



UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS
Centro de Calidad para Soluciones Informáticas
CALISOFT

**Desarrollo de una ontología de apoyo al procedimiento del
Departamento de Pruebas de Software**

Trabajo final presentado en opción al título de
Máster en Calidad de Software

Autor:

Ing. Delvis Echeverría Perez

Tutor:

Msc. Yamilis Fernández Pérez

Ciudad de La Habana, Diciembre 2011.
“Año 53 de la Revolución”.

Gracias a Dios.

“Porque grande es tu amor por mí”

Gracias a mi familia.

Que por pequeña no deja de ser grande: A mi mamá, mi abuela, mi tía, mi hermano y mi prima; por su cariño, comprensión y apoyo sin condiciones ni medida.

Gracias a mis hermanos.

Roig y Yetel sin ustedes no estuviera donde estoy.

Gracias a mi tutora.

Por ayudarme, por sus opiniones, y por dedicarle tiempo a mí trabajo.

Gracias a todos mis compañeros de trabajo.

.

.... y a “Google” por mostrarme siempre una respuesta.

“A todos muchísimas gracias”

Dedicatoria

A mi familia y Amigos

DECLARACIÓN JURADA DE AUTORÍA Y AGRADECIMIENTOS

Declaro por este medio que yo Delvis Echeverría Perez, con carné de identidad 83041911067, soy el autor principal del trabajo final de maestría: Desarrollo de una Ontología de apoyo al procedimiento del Departamento de Pruebas de Software, desarrollada como parte de la Maestría en Calidad de Software y que autorizo a la Universidad de las Ciencias Informáticas a hacer uso de la misma en su beneficio, así como los derechos patrimoniales con carácter exclusivo.

Y para que así conste, firmo la presente declaración jurada de autoría en Ciudad de La Habana a los días del mes de del año.

Resumen

La gestión del conocimiento provee técnicas y métodos que ayudan a reducir la pérdida o el desaprovechamiento de conocimiento. Una de las técnicas para la gestión del conocimiento es el desarrollo de Ontologías. En la presente investigación se propone una ontología enmarcada en el dominio de Pruebas de Software, específicamente en el Departamento de Pruebas de Software de CALISOFT, para su creación se describen los principales elementos que la componen, como clases, subclases, relaciones, instancias, además se especifican las principales herramientas y lenguajes utilizados para su desarrollo. Teniendo como objetivo primordial compartir y organizar el conocimiento acumulado en esta área se pretende que pueda ser utilizada por estudiantes y profesores involucrados en el procedimiento de pruebas. Para validar la propuesta se implementó una pequeña herramienta que permite responder a las preguntas de competencias antes definidas en la investigación, y se utilizó el servicio de validación de W3C para validar la consistencia del código generado a partir de la ontología.

Palabras Claves: Gestión del Conocimiento, Ontología, Procedimiento de Pruebas.

Índice de Contenidos

Resumen	IV
Introducción.....	1
Capítulo 1: Fundamentación Teórica.....	6
Introducción al Capítulo.....	6
1.1 Gestión del Conocimiento.	6
1.1.1 Procesos en la Gestión del Conocimiento	7
1.1.2 Estrategias de Gestión del Conocimiento.	8
1.2 Ontología.....	9
1.2.1 Conceptos relacionados.	9
1.2.2 Tipos de Ontología.	10
1.2.3 Componentes de las Ontologías.....	11
1.2.4 Lenguajes para la representación de Ontologías.....	12
1.2.5 Herramientas para la construcción de Ontologías.	14
1.2.6 Metodologías para la construcción de ontologías.	15
1.2.7 Razonadores.....	17
1.2.8 Sistemas de Almacenamiento.....	18
1.2.9 IDE de desarrollo.....	19
1.2.10 Reutilización de Ontologías.....	20
1.3 Conclusiones Parciales.....	21
Capítulo 2. Diseño de la Solución.	23
Introducción al Capítulo.....	23
2.1 Descripción de la Información presente en el procedimiento del DPSW.....	23
2.2 Metodología METHONTOLOGY.....	¡Error! Marcador no definido.

2.3	Actividad: Especificación.....	25
2.3.1	Departamento de Pruebas de Software de Calisoft	25
2.3.2	Preguntas de Competencia.....	26
2.4	Actividad: Conceptualización.....	27
2.4.1	Glosario de términos.....	28
2.4.2	Taxonomía de Conceptos.....	33
2.4.3	Construir diagramas de relaciones binarias.....	34
2.4.4	Construir el diccionario de conceptos.....	35
2.4.5	Definir Axiomas.....	36
2.4.6	Describir las instancias.....	36
2.5	Actividad: Formalización.....	37
2.5.1	Creación de las Clases.....	38
2.5.2	Creación de las Propiedades	39
2.5.3	Creación de las Instancias.....	41
2.5.4	Obtención del grafo de la Ontología propuesta	42
2.5.5	Creación de Axiomas.....	43
2.6	Actividad: Implementación.....	44
2.7	Actividad: Mantenimiento.....	45
2.8	Conclusiones Parciales.....	46
Capítulo 3. Validación de la Ontología.....		47
3.1	Transformación Modelo Ontológico al Modelo Persistente.....	47
3.2	Cargar una ontología con Jena desde almacenamiento persistente.....	49
3.3	Herramienta Informática basada en la Ontología.....	50
3.4	Resultados obtenidos con el Sistema en el DPSW.....	54

3.5 Validación mediante el Servicio de Validación Web de la W3C.	52
3.6 Conclusiones Parciales.....	61
Conclusiones.....	62
Recomendaciones.	63
Referencias Bibliográficas.....	64
Bibliografía.....	68

Índice de Figuras.

Figura 1. Captura de Conocimiento para el desarrollo de la Ontología.....	24
Figura 2. Parte de la Estrategia de Evaluación.....	24
Figura 3. Estructura Organizativa del Centro.	26
Figura 4. Taxonomía de Conceptos.....	34
Figura 5. Relaciones Binarias.....	34
Figura 6. Interfaz Principal Protégé.....	38
Figura 7. Definición de las clases creadas en Protégé.	39
Figura 8. Descripción de las Propiedades en Protégé.	40
Figura 9. Descripción de los atributos de Instancias.	41
Figura 10. Creación de Instancias.....	42
Figura 11. Grafo generado en Protégé.	43
Figura 12. Descripción de Axiomas.	44
Figura 13. Exportación del Modelo Ontológico al Modelo Persistente.	47
Figura 14. Tabla de la BD generada por Jena y Protégé.....	48
Figura 15. Uso del NetBeans para la visualización del modelo persistente.....	49
Figura 16. Herramienta para realizar consultas a la ontología desarrollada.	50
Figura 17. Encuesta 1er Aspecto.	56
Figura 18. Encuesta 2do Aspecto.	56
Figura 19. Encuesta 3er Aspecto.	57
Figura 20. Encuesta 4to Aspecto.	58
Figura 21. Encuesta 5to Aspecto.	58
Figura 22. Encuesta 6to Aspecto.	59
Figura 23. Servicio de Validación de Ontologías.....	53
Figura 24. Resultado de la ontología por el Validador W3C.....	54

Índice de Tabla

Tabla 1. Métodos de Investigación.	4
Tabla 2. Glosarios de términos.	28
Tabla 3. Diccionario de Conceptos.	35
Tabla 4. Definición de Axiomas.	36
Tabla 5. Descripción de Instancias.	36

Introducción

A través de la historia de la humanidad, el hombre se ha enfrentado a diversos problemas relacionados con el dominio del conocimiento, el manejo, la modificación y las formas de compartirlo. La aparición y creciente importancia de este elemento como un nuevo factor de producción hace que el desarrollo de tecnologías, metodologías y estrategias para su medición, creación y difusión se convierta en una de las principales prioridades de las organizaciones en la sociedad del conocimiento. Sin embargo, también podemos considerar que ha sido precisamente el desarrollo de esas tecnologías y metodologías para la medición y difusión del conocimiento las que han convertido el conocimiento en un elemento indispensable para el desarrollo económico y social.

El ámbito de la representación y recuperación de la información ha tenido que asumir el impacto de Internet y sus tecnologías asociadas, en especial la Web. Estos cambios en la tecnología están conduciendo a una progresiva digitalización del campo de la representación y recuperación de información que afecta por igual a los recursos de información, las herramientas de representación, recuperación y los requerimientos de los usuarios. En este ambiente de creciente digitalización, son diversas las herramientas de representación y recuperación de la información que pueden ser objeto de estudio. De igual forma son variados los campos de conocimiento donde se aplican estas herramientas: la Lingüística, la Inteligencia Artificial, y la Documentación.

La Web actual a pesar de haber alcanzado un gran éxito presenta algunas limitaciones, las cuales si se pudiesen mejorar permitirían un mayor grado de aceptación por parte de los usuarios. La existencia de un gran volumen información en la web y la falta de organización de la misma dificulta el trabajo a la hora de realizar alguna búsqueda sobre un tema determinado, pues en ocasiones se busca algo sobre un tema y se obtiene como resultado documentos cuyo contenido no guarda relación alguna con lo que se quiere encontrar, o se encuentra lo que se buscaba sólo después de visitar varias páginas.

Una posible solución a estos problemas es el uso de la Web Semántica (WS), la cual no es más que “Una visión de una futura Web en la cual la información que en la versión actual de la Web es comprensible solamente por los seres humanos también esté

disponible de una manera formal para sistemas inteligentes". El uso de la Web Semántica no sería la completa solución a este problema, es necesario que la información se encuentre estructurada de manera que facilite el razonamiento automatizado de la misma.

En la literatura especializada [1, 16, 18] se analizan como herramientas para la representación y recuperación de información: taxonomías, sistemas de clasificación, bases de datos léxicas, tesauros, bases de conocimiento, mapas conceptuales, ontologías, entre otros. Dentro de este amplio espectro de herramientas de representación y recuperación de información, las ontologías constituyen una de las que con mayor frecuencia son objeto de vinculación en la bibliografía. Todas estas herramientas colaboran en la descripción de los diferentes recursos de información y en su posterior recuperación, en términos de efectividad, rapidez y facilidad de acceso a la información. [1]

Una ontología es un tipo de instrumento que permite la representación del conocimiento en un área determinada en clara conexión con su recuperación en entornos informáticos, es la descripción explícita de un dominio la cual define un vocabulario común [2]. Especifican un vocabulario relativo a un cierto dominio este vocabulario define entidades, clases, propiedades, predicados, funciones, y las relaciones de estos componentes. Las ontologías se encargan de definir los términos utilizados para describir y representar un área de conocimiento. Algunas de las ventajas de su utilización son: [3].

- Optimización de las aplicaciones de sistemas basados en conocimiento, su desarrollo e interoperabilidad.
- Preservación del conocimiento persistente de los expertos en cualquier campo de aplicación.
- La descripción de los conceptos y sus relaciones permiten una gestión rápida, eficaz y permanentemente de la información depositada en Internet permitiendo su recuperación y divulgación.

A raíz de la necesidad de garantizar un producto con mayor calidad en los servicios informáticos que ofrece la Universidad de las Ciencias Informáticas surge CALISOFT, Centro adscrito a la Agencia de Control y Supervisión del Ministerio de la Informática y las

Comunicaciones (MIC) de Cuba. Una organización enfocada a contribuir al desarrollo de la Industria Cubana de Software (InCuSoft) facilitando la implementación de las mejores prácticas en el proceso de desarrollo y/o mantenimiento de software. Responsable de la verificación y validación de productos, procesos y organizaciones según normas nacionales e internacionales. Dentro de su estructura organizativa se encuentra el Departamento de Pruebas de software (DPSW), encargado de las pruebas de liberación y aceptación de software.

En el presente trabajo se investigó sobre el procedimiento de pruebas del DPSW y la información que se genera a través del mismo, con el propósito de saber la situación real se aplica una encuesta a una muestra de especialistas del DPSW que están involucrados directamente en el procedimiento de Pruebas. Resultado que se constató en entrevistas realizadas a los jefes de grupo por los que se encuentra dividido el departamento. Según las valoraciones de los entrevistados, el DPSW continuamente se enfrenta a una gran cantidad de información proveniente del procedimiento de pruebas que éste lleva a cabo, permitiendo que el acceso a la información relevante se convierta en un factor crítico. Elementos como la transmisión y retroalimentación del conocimiento no son empleados correctamente, trayendo consigo que la experiencia adquirida del trabajo diario quede solamente en los especialistas y no en la organización.

Las siguientes problemáticas están presentes en el procedimiento del DPSW:

- Parte de la información que se genera en el procedimiento de pruebas del DPSW se limita solamente al almacenamiento de grandes volúmenes de información, usualmente está sin clasificar y poco estructurada.
- La falta del hábito de documentar todas las actividades que se desarrollan en el proceso de prueba, conlleva a que se pierda el conocimiento adquirido por el personal y deba invertirse nuevamente en su creación.
- Las búsquedas bibliográficas son realizadas solamente a través de internet, poniéndose de manifiesto la polisemia, donde muchas veces ocurre que el término utilizado para realizar la búsqueda posee disímiles significados, arrojando resultados ajenos al deseado, existiendo diferencias entre el conocimiento que presentan los miembros del departamento.

Por lo que el **Problema Científico** es ¿Cómo mejorar la comunicación, la organización de la información y la calidad del procedimiento de prueba en el departamento de pruebas de software?

Objetivo General: Desarrollar una ontología de apoyo al procedimiento del departamento de pruebas de software.

Idea a defender.

Con la creación de una ontología para el departamento de pruebas de software se contribuirá a mejorar la comunicación, la organización de la información y la calidad del procedimiento de pruebas; a través del uso de una fuente común.

Se define como **objeto de estudio** la disciplina de la Ingeniería Ontológica y enmarcando como **campo de acción** las Ontología como técnicas para gestionar el conocimiento en las pruebas de software.

Objetivos específicos:

- Fundamentar los conceptos relacionados con la Gestión del Conocimiento y la ingeniería ontológica, así como sus técnicas, metodologías, y herramientas.
- Definir la estructura organizativa de la información generada en el procedimiento del DPSW.
- Desarrollar la Ontología para modelar y utilizar la información generada en el procedimiento del DPSW.
- Validar la propuesta ontológica en función de la comunicación, la organización de la información y la calidad del procedimiento de pruebas.

Tabla 1. Métodos de Investigación.

Objetivos Específicos	Métodos
1. Fundamentar los conceptos relacionados con la Gestión del Conocimiento y la ingeniería	Análisis Histórico Lógico (Teóricos)

ontológica, así como sus técnicas, metodologías, y herramientas	
2. Definir la estructura organizativa de la información generada en el procedimiento del DPSW.	Observación (Empíricos) Grupos de Discusión (Empíricos)
3. Desarrollar la Ontología para modelar y utilizar la información generada en el procedimiento del DPSW.	Modelación (Teóricos) Grupos de Discusión (Empíricos) Análisis Síntesis (Teóricos)
<ul style="list-style-type: none"> Validar la propuesta ontológica en función de la comunicación, la organización de la información y la calidad del procedimiento de pruebas. <p>4.</p>	Hipotético Deductivo (Teóricos) Experimentación (Empíricos)

Valor Práctico.

La propuesta ontológica brindará la posibilidad de organizar el conocimiento y compartir la comprensión común de la información entre especialistas del departamento de Pruebas de Software, estudiantes y profesores involucrados en la docencia, eliminando confusiones de conceptos y términos empleados.

Estructura

El trabajo está conformado por 3 capítulos: el primero de ellos dirigido a la fundamentación teórica y estado del arte, el segundo enfocado plasmar la propuesta de solución y el tercero esencialmente recogerá la propuesta de validación de la ontología.

- Capítulo I: Dirigido a la fundamentación teórica sobre el tema de Gestión del Conocimiento, la Ingeniería Ontológica con sus herramientas y conceptos más importantes.

- Capítulo II: Se describe el desarrollo de la propuesta con cada uno de sus aspectos y detalles a través de la metodología seleccionada para la creación de la ontología
- Capítulo III: Estará destinado a la validación de la solución propuesta, respondiendo a las preguntas de competencias definidas, a través de una encuesta realizada a estudiantes y profesores del DPSW, y utilizando el validador web W3C.

Capítulo 1: Fundamentación Teórica.

Introducción al Capítulo

El objetivo fundamental de este capítulo es presentar de manera detallada los conceptos relacionados la Gestión del Conocimiento y Ontologías. En el capítulo se tratarán los principales conceptos, metodologías, herramientas, lenguajes, en los que se enmarca el dominio de nuestro problema.

1.1 Gestión del Conocimiento.

La representación del conocimiento es un mecanismo usado para razonar sobre el mundo, en vez de actuar directamente sobre él. En este sentido, la representación es fundamentalmente un sustituto de lo que se representa, es decir, un modelo de aproximación a un fenómeno en el mundo y ese modelo se construye a través de un lenguaje de representación el cual puede ser más o menos cercano al lenguaje natural, o él mismo.[4]

El conocimiento está en la mente de las personas y es información combinada con experiencia, contexto, interpretación y reflexión, aplicado a decisiones y acciones. Las organizaciones lo consideran, hoy en día, como activo y riqueza de la nueva economía, dándole la mayor importancia a su creación, transferencia y uso eficiente. El conocimiento es uno de los activos claves en las empresas comerciales actuales, así como en la sociedad moderna, se ha identificado como un elemento clave de las organizaciones para lograr ventajas competitivas. [5] El conocimiento no siempre está disponible cuando es necesario para la organización. Para tratar este problema surge la GC cuya principal misión es crear un ambiente en el que el conocimiento y la información disponible en una organización sean accesibles a todos, permitiendo ser usados para estimular la innovación y mejorar las decisiones. [6]

Una definición más sistemática define la Gestión de Conocimiento en un proceso integrador en el que convergen la gestión de la información, la tecnología y los recursos humanos y su implementación se orienta a perfeccionar los procesos de mayor impacto, mejor explotación del conocimiento en función de los procesos y su distribución en toda la organización, sobre la base del uso intensivo de las redes y las tecnologías. [7]

Cuando se hace referencia a la GC no se está hablando de gestión de la información (GI). “El objetivo básico de esta última es organizar y usar los recursos de información de la organización para permitirle operar, aprender y adaptarse a los cambios del ambiente. La GI constituye el fundamento de la fase inicial de la gestión del conocimiento”. [8].

Resulta importante aclarar que existen diferencias entre conocimiento, datos e información. Los datos son una representación simbólica que por sí solo no tiene un valor semántico y constituyen la base de la pirámide del conocimiento. Al conjunto de datos organizados y analizados en un contexto determinado se le denomina información, que no es lo mismo que conocimiento. Recopilar y organizar datos, es algo que puede hacer un software informático.

El conocimiento es un paso adelante, es identificar, estructurar y sobre todo utilizar la información para obtener un resultado. Requiere aplicar la intuición y la sabiduría, propios de la persona a la información. La capacidad de interpretar esos datos es lo que provoca que la información se convierta en conocimiento. Los tres términos suelen utilizarse indistintamente y esto puede llevar a una interpretación libre del concepto de conocimiento. Quizás la forma más sencilla de diferenciar los términos sea pensar que los datos están localizados en el mundo y el conocimiento está localizado en agentes de cualquier tipo (personas, empresas, máquinas...), mientras que la información adopta un papel mediador entre ambos.

En una organización, aparte de las personas y de los objetos que se utilizan, también existen otros aspectos que contienen conocimiento, como por ejemplo los procesos.

1.1.1 Procesos en la Gestión del Conocimiento

En la gestión del conocimiento hay dos procesos fundamentales cada uno de ellos se subdivide en otros, pero los básicos son dos: uno es la creación de conocimiento y el otro, la transmisión de conocimiento. La transmisión puede darse desde muchos puntos de vista y de muchas maneras, incluso en el espacio y en el tiempo. Cuando se intenta poner de forma explícita el conocimiento en una base de datos, lo que se está haciendo, es ponerlo allí para que al cabo de un tiempo alguien pueda recogerlo; en cierta medida,

pues, se está transmitiendo en el tiempo. Y cuando se utilizan herramientas de comunicación se intenta transmitir el conocimiento en el espacio. [9]

Además, estos dos procesos que pueden pensarse por separado también están totalmente interrelacionados, porque la creación de conocimiento no parte de la nada, sino que para crear conocimiento se utiliza conocimiento que proviene de otras personas y de otros lugares por lo tanto, ha habido un proceso de transmisión previo. Son procesos que están muy interrelacionados y que juntos hacen que el conocimiento dentro de las organizaciones mejore y se utilice.

1.1.2 Estrategias de Gestión del Conocimiento.

Entre los principales aspectos a atender como parte de una estrategia para la Gestión del Conocimiento, se pueden destacar: [10]

- La creación de comunidades de conocimientos por ramas afines del saber y transdisciplinar que garanticen la relación humana necesaria para alcanzar un flujo eficaz de conocimientos (en las universidades existen todas las condiciones para ello, pero es necesario actuar conscientemente en ese sentido).
- Disponer de una Intranet eficiente para el intercambio de información, conocimientos y experiencias.
- Actualización sistemática de la información necesaria para agilizar y hacer más efectiva la toma de decisiones.
- Disponer de un repositorio de información -- de conocimiento-- al que tengan acceso todos los trabajadores complementados con herramientas de comunicación para intercambiar experiencias, conocimientos y casos de estudio.

1.2 Ontología

1.2.1 Conceptos relacionados.

La ontología es una antigua disciplina que se define como un esquema específico de categorías que refleja una visión específica del mundo. Desde el punto de vista informático, las ontologías especifican un vocabulario relativo a un cierto dominio. Las ontologías se encargan de definir los términos utilizados para describir y representar un área de conocimiento. Este concepto tiene sus orígenes en la Filosofía, no hay una definición aceptada unánimemente para el concepto de ontología, sino que cada una de las definiciones que aparecen en la literatura aporta diferentes y a la vez complementarias visiones del significado de esta palabra. Las definiciones de ontología en Inteligencia Artificial (IA) son similares a la interpretación del filósofo Quine, quien dijo que todo lo que puede ser cuantificado existe. La primera definición de ontología en Inteligencia Artificial describía: “Una ontología define los términos básicos y relaciones que conforman el vocabulario de un área específica, así como las reglas para combinar dichos términos y las relaciones para definir extensiones de vocabularios”. [12]

Otras de las definiciones por Gruber en 1993 especifica que: [13]

“Una ontología es una especificación explícita de una conceptualización”.

El término proviene de la filosofía, donde una ontología es un recuento sistemático de la existencia. En sistemas de Inteligencia Artificial, lo que existe es lo que puede ser representado. Esos conjuntos de objetos, y las relaciones que se establecen entre ellos, son reflejados en un vocabulario con el cual representamos el conocimiento en un sistema basado en conocimiento. Así, en el contexto de IA, podemos describir la ontología de un programa como un conjunto de términos. En tal ontología, las definiciones asocian nombres de entidades del universo del discurso con textos comprensibles por los humanos que describen el significado de los nombres, y axiomas formales que limitan la interpretación y buen uso de dichos términos. Formalmente, una ontología es una teoría lógica. [14]

Las promesas que las Ontologías pueden resolver son amplias, pero se debe tener en cuenta que estas, no son las indicadas para todos los problemas. Existen situaciones en

las que la construcción de una ontología para una tarea específica, es más difícil o más costosa que la solución de la tarea sin la ontología.

El uso de las Ontologías ha sido resumido por Gruninger y Lee por los siguientes aspectos: [15]

- Comunicación: Compartir la comprensión común de la estructura de información entre personas o agentes de software, lo que debe revertir de forma positiva y casi necesaria en la extracción y recuperación de información, en páginas web, de contenidos conectados temáticamente.
- Permitir la reutilización del conocimiento perteneciente a un dominio.
- Permitir hacer explícitos los supuestos de un dominio.

1.2.2 Tipos de Ontología.

Los autores que se refieren al tema de las ontologías y su clasificación no establecen una única tipología. Los parámetros para agruparlas varían de un autor a otro aunque en algunos casos coinciden.

Uno de los parámetros para clasificar las ontologías se refiere al problema que solucionan. Es aportado por Mizoguchi, Vanwelkenhuysen e Ikeda quienes agrupan bajo este enfoque a las ontologías de [16]:

- De contenido: Permiten reutilizar el conocimiento de un dominio en otros sistemas y aplicaciones.
- De indización: Utilizadas en la recuperación de información.
- De comunicación: Facilitan el proceso de comunicación a través de repuestas a preguntas concretas.
- Meta-Ontologías: Usadas para representar las ontologías de dominios similares o diferentes.

Según Van Heist las ontologías también pueden clasificarse de acuerdo a la cantidad y tipo de estructura de la conceptualización en [17]:

- Ontologías terminológicas, lingüísticas: Especifican los términos usados para representar conocimiento en el dominio.
- Ontologías de información: Especifican la estructura de los registros de la base de datos. Los esquemas de bases de datos serían un ejemplo.
- Ontologías para modelar conocimiento: Especifican conceptualizaciones de conocimiento. Estas ontologías tienen una estructura interna mucho más rica que los anteriores tipos de ontologías, y éstas son las ontologías que interesan a los desarrolladores de sistemas basados en conocimiento.

Una clasificación alternativa fue propuesta por Mizoguchi, donde también se proponen tres categorías [18]:

- Ontologías del dominio: Contienen todos los conceptos asociados a un dominio particular.
- Ontologías de tarea: Establecen la forma en la cual se puede usar el conocimiento del dominio para realizar tareas específicas. De esta forma, una aplicación podría realizar búsquedas de información mientras otra podría gestionar la asignación de bloques libre de memoria.
- Ontologías generales: Contienen descripciones generales sobre objetos, eventos, relaciones temporales, relaciones causales, modelos de comportamiento y funcionalidades.

1.2.3 Componentes de las Ontologías.

Generalizando, se puede afirmar que una Ontología no es más que otra forma de representación del conocimiento, este se formaliza principalmente usando cinco tipos de componentes: conceptos, relaciones, funciones, instancias y axiomas. [19]

- Conceptos: son las ideas básicas que se intentan formalizar. Pueden ser clases de objetos, métodos, planes, estrategias, procesos de razonamiento, etc. El conjunto de conceptos identificados es denominado universo del discurso, es decir, conjunto de objetos que representan el conocimiento de un dominio a través de un formalismo declarativo.

- **Instancias:** Contienen elementos o datos que describen o ejemplifican un concepto. Por lo general son propiedades que no se recogen en el concepto o nociones que sirven para aclarar la connotación de este en un contexto determinado.
- **Relaciones:** Son las relaciones que establecen el tipo de interacción semántica entre los conceptos. Entre las principales relaciones semánticas que pueden ser establecidas se encuentran las relaciones lógicas.
- **Funciones:** Constituyen un tipo especial de relación, a través de las cuales, según se identifica un elemento mediante el cálculo de una función que considera varios elementos de la ontología. Por ejemplo, pueden aparecer funciones como categorizar-clase, asignar-fecha, etc. Este componente es imprescindible cuando las ontologías son usadas para modelar sistemas y procesos.
- **Axiomas:** Son teoremas que se declaran sobre relaciones que deben cumplir los elementos de la ontología. Son los elementos que permiten hacer inferencias que no están explícitas en la taxonomía de conceptos. Una inferencia es un proceso mental por el cual se extraen conclusiones a partir de premisas más o menos explícitas.

Cada componente de la ontología tiene una implicación en la gestión del conocimiento. Así, los conceptos, las instancias y las relaciones entre los conceptos pueden representar el conocimiento tácito de los integrantes de la organización. Las funciones son muy usadas para describir los pasos para desarrollar un proceso. Y los axiomas, permiten hacer inferencias, lo que es de gran utilidad para la toma de decisiones.

1.2.4 Lenguajes para la representación de Ontologías.

Dentro de los principales lenguajes de Ontologías se destacan los siguientes:

- **SHOE:** Simple HTML Ontology Extensions. Fue el primer lenguaje de etiquetado para diseñar Ontologías en la Web. Este lenguaje nació antes de que se ideara la Web Semántica. Las Ontologías y las etiquetas se incrustaban en archivos HTML. Este lenguaje permite definir clases y reglas de inferencia, pero no negaciones o disyunciones. A su albur se desarrollaron muchos editores, buscadores, APIs; el

proyecto fue abandonado a medida que se desarrollaron OIL y DAM; existiendo una socialización de este lenguaje en XML. [20]

- OIL: Ontology Inference Layer. Este lenguaje, derivado en parte de SHOE, fue impulsado también por el proyecto de la Unión Europea On-To-Knowledge. Utiliza la sintaxis del lenguaje XML y está definido como una extensión de RDFS. Se basa tanto en la lógica descriptiva (declaración de axiomas) como en los sistemas basados en frames (taxonomías de clases y atributos). OIL posee varias capas de sub-lenguajes, entre ellas destaca la capa base que es RDFS, a la que cada una de las subsiguientes añade alguna funcionalidad y mayor complejidad. La principal carencia de este lenguaje es la falta de expresividad para declarar axiomas. [21]
- OWL: Web Ontology Language o Lenguaje de Ontologías para la Web es un lenguaje de etiquetado semántico para publicar y compartir Ontologías en la Web. Se trata de una recomendación del W3C, y puede usarse para representar Ontologías de forma explícita, es decir, permite definir el significado de términos en vocabularios y las relaciones entre aquellos términos (Ontologías). En realidad, OWL es una extensión del lenguaje RDF y emplea las tripletas de RDF, aunque es un lenguaje con más poder expresivo que éste. Se trata de un lenguaje diseñado para usarse cuando la información contenida en los documentos necesita ser procesada por programas o aplicaciones, en oposición a situaciones, donde el contenido solamente necesita ser presentado a los seres humanos. 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.). Existen 3 sublenguajes de OWL, los cuales van creciendo respecto al nivel de expresión: [21]
 1. OWL Lite: Útil para la creación de jerarquías y restricciones simples, sólo permite valores de cardinalidad 0 y 1, pierde en expresividad.
 2. OWL DL: (Description Logic), es el lenguaje más sencillo e indicado para los usuarios que requieren el máximo de expresividad y decibilidad (todos los cálculos acaban en un tiempo finito). Una clase puede ser a la vez subclase de muchas clases, no puede ser una instancia de otra clase.

3. OWL Full: Máximo nivel de expresión y la libertad sintáctica de RDF. Permite expresiones de segundo orden. Por ejemplo, una clase puede ser tratada simultáneamente como una colección de individuos y como un individuo por sí mismo. Es el más completo por lo que se necesita mucho poder computacional para poder hacer inferencias es por eso que se dice que no tiene garantía computacional.

Se propone como lenguaje de representación de ontologías el OWL en su versión DL ya que es un lenguaje diseñado para usarse en el procesamiento de la información contenida en el modelo ontológico cuando es procesada por programas o aplicaciones de software brindando propiedades avanzadas para el proceso de inferencia.

1.2.5 Herramientas para la construcción de Ontologías.

Para la construcción de las ontologías se tuvieron en cuenta varias de herramientas:

- Protégé 2000: herramienta a través de la cual el usuario puede construir ontologías de dominio, generar usuarios de entrada de datos y efectuar la propia entrada de datos. Es una herramienta que permite acceso a aplicaciones externas basadas en conocimiento. Además es una biblioteca a la que otras aplicaciones pueden acceder, permitiéndoles acceder a las bases de conocimiento de las cuales se dispone. Esta herramienta utiliza un lenguaje muy flexible para ser utilizado en la Web, por eso se ha acuñado el término de Web Semántica, es decir, que facilitan la tarea no sólo de ser manejable a través de la Web, sino que hacen una descripción semántica de la información, con lo que todo el sistema es menos rígido y por tanto se hace más flexible y potente. Protégé reconoce por tanto, Frames, XML Schema, RDF Schema y OWL, que son lenguajes semánticos utilizados en la Web, en contraposición a la rigidez del HTML. Protégé permite realizar programas en OWL con una gran facilidad, debido a su entorno gráfico, tan sólo con el uso de otro programa asociado a él que se denomina Protégé-OWL Plugin. Este es una especie de anexo a Protégé para poder trabajar con el lenguaje OWL [22].

Este editor incluye además SPARQL, ofrece a los desarrolladores y usuarios finales un camino para presentar y utilizar los resultados de búsquedas a través de

una gran variedad de datos, los cuales pueden ser personales, redes sociales y metadatos sobre recursos digitales como música e imágenes, siendo útil para recuperar y organizar la información.

- TopBraid Composer™ es una herramienta de modelado para la creación y mantenimiento de las ontologías. Es un completo Editor para modelos RDF y OWL. Está desarrollado bajo la plataforma Eclipse, usando Jena como API. Históricamente el desarrollo de TopBraid Composer™ ha tenido sus raíces en Protégé, por lo que muchas características de este, son similares a los de Protégé. [23]
- OilEd: Basado inicialmente para el desarrollo de Ontologías OIL y DAML + OIL se han ido realizando numerosas actualizaciones para que acepte la mayoría de los lenguajes de especificación actuales. Es un editor bastante utilizado por los investigadores porque aporta la posibilidad de interactuar con un razonador como FACT o RACER los cuales permiten comprobar la consistencia de una Ontología. Una de las desventajas que presenta este editor es la carencia de recursos para soportar Ontologías grandes, migración e integración de otras Ontologías y diferenciación de versiones [24].

Se propone como herramienta de modelado para la construcción de la ontología el editor Protégé, ya que permite realizar ontologías con extensión OWL y RDF, debido a su entorno gráfico basado en plugins. Además tiene a su favor una gran comunidad de desarrolladores que brindan una bibliografía amplia y actualizada, así como un entorno extensible que se adapta a la construcción de numerosos tipos de ontologías.

1.2.6 Metodologías para la construcción de ontologías.

Las metodologías proporcionan un conjunto de directrices que indican cómo hay que llevar a cabo las actividades identificadas en el proceso de desarrollo, qué técnicas son las más apropiadas en cada actividad y qué produce cada una de ellas.

Para guiar el proceso de desarrollo de ontologías existen diferentes tipos de metodologías [25]:

- **Metodología Kactus:** Esta metodología centra la construcción de la ontología sobre una base de conocimiento, mediante un proceso de abstracción. Las secuencias de sus pasos son incremental con prototipos de desarrollo. Para ello hace uso de:
 - ✓ Especificación del contexto de la aplicación y el punto de vista de modelado. Lo primero hace referencia a la descripción del dominio que tendrá la aplicación, así como también los objetos de interés y tareas que realizará la ontología, mientras que el punto de vista de modelado se refiere a definir qué tipo de modelado se va a realizar: dinámico-estático, funcional-causal.
 - ✓ Realizar un diseño preparatorio sobre la base de una ontología existente, lo que implica realizar un estudio de ontologías que se hayan construido (mapeo).
 - ✓ Refinamiento y estructuración de la ontología.
 - ✓ Finalmente la documentación y reutilización de la ontología.
- **Metodología Cyc:** nace como un proyecto de inteligencia artificial que busca la construcción de una ontología comprensible para habilitar el razonamiento humano. El ciclo de vida propuesta es secuencial. Presenta características como:
 - ✓ Extracción manual del conocimiento común (de diversas fuentes)
 - ✓ Utilización de herramientas de procesamiento de lenguaje natural o aprendizaje natural para la adquisición de nuevo conocimiento en la ontología.
- **Metodología Methontology:** Es una metodología creada en el Laboratorio de Inteligencia Artificial de la Universidad Técnica de Madrid. El ciclo de vida es secuencial guiado por prototipos de desarrollo y por actividades. La creación de la ontología puede empezar desde cero o en base a la reutilización de otras existentes. Sus pasos son [26]:
 - ✓ Especificación: Consiste en delimitar los objetivos de su creación, se decide el dominio de actuación de la ontología, quién la usará y para qué, las preguntas a las que deberá responder y quién se encargará de su mantenimiento

(decidiendo si se limitará a introducir nuevas instancias, se permitirá la modificación de conceptos o atributos, etc.

- ✓ **Conceptualización:** Consiste en crear un glosario de términos que pertenecen al dominio, definirlos y crear una taxonomía (estableciendo una clasificación o jerarquía entre los conceptos, sus niveles, las relaciones entre ellos, sus instancias, sus propiedades o atributos, e igualmente los axiomas o reglas).
- ✓ **Formalización:** Proceso que consiste en convertir el modelo anterior en un modelo formal o semi-computable. Se puede emplear en este paso una herramienta como Protégé.
- ✓ **Implementación:** Convierte el modelo formalizado en un modelo computable mediante un lenguaje para construcción de ontologías. Se puede emplear en este paso una herramienta como Protégé.
- ✓ **Mantenimiento:** Labor que puede acarrear desde el borrado de instancias ya inútiles o la incorporación de nuevas instancias que se han ido produciendo con el tiempo, hasta las tareas de introducción de cambios en el contenido de la información, ya sea redefiniendo atributos, relaciones o incluso conceptos.

La Metodología que se usará es la METHONTOLOGY esta cuenta con una gran comunidad de usuario, es una de las ontologías que más completas están ya que divide sus actividades desde la especificación hasta el mantenimiento de manera secuencial.

1.2.7 Razonadores.

En la actualidad, son varios los razonadores o sistemas deductivos basados en lógica descriptiva que permiten el razonamiento y la inferencia en las Ontologías. Los principales son:

- **FaCT++:** Es un razonador DL que ha sido desarrollado bajo el proyecto europeo WonderWeb. Está implementado en C++ y corresponde a una nueva versión del razonador DL FaCT. Para realizar inferencia implementa nuevas características y optimizaciones, permitiendo adicionar nuevas tácticas de razonamiento y capacidad de razonar con lógicas descriptivas más potentes y cercanas a la expresividad de

OWL. Entre las ventajas que ofrece FaCT++ están: posee licencia GPL. Además Protégé 4.0.1 lo trae incluidos en sus plugin. [27]

- **HermiT**: es un razonador para ontologías en lenguajes OWL. Dado un fichero OWL, HermiT puede determinar si la ontología es o no consistente. HermiT es el primer razonar OWL, públicamente accesible, es el primer razonador capaz de clasificar un conjunto de ontologías que previamente habían sido consideradas demasiado complejas de gestionar por cualquiera de los sistemas disponibles.[28]
- **Racer** es un razonador DL para la lógica descriptiva SHIQ. Su nombre comercial es RacerPro. Racer, no tiene licencias libres que permitan su utilización en EzWeb, lo que lo descarta como posible motor de inferencia para EzWeb. Soporta OWL DL excepto para los nominales y para tipos de datos no estándar; maneja largas ABoxes en combinación con largas y expresivas Tboxes. [29]

Como razonador se escogió FacT++, este posee licencia GPL, presenta soporte de OWL y viene como plugin de Protégé.

1.2.8 Sistemas de Almacenamiento

Una ontología es una opción muy adecuada para los sistemas de almacenamientos, ya que es un formalismo muy extendido para modelar y almacenar conocimiento. Sin embargo, sin otras herramientas, de una ontología no es posible extraer más conocimiento que los conceptos almacenados en ella. Aun así, su organización permite un razonamiento sencillo sobre ella utilizando un razonador. Estas herramientas permiten, dada una ontología, razonar nuevas relaciones o conceptos no explícitos en el modelo original.

Dentro de los componentes de las Ontologías que mayor importancia merecen para la creación de los sistemas de almacenamiento están:

- Clases e instancias, que son de forma básica los objetos o elementos que componen la ontología.
- Relaciones, que representan las interacciones entre los anteriores.
- Axiomas, que se emplean para modelar el conocimiento que no puede recogerse a través de los elementos anteriores.

Los sistemas de almacenamiento posibilitan mantener las Ontologías en bases de datos e ir añadiendo nueva información, y con la ayuda de razonadores probar la consistencia de la Ontología.

Dentro de los Sistema de almacenamiento para las Ontologías se encuentran:

- **Jena** es un framework desarrollado por los laboratorios de HP programado en Java. Su fin es manipular metadatos desde aplicaciones escritas en Java. Proporciona una API para extraer y escribir datos de un grafo RDF. Los modelos pueden ser consultados mediante SPARQL. Se puede utilizar OWL con Jena. Proporciona varios razonadores internos y se pueden añadir otros mediante una interfaz para Descripción Lógica (DIG). [30]
- **Sesame** es un repositorio para RDF-Schema que permite añadir y eliminar información, para ser almacenada en cualquier tipo de base de datos (MySQL, Oracle etc.). Soporta los lenguajes de consulta RQL, RDQL y SeRQL, para acceder al conocimiento. [31]
- **KAON Tool** Implementa una interfaz independiente del sistema en el que se almacenarán las Ontologías, ya sean cualquier base de datos o un fichero texto. Implementa un API para leer las descripciones de los recursos, emplea RQL para realizar consultas y soporta tanto Ontologías DAML + OIL como RDF. [32]

Se determinó escoger Jena porque provee un entorno de desarrollo para RDF, OWL, SPARQL y RDFS, además cuenta con un motor de inferencia basado en reglas.

1.2.9 IDE de desarrollo.

Un entorno de desarrollo integrado o IDE es un programa informático compuesto por un conjunto de herramientas de programación. Un IDE es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica.

- **NetBeans** es una herramienta para programadores pensada para escribir, compilar, depurar y ejecutar programas. Está escrito en Java y soporta el desarrollo de todos los tipos de aplicación Java (J2SE, web, EJB y aplicaciones

móviles). NetBeans Beta proporciona herramientas para la programación en el lenguaje Java e incluye resaltado de sintaxis, auto-completado de código, templates. Provee soporte para librería Jena y cuenta con un módulo de *subversion* para facilitar la manipulación de las versiones del código durante la implementación. [33]

- **Eclipse** es una plataforma universal para integrar herramientas de desarrollo, con una arquitectura abierta y basada en plug-ins. Además, Eclipse da soporte a todo tipo de proyectos que abarcan desde el ciclo de vida del desarrollo de aplicaciones, incluyendo soporte para modelado. [34]

Dentro de sus principales características se encuentran:

- ✓ Editor visual con sintaxis coloreada
- ✓ Compilación incremental de código
- ✓ Resaltado de sintaxis
- ✓ Compilación en tiempo real
- ✓ Pruebas unitarias con Junit
- ✓ Control de versiones con CVS
- ✓ Integración con AntAsistentes (Wizards).

Como IDE de desarrollo se decidió escoger NetBeans este es un editor de código sensible al contenido. Con soporte para autocompletar el código, coloreado de etiquetas, auto-tabulación y uso de abreviaturas para varios lenguajes de programación, presenta soporte para Java, C, C++, XML y lenguajes HTML, y para el API Jena, además de incluir Control de versiones.

1.2.10 Reutilización de Ontologías.

Durante el desarrollo de la ontología es importante analizar la posibilidad de la reutilización o cualquier otra fuente de información existente en el domino seleccionado. A pesar de que existen varias propuestas de Investigación relacionadas con el tema [24, 31, 35], no

se encontró una ontología para pruebas de software disponible en la red que esté estrechamente vinculada con el dominio del procedimiento de pruebas del Departamento de Pruebas de Software.

Dentro de las consultadas:

Los Autores Katia Cristina Duarte y Ricardo de Almeida Falbo en [35], proponen el desarrollo de una ontología basada en un dominio de calidad de software, específicamente los elementos relacionados con las pruebas de software se limitan a mencionar las características de calidad donde estas están incluidas. Las preguntas de competencias solo proporcionan información con las métricas y las características de calidad.

Por su parte Ellen Francine Barbosa, Elisa Yumi Nakagawa and José Carlos Maldonado *en* [31], proponen OntoTest, una ontología de pruebas de software usada en una arquitectura de referencia que soporta el desarrollo de herramienta automatizadas para actividades de ingeniería de software. OntoTest intenta explorar los diferentes aspectos que están involucrados en las actividades de pruebas, técnicas, criterios, recursos humanos y tecnológicos, y herramientas automatizadas. Como principal dificultad para reutilizar esta propuesta está que los autores no proporcionan código generado, mapas conceptuales completos, descripciones de clases, propiedades, instancias.

Otra propuesta es la de Ing. Adriana Querro, [24] de la Universidad de los Andes de Venezuela, en su tesis de maestría describe la creación de una ontología basada en la guía SWEBok, la cual contiene información relacionada con un conjunto de procedimientos, técnicas, y ayudas de documentación para el desarrollo de productos de software. Esta propuesta dista del dominio que se requiere en la presente investigación, ya que solo hace énfasis en los macro procesos del desarrollo de software.

1.3 Conclusiones Parciales

Se pudo constatar que en la bibliografía consultada y en el estado del arte presentado, no se encontró una propuesta ontológica que cumpliera con las necesidades propias para el dominio especificado. En varios casos éstas están enmarcadas en un dominio más general o en ocasiones no proporcionan el código generado de la ontología. De igual manera en este capítulo se fundamentaron los principales conceptos en los que se enmarca el dominio de nuestro problema, describiéndose y escogiéndose los lenguajes,

las metodologías, herramientas, características de los sistemas de almacenamientos e IDE de desarrollo que se utilizarán a lo largo de la investigación.

Capítulo 2. Diseño de la Solución.

Introducción al Capítulo.

En este capítulo se definirá la estructura organizativa de la información presente en el procedimiento de pruebas de software del DPSW y se desarrollará el diseño de la solución que será guiado por la Metodología METHONTOLOGY estableciendo 4 actividades por donde se desarrollará la ontología, se describirán las preguntas de competencias, las especificaciones de los conceptos, relaciones e instancias.

2.1 Descripción de la Información presente del DPSW.

Para el desarrollo de la Ontología se escogió como Base el “Software Engineering Body of Knowledge” SWEBOK, la ISO 9126 y el procedimiento del DPSW. Se recopiló todo el conocimiento presente en el dominio de pruebas de software enmarcándolo en los procesos de liberación y aceptación del DPSW, ver figura 1. A partir de este conocimiento se define el dominio de la presente ontología, y obtienen los diferentes conceptos y relaciones que son de interés para el área en cuestión a partir de aquí se crean las instancias que están presente en cada uno de estos conceptos. Se obtuvieron un total de 49 conceptos de ellos 29 relacionados con el Procedimiento de pruebas del DPSW, 13 provenientes de SWEBOK, y 7 de la norma ISO 9126.

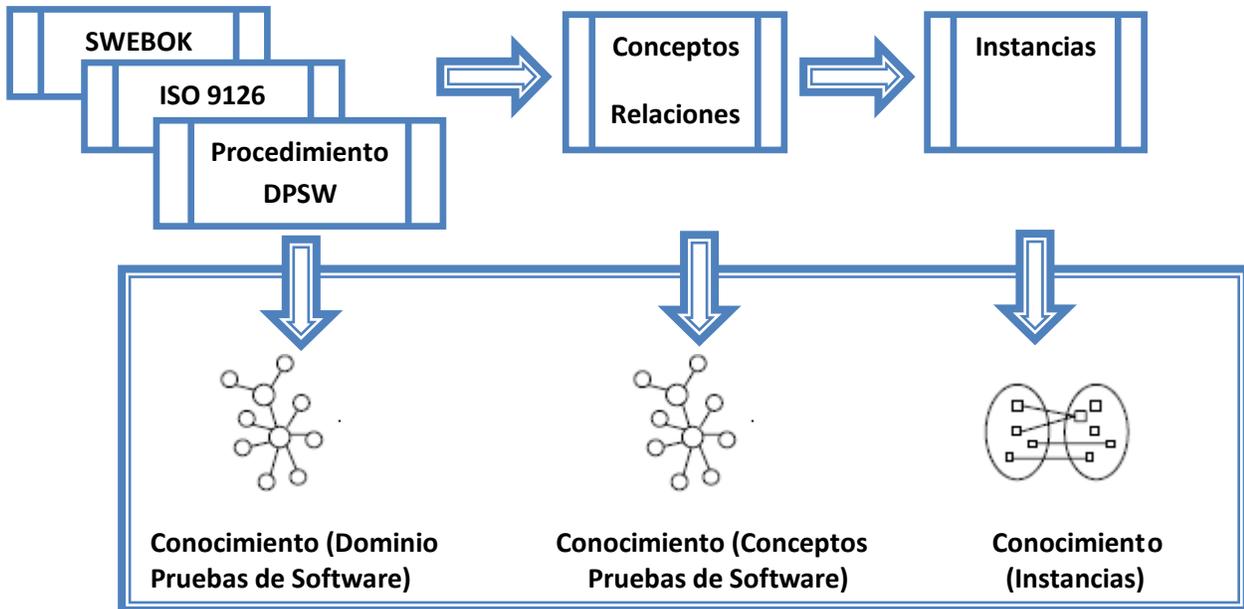


Figura 1. Captura de Conocimiento para el desarrollo de la Ontología.

A partir de toda la información generada en la descripción anterior se definieron un conjunto de clases y relaciones que se describen a continuación en la Figura 2.

El departamento de pruebas de software es quien lleva a cabo las pruebas de liberación y aceptación a los artefactos generados en el desarrollo de software. A todos los artefactos se le aplican evaluaciones, éstas pueden ser de dos tipos: Evaluaciones Dinámicas y Evaluaciones Estáticas, cada evaluaciones tendrá una Estrategia que a su vez tiene Criterios de Criticidad, Actividades, Documentación, y Niveles de Pruebas. Los tipos de Pruebas son ejecutados en cada uno de los niveles. Los Tipos de Pruebas presentan Herramientas y técnicas de pruebas. Las herramientas son coordinadas por un Especialista y estos coordinan los Proyectos. Por su parte los Artefactos presentan No Conformidades, Tipos de Pruebas, y Evaluaciones. Ver **Figura 2**.

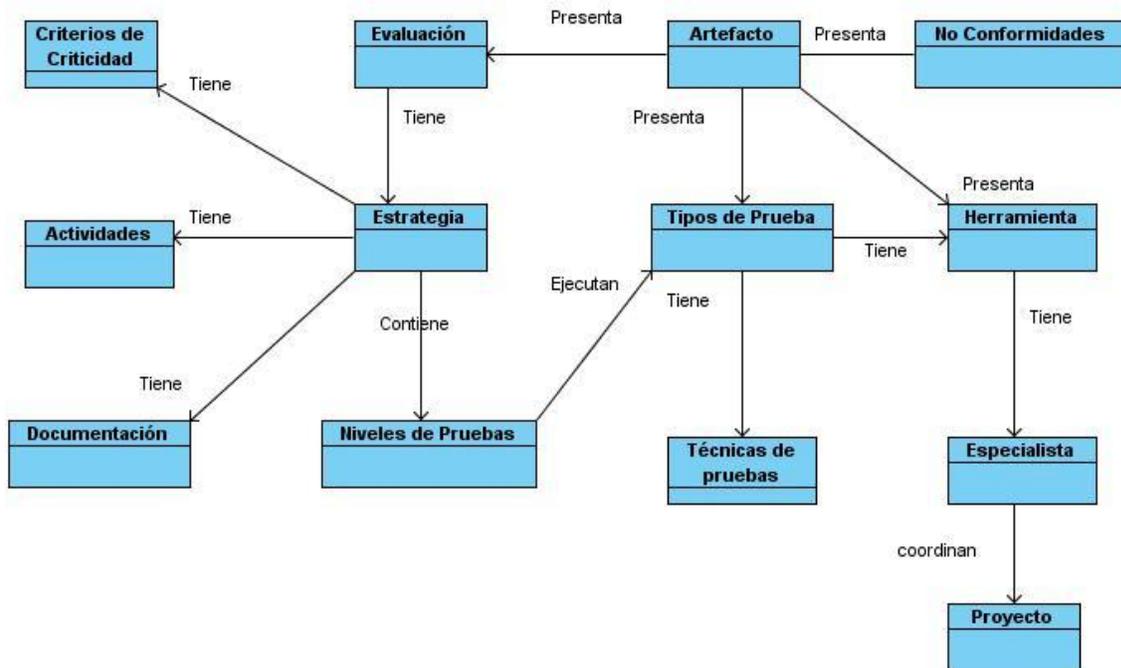


Figura 2. Parte de la Estrategia de Evaluación.

2.2 Definición y desarrollo de la Ontología. Metodología METHONTOLOGY.

METHONTOLOGY proporciona guías sobre cómo llevar a cabo el desarrollo de la ontología a través de las actividades de especificación, conceptualización, formalización, implementación y mantenimiento.

- Especificación: permite determinar por qué se construye la ontología, cuál será su uso, y quiénes serán sus usuarios finales.
- Conceptualización: se encarga de organizar y convertir una percepción informal del dominio en una especificación semi-formal, El resultado de esta actividad es el modelo conceptual de la ontología.
- Formalización: se encarga de la transformación de dicho modelo conceptual en un modelo formal o semicomputable.
- Implementación: construye modelos computables en un lenguaje de ontologías (Ontolingua, RDF Schema, OWL.). La mayor parte de las herramientas de ontologías permiten llevar a cabo esta actividad.
- Mantenimiento: se encarga de la actualización y/o corrección de la ontología, en caso necesario.

2.2.1 Actividad: Especificación.

Esta actividad consiste en determinar el alcance, objetivos, y usuarios finales de la ontología, así como las preguntas de competencias a las que deberá responder la ontología una vez concluida.

2.2.1.1 Departamento de Pruebas de Software de Calisoft.

El departamento de Pruebas de Software es el dominio de aplicación del presente trabajo, éste forma parte de los tres departamentos del Calisoft, es responsable de las pruebas de liberación y aceptación de software. La estructura organizativa del departamento se muestra en la figura 3. En el laboratorio Industrial de Pruebas de Software se ejecutan las pruebas, con probadores de 2do año de la carrera, a través de la asignatura de Práctica Profesional II. En el grupo de ingeniería y pruebas de software se encuentran los

especialistas que coordinan los determinados tipos de pruebas que se aplican en el departamento.

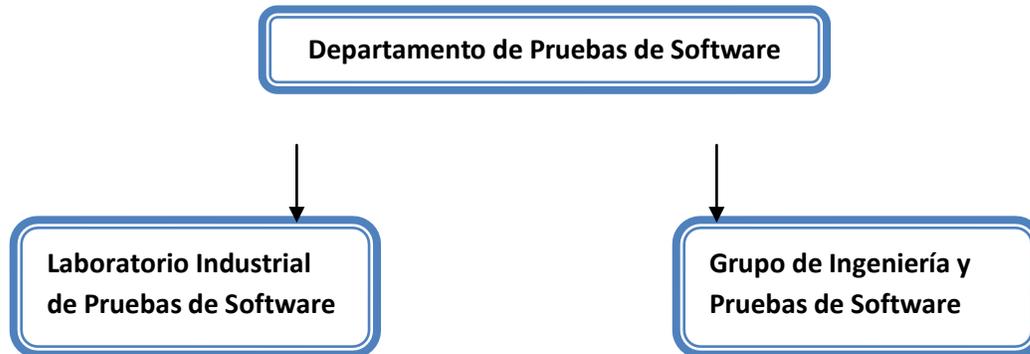


Figura 3. Estructura Organizativa del DPSW

El objetivo principal de la ontología está dado desde su utilización, a través de la misma, se contribuirá a organizar el conocimiento y la información generada en el procedimiento de pruebas del DPSW, favorecerá la comunicación entre especialista, probadores y la calidad del propio procedimiento. Los usuarios finales serán: probadores y especialista del DPSW.

2.2.1.2 Preguntas de Competencia.

Según Gruninger y Fox[10] una de las formas de determinar el alcance de la ontología es bosquejando una lista de preguntas que la base de conocimientos basada en la ontología debería ser capaz de responder. Estas preguntas servirán después como prueba de control de calidad: ¿La ontología contiene suficiente información para responder esos tipos de preguntas? ¿Las respuestas requieren un nivel particular de detalle o representación de un área particular? Las preguntas de competencia son solamente un bosquejo y no necesitan ser exhaustivas.

Las respuestas a esas preguntas pueden cambiar durante el proceso del diseño de la ontología, pero en cualquier momento dado ellas ayudarán a limitar el alcance del modelo. La propuesta ontológica debe responder a las siguientes preguntas:

1. ¿Qué proyecto ha coordinado un determinado especialista?
2. ¿Qué herramientas se le ha aplicado a un determinado proyecto?
3. ¿Qué tipo de No conformidades presenta un determinado artefacto?

4. ¿Qué tipo de evaluación presenta un determinado artefacto?
5. ¿Qué herramienta es coordinada por un determinado especialista?
6. ¿Qué criterios de criticidad presenta la estrategia del DPSW?
7. ¿Qué herramienta presenta un determinado tipo de prueba?
8. ¿Qué técnicas de pruebas utiliza un determinado tipo de prueba?
9. ¿Qué niveles de prueba presenta la estrategia de pruebas del DPSW?
10. ¿Qué documentación presenta la estrategia de prueba del DPSW?
11. ¿Qué actividades presenta la estrategia de pruebas del DPSW?
12. ¿Qué nivel de prueba presenta un determinado tipo de prueba?
13. ¿Qué especialista coordina un determinado tipo de prueba?
14. ¿Qué documentos están presentes en las actividades?
15. ¿Qué es: “Técnicas de Pruebas”?
16. ¿Qué es: “Herramientas Automatizadas”?
17. ¿Qué es: “Estrategia de Prueba”?
18. ¿Qué es: “Niveles de Pruebas”?
19. ¿Qué es: “No Conformidades”?
20. ¿Qué es: “Criterio de Criticidad”?

2.2.2 Actividad: Conceptualización.

Esta actividad consiste en organizar y convertir una percepción informal de un dominio en una especificación semi-formal usando un conjunto de representaciones intermedias (tablas, diagramas) que puedan ser entendidas por los expertos del dominio y los desarrolladores de ontologías. Para asegurar en cierta medida la consistencia y completitud de la ontología que se construye, METHONTOLOGY recomiendan realizar una serie de tareas en un orden determinado. A continuación se muestran las tareas

propuestas por METHONTOLOGY para la actividad de conceptualización, haciendo énfasis en los principales elementos especificados en la Ontología.

Tareas de Conceptualización:

1. Construcción del Glosario de Términos.
2. Construcción de la Taxonomía de Conceptos.
3. Construcción Relaciones Binarias.
4. Construcción del Diccionario de Conceptos.
5. Definir Axiomas.
6. Describir Instancias.

2.2.2.1 Glosario de términos.

Este proceso de encarga de la construcción de un glosario de términos que incluye todos los términos relevantes del dominio (conceptos, instancias, relaciones entre conceptos) y sus descripciones en lenguaje natural. La tabla 2 muestra el glosario de términos de los fundamentos de pruebas del Departamento de Pruebas de Software.

Tabla 2. Glosarios de términos.

No.	Término	Descripción	Tipo
1	Proyecto	Equipos de Desarrollo de Software, que intervienen en los procesos de Pruebas de Liberación y Aceptación del Departamento de Pruebas de Software.	Conceptos
2	Especialista	Persona encarga de coordinar las pruebas en el Departamento de Pruebas de Software.	Conceptos
3	Herramientas	Instrumento o recurso que se utiliza para realizar le ejecución de las pruebas: Software, Listas de chequeos.	Conceptos
4	Herramientas Manuales	Son utilizadas por los probadores para realizar pruebas manualmente sin la ayuda de un	Conceptos

		software.	
5	Herramientas Automatizadas	Son utilizadas por los probadores para realizar pruebas con la ayuda de un software.	Conceptos
6	Tipos de Pruebas	Clasificación de las Pruebas.	Conceptos
7	Pruebas Funcionales	Pruebas donde se verifican o validan los requisitos funcionales pactados en un principio en el desarrollo de software.	Conceptos
8	Pruebas No Funcionales	Pruebas donde se verifican o validan los requisitos no funcionales pactados en un principio en el desarrollo de software.	Conceptos
9	Artefactos	Entregable que generan en cada uno de las etapas del desarrollo del software	Conceptos
10	No Conformidades	Errores, Inconsistencia, inconformidades relacionadas con los requisitos de software.	Conceptos
11	Técnicas de Pruebas	Formas de cómo ejecutar las pruebas.	Conceptos
12	Técnicas Basada en Experiencia.	Pruebas realizadas basándose en las habilidades, intuición y en la experiencia de los probadores con sistemas y tecnología similares.	Conceptos
13	Técnicas Caja Negra	Pruebas que se llevan a cabo sobre la interfaz del software. El objetivo es demostrar que las funciones del software son operativas, que las entradas se aceptan de forma adecuada y se	Conceptos

		produce un resultado correcto, y que la integridad de la información externa se mantiene (no se ve el código).	
14	Técnicas Caja Blanca	Se comprueban los caminos lógicos del software Se puede examinar el estado del programa en varios puntos para determinar si el estado real coincide con el esperado.(sobre el código)	Conceptos
15	Niveles de Pruebas	Diferentes escenarios o momentos dentro del desarrollo del software en que una prueba puede realizarse.	Conceptos
16	Nivel de desarrollador	Este nivel está relacionado sobre quien las desarrolla.	Conceptos
17	Nivel de sistema	Este nivel está relacionado sobre que elemento del sistema actúa.	Conceptos
19	Estrategia	Describe el enfoque y los objetivos generales de las actividades de prueba.	Conceptos
20	Criterios de Criticidad	Elementos o bases para determinar el estado a un proyecto al presentar un número considerable de No Conformidades.	Conceptos
21	Criterios para Abortar	Elementos o bases para determinar cuando las pruebas de un proyecto están abortadas	Conceptos
22	Criterios para Detener	Elementos o bases para determinar cuando las pruebas de un proyecto están detenidas.	Conceptos

23	Evaluación	Revisión a un determinado artefacto.	Conceptos
24	Evaluaciones Estáticas	Evaluaciones realizadas a documentos.	Conceptos
25	Evaluaciones Dinámicas	Estas incluyen todas las pruebas realizadas a los diferentes artefactos.	Conceptos
26	Documentación	Documentos generados en cada una de las actividades de las pruebas	Conceptos
27	Actividades	Etapas por las que transita el procedimiento de pruebas.	Conceptos
28	presentadocumento	Relación para determinar los documentos que están presentes en la estrategia del DPSW.	Relación
29	escoordinadopor	Relación para determinar que proyecto es coordinado por un especialista.	Relación
30	estaennivel	Relación para determinar el nivel en que se encuentra una determinada prueba.	Relación
31	presentaactividad	Relación para determinar que actividades están presentes en la estrategia del DPSW.	Relación
32	presentadocumentacion	Relación para determinar que documentos están presentes en una actividad específica.	Relación
33	presentaniveles	Relación para determinar que niveles están presentes en la estrategia del DPSW.	Relación
34	utilizatecnica	Relación para determinar que técnicas de prueba utiliza un	Relación

		determinado tipo de prueba.	
35	presentaherramienta	Relación para determinar que herramientas presenta un tipo de prueba determinado.	Relación
36	presentacriterio	Relación para determinar cuales son los criterios de criticidad presente en la estrategia del DPSW.	Relación
37	coordinaherramienta	Relación para determinar que herramientas coordina un especialista.	Relación
38	presentaevaluacion	Relación para determinar que evaluaciones presenta un determinado artefacto.	Relación
39	presentaNc	Relación para determinar que tipos de No Conformidades presenta un determinado artefacto	Relación
40	aplicaherramienta	Relación para determinar a que proyecto se le ha aplicado una herramienta determinada.	Relación
41	coordinaproyecto	Relación para determinar que proyecto ha sido coordinado por un especialista determinado.	Relación
42	delvis	Instancia perteneciente a la clase Especialista.	Instancia
43	sageb	Instancia perteneciente a la clase Especialista.	Instancia
44	jmeter,	Instancia perteneciente a la clase Herramienta.	Instancia
45	appweb	Instancia perteneciente a la clase Artefacto.	Instancia
46	strategiadpsw	Instancia perteneciente a la clase	Instancia

		Estrategia.	
47	carga	Instancia perteneciente a la clase Tipo de prueba.	Instancia
48	cajanegra	Instancia perteneciente a la clase Tecnica de prueba.	Instancia
49	sistema	Instancia perteneciente a la clase Nivel de Prueba.	Instancia
50	casoprueba	Instancia perteneciente a la clase Documentación.	Instancia
51	solicitud	Instancia perteneciente a la clase Actividades.	Instancia

2.2.2.2 Taxonomía de Conceptos.

Las taxonomías representan a través de las clases y las instancias propias de esas clases una realidad contextualizada que permitirá la organización y recuperación efectiva de los conceptos representados pertenecientes a un determinado dominio. Para construir la taxonomía de conceptos, se seleccionan del glosario de términos aquellos términos que son conceptos ver figura 4. Esta tarea se puede realizar de tres formas, la primera es definiendo desde los conceptos más generales hasta los más específicos dentro del dominio (top-down). La segunda consiste en el proceso inverso, o sea, ir desde lo más específico hasta lo más general (bottom-up) y la tercera vía es la combinación de los dos procesos anteriores, primero se detallan los conceptos más relevantes y luego se generalizan y especifican apropiadamente. En el presente trabajo se escogió top-down.

2.2.2.4 Construir el diccionario de conceptos.

Una vez que las taxonomías de conceptos han sido generadas, se especifica cuáles son las propiedades que describen cada concepto de la taxonomía y las instancias de cada uno de los conceptos. El diccionario de conceptos contiene todos los conceptos del dominio, sus relaciones, sus instancias, ver Tabla 3.

Tabla 3. Diccionario de Conceptos.

Clases	Instancias	Relaciones
Proyecto	sageb, cicpc, alashis, erp, identidad	coordinaproyecto aplicaherramienta
Especialista	Delvis, asnier, roig, delmys, liudmila	Coordinaproyecto coordinaherramienta escoordinador
Herramienta	jmeter, selenium, datagenerator	aplicaherramienta coordinaherramienta
Artefacto	appweb, appescritorio, documentación	presenta herramienta
Estrategia	estrategiadpsw	presenta actividad presenta documentacion presenta niveles presenta criterio
Tipo de Prueba	carga, estres, funcional, volumen	escoordinador esta en nivel utiliza tecnica presenta herramienta
Tecnica Prueba:	basada experiencia, cajanegra, cajablanca	utiliza tecnica
Nivel de prueba	sistema, integración, unidad	presenta niveles esta en nivel
Documentos	casoprueba, historialprueba, minuta	Presenta documento presenta documentacion
Actividades	solicitud, planificacion,	presenta actividad

	ejecución, gestionnc	presentadocumento
--	----------------------	-------------------

2.2.2.5 Definir Axiomas.

Methontology propone especificar la siguiente información para cada axioma formal: nombre, descripción en lenguaje natural, expresión que define de manera formal el axioma, los conceptos y las relaciones utilizadas en el axioma.

Tabla 4. Definición de Axiomas.

Nombre del axioma	Descripción	Expresión	Conceptos	Relaciones
axioma Tipo de Prueba	un Tipo de Prueba tiene más de una Técnica de Prueba y más de una Herramienta.	isUsedBy more (Tecnica_de_Prueba and Herramienta)	Tipo de Prueba Técnica de prueba Herramienta	utiliza

En la Tabla 4 se muestra un axioma formal de la ontología Procedimiento de Pruebas de Software, que establece que; un *Tipo de Prueba* utiliza solamente una *Técnica de Prueba* y una *Herramienta*, en la columna de la expresión se muestra la sintaxis utilizada en el Protégé. Las columnas que corresponden a conceptos y relaciones referidas contienen los conceptos y relaciones utilizadas en la expresión formal del axioma.

2.2.2.6 Describir las instancias.

En la tabla 5 se describen a continuación las Instancias con sus atributos.

Tabla 5. Descripción de Instancias.

Instancia	Atributos
sageb, cicpc, alashis, erp, identidad	nombrepryecto descripcionproyecto
Delvis, asnier, roig, delmys, liudmila	nombreespecialista descripcionespecialista

jmeter, selenium, datagenerator	nombreherramienta descripcionherramienta
appweb, appescritorio, documentación	nombreartefacto descripcionartefacto
estrategiadpsw	nombreestrategia descripcionestrategia
carga, estres, funcional, volumen	nombretipoprueba descripciontipoprueba
basadaexperiencia, cajanegra, cajablanca	nombretecnicaprueba descripciontecnicaprueba
sistema, integración, unidad	nombrenivel descripcionnivel
casoprueba, historialprueba, minuta	nombredocumentacion descripciondocumentacion
solicitud, planificacion, ejecución, gestionnc	nombreactividad descripcionactividad

2.2.3 Actividad: Formalización

La actividad de formalización se encarga de la transformación del dicho modelo conceptual obtenido en la Anterior actividad en un modelo formal o semi-computable. Para la transformación se utilizó el editor Protégé 4.1 ver figura 6, donde se muestran los conceptos, propiedades, restricciones, e instancias de la ontología.

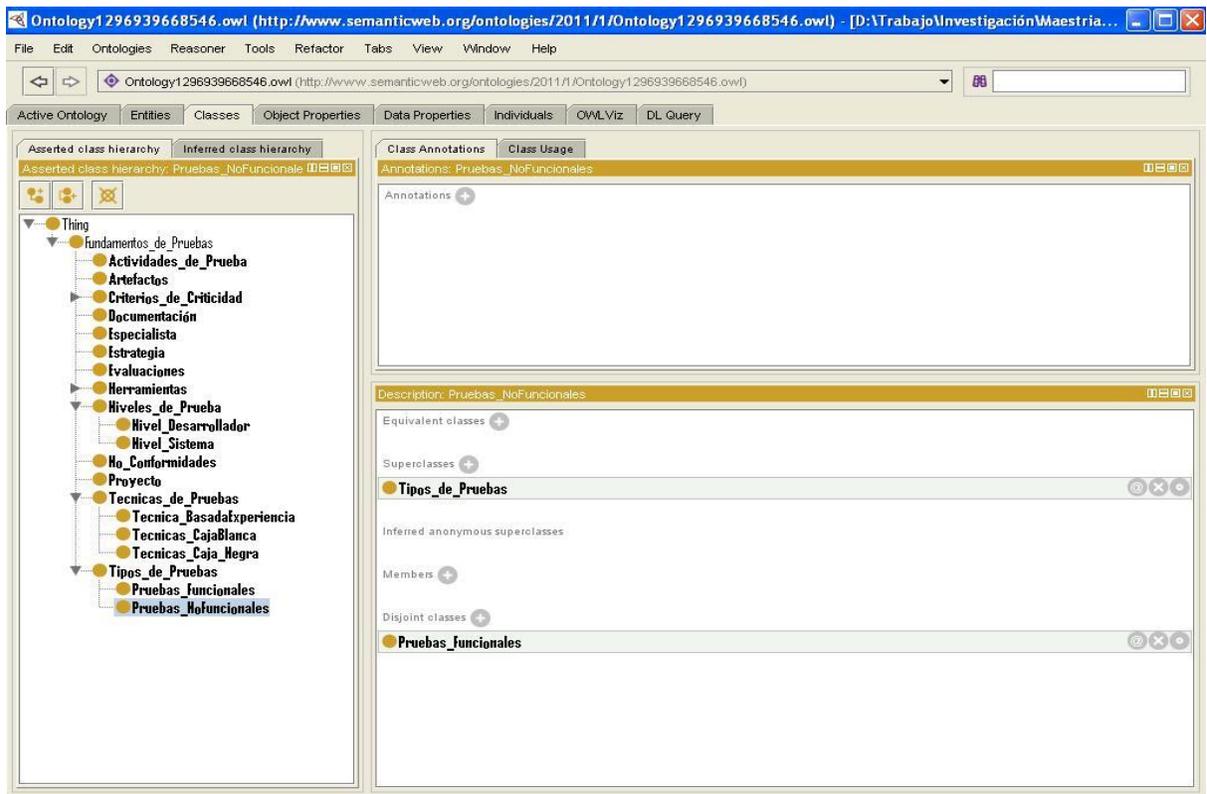


Figura 6. Interfaz Principal Protégé.

2.2.3.1 Creación de las Clases.

Las Clases son creadas por defecto a partir desde la Clase Padre “Thing”, que es una palabra clave que hace referencia a la W3C. Cada clase presentará: Anotaciones, Clases equivalentes, Sub-Clases, Instancias, restricciones.

En la figura 7 se muestra la jerarquía de clases creada en Protégé.

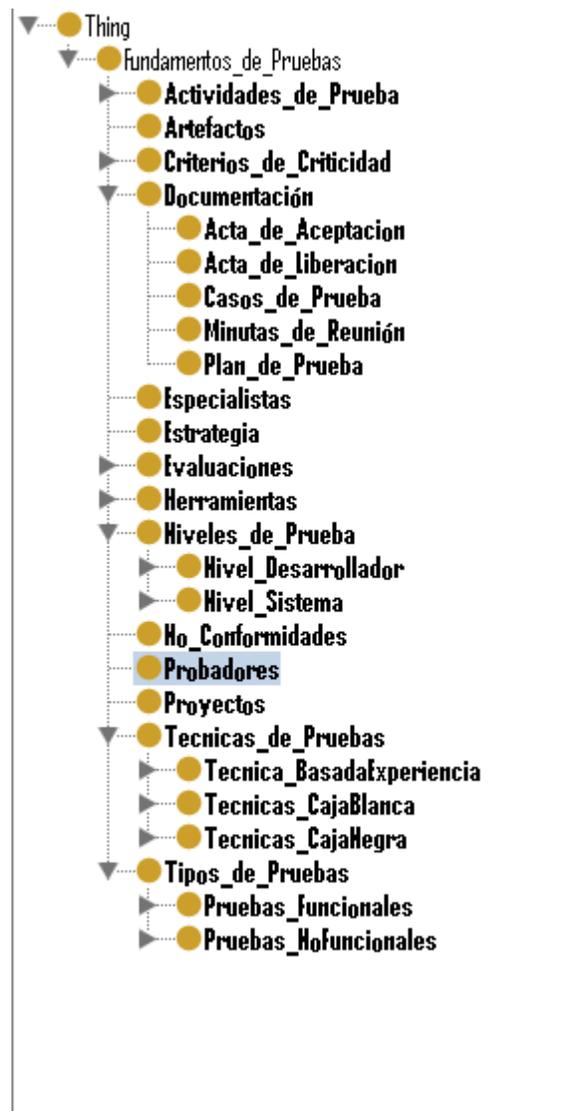


Figura 7. Definición de las clases creadas en Protégé.

2.2.3.2 Creación de las Propiedades

Las clases aisladas no proveerán suficiente información para responder las preguntas de competencia. Las propiedades son los elementos ontológicos que permiten establecer las relaciones, estas describen las cualidades internas de los conceptos. Representan las propiedades distintivas de los objetos. Pueden ser de 2 tipos, “**Object properties**” y “**Datatype properties**” En la figura 8 se muestran las relaciones “Object properties” que están presente en la Ontología propuesta.

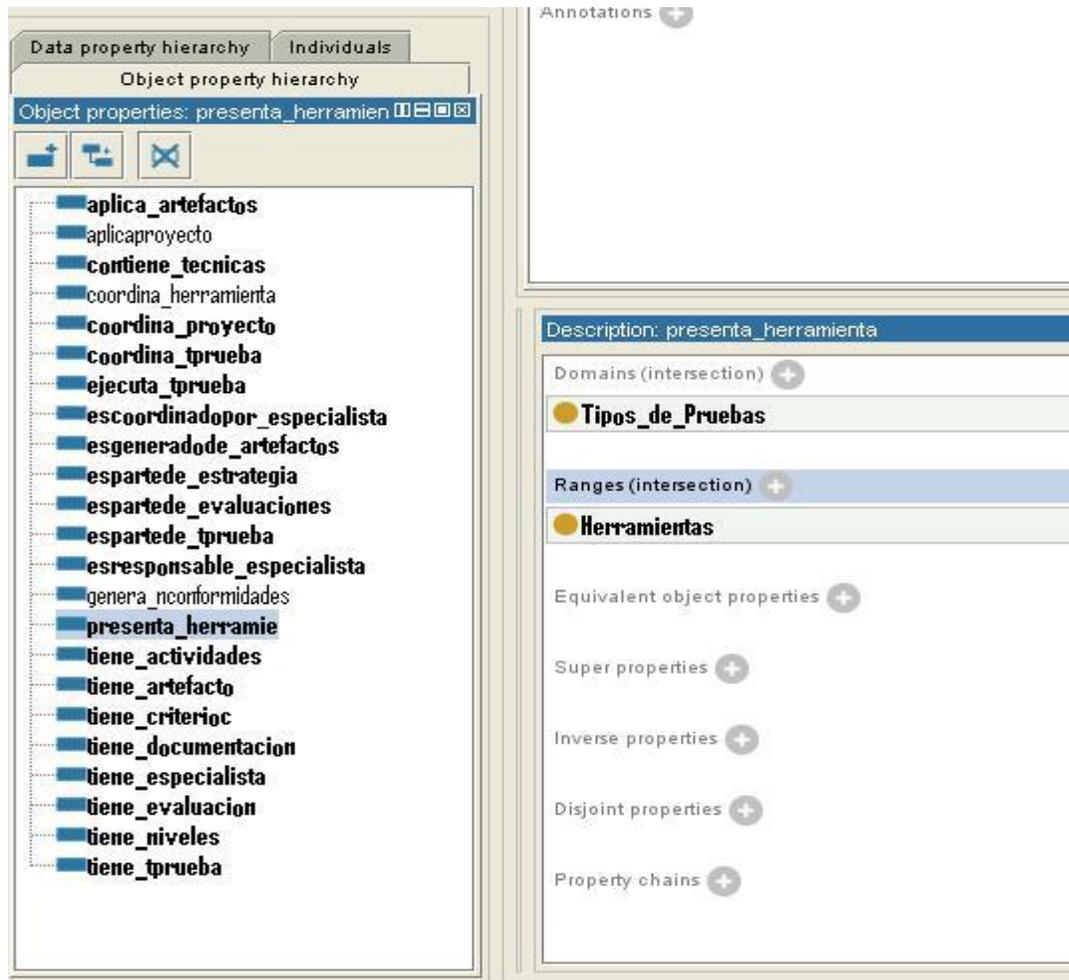


Figura 8. Descripción de las Propiedades en Protégé.

Cada relación descrita anteriormente constituye una propiedad que relaciona dos conceptos. Existe la propiedad “presenta_herramienta” que relaciona los conceptos “Tipo_de_Prueba” y “Herramientas”, de esta manera se representa en el modelo ontológico una herramienta de un determinado tipo de prueba, pero hay que especificar en la ontología que esta propiedad va a relacionar estos conceptos y no otros. La manera de especificar las propiedades que relacionan dos clases dadas es mediante la definición del rango y el dominio de la misma. El dominio es el conjunto de clases que tendrán la propiedad (Ej. para el caso anterior “Tipo_de_Prueba” pues es la clase que inicia la relación lógica: “Un Tipo de Prueba tiene una Herramienta”); mientras que el rango es la clase que se implica en la relación (Ej. para el caso anterior “Herramienta”).

Los Datatype Property relaciona un individuo con un valor de tipo dato. Por ejemplo el individuo “JMeter” perteneciente a la Clase Herramienta Automatizada tiene como datos, “Características”, “Descripción”. En la figura 9 se muestran las definidas en la Ontología.

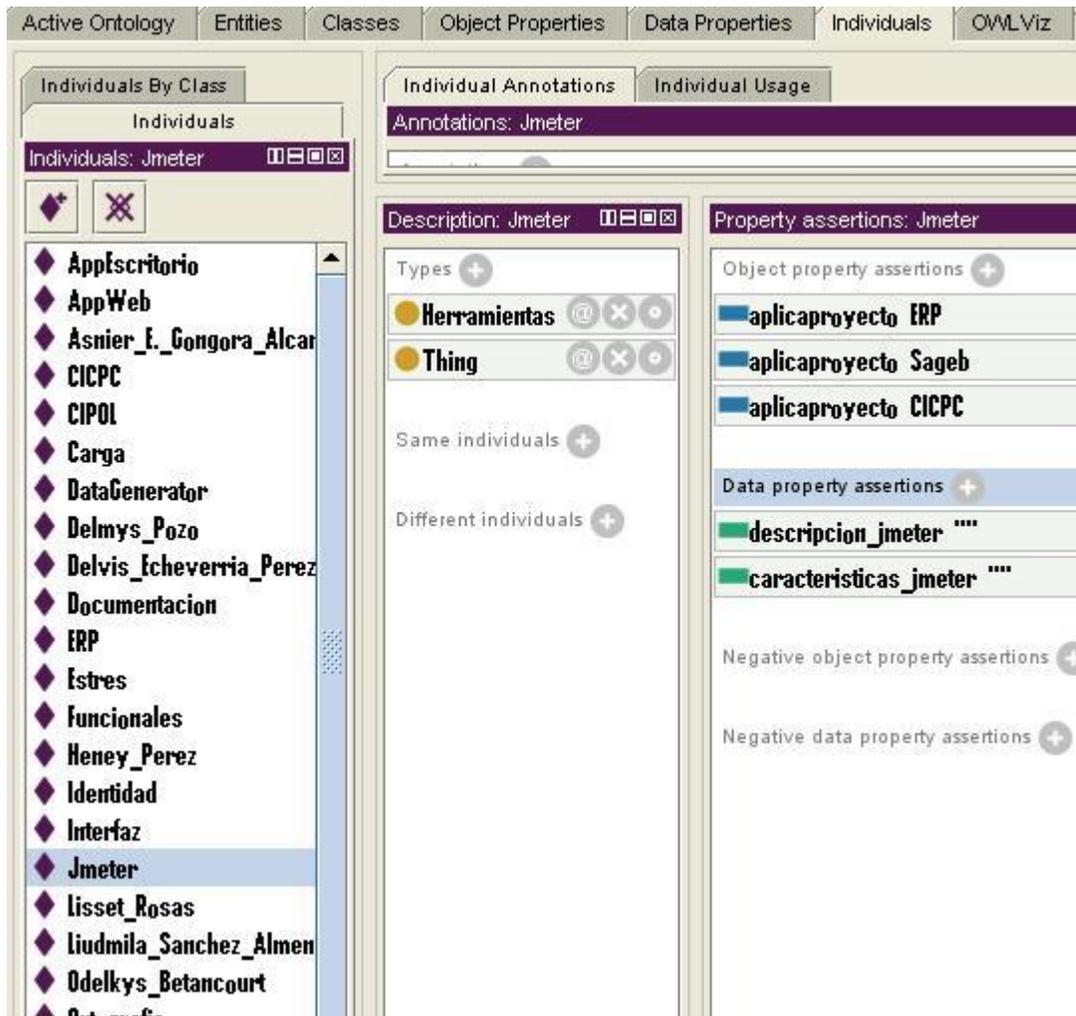


Figura 9. Descripción de los atributos de Instancias.

2.2.3.3 Creación de las Instancias

Las instancias son una parte importante del sistema de almacenamientos ya que mediante estas se realiza gran parte del proceso de razonamiento, además de que muestran en la práctica, la funcionalidad del sistema. En el caso específico de esta propuesta se crearon varias instancias (ver figura 10) de cada tipo de entidad para comprobar las preguntas antes definidas.

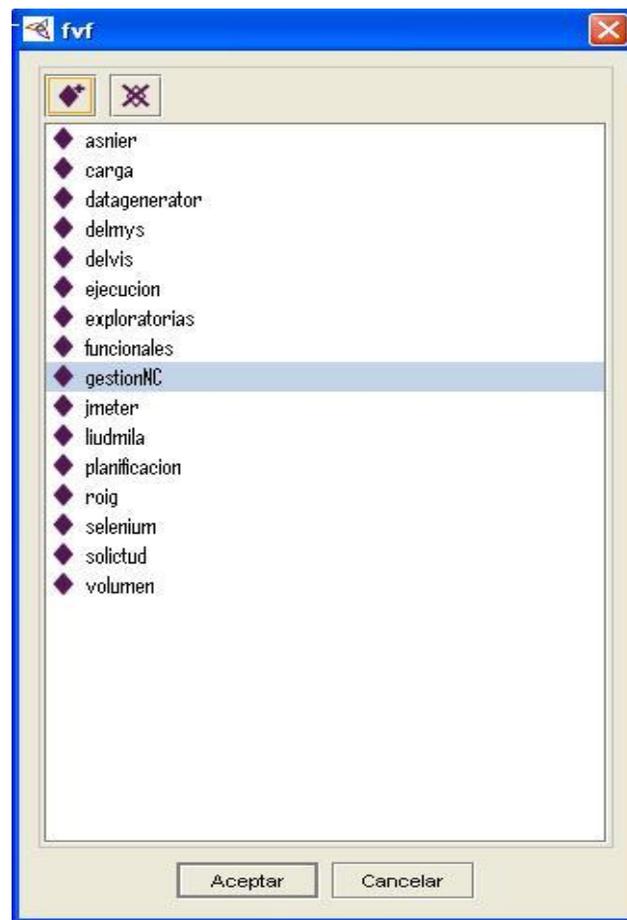


Figura 10. Creación de Instancias.

2.2.3.4 Obtención del grafo de la Ontología propuesta

OWLviz es un plugin para Protégé que permite visualizar con grafos los conceptos y las relaciones que tienen creadas, adicional a este plugin se debe tener instalado el GraphViz.

En la figura 11 se muestra la gráfica de cómo quedaría la ontología de apoyo al procedimiento de Pruebas de Software del DPSW en el editor Protégé.

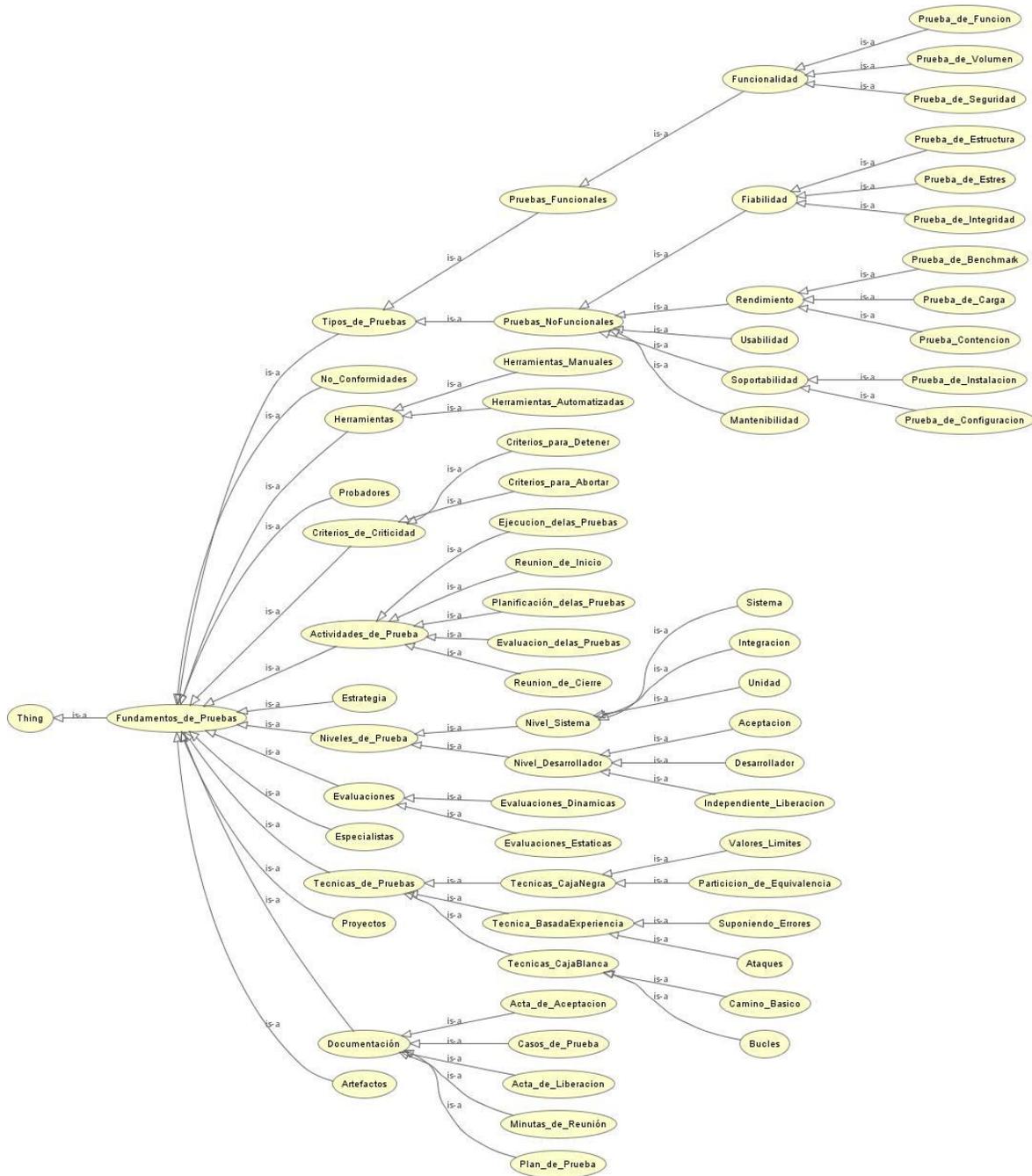


Figura 11. Grafo de la Ontología generado en Protégé.

2.2.3.5 Creación de Axiomas.

En la Figura 12 se muestra el editor Protégé modelando como quedan los axiomas entre diferentes clases, ejemplo de un axiomas sería, Tipo de Prueba utiliza solamente una Técnica de Prueba y una Herramienta, quedando en el Protégé *isUsedBy only (Tecnica_de_Prueba and Herramienta)*, un Tipo de Prueba utiliza alguna Técnica de Prueba y alguna Herramienta, y su expresión en el Protégé quedaría *(isUsedBy some Tecnica_de_Prueba) and (isUsedBy some Herramienta)*.

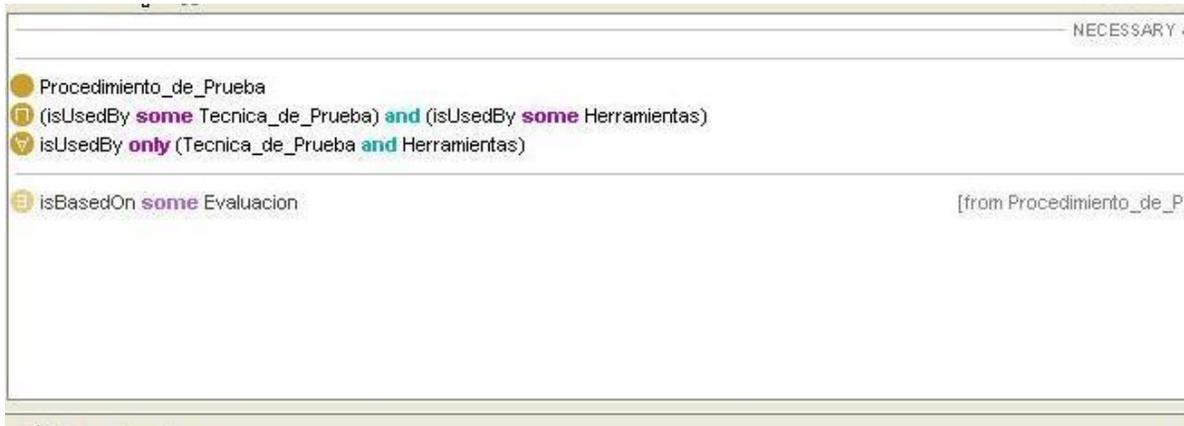


Figura 12. Descripción de Axiomas.

2.2.4 Actividad: Implementación.

El Lenguaje de Ontologías Web (OWL) proporciona un lenguaje para definir ontologías estructuradas basadas en Web. OWL pretende facilitar un lenguaje para ser usado con el fin de describir las clases, las propiedades de las clases y la relación entre las clases. OWL se construye sobre RDF, quien ofrece la base adecuada para desarrollar ontologías. Las ontologías basadas en RDF podrán ser distribuidas en muchos sistemas y serán compatibles con otros estándares web.

Aquí se muestran algunos fragmentos que describen la manera en que se codifican los datos de la propuesta ontológica.

La declaración de las clases se hace de la siguiente manera, donde se declaran las clases *Herramientas* y *Proyectos*:

```
<Declaration>
  <Class IRI="#Herramientas"/>
</Declaration>
<Declaration>
```

```
<Class IRI="#Proyectos"/>
</Declaration>
```

La declaración de una subclase de otra clase, se realiza de la siguiente manera, mostrando que las clases, *Basadas en Experiencia*, *Caja Negra* y *Caja Blanca* son subclases de la clase *Técnica de Prueba*:

```
<SubClassOf>
  <Class IRI="#Basadas_en_Experiencia"/>
  <Class IRI="#Tecnica_de_Prueba"/>
</SubClassOf>
<SubClassOf>
  <Class IRI="#Caja_Blanca"/>
  <Class IRI="#Tecnica_de_Prueba"/>
</SubClassOf>
<SubClassOf>
  <Class IRI="#Caja_Negra"/>
  <Class IRI="#Tecnica_de_Prueba"/>
</SubClassOf>
```

La declaración de las relaciones se realiza de la siguiente manera, donde se describe las relaciones que existen en la propuesta ontológica, *presentaherramienta* y *coordinaproyecto*:

```
<Declaration>
  <ObjectProperty IRI="#presentaherramienta "/>
</Declaration>
<Declaration>
  <ObjectProperty IRI="# coordinaproyecto "/>
</Declaration>
```

2.2.5 Actividad: Mantenimiento.

El mantenimiento de software se define según S. Mamone como: “*cualquier modificación de un producto de software, después de su entrega, para corregir errores, mejorar el rendimiento u otros atributos, o a la acción de adaptar el producto a un entorno que cambia*” []; y según R. Singh: “*los cambios en la gestión de productos de software para mantenerlos actualizados y en pleno funcionamiento*”.

Para el mantenimiento de la Ontología se tendrá en cuenta cualquier cambio que se produzca tanto en la estrategia de pruebas, como en el procedimiento del laboratorio. De

igual manera si cualquier aspecto novedoso se incorporara al departamento de pruebas de software este podría ser incluido en la ontología, se seguirán un conjunto de políticas que deberán ser implementadas en un periodo de un año para el caso de las actualizaciones, dentro de esta actividad se darán cursos de capacitación relacionado con las tecnologías utilizadas en el desarrollo de la propuesta.

2.3 Conclusiones Parciales.

En este capítulo se lograron todos los aspectos del diseño de la propuesta a través de la metodología utilizada se definió la estructura organizativa de la información generada en el procedimiento de pruebas así como el desarrollo de la propuesta ontológica con sus principales elementos: Clases, relaciones, instancias, se describieron cada uno de los conceptos con descripciones, a través de la herramienta Protégé se generó el código de la Ontología que será utilizado en el próximo capítulo.

Capítulo 3. Validación de la Ontología

En el presente capítulo se pondrá en práctica la propuesta planteada en el capítulo anterior. Se implementará una herramienta basada en la ontología que será capaz de responder a través de consultas cada una de las preguntas definidas al comienzo de la investigación. Se validará en la docencia del Laboratorio Industrial de Pruebas de Software con estudiantes y profesores pertenecientes al DPSW, y además se describirá la validación del modelo ontológico mediante el Servicio de Validación Web de la W3C (World Wide Web Consortium).

3.1 Transformación Modelo Ontológico al Modelo Persistente.

Una vez creada la ontología en formato OWL o RDF mediante Protégé, esta puede ser exportada a un modelo relacional con el plugin de Jena con el objetivo de persistir la información que contienen. Mediante el plugin Protege2Jena exportamos la ontología en formato OWL a un modelo persistente ver **figura 13**. Se configuró Protégé para poder usar el plugin y además se colocó el driver de MySQL “mysql-connector-java-5.1.6.jar” en los archivos de Protégé.

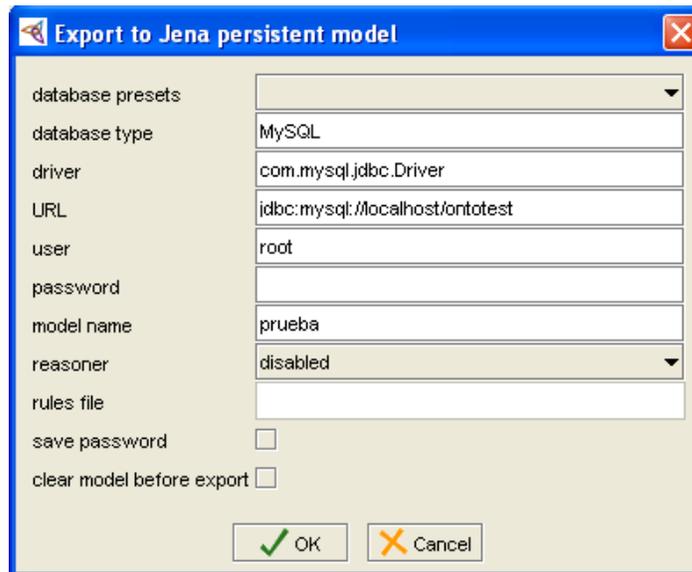


Figura 13. Exportación del Modelo Ontológico al Modelo Persistente.

Una ontología puede ser utilizada desde el propio archivo OWL o RDF debido a que en estos se encuentran definidos las relaciones, atributos, propiedades, instancias; sin embargo el propósito de un almacenamiento persistente es recuperar los datos de una ontología sin necesidad de cargarlos cada vez que se necesita un acceso a la base de datos del conocimiento. En el proceso de transformación a un modelo persistente se necesita de un esquema de base de datos; en el cual Protégé creará las tablas necesarias en MySQL automáticamente. En la figura 14 se muestra las tablas de la Base Datos.

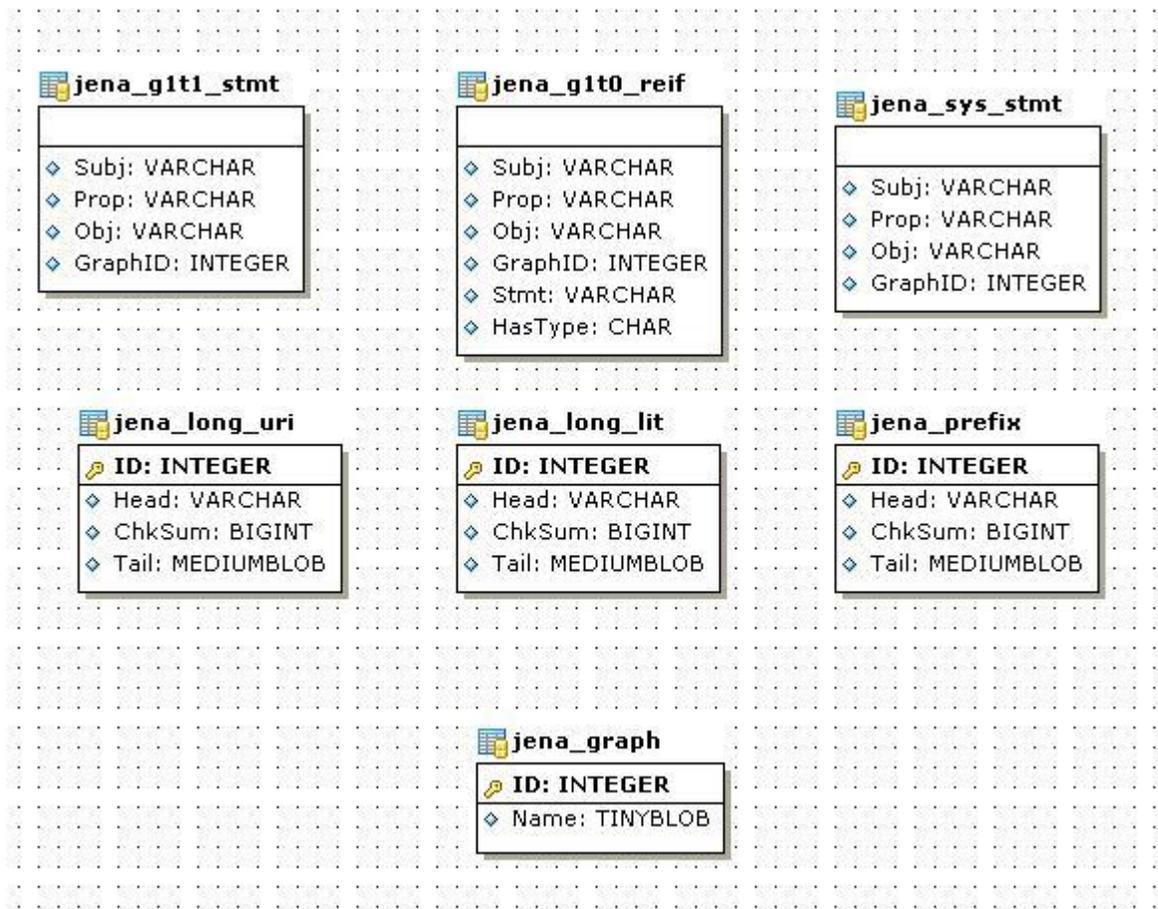


Figura 14. Tabla de la BD generada por Jena y Protégé.

3.2 Cargar una ontología con Jena desde almacenamiento persistente.

Para el trabajo con el modelo persistente, se utilizó el IDE Netbeans 6.9, por ser este un IDE multiplataforma, multilinguaje y de código abierto u open source. Seguidamente se creó la conexión a la base de datos, especificando el nombre de la base de datos, el driver de conexión que en este caso es el JDBC. A través del Netbeans se añadió el drive JDBC:Mysql para la conexión con la Base de Datos y la librería de Jena, para acceder a la información representada en el modelo persistente. Ver Figura 15.

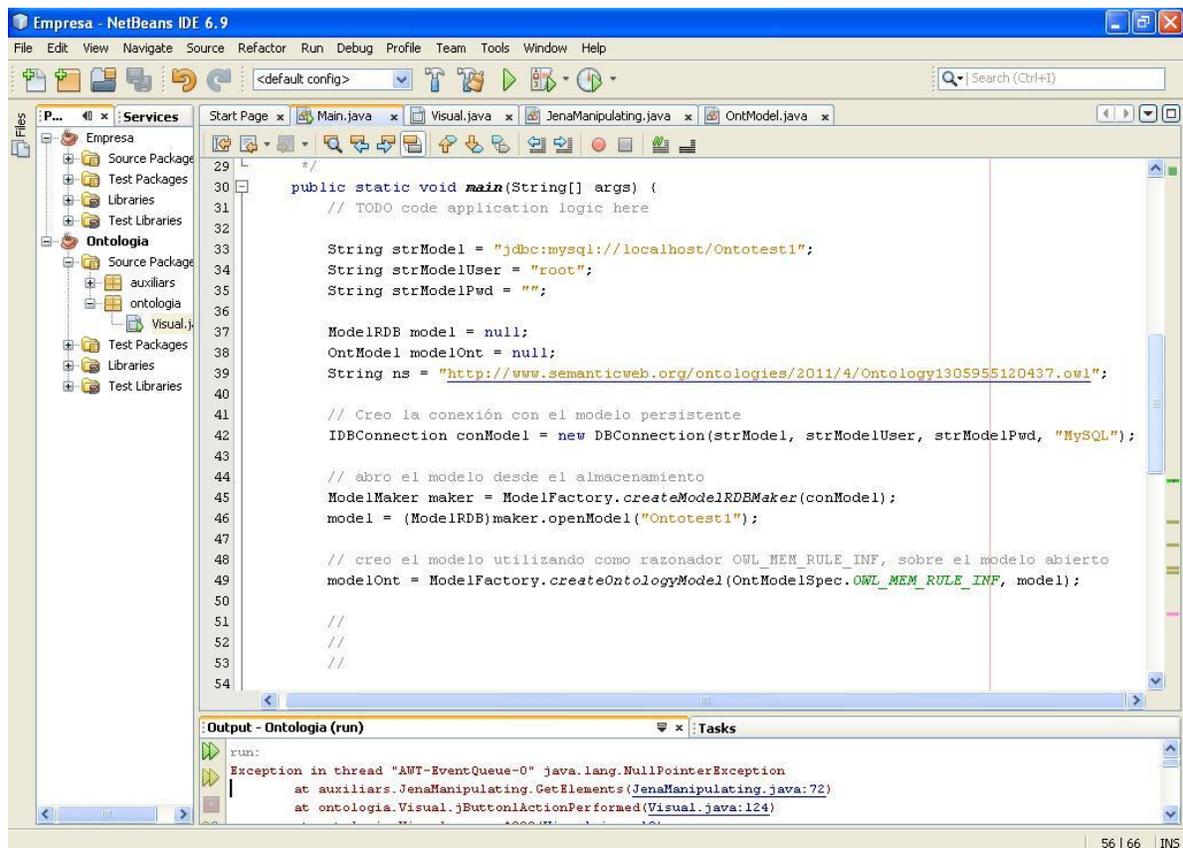


Figura 15. Uso del NetBeans para la visualización del modelo persistente.

Para utilizar el modelo persistente desde el API Jena, se utilizaron diferentes métodos a continuación se muestran algunos:

1. *IDBConnection conModel*: Conexión a la Base Datos.
2. *Individual Especialista = modelOnt.getIndividual()*: Obtener la instancia de un especialista.
3. *DatatypeProperty descripcion = modelOnt.getDatatypeProperty()*: Obtener la propiedad de las instancias.

3.3 Herramienta Informática basada en la Ontología.

Para darle respuesta a cada una de las preguntas definidas en la actividad de Especificación propuesta por la metodología escogida, se realizaron consultas al modelo persistente a través de Jena. Se implementó una herramienta que muestra de manera visual las respuestas a cada una de las preguntas. Para el uso de esta herramienta, primeramente se introduce la instancia y la relación que están involucradas en la consulta que se irá a realizar. Seguidamente se selecciona la opción Realizar consulta y mostrando en un "textField" la respuesta, como se ejemplifica en la figura 16.

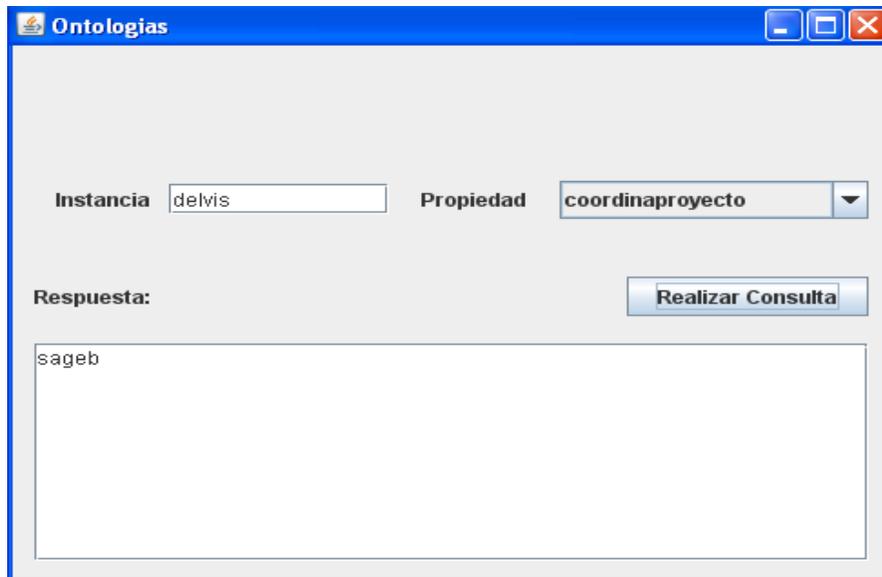


Figura 16. Herramienta para realizar consultas a la ontología desarrollada.

Para el caso que se muestra en la imagen la pregunta sería qué proyecto ha coordinado un determinado especialista.

A continuación la especificación del resto de las preguntas con sus instancias y sus relaciones.

1. ¿Que proyecto ha coordinado un determinado especialista?
Instancia: Especialista **Relación:** Especialista *coordinaproyecto* Proyecto.
2. ¿Que herramientas se le ha aplicado a un determinado proyecto?
Instancia: Proyecto **Relación:** Proyecto *aplicaherramienta* Herramienta
3. ¿Que tipo de No conformidades presenta un determinado artefacto?
Instancia: Artefacto **Relación:** Artefacto *presentaNc* Noconformidades
4. ¿Qué tipo de evaluación presenta un determinado artefacto?
Instancia: Artefacto **Relación:** Artefacto *presentaevaluacion* Evaluacion
5. ¿Que herramienta es coordinada por un determinado especialista?
Instancia: Especialista **Relación:** Especialista *coordinaherramienta* Herramienta
6. ¿Que criterios de criticidad presenta la estrategia del DPSW?
Instancia: Estrategia **Relación:** Estrategia *presenta criterio* Criterios
7. ¿Qué herramienta presenta un determinado tipo de prueba?
Instancia: Tipo Prueba **Relación:** Tipo Prueba *presenta herramienta* Herramienta
8. ¿Qué técnicas de pruebas utiliza un determinado tipo de prueba?
Instancia: Tipo Prueba **Relación:** Tipo Prueba *utilizatecnica* Tecnica Prueba
9. ¿Qué niveles de prueba presenta la estrategia de pruebas del DPSW?
Instancia: Estrategia **Relación:** Estrategia *presenta niveles* Nivel de prueba
10. ¿Qué documentación presenta la estrategia de prueba del DPSW?
Instancia: Documentos **Relación:** Estrategia *presenta documentacion* Documentos
11. ¿Qué actividades presenta la estrategia de pruebas del DPSW?
Instancia: Actividades **Relación:** Estrategia *presenta actividad* Actividades
12. ¿Qué nivel de prueba presenta un determinado tipo de prueba?

Instancia: Tipo prueba **Relación:** Tipo prueba *estaen nivel* Nivel de prueba

13. ¿Que especialista coordina un determinado tipo de prueba?

Instancia: Tipo prueba **Relación:** Tipo de prueba *escoordinadopor* Especialista

14. ¿Qué documentos están presentes en las actividades?

Instancia: Actividad **Relación:** Actividades *presentadocumento* Documentos

3.4 Validación mediante el Servicio de Validación Web de la W3C.

El World Wide Web Consortium (W3C) es una comunidad internacional que desarrolla estándares que aseguran el crecimiento de la Web a largo plazo.

Además de la clásica “Web de los documentos” El W3C está ayudando a construir un conjunto de tecnologías para apoyar una “Red de datos,” el tipo de datos que se encuentran en bases de datos. El objetivo último de la Web de los datos es permitir a las computadoras hacer el trabajo más útil y desarrollar sistemas que puedan apoyar las interacciones de confianza en la red. El término “Web Semántica” se refiere a la visión del W3C de la Web de los datos vinculados. Las tecnologías de la Web Semántica permiten a las personas crear recopilaciones de datos en la Web, crear vocabularios, y escribir las reglas para el manejo de datos.

La W3C cuenta con un servicio de validación de archivos RDF, servicio que permite al usuario conocer cuando la ontología o el archivo poseen inconsistencias, verifica la taxonomía de dicha ontología, etc.

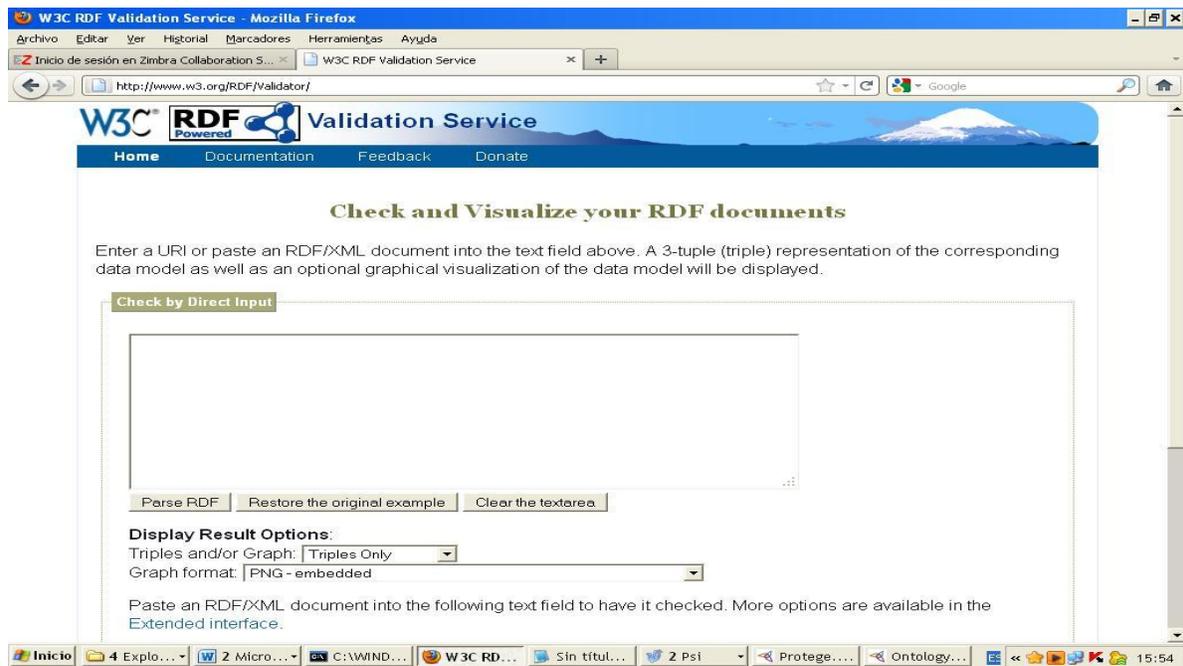


Figura 17. Servicio de Validación de Ontologías.

En la página mostrada en la **figura 17**, el usuario puede verificar insertando íntegramente el código RDF de su ontología, lo anteriormente mencionado. El sistema muestra, luego de parsear el código RDF, los datos de la ontología, las relaciones, etc. Mostrando además el resultado de la revisión, si se realizó y validó correctamente o si por algún error no se pudo validar el archivo.

Luego de generar el código RDF de la propuesta ontológica en cuestión, y siguiendo los pasos descritos a lo largo del epígrafe, el validador arrojó el resultado que se muestra en la **Figura. 18**, mostrando que la ontología y el archivo RDF como tal se validó correctamente y sin incongruencias.



Figura 18. Resultado de la ontología por el Validador W3C

3.5 Resultados obtenidos con la propuesta ontológica en el DPSW.

Luego de haber validado la Ontología técnicamente a través del W3, se validó la propuesta Ontológica en función de la comunicación, organización de la información y la calidad del procedimiento de pruebas. En la siguiente tabla se muestran los estados anteriores de estos 3 indicadores.

Tabla 6. Indicadores antes de la Propuesta.

	Antes
Comunicación.	<ul style="list-style-type: none"> Las búsquedas bibliográficas son realizadas solamente a través de internet. Existencia de la "Polisemia".
Organización de la Información.	<ul style="list-style-type: none"> Grandes volúmenes de información, sin clasificar, y poco estructurada.
Calidad del	<ul style="list-style-type: none"> Uno de los objetivos del procedimiento de pruebas es detectar la

procedimiento de pruebas.	<p>mayor cantidad de NC posibles, de manera que se entregue al cliente un producto con la menor cantidad de errores. En la actualidad la generación de No Conformidades, proveniente de las pruebas realizadas por los estudiantes, son insuficientes, en ocasiones mal redactadas y sin valor técnico.</p> <ul style="list-style-type: none"> • Los probadores no presentan una base de conocimientos sólida de la disciplina de Ingeniería de Software y no cuentan con acceso a bibliografía especializada.
---------------------------	---

Para saber el grado de aceptación de la propuesta ontológica se desarrolló una encuesta a una muestra significativa de varios de los involucrados en el procedimiento de pruebas, para este caso estudiantes de segundo año de la asignatura de Práctica Profesional II y los especialistas del DPSW.

3.5.1 Selección y Cálculo de la Muestra

La muestra es, en esencia, un subgrupo de la población. Digamos que es un subconjunto de elementos que pertenecen a ese conjunto definido en sus características al que llamamos población. Para la selección de una muestra necesitamos principalmente, determinar el tamaño de la muestra (n) y seleccionar los elementos muestrales, de manera que todos tengan la misma posibilidad de ser elegidos. [36]

$$n' = \frac{S^2}{V^2} \frac{\text{varianza de la muestra}}{\text{varianza de la población}} \quad (\text{Ecuación 1})$$

Ajustándose si se conoce el tamaño de la población N entonces tendremos que:

$$n' = \frac{n'}{1 + n' / N} \quad (\text{Ecuación 2})$$

S² Varianza de la Muestra, es la probabilidad de ocurrencia esperada de la variable que se pretende medir. Para este caso en particular la variable sería “*aceptación de la propuesta*”, se ha determinado un 90%.

V² Varianza de la Población, es el margen de error estándar con el que queremos trabajar para proyectar nuestros resultados. Generalmente, se utilizan valores que oscilan entre 0.010 y 0.015.

Sustituyendo en las ecuaciones antes descritas la muestra sería aproximadamente de 121 individuos.

3.5.2 Aplicación de la encuesta

Primer Aspecto de la Encuesta.

¿La ontología responde a las preguntas realizadas al principio de la Investigación?

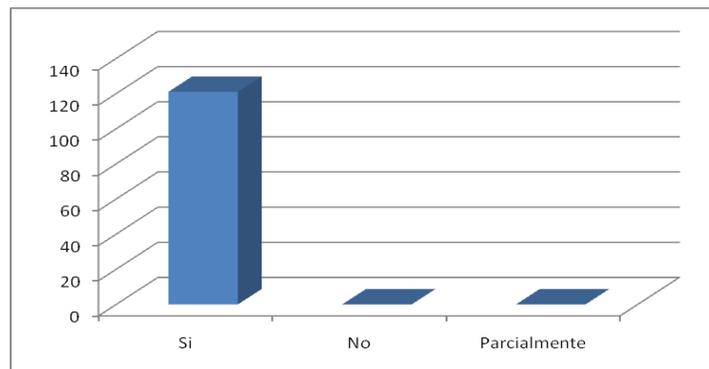


Figura 19. Encuesta 1er Aspecto.

Observaciones: El 100% de los encuestados respondió "Si".

Segundo Aspecto de la Encuesta.

¿El dominio de la Ontología está acorde al dominio del Departamento de Pruebas de Software (Pruebas de Software)?

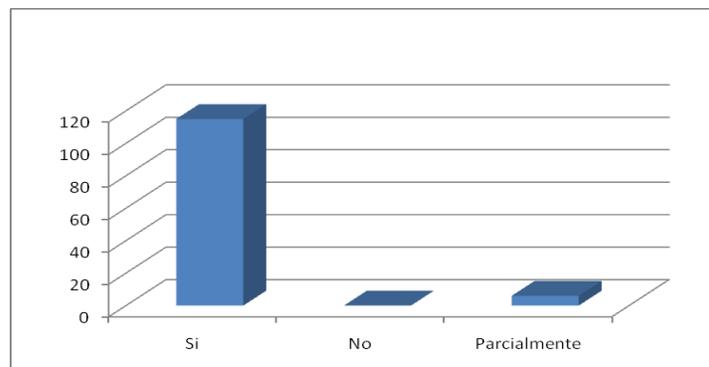


Figura 20. Encuesta 2do Aspecto.

Observaciones: El 95.047 respondió Sí, el resto respondió parcialmente, en los argumentos especificaban aspectos como:

- No se definen ejemplos de No Conformidades comunes.
- Existen elemento que no están en uso actualmente en las clases de PP por ejemplo la Gestión de las Solicitudes.
- Se recomendó que se añadiera clasificar los elementos que eran para estudiantes y los que eran para profesores.

Tercer Aspecto de la Encuesta.

¿La propuesta abarca la mayoría de los temas impartidos en la docencia de la Asignatura Práctica Profesional II?

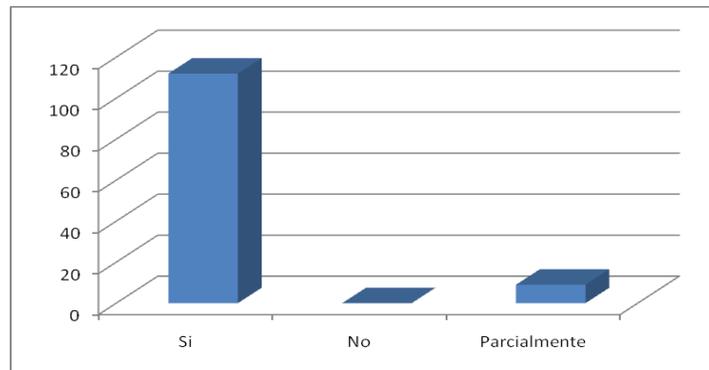


Figura 21. Encuesta 3er Aspecto.

Observaciones: El 92.56% respondió “Sí” el resto Parcialmente. En los argumentos especificaban aspectos como:

- La ontología recoge muchos más aspectos que los impartidos en clases.

Cuarto Aspecto de la Encuesta.

¿La ontología proporciona una guía de soporte que puede ser utilizada en las Clases de Práctica Profesional, en el Estudio Independiente, y en el trabajo de probadores detectando mayor cantidad de No Conformidades en el procedimiento de pruebas?

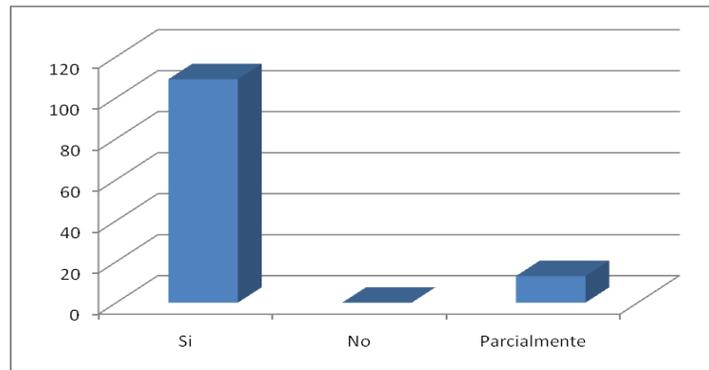


Figura 22. Encuesta 4to Aspecto.

Observaciones: El 90.08 % de los encuestados respondió “Sí”.

Quinto Aspecto de la Encuesta.

¿La Ontología favorecerá la comunicación entre los especialistas del departamento de pruebas de software y ayudará a la toma de decisiones en los procesos de las pruebas de liberación? (Solo para especialistas del DPSW).

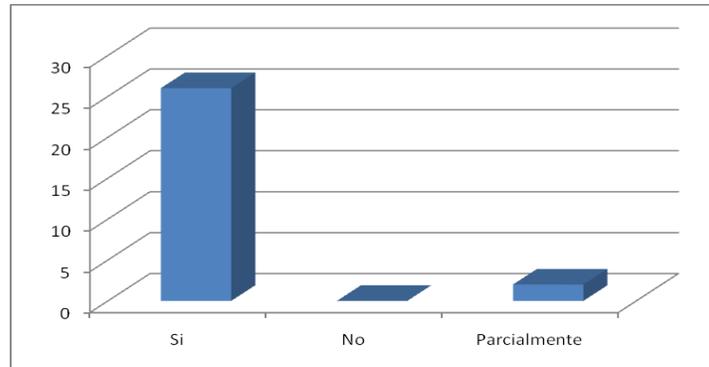


Figura 23. Encuesta 5to Aspecto.

Observaciones: El 92.85% respondió “Sí” el resto Parcialmente. En los argumentos especificaban aspectos como:

- Se le pueden añadir más aspectos para enriquecerla.
Solicitudes de pruebas de liberación, términos de pruebas de aceptación.
- Pudiera pensarse la opción de montarle en una aplicación web para un mejor uso de la misma.

Sexto Aspecto de la Encuesta.

¿La propuesta ontológica contribuye a una mejor organización de la información presente en el procedimiento de pruebas? (Solo para especialistas del DPSW).

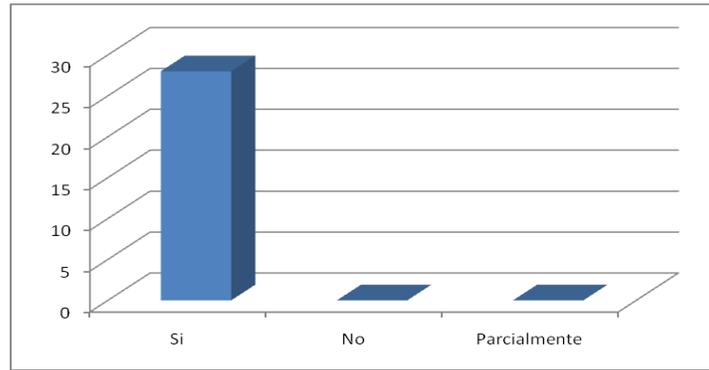


Figura 24. Encuesta 6to Aspecto.

Observaciones: El 100% de los encuestados respondió “Si”.

Después de haber desplegado la herramienta y realizado la encuesta se pudo constatar una diferencia en los indicadores antes expuestos.

Tabla 7. Indicadores después de la Propuesta.

	Actualidad
Comunicación	<ul style="list-style-type: none"> • Contribuye a mejorar el proceso de enseñanza aprendizaje a través del uso de una fuente de conocimiento común, utilizada por profesores y estudiantes en Preparaciones Metodológicas, Conferencias, Clases Prácticas, Tareas Extra-clases, etc. • A través de la ontología se pueden descubrir distorsiones que estén presente en los procesos de aprendizaje en un mismo contexto.
Organización de la Información	<ul style="list-style-type: none"> • La información generada a partir del procedimiento de pruebas, ha quedado estructurada y organizada con el uso de la ontología. • La búsqueda de un determinado aspecto se puede realizar a través de la taxonomía de conceptos.
Calidad del	<ul style="list-style-type: none"> • En función del estado actual de las variables anteriores las No

procedimiento de pruebas.	<p>Conformidades generadas, aumentaron en número. En los últimos encuentros la diferencia entre los NC generadas y NC correctas fue aceptable en comparación con los primeros encuentros. Ver Tabla 8</p> <ul style="list-style-type: none">• Los probadores podrán consolidar sus conocimientos en la disciplina de pruebas de software a través de la información presente en la propuesta ontológica, empleando estos conocimientos en función de la realización de tareas, proyectos finales y en la ejecución de las pruebas, a través del uso de la ontología se accederá a una bibliografía especializada. Ver en la Tabla 8, el promedio de Notas.
---------------------------	--

Tabla 8. Relación de los indicadores.

Encuentros	NC Generadas	NC Correctamente	Promedio Notas Preguntas Escritas	Promedio Notas Trabajos Independientes, Seminarios.
Antes				
1	30	15	2.083 Ptos	-
2	35	8	2.333 Ptos	3.800 Ptos
3	24	14	3.333 Ptos	3.565 Ptos
4	35	22	3.125 Ptos	4.012 Ptos
5	28	18	3.541 Ptos	4.120 Ptos
Después				
6	15	13	4.160 Ptos	4.980 Ptos
7	20	20	4.080 Ptos	5.000 Ptos
8	25	24	4.430Ptos	4.659Ptos

3.6 Conclusiones Parciales

En este capítulo se demostró que la Ontología responde a las preguntas de competencia antes definida en la investigación a través del uso de una herramienta. Se aplicó una encuesta con una muestra de probadores y especialistas involucrados en las pruebas, arrojando resultados aceptables, técnicamente se realizó una validación de la consistencia de la Ontología mediante el validador web de la W3C con resultados satisfactorios.

Conclusiones

- A través de una evaluación del estado del arte se seleccionó las Ontologías como técnicas para la gestión del conocimiento, la metodología Methontology para su construcción, OWL como lenguaje, Protegé como herramienta de modelado, y Jena como framework para la implementación de la herramienta.
- Se definió una estructura organizativa de la información que permitirá un mayor entendimiento del dominio seleccionado, basándose en la información que se genera en el procedimiento de pruebas del DPSW.
- La ontología resultante engloba el conocimiento que está presente en el dominio de de las pruebas de software, con todos los componentes descritos en la investigación, mejorando la organización de la información, la comunicación entre los especialista del el DPSW y la calidad del procedimiento de pruebas.
- Se logró la validación de la propuesta a través de una herramienta basada en consultas sobre el modelo persistente obtenido. Se aplicó una encuesta con una muestra de probadores y especialistas involucrados en las pruebas arrojando resultados aceptables, técnicamente se realizó una validación de la consistencia de la Ontología mediante el validador web de la W3C con resultados satisfactorios.

Recomendaciones.

- Extender la herramienta en el Laboratorio Industrial de Pruebas de Software para que pueda ser utilizada por estudiantes y especialistas.
- Hacer extensible el modelo ontológico y la Herramienta a las demás áreas de la Universidad relacionadas con las pruebas de software.
- Introducir la mayor cantidad de Instancias y relaciones en la Ontología para que se haga un uso más efectivo y real de la misma como parte del mantenimiento.

Referencias Bibliográficas

1. ARANO, Silvia. *Los tesauros y las ontologías en la Biblioteconomía y la Documentación*, [ref. diciembre de 2010]. Disponible en Web:
http://eprints.rclis.org/bitstream/10760/8972/1/Los_tesauros_y_las_ontolog%C3%ADas_en_la_Biblioteconom%C3%ADa_y_la_Documentaci%C3%B3n.pdf
2. RODRÍGUEZ, Keilyn. RONDA, Rodrigo. *Web semántica: un nuevo enfoque para la organización y recuperación de información en el web*. Ciudad de La Habana: s.n., 2005.
3. MARTIN, Silvia. OBEJERO Beatriz. *¿Qué son los tesauros y las ontologías?* [ref. diciembre de 2010]. Disponible en Web:
http://www.avizora.com/publicaciones/pedagogia/textos/0047_que_es_tesauro_ontologia.htm.
4. CRUZ, Raimil. CRUZ, Liz Mary. REYES, Liannet Lucia. *Utilización de la representación formal del conocimiento de la toma de decisiones*. [ref. diciembre de 2010]. Disponible en Web:
<http://ocs.crv.matcom.uh.cu/index.php/jces/jce2010/paper/downloadSuppFile/204/140>
5. PALMA, Álvaro. RODRÍGUEZ, Emilio. *Gestión del conocimiento y gestión de calidad en la educación superior*. 2008, Vol. II, ISBN 0718-3429
6. CANALS, Agustín. *Hacia la Gestión el conocimiento*. [ref. enero de 2011]. Disponible en Web: http://www.uoc.edu/web/esp/art/uoc/canals/canals_imp.html
7. SOTO, María Aurora. BARRIOS Norma M. *Gestión del conocimiento. Parte I. Revisión crítica del estado del arte* [ref. noviembre de 2010]. Disponible en Web: http://bvs.sld.cu/revistas/aci/vol14_2_06/aci04206.htm
8. MARTÍNEZ, Francisco Javier *El salto desde la Gestión de Información a la Gestión del Conocimiento*. [ref. noviembre de 2010]. Disponible en Web: <http://um.es/gtiweb/fjmm/elsalto.doc>
9. CANALS, Agustín. *La gestión del conocimiento*. [ref. enero de 2011]. Disponible en Web: <http://www.uoc.edu/dt/20251>

10. ESTRADA, Vivian. BENÍTEZ, Francisco. *La gestión del conocimiento en la nueva universidad Cubana*. [ref. enero de 2011]. Disponible en Web:
<http://www.ucf.edu.cu/ojsucf/index.php/uys/article/viewFile/39/82>
11. LENDINEZ, Sebastián. *Aplicación para la Población de ontologías*, [ref. enero de 2011]. Disponible en Web: http://e-archivo.uc3m.es/bitstream/10016/8522/1/PFC_Sebastian_Lendinez_Teixeira.pdf
12. FERNÁNDEZ, Jesualdo. *Un entorno de integración de ontologías para el desarrollo de sistemas de gestión de conocimiento*. [ref. enero de 2011]. Disponible en Web: www.tdr.cesca.es/TDR-1123105-134449/
13. Castells, Pablo. *Aplicación de técnicas de la web semántica*. ref. enero de 2011]. Disponible en Web: <http://arantxa.ii.uam.es/~castells/publications/coline02.pdf>
14. VALENCIA, Rafael. *La siguiente versión de la web, la web semántica*. [ref. diciembre de 2010]. Disponible en Web:
http://www.ciimurcia.es/informas/ene05/articulos/La_siguiete_version_de_la_web_la_web_semantica.pdf
15. MARTIN, Hepp. DE LEENHEER, Pieter. "Ontologies: state of the art, business potential, and grand challenges." En: *Ontology Management. Semantic Web, Semantic Web Services, and Business Applications*. Springer 2008, e-ISBN: 978-0-387-4. P 3-23.
16. GRELA, Laura. SAURI, Elena. *Ontologías en documentación*. [Ref. Noviembre de 2010]. Disponible en Web: <http://personales.upv.es/ccarrasc/doc/2001-2002/ontologias/INICIO.htm>
17. FERNÁNDEZ, Jesualdo. *Un Entorno de Integración de Ontologías para el Desarrollo de Sistemas de Gestión de Conocimiento*. www.tdr.cesca.es/TDR-1123105-134449/
18. LOZANO, Adolfo. *Ontologías en la Web Semántica*. [Ref. Noviembre de 2010]. Disponible en Web: http://www.anobium.es/docs/gc_fichas/doc/68ERfhjkmv.pdf.
19. LAMARCA, María Jesús. *Hipertexto: El nuevo concepto de documento en la cultura de la imagen*. [Ref. Diciembre de 2010]. Disponible en Web:
http://www.hipertexto.info/documentos/meta_html.htm.
20. LAMAS, Maria Isabel. *Lenguajes de consulta para documentos RDF*. [Ref. Diciembre de 2010]. Disponible en Web:
<http://openaccess.uoc.edu/webapps/o2/bitstream/10609/640/1/38449tfc.pdf>.

21. ABIÁN, Miguel Ángel. *El Futuro de la Web XML, RDF/RDFS, Ontologías y la web Semántica*. [Ref. Diciembre de 2010]. Disponible en Web:
<http://www.javahispano.org/licencias/>
22. WATERFELD, Walter. WEITEN, Moritz. *Ontology management infrastructures* En: *Ontology Management. Semantic Web, Semantic Web Services, and Business Applications*. Springer 2008, e-ISBN: 978-0-387-4. P.
23. SAMPER, José Javier. *Ontologías Para Servicios Web Semánticos De Información De Tráfico: Descripción Y Herramientas De Explotación* [Ref. Diciembre de 2010]. Disponible en Web: Http://Www.Tdr.Cesca.Es/Tesis_Uv/Available/Tdx-0628106085805//Samper.Pdf
24. QUERRO, Adriana. . [Ref. Diciembre de 2010]. Disponible en Web:
http://tesis.ula.ve/postgrado/tde_busca/archivo.php?codArchivo=275
25. CORCHO, Oscar. FERNANDEZ-LOPEZ, Mariano. GOMEZ, Asuncion. *Methodologies, tools and languages for building ontologies. Where is their meeting point?* [Ref. Diciembre de 2010]. Disponible en Web:
http://www.ct.aegean.gr/people/vkavakli/MIS/papers/Corcho_2003.pdf.
26. RODRÍGUEZ, Claudia Milena. CANO, William. MARTÍNEZ, José Mauricio. *Razonadores semánticos: un estado del arte*. [Ref. Diciembre de 2010]. Disponible en Web:
http://openebop.labs.andago.com/c/document_library/get_file?p_l_id=10681&folderId=12635&name=DLFE-316.pdf
27. GÓMEZ, Begoña. *Definición de Servicios y Subsistemas de Seguridad*. [Ref. Diciembre de 2010]. Disponible en Web:
http://www.usbbog.edu.co/Nuestra_UniversidadPublicacionesIngeniumSemantics.pdf
28. RAMÍREZ Kryscia. *Ontologías*. [Ref. Diciembre de 2010]. Disponible en Web:
<http://zafiro.ecci.ucr.ac.cr/pf3341/presentaciones/Ontologias/Ontologias.ppt>
29. Jena Semantic Web Framework. [Ref. Diciembre de 2010]. Disponible en Web:
<http://jena.sourceforge.net/>
30. LEIGH James. *An Open Source Framework for RDF Data*. [Ref. enero de 2011]. Disponible en Web: <http://www.devx.com/semantic/Article/40987/1954>
31. Kaon Tool. [Ref. enero de 2011]. Disponible en Web: <http://kaon.semanticweb.org/>
32. Netbeans. [Ref. febrero de 2011]. Disponible en Web: <http://www.netbeans.org/>

33. Eclipse. [Ref. febrero de 2011]. Disponible en Web: <http://www.eclipse.org/>
34. DUARTE, Katia Cristina. ALMEIDA, Ricardo *Uma Ontologia do Qualidade do Software*. [Ref. Diciembre de 2010]. Disponible en Web:
www.cefetrn.br/~placido/disciplina/pgp/material/Wqs2000.pdf
35. FRANCINE, Ellen. YUMI, Elisa. MALDONADO José Carlos. *Towards the Establishment of an Ontology of Software Testing*. [Ref. noviembre de 2010].
Disponible en Web:
<http://www.labes.icmc.usp.br/moduloeducacional/publicacoes/SK06Ellen.pdf>
36. HERNANDEZ SAMPIER, Roberto. *Metodología de la Investigación 1*. Cap. 8, Pág 219-221.

Bibliografía

SAMPER, José Javier. *Ontologías Para Servicios Web Semánticos De Información De Tráfico: Descripción Y Herramientas De Explotación* [Ref. Diciembre de 2010]. Disponible en Web: [Http://Www.Tdr.Cesca.Es/Tesis_Uv/Available/Tdx-0628106_085805//Samper.Pdf](http://www.Tdr.Cesca.Es/Tesis_Uv/Available/Tdx-0628106_085805//Samper.Pdf)

LAMAS, Maria Isabel. *Lenguajes de consulta para documentos RDF*. [Ref. Diciembre de 2010]. Disponible en Web: <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/640/1/38449tfc.pdf>

Jena Semantic Web Framework. [Ref. Diciembre de 2010]. Disponible en Web: <http://jena.sourceforge.net/>

Netbeans. [Ref. febrero de 2011]. Disponible en Web: <http://www.netbeans.org/>

Guía breve sobre Web Semántica. [Ref. noviembre 2010]. Disponible en Web: <http://www.w3c.es/Divulgacion/Guiasbreves/WebSemantica>

Lenguaje de Ontologías Web (OWL). Vista General. [Ref. enero 2011]. Disponible en Web: <http://www.w3.org/2007/09/OWL-Overview-es.html>

The Protégé Ontology Editor and Knowledge Acquisition System. [Ref. octubre 2010]. Disponible en Web: <http://protege.stanford.edu>

HORRIDGE Matthew Protege OWL tutorial at Manchester (School of Computer Science) [Ref. diciembre 2010]. Disponible en Web: <http://owl.cs.manchester.ac.uk/tutorials/protegeowl/tutorial/>

OWL Web Ontology Language Overview. [Ref. enero 2011]. Disponible en Web: <http://www.w3.org/TR/owl-features/>

Resource Description Framework (RDF). . [Ref. enero 2011]. Disponible en Web:
<http://www.w3.org/RDF/>

Components of an Ontology. [Ref. enero 2011]. Disponible en Web:
<http://ontogenesis.knowledgeblog.org/514>

Validator Service. Ref. abril 2011]. Disponible en Web: <http://www.w3.org/RDF/Validator/>

F. Noy, Natalya and L. McGuinness, Deborah. Desarrollo de Ontologías 101. California, EEUU : s.n., 2005.

Ontology management infrastructures En: Ontology Management. Semantic Web, Semantic Web Services, and Business Applications. Springer 2008, e-ISBN: 978-0-387-4.

Hurtado Bustamante, Diana Paola and Sequeda Sanclemente, Juan Federico. Propuesta del Uso de Ontologías para la Búsqueda Semántica en Laboratorios de Investigación y desarrollo: OLID. Cali, Colombia

W3C Candidate Recommendation. [Online] 03 27, 2000. <http://www.w3.org/TR/2000/CR-rdf-schema-20000327/>.

McGuinness, D and van Harmelen, F. World Wide Web Consortium. [Online]
<http://www.w3.org/TR/2003/WD-owl-features-20030331>

Protégé. [Online] <http://protege.stanford.edu/>.

Protege. [Online] <http://protege.stanford.edu/plugins/>.


```
<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#ap
lica_artefactos -->

    <owl:ObjectProperty rdf:about="#aplica_artefactos"/>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#ap
licaproyecto -->

    <owl:ObjectProperty rdf:about="#aplicaproyecto"/>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#co
ntiene_tecnicas -->

    <owl:ObjectProperty rdf:about="#contiene_tecnicas"/>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#co
ordina_herramienta -->

    <owl:ObjectProperty rdf:about="#coordina_herramienta"/>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#co
ordina_proyecto -->

    <owl:ObjectProperty rdf:about="#coordina_proyecto">
      <rdfs:domain rdf:resource="#Especialista"/>
      <rdfs:range rdf:resource="#Proyecto"/>
    </owl:ObjectProperty>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#co
ordina_tprueba -->

    <owl:ObjectProperty rdf:about="#coordina_tprueba"/>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#ej
ecuta_tprueba -->
```

```
<owl:ObjectProperty rdf:about="#ejecuta_tprueba"/>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#es
coordinadopor_especialista -->

<owl:ObjectProperty rdf:about="#escoordinadopor_especialista"/>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#es
generadode_artefactos -->

<owl:ObjectProperty rdf:about="#esgeneradode_artefactos"/>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#es
partede_estrategia -->

<owl:ObjectProperty rdf:about="#espartede_estrategia"/>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#es
partede_evaluaciones -->

<owl:ObjectProperty rdf:about="#espartede_evaluaciones"/>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#es
partede_tprueba -->

<owl:ObjectProperty rdf:about="#espartede_tprueba"/>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#es
responsable_especialista -->

<owl:ObjectProperty rdf:about="#esresponsable_especialista">
  <rdfs:range rdf:resource="#Especialista"/>
  <rdfs:domain rdf:resource="#Proyecto"/>
</owl:ObjectProperty>
```

```
<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#ge
nera_nconformidades -->

<owl:ObjectProperty rdf:about="#genera_nconformidades"/>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#pr
esenta_herramienta -->

<owl:ObjectProperty rdf:about="#presenta_herramienta"/>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#ti
ene_actividades -->

<owl:ObjectProperty rdf:about="#tiene_actividades"/>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#ti
ene_artefacto -->

<owl:ObjectProperty rdf:about="#tiene_artefacto"/>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#ti
ene_criterioc -->

<owl:ObjectProperty rdf:about="#tiene_criterioc"/>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#ti
ene_documentacion -->

<owl:ObjectProperty rdf:about="#tiene_documentacion"/>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#ti
ene_especialista -->

<owl:ObjectProperty rdf:about="#tiene_especialista"/>
```

```
<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#ti
ene_evaluacion -->

    <owl:ObjectProperty rdf:about="#tiene_evaluacion"/>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#ti
ene_niveles -->

    <owl:ObjectProperty rdf:about="#tiene_niveles"/>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#ti
ene_tprueba -->

    <owl:ObjectProperty rdf:about="#tiene_tprueba"/>

<!--

////////////////////////////////////
////////////////////////////////////
//
// Classes
//

////////////////////////////////////
////////////////////////////////////
-->

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Ac
tividades_de_Prueba -->

    <owl:Class rdf:about="#Actividades_de_Prueba">
        <rdfs:subClassOf rdf:resource="#Fundamentos_de_Pruebas"/>
    </owl:Class>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Ar
tefactos -->

    <owl:Class rdf:about="#Artefactos">
```

```
<rdfs:subClassOf rdf:resource="#Fundamentos_de_Pruebas"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Automatizadas -->

<owl:Class rdf:about="#Automatizadas">
  <rdfs:subClassOf rdf:resource="#Herramientas"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Criterios_de_Criticidad -->

<owl:Class rdf:about="#Criterios_de_Criticidad">
  <rdfs:subClassOf rdf:resource="#Fundamentos_de_Pruebas"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Criterios_para_Abortar -->

<owl:Class rdf:about="#Criterios_para_Abortar">
  <rdfs:subClassOf rdf:resource="#Criterios_de_Criticidad"/>
  <owl:disjointWith rdf:resource="#Criterios_para_Detener"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Criterios_para_Detener -->

<owl:Class rdf:about="#Criterios_para_Detener">
  <rdfs:subClassOf rdf:resource="#Criterios_de_Criticidad"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Documentaci&#243;n -->

<owl:Class rdf:about="#Documentaci&#243;n">
  <rdfs:subClassOf rdf:resource="#Fundamentos_de_Pruebas"/>
</owl:Class>
```

```
<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Especialista -->

<owl:Class rdf:about="#Especialista">
  <rdfs:subClassOf rdf:resource="#Fundamentos_de_Pruebas"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Estrategia -->

<owl:Class rdf:about="#Estrategia">
  <rdfs:subClassOf rdf:resource="#Fundamentos_de_Pruebas"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Evaluaciones -->

<owl:Class rdf:about="#Evaluaciones">
  <rdfs:subClassOf rdf:resource="#Fundamentos_de_Pruebas"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Fundamentos_de_Pruebas -->

<owl:Class rdf:about="#Fundamentos_de_Pruebas"/>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Herramientas -->

<owl:Class rdf:about="#Herramientas">
  <rdfs:subClassOf rdf:resource="#Fundamentos_de_Pruebas"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Manuales -->

<owl:Class rdf:about="#Manuales">
  <rdfs:subClassOf rdf:resource="#Herramientas"/>
```

```
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Ni
vel_Desarrollador -->

<owl:Class rdf:about="#Nivel_Desarrollador">
  <rdfs:subClassOf rdf:resource="#Niveles_de_Prueba"/>
  <owl:disjointWith rdf:resource="#Nivel_Sistema"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Ni
vel_Sistema -->

<owl:Class rdf:about="#Nivel_Sistema">
  <rdfs:subClassOf rdf:resource="#Niveles_de_Prueba"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Ni
veles_de_Prueba -->

<owl:Class rdf:about="#Niveles_de_Prueba">
  <rdfs:subClassOf rdf:resource="#Fundamentos_de_Pruebas"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#No
_Conformidades -->

<owl:Class rdf:about="#No_Conformidades">
  <rdfs:subClassOf rdf:resource="#Fundamentos_de_Pruebas"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Pr
oyecto -->

<owl:Class rdf:about="#Proyecto">
  <rdfs:subClassOf rdf:resource="#Fundamentos_de_Pruebas"/>
</owl:Class>
```

```
<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Pr
uebas_Funcionales -->

<owl:Class rdf:about="#Pruebas_Funcionales">
  <rdfs:subClassOf rdf:resource="#Tipos_de_Pruebas"/>
  <owl:disjointWith rdf:resource="#Pruebas_NoFuncionales"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Pr
uebas_NoFuncionales -->

<owl:Class rdf:about="#Pruebas_NoFuncionales">
  <rdfs:subClassOf rdf:resource="#Tipos_de_Pruebas"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Te
cnica_BasadaExperiencia -->

<owl:Class rdf:about="#Tecnica_BasadaExperiencia">
  <rdfs:subClassOf rdf:resource="#Tecnicas_de_Pruebas"/>
  <owl:disjointWith rdf:resource="#Tecnicas_CajaBlanca"/>
  <owl:disjointWith rdf:resource="#Tecnicas_Caja_Negra"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Te
cnicas_CajaBlanca -->

<owl:Class rdf:about="#Tecnicas_CajaBlanca">
  <rdfs:subClassOf rdf:resource="#Tecnicas_de_Pruebas"/>
  <owl:disjointWith rdf:resource="#Tecnicas_Caja_Negra"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Te
cnicas_Caja_Negra -->

<owl:Class rdf:about="#Tecnicas_Caja_Negra">
  <rdfs:subClassOf rdf:resource="#Tecnicas_de_Pruebas"/>
</owl:Class>
```

```
<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Te
cnicas_de_Pruebas -->

<owl:Class rdf:about="#Tecnicas_de_Pruebas">
  <rdfs:subClassOf rdf:resource="#Fundamentos_de_Pruebas"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Ti
pos_de_Pruebas -->

<owl:Class rdf:about="#Tipos_de_Pruebas">
  <rdfs:subClassOf rdf:resource="#Fundamentos_de_Pruebas"/>
</owl:Class>

<!-- http://www.w3.org/2002/07/owl#Thing -->

<owl:Class rdf:about="&owl;Thing"/>

<!--

////////////////////////////////////
////////////////////////////////////
//
// Individuals
//

////////////////////////////////////
////////////////////////////////////
-->

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Ap
pEscritorio -->

<owl:Thing rdf:about="#AppEscritorio">
  <rdf:type rdf:resource="#Artefactos"/>
</owl:Thing>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Ap
pWeb -->
```

```
<owl:Thing rdf:about="#AppWeb">
  <rdf:type rdf:resource="#Artefactos"/>
  <genera_nconformidades rdf:resource="#Interfaz"/>
  <genera_nconformidades rdf:resource="#Validacion"/>
</owl:Thing>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#As
nier_E._Gongora_Alcantara -->

<owl:Thing rdf:about="#Asnier_E._Gongora_Alcantara">
  <rdf:type rdf:resource="#Especialista"/>
  <coordina_proyecto rdf:resource="#CICPC"/>
  <coordina_proyecto rdf:resource="#ERP"/>
</owl:Thing>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#CI
CPC -->

<owl:Thing rdf:about="#CICPC">
  <rdf:type rdf:resource="#Proyecto"/>
</owl:Thing>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#CI
POL -->

<owl:Thing rdf:about="#CIPOL">
  <rdf:type rdf:resource="#Proyecto"/>
</owl:Thing>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Ca
rga -->

<Tipos_de_Pruebas rdf:about="#Carga">
  <rdf:type rdf:resource="&owl;Thing"/>
  <presenta_herramienta rdf:resource="#Jmeter"/>
</Tipos_de_Pruebas>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Da
taGenerator -->
```

```
<owl:Thing rdf:about="#DataGenerator">
  <rdf:type rdf:resource="#Herramientas"/>
</owl:Thing>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#De
lmys_Pozo -->

<owl:Thing rdf:about="#Delmys_Pozo">
  <rdf:type rdf:resource="#Especialista"/>
  <coordina_proyecto rdf:resource="#Identidad"/>
  <coordina_herramienta rdf:resource="#Selenium"/>
</owl:Thing>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#De
lvis_Echeverria_Perez -->

<owl:Thing rdf:about="#Delvis_Echeverria_Perez">
  <rdf:type rdf:resource="#Especialista"/>
  <coordina_proyecto rdf:resource="#Identidad"/>
  <coordina_herramienta rdf:resource="#Jmeter"/>
  <coordina_proyecto rdf:resource="#Prisiones"/>
  <coordina_proyecto rdf:resource="#SICI"/>
  <coordina_proyecto rdf:resource="#Sageb"/>
</owl:Thing>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Do
cumentacion -->

<owl:Thing rdf:about="#Documentacion">
  <rdf:type rdf:resource="#Artefactos"/>
  <genera_nconformidades rdf:resource="#Ortografia"/>
</owl:Thing>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#ER
P -->

<Proyecto rdf:about="#ERP">
  <rdf:type rdf:resource="&owl;Thing"/>
</Proyecto>
```

```
<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Es
tres -->

<Tipos_de_Pruebas rdf:about="#Estres">
  <rdf:type rdf:resource="&owl;Thing"/>
  <presenta_herramienta rdf:resource="#Estres"/>
</Tipos_de_Pruebas>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Fu
ncionales -->

<owl:Thing rdf:about="#Funcionales">
  <rdf:type rdf:resource="#Tipos_de_Pruebas"/>
  <presenta_herramienta rdf:resource="#Selenium"/>
</owl:Thing>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#He
ney_Perez -->

<owl:Thing rdf:about="#Heney_Perez">
  <rdf:type rdf:resource="#Especialista"/>
</owl:Thing>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Id
entidad -->

<owl:Thing rdf:about="#Identidad">
  <rdf:type rdf:resource="#Proyecto"/>
</owl:Thing>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#In
terfaz -->

<No_Conformidades rdf:about="#Interfaz">
  <rdf:type rdf:resource="&owl;Thing"/>
</No_Conformidades>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Jm
eter -->
```

```
<owl:Thing rdf:about="#Jmeter">
  <rdf:type rdf:resource="#Herramientas"/>
  <aplicaproyecto rdf:resource="#CICPC"/>
  <aplicaproyecto rdf:resource="#ERP"/>
  <aplicaproyecto rdf:resource="#Sageb"/>
</owl:Thing>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Lisset_Rosas -->

<Especialista rdf:about="#Lisset_Rosas">
  <rdf:type rdf:resource="#owl:Thing"/>
</Especialista>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Liudmila_Sanchez_Almenares -->

<owl:Thing rdf:about="#Liudmila_Sanchez_Almenares">
  <rdf:type rdf:resource="#Especialista"/>
</owl:Thing>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Odelkys_Betancourt -->

<Especialista rdf:about="#Odelkys_Betancourt">
  <rdf:type rdf:resource="#owl:Thing"/>
</Especialista>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Ortografia -->

<No_Conformidades rdf:about="#Ortografia">
  <rdf:type rdf:resource="#owl:Thing"/>
</No_Conformidades>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Prisiones -->

<owl:Thing rdf:about="#Prisiones"/>
```

```
<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#RN
-->

<owl:Thing rdf:about="#RN"/>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Roig_Calzadilla_Diaz -->

<owl:Thing rdf:about="#Roig_Calzadilla_Diaz">
  <rdf:type rdf:resource="#Especialista"/>
  <coordina_proyecto rdf:resource="#Identidad"/>
  <coordina_proyecto rdf:resource="#RN"/>
</owl:Thing>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#SICI
-->

<owl:Thing rdf:about="#SICI"/>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Sageb
-->

<owl:Thing rdf:about="#Sageb">
  <rdf:type rdf:resource="#Proyecto"/>
</owl:Thing>

<!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Selenium
-->

<Herramientas rdf:about="#Selenium">
  <rdf:type rdf:resource="&owl;Thing"/>
  <aplicaproyecto rdf:resource="#RN"/>
  <aplicaproyecto rdf:resource="#SICI"/>
</Herramientas>
```

```

    <!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Tayche_Capote -->

    <Especialista rdf:about="#Tayche_Capote">
      <rdf:type rdf:resource="#owl:Thing"/>
    </Especialista>

    <!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Validacion -->

    <owl:Thing rdf:about="#Validacion">
      <rdf:type rdf:resource="#No_Conformidades"/>
    </owl:Thing>

    <!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Yadira_Machado -->

    <owl:Thing rdf:about="#Yadira_Machado">
      <rdf:type rdf:resource="#Especialista"/>
    </owl:Thing>

    <!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Yaneida_Rondon -->

    <owl:Thing rdf:about="#Yaneida_Rondon">
      <rdf:type rdf:resource="#Especialista"/>
    </owl:Thing>

    <!--
http://www.semanticweb.org/ontologies/2011/1/Ontology1296939668546.owl#Yoanis_Costilla_Camejo -->

    <owl:Thing rdf:about="#Yoanis_Costilla_Camejo">
      <rdf:type rdf:resource="#Especialista"/>
    </owl:Thing>

    <!--

////////////////////////////////////
////////////////////////////////////
//

```

```
// General axioms
//

////////////////////////////////////
////////////////////////////////////
-->

<rdf:Description>
  <rdf:type rdf:resource="&owl;AllDisjointClasses"/>
  <owl:members rdf:parseType="Collection">
    <rdf:Description rdf:about="#Actividades_de_Prueba"/>
    <rdf:Description rdf:about="#Artefactos"/>
    <rdf:Description rdf:about="#Criterios_de_Criticidad"/>
    <rdf:Description rdf:about="#Documentaci&#243;n"/>
    <rdf:Description rdf:about="#Estrategia"/>
    <rdf:Description rdf:about="#Evaluaciones"/>
    <rdf:Description rdf:about="#Herramientas"/>
    <rdf:Description rdf:about="#Niveles_de_Prueba"/>
    <rdf:Description rdf:about="#No_Conformidades"/>
    <rdf:Description rdf:about="#Tecnicas_de_Pruebas"/>
    <rdf:Description rdf:about="#Tipos_de_Pruebas"/>
  </owl:members>
</rdf:Description>
</rdf:RDF>

<!-- Generated by the OWL API (version 2.2.1.1138)
http://owlapi.sourceforge.net -->
```