

**CURRICULO ADAPTATIVO INTELIGENTE BASADO EN ONTOLOGIAS DE
DESCRIPCIÓN DE COMPETENCIAS PARA EL PROGRAMA DE INGENIERIA
DE SISTEMAS DE LA UNIVERSIDAD DE NARIÑO**

EDIER EDISON ARTEAGA DIAZ

**UNIVERSIDAD DE NARIÑO
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA DE SISTEMAS
SAN JUAN DE PASTO
2012**

**CURRICULO ADAPTATIVO INTELIGENTE BASADO EN ONTOLOGIAS DE
DESCRIPCIÓN DE COMPETENCIAS PARA EL PROGRAMA DE INGENIERIA
DE SISTEMAS DE LA UNIVERSIDAD DE NARIÑO**

**TRABAJO DE GRADO PRESENTADO COMO REQUISITO PARCIAL PARA
OPTAR POR EL TÍTULO DE INGENIERO DE SISTEMAS**

EDIER EDISON ARTEAGA DIAZ

Director Ingeniero. MS. MANUEL BOLAÑOS

**UNIVERSIDAD DE NARIÑO
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA DE SISTEMAS
SAN JUAN DE PASTO
2012**

NOTA DE RESPONSABILIDAD

Las ideas y conclusiones aportadas en el siguiente trabajo son responsabilidad exclusiva del autor.

Artículo 1^o del Acuerdo No. 324 de octubre 11 de 1966 emanado del Honorable Consejo Directivo de la Universidad de Nariño.

NOTA DE ACEPTACIÓN

Jurado:

Jurado:

San Juan de Pasto, Octubre de 2012

RESUMEN

En este trabajo se presenta el proceso realizado en la construcción de un prototipo que se encarga de generar un currículo adaptativo que se acoge a los conocimientos de un estudiante de Ingeniería de Sistemas, que sea próximo a cursar la materia Estructuras de Información en la Universidad de Nariño. Este prototipo genera el currículo basado en evaluaciones de ciertas competencias todo esto se logra sobre la implementación de un software o servicio web que fue construido previamente usando la plataforma netbeans y otras herramientas complementarias como lo son protege, mysql, librerías importantes, etc.

El proceso que realiza este sistema consta de pasos muy básicos y requisitos esenciales, requisitos como capital humano se habla de estudiante vs profesor para el desarrollo de la actividad del sistema y los pasos que estos deben seguir para llevar a cabo una actividad de generación del currículo para este estudiante de pregrado.

ABSTRACT

This paper presents the processes involved in building a prototype that is responsible for generating an adaptive curriculum that is home to a student's knowledge of systems engineering, which is next to take the subject information structures at the University of Nariño. This prototype generates the curriculum based on assessments of certain powers all this is achieved on the implementation of software or web service that was previously constructed using the netbeans platform and other complementary tools such as protecting, mysql, major libraries, etc.

The process that makes this system consists of steps very basic and essential requirements, human capital requirements as we talk about student vs teacher for the development of system activity and the steps they must follow to carry out an activity curriculum generation the undergraduate student.

CONTENIDO

pág.

1.	INTRODUCCION	11
2.	MARCOTEORICO	15
2.1	ONTOLOGIAS	15
2.1.1	Modelado de ontologías.	16
2.1.2	Razonamiento sobre ontologías. Cuando se habla de razonamiento el..	17
2.2	METADATOS DE EMPAQUETADO Y SECUENCIACIÓN DE CURSOS	19
2.2.1	LOM (Learning Object Metadata).	19
2.2.2	IMS MD (Meta-Data).....	22
2.2.3	IMS CP (Content Packaging).....	23
2.2.4	IMS-LD (Learning Design Specification).....	26
2.2.5	IMS SS (Secuencia Simple).	30
2.2.6	SCORM (Sharable Content Object Reference Model).....	33
2.2.7	Situación actual de la composición automática de cursos virtuales.	33
2.2.8	Plataformas LCMS (Learning Management System).....	35
2.3	COMPETENCIAS	39
2.3.1	Definición.....	39
2.3.2	Estándares y especificaciones.	39
2.4	CURRICULO	41
2.4.1	Currículo basado en competencias.	41
3.	METODOLOGIA	44
4.	DESARROLLO DEL PROYECTO.....	45
4.1	IDENTIFICACIÓN DEL Espacio ACADÉMICO	45
4.2	DESCRIPCIÓN DEL ESPACIO ACADÉMICO	46
4.3	IDENTIFICACIÓN DE COMPETENCIAS.....	46
4.3.1	Competencia central.....	46
4.3.1	Competencias específicas.....	46
4.4	IDENTIFICACIÓN DE LAS HERRAMIENTAS DE DESARROLLO.....	47
4.5	CONSTRUCCIÓN DE LA ONTOLOGÍA	52
4.6	DESARROLLO DEL PROTOTIPO.....	55
4.6.1	Análisis.	57
4.6.2	Diseño.	79
5.	CURRICULO ADAPTATIVO	95
5.1	ONTOLOGÍA.....	95

5.1.1	Objetivos de la ontología.	95
5.1.2	Recursos de la ontología.	96
5.1.3	Formato de un recurso.	96
5.1.4	Ontología currículo adaptativo: CurrículoAdaptativo.owl.	97
5.2	REQUERIMIENTOS DE LA APLICACIÓN.....	98
5.3	ANÁLISIS DE FUNCIONALIDADES	98
5.3.1	Funcionalidades importantes.....	99
5.4	EMPAQUETAMIENTO ESTÁNDAR IMS-CP.....	101
5.5	ESTRUCTURA DE CAPAS.....	102
6.	TRABAJOS FUTUROS.....	106
7.	CONCLUSIONES	107
8.	RECOMENDACIONES.....	108
	BIBLIOGRAFIA.....	109
	ANEXOS.....	112

LISTA DE FIGURAS

	pág.
Figura 1. Clases y subclases	15
Figura 2. Slots o propiedad dirige_a	16
Figura 3. RDF/XML definición de clases (Persona y Estudiante).....	16
Figura 4. Arquitectura de JENA.	17
Figura 5. Arquitectura de JENA.	18
Figura 6. La raíz de "vista de árbol" de la hoja de meta-datos.....	22
Figura 7. Elementos y la estructura del esquema conceptual LOM	23
Figura 8. Modelo conceptual IMS Content Packaging. [3]	25
Figura 9. Modelo del curso adaptativo	37
Figura 10. Instalación Protege	47
Figura 11. Entorno de las clases en Protege.	48
Figura 12. Entorno de las propiedades en Protege.....	49
Figura 13. Entorno de las data propiedades en Protege.....	50
Figura 14. Entorno de los individuos en Protege.	51
Figura 15. Tripleta.....	52
Figura 16. Estructura jerárquica de la ontología Clases y Subclases	54
Figura 17. Relaciones entre clases.....	54
Figura 18. Data propiedades y los individuos de las clases.....	54
Figura 19. Contenido estudiante	65
Figura 20. Contenido profesor	66
Figura 21. Registrar estudiante.....	66
Figura 22. Registrar profesor	67
Figura 23. Iniciar sesión estudiante	67
Figura 24. Iniciar sesión profesor.....	68
Figura 25. Registrar un estudiante a un curso	68
Figura 26. Matricular un estudiante a un curso	69
Figura 27. Ingresar un nuevo curso	69
Figura 28. Ingresar prerrequisito a un curso	70
Figura 29. Dar de alta/baja un curso.....	70
Figura 30. Ingresar nueva competencia.....	71
Figura 31. Modelo conceptual del prototipo	72
Figura 32. Registrarse como estudiante	73
Figura 33. Registrarse como profesor.....	74
Figura 34. Logearse como estudiante.....	74
Figura 35. Logearse como profesor	75
Figura 36. Registrarse a un curso.....	75
Figura 37. Matricularse a un curso.....	76
Figura 38. Ingresar nuevo curso	76
Figura 39. Ingresar nuevo prerrequisito	77

Figura 40.	Dar de alta o de baja a un curso	77
Figura 41.	Ingresar nueva competencia.....	78
Figura 42.	Iniciar sesión estudiante	79
Figura 43.	Iniciar sesión profesor.....	81
Figura 44.	Registrar estudiante.....	82
Figura 45.	Registrar profesor	84
Figura 46.	Registrar un estudiante a un curso	86
Figura 47.	Matricular un estudiante a un curso	87
Figura 48.	Ingresar un nuevo curso	89
Figura 49.	Ingresar prerrequisito a un curso	90
Figura 50.	Dar de alta/baja un curso.....	91
Figura 51.	Ingresar nueva competencia.....	92
Figura 52.	Diagrama de clases	94
Figura 53.	Modelo de clases de la ontología	97
Figura 54.	Archivo xml de empaquetado IMS_CP	102
Figura 55.	Estructura de capas del sistema	103
Figura 56a.	Pantallazos finales de la aplicación	104
Figura 56b.	Pantallazos finales de la aplicación	105
Figura 57.	Interfaz principal.....	113
Figura 58.	Interfaz iniciar sesión	114
Figura 59.	Interfaz registrar estudiante	115
Figura 60.	Interfaz registrar profesor.....	115
Figura 61.	Interfaz contenido estudiante.....	116
Figura 62.	Interfaz contenido profesor	117
Figura 63.	Interfaz ingresar prerrequisito a un nuevo curso.....	118
Figura 64.	Interfaz ingresar nuevo curso	118
Figura 65.	Interfaz ingresar nueva competencia.....	119

1. INTRODUCCION

La tecnología de información se encuentra presente en casi todos los campos de aprendizaje por ende es fácil deducir que el conocimiento también, una manera óptima de llegar a él ha sido el uso de Internet que brinda de manera rápida cantidades de información. Hoy en día las instituciones educativas se preocupan por llegar a aquellos estudiantes que buscan en estas la mejor manera de aprender y dar rienda a sus aptitudes y conocimiento. Las universidades optan por carreras y cursos diseñados de forma estratégica para lograr un buen posicionamiento utilizando múltiples formas de difundir sus competencias.

La generación de currículos tiene un alto nivel de complejidad debido a que debe poseer una coherente estructura que permita al estudiante una fácil comprensión de su contenido, es decir la información más importante de los objetos de aprendizaje sobre un determinado tema como lo son las estructuras de información, para lograr la construcción de un currículo que se adapte inteligentemente a los requerimientos de un estudiante que busca enfatizar en esta área se necesitará de la ayuda de ontologías específicas basadas en competencias que determinen de la manera más óptima el contenido correcto, además la utilización de estándares que permiten documentar, buscar y distribuir la información, como lo hace el estándar IMS-CP que describe como empaquetar el contenido de la información y el buen manejo de los metadatos.

En la actualidad existen herramientas para la creación de cursos orientados a la web que sirven de apoyo en la obtención del conocimiento, este prototipo además de generar ese apoyo construirá un currículo adaptado principalmente a exigencias de un estudiante logrando así disminuir la posibilidad de que por el hecho de realizar un curso ya existente no han logrado su satisfacción educativa y ha desertado en la culminación de su estudio.

1.1 TEMA

1.1.1 Título. Currículo Adaptativo Inteligente basado en ontologías de descripción de competencias para el programa de Ingeniería de Sistemas de la Universidad de Nariño.

1.1.2 Modalidad. El presente anteproyecto de trabajo de grado se enmarca en la modalidad de Trabajo de Investigación.

1.1.3 Línea de investigación. La presente propuesta de trabajo de grado se encuentra inscrita bajo la línea de Sistemas computacionales.

1.1.4 Alcance y delimitación. Este currículo adaptativo inteligente es un prototipo, que permite a un estudiante crear su curso de estructuras de información adaptado a sus conocimientos y habilidades previas en el área.

La implementación del prototipo se realizó mediante la utilización de paquetes IMS-CP utilizando registros de metadatos basados en ontologías de descripción de competencias de manera aleatoria preferiblemente con estudiantes y docentes de la Universidad de Nariño.

1.2 DESCRIPCIÓN DEL PROBLEMA

1.2.1 Planteamiento del problema. En la actualidad se encuentran muchos estudios sobre plataforma LCMS, en cuanto al desarrollo y utilización de diferentes elementos y herramientas como objetos de aprendizaje, repositorios de registros de metadatos entre otros, para apoyar los procesos de aprendizaje. Pero existe una carencia respecto al uso de las competencias curriculares en la composición de cursos. Este tipo de carencia sugiere la posibilidad de crear un nuevo escenario en el que se utilicen los estándares definidos por organizaciones como IEEE, IMS Learning Global Consortium y ARIANE, y especificaciones de competencias para la composición automática de cursos personalizados en función del currículo deseado por una persona, teniendo en cuenta sus habilidades previas (competencias).

1.2.2 Formulación del problema. ¿De qué manera un estudiante interesado en aprender y estudiar estructuras de datos, puede lograr conseguir un currículo adecuado en función de sus conocimientos, habilidades y destrezas?

1.2.3 Sistematización del problema. ¿Cómo recolectar información relacionada y correcta del requerimiento pedido por un usuario del sistema?

¿De qué manera se optimiza la recolección de la información suministrada por la persona interesada en este proceso de aprendizaje?

¿Cuáles son los beneficios básicos que este tipo de currículo generaría?

¿De qué forma se manejará la información en el transcurso de la generación del currículo deseado?

¿El uso de metadatos ayudará a la elaboración del currículo inteligente?

¿Qué documentación se debe elaborar a partir de la implementación de este currículo adaptativo inteligente?

1.3 OBJETIVOS

1.3.1 Objetivo general. Desarrollar un currículo adaptativo inteligente basado en ontologías de descripción de competencias para el Programa de Ingeniería de Sistemas de la Universidad de Nariño.

1.3.2 Objetivos específicos.

- Recolectar y analizar información sobre currículo basado en competencias.
- Definir el diseño de la ontología y la herramienta en la cual será implementada.
- Identificar los requerimientos de hardware y software necesarios para la implementación de un prototipo que genera un currículo adaptativo inteligente.
- Implementar los servicios básicos de la arquitectura logrando la generación de un currículo coherente.
- Implementar un prototipo que permita generar un currículo adaptativo inteligente basado en ontologías de competencias para estructuras de información en el programa de Ingeniería de Sistemas de la Universidad de Nariño
- Desarrollar la documentación necesaria para la comprensión y utilización del prototipo resultante.

1.4 JUSTIFICACIÓN

En la actualidad existe una carencia respecto al uso de las competencias curriculares en la creación de cursos, lo cual hace necesario la utilización de estándares definidos por diferentes organizaciones y especificaciones de competencias para la composición automática de cursos basados en las habilidades y conocimientos de un estudiante.

La implementación de un prototipo de un currículo adaptativo inteligente que se basa en ontologías de descripción de competencias, permitirá a un estudiante generar un curso coherente con sus competencias (ser, hacer, y saber hacer), garantizando de alguna manera un mejor desempeño y comprensión de las temáticas abordadas o tratadas en el curso.

Con el desarrollo de este proyecto se pretende solucionar en parte el inconveniente que presentan muchas de las plataformas existentes que tienen una inclinación comercial, por lo que el contenido de los cursos generados se basa

en una temática específica y rígida, que en la mayoría de los casos no se adecua a los requerimientos propios de cada estudiante, por ende los procesos de aprendizaje no se desarrollan de la mejor manera conllevando así a un alto nivel de deserción.

2. MARCOTEORICO

2.1 ONTOLOGIAS

Recurrir a la red es cada vez más común cuando se habla de investigar para dar con la información necesitada, ¿esta información será la correcta o la más acertada? debido a la amplia variedad de la misma las Ontologías ayudan cumpliendo un papel importante para solución a este labor sirviendo como modelo de representación de conocimiento.

Una ontología puede servir como almacenamiento de contenido (base de datos) pero no es una base de datos, se puede poner código en ella pero no es un programa, se definen términos pero no es un diccionario. Una ontología es una instancia de conceptos y reglas de inferencia relacionados entre sí de manera que proveerán de una visión clara a medida que se desarrolla la aplicación para la cual se definió la ontología esta a su vez compuesta de estos conceptos que se manejan en vez de palabras clave que es lo que hace a una ontología importante en el campo de la web semántica ya que permite la inferencia de nuevo conocimiento, además maneja un esquema que permite datar el contenido de la información e información de esta información para lograr lo antes mencionado.

Para entender más, se debe tener presente ciertas definiciones de los elementos que rodean a una ontología:

- Conceptos: se centra en la declaración de la idea fundamental para llevar a cabo la formalización estos conceptos pueden ser entidades, clases, procesos etc.
- Relaciones: es el enlace entre los conceptos definidos anteriormente y su relación.
- Instancias: utilizados en la representación de los objetos de la ontología.

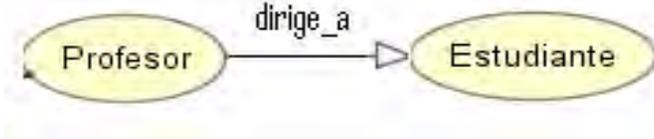
Cuando se va a desarrollar una ontología siempre se debe definir previamente un dominio, es decir, el campo de acción de la aplicación o sus límites, posteriormente como se observa en la figura 1 donde se definen las clases en forma estructural de manera jerárquica.

Figura 1. Clases y subclases



Posteriormente se definen las propiedades figura 2 (relaciones) o slots.

Figura 2. Slots o propiedad dirige_a



Con las ontologías se logra mayor expresividad para ello se necesita un lenguaje el existente para este contexto es OWL y RDF, que son los estándares manejados para la web semántica necesarios para la administración de los datos en la web. RDF utiliza lenguaje XML, define modelos de datos así representa objetos en la web al igual que XML, utiliza su URI (Identificador Uniforme de Recursos) para la identificación y posterior manejo, además usa tripletas (objeto – predicado - objeto) de cada concepto dentro de la ontología (Figura 3).

Figura 3. RDF/XML definición de clases (Persona y Estudiante)

```
<!-- http://www.semanticweb.org/ontologies/2011/CurriculoAdaptativo.owl#persona -->
<owl:Class rdf:about="#persona"/>

<!-- http://www.semanticweb.org/ontologies/2011/CurriculoAdaptativo.owl#estudiante -->
<owl:Class rdf:about="#estudiante">
  <rdfs:subClassOf rdf:resource="#persona"/>
</owl:Class>
```

OWL en cambio es un poco más descriptivo que RDF ya que posee más vocabulario a la hora de datar las propiedades y conceptos de una ontología. OWL se compone a su vez de OWL Lite, OWL DL, OWL Full que extienden sus funciones dependiendo el contexto y manera de aplicarlos.

2.1.1 Modelado de ontologías. Se puede modelar una Ontología con diferentes técnicas de representación de conocimiento e implementarlas en diferentes lenguajes. Las ontologías pueden ser informales, semi-informales, semi-formales y rigurosamente formales [27].

Prof. Iván J. menciona que “Las Ontologías fueron construidas utilizando principalmente técnicas de representación de conocimiento en Inteligencia Artificial (IA), como son los Marcos y la Lógica de Primer Orden” [27].

Para modelar ontologías se han utilizado técnicas como: lógica descriptiva utilizando distintos lenguajes para tal fin, el Lenguaje de Modelado Unificado (UML) en Ingeniería de Software y Diagramas E/R, utilizados en Base de Datos para modelar conceptos y las relaciones entre estos. Véase [27].

2.1.2 Razonamiento sobre ontologías. Cuando se habla de razonamiento el Prof. Iván J. da a entender su visión “Uno de los aspectos de la inteligencia es la capacidad de razonar, es decir, la capacidad de obtener nueva información de la ya disponible. En la IA existen diversas técnicas para hacer razonamiento aplicando diversas estrategias de inferencia sobre el conocimiento” [27].

“Para cada formalismo de representación del conocimiento existen varias técnicas que permiten la obtención de nuevo conocimiento. Estas responden en gran parte a la naturaleza del mismo, pero también dependerá del tipo de manipulación que se vaya a realizar, según los objetivos que se persigan. Para razonar con Ontologías principalmente se utilizan tres técnicas: Razonamiento con Lógica de Primer Orden (LPO), Razonamiento con Lógica Descriptiva (DL) y Razonamiento con Reglas” [27].

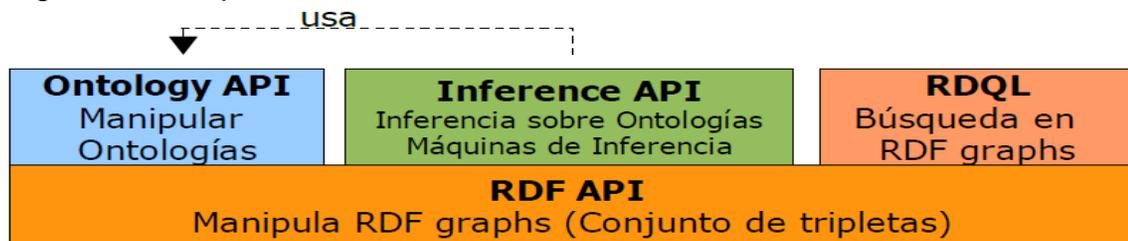
Para ampliar este tema diríjase a [27].

2.1.2.1 Herramientas de razonamiento utilizadas con ontologías. A continuación se menciona de forma rápida una herramienta de razonamiento utilizada para el desarrollo de aplicaciones que hace uso de representaciones Ontológicas.

Jena: es un Framework de Java para desarrollar aplicaciones de Web Semántica. Ella provee un ambiente para manipulación de Ontologías representadas en RDF, RDFS y OWL (<http://jena.apache.org/>). [27]

Su arquitectura se aprecia a continuación (Figura 4):

Figura 4. Arquitectura de JENA.



[27].

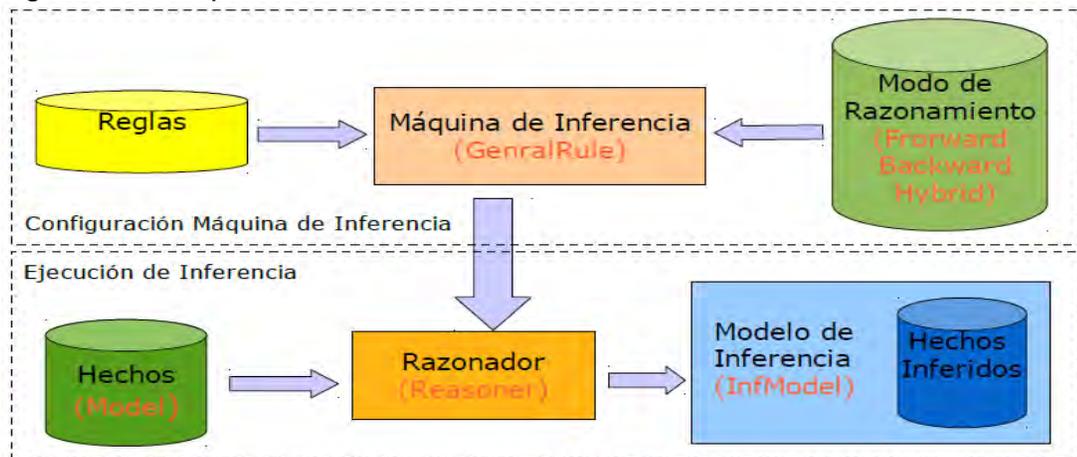
Como muestra la anterior figura esta arquitectura consta de cuatro apartados: La API de Inferencia: “La API de inferencia se caracteriza por ofrecer distintos tipos de razonadores para Ontologías e intenta crear un modelo de Inferencia, el

cual inferirá los hechos sobre un modelo dado.” [27] En la siguiente figura se puede ver este proceso.

“La Inferencia, se hace en dos etapas. Primero, se debe configurar la máquina de inferencia indicando las reglas y la forma en que se realizará la inferencia para instanciar un razonador. Segundo, una vez instanciado el razonador, se suministran los hechos y se obtiene un modelo de inferencia que inferirá los nuevos hechos” [27].

“Para razonar con Ontologías es necesario que ésta sea cargada en el Framework, haciendo uso de la API de Ontología, para que JENA intente crear el modelo RDF que utilizará” [27]. A continuación se ve en la Figura 5.

Figura 5. Arquitectura de JENA.



[27].

Razonadores de JENA:

a) Razonador general de reglas: “Es un razonador general basado en reglas y en éste se basan los demás razonadores que ofrece JENA. Este razonador puede inferir hechos utilizando encadenamiento hacia adelante, hacia atrás e híbrido. La estructura de las reglas está definida por una lista de antecedentes (body), una lista de consecuentes (head), un nombre para la regla (opcional) y una dirección que representa la forma en que se resuelve la regla” [27].

b) Razonador RDF(S): Razonador basado en reglas que soporta todas las implicaciones sobre RDFS [27].

c) Razonador OWL: Razonador para OWL que sólo puede ser utilizado para razonar sobre OWL/Lite [27].

d) Razonador transitivo: se utiliza para soportar datos almacenados con clases y propiedades. Su implementación sólo soporta propiedades de `rdfs:subPropertyOf` o `rdfs:subClassOf` [27].

2.2 METADATOS DE EMPAQUETADO Y SECUENCIACIÓN DE CURSOS

2.2.1 LOM (Learning Object Metadata). Según el estándar para metadatos de objetos educativos es “Este Estándar es un estándar multi-parte que especifica Metadatos para Objetos Educativos, LOM especifica un esquema conceptual de datos que define la estructura de una instancia de metadatos para un objeto educativo. Para este Estándar, un objeto educativo se define como cualquier entidad, digital o no, susceptible de ser usada en aprendizaje, educación o formación” [1].

Este modelo de datos que sirve para describir objetos educativos fue codificado en XML y tiene como fin reutilizar objetos de aprendizaje para que personas dedicadas a este tipo de proyectos, ya sean estudiantes, profesores y la aplicación en software con fines educativos hagan de la información una forma más fácil de aprender.

En lo que respecta a este Estándar, una instancia de metadatos para un objeto educativo describe las características relevantes del objeto educativo al que se aplica. Dichas características se pueden agrupar en las categorías general, ciclo de vida, meta-metadatos, técnica, uso educativo, derechos, relación, anotación y clasificación. [1]

2.2.1.1 Propósito de esta norma. Según el estándar para metadatos de objetos educativos es “El propósito de este Estándar multi-parte es facilitar la búsqueda, evaluación, adquisición y uso de los objetos educativos, por ejemplo, por alumnos, profesores o procesos automáticos de software. Este estándar multi-parte también facilita el intercambio y uso compartido de objetos educativos, permitiendo el desarrollo de catálogos e inventarios al tiempo que se toman en consideración la diversidad cultural y los contextos lingüísticos en los que los objetos educativos y sus metadatos serán reutilizados” [1].

Esta norma también sirve en propósitos para:

- Habilitar a los alumnos o instructores a buscar, evaluar, adquirir y utilizar los objetos de aprendizaje.
- Habilitar el uso compartido y el intercambio de objetos de aprendizaje a través de cualquier tecnología de Sistemas de aprendizaje.
- Permitir el desarrollo de objetos de aprendizaje en las unidades que se pueden combinar y se descomponen de manera significativa.

- Habilitar a los agentes de equipo de forma automática y dinámica a componer lecciones personalizadas para un estudiante individual.
- Complementar el trabajo directo en las normas que se centran en capacitar a varios Objetos de Aprendizaje para trabajar juntos dentro de un entorno abierto de aprendizaje distribuido.
- Permitir, si así lo desea, la documentación y el reconocimiento de la finalización de los actuales o nuevos aprendizajes y los objetivos de rendimiento relacionados con objetos de aprendizaje.
- Habilitar una economía fuerte y creciente de objetos de aprendizaje que apoye y sostenga todas las formas de distribución, sin ánimo de lucro, sin fines de lucro y con fines de lucro.
- Habilitar la educación, la formación y las organizaciones de aprendizaje, tanto del gobierno públicos como privado, para expresar contenidos educativos y niveles de rendimiento en un formato estándar que es independiente del contenido en sí.
- Proporcionar a los investigadores de normas que apoyen la recogida e intercambio de datos comparables sobre la aplicabilidad y eficacia de los Objetos de Aprendizaje.
- Definir una norma que es simple pero extensible a varios dominios y jurisdicciones a fin de ser más fácil y ampliamente adoptada y aplicada.
- Dar la seguridad y autenticación necesarias para la distribución y el uso de objetos de aprendizaje.

2.2.1.2 Estructura básica de los metadatos. Estos elementos describen un objeto educativo y se clasifican en las siguientes categorías según el Esquema Base LOM [1]:

- La categoría General agrupa la información general que describe un objeto educativo de manera global.
- La categoría Ciclo de Vida agrupa las características relacionadas con la historia y el estado actual del objeto educativo, y aquellas que le han afectado durante su evolución.
- La categoría Meta-Metadatos agrupa la información sobre la propia instancia de Metadatos, (en lugar del objeto educativo descrito por la instancia de metadatos).

- La categoría Técnica agrupa los requerimientos y características técnicas del objeto educativo.
- La categoría Uso Educativo agrupa las características educativas y pedagógicas del objeto.
- La categoría Derechos agrupa los derechos de propiedad intelectual y las condiciones para el uso del objeto educativo.
- La categoría Relación agrupa las características que definen la relación entre este objeto educativo y otros objetos educativos relacionados.
- La categoría Anotación permite incluir comentarios sobre el uso educativo del objeto e información sobre cuándo y por quién fueron creados dichos comentarios.
- La categoría Clasificación describe este objeto educativo en relación a un determinado sistema de clasificación.

Elementos de datos. “Las categorías agrupan elementos de datos. El modelo de datos de LOM es una jerarquía de elementos de datos, incluyendo elementos de datos agregados y simples (nodos hoja en la jerarquía). En el esquema base LOMv1.0 sólo los nodos hoja tienen valores individuales definidos a través de sus espacios de valores y tipos de datos asociados. En el esquema base LOMv1.0 los elementos de datos agregados no tienen valores individuales. Por lo tanto, no tienen espacios de valores o tipos de datos. Para cada elemento de datos el esquema base LOM v1.0 define” [1]:

- nombre: el nombre de referencia del elemento de datos;
- explicación: la definición del elemento de datos;
- tamaño: el número de valores permitido;
- orden: relevancia de la ordenación de los valores.

La tabla 1 muestra el encabezado de la tabla que representa al esquema Base LOMv1.0.

Tabla 1. Esquema Base LOMv1.0.

No	Nombre	Explicación	Tamaño	Orden	Espacio de valores	Tipo de datos	Ejemplo

2.2.2 IMS MD (Meta-Data). Es un estándar de metadatos, antes conocido como IMS Learning Resource Meta-data (LRM) que se diseñó para garantizar la transformación de datos para que estos se adapten a las necesidades de los usuarios en particular. MD proporciona descripción y acceso potente para gestionar las unidades de aprendizaje gracias a sus funciones como lo son su vocabulario y sus campos.

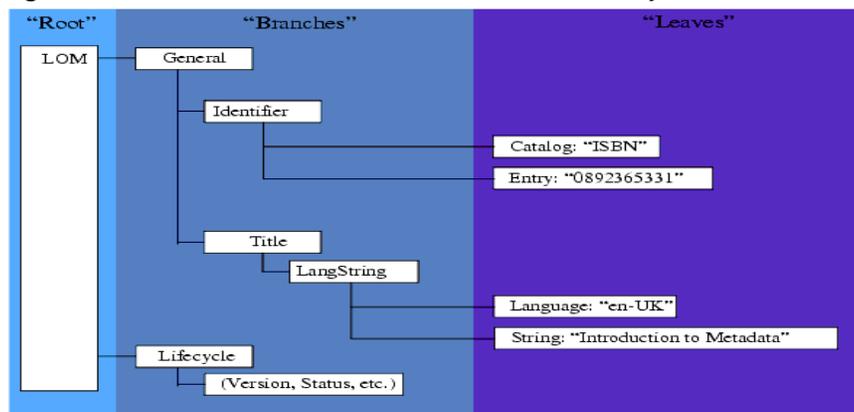
Se pueden utilizar las etiquetas a la hora de manejar cualquier clase de recursos y así poderlos encontrar y manipular, en el caso de los recursos de aprendizaje esto no es excepción estas etiquetas vienen siendo los “datos sobre los datos” IMS-LD es la declaración formal de estas etiquetas.

IMS MD tiene como objetivo hacer que el proceso del uso y búsqueda de un recurso de aprendizaje se logre de manera eficiente porque provee de una estructura bien definida y descriptiva para poder utilizar y representar los elementos. [2]

2.2.2.1 Alcance. La norma IEEE LOM es la encargada de definir el conjunto de elementos de metadatos que se pueden utilizar para describir recursos de aprendizaje. Esto incluye los nombres de elementos, definiciones, tipos de datos, y la longitud de los campos. El estándar se conoce como un estándar multi-parte y define un modelo conceptual para los “meta-datos”. La norma incluye las declaraciones de conformidad para la forma de meta-datos como deben ser organizados y cómo las aplicaciones deben comportarse para ser considerados IEEE. [2]

2.2.2.2 Información general. Como muestra la figura 6 el esquema de datos conceptual IEEE para definir los metadatos es jerárquico, posee su base que es la raíz y sus sub elementos que se consideran las “ramas” , y los subelementos que ya no poseen más subelementos serán las” hojas” de este árbol. [2]

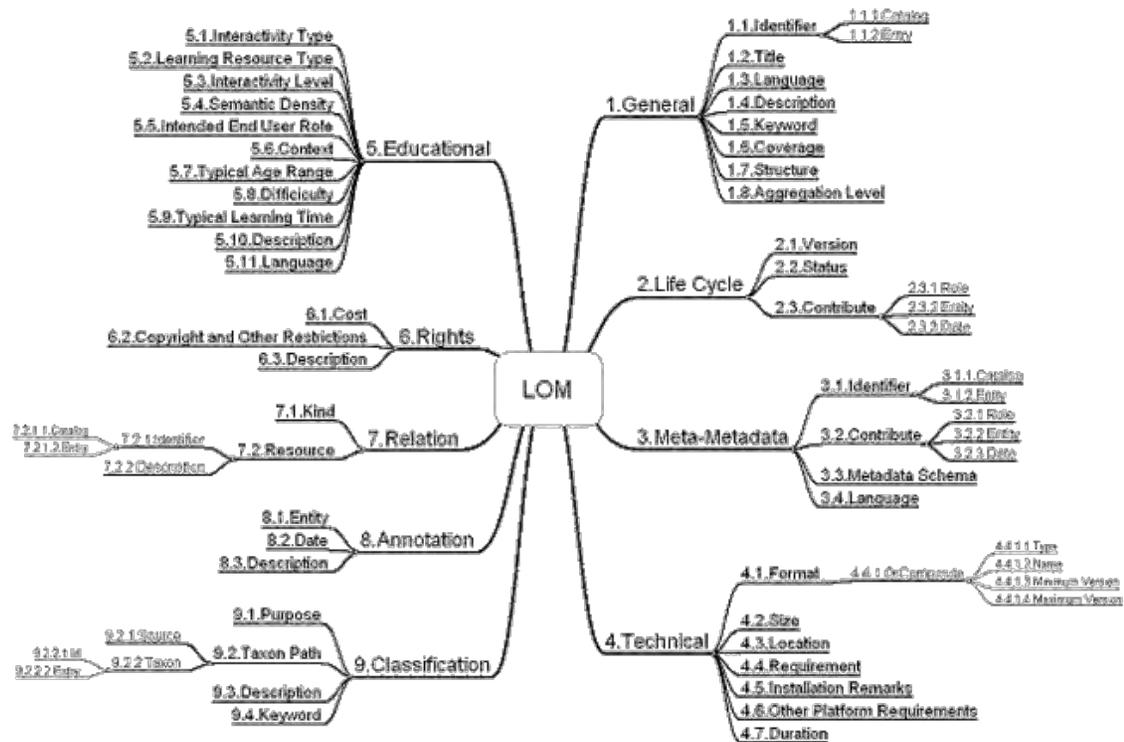
Figura 6. La raíz de "vista de árbol" de la hoja de meta-datos.



[2]

2.2.2.3 IEEE datos meta-elementos, estructura y tipos de datos. La figura 7 representa los elementos en el esquema de datos, muestra como los elementos se categorizan de nueve maneras diferentes todas estas de nivel superior a la vez estas se subdividen en ramas u hojas según el caso, además cada elemento de la hoja en este esquema LOM tiene un tipo de datos que define la codificación de los datos de este elemento, a veces este tipo de datos restringen la manera como un elemento puede ser utilizado. [2]

Figura 7. Elementos y la estructura del esquema conceptual LOM



[2]

2.2.3 IMS CP (Content Packaging). La especificación IMS Content Packaging de IMS (de ahora en adelante, IMS CP) describe las estructuras de datos que se utilizan para gestionar datos entre sistemas que desean importar, exportar, agregar y desagregar los paquetes de contenido en la utilización de objetos de aprendizaje además describe el medio de contención de estos paquetes. IMS CP se concentra en el empaquetamiento y el transporte de los recursos pero no determina la naturaleza de esos recursos. Debido a la existencia de infinidad de formatos con los cuales los que usan esta especificación cuentan a la hora de manejar sus recursos. [3]

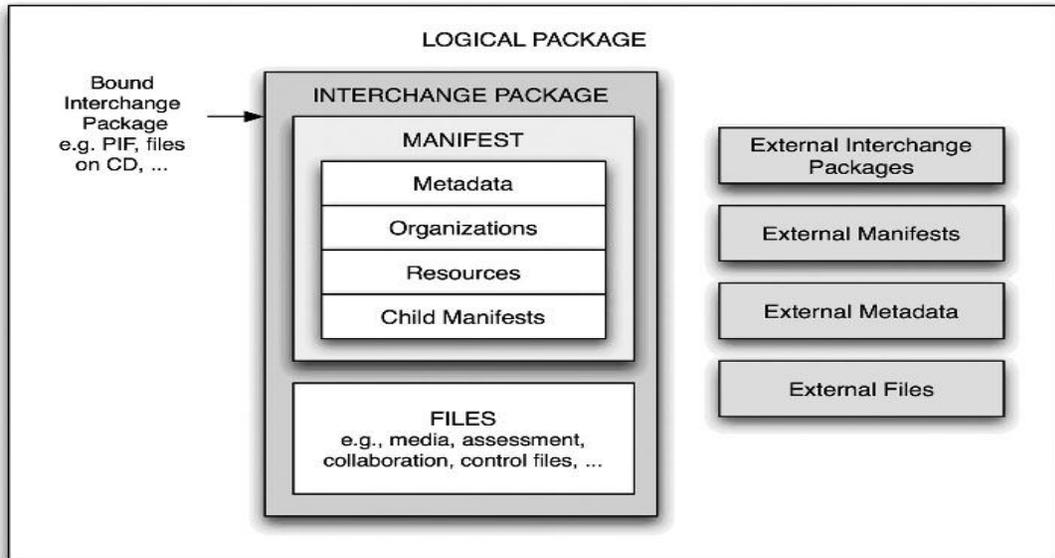
IMS-CP facilita la *interoperabilidad* entre los sistemas de e-learning gracias a que es capaz de abrir los paquetes IMS, si tener en cuenta la manera en que estos paquetes hayan sido producidos.

2.2.3.1 Descripción general. Los paquetes IMS se componen de dos elementos. El primero es el manifiesto, un documento XML en el que se describen los contenidos encapsulados y su organización. El segundo son los propios contenidos educativos, descritos en el manifiesto, tales como páginas Web, ficheros de texto, objetos de evaluación o cualquier otro tipo de material de datos. Cuando estos elementos se encapsulan en un fichero único (Ej. un fichero comprimido .zip, .jar, o .cab), el archivo resultante se denomina fichero de intercambio de paquete. [4]

Dentro del manifiesto existe el sub-elemento "metadatos" utilizado para describir en general el contenido empaquetado bajo un esquema de metadatos de objetos de aprendizaje. Ejemplos válidos pueden ser IEEE LOM, IMS-MD, LTSN, UKCMF o algún otro perfil de aplicación de los citados. Gracias a la descripción realizada bajo estos esquemas, el objeto de aprendizaje empaquetado puede ser localizado por sistemas de búsqueda externos. Por ejemplo, un objeto descrito bajo el perfil de aplicación LOM-LTSN [FT01] puede ser identificado y clasificado dentro de un árbol jerárquico gracias a la categoría "clasificación" del esquema de metadatos utilizado, facilitando así la localización y clasificación de contenidos para su posterior búsqueda y navegación. [4]

Otro de los componentes fundamentales del manifiesto es el sub-elemento "organizaciones". Este elemento se utiliza para especificar una o varias organizaciones alternativas para los recursos incluidos en el paquete. Cada organización define las relaciones estáticas existentes entre los recursos de la agregación como un árbol jerárquico, donde cada elemento (ítem) se corresponde bien con un recurso educativo o bien con una agregación de elementos (ítems) de menor nivel. Además, en el sub-elemento "recursos" dentro del manifiesto, se incluyen las referencias a todos los recursos utilizados. La especificación define dos tipos posibles de referencias: locales y globales. Las referencias locales son utilizadas para localizar únicamente los recursos almacenados dentro del paquete. La localización es posible gracias al uso de identificadores internos (Ej. RESOURCE-10221). Por otro lado, las referencias globales permiten el uso de recursos externos al paquete. En este tipo de referencia, el direccionamiento es posible gracias al uso de URL's. Cada recurso en IMS Content Packaging (IMS-CP) puede contener una o varias referencias a varios ficheros. En el caso de una página Web, si en esta existen varias imágenes, en el recurso aparecerán las referencias al fichero HTML y a las imágenes contenidas (Figura 8). [4]

Figura 8. Modelo conceptual IMS Content Packaging. [3]



2.2.3.2 Tipos de paquetes

El paquete simple: Se basa en el uso de un archivo .Zip para su intercambio, solo supone un nivel de manifiesto. [3]

El manifiesto al descubierto: No incluye todos los recursos en el intercambio de archivos. Se hacen referencias de enlaces y provee ventajas como la reutilización, actualización, rastreo de los recursos siempre que sea necesario, la desventaja radica en la no confidencialidad de todos los recursos. [3]

El compuesto o meta-paquete: Lleva el manifiesto al descubierto un paso más allá al vincular a otros paquetes completos en lugar de sólo los recursos convencionales, Como las páginas Web.

La principal ventaja del paquete compuesto es que tiene la reutilización de contenido. La desventaja es que la funcionalidad que permite a este tipo de paquetes es relativamente nueva, lo que significa que no todos los lectores de paquetes serán capaces de soportar todas las funcionalidades. [3]

El paquete de archivo: Se utiliza para almacenar los recursos y la definición de las estructuras básicas para cuando se necesiten. El uso de estos paquetes sirve para el intercambio de los mismos sin procesar su contenido. [3]

Paquetes especializados: Se utilizan para transmitir tipos de contenidos muy específicos. La mayoría de estos paquetes son combinaciones de paquetes con

contenidos definidos en otras especificaciones de IMS, en el núcleo de estos paquetes se encuentra la descripción de los datos específicos de un o un IMS Content Packaging simplemente proporciona una manera conveniente de agregar descripciones. Esta especificación es muy amplia y para mayor comprensión se puede recurrir al el sitio oficial [3].

2.2.4 IMS-LD (Learning Design Specification). Las especificaciones de diseño de aprendizaje IMS Learning Design colaboran a la educación en línea mediante el apoyo a pedagogías y la variedad de trabajo que estas implican mediante la libre expresión de cada una [5].

Trabaja con estructuras complejas permitiendo la interacción de muchos usuarios a la vez, algunas de sus funciones son el intercambio de información y servicio de mensajería. A la hora de reutilizar objetos de aprendizaje IMS LD es complejo en comparación con otras especificaciones similares existentes hoy en día, en la actualidad se adelantan investigaciones que desarrollan y construyen herramientas que gestionen de forma fácil el trabajo con esta especificación para que profesores o usuarios hagan uso eficaz sin necesidad de conocer a fondo el uso de estas especificaciones [6].

2.2.4.1 La contribución de IMS-LD. “En 1997 la Open University of the Netherlands (OUNL) decidió convertir todos sus cursos en cursos on-line”[7].

Todos los cursos presentan tipos de pedagogías específicas debido a la visión de cada profesor y la implementación de plantillas para cada pedagogía sería extensa aunque se estaba haciendo, mas sin embargo en la práctica los tipos de pedagogía se resumen en tres elementos: “recursos educativos, múltiples personas actuando en varios roles, y actividades pedagógicas” [7].

El EML (Educational Modelling Language) es una especificación para definir aspectos pedagógicos. Los tres elementos antes mencionados son definidos por esta especificación “y así especificar la estructura de una Unit of Learning, UoL en un documento XML”, debe existir una instancia definida previamente con ciertos parámetros e integrantes para que pueda ser utilizada [7].

EML sirvió como base para conformar una especificación que supliera las limitaciones que otras especificaciones existentes poseen como resultado de esta conformación surgió IMS Learning Design con cambios en su forma y enfoque igualmente se mantiene la concepción básica que caracterizaba a EML.

Un resumen de los aspectos más relevantes es:

- Ofrece soporte a múltiples alumnos
- Representar el papel de profesor
- Permitir combinar recursos educativos

Un ejemplo práctico donde se reflejan las bondades de IMS LD se da cuando permite modelar actividades de negocios con participación en grupo y documentación necesaria como resultado una estructura que puede aplicarse a otra actividad totalmente diferente pero siempre manteniendo la misma estructura. Un recurso pasa a ser una parte importante en el proceso de aprendizaje cuando es utilizado activamente como método de estudio y no es indispensable, es decir, con IMS LD un recurso es bien aprovechado cuando pasa a ser parte de una Unit of Learning, “La UoL puede tener muchas formas distintas incluso con el mismo recurso. Por ejemplo, un texto que describa una ciudad puede formar parte de una Unit of Learning (UoL) tratando de historia, o de literatura, o de lengua, o de formación de educadores, entre otros temas, y cada contexto tendría actividades y / o roles distintos. Las diferentes UoLs serían reutilizables por profesores distintos, con cohortes distintas de alumnos” [7].

Siempre para un profesor ha sido más importante la calidad de pedagogía que el recurso utilizado a la hora de enseñar por ello IMS LD permite representar correctamente este tipo de información sobre el uso en el aula que combina recursos con actividades y roles [7].

2.2.4.2 Utilización de estructuras de aprendizajes IMS-LD en la enseñanza de la programación. Como se mencionó anteriormente la especificación IMS LD trabaja de manera más competente las exigencias de los docentes a la hora de modelar y utilizar recursos para trabajo educativo permitiendo incluir roles e interacciones, así como estructuras pedagógicas [8].

Hoy en día gracias a la variedad de información que se encuentra el aprendizaje es visto como una manera práctica para escoger el material existente y ofrecerlo de forma estratégica para después observar los resultados que surgen al aplicarlos a los estudiantes.

IMS LD aporta flexibilidad y múltiples recursos que hacen posible implementar escenarios de aprendizaje. Así usar IMS LD para desarrollar objetos de aprendizaje permite un gran avance para la educación, como lo es la adaptación de contenido asociado al saber del estudiante, la colaboración entre alumnos con alumnos y con los profesores, y la monitorización por parte del docente sobre el trabajo del estudiante [8].

Las facilidades que brinda IMS-LD pueden tener múltiples aplicaciones didácticas como por ejemplo: la adaptación de contenido en dependencia del nivel de conocimiento del alumno, la colaboración alumno-alumno o alumno-profesor, o la monitorización por parte del profesor del trabajo del estudiante. Complementando esta parte el estándar IMS-LD aplica muy satisfactoriamente a la hora de enseñar los conceptos de la asignatura de programación, en resumen, parte desde la utilización de la herramienta CooperCore diseñada por el Departamento de

Informática de la UNAH, esta herramienta la única hasta el momento que soporta los tres niveles de IMS-LD, una característica se refleja en que un curso aplica a diferentes tipos de estudiantes y esta aplicación se acopla a diferentes niveles de estudio a través de la determinación de estereotipos y el avance de un alumno de un nivel a otro depende de los resultados de los ejercicios. [8]

Utilizando un estándar para implementar la adaptación en el aprendizaje se crearán módulos o Unidades de Aprendizajes totalmente ínter operable. [8]

Una de las UAs desarrolladas en la UNAH que se encuentra disponible en: <ftp://www.virtua.ua.es/lora/IMSLED/ciclos.zip>, se dedicó al estudio de estructuras cíclicas. [2]

Una de las UAs desarrolladas (<ftp://www.virtua.ua.es/lora/IMSLED/TDA.zip>) está dirigida al trabajo con los Tipos de Datos Abstractos (TDA). [8]

2.2.4.3 Modelado de procesos de enseñanza-aprendizaje. Un proceso puede definirse como el conjunto de actividades o tareas realizadas por entidades máquinas o la persona que se interrelaciona para así lograr un fin previsto, por ejemplo el proceso de matrícula de un estudiante en una institución educativa.

Cuando se habla de proceso un concepto que se relaciona es el de flujo de trabajo más conocido como workflow (automatizar procesos). “Un Sistema de Gestión de Workflow es un sistema informatizado que permite la definición (modelado) de los procesos que constituyen un flujo de trabajo, y la gestión de su ejecución mediante la interpretación de los modelos y la ejecución de las aplicaciones necesarias para la elaboración de los documentos implicados en el trabajo o para la intercomunicación de los participantes en el mismo. Se trata, en definitiva, de automatizar el trabajo en una organización, de tal forma que un usuario que se conecte al sistema de workflow, sepa en todo momento el siguiente trabajo a realizar y disponga, en cada instante, de la información necesaria para ello (por ejemplo, documentos recibidos por correo electrónico)” [9].

Para modelar procesos existe XML que es un lenguaje de programación basado en etiquetas predefinidas usa elementos para referirse a actividades como <actividad>, a participantes como <participante>, un estado <estado>, etc. “Para que todos los modelos de procesos se basen en la misma lista de etiquetas, se debe crear lo que se conoce como una definición de estructura o esquema (XML Schema), que es un fichero con extensión .xsd al que deben hacer referencia todos los documentos XML que representen modelos de procesos basados en las mismas etiquetas”. Cuando se habla de procesos orientados a la educación IMS colabora con bibliotecas de archivos que ayudara en el avance de trabajos publicando en direcciones como <http://www.imsglobal.org/xsd> aquí se encuentran los archivos con extensión .xsd que son públicos para poder así trabajar con el lenguaje XML [9].

“A continuación se muestra un ejemplo de archivo xml que utiliza las etiquetas definidas por IMS para representar los diferentes elementos que deben especificarse al diseñar un proceso de enseñanza-aprendizaje, como los objetivos didácticos del mismo (<learning-objetives>), los participantes (<roles>), las actividades a realizar (<activities>), el método pedagógico a aplicar (<method>), o los objetos didácticos a utilizar (<learning-object>)” [9].

Extracto simplificado de un archivo XML que representa el diseño de un proceso de enseñanza-aprendizaje utilizando las etiquetas definidas por la especificación IMS-LD.

```
<learning-design>
<title>Curso de Introducción a la Informática</title>
<learning-objetives>..</learning-objetives>
<prerequisites>..</prerequisites>
<components>
<roles>
<learner>..</learner>
<staff>.. </staff>
</roles>
<activities>
<support-activity>..</support-activity>
<activity-structure>
<learning-activity>..</learning-activity>
<learning-activity>..</learning-activity>
</activity-structure>
</activities>
<environments>
<environment>
<learning-object>..</learning-object>
</environment>
<environment>
<service><conference>..</conference></service>
<environment>
</environments>
</components>
<method>
<play>
<act>
<role-part>..</role-part>
<complete-act>..<complete-act>
</act>
<complete-play>..</complete-play>
</play>
```

```
<complete-unit-of-learning>..</complete-of-learning>  
</method>  
</learning-design>” [9]
```

2.2.4.4 Diseño y ejecución de procesos de enseñanza-aprendizaje. “La principal utilidad del modelado de procesos, en general, y de procesos de enseñanza-aprendizaje, en particular, es la posibilidad de su procesamiento automático mediante herramientas informáticas desarrolladas al efecto y compartidas por los sujetos implicados en el proceso”, todo esto gracias al arduo trabajo aplicado junto a una buena técnica de modelado en todos los procesos. Al utilizar el lenguaje XML para dar rienda a las técnicas de estructuración y un sentido formal a los procesos se consigue una variedad de herramientas de gestión de procesos aprovechando software de construcción de entornos como lo son .net o java que tienen en sus funciones el manejo del lenguaje XML. Con estas aplicaciones desarrolladas un docente no debe preocuparse por detalles como lo es el manejo de código o conceptos a veces complejos que se manejan en estos estándares en el uso de objetos de aprendizaje, sino por lo verdaderamente importante que es diseñar modelos de estudio, y debido al estándar IMS LD estas herramientas brindan la posibilidad de reutilizar los procesos existentes [9].

2.2.5 IMS SS (Secuencia Simple). Es una especificación de la IMS que está encargada de la definición de un método para la representación del conocimiento que se obtiene cuando se trabaja con objetos de aprendizaje E-Learning de la tecnología LTS (Tecnología del Sistema de Aprendizaje), además de definir los comportamientos requeridos y la funcionalidad que los sistemas deben implementar. [10]

IMS SS dota de reglas que describen el flujo de actividades de aprendizaje que genera la interacción de un estudiante con el contenido de acuerdo a los resultados generados por esta interacción, este flujo resultante se ajusta a la especificación IMS SS de forma manual o mediante programas creados para este fin y se facilita el intercambio de esta secuencia y cuando un contenido se entrega a un alumno se debe hacer la respectiva referencia en esta especificación como un “motor de secuencia”. [10]

El método de trabajo que IMS SS ejecuta acoge solamente el papel del estudiante. [10]

2.2.5.1 Secciones del IMS SS

- **Secuencia de definición de modelos.** El modelo de información en secuencia simple se utiliza para describir el comportamiento de la secuencia deseada.

- **Seguimiento de modelo.** El modelo de información en la IMS SS utilizado para registrar información sobre los resultados de la interacción del alumno con las

actividades, y el registro del alumno de los objetivos (por ejemplo, la terminación, medida) para controlar la selección y secuenciación de las actividades de otros.

- **Actividad modelo de estado.** El modelo de información en la secuencia de simples utilizados para registrar información sobre el estado de las interacciones de un alumno con una actividad y un conjunto de atributos globales para las actividades.

- **Proceso de secuenciación en general** - El proceso general de secuenciamiento sencillo que se relaciona con la selección y la asignación al azar, la navegación, la secuencia de salida, de resumen y los procesos de entrega.

- **Modelo del comportamiento de navegación.** El proceso de secuenciación simple que evalúa una solicitud de navegación y determina la secuencia de peticiones y de terminación que debe ser procesada para identificar y entregar el contenido para el alumno.

- **La terminación de modelo de comportamiento.** El proceso de secuenciación simple que evalúa una solicitud de finalización para poner fin a un intento de una actividad.

- **Acumulativo de modelo de comportamiento.** El proceso de secuenciación simple que calcula los datos de los resultados de una actividad a partir de los datos de los resultados de los hijos de la actividad.

- **Selección y modelo de asignación al azar de comportamiento.** El proceso de secuenciación simple que selecciona y reordena un subconjunto de actividades.

- **La secuenciación de modelo de comportamiento.** El proceso de secuenciación simple que evalúa una solicitud de la secuencia en términos del modelo de contenido describe el árbol de actividades y determina qué objeto de contenido real debe ser entregada al alumno.

- **Comportamiento del modelo de entrega.** Es el proceso de secuenciación simple que valida que los recursos de aprendizaje para la actividad elegida puedan ser entregados.

- **Procesos de servicios públicos.** Secuencia procesos que son utilizados por varios modelos de comportamiento.

- **Extensibilidad.** Los mecanismos de extensibilidad de secuenciación simple.

- **Conformidad.** Requisitos para una implementación para cumplir con la especificación de secuencia simple.

- **Glosario.** Una colección de términos utilizados en el documento conjunto de Secuenciación simple para describir las partes de los modelos de información de la secuencia de conductas específicas a esta descripción de la secuencia simple.

- **Casos de uso.** Una colección de casos de la muestra la secuencia utilizada en el proceso de secuenciación de desarrollo simple para guiar la selección de características y funciones.

Las secciones pueden incluir:

- **Modelo de información.** La descripción detallada del modelo de información correspondiente (el Modelo de Definición de secuenciación, el Modelo de Seguimiento, y el modelo de actividad del Estado). Cada modelo de la información se divide en conjuntos relacionados de forma lógica de los atributos de datos. Cada conjunto se describen los elementos de datos de forma vinculante independiente. A continuación se muestra las secciones del modelo (Tabla 2)

Tabla 2. Secciones del IMS SS

Sección	Mnemotécnico
Introducción	Yo
La secuencia de definición de modelos	SM
Modelo de seguimiento	TM
Actividad modelo de Estado	AM
Proceso general de secuenciación	OP
Modelo del comportamiento de navegación	NB
La terminación de modelo de comportamiento	Tuberculosis
Acumulativo de modelo de comportamiento	RB
Selección y modelo de comportamiento de aleatorización	SR
Comportamiento de la secuencia de modelos	SB
Comportamiento del modelo de entrega	DB
Procesos de servicios públicos	UP
Extensibilidad	E
Conformidad	C
Glosario	G
Casos de Uso	UC

[10]

- **Comportamiento.** La descripción detallada de las conductas y procedimientos que describen las partes de la secuencia simple (para cada uno de los modelos de

comportamiento). Cada comportamiento incluye uno o más descripciones de procesos en pseudo-código.

Esta especificación es muy amplia y para mayor comprensión se puede recurrir al sitio oficial [10].

2.2.6 SCORM (Sharable Content Object Reference Model). Este es un estándar americano de paquetes de objetos de aprendizaje para ser reutilizados y que permite crear objetos pedagógicos estructurados, es decir, mientras un contenido educativo cumpla con la norma **Scorm** estos contenidos se pueden importar, exportar, reutilizar, etc., dentro del sistema de aprendizaje.

2.2.6.1 ¿Que es un contenido de e-learning SCORM? ¿Qué es una plataforma de e-learning SCORM?. “Un contenido SCORM es un contenido que cumple con las especificaciones del modelo SCORM. Un contenido cumple con SCORM si está” [11]:

- Diseñado para ser exhibido en un navegador.
- Descripto por meta-datos
- Organizado como un conjunto estructurado de objetos más pequeños
- Empaquetado de tal manera de que pueda ser importado por cualquier plataforma SCORM compatible.
- Creado para ser portable, de forma que pueda ser distribuido por cualquier servidor web en cualquier sistema operativo

”Una plataforma SCORM, es una plataforma que cumple con las especificaciones del modelo SCORM. Una plataforma cumple SCORM si” [11]:

- Puede aceptar cualquier contenido SCORM y ponerlo disponible para que sean visualizados y recorridos por los usuarios de la plataforma
- Provee de un entorno de ejecución en el cual los contenidos son iniciados y exhibidos mediante un navegador web
- El entorno de ejecución cumple con ciertos requisitos técnicos definidos por SCORM

2.2.7 Situación actual de la composición automática de cursos virtuales.

Opinión de Sergio Gago en un documento que habla de la educación virtual y lo que vendrá: “Actualmente la educación virtual todavía está en vías de desarrollo. Si bien es cierto que se dispone de sistemas muy completos que permiten ejecutar prácticamente todas las fases de la realización de un curso (planificación, búsqueda y acceso de alumnos, realización del curso, evaluaciones, valoración final y revisión de temarios, y áreas de profesores), aún no se llega (ni en un futuro cercano), a conseguir el mismo grado de eficiencia que la educación presencial” [12].

Los intereses constantes de poder llevar la calidad de educación a un nivel superior se han fortalecido con el avance de las TICs (Tecnologías de la Información y la Comunicación) los avances que presentan las plataformas e-Learning han sido de mucho apoyo para que estudiantes interesados aprendan más y a su propio ritmo, mas sin embargo haciendo una reflexión más profunda sobre las ayudas o soluciones existentes que solo se centran en sistematizar lo que en papel se hace y no del todo se le da una revolución a la educación esperada, aunque si existen organismos y procesos de investigación que velan por alcanzar el ideal de individualizar la enseñanza y mejores ambientes educativos [13].

Cabe mencionar que un curso virtual necesariamente está orientado a la web y está conformado por múltiples herramientas que hacen que la gestión de elementos educativos sea más fácil de asimilar.

En la actualidad las exigencias de las personas por aprender y la característica para utilizar la información como principal insumo para el progreso económico y además la Internet como una de las herramientas más importantes existentes hasta hoy brindando a todos la posibilidad de acceder libremente a un sin fin de información que se presta en múltiples formas y formatos, de la mano de toda esta variada información, la educación no puede ser ajena, a pesar que existe una natural resistencia que se da por algunos docentes que prefieren seguir sumidos en las conductas tradicionales de la enseñanza personalizada.

Internet se muestra como un transformador de la metodología de enseñanza, de la forma en que las instituciones brindan la enseñanza, y también un transformador del "mercado" de la enseñanza. [14]

Estas transformaciones serán cada vez más aceleradas a medida que los distintos componentes de la cadena productiva (de la producción de cursos en general) se vayan haciendo accesibles a un espectro más amplio de personas. La accesibilidad está relacionada directamente con la facilidad de manejo brindada por las herramientas. La facilidad de manejo a su vez es influida directamente por el gran desarrollo hecho en el área de las Interfaces gráficas, que permiten el manejo de herramientas complicadas de una manera intuitiva y con una curva de aprendizaje pronunciada.[14]

La composición automática de cursos hoy en día se ve limitada o todavía no se encuentran de manera aglomerada. Se dice que la estos proyectos se encuentran a nivel mundial en etapas de investigación y desarrollo, además el docente promedio no se encuentra totalmente preparado para aprovechar toda la potencia brindada por Internet, también existen empresas que tratan de dar soluciones parciales para resolver algunos problemas específicos en lo que respecta al ámbito educativo.

Debido a que el desarrollo de herramientas para el apoyo a la educación y el aprendizaje se encuentran en su mayoría en etapas de investigación y desarrollo las aplicaciones existentes son escasas y de muy alto costo lo que hace más rentable seguir en la construcción de herramientas propias y de más alta calidad.

2.2.7.1 Características de los cursos

Adaptación en cursos virtuales. “La adaptación abarca los procesos de adaptar las características de un sistema para alcanzar alguna meta. Son posibles dos enfoques: La adaptabilidad, un proceso automático de adaptación, es decir controlado por el sistema. Por otro lado, la adaptabilidad que significa que el usuario adapte el sistema a su gusto o sea, un proceso manejado por el usuario” [13].

La adaptabilidad es una característica especial de los cursos virtuales automáticos que consiste en la capacidad que posee un software educativo o sistema a moldearse a los requerimientos de un estudiante permitiéndole a este interactuar cómodamente entre sus contenidos.

Para que un curso sea adaptable es importante considerar los requerimientos del estudiante o el modelo de estudiante, es decir a quien va orientado el curso y los contenidos o propiedades del curso para que estos sean manipulados y hasta modificados por los estudiantes, también diseñar métodos que relacionen estos requerimientos [13].

2.2.7.2 Tipos de cursos. Existen dos tipos de cursos virtuales: el aprendizaje a distancia sincrónica, es decir casi en tiempo real como si se estuviera en un salón de clase pero obviamente desde otro lugar, pero en un tiempo determinado, y el aprendizaje a distancia asíncrona, es decir se cumple con una temática pero o necesariamente en horas predefinidas sin embargo los tiempos no son de libre albedrío si no que están también definidos.

2.2.8 Plataformas LCMS (Learning Management System). Antes de definir que es una plataforma LCMS se debe definir lo que es una plataforma e-Learning “Dado un escenario educativos de eLearning, la plataforma para eLearning es la herramienta tecnológica basada en Internet que se utiliza para distribuir el conocimiento” [15].

Además, se debe definir que es un sistema CMS y un sistema LMS: un CMS (Content Management System) es un “Software para crear, editar, gestionar y publicar contenido digital en diversos formatos. Interactúa con el servidor para generar dinámicamente la página web bajo petición del usuario, con el formato predefinido y el contenido extraído de la base de datos del servidor” y un LMS (Learning Management System) es un “Software instalado en un servidor que se

emplea para administrar, distribuir y controlar las actividades de formación no presencial o de eLearning de una institución u organización” [15].

Con estos conceptos más claros es fácil entender el concepto de una plataforma LCMS, que en pocas palabras es la unión de estas dos plataformas anteriormente definidas, es decir una plataforma LCMS es también un software que combina las funcionalidades de las CMS y de las LMS, lo que resulta una gran herramienta con mucha más complejidad que además de crear contenidos trabaja para la distribución de este, esta unión hará que personas dedicadas al manejo de elementos educativos puedan con más facilidad crear, diseñar y distribuir procesos de aprendizaje [15] [16].

2.2.8.1 Características de la plataforma LCMS. Las LCMS se caracteriza por su complejidad además respeta a estándares como IMS SCORM y se consideran de alto costo debido a operaciones o funciones básicas como:

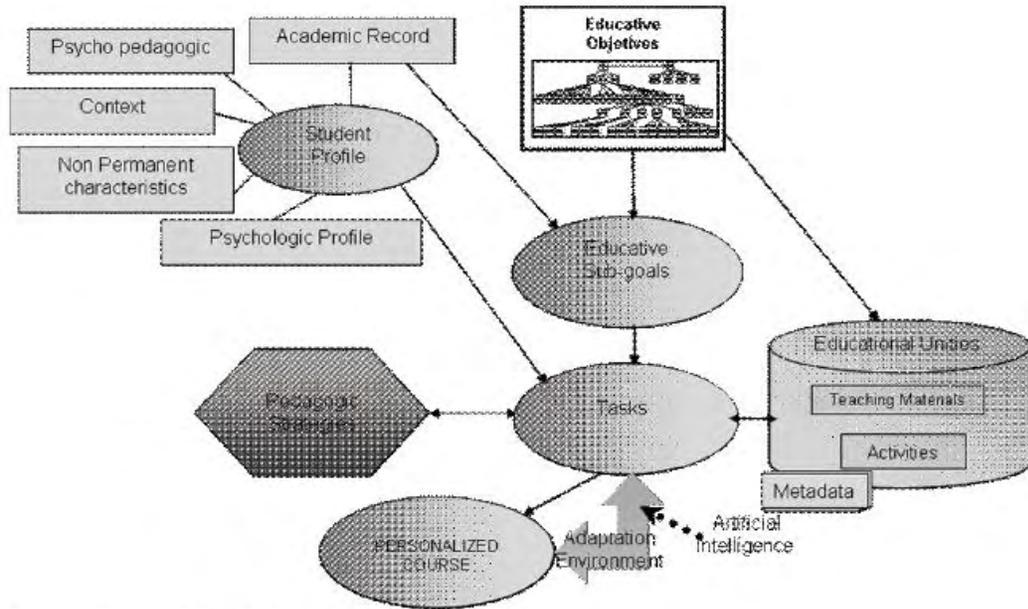
- Autenticación al sistema.
- Generación de contenidos
- Visualización de contenidos
- Realización de actividades como tareas, trabajos en grupo
- Reporte de las actividades realizadas por el alumno, herramientas evaluación, etc.

Adicionalmente las plataformas LCMS gestionan contenidos mediante las siguientes características [16]:

- Herramientas para la generación y uso de contenidos.
- Herramientas encargadas de la distribución de contenidos de manera inteligente, de tal manera que los usuarios se identifique con cada curso.
- Objetos reutilizables.
- Herramientas que gestionen la interacción del usuario con el sistema.
- Herramientas para la evaluación de estudiantes.
- Compatibilidad con otro tipo de plataformas y interoperabilidad para la interacción de estándares como IMS, además de posibilidades de exportación e importación de objetos de manera sencilla, etc.

2.2.8.2 Composición automática de cursos. Cuando se habla de construir un curso que se adapte a un estudiante se obliga a utilizar algún tipo de estrategia, a continuación un modelo (SICAD) que plantea una estrategia que se adaptada a este, y una característica relevante de este modelo es tal vez su orientación a los Objetivos Educativos (OE), estos serian las metas que un alumno debería alcanzar cuando haya finalizado el curso [13]. La figura 9 muestra un modelo de curso.

Figura 9. Modelo del curso adaptativo



[13]

Las metas antes planteadas, fijadas según el estudiante de acuerdo a sus aptitudes, previos conocimientos y capacidad académica pueden ser abordadas por diferente material educativo como también tareas respectivas que el modelo soporta [13].

Para la comprensión más clara de este modelo se puede recurrir a las Págs. 3, 4, 5 y 6 del artículo referenciado en [13].

Cabe resaltar que este modelo presenta ventajas debido a la estrategia que presenta, cuando se va a automatizar un curso los aspectos más importantes se derivan del uso de metadatos y estándares que facilitan el trabajo de compartir muchos recursos, además este sistema no se restringe a un solo estilo de aprendizaje lo que lo hace más complejo y conveniente [13].

Lo anterior arroja la conformación de un curso personalizado complejo que al realizarlo de forma manual sería bastante difícil debido a esto una opción válida es utilizar técnicas de planificación inteligente (AI Planning) para poder automatizar este proceso, una parte importante en todo el proceso de adaptación de este curso radica en la determinación de los parámetros iniciales que el estudiante posee, es decir, las características propias y como estas junto con componentes importantes serán adaptadas al curso automático, como estas características y componentes se presentan de forma extensa se necesitan técnicas poderosas para especificar la búsqueda y fijar un dominio exacto para ello se utiliza la técnica antes mencionada, que se describe con más puntualidad en [13].

Cabe resaltar puntos importantes a la hora de automatizar un curso, como lo son las interfaces que el sistema tiene con el estudiante, el plan del curso, las estrategias educativas, el filtrado de información y el proceso evaluativo, integrando actividades individuales con ambientes colaborativos que son también partes importantes a la hora de adaptarlas en el proceso de generación del curso [17].

2.2.8.3 Utilización de repositorios de aprendizaje. “Un repositorio digital es un archivo local, institucional o centralizado para depositar y proveer acceso a contenidos digitales” [18], además estos repositorios deben permitir el almacenamiento, recopilación y administración (buscar, consultar...) de los objetos de aprendizaje.

La existencia de objetos de aprendizaje hacen necesaria la creación de estos repositorios así se pueden almacenar y gestionar de manera práctica, teniendo en cuenta estas definiciones, un repositorio de objetos de aprendizaje también es un sistema y como tal un software que tiene como fin la gestión de metadatos para que otros software o usuarios tengan acceso a estos por medio de interfaces específicas o correspondientes [2]. La tabla siguiente muestra algunos repositorios.

Tabla 3. Ejemplos de Repositorios.

Nombre del Repositorio	País
MOREA (http://www.usc.es/morea/) Múltiples Objetos Reutilizables para Enseñanza y Aprendizaje	España
Biblioteca Semántica de WebQuest (http://cfievalladolid2.net/webquest/common/index.php)	España
MERLOT : Multimedia Educational Resource for Learning and Online Teaching (http://www.merlot.org/merlot/index.htm)	International
CAREO : Learning Commons, University of Calgary http://careo.ucalgary.ca/cgi-bin/WebObjects/CAREO.woa	Canada
LoLa : Learning Objects, Learning Activities (http://www.lolaexchange.org/)	USA

[2]

De acuerdo con las definiciones anteriores otros ejemplos de repositorios son:

Dspace: Es una aplicación Web desarrollada por el MIT y HP, se utiliza como repositorio de objetos de aprendizaje para lo cual fue creada, se construyo bajo Java y provee Java API, su sistema utiliza archivos del sistema, su estructura se compone de un modelo jerárquico (comunidades, colecciones y subcomunidades, ítems, paquetes de archivos, archivos). [18]

Fedora: Es un software de código abierto basado en la Web, su tarea es gestionar contenido digital también es utilizado para crear aplicaciones como repositorios institucionales, también fue desarrollado en java y funciona en la Web. Posee muchas herramientas para su fácil utilización, además consta de tres tipos de objetos: los datos, definición de comportamiento y mecanismo de comportamiento. [18]

Digitalcommons: Desarrollado bajo tecnología ASP tiene inconvenientes para ser modificado e integrado con otros sistemas, además no posee documentación completa de algunos de sus componentes. [18]

Otros ejemplos de repositorios son: ECL, EPRINTS

2.3 COMPETENCIAS

2.3.1 Definición. “En lo que se refiere a la definición y estandarización de competencias existen dos esfuerzos principales. Competency Definitions de LTSC es un trabajo en curso cuyo objetivo es definir un modelo de datos para las definiciones de competencia, creando identificadores únicos para los llamados “objetivos didácticos” que faciliten la referencia no ambigua desde los recursos didácticos. Un trabajo similar es IMS-RDCEO (Reusable Definition of Competency or Educational Objective) que persigue la definición de competencias que aparecen como prerrequisitos para la realización de experiencias educativas o como objetivo didáctico de las mismas” [19].

Una definición de competencia dada por Néstor D, Duque M, etc. “La competencia se refiere a la formación o preparación de un individuo para intervenir de un modo eficaz en un proceso o un contexto. En la competencia se incluye tanto la actuación, como los conocimientos como los valores de los individuos. Se diferencia del concepto de capacidad, en que además de incluirse la noción de poder en el de la competencia, el énfasis está en la adquisición por medio del aprendizaje estructurado, del poder para intervenir de una manera que sea observable. Por lo tanto la competencia no solo es la adquisición de un poder específico sino que es algo que se sitúa en el ámbito de lo demostrable” [13].

2.3.2 Estándares y especificaciones. “Frecuentemente los términos estándar y especificación se utilizan indistintamente, no obstante, es importante puntualizar su diferencia. Si una tecnología, formato o método ha sido ratificado por algún organismo oficial de estandarización, se trata de un estándar. Pero si una tecnología, formato o método propuesto no ha sido aprobado por algún organismo oficial de estandarización, se trata de una especificación. Aunque, en algunos casos, una especificación puede considerarse un estándar *de facto* si su uso es extendido y entretanto se ratifica como estándar” [20].

2.3.2.1 Problemática. Hoy en día gracias a las ventajas que brinda el avance de la computación se desarrollan herramientas que gestionan la organización y generación de cursos, es impresionante el número de estas herramientas que se encuentran en el mercado y de la mano la infinidad de plataformas e-Learning que han sido fabricadas tanto por particulares como empresas y hasta las universidades que necesitan gestionar cursos para ofrecerlos a la amplia y variada demanda de estudiantes [21].

Los LMS han sido impulsados por empresas que ven en estos sistemas una gran visión comercial dando mejoras a estos sistemas tanto en apoyo como complementos a los servicios que ofrecen estos LMS. Las plataformas e-Learning no son sistemas independientes estos se pueden apoyar con herramientas de terceros o por integraciones fabricadas por las propias instituciones, pero estas se ven limitadas debido a la especificidad de sus modelos y la improbabilidad de fabricación de herramientas o avances para su institución, y depende únicamente de ellas mismas el avance en sus mejoras [21].

Por lo anterior las herramientas son muy diversas cuando de autonomía se trata y en la actualidad cada LMS ha creado y lo seguirá haciendo sus propios formatos de estructuración de contenidos lo que en realidad hace imposible la interoperabilidad [21].

2.3.2.2 Objetivo. El objetivo cuando se establecen estándares para definir metadatos educativos es buscar el punto en el cual las características que un elemento debe tener para permitir que un ambiente e-learning cuente con las siguientes habilidades [22]:

- Accesibilidad para localizar y acceder materiales instruccionales independientemente de su localización.
- Adaptabilidad para ajustar la instrucción a las necesidades individuales de los estudiantes.
- Asequibilidad para aumentar la eficiencia y productividad disminuyendo el tiempo y costos del proceso.
- Durabilidad para resistir los cambios tecnológicos sin la necesidad de rediseñar, recodificar o reconfigurar.
- Gestionabilidad para monitorear información sobre el estudiante y los contenidos de aprendizaje.
- Reusabilidad para integrar los componentes instruccionales en una variedad de aplicaciones, sistemas y contextos.

2.3.2.3 ¿Cómo influyen los estándares en el aprendizaje? Una problemática que también se recalca es que los estándares, al no referirse a los valores y solo se enfocan en las materias para las cuales estos han sido diseñados y se dejan de lado aquellas materias a las cuales los estándares no aplican y si en cuestión una materia bien formulada y orientada arroje resultados diferentes en distintos puntos

de una localidad o al contrario que todos alcancen de manera fácil un curso, dará como resultado la baja en la calidad del promedio de la educación en algunos sitios. “Por otra parte, se adoptan estándares deseables o de excelencia, los sectores que no los logren podrían frustrarse y favorecer la repitencia y el abandono” [13].

2.4 CURRÍCULO

2.4.1 Currículo basado en competencias. “Un currículo basado en competencias, parte de la premisa que son éstas las que orientan el proceso de enseñanza y son los contenidos los que se deberán movilizar para su adquisición” [23].

Para poder mejorar la calidad de educación se debe enfatizar en la mejora de los currículos que se aplican, ya que los estudiantes no alcanzan el máximo de un tema si los profesores o no saben de pedagogía pero saben de su cátedra o viceversa, y eso es lo que a menudo siempre sucede.

En alusión al tema de pedagogía es común escuchar o se asocia mucho el concepto de un currículo basado en competencias y este en la actualidad ha tomado cada vez más fuerza cuando de educación se trata.

La importancia de formar currículos que se basen en competencias que el mundo ofrece y que en esta actualidad han cambiado de manera acelerada gracias a todos los avances que la sociedad implementa no puede ser baja ya que se deben formar personas aptas para combatir las tendencias de este mundo cambiante, tendencias tales como: Avance tecnológico en la informática, las telecomunicaciones etc., también el alto desempleo que existe debido a la incompatibilidad de destrezas no muy bien desarrolladas de los profesionales para con sus actividades laborales, el consumo masivo de países desarrollados, etc.

“Es necesario tomar nota del cambio entre un currículo tradicional y uno basado en competencias. En tanto el primero está centrado en el contenido, el segundo en los indicadores de desempeño. Para el primero los tiempos de enseñanza son fijos y para el segundo son variables ya que se tiene en cuenta el ritmo personal de aprendizaje, es decir, centrado en las necesidades individuales” (Amadeo Sosa) [23].

El currículo basado en competencias se caracterizado por ser una tendencia que la sociedad debe asimilar y seguir, “porque aporta una visión nueva e integradora de la educación , a la vez que proporciona un léxico y una lógica que permiten sistematizar prácticas ya vigentes en muchos ámbitos del medio educativo” [24].

Gracias a la importancia que se le ha dado a las competencias se puede afirmar que la formulación del currículo basado en competencias se está haciendo cada

vez más común para los diferentes niveles de educación y en distintas partes del mundo.

2.4.1.1 Características de los nuevos modelos curriculares. Como se menciona anteriormente los currículos basados en competencias se diferencian a los currículos tradicionales de diferentes maneras una de ellas son sus características [25]:

- Pertinencia
- Calidad
- Flexibilidad
- Internacionalización
- Interdisciplinariedad y otras modalidades de organización del conocimiento
- Polivalencia
- Integralidad
- Vinculación con diversos sectores
- Innovación
- Aplicabilidad y transferibilidad
- Énfasis en valores
- Movilidad
- Centrado en el estudiante

2.4.1.2 Estudios pedagógicos – modo de hacer un currículo. A partir de la propuesta estructural de un currículo por competencias para la formación de profesionales presentada en un seminario internacional patrocinado por la Universidad del Norte (Colombia) y CINDA (Chile), y de la socialización de la misma en la Universidad Politécnica de Nicaragua y las Universidades Católica y Javeriana de Bogotá, se presenta este trabajo, en el que se cualifica la estructura y los procesos funcionales de un currículo basado en competencias bajo la concepción de una formación integral y adecuada a las circunstancias determinadas como los “retos de la modernidad” [19].

2.4.1.3 Las bases de un currículo basado en las competencias. Para la construcción de un currículo basado en competencias es estratégico el documento de Alonso Tejada Zabaleta orienta en este camino basando esta construcción en una estructura que posibilita a la mejor formación de un profesional basándose en las siguientes bases o pilares como él los llama [19]:

- Las Bases Ontológicas y Epistemológicas
- Las Bases Conceptuales y Teóricas
- Las Bases Metodológicas
- Las Bases Tecnológicas y Aplicadas o de Acción

Las bases ontológicas y epistemológicas. “La propuesta se basa en las bases epistemológicas y ontológicas que sostienen que el concepto de competencias

debe ser definido como un complejo total e integral, en contraposición a las miradas dualistas, reduccionistas y fragmentarias que aun son dominantes en los estudios sobre competencias”. Entonces es importante fomentar las siguientes proposiciones según Alonso Tejada Zabaleta. :

- “Confrontar el paradigma dominante dualista, mecanicista, reduccionista y fragmentario que se debate entre dos polos: o desde las competencias reducidas a lo cognitivo, o la reducción al concepto de competencias laborales” [19].
- “Proponer, legitimar y validar una concepción sobre las competencias como una dimensión total, integral, compleja, procesal e interaccional; significadas en las competencias para la vida” [19].

Las bases conceptuales y teóricas. Se define un complemento entre el conocimiento, el aprendizaje y las competencias [19].

Las bases metodológicas. “Desde el punto de vista metodológico se propone un diseño que posibilite la interacción sistemática, contextual y significativa de las Dimensiones del Conocimiento con los procesos del aprendizaje y con la progresión de las competencias claves en la formación de un profesional. Esta integración y sistematización da valor a una interacción permanente entre la teoría y la práctica a través de la investigación y a una relación coherente y consistente entre la Educación o Formación y el medio productivo y del trabajo” [19].

Las bases tecnológicas y aplicadas o de acción. “Desde el punto de vista tecnológico y técnico-aplicado, la propuesta las integra al marco total del diseño curricular por competencias. Así” [19]:

- “Se explicita la relacionabilidad entre la utilización de instrumentos y el desarrollo de estrategias de intervención profesional a través de un proceso explícito e intencionado de relacionar la teoría y la práctica en una interacción sustentada en la investigación y en una condición de proceso de complejidad creciente. Esto se fortalece en el inter-juego de la acción y la reflexión como proceso continuo” (Alonso Tejada Zabaleta).[19]
- “Se valida la evaluación de las competencias como productos totales, que impliquen la construcción individual y colectiva y no como fragmentos. Metodologías como la de Problema>Proyecto>Producto a través de Portafolios, o las de estudios de casos, o las de las Prácticas Supervisadas y Acompañadas o los Trabajos de Grado, son ejemplos de productos terminales que pueden especificar el logro de competencias” (Alonso Tejada Zabaleta). [19]

3. METODOLOGIA

Para lograr cumplir con el desarrollo de la aplicación se siguió la siguiente metodología:

Modelo en cascada

En la primera etapa de este proyecto se realizó el análisis y diseño del Sistema, para lo cual se aplicó el Lenguaje Unificado de Modelado (UML), junto con el diseño de la base de datos.

La segunda parte correspondió a la etapa de desarrollo del sistema, la cual esta soportada por tecnologías Cliente Servidor (Java, MySQL, Ontologias, web semántica) incluyendo librerías (Jena) para el manejo de ontologías, que hacen de este Software un sistema planificado y ejecutado en entorno Web.

4. DESARROLLO DEL PROYECTO

En este apartado se lleva a cabo el desarrollo para lo cual se identificaron requerimientos y la definición de una ontología, que con base en competencias, permita identificar automáticamente el currículo que un estudiante cursaría teniendo en cuenta una asignatura y el estudio de las temáticas anteriores, las cuales sirven de base para plantear el modelo de este prototipo.

Para llevar a cabo este modelo se tuvo en cuenta características de los LCMS, el diseño de cursos basado en competencias y una ontología para la gestión de la mayoría de los datos y su previo razonamiento entre ellos.

4.1 IDENTIFICACIÓN DEL ESPACIO ACADÉMICO

El currículo es el conjunto de competencias básicas, objetivos, contenidos, criterios metodológicos y de evaluación que los estudiantes deben alcanzar en un determinado nivel educativo. Responde a las preguntas ¿qué enseñar?, ¿cómo enseñar?, ¿cuándo enseñar? y ¿qué, cómo y cuándo evaluar? el currículo, en el sentido educativo, es el diseño que permite planificar las actividades académicas.

El área de conocimiento que fue base y para el cual se implementó el modelo, fue la asignatura Estructura de Datos perteneciente a la disciplina de la Computación vigente hoy en día propuesta por el programa de Ingeniería de Sistemas de la Universidad de Nariño.

Tabla 4. Identificación del espacio académico.

1.1 CICLO	BASICO
1.2 ÁREA	APLICACIÓN PROFESIONAL
1.3 COMPONENTE:	PROGRAMACION
1.4 ESPACIO ACADÉMICO:	ESTRUCTURAS DE INFORMACION
1.5 N° DE CRÉDITOS:	3
1.6 SEMESTRE:	5º
1.7 PRERREQUISITO:	TALLER DE PROGRAMACION III
1.8 NÚCLEO PROBLÉMICO DEL CICLO	¿Cómo potenciar las capacidades de los estudiantes de ingeniería de sistemas en investigación, en la fundamentación científica, lógica y programación para desarrollo del pensamiento ingenieril?
1.9 EJE	EL SER HUMANO Y SU FUNDAMENTACION CIENTIFICA, INVESTIGATIVA Y PROGRAMACIÓN.

4.2 DESCRIPCIÓN DEL ESPACIO ACADÉMICO

La administración de la información exige cada vez más, procesos dinámicos que permitan manipular datos de manera precisa, lógica y rápida. El manejo de estructuras de información permite el desarrollo de una estructura mental que ayuda a plantear propuestas adecuadas a diferentes problemas reales que exigen soluciones informáticas, y que pueden ser implementadas en diferentes lenguajes de programación.

La asignatura de Estructuras de Información, permite la comprensión de la forma como se almacenan los datos en memoria y por ende permite desarrollar lógicas adecuadas para lograr la optimización de su uso. La información puede ser almacenada de diferentes maneras y aunque las estructuras estáticas son indispensables, estas presentan algunas limitaciones que serán salvadas con la utilización de estructuras dinámicas que amplían la posibilidad de la administración de datos. Esta será fundamentalmente la base que guía el contenido de la asignatura; para lograrlo se utilizará como herramienta de trabajo el lenguaje C, que tiene las características necesarias para el desarrollo de programas que utilicen diferentes tipos de estructuras de datos.

Estructuras de Información. Las estructuras de información permitirán la construcción de una estructura mental y lógica que ayuda a ver más claramente cómo desarrollar problemas que se le presenten en el mundo de la programación además de formar al estudiante de ingeniería en la fundamentación en el análisis y diseño de alternativas de solución para los sistemas de información en una organización

4.3 IDENTIFICACIÓN DE COMPETENCIAS

Una vez definida la disciplina sobre la cual se desarrollaría el modelo, se identificó las competencias necesarias que garantizan el conocimiento y el manejo de la disciplina, para lo cual se tuvo en cuenta la siguiente base:

4.3.1 Competencia central. Analizar y comprender los tipos de estructuras dinámicas de datos pilas, colas y listas para implementarlas en la solución de problemas específicos mediante un lenguaje de programación.

4.3.1 Competencias específicas. Como se nombran a continuación:

- Identificará las características de las estructuras dinámicas pilas, colas y listas.

- Desarrollará habilidades de administración de datos utilizando estructuras dinámicas.
- Diseñará estructuras de información adecuadas según el problema a resolver.
- Desarrollará aplicaciones en el lenguaje C, utilizando estructuras dinámicas

4.4 IDENTIFICACIÓN DE LAS HERRAMIENTAS DE DESARROLLO

Para implementar el prototipo, primero se debía analizar algunas de las herramientas, e identificar cual tenía las características necesarias con base en la información y el conocimiento adquiridos en las fases anteriores del proyecto.

La herramientas seleccionadas fueron: Protege versión “Protege_4.0.2” para el diseño y construcción de la ontología, NetBeans IDE 7.0 para la creación y programación del prototipo utilizando como Servidor web “Apache Tomcat 7.0.11”, mysql-connector-java-5.1.17: para la conexión con la base de datos para manejar el login.

Documentos que se gestiona. Un archivo de extensión “.owl” que es generado por Protege al mismo tiempo que es gestionado desde la plataforma Netbeans

Librerías utilizadas. “Jena-2-6-4” para gestionar la ontología desde el entorno Netbeans (archivo “.owl”)

Protege. Consola como se ve en la figura 10 (siempre permanece abierta mientras uno abra una instancia de protege).

Figura 10. Instalación Protege

```

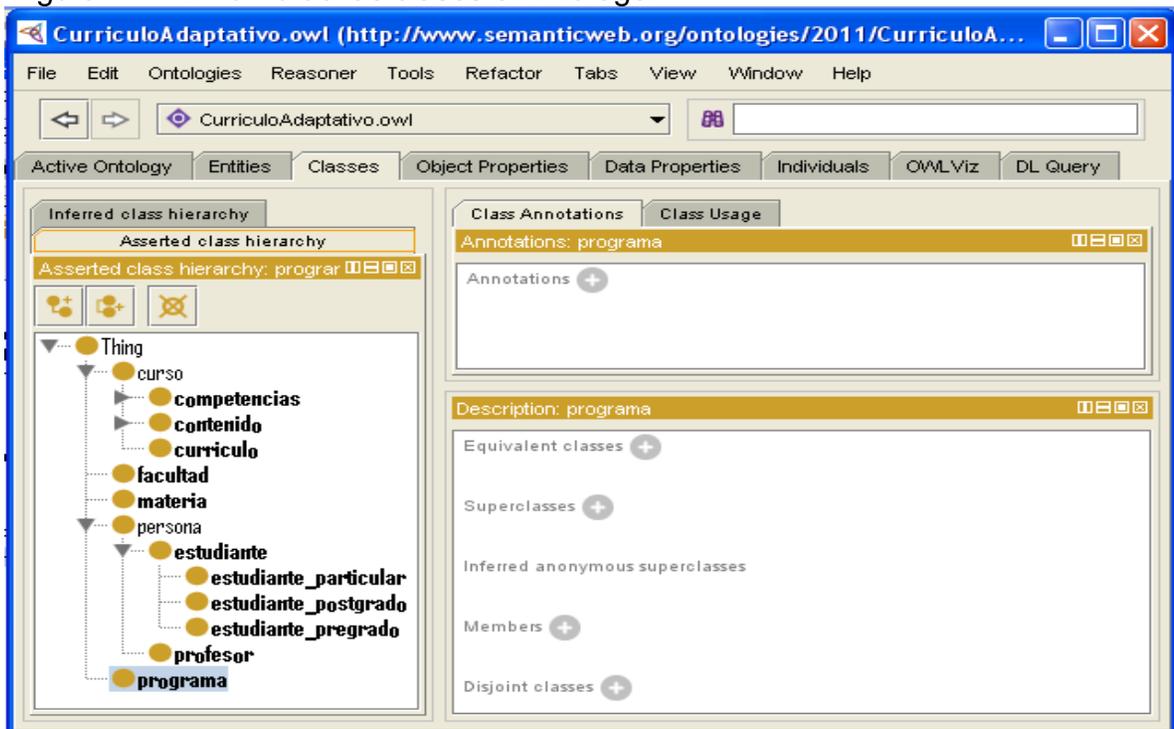
Protege.exe
... rebuilt in 16 ms
... active ontology changed
Saving CurriculoAdaptativo.owl (http://www.semanticweb.org/ontologies/2011/CurriculoAdaptativo.owl) to temp file: C:\DOCUME~1\EDIER1\CONFIG~1\Temp\CurriculoAdaptativo1799019415735919613.owl
Saved CurriculoAdaptativo.owl (http://www.semanticweb.org/ontologies/2011/CurriculoAdaptativo.owl) to file:/C:/CurriculoAdaptativo.owl
Saving CurriculoAdaptativo.owl (http://www.semanticweb.org/ontologies/2011/CurriculoAdaptativo.owl) to temp file: C:\DOCUME~1\EDIER1\CONFIG~1\Temp\CurriculoAdaptativo730496499806680988.owl
Saved CurriculoAdaptativo.owl (http://www.semanticweb.org/ontologies/2011/CurriculoAdaptativo.owl) to file:/C:/CurriculoAdaptativo.owl
No mapping for file:/C:/CurriculoAdaptativo.owl found.
loading file:/C:/CurriculoAdaptativo.owl from file:/C:/CurriculoAdaptativo.owl
Setting active ontology to http://www.semanticweb.org/ontologies/2011/CurriculoAdaptativo.owl
Rebuilding entity indices...
... rebuilt in 0 ms
... active ontology changed
Saving CurriculoAdaptativo.owl (http://www.semanticweb.org/ontologies/2011/CurriculoAdaptativo.owl) to temp file: C:\DOCUME~1\EDIER1\CONFIG~1\Temp\CurriculoAdaptativo8758782960469298253.owl
Saved CurriculoAdaptativo.owl (http://www.semanticweb.org/ontologies/2011/CurriculoAdaptativo.owl) to file:/C:/CurriculoAdaptativo.owl

```

Clase. En protege se puede decir que las clases son un grupo de elementos que comparten propiedades comunes debido a esto siempre se encuentran enlazadas entre si, además comparten un dominio que se define dependiendo el uso que se le da a la ontología, entonces no existe una única jerarquía para las clases ni tampoco que este perfectamente definida.

A continuación el entorno para la generación de las clases (Figura 11):

Figura 11. Entorno de las clases en Protege.



Dominio. En las ontologías el dominio marca un papel fundamental ya sea para la ontología en general, que sería todo lo que esta abarcaría o el dominio de la ontología que abarca a un estudiante(es) un docente (es) y la generación de un currículo por eso los nombres de las clases se relacionan al ambiente estudiantil de estos elementos.

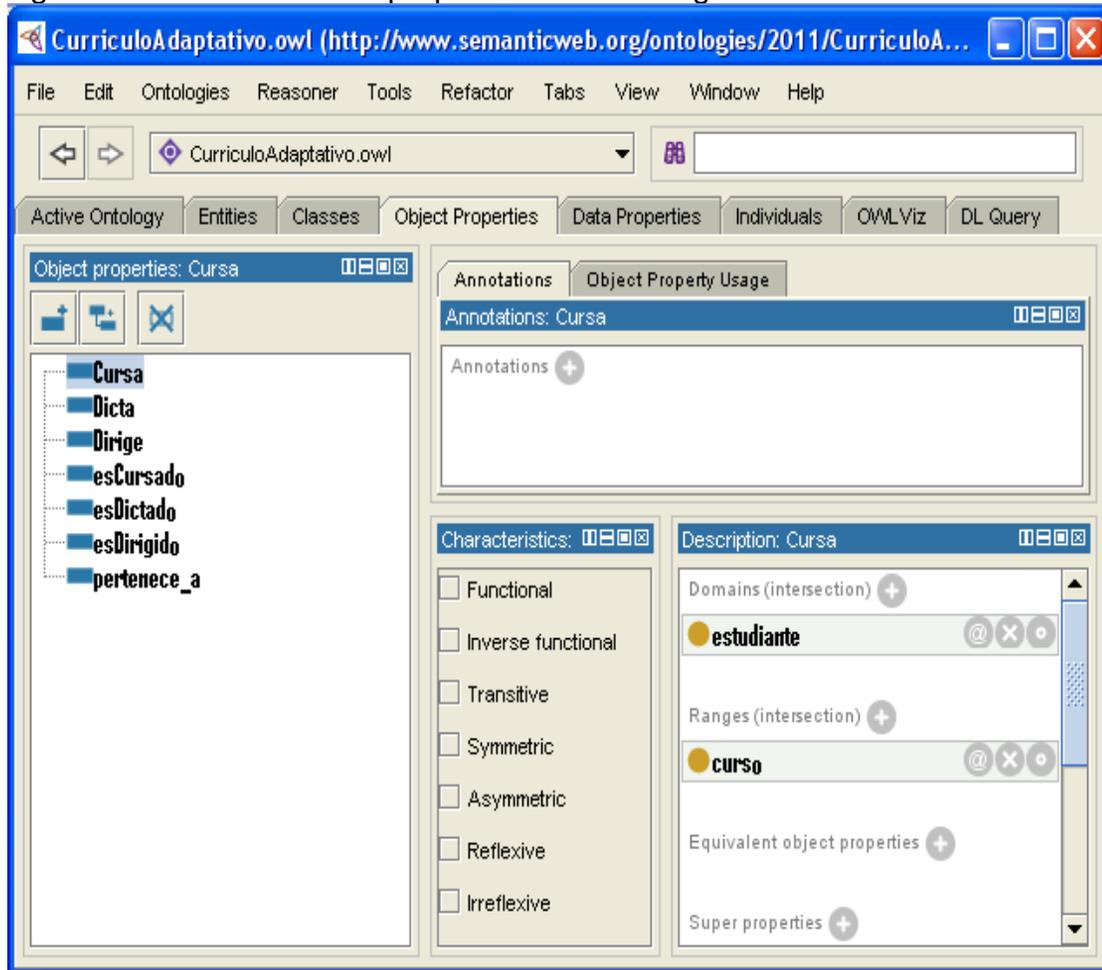
Como se puede observar en la imagen la ontología maneja clases como “curso”, “persona”, como globales y de ellas se desprenden las subclasses que a medida que se va desarrollando la aplicación o el prototipo van tomando su lugar.

Object properties o rdf property. Estas son las que hacen que una ontología se diferencie de un simple documento al cual se le adiciona cualquier tipo de información, porque son aquellas que relacionan individuos (clases) entre sí

además con los demás elementos que existan en la ontología como por ejemplo los datos y los valores de los datos.

A continuación el entorno para la generación de las Propiedades (Figura 12):

Figura 12. Entorno de las propiedades en Protege.

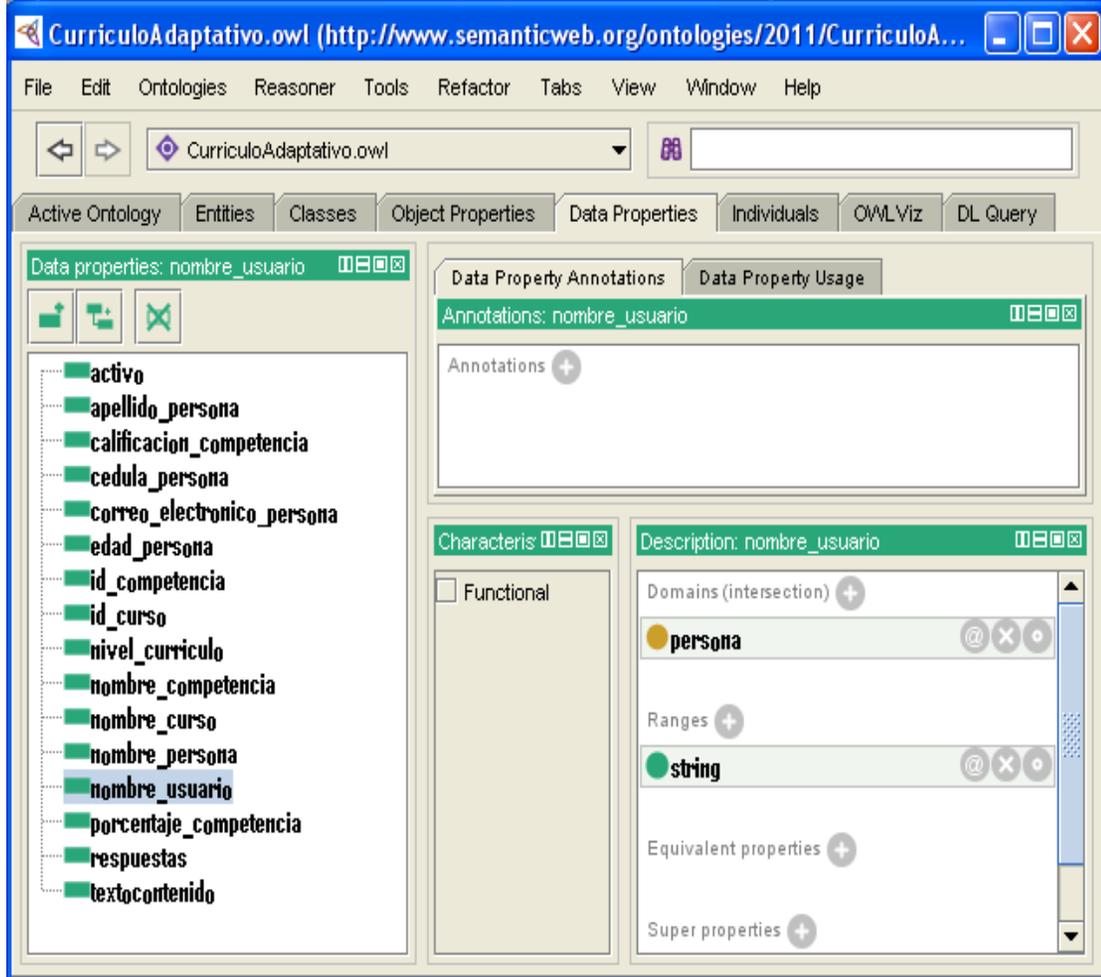


Ejemplo Property “CURSA” que relaciona a la clase estudiante con la clase curso, el sentido que tiene es fácil de comprender “Estudiante” cursa “Curso”, más adelante se verá que el uso de esta simple relación será de mucha utilidad.

Data properties. Como en cualquier aplicación son las definiciones de las variables que se utilizarán para gestionar datos en la ontología se definen para esta, un dominio y un rango de aplicación.

A continuación el entorno para la generación de las Data Propiedades (Figura 13):

Figura 13. Entorno de las data propiedades en Protege.

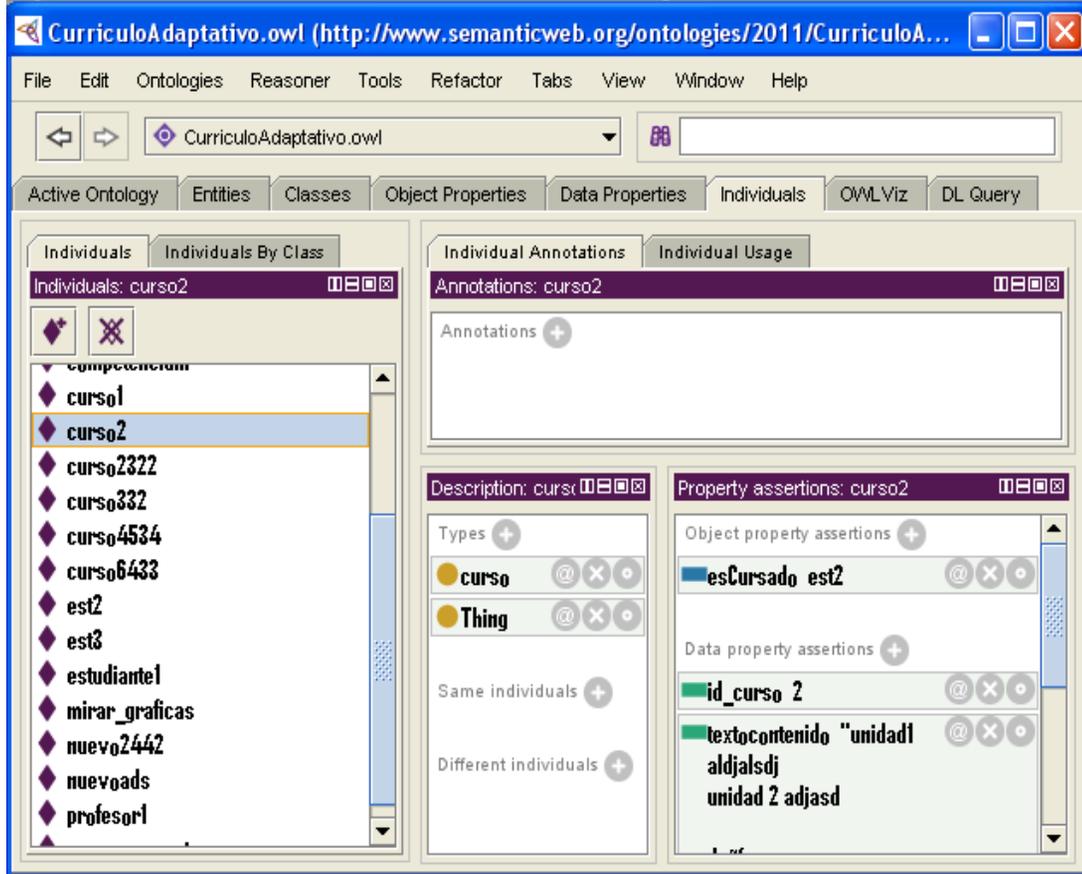


Ejemplo. En la imagen se muestra la Data Propertie “nombre_usuario” el cual lleva como dominio las personas esta Data Propertie también acogería a las subclases de persona que serian para el uso de las clases “Estudiantes” y “Profesores” además de el rango que en este caso es de tipo String.

Individuals. Estos son todas las instancias que pueden generar de las clases, aquí se evidencian las relaciones que tienen los individuos entre sí, las Data propiedades definen el tipo de dato que una instancia de un individuo posee.

A continuación el entorno para la generación de los Individuos (Figura 14):

Figura 14. Entorno de los individuos en Protege.



Ejemplo. En la imagen se observa como un individuo “curso2” es de type “curso” (clase) y este a su vez muestra que es cursado por un solo Estudiante “est2” y sus Data propiedades son básicas como lo puede ser la identificación de este curso entre los demás cursos como lo es su “id_curso”.

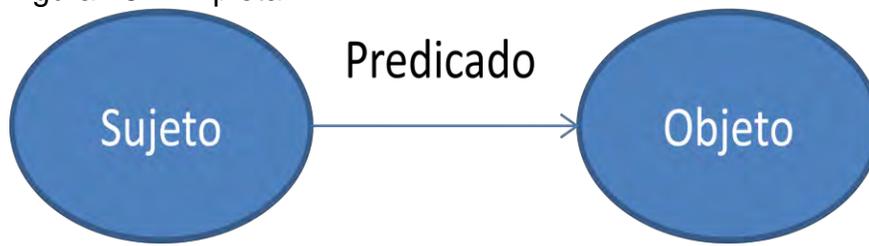
Protege además de ayudar a construir la ontología, proporciona información de referencia sobre el soporte para los datos manejados en ayuda para las tecnologías semánticas, incluido el almacenamiento, la inferencia, y la capacidad de consultar si los datos son adaptables a la hora de trabajar con RDF Schema (RDFS) y Web Ontology Language (OWL). Permite almacenar datos semánticos y puede ampliar la potencia de las consultas sobre los datos semánticos haciendo deducciones lógicas mediante complementos basados en la ontología. Los datos semánticos se gestionan de manera más eficaz a través del “Jena” mediante Netbeans.

Además de su semántica formal, la semántica de datos tiene una estructura simple que se modela con un grafo dirigido. Las declaraciones de metadatos se representan como tripletas: en donde los nodos se utilizan para representar a dos partes de la tripleta, y la tercera parte está representada por un enlace dirigido que

describe la relación entre los nodos. Las tripletas se almacenan en la ontología. Además, la información se mantiene sobre determinados modelos de datos semánticos creados por la aplicación web y la interacción de los usuarios finales. Un modelo creado tiene un nombre identificado por un URI, y se refiere a tripletas almacenadas en el archivo “.owl” generado por la ontología.

Las declaraciones se expresan en tripletas: tema o un recurso, predicado o propiedad, objeto o valor. En este documento sujeto, predicado, objeto se utilizan para describir una tripleta, y los términos de la declaración se pueden usar indistintamente. Los sujetos y objetos de cada tripleta se asignan en nodos, y los predicados se asignan a los enlaces de red que tiene su nodo inicial (sujeto) y nodo final (objeto). Cada tripleta es un hecho completo y exclusivo de un dominio específico, y puede ser representado por un enlace en un grafo dirigido. Como en la figura 15.

Figura 15. Tripletas.



4.5 CONSTRUCCIÓN DE LA ONTOLOGÍA

Teniendo las características de la herramientas descritas anteriormente en especial **Protege** a la hora de trabajar con tripletas, se analizó cada una de las competencias definidas en el curso, además y como parte fundamental se deben tener en cuenta el glosario de términos que arroja el contexto de un estudiante al momento de realizar la búsqueda, inscripción y matrícula de un curso, con el propósito de identificar palabras que se encuentren varias veces y de esta manera poder asociar las información del curso hacia conceptos propios de la disciplina, los conceptos identificados se muestran en la tabla siguiente.

Tabla 5. Conceptos.

Estudiantes	Contenido	Profesores	Curso
Materia	Programa	Prerrequisito	Aplicación
Metodología	Objetivo	Unidades	Facultad
Persona	Usuario	Pregrado	Postgrado
Código	Competencia	Argumentativa	Propositiva
Datos	Web	Bibliografía	Valor
Interpretativa	Cuestionarios	Talleres	Procedimiento
Currículo	Definición	Respuestas	Dirección

El proceso de creación de la ontología semántica partió del análisis de cada uno de los conceptos identificados, para determinar las posibles tripletas que se podían definir y que permitirán tener los datos semánticos que conformaran la ontología. Una vez se definieron las tripletas, se definió cada uno de los objetos para hacer claridad en el hecho que representa cada tripleta, y ayudar a determinar si se acepta o se elimina. Las tripletas se pueden observar en la tabla 6.

Tabla 6. Tripletas

SUJETO	PREDICADO	OBJETO	DEFINICIÓN DE OBJETOS
profesor	Dicta	curso	La asignatura que será dictada por el docente a cargo en este caso específico se puede decir que es Estructuras de Datos.
estudiante	Cursa	curso	La asignatura que será dictada por el docente a cargo en este caso específico se puede decir que es Estructuras de Datos
Profesor	Dirige	estudiante	Uno de los usuarios del sistema en este caso la persona que se encargara de aportar la información necesaria para la generación del currículo.
Curso	Contiene	competencias	Aquellas que el estudiante aporta al sistema para que estas sean evaluadas.
Curso	esCursado	estudiante	Uno de los usuarios del sistema en este caso la persona que se encargara de aportar la información necesaria para la generación del currículo.
Curso	esDictado	profesor	El docente a cargo del curso.
estudiante	esDirigido	profesor	El docente a cargo del curso.

Otros conceptos son inferidos en la construcción de la ontología por ser subclases de otros conceptos por ejemplo persona es superclase de estudiante por ello como tripleta no se la menciona (estudiante es_Una persona). Como se ve en la siguiente figura.

Figura 16. Estructura jerárquica de la ontología Clases y Subclases



Esta estructura la genera Protege además también genera las relaciones entre las clases antes mencionada (Figura 17):

Figura 17. Relaciones entre clases.



También las data propiedades (Figura 18):

Figura 18. Data propiedades y los individuos de las clases.

activo	
apellido_persona	
calificacion_competencia	
cedula_persona	◆ calculos_matematicos
correo_electronico_persona	◆ competencia-1
direccionWeb	◆ competencia-2
edad_persona	◆ curso1
id_competencia	◆ curso2
id_curso	◆ est2
id_unidad	◆ est3
nivel_curriculo	◆ estudiantel
nombre_competencia	◆ estudiante_1
nombre_curso	◆ estudiante_2
nombre_persona	◆ mirar_graficas
nombre_unidad	◆ profesor_1
nombre_usuario	◆ sumas_y_restas
porcentaje_competencia	◆ unidad-1234-1
respuestas	◆ unidad-1234-2
telefono_persona	
textocontenido	

Teniendo la integración de cada una de las tripletas que identifican las relaciones internas en la ontología, se podrá identificar más fácilmente el rumbo del prototipo.

4.6 DESARROLLO DEL PROTOTIPO

El primer paso en la implementación fue la instalación de la herramientas necesarias descritas anteriormente además de Protege para posteriormente crear la base de datos, las tablas e insertar cada uno de los registros que contienen información básica de usuarios que interactuaran con el prototipo.

Se parte de la idea de que la aplicación tendrá una base de datos llamada Cursos las tablas siguientes se las usa para generar el login de los usuarios de la aplicación (prototipo):

```
CREATE TABLE `estudiantes` (
  `idEstudiantes` int(11) NOT NULL AUTO_INCREMENT,
  `nombre_estudiante` varchar(45) NOT NULL,
  `apellido_estudiante` varchar(45) NOT NULL,
  `correo_electronico` varchar(45) NOT NULL,
  `telefono` varchar(9) DEFAULT NULL,
  `usuario_estudiante` varchar(16) NOT NULL,
  `password_estudiante` varchar(45) NOT NULL,
  `rol` int(11) NOT NULL,
  PRIMARY KEY (`idEstudiantes`)
)
CREATE TABLE `profesores` (
  `idProfesores` int(11) NOT NULL AUTO_INCREMENT,
```

```

`nombre_profesor` varchar(45) NOT NULL,
`apellido_profesor` varchar(45) NOT NULL,
`correo_electronico_profesor` varchar(45) NOT NULL,
`telefono_profesor` varchar(9) DEFAULT NULL,
`usuario_profesor` varchar(16) NOT NULL,
`password_profesor` varchar(45) NOT NULL,
`rol` int(11) NOT NULL,
PRIMARY KEY (`idProfesores`)
)
CREATE TABLE `estudiantesudenar` (
  `codigo_estudiante` int(11) NOT NULL,
  `password` varchar(45) DEFAULT NULL,
  PRIMARY KEY (`codigo_estudiante`)
)
CREATE TABLE `profesoresudenar` (
  `codigo_profesor` int(11) NOT NULL,
  `password` varchar(45) DEFAULT NULL,
  PRIMARY KEY (`codigo_profesor`)
)

```

La siguiente tabla es necesaria para manejar las notas que serán el resultado de evaluar las competencias a un estudiante:

```

CREATE TABLE `calificacion` (
  `idcalificacion` int(11) NOT NULL AUTO_INCREMENT,
  `nombre_usuario` varchar(45) NOT NULL,
  `id_curso` int(11) NOT NULL,
  `id_competencia` int(11) NOT NULL,
  `nota_competencia` int(11) NOT NULL,
  `porcentaje_competencia` int(11) NOT NULL,
  `id_unidad` int(11) NOT NULL,
  PRIMARY KEY (`idcalificacion`)
)
CREATE TABLE `cursos`.`registro_aprobado` (
  `idregistro_aprobado` INT NOT NULL ,
  `nombre_usuario` VARCHAR(45) NOT NULL ,
  `id_curso` INT NOT NULL ,
  `nota_pre` INT NOT NULL ,
  `porcentaje_pre` INT NOT NULL ,
  PRIMARY KEY (`idregistro_aprobado`)
)

```

4.6.1 Análisis. Para el desarrollo de este prototipo se procede a la descomposición en partes más simples y llevando un orden de las fases así:

- Definir los requerimientos del sistema.
- Definir los casos de uso que se necesitaran para lograr el prototipo
- Crear y mejorar los diagramas de los casos de uso
- Crear un modelo conceptual que se acoja al sistema
- Definir cada uno de los diagramas de secuencia del prototipo.

Requerimientos del sistema: se procede a describir las necesidades del sistema documentando en lo posible todo cuanto sea conveniente para en un paso posterior sea de fácil comprensión, para esto es importante tener en cuenta lo siguiente:

- Clientes del sistema
- Metas del sistema

A continuación se detallan mejor los apartados anteriores.

Clientes del sistema: Son aquellas personas que cumplen un rol o papel importante en el sistema a desarrollar.

- Estudiante
- Profesor

Metas del sistema: Estas generalizan las funciones que el sistema debe o va a realizar. La meta del software se resume en la obtención de un currículo adaptativo para un estudiante que va a cursar la materia estructuras de información.

Casos de uso: el objetivo de los casos de uso es lograr que las metas y funciones del sistema se vean de una forma más particular o específica gracias a un formato que describe la secuencia de pasos que se realizan en la búsqueda de una determinada tarea.

A continuación se muestran los casos de uso que el sistema abordará durante todo este proceso de desarrollo, es preciso aclarar que un caso de uso es la descripción de los procesos más comunes e importantes que se realizan en un sistema sea cual sea:

1 Iniciar sesión estudiante

UC-0001	Iniciar sesión Estudiante
Autores	<ul style="list-style-type: none">• <u>Edier Edison Arteaga Diaz</u>

Dependencias	□ <u>[UC-0006] Registrar un Estudiante a un curso</u>	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando <i>un usuario <u>Estudiante</u> o un <u>Profesor</u> registrados ingresan a su sitio en el sistema</i>	
Precondición	el <u>Profesor</u> o <u>Estudiante</u> deben estar registrados	
Secuencia normal	Paso	Acción
	1	Si el usuario esta en el sitio web, el actor <u>Estudiante (ACT-0001)</u> <i>ingresa a su sitio en el sistema</i>
	2	El sistema <i>muestra un formulario donde pide el login y la contraseña</i>
	3	El actor <u>Estudiante (ACT-0001)</u> <i>ingresa su login y contraseña</i>
	4	El sistema <i>verifica datos introducidos</i>
	5	El sistema <i>re direcciona al <u>Estudiante</u> a su página o su sitio</i>
Postcondición	el sistema re direcciona al <u>Estudiante</u> a su sitio donde le muestra su contenido	
Excepciones	Paso	Acción
	4	Si el login o password son incorrectos, el sistema <i>muestra mensaje de error después recarga la página y pide nuevamente el login y la contraseña, a continuación este caso de uso queda sin efecto</i>
	4	Si se omite el login o la contraseña, el sistema <i>muestra mensaje de ausencia de login o contraseña y permanece en la página hasta que el <u>Estudiante</u> complete sus datos, a continuación este caso de uso queda sin efecto</i>

2 Iniciar sesión profesor

UC-0002	Iniciar sesión Profesor	
Autores	• <u>Edier Edison Arteaga Diaz</u>	
Dependencias	□ <u>[UC-0008] Ingresar un Nuevo Curso</u>	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando <i>un usuario <u>Estudiante</u> o un <u>Profesor</u> registrados ingresan a su sitio en el sistema</i>	
Precondición	el <u>Profesor</u> o <u>Estudiante</u> deben estar registrados	
Secuencia	Paso	Acción

normal	1	Si <i>el usuario esta en el sitio web</i> , el actor <u>Profesor (ACT-0002)</u> <i>ingresa a su sitio en el sistema</i>
	2	El sistema <i>muestra un formulario donde pide el login y la contraseña</i>
	3	El actor <u>Profesor (ACT-0002)</u> <i>ingresa su login y contraseña</i>
	4	El sistema <i>verifica datos introducidos</i>
	5	El sistema <i>re direcciona al <u>Profesor</u> a su página o su sitio</i>
Postcondición	el sistema re direcciona al <u>Profesor</u> a su sitio donde le muestra su contenido	
Excepciones	Paso	Acción
	4	Si <i>el login o password son incorrectos</i> , el sistema <i>muestra mensaje de error después recarga la página y pide nuevamente el login y la contraseña</i> , a continuación este caso de uso <i>queda sin efecto</i>
	4	Si <i>se omite el login o la contraseña</i> , el sistema <i>muestra mensaje de ausencia de login o contraseña y permanece en la página hasta que el <u>Profesor</u> complete sus datos</i> , a continuación este caso de uso <i>queda sin efecto</i>

3 Registrar estudiante

UC-0003	Registrar Estudiante	
Autores	<ul style="list-style-type: none"> • <u>Edier Edison Arteaga Diaz</u> 	
Dependencias	<input type="checkbox"/> <u>[UC-0001] Iniciar Sesión Estudiante</u>	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando <i>un <u>estudiante</u> no registrado quiere entrar a su sitio</i>	
Precondición	No existir en la base de datos del sistema	
Secuencia normal	Paso	Acción
	1	Si <i>el usuario esta en el sitio web</i> , el actor <u>Estudiante (ACT-0001)</u> <i>ingresa a su sitio en el sistema</i>
	2	El sistema <i>muestra un formulario donde da la opción de registrarse</i>
	3	El actor <u>Estudiante (ACT-0001)</u> <i>ingresa por la opción de registro</i>
	4	El sistema <i>muestra un formulario donde pide los datos del <u>Estudiante</u></i>

	5	El actor <u>Estudiante (ACT-0001)</u> <i>ingresa todos sus datos</i>
	6	El sistema <i>verifica la veracidad de los datos</i>
	7	El sistema <i>registra al <u>estudiante</u> y muestra mensaje de éxito</i>
	8	El sistema <i>re direcciona al <u>Estudiante</u> a su página o su sitio</i>
Postcondición	Ser <u>Estudiante</u> registrado en el sistema	
Excepciones	Paso	Acción
	6	Si <i>Datos erróneos</i> , el sistema <i>muestra un mensaje de error de datos erróneos y no permite avanzar hasta que el <u>estudiante</u> corrija si es posible, a continuación este caso de uso queda sin efecto</i>
	6	Si los <i>datos introducidos no corresponden a un <u>estudiante</u> de la universidad</i> , el sistema <i>actualiza el formulario de ingreso de datos, a continuación este caso de uso queda sin efecto</i>

5 Registrar profesor

UC-0005	Registrar Profesor	
Autores	<ul style="list-style-type: none"> • <u>Edier Edison Arteaga Diaz</u> 	
Dependencias	<input type="checkbox"/> <u>[UC-0002] Iniciar Sesión Profesor</u>	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando <i>un <u>profesor</u> no registrado quiere entrar a su sitio</i>	
Precondición	No existir en la base de datos del sistema	
Secuencia normal	Paso	Acción
	1	Si <i>el usuario esta en el sitio web</i> , el actor <u>Profesor (ACT-0002)</u> <i>ingresa a su sitio en el sistema</i>
	2	El sistema <i>muestra un formulario donde da la opción de registrarse</i>
	3	El actor <u>Profesor (ACT-0002)</u> <i>ingresa por la opción de registro</i>
	4	El sistema <i>muestra un formulario donde pide los datos del <u>Profesor</u></i>
	5	El actor <u>Profesor (ACT-0002)</u> <i>ingresa todos sus datos</i>
	6	El sistema <i>verifica la veracidad de los datos</i>
	7	El sistema <i>registra al <u>profesor</u> y muestra mensaje de éxito</i>

	8	El sistema <i>re direcciona al <u>Profesor</u> a su página o su sitio</i>
Postcondición	Ser <u>Profesor</u> registrado en el sistema	
Excepciones	Paso	Acción
	6	Si <i>Datos erróneos</i> , el sistema <i>muestra un mensaje de error de datos erróneos y no permite avanzar hasta que el <u>profesor</u> corrija si es posible</i> , a continuación este caso de uso <i>queda sin efecto</i>
	6	Si los <i>datos introducidos no corresponden a un <u>profesor</u> de la universidad</i> , el sistema <i>actualiza el formulario de ingreso de datos</i> , a continuación este caso de uso <i>queda sin efecto</i>

6 Registrar un estudiante a un curso

UC-0006	Registrar un Estudiante a un curso	
Autores	<ul style="list-style-type: none"> • <u>Edier Edison Arteaga Diaz</u> 	
Dependencias	<input type="checkbox"/> <u>[UC-0007] Matricular un Estudiante a un curso</u>	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el <u>Estudiante</u> <i>pulsa la opción de registro a un curso que está en la lista de cursos disponibles</i>	
Precondición	el <u>Estudiante</u> no debe estar ya registrado al curso	
Secuencia normal	Paso	Acción
	1	Si el <u>estudiante</u> <i>selecciona la opción registrar del curso</i> , el sistema <i>lista un(os) prerrequisito(s) que el curso posea</i>
	2	Si el <u>Estudiante</u> <i>decide realizar la prueba</i> , el sistema <i>re direcciona al <u>Estudiante</u> a la prueba y este debe desarrollarla</i>
	3	El actor <u>Estudiante (ACT-0001)</u> <i>realiza la prueba y envía las respuestas al sistema</i>
	4	El sistema <i>registra y califica al <u>estudiante</u> y re direcciona a su sitio</i>
Postcondición	medir el nivel del <u>estudiante</u>	
Paso	Acción	Acción
	4	Si el <u>Estudiante</u> <i>falla en la prueba</i> , el sistema <i>re direcciona a la lista de cursos y no permite matricular el curso</i> , a continuación este caso de uso <i>queda sin efecto</i>

7 Matricular un estudiante a un curso

UC-0007	Matricular un Estudiante a un curso	
Autores	<ul style="list-style-type: none"> • <u>Edier Edison Arteaga Diaz</u> 	
Dependencias	Ninguno	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el <u>Estudiante</u> <i>Pulsa la opción de matricularse a un curso que está en la lista de cursos disponibles</i>	
Precondición	Que haya logrado pasar el Prerrequisito del curso	
Secuencia normal	Paso	Acción
	1	Si el <u>Estudiante</u> seleccionó la opción <i>matricular</i> , el sistema <i>lista una prueba o serie de pruebas(competencias) que el curso posea</i>
	2	Si el <u>Estudiante</u> decide realizar la prueba, el sistema <i>re direcciona al Estudiante a la prueba y este debe desarrollarla</i>
	3	El actor <u>Estudiante (ACT-0001)</u> <i>realiza la prueba y envía las respuestas al sistema</i>
	4	El sistema <i>califica al estudiante en la prueba y re direcciona a otra prueba(si existe)</i>
	5	El sistema <i>matricula al estudiante y le genera su currículum correspondiente y se lo muestra</i>
Postcondición	Generar el currículum adaptativo	
Excepciones	Paso	Acción
	1	Si <i>no existe ninguna prueba o competencia a realizar</i> , el sistema <i>salta al paso 5 de este caso de uso, a continuación este caso de uso queda sin efecto</i>
	4	Si el <u>Estudiante</u> <i>no logra pasar la prueba</i> , el sistema <i>califica lo realizado por el Estudiante</i> , a continuación este caso de uso <i>continúa</i>

8 Ingresar un nuevo curso

UC-0008	Ingresar un Nuevo Curso
Autores	<ul style="list-style-type: none"> • <u>Edier Edison Arteaga Diaz</u>
Dependencias	<input type="checkbox"/> <u>[UC-0009] Ingresar prerrequisito a un curso</u>

	<input type="checkbox"/> <u>[UC-0011] Ingresar Nueva competencia</u>	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando <i>un docente decida ingresar un nuevo curso mediante la opción que le brinda el sistema</i>	
Precondición	el <u>profesor</u> debe haber iniciado sesión	
Secuencia normal	Paso	Acción
	1	Si se ha seleccionado opción de nuevo curso, el sistema re direcciona a una nueva página para ingresar los datos básicos del curso
	2	El actor <u>Profesor (ACT-0002)</u> rellena el formulario con los datos del curso
	3	Si todos los datos están correctos y completos, el actor <u>Profesor (ACT-0002)</u> selecciona enviar curso
	4	El sistema ingresa el curso al sistema y re direcciona al <u>Profesor</u> a su sitio
Postcondición	ingresar su prerrequisito	

9 Ingresar prerrequisito a un curso

UC-0009	Ingresar prerrequisito a un curso	
Autores	<ul style="list-style-type: none"> • <u>Edier Edison Arteaga Diaz</u> 	
Dependencias	<input type="checkbox"/> <u>[UC-0010] Dar de alta/baja un Curso</u>	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando <i>el curso ya ha sido agregado y necesite su respectivo prerrequisito para ser publicado</i>	
Precondición	existir el curso	
Secuencia normal	Paso	Acción
	1	Si se ha seleccionado la opción ingresar prerrequisito, el sistema re direcciona al <u>profesor</u> a la página en donde se piden los datos del prerrequisito
	2	El actor <u>Profesor (ACT-0002)</u> rellena el formulario con los datos básicos del prerrequisito
	3	Si todos los datos están correctos y completos, el actor <u>Profesor (ACT-0002)</u> selecciona la opción enviar prerrequisito
	4	El sistema ingresa el prerrequisito al sistema y a su respectivo curso y re direcciona al <u>Profesor</u> a su sitio
Postcondición	completar datos del curso	

10 Dar de alta/baja un curso

UC-0010	Dar de alta/baja un Curso	
Autores	<ul style="list-style-type: none"> • <u>Edier Edison Arteaga Diaz</u> 	
Dependencias	<input type="checkbox"/> <u>[UC-0006] Registrar un Estudiante a un curso</u>	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando <i>el <u>profesor</u> decide dar de alta o baja a un curso en particular</i>	
Precondición	de existir el prerrequisito del curso	
Secuencia normal	Paso	Acción
	1	Si el <u>profesor</u> decide dar de alta/baja el curso, el sistema cambia el estado del curso para que este pueda o no ser publicado para los estudiantes
Postcondición	los estudiantes puedan verlo ,registrarlo, matrricularlo	

11 Ingresar nueva competencia

UC-0011	Ingresar Nueva competencia	
Autores	<ul style="list-style-type: none"> • <u>Edier Edison Arteaga Diaz</u> 	
Dependencias	Ninguno	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando <i>cuando un <u>profesor</u> decide complementar un curso adicionando competencias a un curso</i>	
Precondición	el curso debe existir	
Secuencia normal	Paso	Acción
	1	El actor <u>Profesor (ACT-0002)</u> <i>ingresa al curso</i>
	2	El sistema <i>re direcciona al <u>profesor</u> al contenido del curso</i>
	3	El actor <u>Profesor (ACT-0002)</u> <i>selecciona del contenido la unidad a la cual va a ingresar su competencia</i>
	4	El sistema <i>re direcciona a la página que posee el contenido de la unidad</i>
	5	El actor <u>Profesor (ACT-0002)</u> <i>selecciona ingresar nueva competencia</i>
	6	El sistema <i>re direcciona a una página en donde se piden los datos de esta mediante un formulario</i>
	7	El actor <u>Profesor (ACT-0002)</u> <i>rellena el formulario con los</i>

		<i>datos necesarios</i>
	8	<i>Si todos los datos están correctos y completos, el actor <u>Profesor (ACT-0002)</u> selecciona la opción <u>enviar competencia</u></i>
	9	<i>El sistema <u>ingresa la competencia al sistema</u> y a la respectiva a <u>unidad</u> y re <u>direcciona al profesor al contenido de la unidad</u> por si este decide <u>ingresar otra competencia</u></i>
Postcondición	los estudiantes la realicen	

Diagramas de casos de uso: estos diagramas se emplean para dar una visión más clara del comportamiento deseado del sistema además ayudan cuando se quiere continuar mejor orientados en la construcción del sistema y así comprender la manera en la cual van enlazados los procesos o tareas específicas.

También ayudan en la validación de la arquitectura y a verificar el sistema mientras evoluciona a lo largo del desarrollo, como también una delimitación y alcance que deberían cumplir para no salirse del verdadero fin.

Para este sistema los diagramas de caso de uso son los mostrados a continuación:

Figura 19. Contenido estudiante

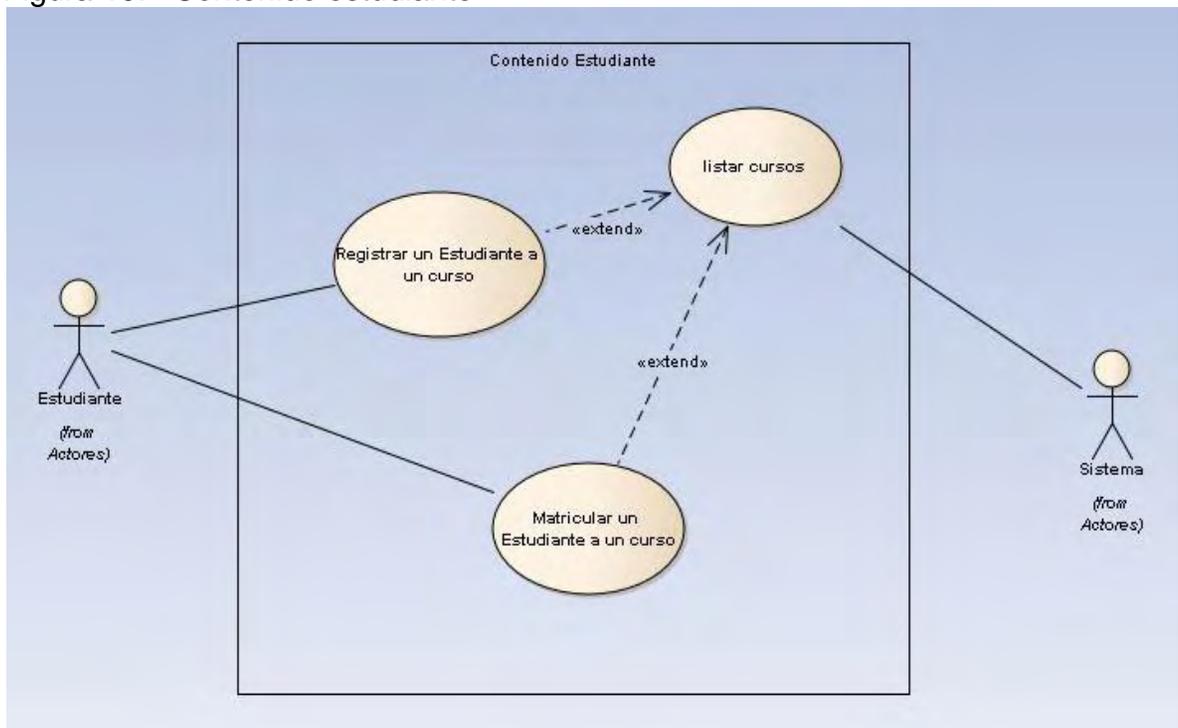


Figura 20. Contenido profesor

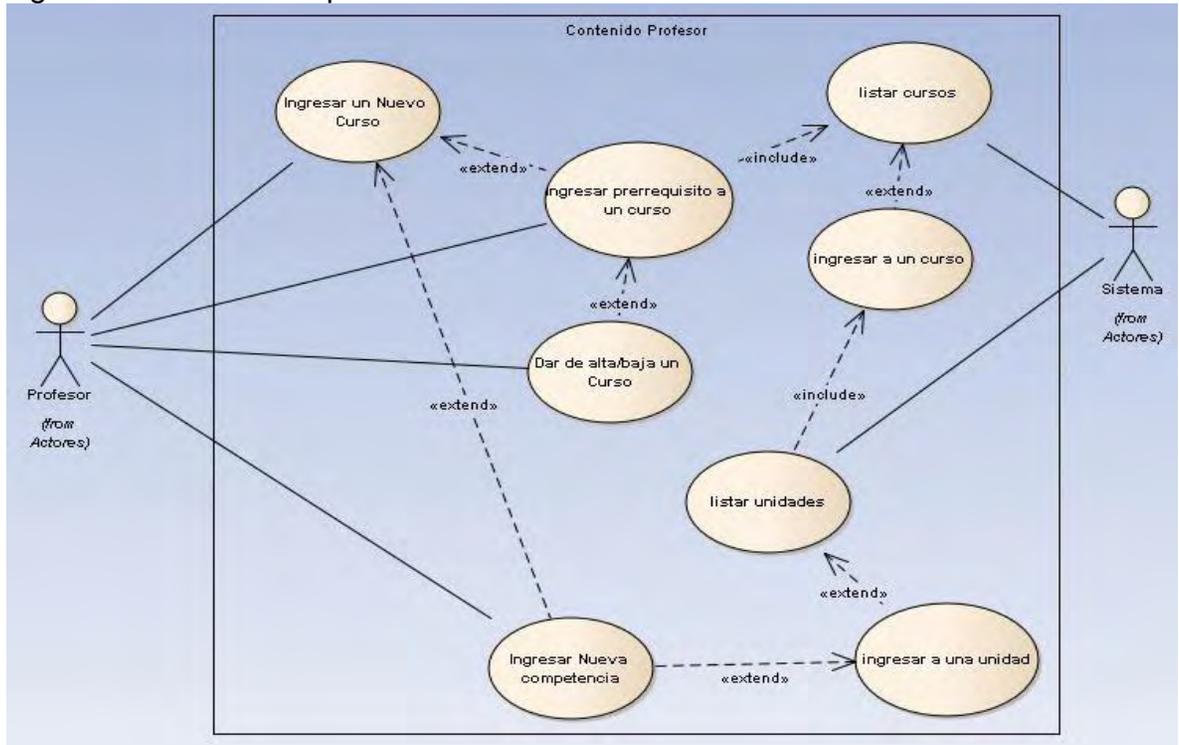


Figura 21. Registrar estudiante

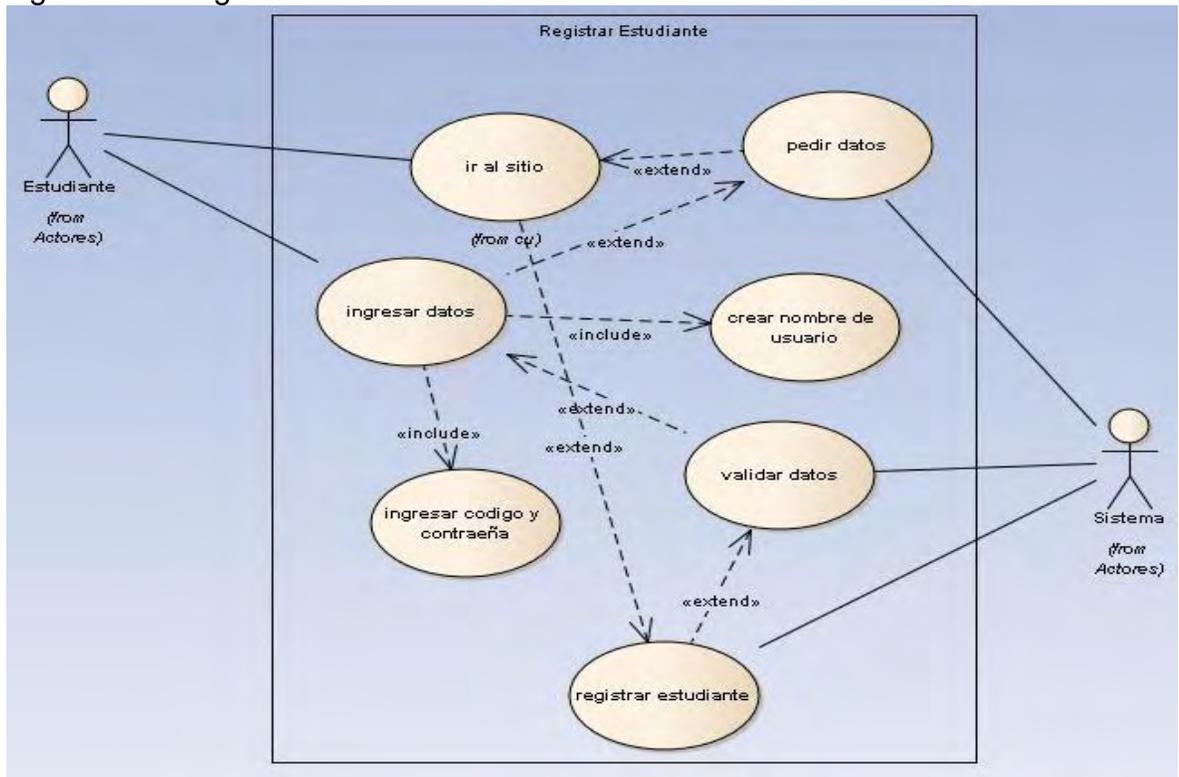


Figura 22. Registrar profesor

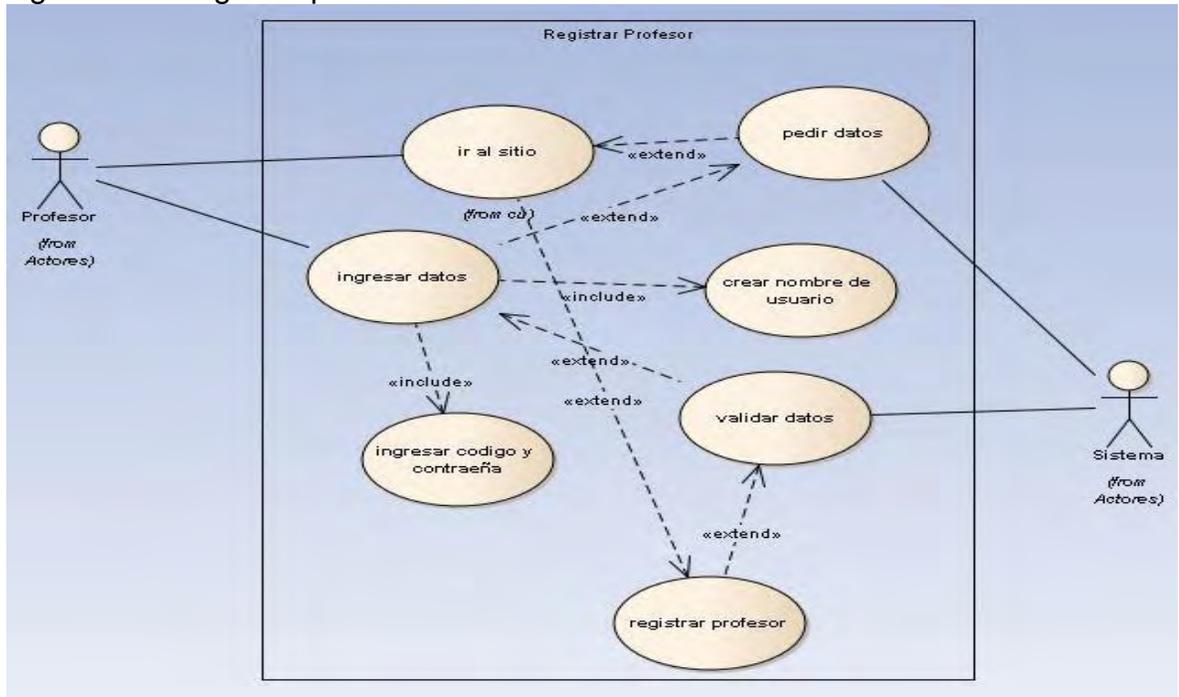


Figura 23. Iniciar sesión estudiante

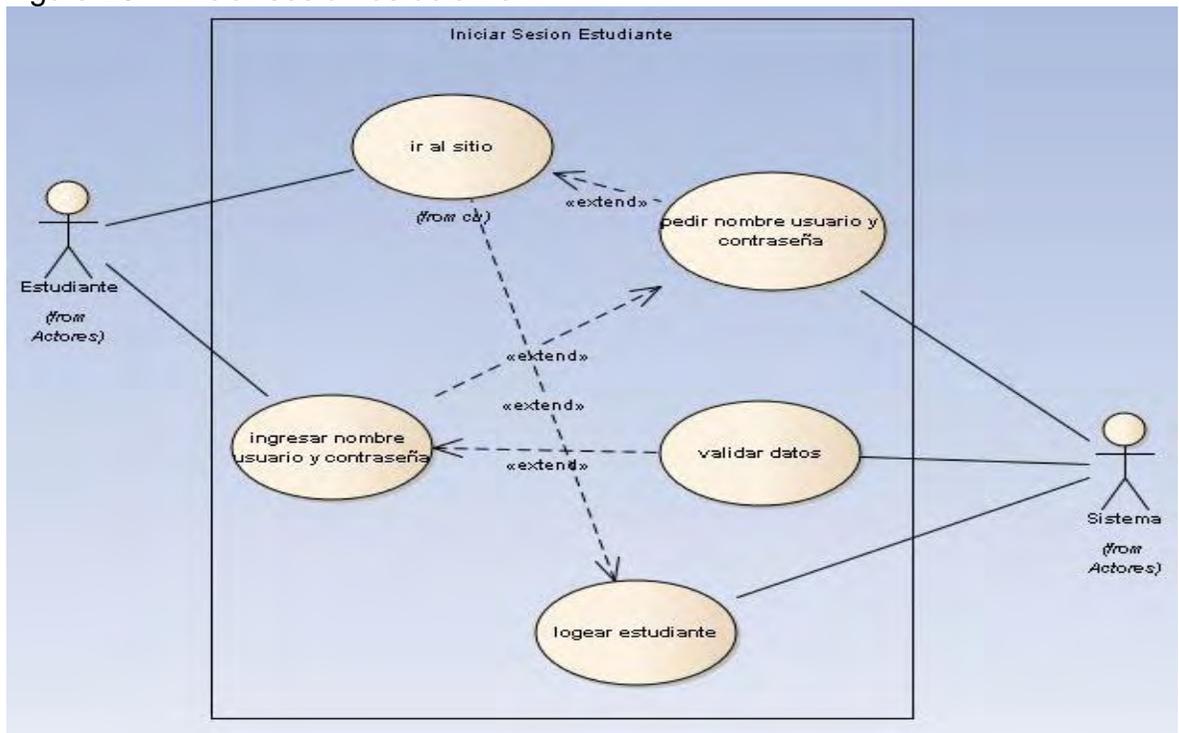


Figura 24. Iniciar sesión profesor

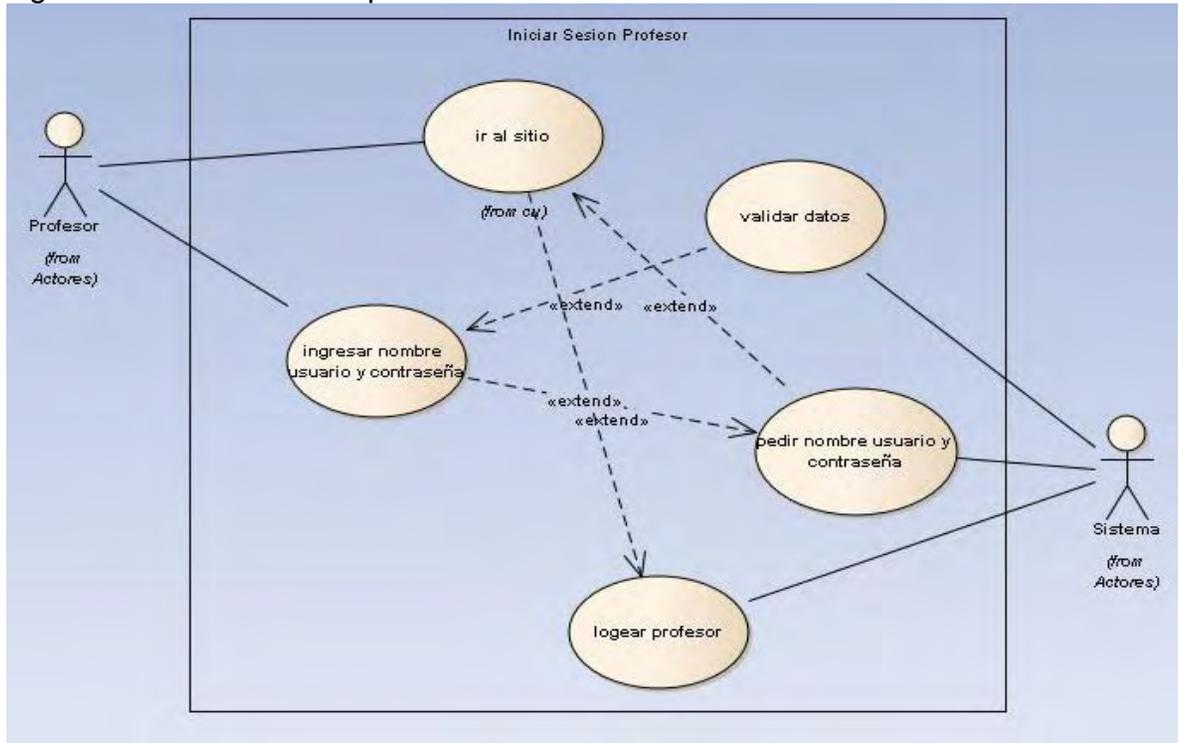


Figura 25. Registrar un estudiante a un curso

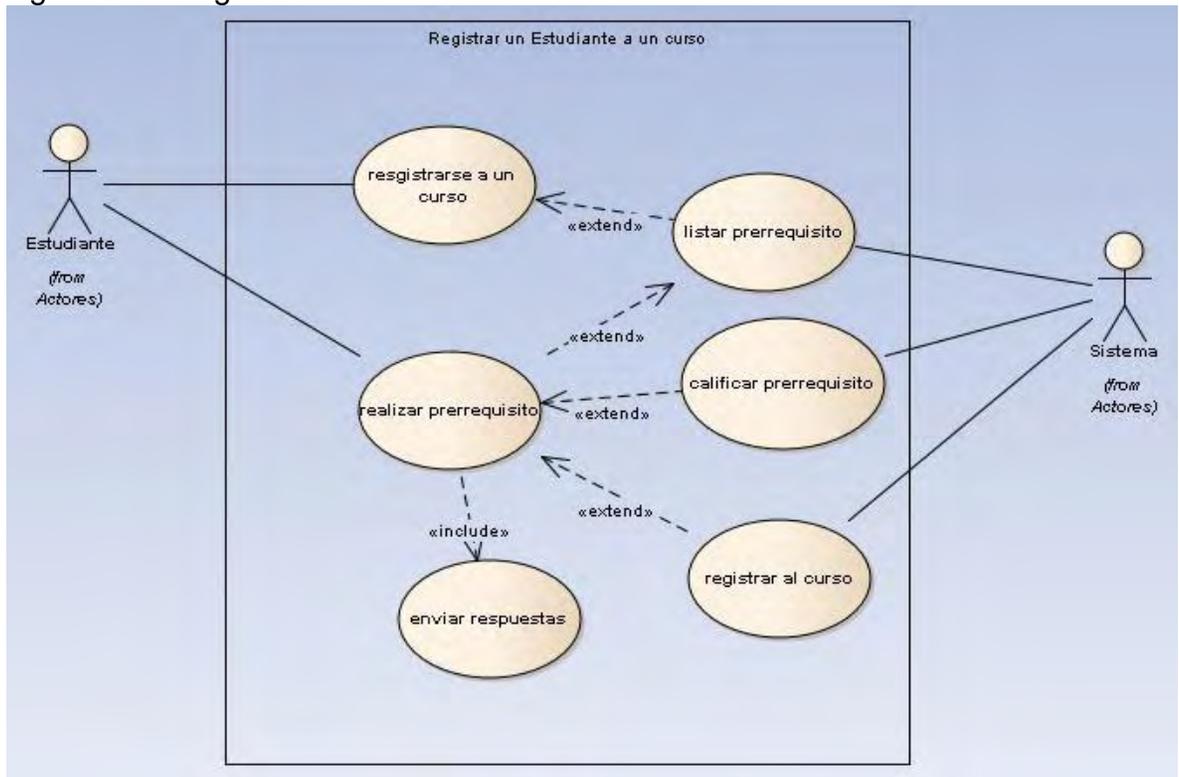


Figura 26. Matricular un estudiante a un curso

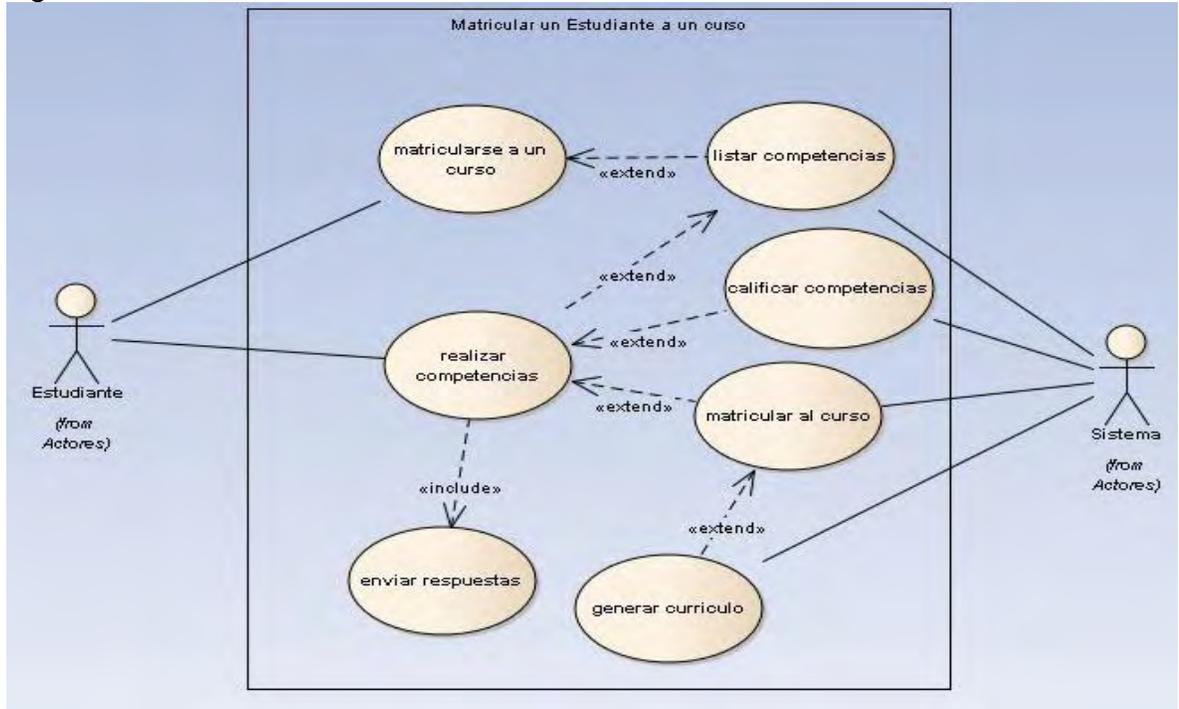


Figura 27. Ingresar un nuevo curso

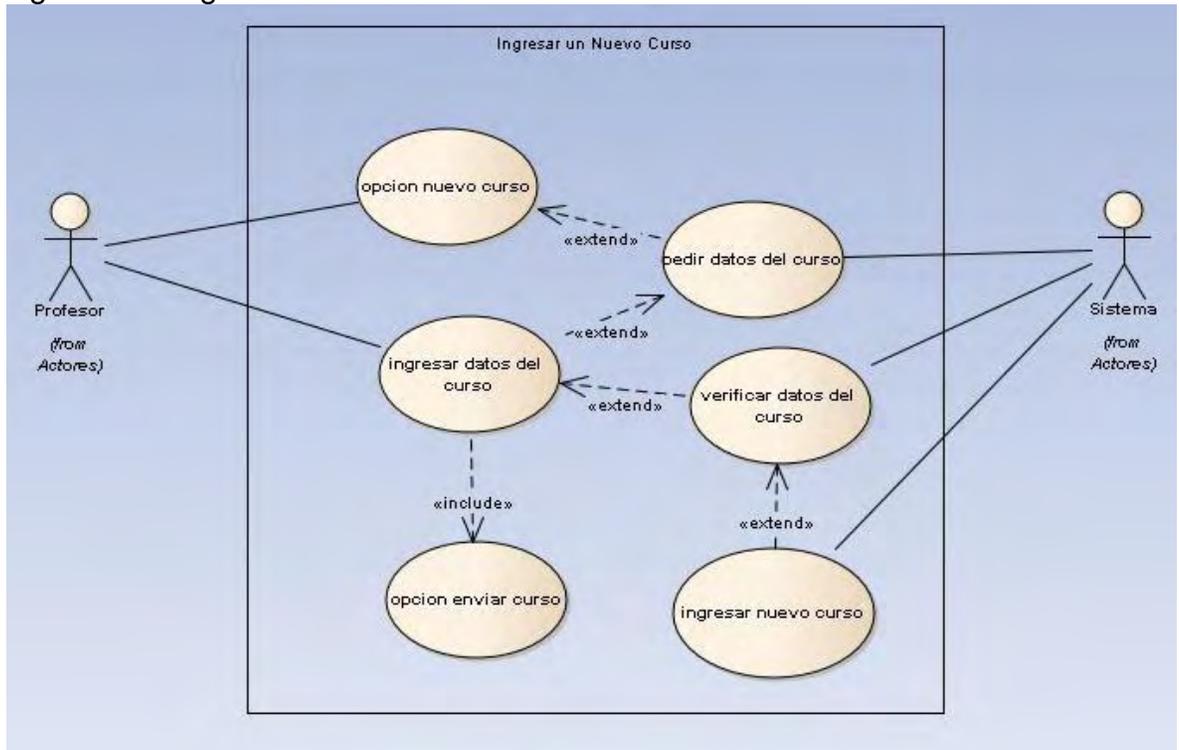


Figura 28. Ingresar prerequisite a un curso

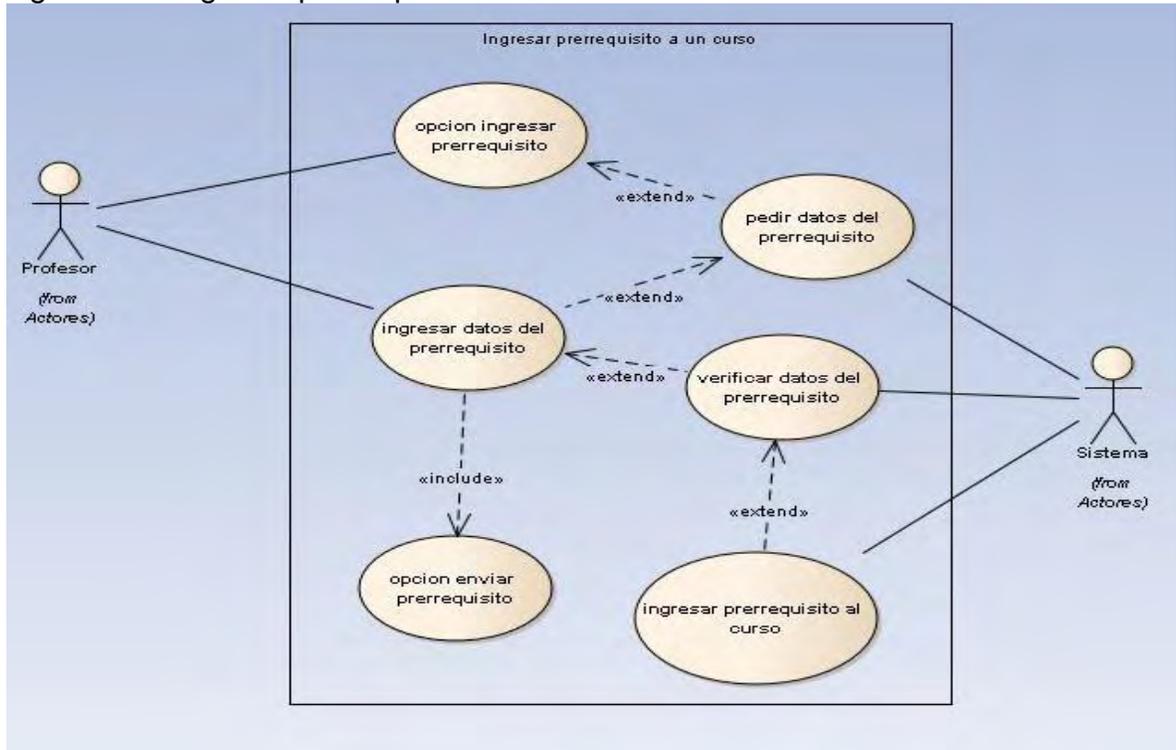


Figura 29. Dar de alta/baja un curso

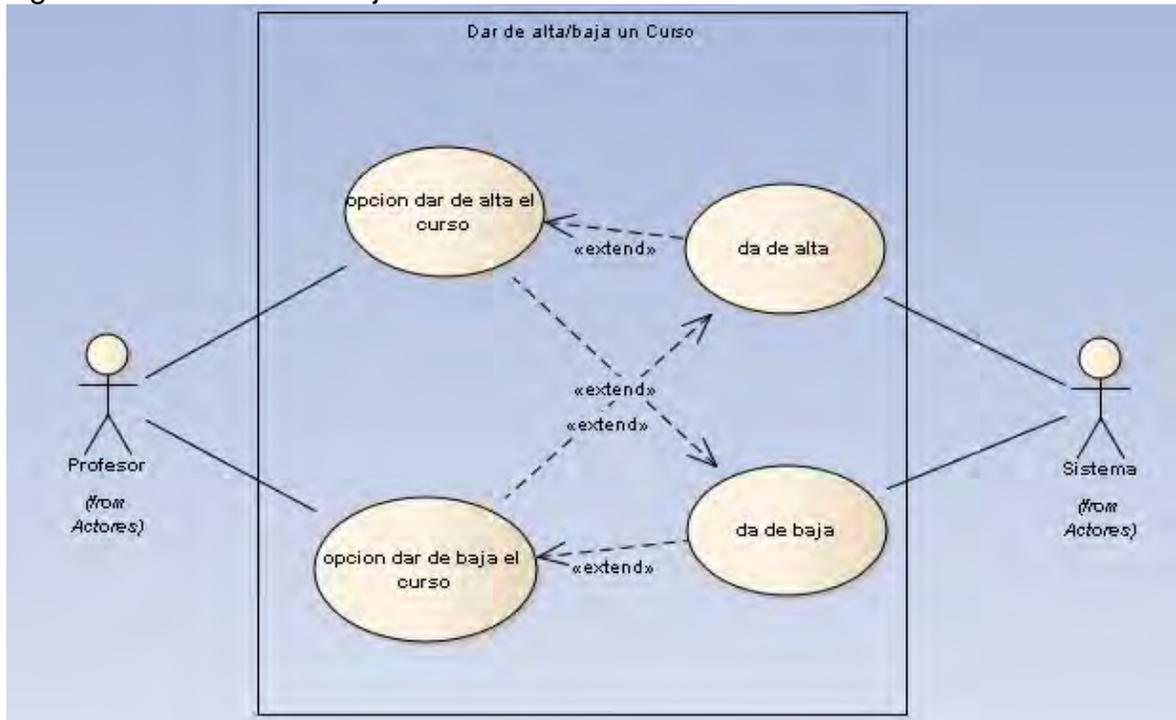
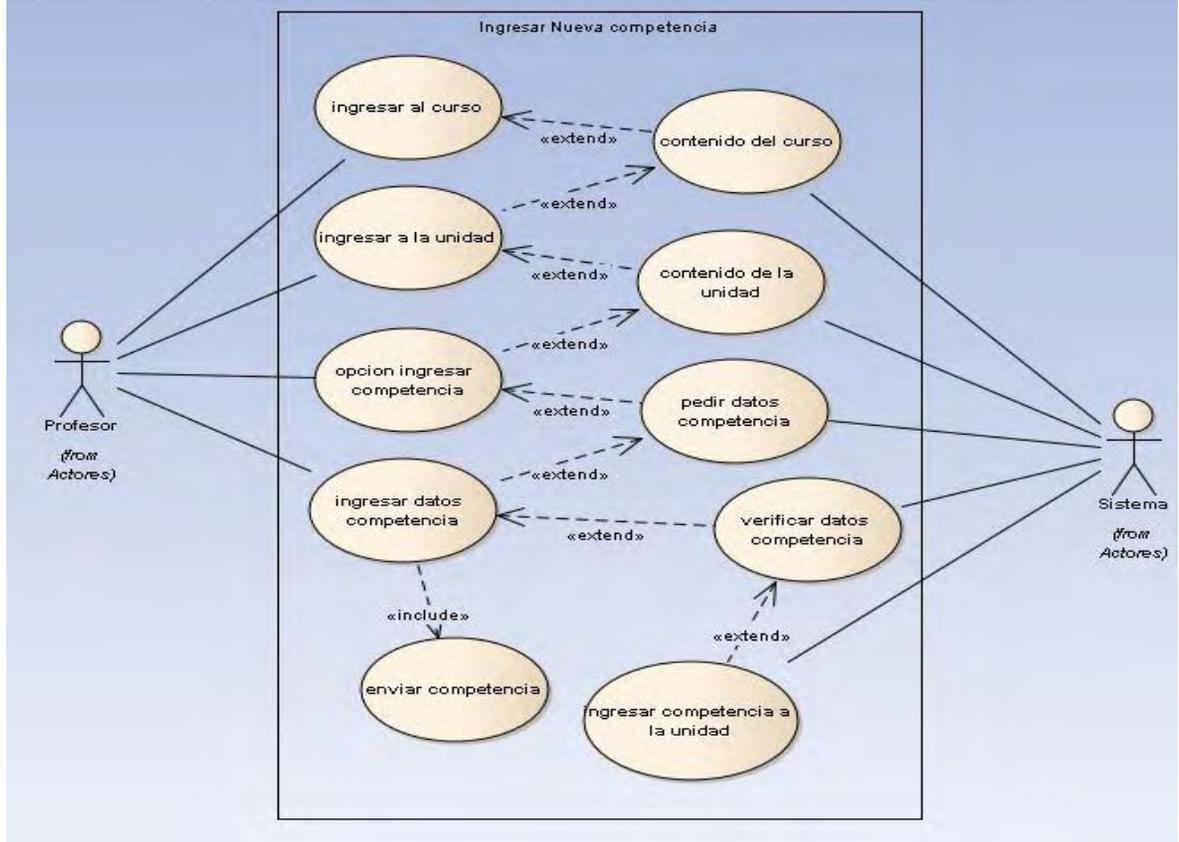


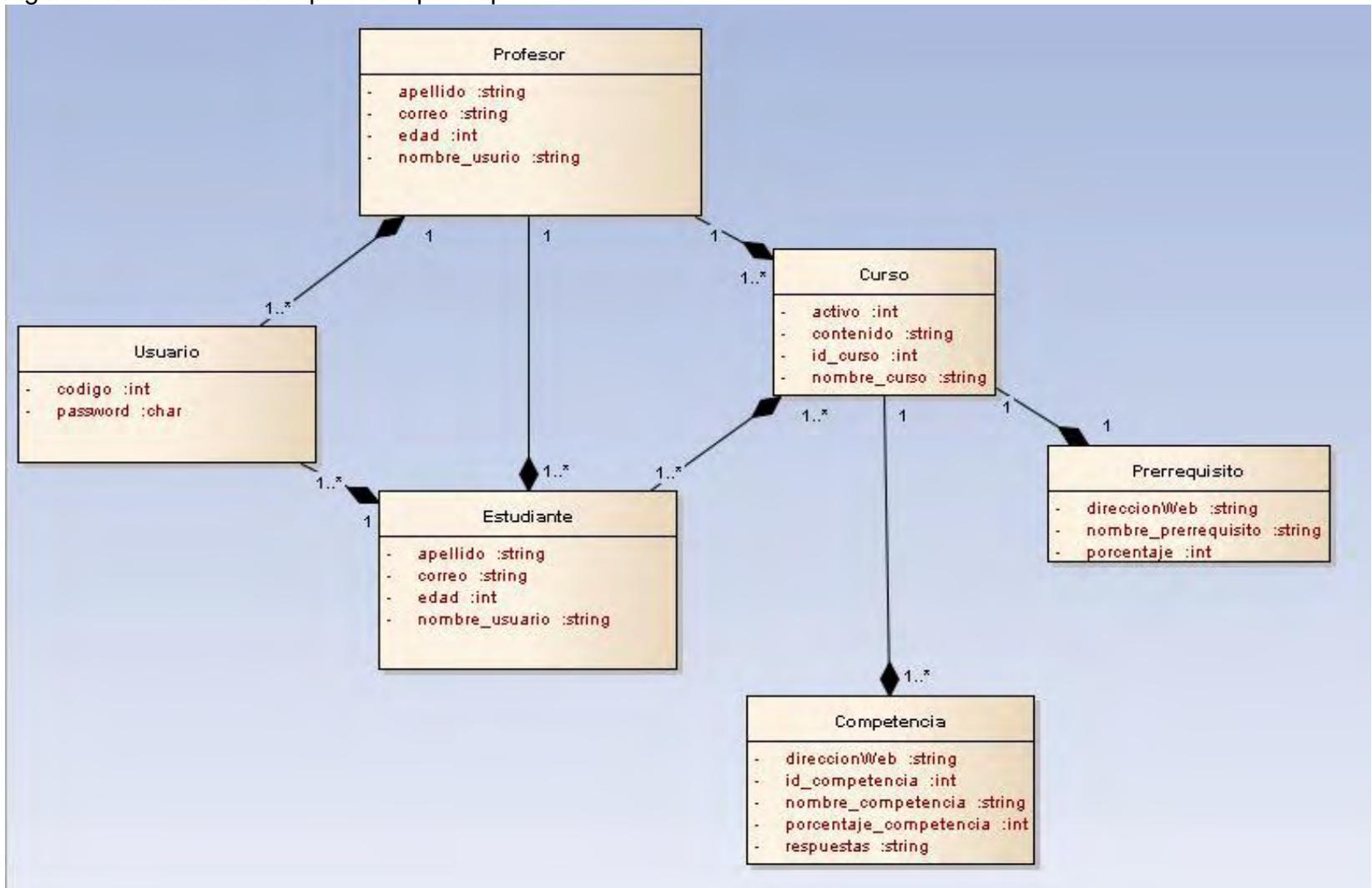
Figura 30. Ingresar nueva competencia



Modelo conceptual: Este modelo conceptual brinda una idea más clara de los conceptos que se deben abordar, estos conceptos son los más relevantes que la aplicación maneja. De manera tal que se observen las relaciones que existen entre estos conceptos proporcionando así una visión real del sistema y como se acoplaría a los requerimientos.

A continuación se presenta el diagrama que representa el modelo conceptual (Figura 31):

Figura 31. Modelo conceptual del prototipo



Diagramas de secuencia del sistema: Un diagrama de secuencia de cualquier sistema muestra de manera gráfica los las acciones que realizan los actores del sistema y desencadenan acciones de aquí que los diagramas de secuencia muestran como los actores interactúan con el sistema.

Es decir, es la representación gráfica a la solución a un caso de uso cuando este es iniciado por un actor del sistema mostrando un orden lógico que permite el desarrollo de la aplicación de manera coherente.

Los diagramas de secuencia son los siguientes:

Figura 32. Registrarse como estudiante

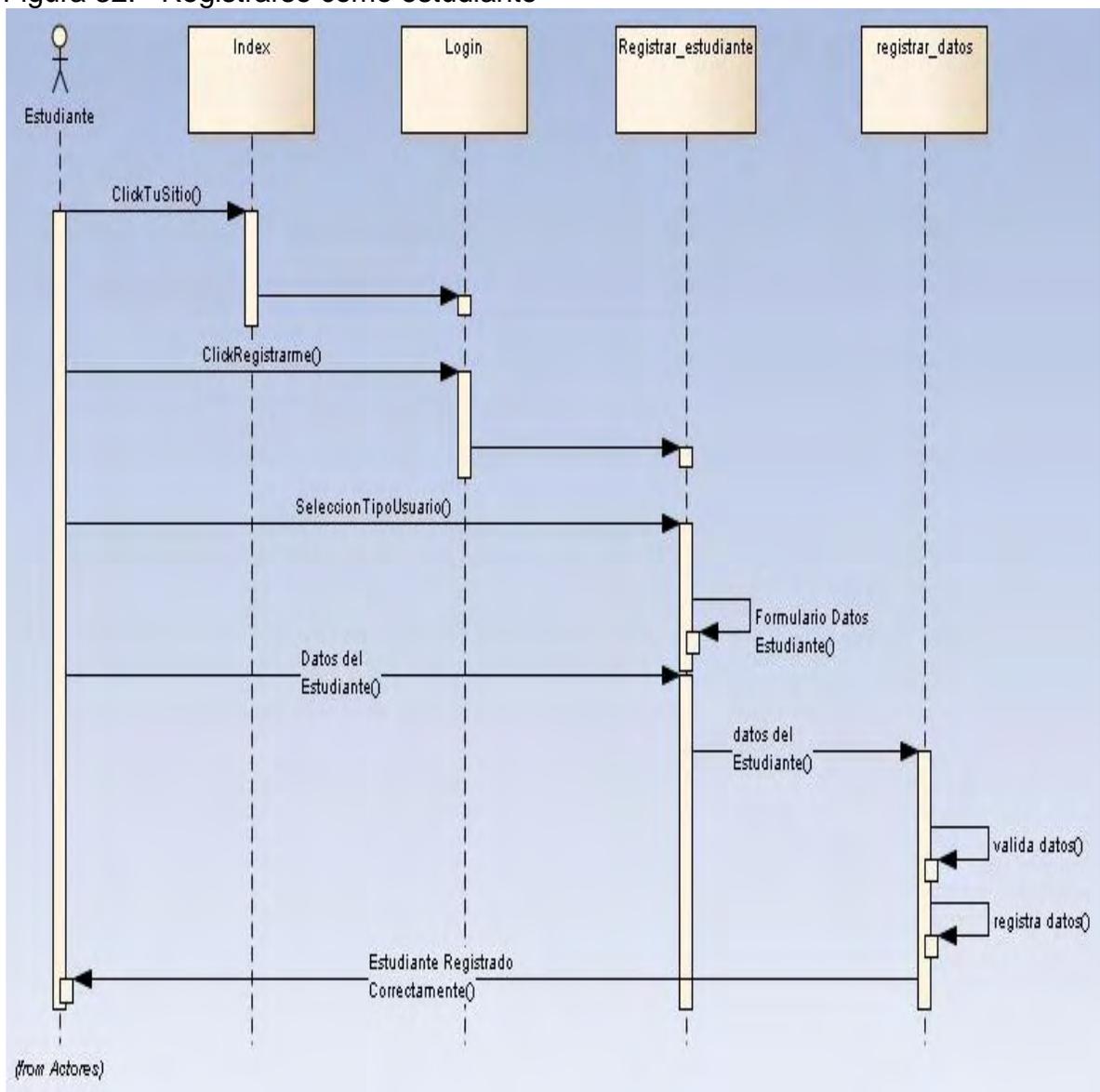


Figura 33. Registrarse como profesor

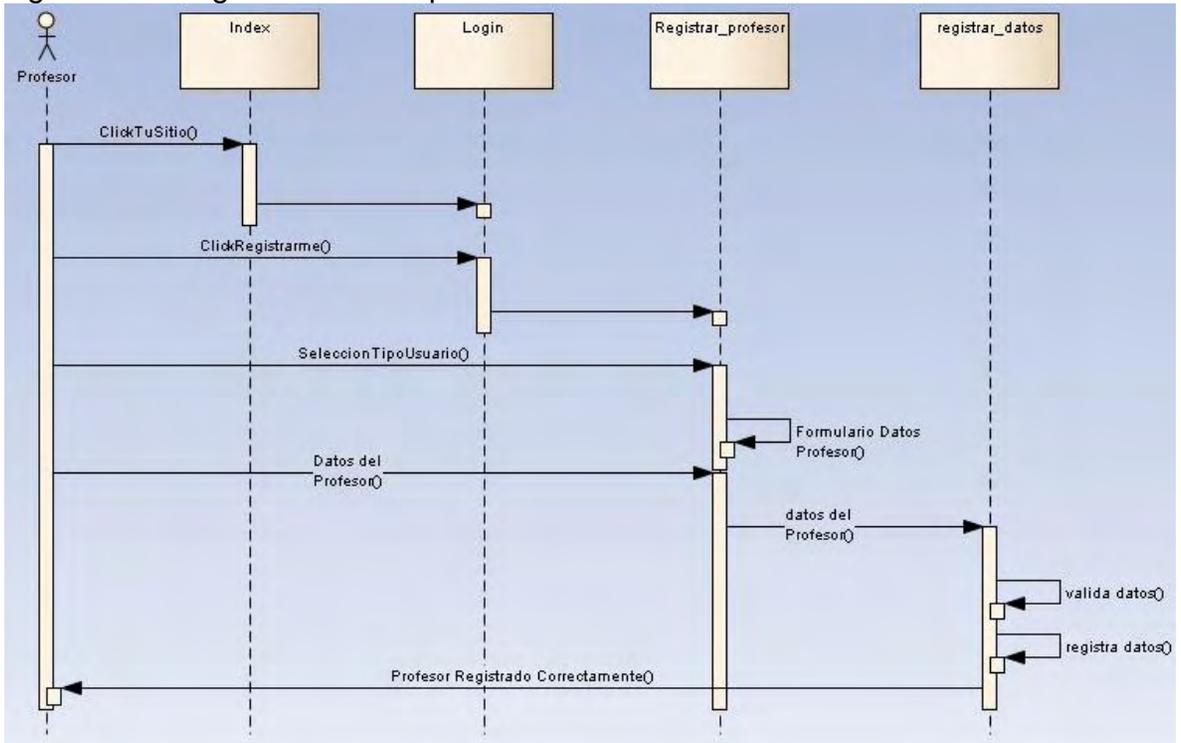


Figura 34. Logearse como estudiante

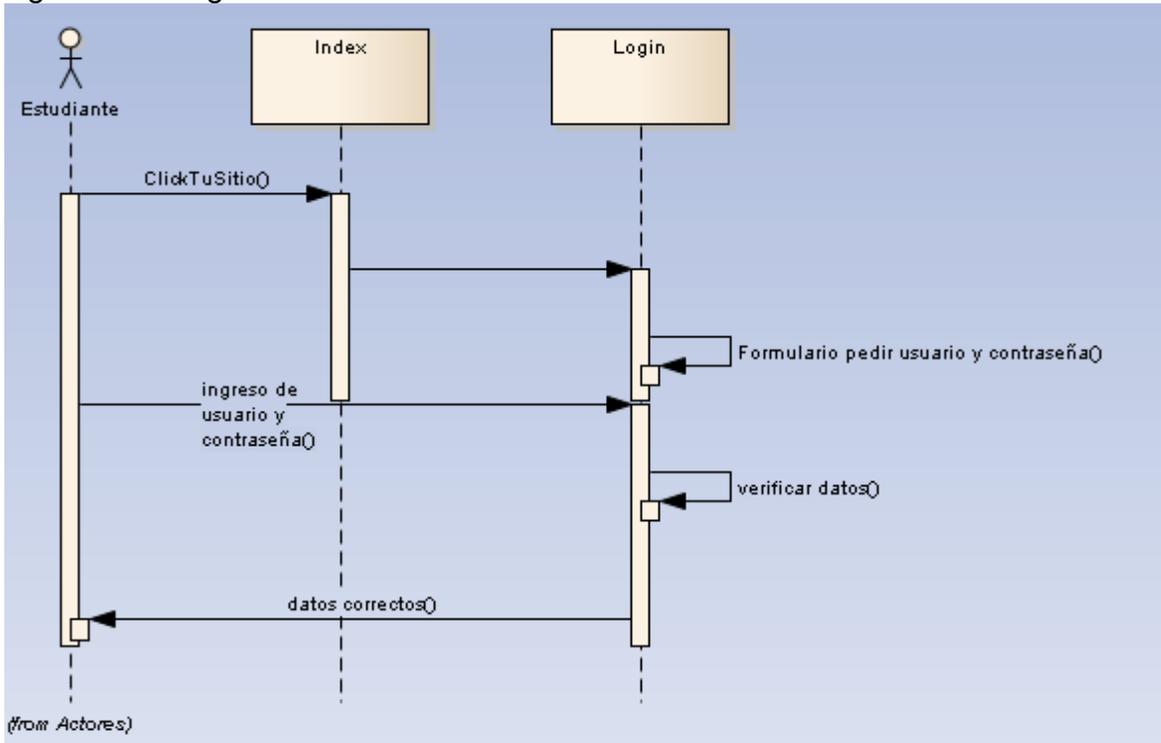


Figura 35. Logearse como profesor

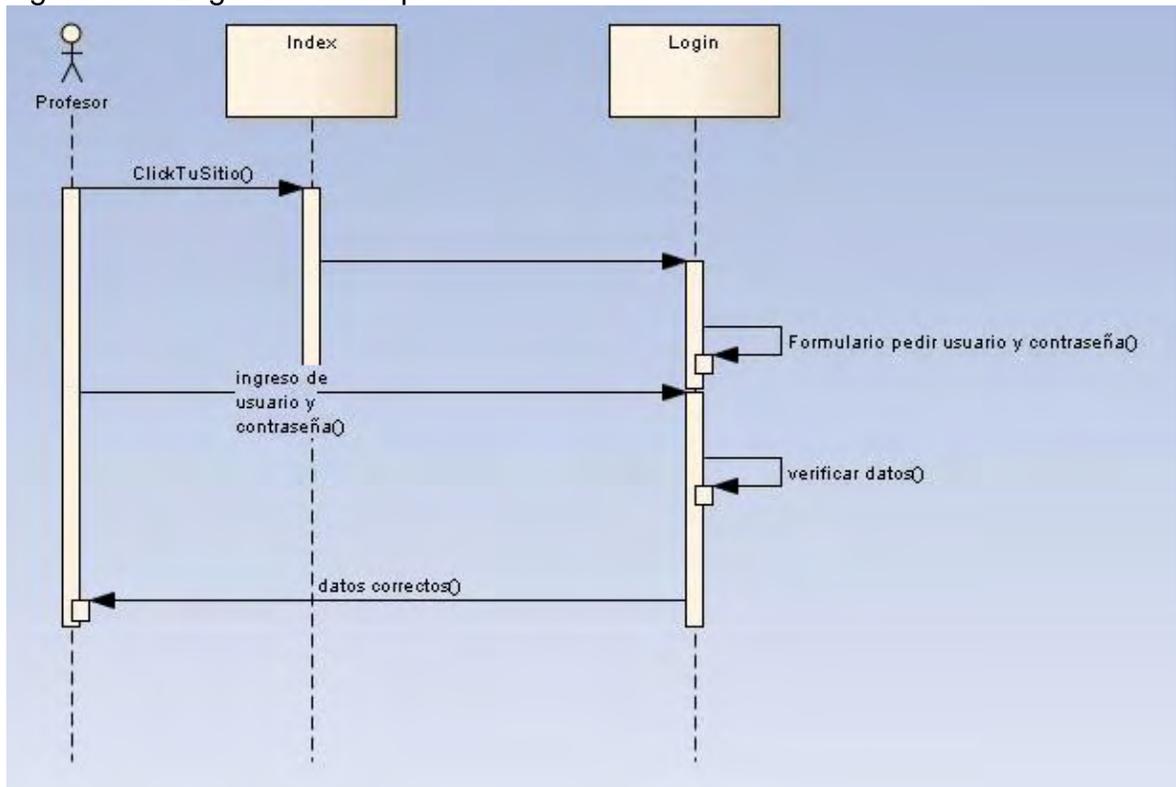


Figura 36. Registrarse a un curso

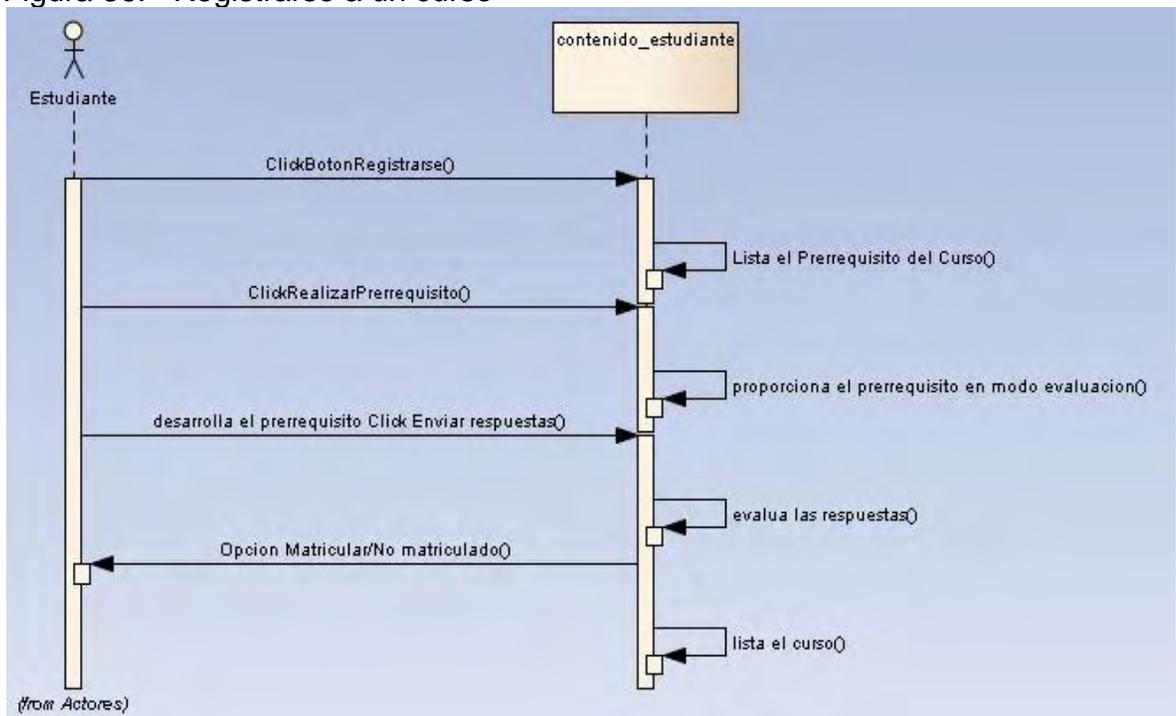


Figura 37. Matricularse a un curso

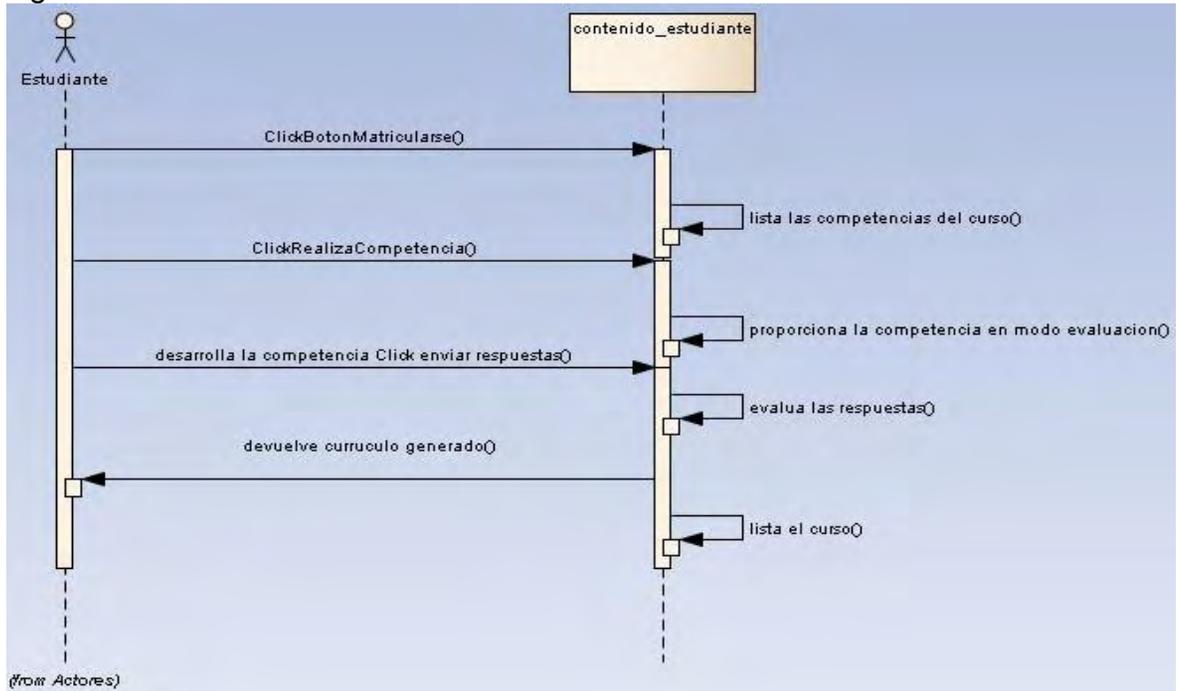


Figura 38. Ingresar nuevo curso

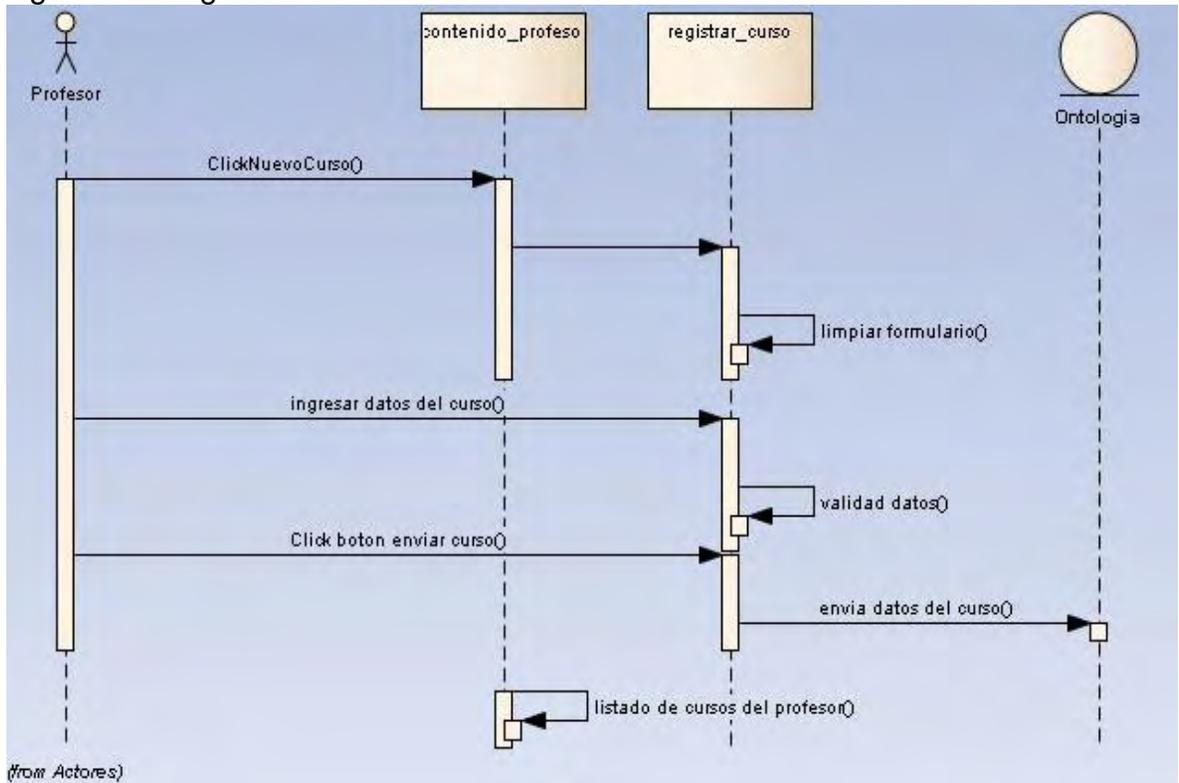


Figura 39. Ingresar nuevo prerrequisito

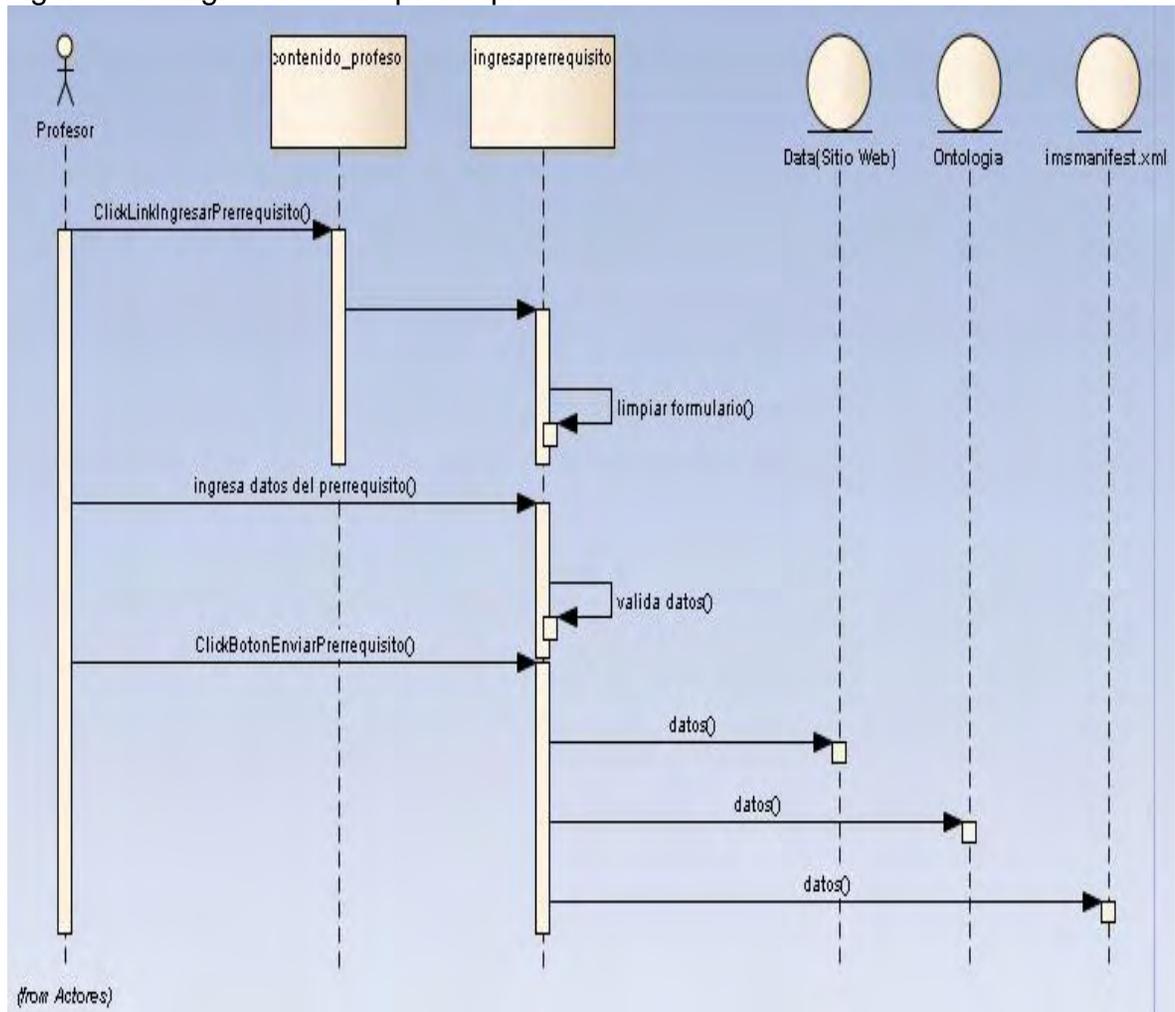


Figura 40. Dar de alta o de baja a un curso

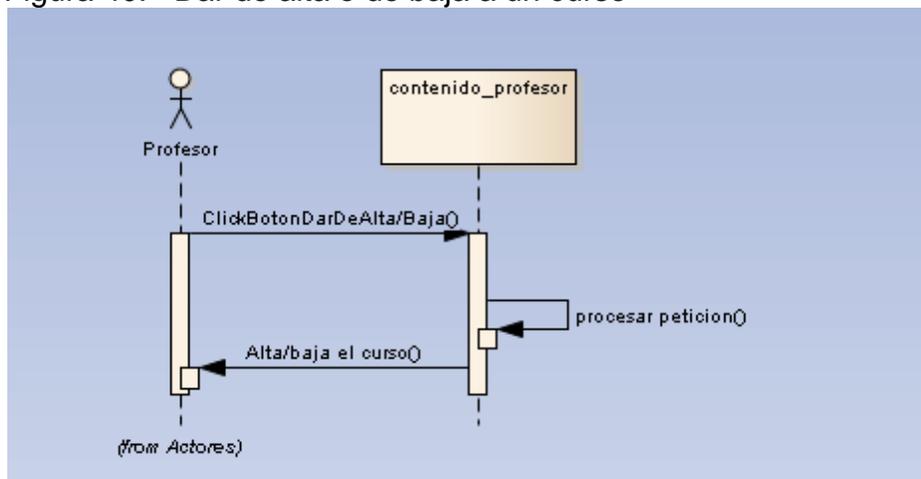
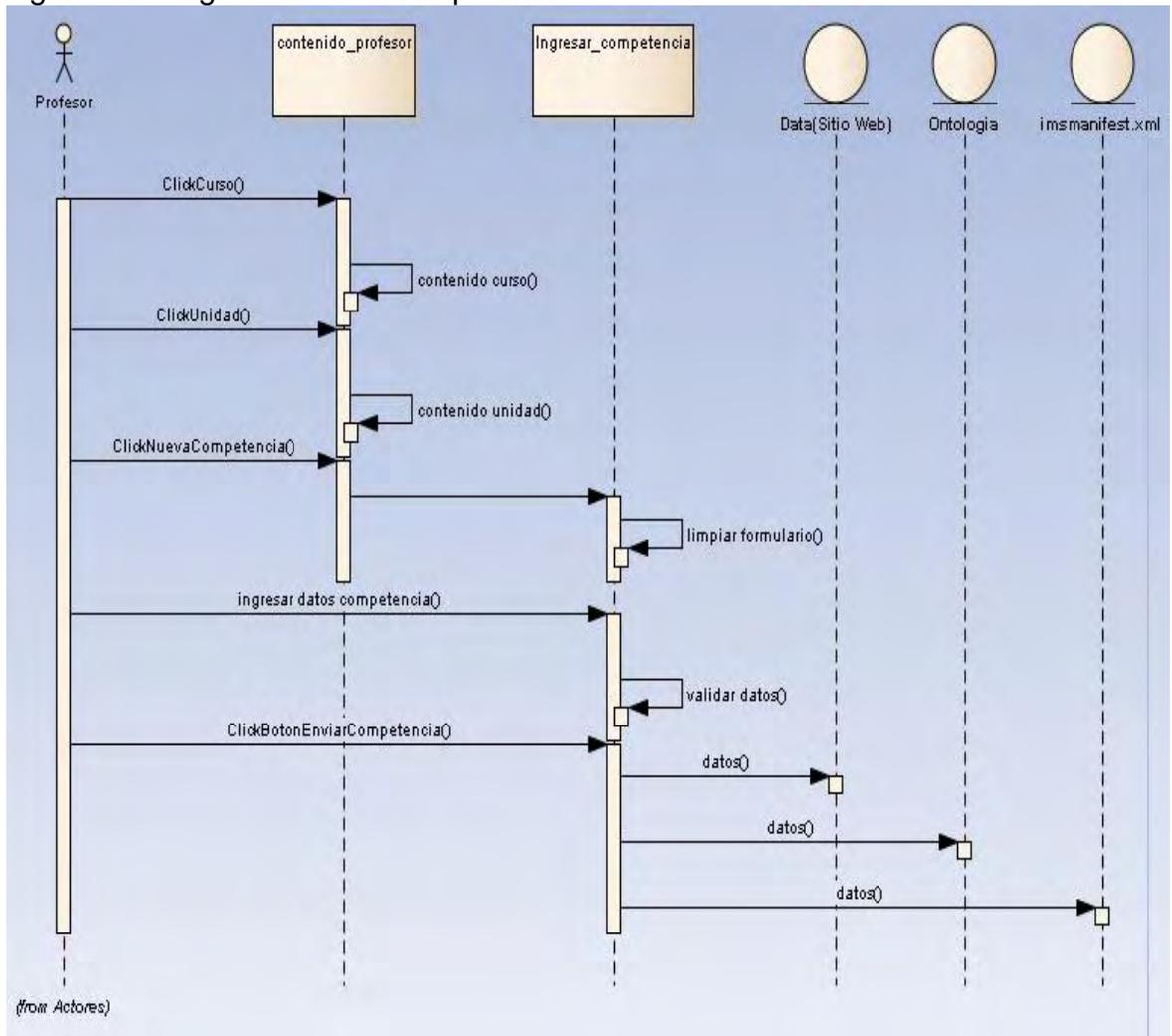


Figura 41. Ingresar nueva competencia



4.6.2 Diseño. Una vez enfatizado en la fase de análisis y de haber aclarado muchos ítems se puede avanzar en el proceso de construcción de la aplicación y se continúa con la fase de diseño, esta fase comprende:

- Definición de los casos de uso reales.
- Definición de los reportes, la interfaz de usuario y la secuencia de las pantallas.
- Perfeccionamiento de la arquitectura del sistema.
- Definición de los diagramas de interacción.
- Definición de los diagramas de diseño de clases.

Casos de uso reales: Estos casos de uso ayudan a comprender con mucha claridad la realización de una acción del sistema, se puede decir que son un complemento de los casos de uso de la fase de análisis, aquí se da una visión más clara del sistema en realidad con imágenes ((Figuras 42 a 51) que describen exactamente los procesos o casos de uso reales, además de la figura 52 que representa el diagrama de clases del sistema,

1 Iniciar sesión estudiante

Figura 42. Iniciar sesión estudiante



UC-0001	Iniciar Sesión Estudiante	
Autores	<ul style="list-style-type: none"> • <u>Edier Edison Arteaga Diaz</u> 	
Dependencias	<input type="checkbox"/> <u>[UC-0006] Registrar un Estudiante a un curso</u>	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando <i>un usuario <u>Estudiante</u> registrado ingresa a su sitio en el sistema</i>	
Precondición	el <u>Estudiante</u> debe estar registrado	
Secuencia normal	Paso	Acción
	1	Si <i>el usuario esta en el sitio web como en la figura 1, el actor <u>Estudiante (ACT-0001)</u> ingresa a su sitio en el sistema numeral A.</i>
	2	El sistema <i>muestra un formulario donde pide el login y la contraseña numeral B en la figura 1.</i>
	3	El actor <u>Estudiante (ACT-0001)</u> <i>ingresa su login y contraseña numeral B.</i>
	4	El sistema <i>verifica datos introducidos</i>
	5	El sistema <i>re direcciona al <u>Estudiante</u> a su página o su sitio numeral C en la figura 1.</i>
Postcondición	el sistema re direcciona al <u>Estudiante</u> a su sitio donde le muestra su contenido	
Excepciones	Paso	Acción
	4	Si <i>el login o password son incorrectos, el sistema muestra mensaje de error después recarga la página y pide nuevamente el login y la contraseña, a continuación este caso de uso queda sin efecto</i>
	4	Si <i>se omite el login o la contraseña, el sistema muestra mensaje de ausencia de login o contraseña y permanece en la página hasta que el <u>Estudiante</u> complete sus datos, a continuación este caso de uso queda sin efecto</i>

1 Iniciar sesión profesor

Figura 43. Iniciar sesión profesor



UC-0002	Iniciar Sesión Profesor	
Autores	<ul style="list-style-type: none"> • <u>Edier Edison Arteaga Diaz</u> 	
Dependencias	<input type="checkbox"/> <u>[UC-0008] Ingresar un Nuevo Curso</u>	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando un usuario <u>Profesor</u> registrado <i>ingresa a su sitio en el sistema</i>	
Precondición	el <u>Profesor</u> debe estar registrado	
Secuencia normal	Paso	Acción
	1	Si el usuario esta en el sitio web, el actor <u>Profesor (ACT-0002)</u> <i>ingresa a su sitio en el sistema tal y como lo muestra la figura 2 numeral A.</i>
	2	El sistema muestra un formulario donde pide el login y la contraseña como en la figura 2 numeral B.
3	El actor <u>Profesor (ACT-0002)</u> <i>ingresa su login y</i>	

		contraseña numeral B de la figura 2.
	4	El sistema verifica datos introducidos
	5	El sistema re direcciona al <u>Profesor</u> a su página o su sitio como se muestra en la figura 2 numeral C .
Postcondición	el sistema re direcciona al <u>Profesor</u> a su sitio donde le muestra su contenido	
Excepciones	Paso	Acción
	4	Si el login o password son incorrectos, el sistema muestra mensaje de error después recarga la página y pide nuevamente el login y la contraseña, a continuación este caso de uso queda sin efecto
	4	Si se omite el login o la contraseña, el sistema muestra mensaje de ausencia de login o contraseña y permanece en la página hasta que el <u>Profesor</u> complete sus datos, a continuación este caso de uso queda sin efecto

2 Registrar estudiante

Figura 44. Registrar estudiante

CurriculoAdaptativo 

A [TuSitio](#)

B [Registrar](#)

Estudiante C

Datos Basicos Del Estudiante	
Nombre Completo	<input type="text"/>
Apellidos	<input type="text"/>
Correo Electronico	<input type="text"/>
Telefono (opcional)	<input type="text"/>
Nombre de Usuario	<input type="text"/>
Codigo Estudiantil	<input type="text"/>
Password	<input type="password"/>
<input type="button" value="Enviar"/>	

UC-0003	Registrar Estudiante	
Autores	<ul style="list-style-type: none"> • <u>Edier Edison Arteaga Diaz</u> 	
Dependencias	<input type="checkbox"/> <u>[UC-0001] Iniciar Sesión Estudiante</u>	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando <i>un <u>estudiante</u> no registrado quiere entrar a su sitio</i>	
Precondición	No existir en la base de datos del sistema	
Secuencia normal	Paso	Acción
	1	Si <i>el usuario</i> esta en el sitio web, el actor <u>Estudiante (ACT-0001)</u> <i>ingresa a su sitio en el sistema como lo muestra la figura 3 numeral A.</i>
	2	El sistema <i>muestra un formulario donde da la opción de registrarse figura 3.</i>
	3	El actor <u>Estudiante (ACT-0001)</u> <i>ingresa por la opción de registro figura 3 numeral B.</i>
	4	El sistema <i>muestra un formulario donde pide los datos del <u>Estudiante</u> figura 3 numeral C.</i>
	5	El actor <u>Estudiante (ACT-0001)</u> <i>ingresa todos sus datos numeral C.</i>
	6	El sistema <i>verifica la veracidad de los datos</i>
	7	El sistema <i>registra al <u>estudiante</u> y muestra mensaje de éxito</i>
	8	El sistema <i>re direcciona al <u>Estudiante</u> a su página o su sitio como en la figura 1 numeral C.</i>
Postcondición	Ser <u>Estudiante</u> registrado en el sistema	
Excepciones	Paso	Acción
	6	Si <i>Datos erróneos</i> , el sistema <i>muestra un mensaje de error de datos erróneos y no permite avanzar hasta que el <u>estudiante</u> corrija si es posible, a continuación este caso de uso queda sin efecto</i>
	6	Si los <i>datos introducidos no corresponden a un <u>estudiante</u> de la universidad</i> , el sistema <i>actualiza el formulario de ingreso de datos, a continuación este caso de uso queda sin efecto</i>

3 Registrar profesor

Figura 45. Registrar profesor

CurriculoAdaptativo π

B [Regístrate](#)

A [Tu Sitio](#)

C Docente

Datos Basicos Del Docente	
Nombre Completos	<input type="text"/>
Apellidos	<input type="text"/>
Correo Electronico	<input type="text"/>
Telefono (opcional)	<input type="text"/>
Nombre de Usuario	<input type="text"/>
Codigo Profesor	<input type="text"/>
Password	<input type="text"/>
<input type="button" value="Enviar"/>	

UC-0005	Registrar Profesor	
Autores	<ul style="list-style-type: none"> • Edier Edison Arteaga Diaz 	
Dependencias	<input type="checkbox"/> [UC-0002] Iniciar Sesión Profesor	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando <i>un profesor no registrado quiere entrar a su sitio tal y como muestra la figura 4 numeral A.</i>	
Precondición	No existir en la base de datos del sistema	
Secuencia	Paso	Acción

normal	1	Si el usuario esta en el sitio web, el actor <u>Profesor (ACT-0002)</u> <i>ingresa a su sitio en el sistema numeral A.</i>
	2	El sistema muestra un formulario donde da la opción de registrarse figura 4 numeral B.
	3	El actor <u>Profesor (ACT-0002)</u> <i>ingresa por la opción de registro numeral B.</i>
	4	El sistema muestra un formulario donde pide los datos del <u>Profesor</u> tal y como muestra la figura 4 numeral C.
	5	El actor <u>Profesor (ACT-0002)</u> <i>ingresa todos sus datos</i>
	6	El sistema verifica la veracidad de los datos
	7	El sistema registra al <u>profesor</u> y muestra mensaje de éxito
	8	El sistema re direcciona al <u>Profesor</u> a su página o su sitio tal y como muestra la figura 2 numeral C.
Postcondición	Ser <u>Profesor</u> registrado en el sistema	
Excepciones	Paso	Acción
	6	Si Datos erróneos, el sistema muestra un mensaje de error de datos erróneos y no permite avanzar hasta que el <u>profesor</u> corrija si es posible, a continuación este caso de uso queda sin efecto
	6	Si los datos introducidos no corresponden a un <u>profesor</u> de la universidad, el sistema actualiza el formulario de ingreso de datos, a continuación este caso de uso queda sin efecto

6 Registrar un estudiante a un curso

Figura 46. Registrar un estudiante a un curso

Curriculo Adaptativo

Inicio

Curso 1 **A**

Curso 2 < No Matriculado > **B**

Usuario: Estudiante
Cerrar Sesión

Para matricularte al Curso es Necesario que relices la siguiente prueba:

Prueba: [prerequisitocurso1](#) **C**

preguntas de opcion multiple con unica respuesta

1: pregunta 1

respuesta 1 Correcta

respuesta 2

respuesta 3

respuesta 4

D

UC-0006	Registrar un Estudiante a un curso	
Autores	<ul style="list-style-type: none"> • Edier Edison Arteaga Diaz 	
Dependencias	<input type="checkbox"/> [UC-0007] Matricular un Estudiante a un curso	
Descripción	<p>El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el <u>Estudiante</u> pulsa la opción de registro a un curso que está en la lista de cursos disponibles como lo muestra la figura 5 numeral A</p>	
Precondición	<p>el <u>Estudiante</u> no debe estar ya registrado al curso</p>	
Secuencia normal	Paso	Acción
	1	<p>Si el <u>estudiante</u> selecciona la opción registrar del curso del numeral A de la figura 5, el sistema lista el <u>prerequisito</u> que el curso posee como se ve en el numeral C de la misma figura.</p>

	2	Si el <u>Estudiante</u> decide realizar la prueba, el sistema re direcciona al <u>Estudiante</u> a la prueba y este debe desarrollarla figura 5 numeral D , en este caso la prueba sol consta de una sola pregunta pero eso depende del docente que es quien realiza la prueba.
	3	El actor <u>Estudiante</u> (ACT-0001) realiza la prueba y envía las respuestas al sistema numeral D .
	4	El sistema registra y califica al <u>estudiante</u> y re direcciona a su sitio numeral D de la figura 1.
Postcondición	medir el nivel del <u>estudiante</u>	
Paso	Acción	Acción
	4	Si el <u>Estudiante</u> falla en la prueba, el sistema re direcciona a la lista de cursos y no permite matricular el curso tal y como muestra el numeral B , a continuación este caso de uso queda sin efecto

7 Matricular un estudiante a un curso

Figura 47. Matricular un estudiante a un curso

Inicio

Curriculo Adeptativo

Usuario: Estudiante1
Cerrar Sesion

Curso1 Matricularme **(A)**

Para iniciar el Curso es Necesario que relices las siguientes pruebas:

Prueba: competencia **(B)**

Prueba: competencia2unidad2 **(B)**

preguntas de opcion multiple con unica respuesta

1: pregunta competencia 1

respuesta competencia 1 Correcta

respuesta competencia 2

respuesta competencia 3

respuesta competencia 4

(C)

Enviar Respuestas

UC-0007	Matricular un Estudiante a un curso	
Autores	<ul style="list-style-type: none"> • <u>Edier Edison Arteaga Diaz</u> 	
Dependencias	Ninguno	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el <u>Estudiante</u> <i>Pulsa la opción de matricularse a un curso que está en la lista de cursos disponibles</i>	
Precondición	Que haya logrado pasar el Prerrequisito del curso	
Secuencia normal	Paso	Acción
	1	Si el <u>Estudiante</u> seleccionó la opción matricular tal y como lo muestra la figura 6 en el apartado A , el sistema lista una prueba o serie de pruebas(competencias) que el curso posea figura 6 numeral B .
	2	Si el <u>Estudiante</u> decide realizar la prueba, el sistema re direcciona al <u>Estudiante</u> a la prueba y este debe desarrollarla como se ve en la figura 6 numeral C .
	3	El actor <u>Estudiante (ACT-0001)</u> realiza la prueba y envía las respuestas al sistema el numeral C de la figura anterior.
	4	El sistema califica al <u>estudiante</u> en la prueba y re direcciona a otra prueba(si existe) como en el numera B en este caso existen dos pruebas.
	5	El sistema <i>matricula al estudiante</i> y le genera su currículum correspondiente y se lo muestra
Postcondición	Generar el currículum adaptativo	
Excepciones	Paso	Acción
	1	Si no existe ninguna prueba o competencia a realizar o falla en una, el sistema salta al paso 5 de este caso de uso, a continuación este caso de uso queda sin efecto

8 Ingresar un nuevo curso

Figura 48. Ingresar un nuevo curso

Logo: UNIVERSIDAD DE MARICAO

Curso: π

Inicio | Contenido General | Homee

Nuevo Curso **(A)**

Usuario: Profesor1
Cerrar Sesión

NO TTENES CURSOS INSCRITOS

Nombre Curso ID **(B)**

Enviar curso **(A)**

Ingrese el contenido del Curso

Unidad 1:	Item 1:	SubItem 1:1	+	+	Enviar Unidad
-----------	---------	-------------	---	---	---------------

Nombre Del Curso **(C)**

[Curso1](#) [ingresar prerrequisito para este curso](#)

UC-0008	Ingresar un Nuevo Curso	
Autores	<ul style="list-style-type: none"> • Edier Edison Arteaga Diaz 	
Dependencias	<input type="checkbox"/> [UC-0009] Ingresar prerrequisito a un curso <input type="checkbox"/> [UC-0011] Ingresar Nueva competencia	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando <i>un docente decida ingresar un nuevo curso mediante la opción que le brinda el sistema tal y como se ve en la figura 7 en la sección A.</i>	
Precondición	el <u>profesor</u> debe haber iniciado sesión	
Secuencia normal	Paso	Acción
	1	Si se ha seleccionado opción de nuevo curso numeral A , el sistema re direcciona a una nueva página para ingresar los datos básicos del curso figura 7 numeral B .

	2	El actor <u>Profesor (ACT-0002)</u> rellena el formulario con los datos del curso en la figura 7 numeral B .
	3	Si todos los datos están correctos y completos, el actor <u>Profesor (ACT-0002)</u> selecciona enviar curso figura 7 numeral C .
	4	El sistema ingresa el curso al sistema y re direcciona al <u>Profesor</u> a su sitio figura 7 numeral C .
Postcondición	ingresar su prerrequisito	

9 Ingresar prerrequisito a un curso

Figura 49. Ingresar prerrequisito a un curso

Inicio

Curriculo.Adaptativo

Usoario: Profesor1
Cerrar Sesión

Nombre Del Curso Curso1 [ingresar prerrequisito para este curso](#) (A)

Nombre Prerrequisito (B)

Nueva Pregunta

Pregunta 1	pregunta 1	
Respuestas	respuesta 1	Comenta
	respuesta 2	<input type="radio"/>
Seleccione la respuesta correcta	respuesta 3	<input type="radio"/>
	respuesta 4	<input type="radio"/>

Nombre Del Curso (C) Curso1

UC-0009	Ingresar prerrequisito a un curso	
Autores	<ul style="list-style-type: none"> • <u>Edier Edison Arteaga Diaz</u> 	
Dependencias	<input type="checkbox"/> <u>[UC-0010] Dar de alta/baja un Curso</u>	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando <i>el curso ya ha sido agregado y necesite su respectivo prerrequisito para ser publicado</i>	
Precondición	existir el curso	
Secuencia normal	Paso	Acción
	1	Si se ha seleccionado la opción ingresar prerrequisito tal

		y como muestra la figura 8 en su sección A , el sistema re direcciona al <u>profesor</u> a la página en donde se piden los datos del prerrequisito figura 8 numeral B .
	2	El actor <u>Profesor (ACT-0002)</u> rellena el formulario con los datos básicos del prerrequisito numeral B de la figura anterior.
	3	Si todos los datos están correctos y completos, el actor <u>Profesor (ACT-0002)</u> selecciona la opción enviar prerrequisito tal y donde indica la figura 8 en el numeral B .
	4	El sistema ingresa el prerrequisito al sistema y a su respectivo curso y re direcciona al <u>Profesor</u> a su sitio figura 8 numeral C , se evidencia que el curso ya puede ser dado de alta.
Postcondición		completar datos del curso

10 Dar de alta/baja un curso

Figura 50. Dar de alta/baja un curso

Logo: UNIVERSIDAD DE NARIÑO

Curso Adaptativo

Usuario: Profesor1
Cerrar Sesión

Nombre Del Curso

Curso1 (A)

Nombre Del Curso (B)

Curso1

UC-0010	Dar de alta/baja un curso	
Autores	• <u>Edier Edison Arteaga Diaz</u>	
Dependencias	<input type="checkbox"/> <u>[UC-0006] Registrar un Estudiante a un curso</u>	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando <i>el profesor decide dar de alta o baja a un curso en particular como se indica en la figura 9 en su sección A.</i>	
Precondición	de existir el prerequisite del curso	
Secuencia normal	Paso	Acción
	1	Si el <u>profesor</u> decide dar de alta/baja el curso como se ve en el numeral A y B de la figura 9, el sistema cambia el estado del curso para que este pueda o no ser publicado para los estudiantes
Postcondición	los estudiantes puedan verlo ,registrarlo, matricularlo	

11 Ingresar nueva competencia

Figura 51. Ingresar nueva competencia

Inicio Contenido General - Homee Usuario: Profesor1 Tercer Sesión

Nombre Del Curso **Curso1** **A** Dar De ALTA

Nombre De la Unidad **unidad1** **B**
unidad2

Esta Unidad no posee competencias aun! **Ingresar Competencia** **C**

Argumentativa **D**
Cuestionario
Cuestionario
Taller

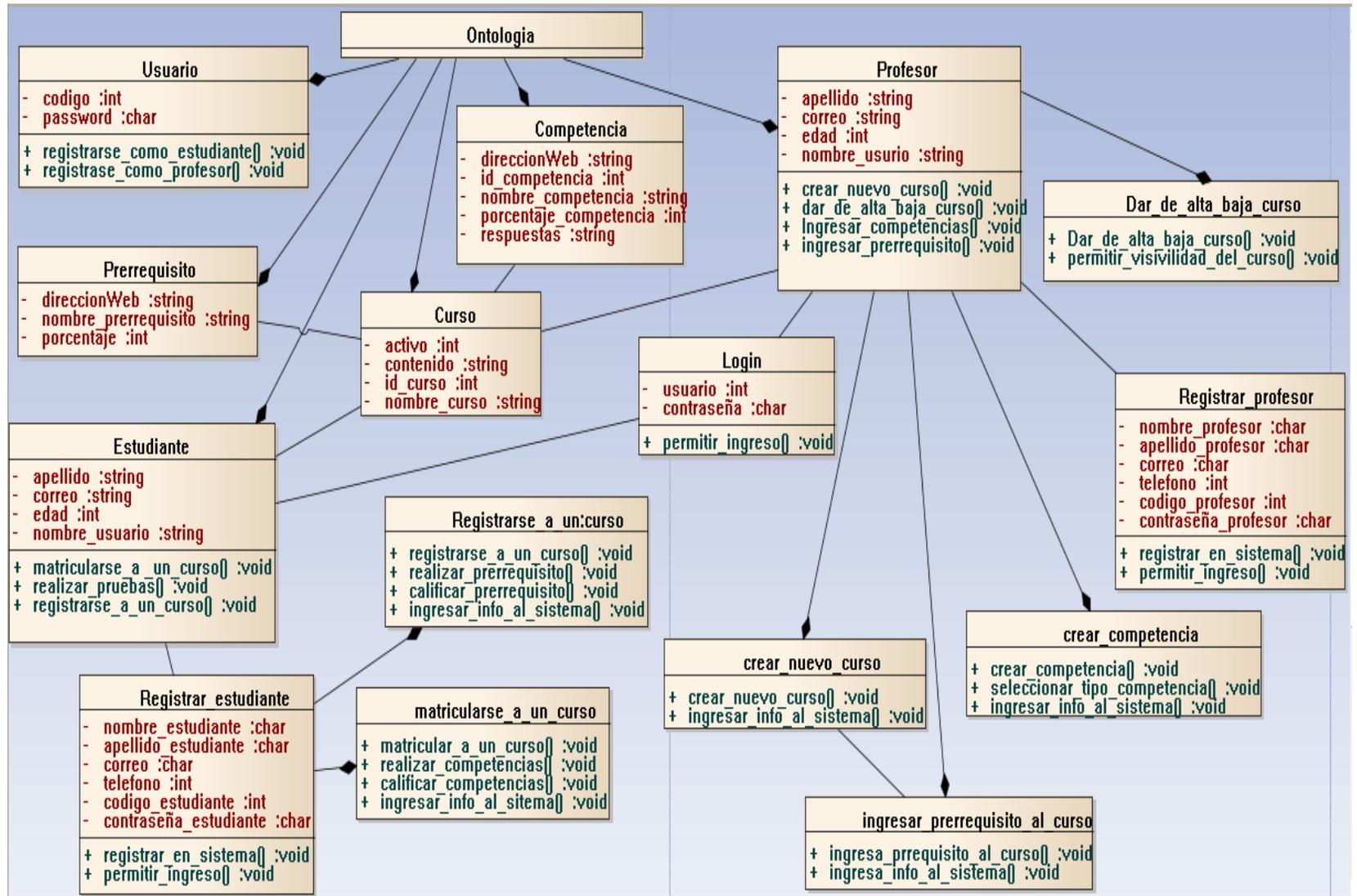
Nombre Competencia **competencia** **Enviar Competencia** **E**
% De Calificación 30

Pregunta 1	pregunta competencia 1	
Respuestas	respuesta competencia 1	Correcta
Seleccione la respuesta correcta	respuesta competencia 2	<input type="radio"/>
	respuesta competencia 3	<input type="radio"/>
	respuesta competencia 4	<input type="radio"/>

Nombre De la Competencia **- competencia** **Ingresar Competencia** **F**

UC-0011	Ingresar nueva competencia	
Autores	<ul style="list-style-type: none"> • <u>Edier Edison Arteaga Diaz</u> 	
Dependencias	Ninguno	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando <i>un profesor decide complementar un curso adicionando una competencias a un curso como se observa en la figura anterior</i>	
Precondición	el curso debe existir	
Secuencia normal	Paso	Acción
	1	El actor <u>Profesor (ACT-0002)</u> <i>ingresa al curso tal y como se ve en el numeral A.</i>
	2	El sistema <i>re direcciona al profesor al contenido del curso mostrando sus unidades en este caso solo dos: numeral B.</i>
	3	El actor <u>Profesor (ACT-0002)</u> <i>selecciona del contenido la unidad a la cual va a ingresar su competencia en el numeral B de la figura 10.</i>
	4	El sistema <i>re direcciona a la página que posee el contenido de la unidad como muestra el numeral C.</i>
	5	El actor <u>Profesor (ACT-0002)</u> <i>selecciona ingresar nueva competencia en el numeral C de la anterior imagen.</i>
	6	El sistema <i>re direcciona a una página en donde se piden los datos de esta mediante un formulario, aquí se pueden seleccionar tipos de competencias por defecto esta la argumentativa y tipo cuestionarios como muestra el numeral D.</i>
	7	El actor <u>Profesor (ACT-0002)</u> <i>rellena el formulario con los datos necesarios</i>
	8	Si todos los datos están correctos y completos, el actor <u>Profesor (ACT-0002)</u> <i>selecciona la opción enviar competencia en el numeral E.</i>
9	El sistema <i>ingresa la competencia al sistema y a la respectiva a unidad y re direcciona al profesor al contenido de la unidad por si este decide ingresar otra competencia como se observa en la figura 10 en el numeral F.</i>	
Postcondición	los estudiantes la realicen	

Figura 52. Diagrama de clases



5. CURRÍCULO ADAPTATIVO

El desarrollo del prototipo proporciona un sencillo y flexible acceso a algunas de las necesidades para los profesores, al mismo tiempo que se ayuda a la adecuación de su implementación usando tecnologías de la información.

Se ha diseñado una aplicación variada para evaluar las competencias del estudiante, en este caso al cursar la materia Estructuras de Información, partiendo de información previamente analizada.

Este contexto existe en diferentes aplicaciones, actualmente la diferencia que esta aplicación muestra radica en la realización de la mayoría de procesos y el uso de una ontología de información y como se mencionó anteriormente sirve de muchas maneras. Esta logra el cumplimiento de todos estos procesos de una manera muy natural y gracias a las ontologías que brindan un sin fin de relaciones entre los recursos de esta aplicación

Siendo el objetivo de este trabajo la creación de un prototipo que genere un currículo adaptado al estudiante de Ingeniería de Sistemas que va a cursar la materia estructuras de información, el primer paso es analizar el dominio por esto es muy importante el uso de una ontología que limita el trabajo con los recursos y conceptos utilizados.

5.1 ONTOLOGÍA

La ontología demuestra su tecnología clave para el acceso efectivo a los datos o información que posee, permite superar algunos de los problemas de la búsqueda basada en textos, relacionando y agrupando elementos según propiedades complejas. Gracias a esto lo más conveniente para la construcción de la aplicación es también la creación de una ontología adecuada para definir el dominio, que clasifique tanto los recursos utilizados, como los diferentes conceptos sobre los que trata cada uno de estos recursos.

5.1.1 Objetivos de la ontología. Para el diseño de la ontología es necesario aclarar qué se pretende lograr con ella. Dado que se quiere gestionar una gran parte de la aplicación, se estudian los requerimientos en base a unas supuestas preguntas que nacen en el desarrollo del prototipo para que la ontología cumpla con su labor. Si se sabe cuál es la pregunta, se sabrá cómo responder.

- ¿Qué información hay sobre estudiantes?
- ¿Qué información hay sobre profesores?
- ¿Qué información hay sobre cursos?

De estas preguntas se obtienen componentes que se utilizarán para construir la ontología. Se tiene datos como “Profesores”, “Estudiantes” e información relacionada a ellos como su “nombre_usuario”, “edades”, etc., y asociaciones a ellos como sus “cursos” y las relaciones como si lo cursan o lo dictan. Así se sabe cuál es el punto de partida y que se quiere obtener, y estudiar cuales son los elementos que se encuentran y qué relaciones hay entre ellos.

5.1.2 Recursos de la ontología. Esta aplicación centra el uso adecuado de la ontología en el manejo de sus recursos. Esto es así porque es el material base del que se dispone. A grandes rasgos, se puede definir los recursos como documentos digitales accesibles a través de sitios locales e incluso desde Internet por medio de URLs. El dónde se encuentra un determinado recurso no es un tema de gran importancia porque el objetivo con los recursos es lograr su ubicación cuando se los requiera, pudiendo indicarse tanto el qué es un usuario, estudiante, un archivo, etc, y cuál es su información contenida y de qué carácter es: descriptivo o un contenido.

5.1.3 Formato de un recurso. Una consideración importante sobre los recursos es su formato. El formato responde a la pregunta “¿Qué es?” Según su formato, se puede encontrar que si es un página web, un profesor, un estudiante, textos y otros tipos de elementos. Cada uno de estos tipos de documento viene identificado en la ontología mediante una jerarquía de clases, y cada instancia que se cree estará identificada por la URL del recurso.

Teniendo en cuenta lo anterior, cualquier documento digital disponible en la aplicación o en Internet puede ser referenciado con este modelo, este prototipo maneja poca clasificación de recursos a fin de mostrar el funcionamiento de la ontología, pero se puede lograr mucho alcance utilizando esta forma de trabajo, sería llevar el prototipo a una aplicación más compleja logrando hacerlo compatible para un sinfín de procesos que usen toda la red.

Es de vital importancia definir una buena taxonomía que organice los diferentes elementos de interés de la aplicación, creando una jerarquía de clases capaz de contener las instancias que se creen y que cumpla los objetivos facilitando la búsqueda de las mismas, esta taxonomía adecuada define conceptos como cursos, unidades, ítems, etc.

Como se describió anteriormente de una propiedad en la ontología, para relacionar los recursos entre sí, indicando sobre qué recurso (estudiante) trata otro determinado recurso (Curso). También la propiedad inversa en donde se puede indicar cuál es el recurso (Curso) que trata un determinado recurso (Estudiante).

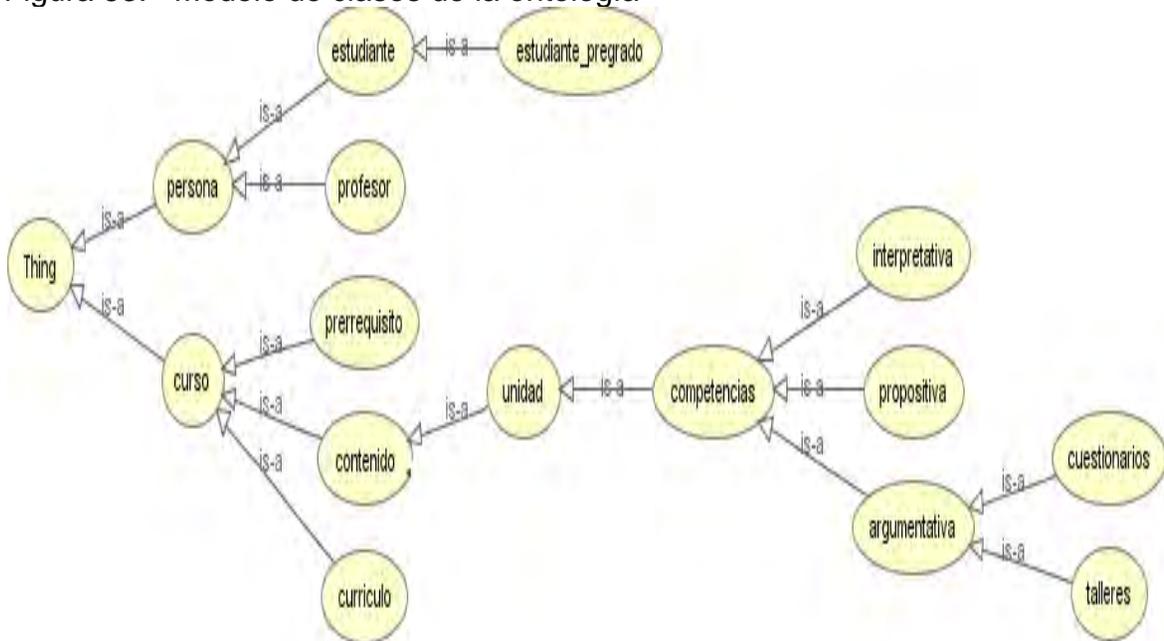
En el análisis realizado anteriormente se establecen los parámetros a seguir para la creación de la ontología y la taxonomía de los recursos en la aplicación de manera que todo forme una unidad.

De este modo, la ontología está compuesta por uno solo documento que se adapta las necesidades de la aplicación. Posteriormente se profundiza en la estructura de este documento.

5.1.4 Ontología currículum adaptativo: CurrículoAdaptativo.owl. Para esta aplicación esta es la ontología que se creó el archivo .owl generado se encuentra incluido, su dominio consta de diferentes clases que describen conceptos del mundo real. Esta ontología define tipos de recursos estableciendo algunos subtipos muy comunes como subtipos de estos.

Estructura general de las clases: como la mayoría de las clases se parte de una clase raíz inicia el árbol que esta desenvuelve a continuación se extienden las clases Curso y Persona que se refieren a las clases más generales de la ontología y son el núcleo de la misma de la clase Curso se extiendes las clases Prerrequisito, Contenido y Currículo que se refieren respectivamente al prerrequisito o prueba que un estudiante debe realizar para mostrar que es apto, contenido del curso (datos básicos del curso), el currículum generado para cada estudiante después de haber aprobado las pruebas. Cabe resaltar que la clase Contenido posee a su vez la clase Unidades y esta es superclase de Competencias que es una clase muy importante en el desarrollo de la aplicación ya que aquí se contendrán las pruebas con las que el alumno demostrará sus aptitudes para que el sistema genere el currículum correspondiente como se ve en la siguiente figura.

Figura 53. Modelo de clases de la ontología



Como se observa en la figura anterior que muestra gráficamente la estructura de la ontología la Clase competencia se subdivide en clases Interpretativa, Propositiva y Argumentativa y esta ultima en Cuestionarios (las pruebas que el estudiante soluciona para avanzar lograr su currículo) y Talleres

5.2 REQUERIMIENTOS DE LA APLICACIÓN

Para la construcción de la aplicación que es el objetivo principal de este trabajo es necesario lograr la conexión de la información anterior, es decir, mostrar en funcionamiento la ontología creada, permitiendo buscar, importar exportar y navegar entre los diferentes recursos instanciados de esta ontología.

En la construcción de esta aplicación web los usuarios que interactúan han de poder gestionar todas las actividades posibles sin tener conocimiento alguno de la información que la ontología maneja de modo que puede pedir o manejar algún recurso sin saber que pertenecen a una clase, y el sistema ha de devolver un resultado perteneciente a esa clase o a una subclase de esta en la jerarquía. Por ejemplo, pedir información sobre Cursos, competencias, unidades y la aplicación ha de devolver los recursos respectivos.

5.3 ANÁLISIS DE FUNCIONALIDADES

A partir de los requerimientos indicados en el apartado anterior, se puede confeccionar la siguiente lista de funcionalidades.

1. La aplicación ha de acceder a la ontología creada para realizar múltiples búsquedas.
2. La aplicación realiza consultas sobre cursos, estudiantes, profesores, unidades, competencias, prerrequisitos, etc. Para trabajar con ellos posteriormente
3. El usuario puede registrarse y una vez registrado autenticarse para ingresar al sistema, siempre y cuando cumpla con parámetros válidos.
4. El usuario puede una vez registrado tomar un rol en el sistema (profesor o estudiante).
5. El profesor puede ingresar el curso al sistema en este caso “Estructuras de información” con sus respectivos datos y el sistema le brindará la opción de ingresar.
6. El Profesor debe ingresar el prerrequisito para este curso, que servirá de prueba para saber si el estudiante cursante es apto para cursarlo.

7. El profesor puede ingresar el número de competencias que sean necesarias para evaluar al estudiante y el sistema permitirá el acceso a los datos respectivos.
8. El profesor deberá dar de alta el curso para que esté disponible para ser cursado por un estudiante.
9. El estudiante puede registrarse a un curso para este caso “Estructuras de información” mientras esté disponible y esta acción se realizará una prueba.
10. El sistema calificará el prerrequisito que el estudiante realice cuando desarrolle la función anterior y permitirá avanzar en el proceso de matrícula si esta evaluación es positiva.
11. El estudiante puede matricularse a un curso en este caso “Estructuras de información” en esta función el sistema evalúa al estudiante mediante las competencias previamente incorporadas.
12. El sistema evalúa al estudiante en el proceso de la función anterior y dependiendo el resultado genera el currículo correspondiente.
13. El sistema empaquetará los archivos resultantes con el estándar IMS-CP.

Cabe resaltar que para lograr el cumplimiento de las anteriores funcionalidades del sistema se usa a la ontología como fuente de búsqueda de información además de la inferencia que las ontologías permiten.

No es necesario describir cada una de las anteriores funciones para la comprensión del funcionamiento y el trabajo que se realiza en esta aplicación, donde se evidencia el uso de la potencialidad de la ontología junto con la programación java en el espacio web y la utilización de Jena que en capítulos anteriores se describieron, pero sí se aclaran algunos de los procesos que la aplicación realiza y siempre maneja.

5.3.1 Funcionalidades importantes. Para manejar la ontología se usa Api de Netbeans y utilizando la librería Jena los requerimientos son:

- El manejo de archivos RDF, RDFS y OWL.
- Permite navegar desde un elemento hasta otros a partir de sus sentencias y relaciones

Además usa funciones como:

OntModel: modelo de Jena, contienen los datos de la ontología aplica para OWL además maneja los ya conocidos triples.

ResultSet: al igual que en java es un conjunto de resultados que se representan mediante de filas que se obtienen del documento .owl de la ontología.

QuerySolution: un modelo que recibe el primer resultado logrado por el ResultSet anterior.

RDFNode: obtiene el nodo respectivo de la solución obtenida en el resultado con el modelo QuerySolution.

Statement: Modela una sentencia de tipo Sujeto-predicado-objeto.

Resource: Indica un elemento de la ontología, así como un sujeto de una Statement.

Property: Modela una propiedad de la ontología como rdf:type o CurrículoAdaptativo:Pertenece_a.

Literal: Modela un valor literal utilizado como objeto de una Statement

CreateStatement: instancia en el modelo una nueva sentencia de tipo Sujeto-predicado-objeto.

Para poder manipular el archivo generado por la ontología se recurre siempre a la instrucción OntModel y la variable que se define permite esta gestión por ejemplo OntModel modelo:

Para escribir modelo.write(elemento especifico), eliminar modelo.remove(elemento especifico), adicionar modelo.add(elemento especifico), etc.

Otra funcionalidad importante que la aplicación maneja mediante la librería de Jena es la inferencia que aplica mediante un razonador. A continuación una breve explicación.

El lenguaje SPARQL es un lenguaje de consulta de grafos RDF en su sintaxis funciona similarmente al SQL de las bases de datos que se conocen hoy en día. La aplicación permite utilizar la forma de consulta SELECT que devuelve el resultado en forma tabular sobre RDF mostrando las vinculaciones encontradas de las variables y WHERE que filtra la consulta Ejemplo.

```
SELECT ?id_curso ?nombre_curso WHERE{?curso Pre:id_curso ?id_curso.  
?curso Pre:nombre_curso ?nombre_curso. ?estudiante pre:Cursa ?curso.  
?estudiante Pre:nombre_usuario "+usuario+"}
```

La sentencia anterior brinda como resultado todos los cursos que un estudiante está cursando en la aplicación y obtiene de esta consulta el ?id_curso(id del curso) y ?nombre_curso(nombre del curso) teniendo en cuenta que:

Pre: es como un prefijo o abreviación de la dirección de donde se va a buscar hace referencia a todo el contenido de la ontología es decir busca todos los cursos existentes en la ontología.

?Estudiante y ?Curso: hace referencia a las clases Estudiante y Curso en la ontología.

Id_curso, Nombre_curso: la identificación del curso.

Cursa: es la relación entre las clases Estudiante y Curso.

"+usuario+": hace referencia a un estudiante en particular.

En esta simple sentencia de búsqueda se observa como la inferencia hace su trabajo cuando solo con la relación Cursa se llega a identificar en la ontología el curso que un estudiante tiene matriculado y no hace falta nada más para lograr este resultado.

5.4 EMPAQUETAMIENTO ESTÁNDAR IMS-CP

Cuando se genera el currículo utilizando las funciones anteriores, además de la programación en la Api de java la aplicación empaqueta los archivos resultantes del proceso de generación del currículo para un estudiante, este proceso procede a clasificar adecuadamente estos archivos por medio del estándar IMS_CP que brinda la opción de catalogar la información en una forma ordenada. La jerarquía que se construye para este proceso identifica cada parte y sub parte de los archivos como lo son el archivo generado cuando el profesor ingresa un nuevo prerrequisito al curso y el archivo generado cuando se ingresan las competencias. Así quedan referenciadas en un archivo de extensión .xml que es el que indica como se llaman, su ubicación y qué tipo de archivo para el caso archivos de extensión .jsp, un modelo se muestra a continuación, a través de un ejemplo (Figura 54).

Figura 54. Archivo xml de empaquetado IMS_CP

```
▼<manifest xmlns="http://www.imsglobal.org/xsd/imscp_v1p1"
xmlns:imsmd="http://www.imsglobal.org/xsd/imsmd_v1p2"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.imsglobal.org/xsd/imscp_v1p1 ../xsds/imscp_v1p2.xsd
http://www.imsglobal.org/xsd/imsmd_v1p2 http://www.imsglobal.org/xsd/imsmd_v1p2p4.xsd "
identifier="ManifestPrototipoCurriculoAdaptativo" version="IMS CP 1.2">
▼<metadata>
  <schema>IMS Content</schema>
  <schemaversion>1.2</schemaversion>
  ▼<imsmd:lom>
    ▼<imsmd:general>
      ▼<imsmd:title>
        <imsmd:langstring xml:lang="en-US">IMS Content Packaging - Curriculo
        Adaptativo</imsmd:langstring>
      </imsmd:title>
    </imsmd:general>
  </imsmd:lom>
</metadata>
▼<organizations default="TOC1">
  ▼<organization identifier="TOC1" structure="hierarchical">
    <title>default</title>
    ▼<item identifier="ITEM1" identifierref="RESOURCE1">
      <title>Lesson 1</title>
    </item>
  </organization>
</organizations>
▼<resources>
  ▼<resource identifier="RESOURCE1" type="webcontent" href="lesson1.htm">
    <file href="lesson1.htm"/>
  </resource>
</resources>
</manifest>
```

El anterior es un tipo modelo o formato en la especificación IMS-CP existen infinidad de tipos de formatos para esta aplicación no es necesaria la adopción de un modelo más complejo.

5.5 ESTRUCTURA DE CAPAS

A grandes rasgos la información anterior es la encargada junto con la programación en el Api de java de lograr el manejo y funcionamiento de la aplicación web a continuación la estructura por capas que resulta de todo este trabajo.

Figura 55. Estructura de capas del sistema



En resumen esta estructura (Figura 55) permite combinar la web semántica con un sistema de gestión de información referente a un curso.

Esta aplicación permite construir cuestionarios con datos de usuario profesor que incorporan información detallada para el estudiante y quedan disponibles por medio del servicio web.

Se maneja una estrategia muy simple para evaluar las competencias en principio usando cuestionarios que constan de cuatro preguntas con una única respuesta que se describen y muestran mediante una página web a medida que el estudiante responda a las preguntas que se encuentran en una determinada prueba y a la vez está en una unidad del contenido del curso, mientras la prueba sea superada el estudiante podrá continuar con las pruebas posteriores si existen, el sistema evaluará una a una las pruebas realizadas por el estudiante. Para esta evaluación se tendrá en cuenta que el porcentaje de las respuestas correctas de la prueba se igual o mayor a un 60%. Una vez logrado este resultado se puede decir

que el estudiante es apto o no, una vez culminadas las pruebas y dependiendo el desempeño y conocimiento del alumno el sistema generara el currículo para el estudiante (el currículo seria completo para un estudiante que no logro aprobar la primera prueba).

Como resultado final la aplicación muestra al usuario estudiante un pantallazo del contenido de su currículo en un listado de las unidades que contiene, acompañado del nivel que obtuvo en cada una por medio de la evaluación de las competencias (Figura 56^a y Figura 56^b):

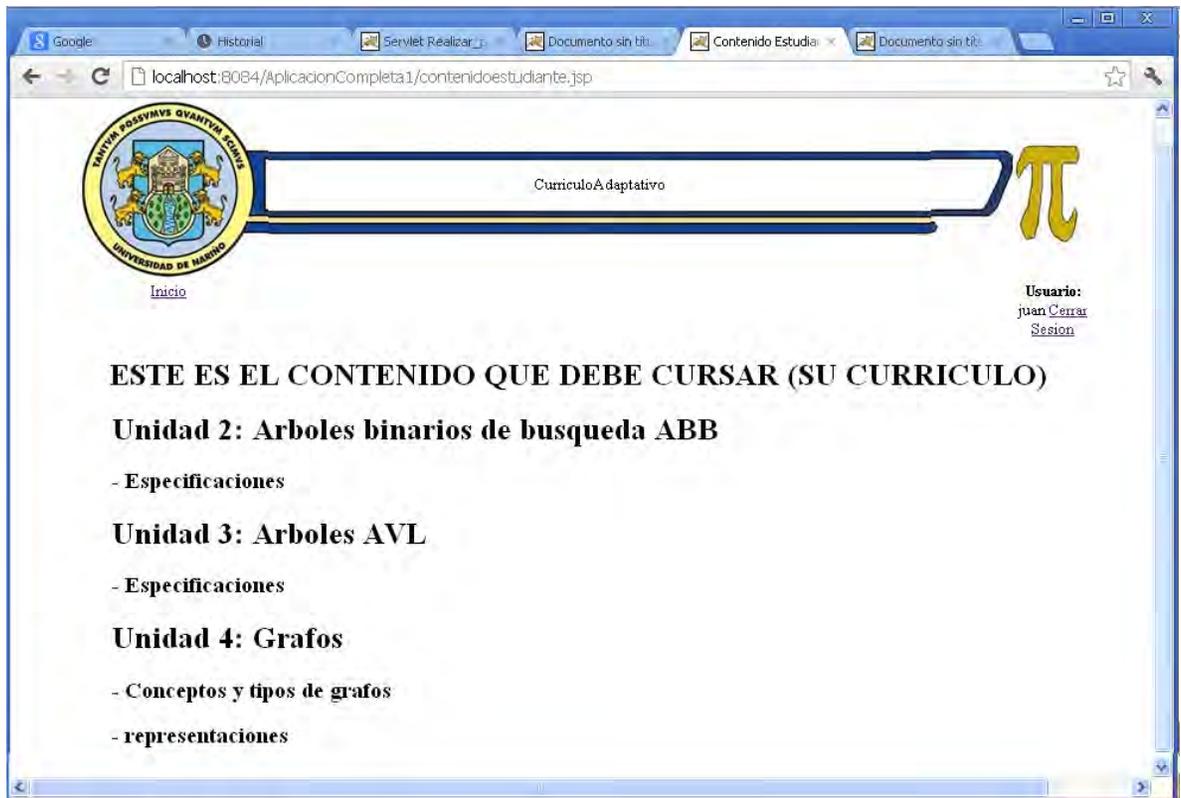
Figura 56a. Pantallazos finales de la aplicación



Frente a cada unidad el nivel de facilidad con la que el estudiante debería cursar el curso.

Para después pulsar en el botón “Generar mi Currículo” y así obtener las unidades y los temas a ver respectivamente.

Figura 56b. Pantallazos finales de la aplicación



6. TRABAJOS FUTUROS

La construcción de este prototipo muestra como se combinan diferentes tecnologías para obtener una herramienta que ayuda a optimizar la educación en línea, no obstante es un principio que muchos tendrán en cuenta a la hora de complementar trabajos relacionados, la información aquí contenida sirve de base a muchos trabajos futuros que utilicen tanto como web semántica en la construcción de herramientas similares como las solas ontologías en la construcción de aplicaciones diferentes.

Muchas de las mejoras que surgirían en este proyecto serían:

- Actualizar el prototipo para aplicarlo a un sin fin de materias.
- Creación de nuevos procesos e interfaces más complejas con el objetivo de acoplarlo a otras web similares.
- Permitir la exportación de la información en diferentes formatos.
- Complementar una base de datos, accesible, interactiva de todos los usuarios posibles.
- Ampliar su uso para que este realice carga de datos multimedia en la generación de las pruebas.
- En cuanto al trabajo con la ontología se podría mejorar el motor de consultas y así ofrecer a los usuarios búsquedas avanzadas que interactúen con web similares.

7. CONCLUSIONES

- Las ontologías y la propuesta de diferentes estándares para la representación de datos, ofrece a e-Learning la posibilidad de solucionar los problemas de comunicación entre tutores, estudiantes y sistemas de aprendizaje TIC, ya que los nuevos enfoques pedagógicos son más interactivos y permiten una mejor identificación de las temáticas y herramientas que requiere alguien que desea capacitarse a través de e-Learning.
- La eficiencia de una estrategia de e_Learning, está determinada en gran parte por las características de los cursos implementados. Una ontología basada en competencias permite identificar las necesidades reales de un estudiante, de manera que se puedan crear de manera sistemática cada una de las actividades de aprendizaje, garantizando una estructuración que permita a los aprendices la interacción directa y continua con los diferentes materiales de estudio, incluyendo procesos evaluativos para determinar el impacto sobre los procesos de construcción del conocimiento.
- Las ontologías son una fuente poderosa de información y herramienta que llevada de manera óptima producen buenos resultados, además facilita el orden y el manejo de los metadatos que fueron necesarios para la construcción de este prototipo.
- Esta herramienta puede generar un archivo de salida con el que es fácil trabajar. Este archivo mantiene un formato de metadatos y permite gestionar información que en él existe y así lograr un resultado de adaptación de contenidos a la necesidad de un estudiante mediante el desarrollo de cuestionarios sencillos y definiendo interfaces sencillas pero con la posibilidad de que se incrementen sus funciones hacia un futuro, junto con estudio de algunas técnicas de adaptación usadas en sistemas de evaluación.

8. RECOMENDACIONES

Posteriormente al desarrollo del proyecto surgen diferentes recomendaciones que puedan complementar una versión futura o proyectos similares:

- Enfatizar cuidadosamente el contenido de este proyecto para así definir con claridad todos los procesos a realizar.
- Este proyecto es un prototipo, por ende algunos procesos están en construcción y no contienen complejidad aun así logran ser idea clara para su posterior complementación, una de ellas puede ser que se amplíe su funcionamiento para todas las asignaturas del Departamento de Sistemas de la Universidad de Nariño.
- Optimizar o cambiar la interfaz manejada con el fin de que sea adaptable al usuario y así permita la mejor administración de la información.

BIBLIOGRAFIA

- [1] Estándar para Metadatos de Objetos Educativos. IEEE el 12 de Junio de 2002.
- [2] IMS Meta-data Best Practice Guide for IEEE 1484.12.1-2002 Standard for Learning Object Metadata, Version 1.3 Final Specification, 2006
- [3] IMS Content Packaging Specification Primer. Version 1.2 Public Draft v2.0, 2007
- [4] Mecanismos semánticos orientados a la flexibilidad de los repositorios para objetos de aprendizaje.
<http://biblioteca.universia.net/ficha.do?id=38082259>
- [5] IMS GLC:Learning Design Specification.
<http://www.imsglobal.org/learningdesign/>
- [6] RED. Revista de Educación a Distancia. Mapas conceptuales y objetos de aprendizaje. Murcia, España. 2006.
<http://www.um.es/ead/red/M2/leonel21.pdf>
- [7] David Griffiths, Josep Blat, Rocío Garcia y Sergio Sayago. La aportación de IMS Learning Design a la creación de recursos pedagógicos reutilizables. Universitat Pompeu Fabra.
- [8] Utilización de estructuras de aprendizajes IMS-LD en la enseñanza de la programación.
<http://www.revistavoxingenium.edu.co/?p=67>
- [9] José R. Hilerá1, David Palomar. Modelado de procesos de enseñanza-aprendizaje reutilizables con XML, UML e IMS-LD. Universidad de Alcalá, Departamento de Ciencias de la Computación, Alcalá de Henares, España. Universidad Carlos III, Departamento de Informática, Madrid, España
- [10] IMS Simple Sequencing Information and Behavior Model. Version 1.0 Final Specification, 2003
- [11] ¿Que es el estándar SCORM?
<http://scorm.com/>
- [12] Alejandro Karpicius. Educación Virtual y lo que vendrá.
http://karpicius.freelflux.net/files/educacion_virtual_lo_que_vendra.pdf

- [13]. Néstor D. Duque M., Demetrio A. Ovalle, Jovani A. Jiménez B. Modelo Adaptativo para Cursos Virtuales basado en Técnicas de Planificación Inteligente. Universidad Nacional de Colombia sede Medellín. 2007
- [14] P.C.D Un asistente en el desarrollo de Cursos a Distancia para Internet.
<http://cumincades.scix.net/data/works/att/a65b.content.pdf>
- [15] María Paula González. Uns – Dcic – Didáctica De La Computación. Universidad nacional del sur. Buenos Aires. Argentina. 2009
- [16] Jesús Romo Uriarte, Manuel Benito Gomez. E-Learning: perspectivas de las plataformas que lo soportan.
http://www.uv.es/ticape/docs/lcms_castellano.pdf
- [17] Néstor Darío Duque Méndez, Demetrio Ovalle Carranza. Modelo Inteligente Genérico Para Adaptatividad De Cursos Virtuales. Universidad Nacional de Colombia.
- [18] Santiago Blanco Suárez. Objetos y Repositorios de Aprendizaje.
<http://www.educared.org/global/educared/>
- [19] Alonso Tejada Zabaleta. Diseño Curricular Basado En Competencias Para Formar Profesionales Integrales. Universidad del Valle. Cali – Colombia. Julio 2006
- [20] Liber, O., Corley, L.: Learning Technology Standards - Where's The Pedagogy?. In A Méndez-Vilas et al. (eds.): Proceedings 2nd International Conference on Multimedia and Information & Communication Technologies in Education, Consejería de Educación, Ciencia y Tecnología de la Junta de Extremadura (2003) 855-859.
- [21] Eduardo Hernández . Estándares y Especificaciones de E-learning: Ordenando el Desorden.
http://www.hugocarrion.com/index_archivos/Docs/A_elearning.pdf
- [22] Adriana J. Berlanga Flores¹, Francisco J. García Peñalvo. Introducción a los Estándares y Especificaciones para Ambientes e-learning. Universidad de Salamanca, España.
- [23] Amadeo Sosa Santillán. Apuntes Para Un Curriculum Basado En Competencias. Una Mirada Desde VYGOTSKI.
- [24] Gustavo Hawes Barrios. QBC: El curriculum Basado en Competencias. Instituto de investigación y desarrollo educacional. Universidad de Talca. Chile.

[25] Diseño curricular Basado en competencias.
http://educacion.ucv.cl/prontus_formacion/site/artic/20061220/asocfile/ASOCFILE120061220151232.pdf

[26] Association for Computing Machinery ACM, IEEE Computer Society. Computer Science Curriculum 2008: An Interim Revision of CS 2001. Report from the Interim Review Task. December 2008.

[27] Introducción al Razonamiento Sobre Ontologías, Prof. Iván J. Flores Vitelli, Universidad Central de Venezuela Facultad de Ciencias Escuela de Computación, Lecturas en Ciencias de la Computación.

ANEXOS

INSTALACIÓN

Este prototipo esta generado en un sitio web por ende no se creó un tipo de instalador pero no es difícil su instalación para ello se deben tener en cuenta los siguientes aspectos:

Requerimientos de la aplicación. Se necesita obligatoriamente un servidor de aplicaciones java. En este caso se diseño para correr normalmente sobre Apache Tomcat 7.0.22 es aconsejable que sea esta la versión de uso.

El servidor debe estar instalado correctamente es decir poseer una máquina virtual Java para su normal funcionamiento. Se utilizo en el desarrollo de esta aplicación la plataforma JDK 1.7 de Netbeans.

El archivo manejado es de extensión “.owl” y es cargado desde su sitio local y puede ser cargado desde internet en un servicio web por ende se recomienda poseer internet.

Requerimientos:

- Máquina virtual Java con soporte para la versión 1.7
- Servidor de aplicaciones Tomcat 7.0.22.
- Acceso a Internet

Opcionales:

- Aplicación Tomcat Manager para facilitar la instalación a partir de .WAR

Instalación a partir del .WAR. El documento de extensión “.war” contiene toda la aplicación para ser instalado en un servidor Tomcat, para instalarlo solo hay que gestionar este documento desde la aplicación Tomcat Manager la cual se encargará de desplegarlo.

Instalación a partir de la información compresada en un archivo. Este documento posee toda la información que se necesita para correr el servicio utilizando la plataforma **Netbeans** se debe extraer todo el contenido en la carpeta de aplicaciones de la plataforma en este caso y como ejemplo: C:\Documents and Settings\EDIER1\Mis documentos\NetBeansProjects

Después se deben corregir todos los conflictos que se generan al abrir el proyecto (referencias a carpetas erróneas, activación de servidor web, etc.), si se a instalado todo a la perfección y no se presentan más inconvenientes solo se necesita correr la aplicación y probar su funcionamiento.

Interfaces. Como se puede ver a continuación (Figura 57) es la primera interfaz que se genera al correr la aplicación, donde se identifica el índice del sitio.

Figura 57. Interfaz principal



Esta interfaz significa el índice del sitio no posee grandes contenidos ya que solo es un prototipo y algunas de las opciones están deshabilitadas excepto:

TuSitio: En esta opción se re direcciona al estudiante o profesor a su lugar en la aplicación web de no estar Autenticados se re direccionaría al Login del sitio como se muestra en la siguiente figura.

Contactos: Direcciona a una interfaz que no tiene funcionamiento y no es importante en la aplicación en este momento.

Programas: Deshabilitada.

Facultades: Deshabilitada.

Figura 58. Interfaz iniciar sesión



En caso de no tener un Usuario y una contraseña la interfaz brinda la opción de registro, este registro como se muestra en las siguientes Figuras (59 y 60) no es para cualquier usuario, sino para un estudiante que debe poseer un código de estudiante y una contraseña que demuestran su pertenencia a la universidad.

El Login anterior se valida desde una base de datos normal pero el nombre de usuario manejado es el que toma un papel importante en la aplicación ya que la base de datos en principio sirve solo para el Login y algunas otras básicas funciones lo demás lo realiza la ontología creada para este sistema.

A continuación se muestran los dos tipos de registro uno para el estudiante y el otro para el Profesor. Como lo dije anteriormente aquí la ontología es la que juega el papel de base de datos en ella se registran la mayoría de los datos cuando se ingresan usuarios y demás como se observa posteriormente.

Después de los dos tipos de registro están las interfaces de contenidos: una para el contenido del Estudiante y la otra para el contenido del docente, estas interfaces son muy importantes debido a la partición de la aplicación y el funcionamiento de la misma desde dos actores diferentes Figura 55 y figura 56.

Figura 59. Interfaz registrar estudiante

Principal

localhost:8084/AplicacionCompleta1/registrar_usuario.jsp

Curriculo Adaptativo π

TuSitio

Estudiante

Indice
Contactos
Programas
Facultades

Datos Basicos Del Estudiante	
Nombre Completos	<input type="text"/>
Apellidos	<input type="text"/>
Correo Electronico	<input type="text"/>
Telefono (opcional)	<input type="text"/>
Nombre de Usuario	<input type="text"/>
Codigo Estudiantil	<input type="text"/>
Password	<input type="text"/>
<input type="button" value="Enviar"/>	

Figura 60. Interfaz registrar profesor

Principal

localhost:8084/AplicacionCompleta1/registrar_usuario.jsp

Curriculo Adaptativo π

TuSitio

Docente

Indice
Contactos
Programas
Facultades

Datos Basicos Del Docente	
Nombre Completos	<input type="text"/>
Apellidos	<input type="text"/>
Correo Electronico	<input type="text"/>
Telefono (opcional)	<input type="text"/>
Nombre de Usuario	<input type="text"/>
Codigo Profesor	<input type="text"/>
Password	<input type="text"/>
<input type="button" value="Enviar"/>	

Figura 61. Interfaz contenido estudiante



Como se mencionó anteriormente esta interfaz es muy importante aquí el estudiante tiene la opción de gestionar los cursos existentes en el sistema que es para lo cual la aplicación fue construida:

Inicio: direcciona al estudiante al índice.

Cerrar Sesión: Cierre de sesión en el sistema.

Registrarse: En esta opción el sistema direcciona al estudiante a una única prueba que consta de un cuestionario que debe realizarse para ver si este la cumple y así tiene opción de continuar con el curso en cuestión, si el estudiante no logra realizar la prueba con éxito el curso el curso se verá en adelante como No Matriculado y no se permitirá ninguna opción.

Matricularse: Esta opción al igual que la anterior direcciona al estudiante pero no solo a una prueba sino a las que el curso posea, es decir, se le evaluarán diferentes competencias par a medir la capacidad estudiantil respecto al curso con esto generarle el currículo correspondiente, si el estudiante no lograra asar ninguna prueba, de todas maneras quedara matriculado al curso con su respectivo currículo generado.

Figura 62. Interfaz contenido profesor



Esta interfaz al igual que la anterior es importante porque se gestiona el contenido del Profesor así:

Contenido General: lleva al Profesor al sitio que muestra la imagen anterior siempre.

Nuevo Curso: El sistema direcciona al Profesor a otra interfaz como se ve en una figura posterior (Figura 58) para que este ingrese un nuevo curso el docente es libre de ingresar tanto curso quiera ya que la aplicación se lo permite, en un futuro solo debería poder ingresar los necesariamente posibles.

Ingresar Prerrequisito Para Este Curso: El sistema direcciona al estudiante a otra interfaz para que este gestione los datos del prerrequisito correspondiente al curso como se muestra en la figura siguiente.

Dar De Alta/Baja: Estas dos opciones como su nombre lo dicen cumplen con dar la opción de que el curso sea visto o no por un estudiante.

Las demás funciones vistas en esta interfaz cumplen con lo que en ellas dicen como lo son Cerrar Sesión e Inicio.

Es posible ingresar al curso y a su contenido para seguir con el normal procedimiento de ingreso de datos del curso como lo es el ingreso de una nueva competencia como se ve en la figura 65.

Figura 63. Interfaz ingresar prerequisite a un nuevo curso

Contenido Docente

localhost:8084/AplicacionCompleta1/contenidoprofesor.jsp

Curriculo Adaptativo π

[Inicio](#) Contenido General Homee

Usuario: Profesor1
[Cerrar](#)
[Sesion](#)

Nombre Prerequisite

Pregunta 1	<input type="text"/>
Respuestas	<input type="text"/> <input type="radio"/>
Seleccione la respuesta correcta	<input type="text"/> <input type="radio"/>
	<input type="text"/> <input type="radio"/>

Figura 64. Interfaz ingresar nuevo curso

Nuevo Curso

localhost:8084/AplicacionCompleta1/registrar_curso.jsp

Curriculo Adaptativo π

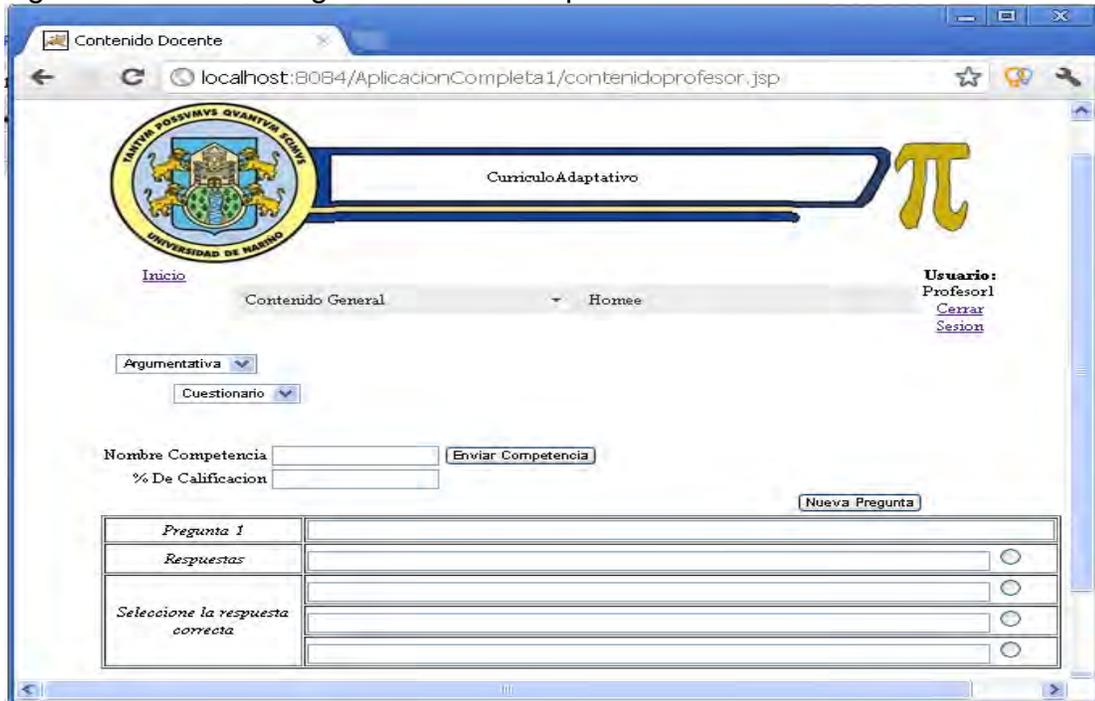
[Inicio](#) Contenido General Homee

Nombre Curso
 ID

Ingrese el contenido del Curso

Unidad 1:	Item 1:	SubItem 1:1	<input type="button" value="+"/>	<input type="button" value="+"/>	<input type="button" value="Enviar Unidad"/>
-----------	---------	-------------	----------------------------------	----------------------------------	--

Figura 65. Interfaz ingresar nueva competencia



Para ingresar una nueva competencia se debe navegar en el curso en principio al ingresar al curso se listan las unidades que el curso posee posteriormente se ingresa a cualquier unidad es esta en la disposición del docente una vez en su contenido el sistema brinda la opción de ingresar la competencia y en adelante se llenan los datos que muestra la figura anterior:

Interpretativa: La imagen muestra esta opción por defecto pero también es posible seleccionar una competencia propositiva recomendable dejar por defecto la opción argumentativa ya que en principio funciona completamente.

Cuestionario: la otra opción es "Taller" pero como mencione anteriormente es mejor trabajar en las opciones por defecto ya que la aplicación es solo un prototipo y funciona bien mientras no se cambien las opciones todas las demás serán habilitadas en un futuro si se sigue con la construcción completa de la aplicación.

Las demás opciones son evidentes a su descripción y basta con seguir el normal procedimiento para la culminación del ingreso de la competencia.