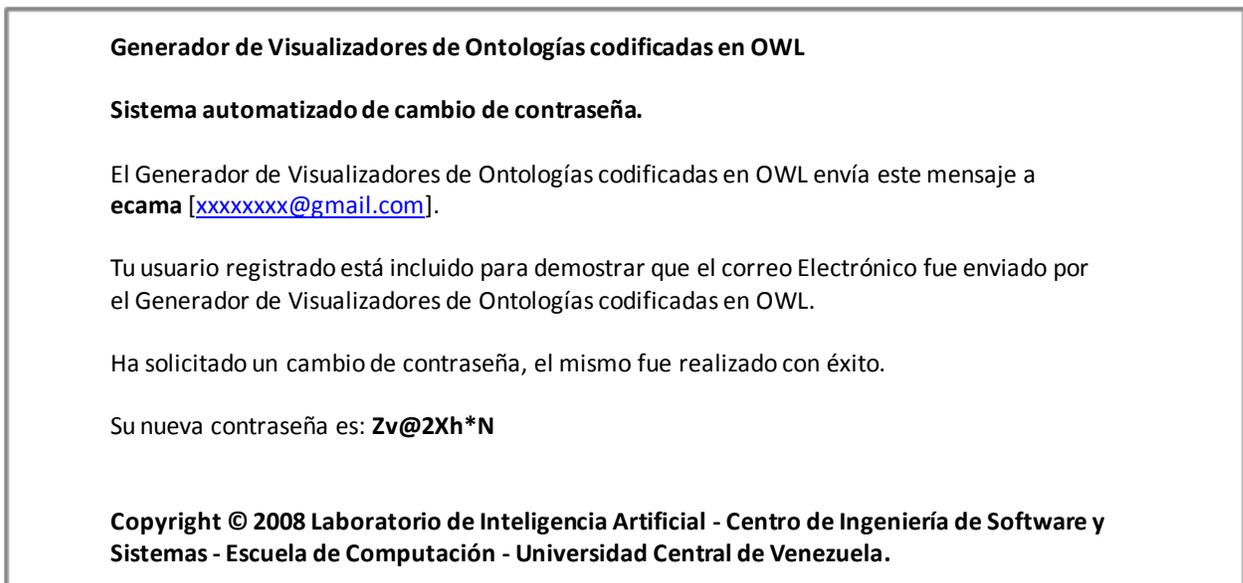
Respuesta secreta:

Listado de navegadores compatibles - La herramienta XXXX está optimizada para verse bajo una resolución de 1024x768 pixels  
Copyright © 2008 Laboratorio de Inteligencia Artificial - Centro de Ingeniería de Software y Sistemas  
Escuela de Computación - Universidad Central de Venezuela.

**Figura 31.** Página de registro de nuevo usuario de la herramienta.

- **Olvido de contraseña:** permite realizar el cambio de contraseña en caso de que el usuario no recuerde la misma. A través de la página de *olvido de contraseña* la herramienta solicita al usuario su *Identificador de usuario* y *Correo electrónico* y

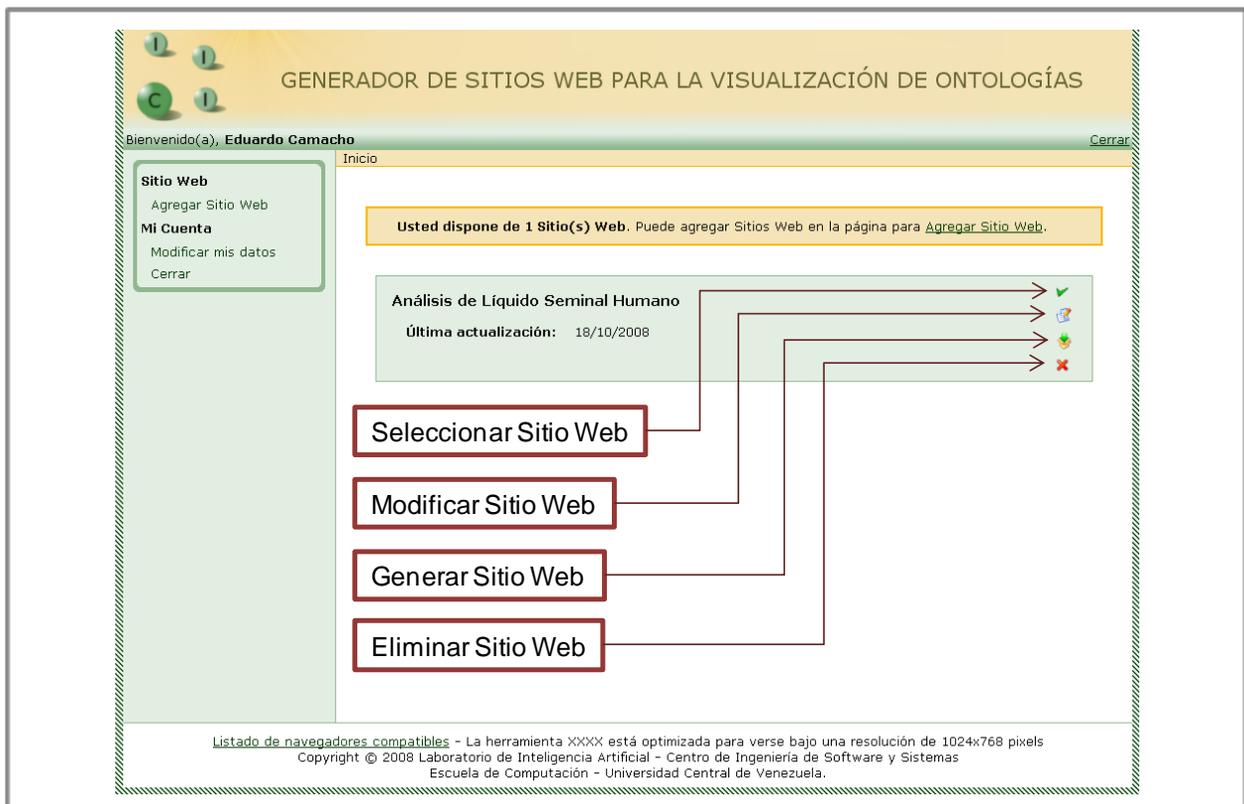
determina si estos coinciden, en caso afirmativo, muestra al usuario la *Pregunta secreta* introducida por el mismo al momento de realizar el registro como nuevo usuario. Si la respuesta introducida por el usuario es correcta la herramienta procede a generar una nueva contraseña de forma aleatoria, la envía al correo electrónico del usuario y guarda la nueva contraseña en la Base de Datos de la herramienta. La acción de envío de correo utiliza parámetros configurables, estos se encuentran en el archivo de configuración *config.properties*. En la **Figura 32** se muestra el mensaje enviado al correo del usuario una vez que el mismo solicitara el cambio de contraseña.



**Figura 32.** Mensaje enviado al usuario con una nueva contraseña.

- **Agregar Sitio Web:** a través de la página agregar Sitio Web el usuario puede registrar un nuevo Sitio Web en su cuenta de usuario. El usuario ingresa los datos por medio de un asistente. Los mismos son validados por la herramienta y en caso de que no exista errores, el Sitio Web es registrado en la Base de Datos.
- **Modificar Sitio Web:** dicha funcionalidad permite al usuario modificar un Sitio Web registrado en su cuenta en la herramienta. El usuario logra modificar los datos correspondientes al Sitio Web elegido como son: el nombre del Sitio Web, sus siglas y el estilo visual asignado al mismo al momento de crearlo.

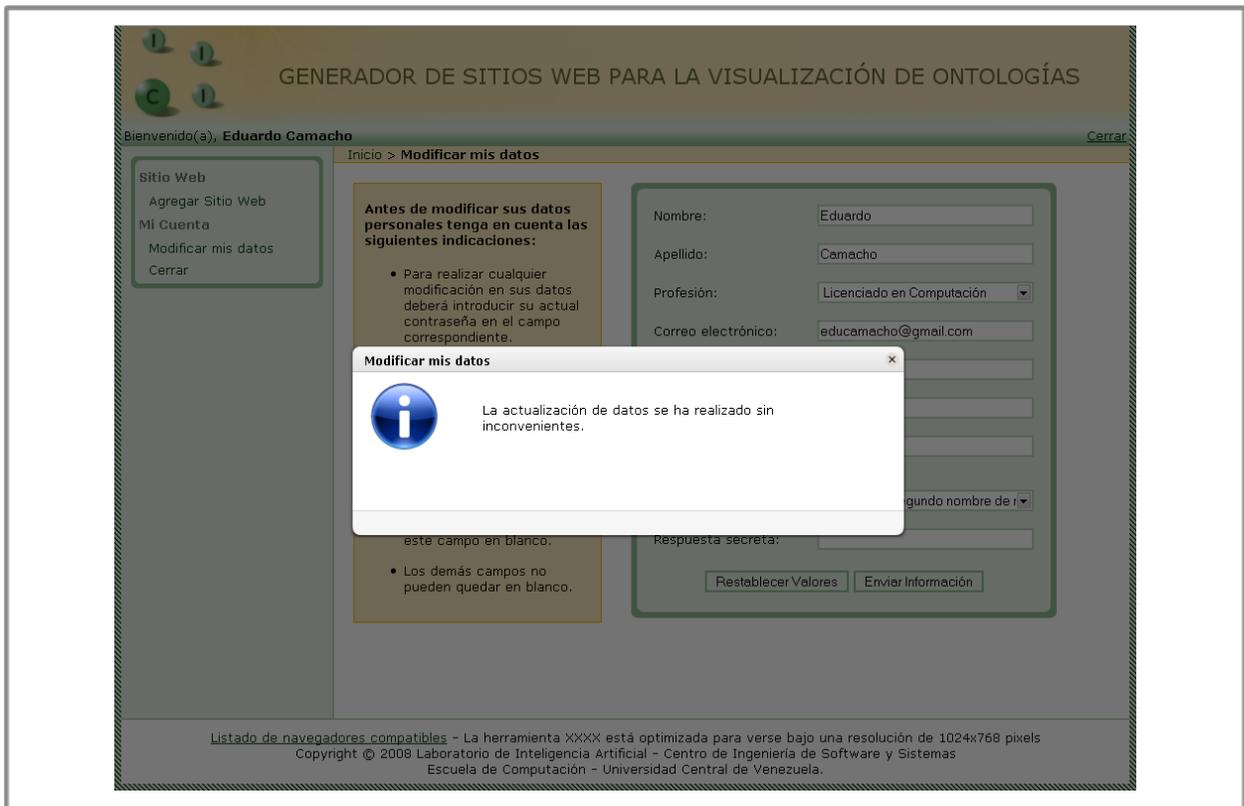
- Eliminar Sitio Web:** esta funcionalidad provee al usuario la opción de eliminar un Sitio Web ya creado. Por ser una opción irreversible, la herramienta pregunta al usuario a través de un cuadro de dialogo si realmente desea eliminar el Sitio Web seleccionado. Si el usuario ha decidido eliminar el Sitio Web permanentemente, la herramienta procede a eliminar los registros correspondientes y notifica al usuario el éxito de la operación. En la **Figura 33** se muestra la página principal una vez que el usuario ha iniciado sesión y las opciones para administrar un Sitio Web.



**Figura 33.** Opciones para administrar un Sitio Web.

- Modificar datos personales:** por medio de esta opción, el usuario puede cambiar los datos asociados a su cuenta. Como el *Identificador de usuario* es un dato univoco el mismo no puede ser modificado. La herramienta valida los datos introducidos localmente y en el servidor. Una vez que se confirman que los datos son correctos se guardan en la Base de Datos de la herramienta y se notifica al

usuario a través de un mensaje. Por razones de seguridad, el usuario de suministrar su contraseña para que los cambios puedan ser aplicados. En la **Figura 34** se muestra como el sistema indica por medio de un cuadro de dialogo que la actualización fue realizada con éxito.



**Figura 34.** Notificación exitosa de actualización de datos personales.

- **Seleccionar un Sitio Web:** una vez que el usuario ha iniciado sesión en la herramienta, se listan todos los Sitios Web creados por el mismo. El usuario tiene la opción de seleccionar algunos de los Sitios Web creados para luego modificar su contenido, ya sean las páginas que lo conforman o la ontología asociada al mismo.
- **Agregar una página al Sitio Web:** cuando el usuario selecciona un Sitio Web tiene la posibilidad de agregar cuatro tipos de páginas. La página principal con la que contará el Sitio Web es obligatoria y no se podrá agregar ninguna otra página al Sitio Web ni asociar una ontología al mismo hasta que el usuario

agregue esta página. Una vez que el usuario agrega una página principal, la herramienta muestra las opciones para agregar páginas de documentos, enlaces, imágenes y cargar una ontología al Sitio Web. También permitirá generar el Sitio Web y descargarlo en formato comprimido. El usuario cuenta con una vista previa de las páginas que conforman el Sitio Web. Al momento de agregar una nueva página, la herramienta muestra un editor de tipo *lo que ves es lo que obtienes* con la finalidad de facilitar la introducción de contenido HTML. Se utilizó el *editor de texto para Internet FCKeditor* ya que es uno de los más completos, estables y compatibles con la mayoría de los navegadores. Dicho editor está codificado en lenguaje de programación Javascript, por lo cual funciona del lado del cliente y el navegador Web utilizado por el mismo deberá tener esta opción habilitada. Es compatible con varios idiomas y es muy configurable a través de un archivo de configuración. El editor FCKeditor es compatible con varios de los lenguajes de programación Web utilizados en la actualidad, como es el caso de PHP y Java, gracias a esta compatibilidad el editor puede ser configurado desde el propio lenguaje de programación del lado del servidor así como permitir la carga de archivos y visualización de archivos cargados con anterioridad. Este editor es de código abierto.

- **Editar una página del Sitio Web:** luego de que el usuario seleccione una página, su contenido será mostrado a través del *editor de texto para Internet FCKeditor* por lo que el usuario verá exactamente como quedará la página una vez generada, el editor de texto facilita tanto la introducción de contenido como la modificación del mismo. Una vez que el usuario realice los cambios pertinentes podrá guardarlos. En la **Figura 35** puede apreciarse la página para modificar la página principal del Sitio Web.

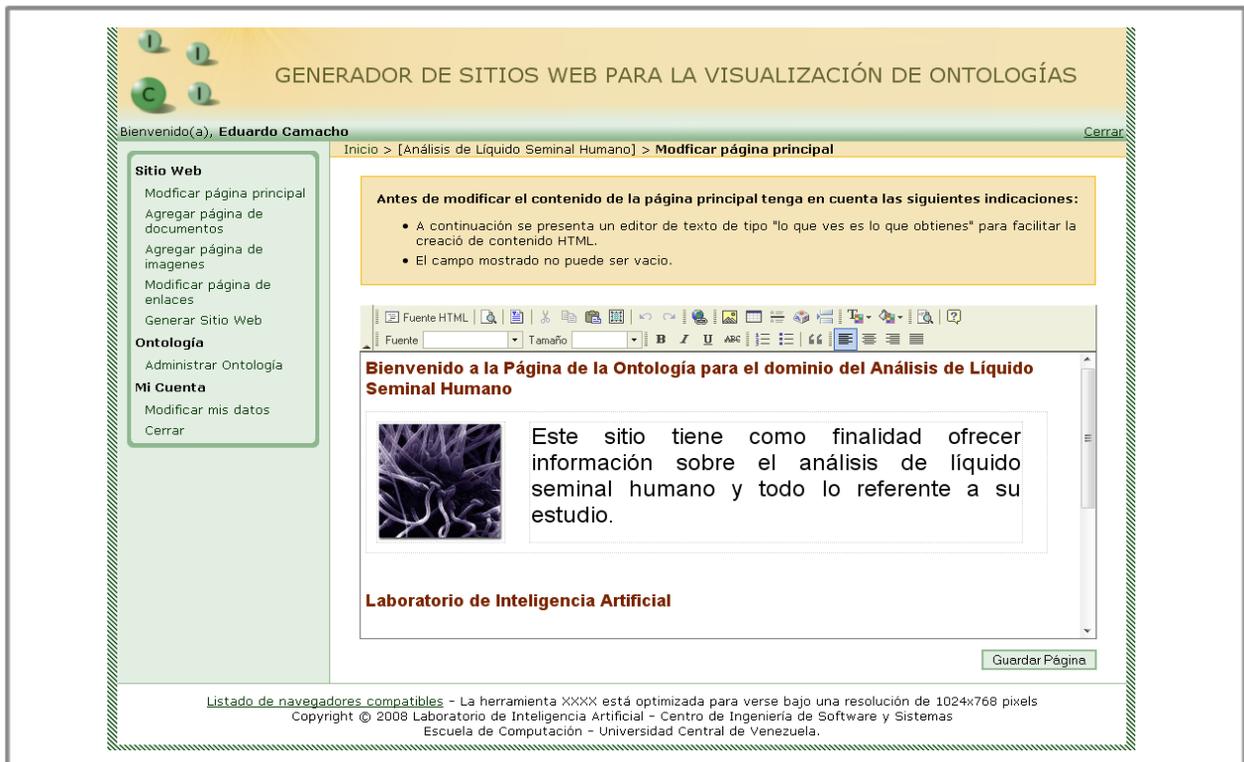


Figura 35. Página para modificar página principal de Sitio Web.

- **Eliminar una página del Sitio Web:** a través de esta opción el usuario puede eliminar una página del Sitio Web. Al ser esta opción irreversible la herramienta pregunta al usuario, por medio de un cuadro de dialogo, si realmente desea eliminar la página, al usuario confirmar la acción la página será eliminada del Sitio Web y se mostrará un mensaje de que la operación fue o no realizada con éxito. Si el usuario decide eliminar la página principal del Sitio Web, la herramienta procederá a eliminar las otras páginas que conformen el sitio como también la(s) ontología(s) asociada(s).
- **Visualizar página del Sitio Web:** la herramienta provee al usuario la posibilidad de tener una vista previa de las páginas creadas, con la finalidad de que el mismo verifique si realmente la página es de su agrado.
- **Administrar ontología:** esta opción muestra un formulario que permite al usuario cargar una ontología y asociarle una descripción. La herramienta provee la posibilidad de cargar distintas versiones de la ontología, por lo tanto, cada vez

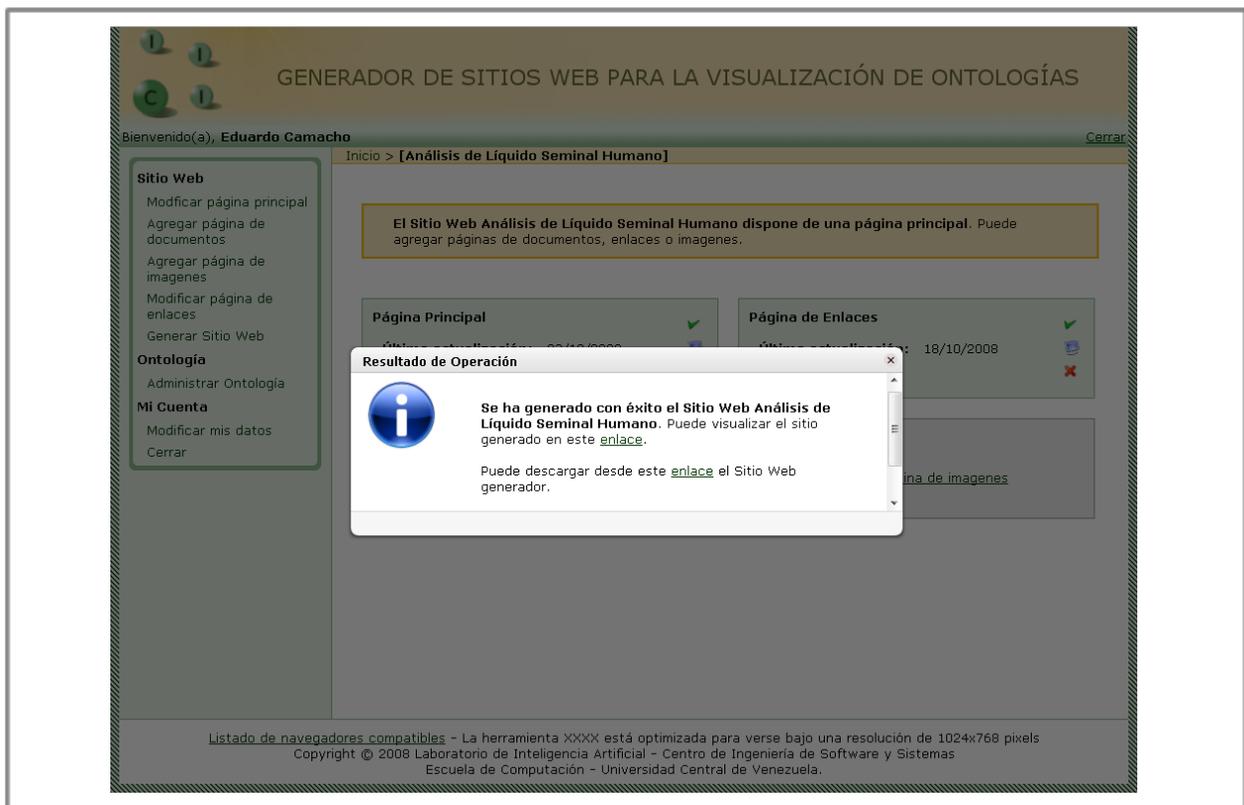
que el usuario realiza la acción de cargar una nueva versión, la herramienta se encarga de asignar un número de versión y asignar a la ontología cargada el estatus de activa dentro del Sitio Web con lo cual al momento de generar el Sitio Web esta será tomada en cuenta para generar la página de visualización de ontología. La página de Administrar ontología, muestra un listado de todas las versiones de las ontologías cargadas, identificando la ontología activa del resto. A través de esta página, el usuario puede activar cualquiera de las versiones de la ontología, de igual forma eliminarlas. En la **Figura 36** se muestra la página para administrar las ontologías asociadas al Sitio Web.



Figura 36. Página administrar ontología.

- **Generar Sitio Web:** esta funcionalidad de la herramienta permite al usuario generar el Sitio Web seleccionado. Se crean todas las páginas agregadas al Sitio Web y son comprimidas junto con la ontología activa y su visualizador, el formato de compresión utilizado es zip por su popularidad. Una vez que es generado el

Sitio Web el usuario tiene la posibilidad de visualizar el mismo o descargarlo para colocarlo en algún servidor compatible con Java. A excepción del visualizador de ontología, todas las páginas están en formato HTML por lo cual cualquier servidor debe ser capaz de mostrarlas sin mayores inconvenientes. Sin embargo, el visualizador procesa la ontología codificada en OWL a través del lenguaje de programación Java y el resultado es desplegado en una página dinámica JSP. Esta página también utiliza la librería JSON-RPC, la cual permite invocar métodos en el servidor a través de AJAX (JavaScript y XML Asíncronos) sin necesidad de recargar toda la página. En la **Figura 37** se muestra el cuadro de dialogo resultante de la operación generar Sitio Web.



**Figura 37.** Generar Sitio Web.

- **Adaptación de Código de Visualizador de Ontología:** la herramienta desarrollada dispone de un módulo que permite la visualización de ontologías codificadas en OWL el cual fue implementado en trabajos anteriores. Dicho

módulo está presente en la herramienta como en los Sitios Web generados y es de gran importancia. Trabajos anteriores han sido una base fundamental para la integración de la herramienta con el visualizador de ontologías. El trabajo “Desarrollo de una Aplicación en Ambiente Web para la visualización de una Ontología para el dominio del Análisis de Semen Humano” (Reyes, 2008) proporciona el módulo visualizador que es el resultado de una adaptación, realizado por el autor mencionado anteriormente, al módulo visualizador del trabajo “Visualizador de una Ontología de Insectos Acuáticos” (Labrador, 2007). Sin embargo, este módulo fue adaptado para su correcto funcionamiento en el contexto del generador y de cualquier otra herramienta o aplicación Web ya que presentaba ciertas carencias que restaban funcionalidad al mismo, entre ellas se puede mencionar, la inclusión de datos necesarios en las clases de Java encargadas del procesamiento de archivos ontológicos, los cuales deben obtenerse de forma dinámica. A continuación se describen las actividades realizadas para corregir el código fuente del visualizador:

- **Detección de código fuente problemático:** el módulo visualizador desarrollado en trabajos anteriores presentaba deficiencias en cuanto a la obtención de valores necesarios para realizar la carga de la ontología. Estos valores, fueron tipados directamente en las clases de Java involucradas en el proceso de carga, restando al módulo portabilidad y utilidad generalizada. De igual forma el visualizador presentaba problemas en el manejo de rutas de archivos absolutas necesario para el procesamiento del archivo ontológico, perdiendo así la característica de multiplataforma.
- **Incorporación de nuevas clases de Java:** en vista de la problemática presentada en el módulo visualizador, fue necesario incorporar clases utilitarias para solventar los problemas presentes en el visualizador. A continuación se listan las clases mencionadas:
  - **Clase para procesamiento del archivo ontológico:** el API encargado de procesar la ontología codificada en OWL requiere de

dos valores para realizar su tarea, una URI o prefijo y el nombre de la superclase que representa la ontología. Originalmente estos valores se encontraban tipados directamente en las clases de Java. Se desarrolló una clase en Java que permite obtener estos valores dinámicamente procesando el archivo ontológico a través del API Xerces implementando DOM para el parseo de archivos XML.

- **Clase para obtener rutas absolutas:** esta clase soluciona algunos de los problemas presentes en el visualizador original. Permite obtener la ruta absoluta en donde se encuentra la carpeta del visualizador en el servidor. Esta clase permite que la aplicación sea totalmente portable e independiente de la ruta en donde pueda estar alojada la aplicación.
- **Modificación de las clases encargadas de la visualización de la ontología:** con la finalidad de que el módulo de visualización de ontologías existente funcionara de forma general, tomando en cuenta ciertas condiciones en la que el archivo OWL debe estar codificado, fue necesario modificar algunas de las clases necesarias para el procesamiento del archivo ontológico. Las mismas realizaban acciones dependientes de valores tipados directamente en ellas u otras clases. Por lo tanto, sólo trabajan de forma correcta en el contexto de la ontología utilizada y para visualizar alguna otra ontología era necesario cambiar dichos valores tipados. Las clases involucradas fueron adaptadas para tomar los valores necesario de manera dinámica, y así generalizar el visualizador.

## CONCLUSIONES Y RECOMENDACIONES

Una vez concluido el desarrollo de la herramienta para generar Sitios Web que permitan la visualización de ontologías codificadas en OWL, el cual estuvo basado en un estudio sistemático del problema y de los objetivos que se deseaban obtener, se puede concluir que se logró identificar los requerimientos funcionales con los que debería contar dicha herramienta y en base a estos se diseñó e implemento la misma.

Se realizó una optimización del módulo que permite visualizar ontologías de la *Aplicación en Ambiente Web para la visualización de una ontología para el Dominio del Análisis de Semen Humano*. Se logró generalizar el módulo antes mencionado para cargar ontologías codificadas en OWL bajo cualquier dominio y que cumplan ciertas condiciones. Este módulo se integra con la herramienta desarrollada así como también los Sitios Web generados a través de la misma.

La herramienta desarrollada permite generar Sitios Web para la visualización de ontologías, sin embargo, puede ser utilizada para generar sitios que abarquen otros contextos ya que ofrece la funcionalidad de crear páginas muy comunes en algunos Sitios Web como es el caso de página principal y páginas de enlaces de interés, de documentos y de imágenes (galerías).

La herramienta fue probada en diferentes plataformas para verificar el correcto funcionamiento de la aplicación y sus componentes, asegurando de esta forma, que el producto obtenido posee la característica de ser multiplataforma. Las pruebas fueron realizadas en los Sistemas Operativos Windows XP y Linux Ubuntu. De igual forma, la herramienta se probó en distintos contenedores Web.

En el transcurso del desarrollo de la herramienta se presentaron algunos inconvenientes o dificultades, entre estos, se tiene que el módulo visualizador de ontologías desarrollado previamente, presentaba algunos fallos debido a la inclusión de valores necesarios para cargar la ontología dentro de las clases de Java, restando el concepto de portabilidad y multiplataforma al mismo. Estos valores pueden ser cargados directamente de la ontología pero debido a la poca documentación existente

acerca del API JENA, fue necesario buscar métodos alternos para solucionar el inconveniente. Otra de las dificultades durante el desarrollo de la herramienta, fue mantener la compatibilidad con los tres navegadores Web establecidos.

Algunas recomendaciones para trabajos futuros son:

- Realizar un módulo que permita la generación de hojas de estilos en cascada e integrarlo al generador. Con la finalidad, de proveer al usuario una mayor versatilidad en cuanto a plantillas de estilos visuales y obtener así una herramienta más personalizable para crear Sitios Web.
- Continuar con la práctica de programación empleada para que la herramienta mantenga el agregado de ser multiplataforma.
- Cada nuevo módulo desarrollado, deberá ser probado en varios Sistemas Operativos y contenedores Web para asegurar su compatibilidad. Esta recomendación debe ser tomada en cuenta sin omisión ya que algunas de las funcionalidades del lenguaje de programación Java tiene distinto comportamiento en diferentes Sistemas Operativos.
- Mantener la compatibilidad, como mínimo, con los navegadores Web en los que fue probada la herramienta.

**REFERENCIAS**

- ALTOVA. (2008) Altova Company. Disponible en <http://www.altova.com/>
- BBN TECHNOLOGIES. (2008) SWeDE. BBN Technologies. Disponible en <http://owl-eclipse.projects.semwebcentral.org/>
- BORGES, C.; RIVERO, A. (2006) *GENCI: Una herramienta para la presencia en Internet de los Centros de Investigación*. Trabajo Especial de Grado. Escuela de Computación. Facultad de Ciencias. Universidad Central de Venezuela.
- BORST, W. (1997) *Construction of engineering ontologies for knowledge sharing and reuse*. PhD Thesis. University of Twente, The Netherlands, Enschede.
- CO-ODE. (2008) Bio Health Informatics Group. Department of Computer Science. The University of Manchester. Disponible en <http://www.co-ode.org/ontologies/>
- DUQUE, N.; CHAVARRO, J.; MORENO, R. (2006) *EVOLUCIÓN DE LOS LENGUAJES UTILIZADOS EN LA CONSTRUCCIÓN DE LA WEB SEMÁNTICA* [en línea]. [Consulta: 09 nov. 2007]. Disponible en <http://www.utp.edu.co/php/revistas/ScientiaEtTechnica/docsFTP/11551381-386.pdf>
- FOSSI, A.; OLIVEIRA, C. (2006) *Generador de Sitios Web de Alcaldías*. Trabajo Especial de Grado. Escuela de Computación. Facultad de Ciencias. Universidad Central de Venezuela.
- GARCÍA, E. (2005) *Aplicación de las Ontologías para la Representación del Conocimiento* [en línea]. [Consulta: 26 ene. 2008]. Disponible en <http://bd.ub.es/isko2005/garcia.pdf>
- GRUBER, T. (1993) *Toward Principles for the Design of Ontologies Used for Knowledge Sharing*. Technical Report KSL-93-04, Knowledge Systems Laboratory, Stanford University, CA.

- HURTADO, D. (2006) *Desarrollo de un Módulo Ontológico para las Consultas Avanzadas de Información de la Biblioteca Digital de la Escuela de Ingeniería de Sistemas y Computación de la Universidad del Valle* [en línea]. [Consulta: 18 oct. 2007]. Disponible en [http://dipahurb.googlepages.com/AnteProyecto\\_DianaPaolaHurtado.pdf](http://dipahurb.googlepages.com/AnteProyecto_DianaPaolaHurtado.pdf)
- LABRADOR, Y. (2007) *Desarrollo de una Aplicación en Ambiente Web para la visualización de una Ontología de Insectos Acuáticos*. Trabajo Especial de Grado. Escuela de Computación. Facultad de Ciencias. Universidad Central de Venezuela
- LAMARCA, M. (2007a) *Ontologías* [en línea]. [Consulta: 18 oct. 2007]. Disponible en <http://www.hipertexto.info/documentos/ontologias.htm>
- LAMARCA, M. (2007b) *RDF* [en línea]. [Consulta: 03 nov. 2007]. Disponible en <http://www.hipertexto.info/documentos/rdf.htm>
- LAMARCA, M. (2007c) *OWL* [en línea]. [Consulta: 09 nov. 2007]. Disponible en <http://www.hipertexto.info/documentos/owl.htm>
- MONSERRATT, R.; RODRÍGUEZ E. (2006) *Generador de Revistas Electrónicas: Una herramienta para publicar en la sociedad de la información*. Trabajo Especial de Grado. Escuela de Computación. Facultad de Ciencias. Universidad Central de Venezuela.
- MYERS, B y Rosson, M. (1996). Survey on user interface programming. In Human factors in computing systems. SIGCHI-ACM.
- NIELSEN J. (1990) *Ten Usability Heuristics* [en línea]. [Consulta: 25 ago. 2008]. Disponible en: [http://www.useit.com/papers/heuristic/heuristic\\_list.html](http://www.useit.com/papers/heuristic/heuristic_list.html).
- NÚÑEZ, H; RAMOS, E. (2007) *ONTOLOGÍAS: componentes, metodologías, lenguajes, herramientas y aplicaciones*. Laboratorio de Inteligencia Artificial -LIA-. Escuela de Computación. Facultad de Ciencias. Universidad Central de Venezuela.

- ONTOPRISE. (2008) Ontoprise GmbH. Disponible en <http://www.ontoprise.de/>
- PEDRAZA, F. (2003) *Generador de Sitios Web de Asignaturas*. Trabajo Especial de Grado. Escuela de Computación. Facultad de Ciencias. Universidad Central de Venezuela.
- PEREIRA, Y. (2006) *Ontología en ambiente Web para el dominio del Análisis de Líquido Seminal Humano*. Trabajo Especial de Grado. Escuela de Computación. Facultad de Ciencias. Universidad Central de Venezuela.
- PROTÉGÉ. (2008) Stanford Center for Biomedical Informatics Research. The Stanford University School of Medicine. Stanford University. Disponible en <http://protege.stanford.edu/>
- REYES, E. (2008) *Desarrollo de una Aplicación en Ambiente Web para la visualización de una Ontología para el dominio del Análisis de Semen Humano*. Trabajo Especial de Grado. Escuela de Computación. Facultad de Ciencias. Universidad Central de Venezuela
- SHNEIDERMAN B. (1998) *Designing the user interface*. Addison-Wesley.
- STEVE, G, A. Gangemi, D. Pisanelli. (1998) *Integrating Medical Terminologies with ONIONS Methodology*.

## **BIBLIOGRAFÍA CONSULTADA**

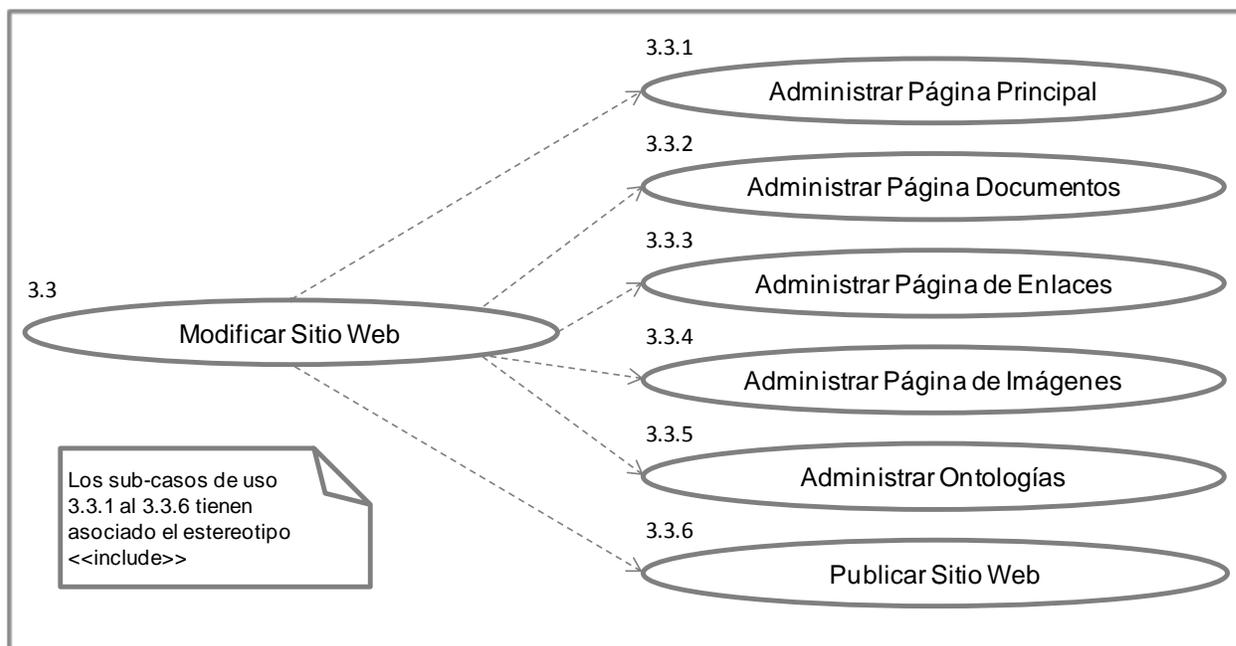
- ÁLVAREZ, M. (2004) *BÚSQUEDAS MEDIANTE ONTOLOGÍAS* [en línea]. [Consulta: 18 oct. 2007]. Disponible en [http://es.geocities.com/busquedas\\_internet/mediante\\_ontologias.html](http://es.geocities.com/busquedas_internet/mediante_ontologias.html)
- ARANO, S. *La ontología: una zona de interacción entre la Lingüística y la Documentación* [en línea]. [Consulta: 18 oct. 2007]. Disponible en <http://www.hipertext.net/web/pag220.htm>

- 
- AYLLÓN, J. (2007) *Metadatos y documentos XML/RDF para Recuperación* [en línea]. [Consulta: 03 nov. 2007]. Disponible en <http://metadatos-xml-rdf.awardspace.com/rdf.html>
- CODINA, L.; ROVIRA, C. (2006) *La Web Semántica* [en línea]. [Consulta: 18 oct. 2007]. Disponible en [http://eprints.rclis.org/archive/00008637/01/web\\_semantica\\_.pdf](http://eprints.rclis.org/archive/00008637/01/web_semantica_.pdf)
- DORAY, A. (2007) *Programación con Apache Struts*. Ediciones Anaya Multimedia.
- GARCÍA, F. *Web Semántica y Ontologías* [en línea]. [Consulta: 18 oct. 2007]. Disponible en <http://zarza.usal.es/~fgarcia/doctorado/iuce/WSemantica.pdf>
- Generadores* [en línea]. [Consulta: 08 dic. 2007]. Disponible en <http://baobab.ciens.ucv.ve/#>
- HJELM, J. *Creating the semantic web in RDF*. New York: Wiley Computer Publishing; John Wiley & sons, 2001.
- INTRODUCCIÓN A LAS ONTOLOGÍAS* [en línea]. [Consulta: 18 oct. 2007]. Disponible en [http://es.geocities.com/ontologias\\_y\\_tesauros/introduccion\\_a\\_las\\_ontologias.htm](http://es.geocities.com/ontologias_y_tesauros/introduccion_a_las_ontologias.htm)
- ISYS. (2008) Centro de Ingeniería de Software y Sistemas. Escuela de Computación. Facultad de Ciencias. Universidad Central de Venezuela. Disponible en <http://baobab.ciens.ucv.ve/>
- LOZANO, A. (2001) *Ontologías en la Web Semántica* [en línea]. [Consulta: 18 oct. 2007]. Disponible en <http://www.informandote.com/jornadasIngWEB/articulos/jiw02.pdf>
- MARTIN, A. (2007) *AJAX en J2EE*. Primera edición. Alfaomega Grupo Editor.
- MÉNDEZ, E. (2001) *Resource Description Framework (RDF). Especificación del Modelo y la Sintaxis* [en línea]. [Consulta: 03 nov. 2007]. Disponible en <http://sidar.org/recur/desdi/traduc/es/rdf/rdfesp.htm>

- 
- MÉNDEZ, M. *RDF: UN MODELO DE METADATOS FLEXIBLE PARA LAS BIBLIOTECAS DIGITALES DEL PRÓXIMO MILENIO* [en línea]. [Consulta: 03 nov. 2007]. Disponible en <http://www.cobdc.org/jornades/7JCD/1.pdf>
- MILLER, E. *An introduction to the Resource Description Framework* [en línea]. D-Lib Magazine, May 1998. [Consulta: 03 nov. 2007]. Disponible en: <http://www.dlib.org/dlib/may98/miller/05miller.html>
- Ontología (informática)* [en línea]. [Consulta: 18 oct. 2007]. Disponible en [http://es.wikipedia.org/wiki/Ontolog%C3%ADa\\_\(Inform%C3%A1tica\)](http://es.wikipedia.org/wiki/Ontolog%C3%ADa_(Inform%C3%A1tica))
- Ontologías de Metadatos y su fusión mapeados de ontologías* [en línea]. [Consulta: 18 oct. 2007]. Disponible en <http://ontologias.bravehost.com/metadatos.html>
- PÉREZ, C.; MORENO, A. (2000) *Modelado conceptual para una terminología basada en el conocimiento* [en línea]. [Consulta: 18 oct. 2007]. Disponible en <http://www.ritem.net/actes/7simposio/perez.htm>
- PÉREZ, C. (2002) *Definición de ontología como especificación del conocimiento* [en línea]. [Consulta: 18 oct. 2007]. Disponible en <http://elies.rediris.es/elies18/531.html>
- RAMÍREZ, Z. *Las ontologías como herramienta en la Gestión del Conocimiento* [en línea]. [Consulta: 18 oct. 2007]. Disponible en <http://www.congreso-info.cu/UserFiles/File/Info/Info2006/Ponencias/208.pdf>
- RODRÍGUEZ, S. *WEB SEMÁNTICA (PARTE I): VISTA GENERAL DE RDF* [en línea]. [Consulta: 03 nov. 2007]. Disponible en [http://ingenieria.url.edu.gt/boletin/URL\\_05\\_SIS02.pdf](http://ingenieria.url.edu.gt/boletin/URL_05_SIS02.pdf)
- SAMPER, J. (2005) *Ontologías para servicios Web Semánticos de información de tráfico: descripción y herramientas de explotación* [en línea]. Disponible en [http://www.tdx.cesca.es/TESIS\\_UV/AVAILABLE/TDX-0628106-085805//SAMPER.pdf](http://www.tdx.cesca.es/TESIS_UV/AVAILABLE/TDX-0628106-085805//SAMPER.pdf)

- SENSO, J. (2007) *Resource Description Framework* [en línea]. [Consulta: 03 nov. 2007]. Disponible en <http://www.hipertext.net/web/pag236.htm>
- SOBRE ONTOLOGÍAS [en línea]. [Consulta: 18 oct. 2007]. Disponible en [http://es.geocities.com/recupdeinformacion\\_ontologias/sobreontolgias.htm](http://es.geocities.com/recupdeinformacion_ontologias/sobreontolgias.htm)
- The World Wide Web Consortium - W3C. (2004) OWL Web Ontology Language Overview [en línea]. [Consulta: 09 nov. 2007]. Disponible en <http://www.w3.org/TR/owl-features/>
- TRAMULLAS, J. *Agentes y ontologías para el tratamiento de información: clasificación y recuperación en Internet* [en línea]. [Consulta: 18 oct. 2007]. Disponible en <http://tramullas.com/papers/isko99.pdf>
- VALENCIA, R. (2005) *La siguiente versión de la web, la web semántica* [en línea]. [Consulta: 18 oct. 2007]. Disponible en [http://www.cii-murcia.es/informas/ene05/articulos/La\\_siguiente\\_version\\_de\\_la\\_web\\_la\\_web\\_semantica.html](http://www.cii-murcia.es/informas/ene05/articulos/La_siguiente_version_de_la_web_la_web_semantica.html)

## ANEXO 1: CASOS DE USO DE NIVEL 2.



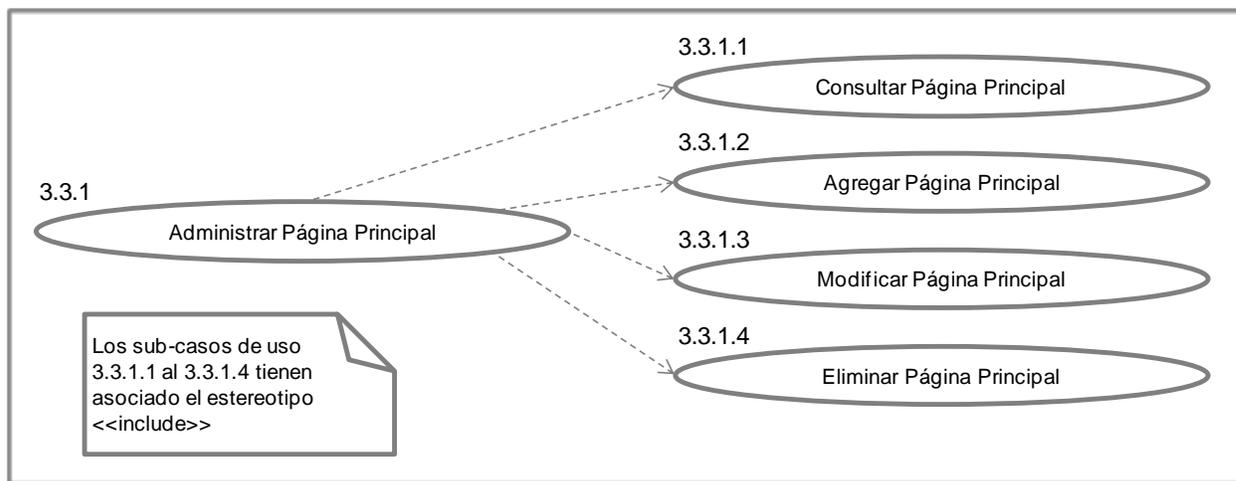
**Figura 38.** Diagrama de CU – nivel 2 – sub-módulo modificar Sitio Web.

- **Nombre:** *Administrar Página de Principal.*
- **Identificador:** CU 3.3.1.
- **Descripción:** permite al usuario realizar las actividades concernientes a la publicación de la página principal del Sitio Web creado.
- **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Sitio Web*.
- **Pos-condición:** se añade, modifica o elimina la página principal del Sitio Web creado.
  
- **Nombre:** *Administrar Página de Documentos.*
- **Identificador:** CU 3.3.2.
- **Descripción:** permite al usuario realizar las actividades concernientes a la publicación de una página de documentos en el Sitio Web creado.

- **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Sitio Web*. Debe existir una página principal
- **Pos-condición:** se añade, modifica o elimina la página de documentos del Sitio Web creado.
  
- **Nombre: *Administrar Página de Enlaces*.**
- **Identificador:** CU 3.3.3.
- **Descripción:** permite al usuario realizar las actividades concernientes a la publicación de una página de enlaces en el Sitio Web creado.
- **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Sitio Web*. Debe existir una página principal.
- **Pos-condición:** se añade, modifica o elimina la página de enlaces de interés del sitio Web creado.
  
- **Nombre: *Administrar Página de Imágenes*.**
- **Identificador:** CU 3.3.4.
- **Descripción:** permite al usuario realizar las actividades concernientes a la publicación de una página de imágenes en el Sitio Web creado.
- **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Sitio Web*. Debe existir una página principal.
- **Pos-condición:** se añade, modifica o elimina la página de imágenes del Sitio Web creado.
  
- **Nombre: *Administrar Ontología*.**
- **Identificador:** CU 3.3.5.
- **Descripción:** permite al usuario realizar las actividades concernientes a la publicación de una ontología en el sitio Web creado.

- **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Sitio Web*. Debe existir una página principal.
- **Pos-condición:** se añade, modifica o elimina una ontología del sitio Web creado. El sistema permite tener más de una versión de la ontología y activar cual será la mostrada en el sitio Web creado.
  
- **Nombre:** *Publicar Sitio Web*.
- **Identificador:** CU 3.3.6.
- **Descripción:** permite al usuario publicar el sitio Web creado. El sistema presentará dos opciones: la primera permitirá visualizar el sitio Web; la segunda opción permitirá descargarse la estructura del sitio Web junto a todos los archivos relacionados en formato comprimido para una mejor manipulación a la hora de colocar el sitio Web en un contenedor Web.
- **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Sitio Web*. Debe existir una página principal.
- **Pos-condición:** el usuario logra visualizar el sitio Web creado o descargarse el mismo en un formato comprimido.

## ANEXO 2: CASOS DE USO DE NIVEL 3.



**Figura 39.** Diagrama de CU – nivel 3 – sub-módulo administrar página principal.

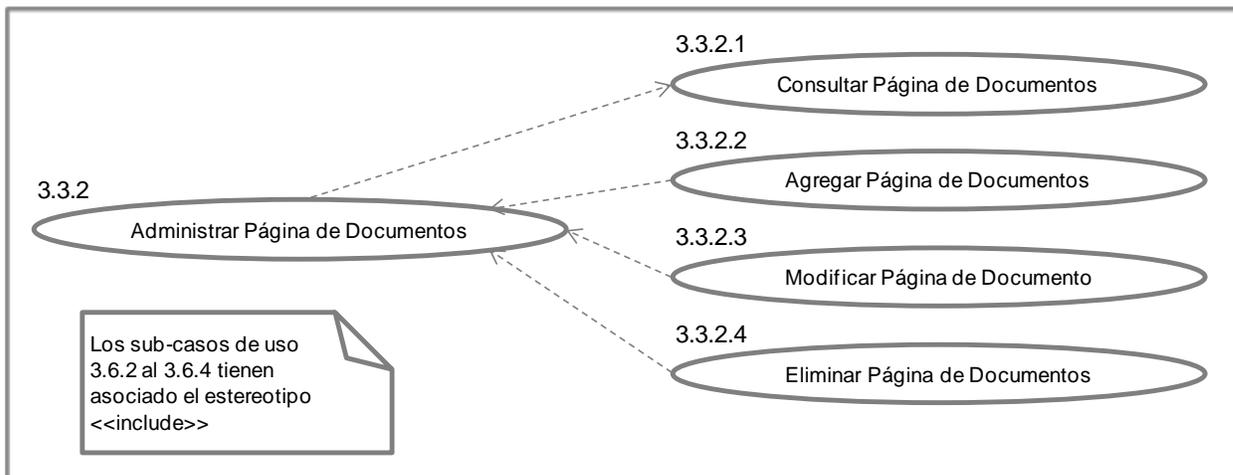
- **Descripción de los casos de uso**

- **Nombre:** *Consultar Página Principal.*
  - **Identificador:** CU 3.3.1.1.
  - **Descripción:** permite visualizar el contenido de la página principal del Sitio Web creado por el usuario.
  - **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Página Principal*. Debe existir una página principal.
  - **Pos-condición:** Es mostrado al usuario la información concerniente a la página de principal del Sitio Web, en caso de no existir contenido, el sistema indica dicha situación a través de un mensaje y provee la opción para crear una página de este tipo y agregar el contenido deseado.
- **Nombre:** *Agregar Página Principal.*
  - **Identificador:** CU 3.3.1.2.
  - **Descripción:** permite al usuario agregar una página principal a un Sitio Web y añadir información concerniente a la misma.

- **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Página Principal*. Debe existir el Sitio Web al que se desea agregar una página principal.
- **Pos-condición:** se le muestra al usuario un formulario con el valor requerido por el sistema para agregar una nueva página principal junto con su contenido.
- **Curso básico de acción:**
  1. El usuario selecciona la opción *Agregar Página Principal*.
  2. Se despliega una interfaz para introducir los valores correspondientes.
  3. El usuario hace clic en el botón *Agregar Página Principal*.
  4. El sistema realiza una primera validación de los valores introducidos, con la finalidad de determinar que el usuario no deje campos en blanco. [Curso alternativo A].
  5. El sistema recibe los datos introducidos por el usuario y valida que los mismo sean correctos.
  6. Si los datos son correctos, los mismos son almacenados en la Base de Datos y le es notificado al usuario que la operación solicitada fue realizada con éxito. [Curso alternativo B].
  7. Finaliza el caso de uso.
- **Cursos alternos:**
  - **Curso alternativo A:** el usuario no introdujo los datos necesarios.
    - **A.1:** el sistema determina que el usuario no introdujo ningún valor en el campo necesario.
    - **A.2:** el sistema informa del error ocurrido a través de un mensaje.
    - **A.3:** el control de flujo regresa al punto 2 del curso básico de acción.
  - **Curso alternativo B:** la información correspondiente no pudo ser almacenada en la Base de Datos.

- **B.1:** el sistema determina que la operación solicitada no pudo ser realizada.
  - **B.2:** el sistema informa del error ocurrido a través de un mensaje.
  - **B.3:** el control de flujo regresa al punto 2 del curso básico de acción.
- 
- **Nombre: *Modificar Página de Principal.***
    - **Identificador:** CU 3.3.1.3.
    - **Descripción:** permite al usuario modificar el contenido asociado a la página principal.
    - **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Página Principal.* Debe existir una página principal.
    - **Pos-condición:** se le muestra al usuario un formulario con el contenido de la página principal.
    - **Curso básico de acción:**
      1. El usuario selecciona la opción *Modificar Página Principal.*
      2. Se despliega una interfaz con el contenido de la página principal, la cual permite modificarlo.
      3. El usuario hace clic en el botón *Modificar Página Principal.*
      4. El sistema realiza una primera validación del valor introducido, con la finalidad de determinar que el usuario no deje el campo en blanco. [Curso alterno A].
      5. El sistema recibe los datos introducidos por el usuario y valida que los mismo sean correctos.
      6. Si los datos son correctos, el sistema actualiza el campo correspondiente en la Base de Datos y le es notificado al usuario que la operación solicitada fue realizada con éxito. [Curso alterno B].
      7. Finaliza el caso de uso.

- **Cursos alternos:**
  - **Curso alternativo A:** la información correspondiente no pudo ser almacenada en la Base de Datos.
    - **A.1:** el sistema determina que el usuario no introdujo ningún valor en el campo necesario.
    - **A.2:** el sistema informa del error ocurrido a través de un mensaje.
    - **A.3:** el control de flujo regresa al punto 2 del curso básico de acción.
  - **Curso alternativo B:** no logra realizar la operación solicitada.
    - **B.1:** el sistema determina que la operación solicitada no pudo ser realizada.
    - **B.2:** el sistema informa del error ocurrido a través de un mensaje.
    - **B.3:** el control de flujo regresa al punto 2 del curso básico de acción.
  
- **Nombre:** *Eliminar Página Principal.*
- **Identificador:** CU 3.3.1.4.
- **Descripción:** permite al usuario eliminar la página principal junto con su contenido.
- **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Página Principal*; Debe existir una página principal.
- **Pos-condición:** la página principal y su contenido es eliminado.
- **Curso básico de acción:**
  1. El usuario selecciona la opción *Eliminar Página Principal.*
  2. Se elimina el registro asociado a la página en la Base de Datos.
  3. Finaliza el caso de uso.



**Figura 40.** Diagrama de CU – nivel 3 – sub-módulo administrar página de documentos.

- **Descripción de los casos de uso**

- **Nombre: Consultar Página de Documentos.**

- **Identificador:** CU 3.3.2.1.
- **Descripción:** permite visualizar el contenido de la página de documentos del Sitio Web creado por el usuario.
- **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Página de Documentos*. Debe existir una página de documentos.
- **Pos-condición:** Es mostrado al usuario la información concerniente a la página de documentos del Sitio Web, en caso de no existir contenido, el sistema indica dicha situación a través de un mensaje y provee la opción para crear una página de este tipo y agregar el contenido deseado.

- **Nombre: Agregar Página de Documentos.**

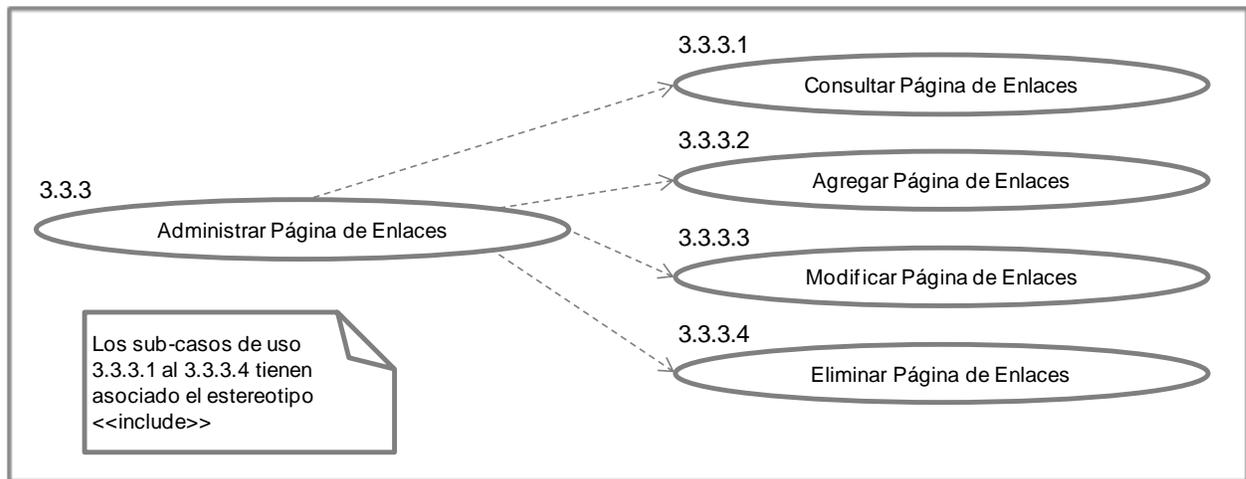
- **Identificador:** CU 3.3.2.2.
- **Descripción:** permite al usuario agregar una página de documentos a un Sitio Web y añadir información concerniente a la misma.

- **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Página de Documentos*. Debe existir una página principal.
- **Pos-condición:** se le muestra al usuario un formulario con el valor requerido por el sistema para agregar una nueva página de documentos junto con su contenido.
- **Curso básico de acción:**
  1. El usuario selecciona la opción *Agregar Página de Documentos*.
  2. Se despliega una interfaz para introducir los valores correspondientes.
  3. El usuario hace clic en el botón *Agregar Página de Documentos*.
  4. El sistema realiza una primera validación de los valores introducidos, con la finalidad de determinar que el usuario no dejara campos en blanco. [Curso alternativo A].
  5. El sistema recibe los datos introducidos por el usuario y valida que los mismo sean correctos.
  6. Si los datos son correctos, los mismos son almacenados en la Base de Datos y le es notificado al usuario que la operación solicitada fue realizada con éxito. [Curso alternativo B].
  7. Finaliza el caso de uso.
- **Cursos alternos:**
  - **Curso alternativo A:** el usuario no introdujo los datos necesarios.
    - **A.1:** el sistema determina que el usuario no introdujo ningún valor en el campo necesario.
    - **A.2:** el sistema informa del error ocurrido a través de un mensaje.
    - **A.3:** el control de flujo regresa al punto 2 del curso básico de acción.
  - **Curso alternativo B:** la información correspondiente no pudo ser almacenada en la Base de Datos.

- **B.1:** el sistema determina que la operación solicitada no pudo ser realizada.
  - **B.2:** el sistema informa del error ocurrido a través de un mensaje.
  - **B.3:** el control de flujo regresa al punto 2 del curso básico de acción.
- 
- **Nombre: *Modificar Página de Documentos.***
    - **Identificador:** CU 3.3.2.3.
    - **Descripción:** permite al usuario modificar el contenido asociado a una página de documentos.
    - **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Página de Documentos*. Debe existir una página principal y una página de documentos.
    - **Pos-condición:** se le muestra al usuario un formulario con el contenido de la página de documentos.
    - **Curso básico de acción:**
      1. El usuario selecciona la opción *Modificar Página de Documentos*.
      2. Se despliega una interfaz con el contenido de la página de documentos, la cual permite modificarlo.
      3. El usuario hace clic en el botón *Modificar Página de Documentos*.
      4. El sistema realiza una primera validación del valor introducido, con la finalidad de determinar que el usuario no deje el campo en blanco. [Curso alternativo A].
      5. El sistema recibe los datos introducidos por el usuario y valida que los mismo sean correctos.
      6. Si los datos son correctos, el sistema actualiza el campo correspondiente en la Base de Datos y le es notificado al usuario que la operación solicitada fue realizada con éxito. [Curso alternativo B].

7. Finaliza el caso de uso.
- **Cursos alternos:**
    - **Curso alternativo A:** la información correspondiente no pudo ser almacenada en la Base de Datos.
      - **A.1:** el sistema determina que el usuario no introdujo ningún valor en el campo necesario.
      - **A.2:** el sistema informa del error ocurrido a través de un mensaje.
      - **A.3:** el control de flujo regresa al punto 2 del curso básico de acción.
    - **Curso alternativo B:** algunos de los campos tiene un valor nulo o no respeta los formatos establecidos.
      - **B.1:** el sistema determina que la operación solicitada no pudo ser realizada.
      - **B.2:** el sistema informa del error ocurrido a través de un mensaje.
      - **B.3:** el control de flujo regresa al punto 2 del curso básico de acción.
  - **Nombre:** *Eliminar Página de Documentos.*
  - **Identificador:** CU 3.3.2.4.
  - **Descripción:** permite al usuario eliminar la página de documentos junto con su contenido.
  - **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Página de Documentos*; Debe existir una página principal y una página de documentos.
  - **Pos-condición:** la página de documentos y su contenido es eliminado.
  - **Curso básico de acción:**
    1. El usuario selecciona la opción *Eliminar Página de Documento.*
    2. Se elimina el registro asociado a la página en la Base de Datos.

## 3. Finaliza el caso de uso.



**Figura 41.** Diagrama de CU – nivel 3 – sub-módulo administrar enlaces.

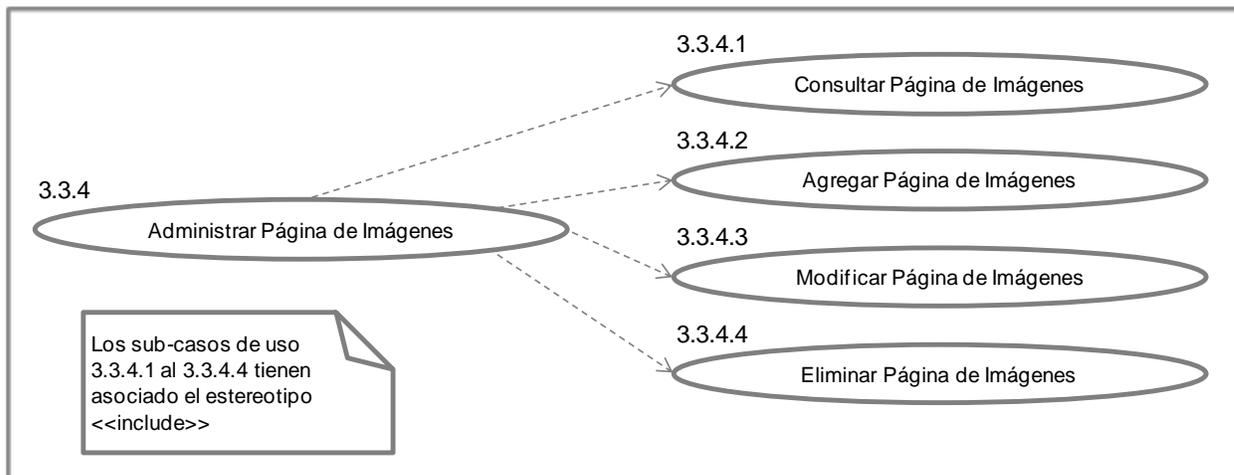
- **Descripción de los casos de uso**

- **Nombre:** *Consultar Página de Enlaces.*
  - **Identificador:** CU 3.3.3.1.
  - **Descripción:** permite visualizar el contenido de la página de enlaces del Sitio Web creado por el usuario.
  - **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Página de Enlaces*. Debe existir una página de enlaces.
  - **Pos-condición:** Es mostrado al usuario la información concerniente a la página de enlaces del Sitio Web, en caso de no existir contenido, el sistema indica dicha situación a través de un mensaje y provee la opción para crear una página de este tipo y agregar el contenido deseado.
- **Nombre:** *Agregar Página de Enlaces.*
  - **Identificador:** CU 3.3.3.2.
  - **Descripción:** permite al usuario agregar una página de enlaces a un Sitio Web y añadir información concerniente a la misma.

- **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Página de Enlaces*. Debe existir una página principal.
- **Pos-condición:** se le muestra al usuario un formulario con el valor requerido por el sistema para agregar una nueva página de enlaces junto con su contenido.
- **Curso básico de acción:**
  1. El usuario selecciona la opción *Agregar Página de Enlaces*.
  2. Se despliega una interfaz para introducir los valores correspondientes.
  3. El usuario hace clic en el botón *Agregar Página de Enlaces*.
  4. El sistema realiza una primera validación de los valores introducidos, con la finalidad de determinar que el usuario no dejara campos en blanco. [Curso alterno A].
  5. El sistema recibe los datos introducidos por el usuario y valida que los mismo sean correctos.
  6. Si los datos son correctos, los mismos son almacenados en la Base de Datos y le es notificado al usuario que la operación solicitada fue realizada con éxito. [Curso alterno B].
  7. Finaliza el caso de uso.
- **Cursos alternos:**
  - **Curso alterno A:** el usuario no introdujo los datos necesarios.
    - **A.1:** el sistema determina que el usuario no introdujo ningún valor en el campo necesario.
    - **A.2:** el sistema informa del error ocurrido a través de un mensaje.
    - **A.3:** el control de flujo regresa al punto 2 del curso básico de acción.
  - **Curso alterno B:** la información correspondiente no pudo ser almacenada en la Base de Datos.

- **B.1:** el sistema determina que la operación solicitada no pudo ser realizada.
  - **B.2:** el sistema informa del error ocurrido a través de un mensaje.
  - **B.3:** el control de flujo regresa al punto 2 del curso básico de acción.
- 
- **Nombre: *Modificar Página de Enlaces.***
    - **Identificador:** CU 3.3.3.3.
    - **Descripción:** permite al usuario modificar el contenido asociado a una página de enlaces.
    - **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Página de Enlaces*. Debe existir una página principal y una página de enlaces.
    - **Pos-condición:** se le muestra al usuario un formulario con el contenido de la página de enlaces.
    - **Curso básico de acción:**
      1. El usuario selecciona la opción *Modificar Página de Enlaces*.
      2. Se despliega una interfaz con el contenido de la página de enlaces, la cual permite modificarlo.
      3. El usuario hace clic en el botón *Modificar Página de Enlaces*.
      4. El sistema realiza una primera validación del valor introducido, con la finalidad de determinar que el usuario no deje el campo en blanco. [Curso alterno A].
      5. El sistema recibe los datos introducidos por el usuario y valida que los mismo sean correctos.
      6. Si los datos son correctos, el sistema actualiza el campo correspondiente en la Base de Datos y le es notificado al usuario que la operación solicitada fue realizada con éxito. [Curso alterno B].
      7. Finaliza el caso de uso.

- **Cursos alternos:**
  - **Curso alternativo A:** la información correspondiente no pudo ser almacenada en la Base de Datos.
    - **A.1:** el sistema determina que el usuario no introdujo ningún valor en el campo necesario.
    - **A.2:** el sistema informa del error ocurrido a través de un mensaje.
    - **A.3:** el control de flujo regresa al punto 2 del curso básico de acción.
  - **Curso alternativo B:** algunos de los campos tiene un valor nulo o no respeta los formatos establecidos.
    - **B.1:** el sistema determina que la operación solicitada no pudo ser realizada.
    - **B.2:** el sistema informa del error ocurrido a través de un mensaje.
    - **B.3:** el control de flujo regresa al punto 2 del curso básico de acción.
- **Nombre:** *Eliminar Página de Enlaces.*
- **Identificador:** CU 3.3.3.4.
- **Descripción:** permite al usuario eliminar la página de enlaces junto con su contenido.
- **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Página de Enlaces*; Debe existir una página principal y una página de enlaces.
- **Pos-condición:** la página de enlaces y su contenido es eliminado.
- **Curso básico de acción:**
  1. El usuario selecciona la opción *Eliminar Página de Enlaces.*
  2. Se elimina el registro asociado a la página en la Base de Datos.
  3. Finaliza el caso de uso.



**Figura 42.** Diagrama de CU – nivel 3 – sub-módulo administrar pagina de imágenes.

- **Descripción de los casos de uso**

- **Nombre: Consultar Página de Imágenes.**

- **Identificador:** CU 3.3.4.1.
- **Descripción:** permite visualizar el contenido de la página de imágenes del Sitio Web creado por el usuario.
- **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Página de Imágenes*. Debe existir una página de imágenes.
- **Pos-condición:** Es mostrado al usuario la información concerniente a la página de imágenes del Sitio Web, en caso de no existir contenido, el sistema indica dicha situación a través de un mensaje y provee la opción para crear una página de este tipo y agregar el contenido deseado.

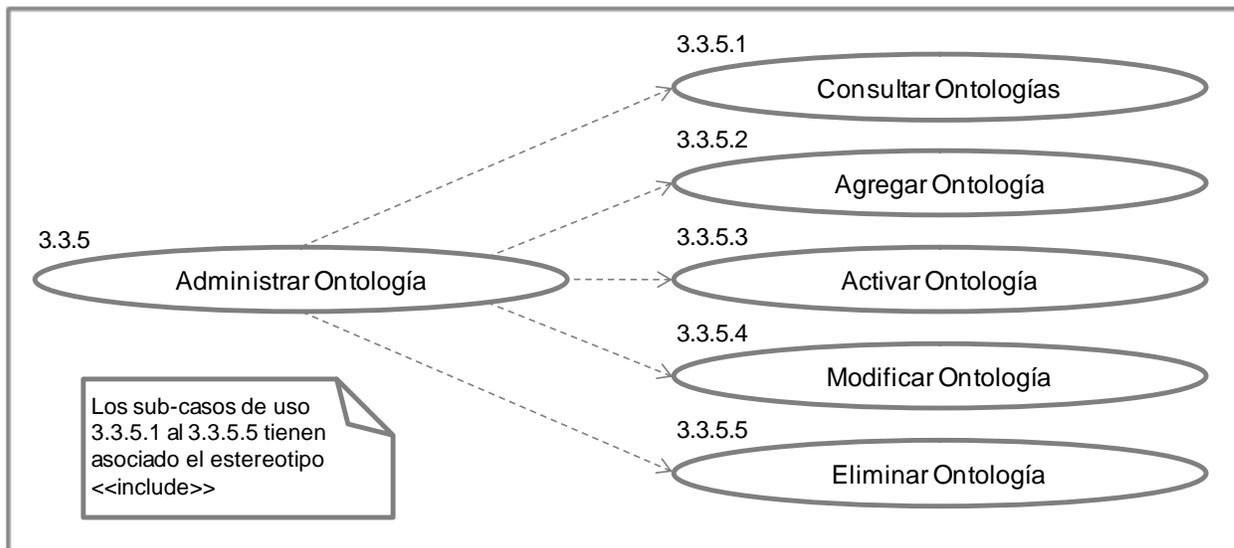
- **Nombre: Agregar Página de Imágenes.**

- **Identificador:** CU 3.3.4.2.
- **Descripción:** permite al usuario agregar una página de imágenes a un Sitio Web y añadir información concerniente a la misma.

- **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Página de Imágenes*. Debe existir una página principal.
- **Pos-condición:** se le muestra al usuario un formulario con el valor requerido por el sistema para agregar una nueva página de imágenes junto con su contenido.
- **Curso básico de acción:**
  1. El usuario selecciona la opción *Agregar Página de imágenes*.
  2. Se despliega una interfaz para introducir los valores correspondientes.
  3. El usuario hace clic en el botón *Agregar Página de Imágenes*.
  4. El sistema realiza una primera validación de los valores introducidos, con la finalidad de determinar que el usuario no dejara campos en blanco. [Curso alternativo A].
  5. El sistema recibe los datos introducidos por el usuario y valida que los mismo sean correctos.
  6. Si los datos son correctos, los mismos son almacenados en la Base de Datos y le es notificado al usuario que la operación solicitada fue realizada con éxito. [Curso alternativo B].
  7. Finaliza el caso de uso.
- **Cursos alternos:**
  - **Curso alternativo A:** el usuario no introdujo los datos necesarios.
    - **A.1:** el sistema determina que el usuario no introdujo ningún valor en el campo necesario.
    - **A.2:** el sistema informa del error ocurrido a través de un mensaje.
    - **A.3:** el control de flujo regresa al punto 2 del curso básico de acción.
  - **Curso alternativo B:** la información correspondiente no pudo ser almacenada en la Base de Datos.

- **B.1:** el sistema determina que la operación solicitada no pudo ser realizada.
  - **B.2:** el sistema informa del error ocurrido a través de un mensaje.
  - **B.3:** el control de flujo regresa al punto 2 del curso básico de acción.
- 
- **Nombre:** *Modificar Página de Imágenes.*
  - **Identificador:** CU 3.3.4.3.
  - **Descripción:** permite al usuario modificar el contenido asociado a una página de imágenes.
  - **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Página de Imágenes*. Debe existir una página principal y una página de imágenes.
  - **Pos-condición:** se le muestra al usuario un formulario con el contenido de la página de imágenes.
  - **Curso básico de acción:**
    1. El usuario selecciona la opción *Modificar Página de Imágenes*.
    2. Se despliega una interfaz con el contenido de la página de imágenes, la cual permite modificarlo.
    3. El usuario hace clic en el botón *Modificar Página de Imágenes*.
    4. El sistema realiza una primera validación del valor introducido, con la finalidad de determinar que el usuario no deje el campo en blanco. [Curso alterno A].
    5. El sistema recibe los datos introducidos por el usuario y valida que los mismo sean correctos.
    6. Si los datos son correctos, el sistema actualiza el campo correspondiente en la Base de Datos y le es notificado al usuario que la operación solicitada fue realizada con éxito. [Curso alterno B].
    7. Finaliza el caso de uso.

- **Cursos alternos:**
  - **Curso alternativo A:** la información correspondiente no pudo ser almacenada en la Base de Datos.
    - **A.1:** el sistema determina que el usuario no introdujo ningún valor en el campo necesario.
    - **A.2:** el sistema informa del error ocurrido a través de un mensaje.
    - **A.3:** el control de flujo regresa al punto 2 del curso básico de acción.
  - **Curso alternativo B:** algunos de los campos tiene un valor nulo o no respeta los formatos establecidos.
    - **B.1:** el sistema determina que la operación solicitada no pudo ser realizada.
    - **B.2:** el sistema informa del error ocurrido a través de un mensaje.
    - **B.3:** el control de flujo regresa al punto 2 del curso básico de acción.
- **Nombre:** *Eliminar Página de Imágenes.*
- **Identificador:** CU 3.3.4.4.
- **Descripción:** permite al usuario eliminar la página de enlaces junto con su contenido.
- **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Página de imágenes*; Debe existir una página principal y una página de imágenes.
- **Pos-condición:** la página de imágenes y su contenido es eliminado.
- **Curso básico de acción:**
  1. El usuario selecciona la opción *Eliminar Página de Imágenes.*
  2. Se elimina el registro asociado a la página en la Base de Datos.
  3. Finaliza el caso de uso.



**Figura 43.** Diagrama de CU – nivel 3 – sub-módulo administrar ontología.

- **Descripción de los casos de uso**

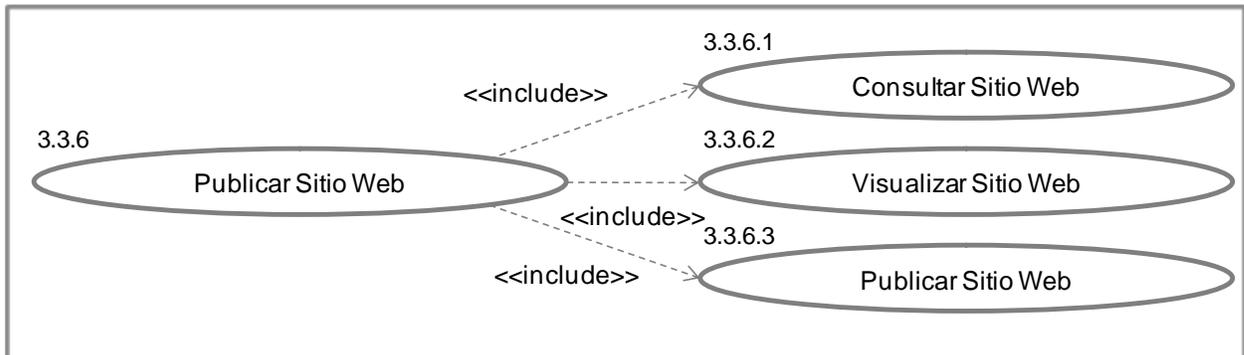
- **Nombre:** *Consultar Ontologías.*
  - **Identificador:** CU 3.3.5.1.
  - **Descripción:** permite listar las distintas versiones de la ontología cargada previamente al sitio Web creado por el usuario.
  - **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Ontología.*
  - **Pos-condición:** es mostrado al usuario un listado de las distintas versiones de la ontología cargada con anterioridad al sitio Web.
- **Nombre:** *Agregar Ontología.*
  - **Identificador:** CU 3.3.5.2.
  - **Descripción:** permite al usuario cargar una nueva versión de la ontología a un sitio Web.
  - **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Ontología.*

- **Pos-condición:** se le muestra al usuario un formulario con los valores requeridos por el sistema para cargar una nueva versión de la ontología.
- **Curso básico de acción:**
  1. El usuario selecciona la opción *Agregar Ontología*.
  2. Se despliega una interfaz para introducir los valores correspondientes.
  3. El usuario hace clic en el botón Agregar Ontología.
  4. El sistema realiza una primera validación de los valores introducidos, con la finalidad de determinar que el usuario no dejará campos en blanco y que el formato de los mismos sea correcto. [Curso alternativo A].
  5. El sistema recibe los datos introducidos por el usuario y valida que los mismo sean correctos.
  6. Si los datos son correctos, el sistema actualiza los campos correspondientes en la Base de Datos y le es notificado al usuario que la operación solicitada fue realizada con éxito. [Curso alternativo B].
  7. El sistema almacena la nueva versión de la ontología en la ruta asociada al contexto del sitio Web. [Curso alternativo C].
  8. Finaliza el caso de uso.
- **Cursos alternos:**
  - **Curso alternativo A:** el usuario no introdujo datos en alguno de los campos correspondientes o al menos uno no coincide con el formato preestablecido.
    - **A.1:** el sistema determina que el usuario no introdujo ningún valor en alguno de los campos necesarios o que al menos uno de los mismos presenta valores erróneos.
    - **A.2:** el sistema informa del error ocurrido a través de un mensaje.
    - **A.3:** el control de flujo regresa al punto 2 del curso básico de acción.

- **Curso alternativo B:** algunos de los campos tiene un valor nulo o no respeta los formatos establecidos.
  - **B.1:** el sistema determina que el usuario dejó en blanco algunos de los campos correspondientes o no concuerdan con el formato preestablecido.
  - **B.2:** el sistema informa del error ocurrido a través de un mensaje.
  - **B.3:** el control de flujo regresa al punto 2 del curso básico de acción.
- **Curso alternativo C:** Ocurre un error en la carga de la nueva versión de la ontología.
  - **C.1:** el sistema determina que ocurrió un error durante la carga de la nueva versión de la ontología al servidor.
  - **C.2:** el sistema informa del error ocurrido a través de un mensaje y realiza un *Rollback* de cualquier cambio realizado a la Base de Datos.
  - **C.3:** el control de flujo regresa al punto 2 del curso básico de acción.
  
- **Nombre: Activar Ontología.**
  - **Identificador:** CU 3.3.5.3.
  - **Descripción:** permite al usuario seleccionar del conjunto de versiones disponibles, la ontología que será mostrada en el sitio Web a generar.
  - **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Ontología*.
  - **Pos-condición:** es almacenado en la Base de Datos la ontología seleccionada para ser visualizada en el sitio Web una vez generado.
  
- **Nombre: Modificar Ontología.**
  - **Identificador:** CU 3.3.5.4.

- **Descripción:** permite al usuario modificar los datos de las distintas versiones de la ontología del sitio Web creado.
- **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Ontología*.
- **Pos-condición:** es mostrado al usuario un formulario con los valores requeridos por el sistema para modificar los datos de la ontología.
- **Curso básico de acción:**
  1. El usuario selecciona la opción *Modificar Ontología*.
  2. Se despliega una interfaz para introducir los valores correspondientes.
  3. El usuario hace clic en el botón Modificar Ontología.
  4. El sistema realiza una primera validación de los valores introducidos, con la finalidad de determinar que el usuario no dejará campos en blanco y que el formato de los mismos sea correcto. [Curso alternativo A].
  5. El sistema recibe los datos introducidos por el usuario y valida que los mismo sean correctos.
  6. Si los datos son correctos, el sistema actualiza los campos correspondientes en la Base de Datos y le es notificado al usuario que la operación solicitada fue realizada con éxito. [Curso alternativo B].
  7. Finaliza el caso de uso.
- **Cursos alternos:**
  - **Curso alternativo A:** el usuario no introdujo datos en alguno de los campos correspondientes o al menos uno no coincide con el formato preestablecido.
    - **A.1:** el sistema determina que el usuario no introdujo ningún valor en alguno de los campos necesarios o que al menos uno de los mismos presenta valores erróneos.
    - **A.2:** el sistema informa del error ocurrido a través de un mensaje.

- **A.3:** el control de flujo regresa al punto 2 del curso básico de acción.
- **Curso alternativo B:** algunos de los campos tiene un valor nulo o no respeta los formatos establecidos.
  - **B.1:** el sistema determina que el usuario dejó en blanco algunos de los campos correspondientes o no concuerdan con el formato preestablecido.
  - **B.2:** el sistema informa del error ocurrido a través de un mensaje.
  - **B.3:** el control de flujo regresa al punto 2 del curso básico de acción.
  
- **Nombre:** *Eliminar Ontología*.
- **Identificador:** CU 3.3.5.5.
- **Descripción:** permite al usuario eliminar una versión de la ontología del sitio Web.
- **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Administrar Ontología*; debe existir al menos una versión de la ontología cargada previamente en el sitio Web.
- **Pos-condición:** la versión de la ontología es eliminada en el servidor y en la Base de Datos.
- **Curso básico de acción:**
  1. El usuario selecciona la opción *Eliminar Ontología* asociada a la versión de la ontología que desea eliminar.
  2. El sistema borra la versión de la ontología en el servidor.
  3. Se elimina el registro asociado a la versión de la ontología en la Base de Datos.
  4. Finaliza el caso de uso.



**Figura 44.** Diagrama de CU – nivel 3 – sub-módulo publicar sitio Web.

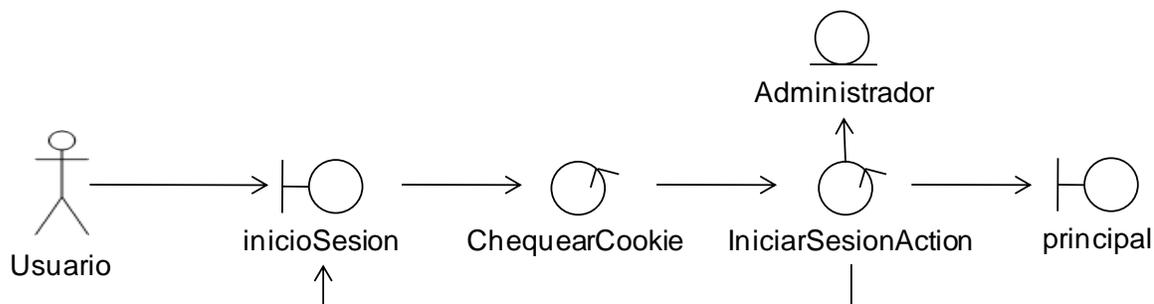
- **Descripción de los casos de uso**

- **Nombre:** *Consultar Sitio Web.*
  - **Identificador:** CU 3.3.6.1.
  - **Descripción:** permite al usuario visualizar un resumen del contenido publicado en el sitio Web a generar.
  - **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Publicar Sitio Web*.
  - **Pos-condición:** es mostrado al usuario un resumen de cada uno de los contenidos creados por el usuario que estarán presentes en el sitio Web una vez generado.
- **Nombre:** *Visualizar Sitio Web.*
  - **Identificador:** CU 3.3.6.2.
  - **Descripción:** permite al usuario visualizar el sitio Web por completo.
  - **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Publicar Sitio Web*.
  - **Pos-condición:** se le muestra al usuario cada una de las páginas que conformarán el sitio Web creado, tomando en cuenta el flujo de navegación entre las mismas.

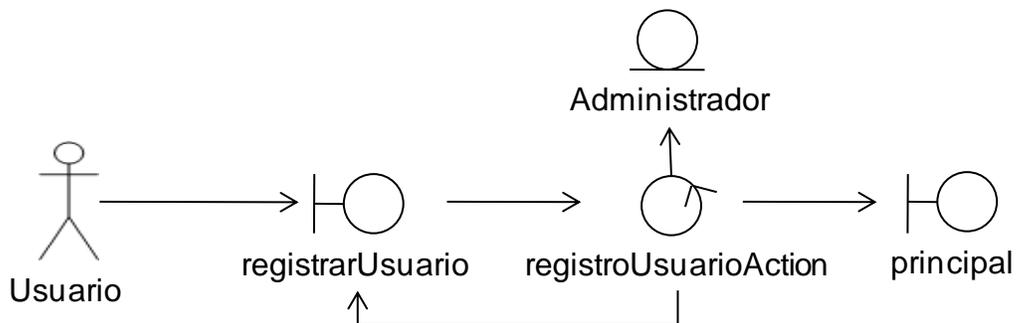
- **Nombre:** *Visualizar Sitio Web.*
- **Identificador:** CU 3.3.6.3.
- **Descripción:** permite al usuario descargar el sitio Web en un formato comprimido.
- **Pre-condición:** el usuario debe iniciar sesión para entrar a las opciones del sistema y debe haber seleccionado la opción *Publicar Sitio Web.*
- **Pos-condición:** el usuario logra descargar el sitio Web creado.

**ANEXO 3: DIAGRAMA DE ROBUSTEZ.**

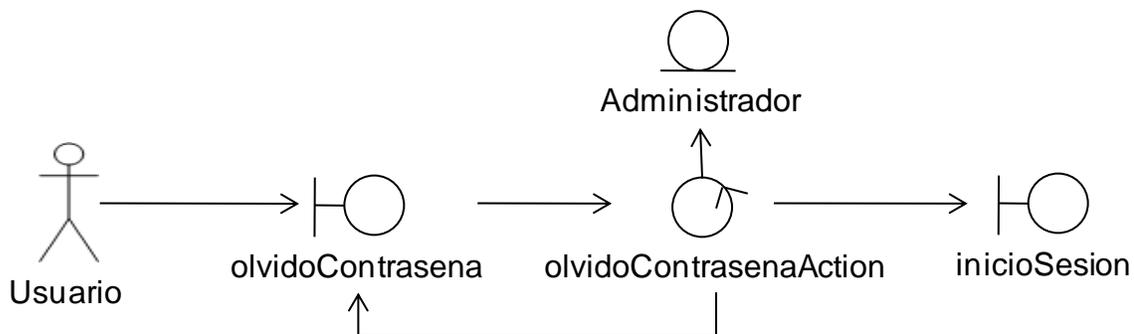
- Iniciar Sesión**



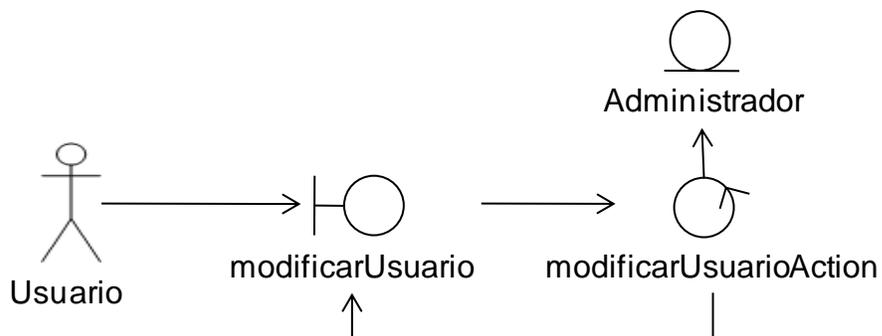
- Registrar Nuevo Usuario**



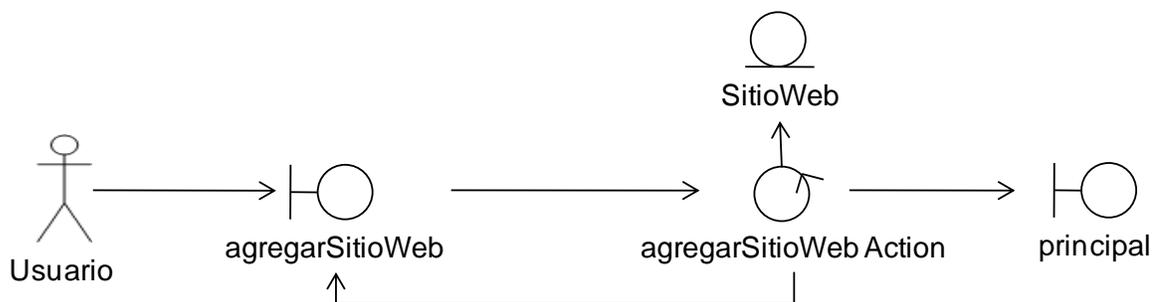
- Olvido de Contraseña**



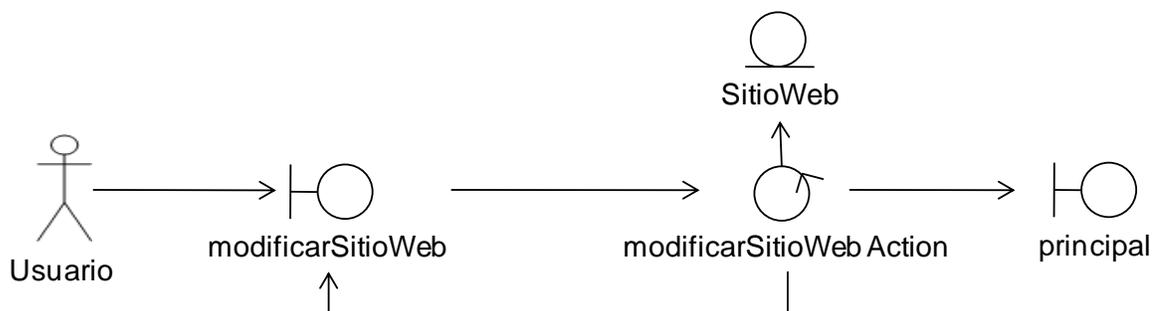
- **Modificar Datos Personales**



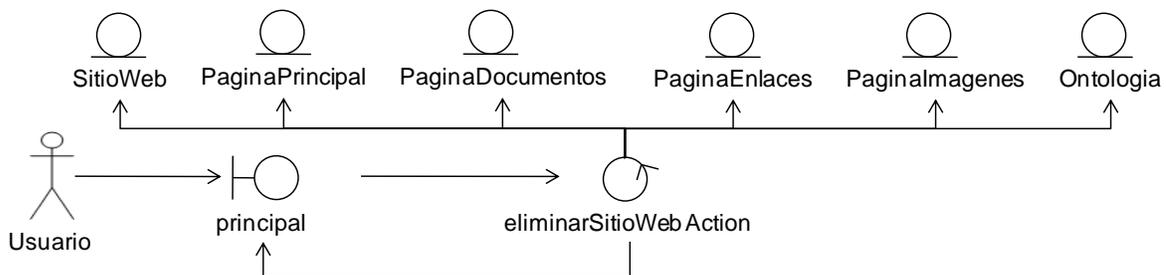
- **Agregar Sitio Web**



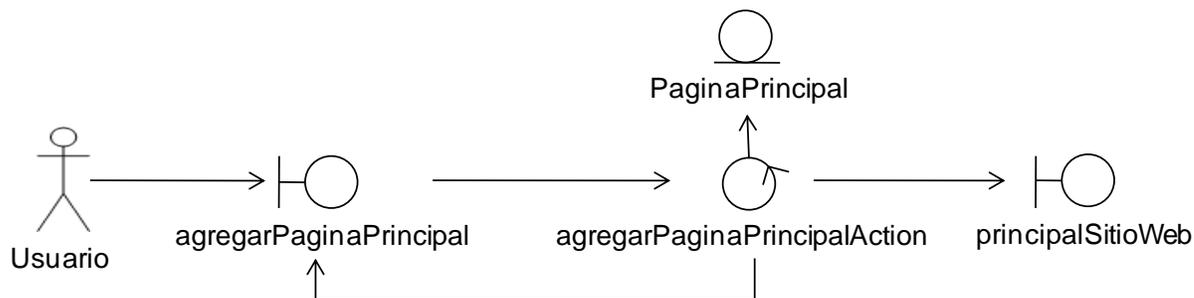
- **Modificar Sitio Web**



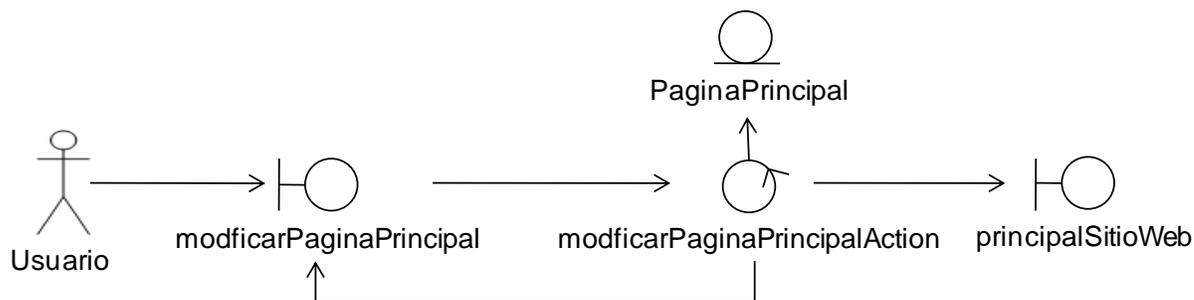
• **Eliminar Sitio Web**



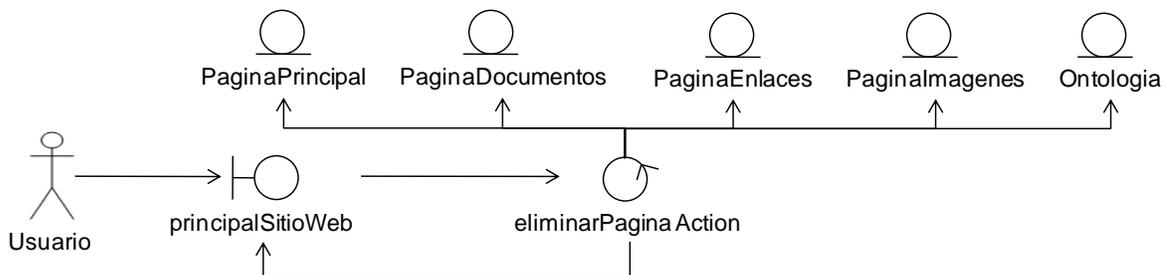
• **Agregar Página Principal**



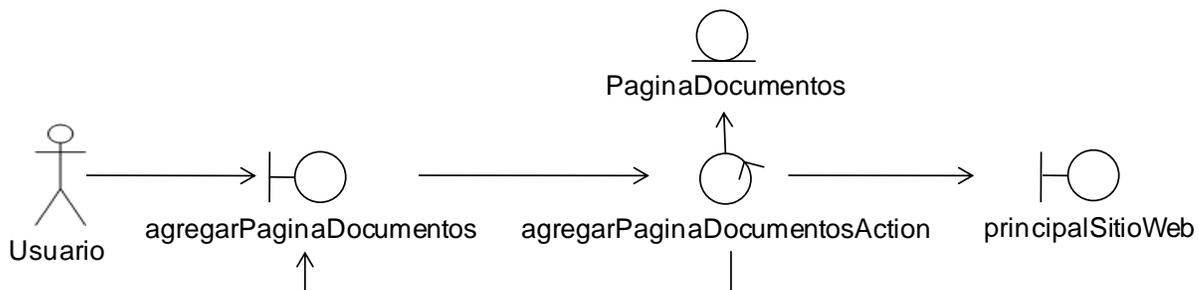
• **Modificar Página Principal**



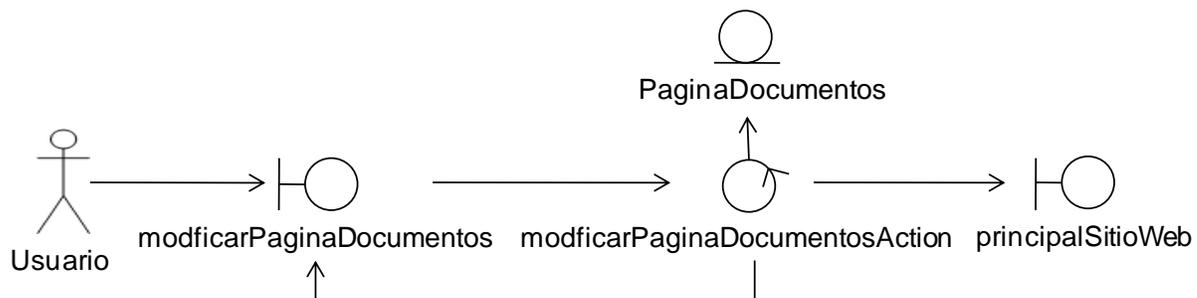
• **Eliminar Página Principal**



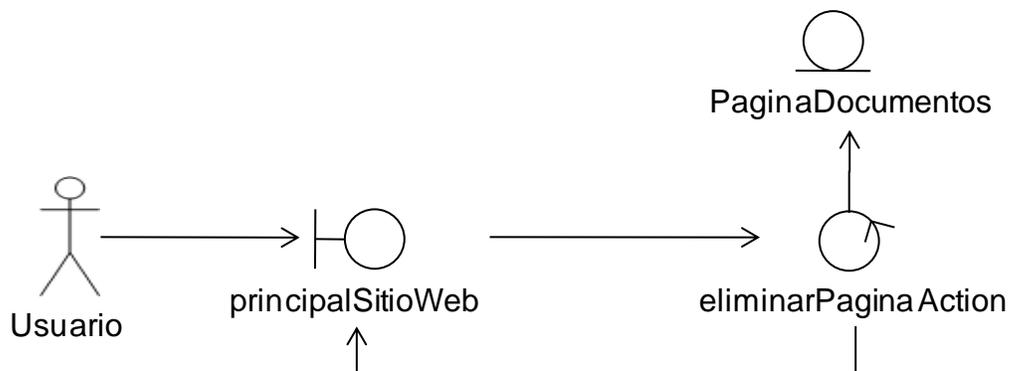
• **Agregar Página de Documentos**



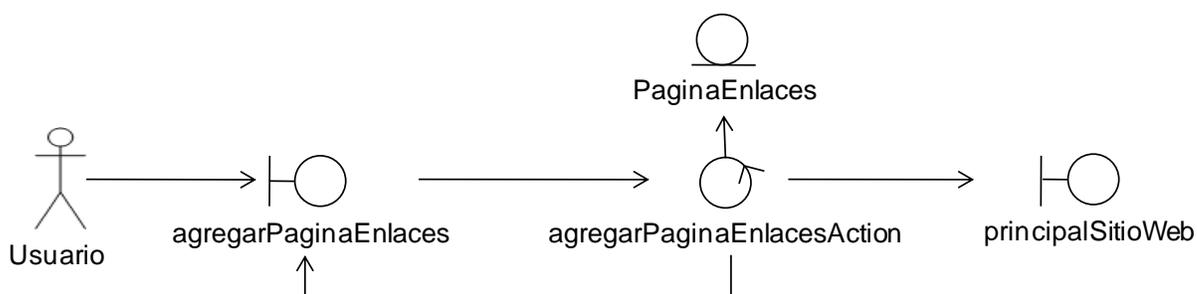
• **Modificar Página de Documentos**



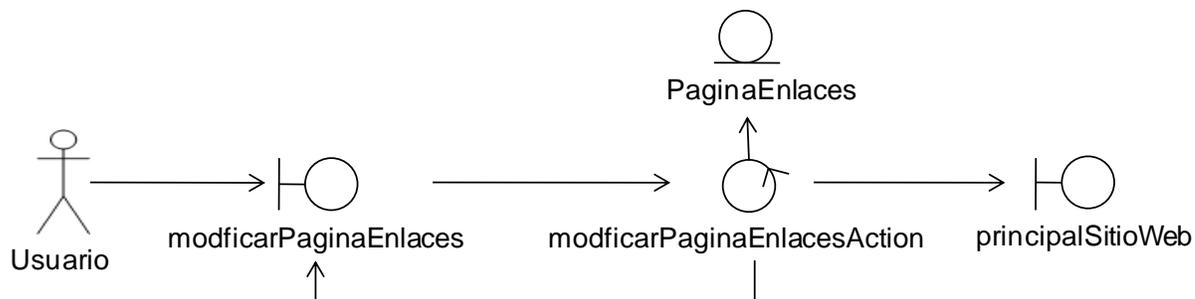
- **Eliminar Página de Documentos**



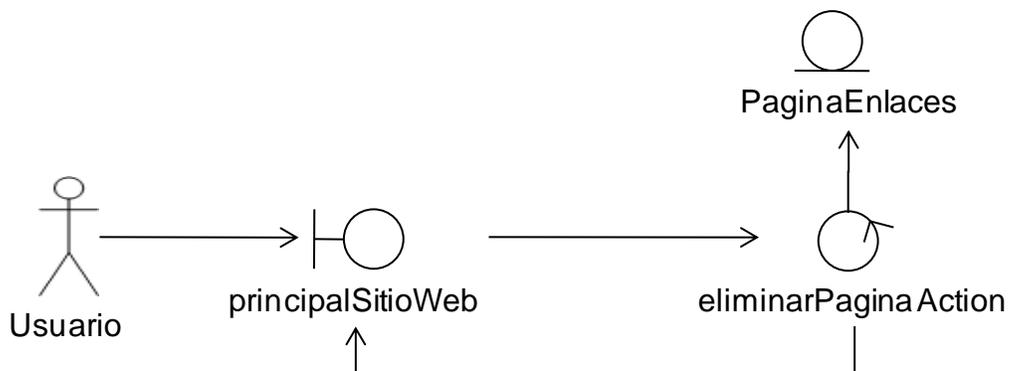
- **Agregar Página de Enlaces**



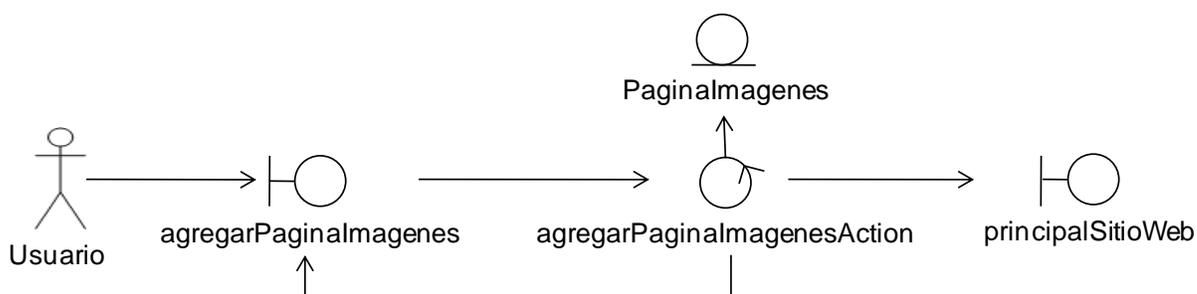
- **Modificar Página de Enlaces**



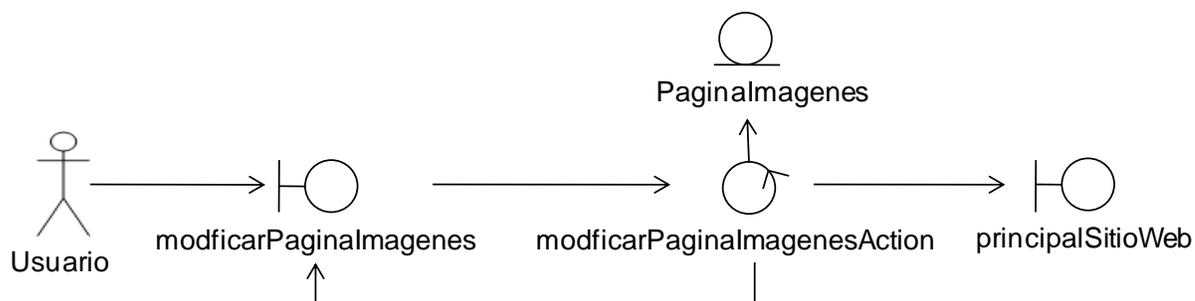
- **Eliminar Página de Enlaces**



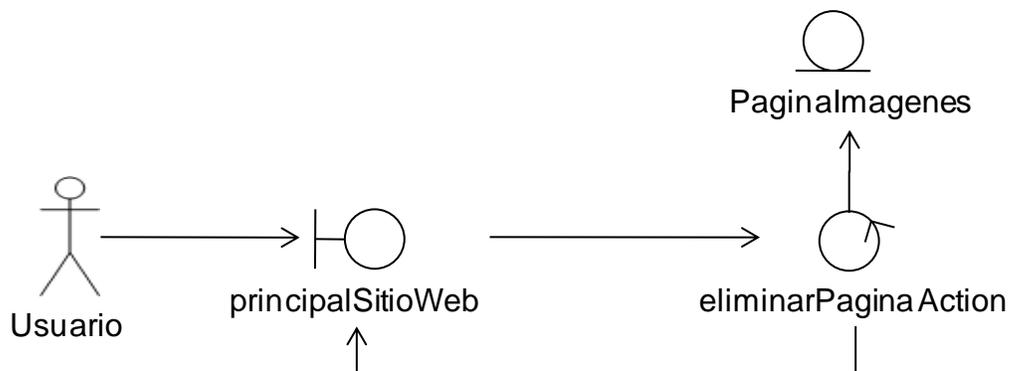
- **Agregar Página de Imágenes**



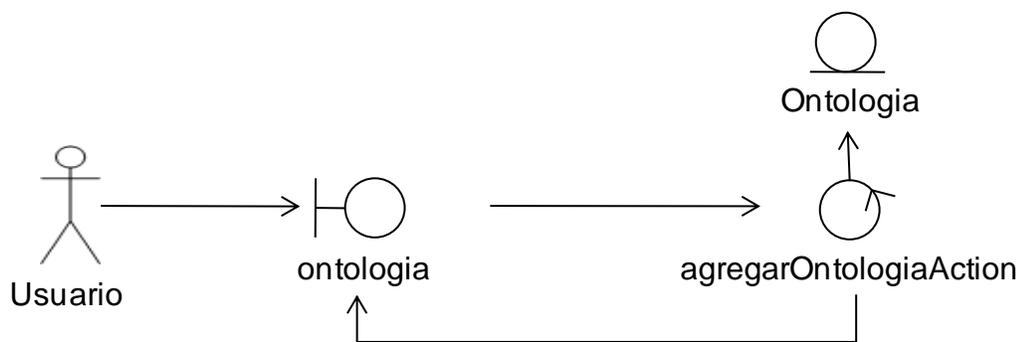
- **Modificar Página de Imágenes**



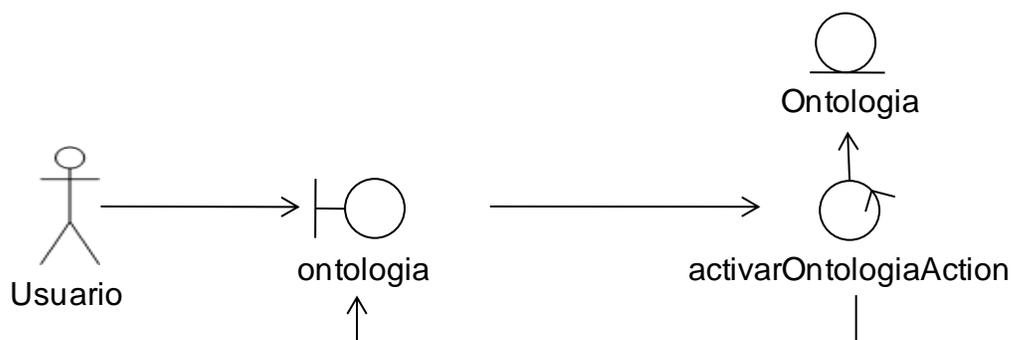
- **Eliminar Página de Imágenes**



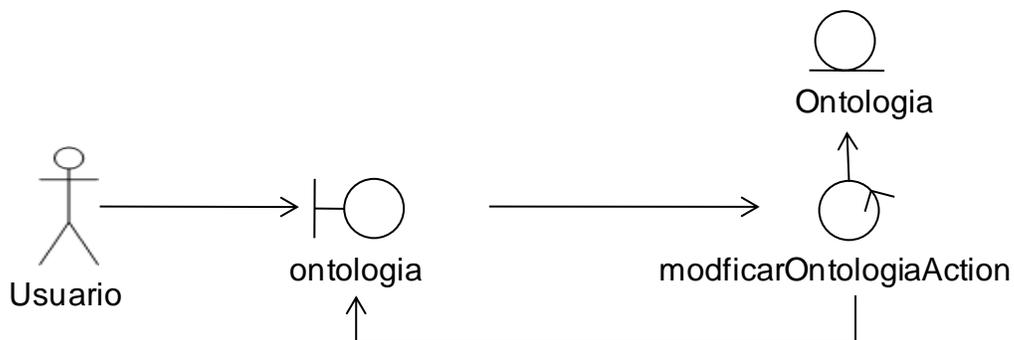
- **Agregar Ontología**



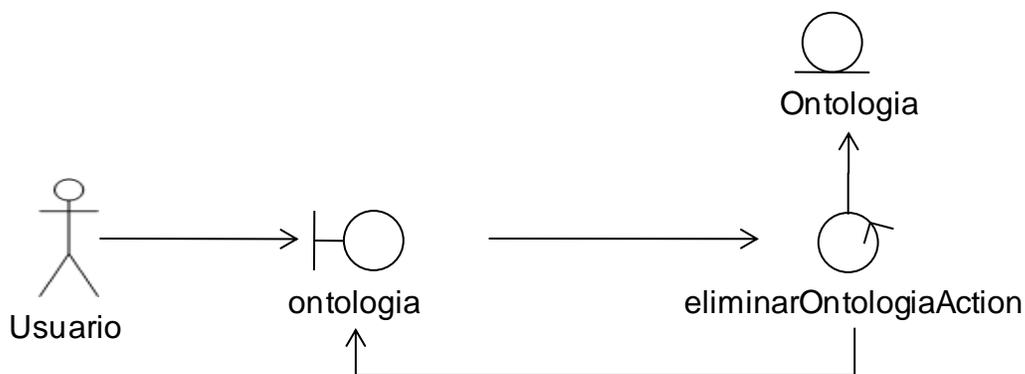
- **Activar Ontología**



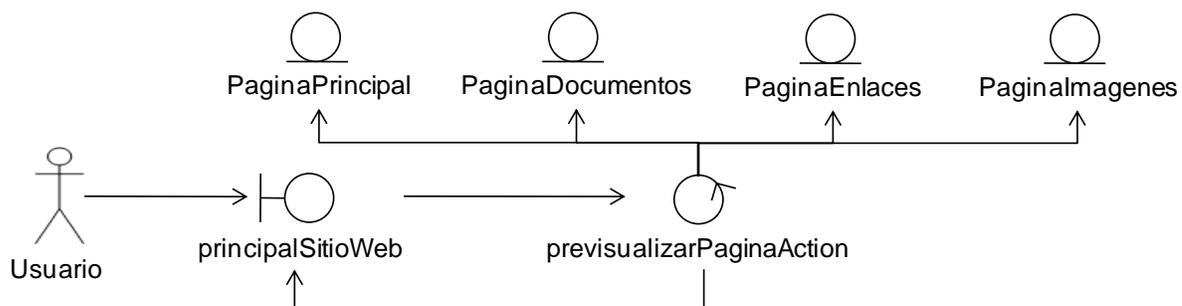
- **Modificar Datos de Ontología**



- **Eliminar Ontología**



- **Vista Preliminar de Página**



- **Generar Sitio Web**

