



UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA
La Universidad Católica de Loja

ÁREA TÉCNICA

TÍTULO DE INGENIERO EN INFORMÁTICA

Construcción de un plugin Moodle para la clasificación temática de recursos Web

TRABAJO DE TITULACIÓN.

AUTOR: Balseca Cevallos, José Vladimiro.

DIRECTOR: Chicaiza Espinosa, Janneth Alexandra, Ing.

CENTRO UNIVERSITARIO QUITO

2017



Esta versión digital, ha sido acreditada bajo la licencia Creative Commons 4.0, CC BY-NY-SA: Reconocimiento-No comercial-Compartir igual; la cual permite copiar, distribuir y comunicar públicamente la obra, mientras se reconozca la autoría original, no se utilice con fines comerciales y se permiten obras derivadas, siempre que mantenga la misma licencia al ser divulgada. <http://creativecommons.org/licenses/by-nc-sa/4.0/deed.es>

Septiembre, 2017

APROBACIÓN DEL DIRECTOR DEL TRABAJO DE TITULACIÓN

Ingeniera.

Janneth Alexandra Chicaiza Espinosa.

DOCENTE DE TITULACIÓN

De mi consideración:

Que el presente trabajo, denominado: **“Construcción de un plugin Moodle para la clasificación temática de recursos Web”** realizado por el profesional en formación: **José Vladimiro Balseca Cevallos**; cumple con los requisitos establecidos en las normas generales para la Graduación en la Universidad Técnica Particular de Loja, tanto en el aspecto de forma como de contenido, por lo cual me permito autorizar su presentación para los fines pertinentes.

Loja, septiembre de 2017

f).....

DECLARACIÓN DE AUTORÍA Y CESIÓN DE DERECHOS

Yo José Vladimiro Balseca Cevallos, declaro ser autor del presente trabajo de Titulación: **“Construcción de un plugin Moodle para la clasificación temática de recursos Web”** de la Titulación de Ingeniero en Sistemas Informáticos y Computación, siendo la Ing. Janneth Alexandra Chicaiza Espinosa directora del presente trabajo; y eximo expresamente a la Universidad Técnica Particular de Loja y a sus representantes legales de posibles reclamos o acciones legales. Además, certifico que las ideas, conceptos, procedimientos y resultados vertidos en el presente trabajo investigativo, son de mi exclusiva responsabilidad.

Adicionalmente declaro conocer y aceptar la disposición del Art. 88 del Estatuto Orgánico de la Universidad Técnica Particular de Loja que en su parte pertinente textualmente dice: “Forman parte del patrimonio de la Universidad la propiedad intelectual de investigaciones, trabajos científicos o técnicos y tesis de grado o trabajos de titulación que se realicen con el apoyo financiero, académico o institucional (operativo) de la Universidad”.

f).....

Autor: José Vladimiro Balseca Cevallos.

Cédula: 1710429059

DEDICATORIA

A mi madre Hermiña Cevallos, quien con su ayuda y consejo constante me ha formado como persona y me alienta a seguir siempre adelante.

AGRADECIMIENTO

A la Universidad Técnica Particular de Loja donde me estoy formando, y que me ha permitido desarrollar y encontrar nuevos valores en el campo espiritual y científico, que con aquellos me han permitido ser una mejor persona y profesional.

A los docentes de la Escuela de Ciencias de la Computación de la modalidad abierta y a distancia que me orientaron e impartieron sus conocimientos para mi preparación.

A la Ing. Janneth Chicaiza Espinosa; directora de la presente tesis, que bajo su dirección y que me orientó en la investigación, ejecución y desarrollo del presente trabajo de tesis.

Un agradecimiento muy especial a la Lcda. Lidia Villacís, de la Secretaria de Titulación Informática, que con su apoyo y constante ayuda me permitió resolver cualquier inconveniente durante estos años.

ÍNDICE DE CONTENIDOS

APROBACIÓN DEL DIRECTOR DEL TRABAJO DE TITULACIÓN.....	i
DECLARACIÓN DE AUTORÍA Y CESIÓN DE DERECHOS.....	ii
DEDICATORIA.....	iii
AGRADECIMIENTO.....	iv
RESUMEN.....	1
ABSTRACT.....	2
INTRODUCCIÓN.....	3
CAPÍTULO I.....	5
ESTADO DEL ARTE ESTUDIO DE TECNOLOGÍAS PARA ANÁLISIS Y CLASIFICACIÓN DEL CONOCIMIENTO.....	5
1.1. Resumen.....	6
1.2. Sistemas de organización del conocimiento.....	6
1.2.1. Introducción.....	6
1.2.2. Características comunes de los sistemas de organización del conocimiento.....	7
1.2.3. Tipos de Sistemas de Organización del Conocimiento.....	7
1.2.3.1. Listas de términos.....	8
1.2.3.2. Clasificaciones y categorías.....	9
1.2.3.3. Listas de Relaciones.....	9
1.2.4. Sistemas tradicionales de clasificación de conocimiento.....	10
1.2.5. Clasificación temática de recursos.....	17
1.2.6. Vocabulario SKOS.....	19
1.2.7. OWL.....	22
1.2.8. SPARQL.....	22
CAPITULO II.....	28
ENTORNOS VIRTUALES DE APRENDIZAJE, MOODLE Y EL PROCESO DE CREACION DE PLUGINS.....	28
2.1. Resumen.....	29
2.2. Entornos virtuales de aprendizaje.....	29
2.3. Moodle.....	31
2.4. Proceso de construcción de plugins en Moodle.....	31
CAPITULO III.....	34
DESARROLLO DEL PROCESO DE PREPARACIÓN DEL PLUGIN PARA LA CLASIFICACIÓN TEMÁTICA.....	34

3.1.	Resumen.....	35
3.2.	Proceso de clasificación temática de recursos web.....	35
3.3.	DBPedia.....	36
3.4.	DBPedia Spotlight.....	37
3.5.	Modelo de desarrollo del plugin.....	38
3.6.	Modelo de implementación del plugin en Moodle.....	41
CAPITULO IV.....		43
CONSTRUCCIÓN DEL PLUGIN MOODLE PARA LA CLASIFICACIÓN TEMÁTICA.....		43
4.1.	Resumen.....	44
4.2.	Entorno de desarrollo del plugin Moodle.....	44
4.3.	Diseño de la base de conocimiento.....	44
4.4.	Estructura de los procesos desarrollados para la clasificación temática.....	47
4.4.1.	Lectura del documento o texto de entrada.....	48
4.4.2.	Extracción del contenido y anotación semántica.....	48
4.4.3.	Proceso de anotación con las entradas de la base de conocimiento local..	50
4.4.4.	Proceso de clasificación temática.....	51
4.4.5.	Adaptación de la salida y presentación al usuario final.....	52
4.5.	Construcción del plugin Moodle.....	54
CAPITULO V.....		68
VALIDACIÓN DEL PLUGIN MOODLE PARA LA CLASIFICACIÓN TEMÁTICA.....		68
5.1.	Resumen.....	69
5.2.	Pruebas de funcionalidad.....	69
5.2.1.	Comportamiento cuando no se encuentra disponible la base de conocimiento local.....	69
5.2.2.	Comportamiento cuando se intenta realizar una clasificación con un contenido diferente al idioma inglés.....	70
5.2.3.	Comportamiento cuando se intenta realizar una clasificación con un contenido diferente al de la ciencia de los ordenadores.....	72
5.3.	Validación de los resultados proporcionados por el clasificador.....	73
5.3.1.	Comportamiento del clasificador al variar el nivel de confianza (desambiguación).....	73
5.4.	Comportamiento del clasificador con diferentes textos analizados.....	76
5.5.	Comportamiento del clasificador respecto a los tiempos de respuesta con textos de diferentes tamaños.....	79
5.6.	Conclusiones sobre los resultados obtenidos.....	80

CONCLUSIONES	82
RECOMENDACIONES Y TRABAJOS FUTUROS	83
BIBLIOGRAFÍA.....	84
ANEXOS.....	87
ANEXO 1. Creación de módulos (plugins) en Moodle.	88
ANEXO 2. Código para mostrar la interfaz de usuario para la lectura del documento o texto de entrada.	104
ANEXO 3. Código para realizar el proceso de extracción de contenido y anotación semántica.....	110
ANEXO 4. Código para la anotación con las entradas en la base de conocimiento local.	117
ANEXO 5. Código para realizar el proceso de borrado y preparación para una nueva clasificación.....	118
ANEXO 6. Código para contrastar las entradas de la base de conocimiento local.	119
ANEXO 7. Código para realizar la clasificación temática.....	120
ANEXO 8. Código para definir el detalle y versión del plugin desarrollado Moodle.....	121
ANEXO 9. Código para definir la disponibilidad del plugin Moodle.	122
ANEXO 10. Código para definir la presentación del plugin dentro del entorno Moodle.	122
ANEXO 11. Código para identificar las librerías utilizadas en la creación del plugin Moodle.	124
ANEXO 12. Código para especificar las entradas del idioma para el plugin Moodle... ..	124
ANEXO 13. Código para definir las clases utilizadas en el plugin Moodle.	125
ANEXO 14. Código para definir las entradas a la base de datos de Moodle asociadas al plugin creado.....	127
ANEXO 15. Código para definir las entradas de la instalación del plugin Moodle.....	128
ANEXO 16. Código para definir las entradas para el proceso de desinstalación del plugin Moodle.....	129
ANEXO 17. Código para definir las entradas para el proceso de actualización del plugin Moodle.	130
ANEXO 18. Código para definir el proceso de respaldo y restauración del plugin Moodle.	132

RESUMEN

La búsqueda de información en Internet es una de las tareas básicas que realizan los usuarios en la red. Debido a la gran cantidad y variedad de recursos digitales e información, los motores de búsqueda no siempre son efectivos para seleccionar los contenidos más relevantes o potencialmente útiles para los usuarios. En grandes colecciones de documentos, como en el caso de las bibliotecas físicas universitarias, generalmente, un conjunto de expertos realiza la clasificación temática de los materiales según las disciplinas de conocimiento definidas por un tesoro. La organización de acuerdo con tópicos o temas es fundamental para que el usuario pueda localizar de forma más ágil la información requerida. En el presente trabajo, se ha implementado un plugin Moodle para realizar la tarea de clasificación temática, de forma automática, utilizando una base de conocimiento y un diseño previamente establecidos.

El plugin implementado podrá ser utilizado por los usuarios de plataformas educativas como Moodle para identificar las disciplinas de conocimiento relacionadas a algún recurso de interés y en función de esta información, determinar si un material académico puede ser de utilidad.

PALABRAS CLAVES: Entidad, extracción, anotación, desambigüación, DBPedia, SKOS, tesoro UNESCO, clasificación temática.

ABSTRACT

Searching for information on the Internet is one of the basic tasks performed by users in the network. Due to the large amount and variety of digital resources and information, search engines are not always effective in selecting the most relevant or potentially useful contents for users. In large collections of documents, as in the case of university physical libraries, generally a group of experts makes the thematic classification of materials according to the disciplines of knowledge defined by a thesaurus. The organization according to themes or topics is fundamental so that the user can locate in a more agile the required information. In the present work, a Moodle plugin has been implemented to perform the thematic classification task, automatically, using a previously established knowledge base and design.

The implemented plugin can be used by users of educational platforms such as Moodle to identify the disciplines of knowledge related to some resource of interest and depending on this information, determine if an academic material can be useful or not.

KEYWORDS: Entity, extraction, annotation, disambiguation, semantic relation, DBPedia, SKOS, thesaurus

INTRODUCCIÓN

Cuando se realiza la consulta o se visualiza el contenido referente a un tema se puede encontrar que una o varias palabras pueden relacionar nuevas fuentes de consulta, a estas se las denomina palabras claves, es decir, con estas se puede encontrar un conjunto de elementos comunes orientados hacia otras fuentes o ramas del conocimiento humano, así como una definición más amplia de cada una de aquellas.

La posibilidad que brinda este proyecto es analizar el texto de un tema en particular proporcionado por un usuario y al final del mismo proporcionarle un conjunto de tópicos asociados al tema; de esta manera, una persona podrá aprovechar de mejor manera una fuente de consulta al poder orientar su estudio o investigación a las diferentes áreas asociadas enriqueciendo su trabajo y orientándolo de una mejor manera.

Con este proyecto se pretende desarrollar un componente (plugin de Moodle) basado en un servicios web que haga el análisis de un recurso encontrado en una página (texto de un tema) y determine su clasificación temática, para este propósito se realizará una tarea de anotación semántica y luego de este proceso permita obtener los temas (tópicos) asociados a las ramas del conocimiento humano, de tal manera que se pueda utilizar como una guía de referencia para ampliar el conocimiento del texto que se está analizando por parte de un usuario. Al mismo tiempo este servicio podrá ser adoptado por otras plataformas que lo hace independiente de Moodle.

Se ha iniciado con el Estado del Arte que corresponde al capítulo I, donde se hace una revisión de los conceptos relacionados al conocimiento, el estudio de tecnologías para el análisis y clasificación del mismo.

En el capítulo II se realiza un análisis de los Entornos Virtuales de Aprendizaje y de manera particular a Moodle hacia donde se orientará la construcción del plugin final, así como los pasos requeridos por esta plataforma para incorporar el desarrollo de un plugin.

El capítulo III corresponde el desarrollo del proceso de preparación del plugin para clasificar el texto fuente en temas (tópicos), para esto se indican los pasos que se han considerado

para el procesado de texto; se explica cómo se ha diseñado el componente de anotación semántica del texto fuente utilizando un componente o librería de DBPedia denominado Spotlight, para ello se utiliza una base de conocimiento preestablecida. Luego que se han encontrado las palabras claves o anotaciones asociados en DBPedia se lo somete a un clasificador de temas (tópicos) basado en los tesauros de UNESCO para finalmente presentar la sugerencia de tópicos al usuario final.

El capítulo IV se centra en la construcción del plugin de Moodle el cual integra todos los servicios creados y combina los resultados generados por cada servicio individual de tal manera que se generen recomendaciones más precisas y útiles de tópicos asociados a un tema de consulta.

Por último, el capítulo V se valida y contrasta los resultados obtenidos por el plugin haciendo un seguimiento minucioso en cada una de las fases hasta llegar a las recomendaciones de tópicos propuestos al usuario final, haciendo énfasis en los tiempos de respuesta y certeza de los resultados.

CAPÍTULO I

ESTADO DEL ARTE ESTUDIO DE TECNOLOGÍAS PARA ANÁLISIS Y CLASIFICACIÓN DEL CONOCIMIENTO

1.1. Resumen.

En este capítulo se realiza una revisión de todos los conceptos asociados a los sistemas de organización del conocimiento, sus características y tipos, y los métodos de clasificación de los mismos. Dado que el conocimiento es la manera de cómo se percibe la información y los hechos por parte de las personas por medio de la experiencia o la educación, es decir la captación teórica o práctica de un asunto o concepto.

Una vez que se tiene organizado el conocimiento en recursos de conceptos se puede realizar una clasificación temática de los mismos para un aprovechamiento de aquellos de manera eficaz y eficiente. Los recursos que son fuentes de información que con el advenimiento de la web se halla disperso en él, de una manera fría y sin una estandarización, es por el ello que nace el concepto de la Web Semántica, cuyo objetivo es dar un sentido a esta información de tal manera que un usuario pueda encontrar de manera rápida y ordenada las respuestas a sus preguntas. A la Web Semántica se le ha dotado de un conjunto de elementos que le permite operar y razonar en la web, estos elementos se los define y estudia en este capítulo.

1.2. Sistemas de organización del conocimiento.

1.2.1. Introducción.

El término sistemas de organización del conocimiento se refiere a la organización de la información y la gestión del conocimiento de todo tipo¹. El sistema de organización del conocimiento permite organizar información a nivel general (como libros en una biblioteca), información más detallada de varias fuentes (como nombres geográficos y nombres personales). Incluye esquemas menos tradicionales, como redes semánticas y ontologías (representación de conocimiento lingüístico en la que los conceptos y sus interrelaciones mediante un gráfico de árbol). Los sistemas de organización del conocimiento se encuentran muy a menudo en bibliotecas como museos y archivos ya que por el uso de aquellos es posible organizar la información.

Los sistemas de organización del conocimiento se utilizan para organizar la información con el propósito de recuperación y para manejar una colección. Un KOS (esta sigla define a Knowledge Organization Systems) sirve como puente entre la necesidad de información del

¹ Definición establecida por Sistemas de Organización del Grupo de Trabajo en Red de Conocimiento en su reunión inicial en la Conferencia de Bibliotecas Digitales O98 ACM en Pittsburgh, Pennsylvania.

usuario y el material de la colección. Con este, el usuario debe ser capaz de identificar un objeto de interés sin conocimiento previo de su existencia. Ya sea a través de navegación o búsqueda directa, a través de temas en una página Web o un motor de búsqueda del sitio, el KOS guía al usuario a través de un proceso de descubrimiento de nuevos conceptos. [1]

1.2.2. Características comunes de los sistemas de organización del conocimiento.

Los seres humanos desde muy temprana edad tienden a organizar el conocimiento para hacer frente a un mundo cambiante comparando los objetos y experiencias, categorizando todo con lo que estamos familiarizados. El énfasis en el desarrollo de los KOS se puede ver en los escritos de nuestros primeros filósofos, muchos de los cuales siguen influyendo en nuestra visión del mundo. Por ejemplo, el esfuerzo de Aristóteles por clasificar el conocimiento en grupos (como la física, la política o la psicología) se refleja en nuestro lenguaje, nuestra educación y nuestra ciencia.

No existe un sistema de clasificación de conocimiento único pues depende de los puntos de vista pueden tener las personas y/o las organizaciones. Por lo tanto, vivimos en un mundo de múltiples y variantes formas de organizar el conocimiento.

A pesar de su diversidad, los KOS tienen las siguientes características comunes:

- El KOS impone una visión particular del mundo sobre una colección y los elementos en ella.
- La misma entidad se puede caracterizar de diferentes maneras, dependiendo del KOS que se use.
- Debe existir suficiente coincidencia entre el concepto expresado en un KOS y el objeto real al que se refiere ese concepto de que una persona conocedora podría aplicar el sistema con fiabilidad razonable. Del mismo modo, una persona que busca material relevante usando un KOS debe ser capaz de conectar su concepto con su representación en el sistema.

1.2.3. Tipos de Sistemas de Organización del Conocimiento.

Una revisión de algunos sistemas típicos de organización del conocimiento muestra su alcance y aplicabilidad a una variedad de configuraciones de bibliotecas digitales. Si bien

hay definiciones específicas para muchos de estos KOS en la ciencia de la informática y la ciencia de la información, e incluso en los documentos de normas, hay debate sobre estas definiciones.

Los sistemas de organización del conocimiento se basan en características tales como estructura y complejidad, relaciones entre términos y función histórica. Estos se agrupan en tres categorías generales: listas de términos, clasificaciones y categorías, y listas de relaciones, que enfatizan las conexiones entre términos y conceptos [2]. Estas categorías las revisaremos a continuación.

1.2.3.1. Listas de términos.

Es un conjunto de definiciones de temas ordenados de una manera lógica dependiendo de su aplicación y se sub-clasifican en:

- **Archivos de autoridad.** Son listas de términos que se utilizan para controlar las variantes de nombre para una entidad o el valor de dominio para un campo en particular como por ejemplo nombres para países, individuos y organizaciones. Este tipo de KOS generalmente no incluye una organización profunda o una estructura compleja. La presentación puede ser alfabética u organizada por un esquema de clasificación. Se puede aplicar una jerarquía limitada para permitir una navegación sencilla.
- **Glosarios.** Un glosario es una lista de términos, por lo general con las definiciones. Los términos pueden ser de un tema específico o de un trabajo en particular.
- **Diccionarios.** Los diccionarios son listas alfabéticas de palabras y sus definiciones. Los diccionarios son de alcance más general que los glosarios. También pueden proporcionar información sobre el origen de una palabra, variantes (por deletreo y morfología) y múltiples significados entre disciplinas. Así mismo un diccionario también puede proporcionar las palabras relacionadas a la inicial.
- **Diccionarios geográficos.** Un diccionario geográfico es una lista de nombres de lugares. Los nombres geográficos han aparecido en Atlas o en libros asociados a la

geografía. Cada entrada también puede ser identificada por tipo de característica, como río, ciudad o escuela.

1.2.3.2. Clasificaciones y categorías.

Consiste básicamente en establecer las cualidades los temas o colecciones de acuerdo con un orden preestablecido, se sub-clasifican en:

- **Los encabezamientos de contenido.** Este tipo de esquema proporciona un conjunto de términos controlados para representar los asuntos de elementos de una colección. Su estructura es generalmente muy superficial y limitada.
- **Los sistemas de clasificación, taxonomías y esquemas de categorización.** Estos términos se usan indistintamente. Estos tipos de KOS proporcionan formas de separar entidades en "cubos" o niveles de tema amplios. Pueden proporcionar una disposición jerárquica de notación numérica o alfabética para representar temas amplios. Estos tipos de KOS pueden no seguir las reglas para la jerarquía requerida en el ANSI NISO Thesaurus Standard (Z39.19) (NISO 1998), y carecen de las relaciones explícitas presentadas en un tesoro. Ejemplos de esquemas de clasificación incluyen la Clasificación de Dewey (un sistema cerrado de 10 secciones numéricas con extensiones decimales), y la Clasificación Decimal Universal (basada en Dewey, pero ampliada para incluir aspectos particulares de un tema).

Las taxonomías se utilizan cada vez más en sistemas de diseño orientado a objetos y de gestión del conocimiento para indicar cualquier agrupación de objetos basada en una característica particular.

Los esquemas de categorización se utilizan para la representación del conocimiento lingüístico en la que los conceptos y sus interrelaciones se representan mediante un grafo.

1.2.3.3. Listas de Relaciones.

Permite asociar los temas con características comunes y asociarlos mediante representación de sus relaciones. Se sub-clasifican en:

- **Tesoros.** Son una lista de palabras o términos controlados empleados para representar conceptos. Se basan en conceptos y muestran las relaciones entre los

términos. Las relaciones comúnmente expresadas en un tesoro incluyen jerarquía, equivalencia (sinonimia), y asociación o relación. Estas relaciones se representan generalmente por la notación BT (término más amplio), NT (término más estrecho), SY (sinónimo), y RT (término asociativo o relacionado).

Existen normas para el desarrollo de tesauros monolingües (NISO 1998, ISO 1986) y tesauros multilingües (ISO 1985).

Muchos tesauros son grandes; pueden incluir más de 50.000 términos. La mayoría son desarrollados para una disciplina específica, un producto específico o familia de productos.

- **Las redes semánticas.** Son una forma de representación de conocimiento de tal modo que los conceptos y sus relaciones se representan mediante un grafo. Con la llegada de procesamiento del lenguaje natural, se han producido avances significativos en las redes semánticas. Estos KOS estructuran conceptos y términos no como jerarquías, sino como una red o una web. Los conceptos son pensados como nodos, y las relaciones se derivan de ellos.
- **Ontologías.** Son una definición formal de tipos, propiedades, y relaciones entre entidades que existen para un dominio de discusión en particular. Pueden representar relaciones complejas entre objetos e incluir las reglas y axiomas que faltan en las redes semánticas. Las ontologías que describen el conocimiento en un área específica a menudo están conectadas con los sistemas de minería de datos y la gestión del conocimiento.

1.2.4. Sistemas tradicionales de clasificación de conocimiento

Los KOS, tradicionalmente son clasificados con el uso de Tesauros, existen varias organizaciones que han definido sistemas de clasificación para mejorar la organización y extracción de documentos, los mayormente usados son:

- **Sistema de Clasificación Decimal Dewey (CDD).** Es un sistema que permite clasificar las obras de una biblioteca por medio de una nomenclatura numérica, fue desarrollado por Melvil Dewey; este sistema de clasificación es adoptado hasta la actualidad en la mayoría de bibliotecas[3].

Melvin Dewey identificó que al clasificar las colecciones de acuerdo con su diseño se tiene la garantía de que los números de clasificación asignados no tendrán que ser modificados en ningún momento así mismo el orden lógico de las colecciones permanecerá estable con el paso del tiempo, pues las obras con temas relacionados se mantendrán juntas. A este modo de organización se le conoce como locación relativa. Otro aspecto innovador es el uso de números decimales para representar los diferentes temas que componen el universo del conocimiento publicado, ya que a través de ellos es posible expresar la posición jerárquica de los temas.

La clasificación principal se detalla en 10 temas como se muestra a continuación (Edición 14 abreviada, del año 2008):

000 - Ciencias de la Computación, Información y Obras Generales.

100 - Filosofía y Psicología.

200 - Religión, Teología.

300 - Ciencias Sociales.

400 - Lenguas.

500 - Ciencias Básicas.

600 - Tecnología y Ciencias Aplicadas.

700 - Artes y recreación.

800 - Literatura.

900 - Historia y Geografía

Estas categorías en un modelo jerárquico abarcan diez nuevas sub categorías de diversos niveles; así mismo se puede ir añadiendo más niveles. Un ejemplo podemos observar:

800 - Literatura

880 - Literatura eslava

882 - Literatura rusa

Como se observa, cada subnivel de uno anterior es la especialidad del mismo, su notación está por completo en números arábigos, se podría también utilizar letras del alfabeto para distinguir algunas temáticas.

- **Sistema decimal universal de clasificación (CDU).** Los precedentes de la Clasificación Decimal Universal (CDU) se hallan en la Clasificación Decimal de Dewey (CDD), adaptada y modificada por los belgas Paul Otlet y Henry La Fontaine. La primera edición data del año 1905, su creación tenía como objetivo general servir en la normalización de la clasificación documental en el plano mundial.[4]

La CDU es una clasificación sistemática, similar a una enciclopedia, que contiene sucesiones de conceptos ordenados por relaciones esenciales, según principios de subordinación lógica; es un sistema mixto entre las clasificaciones enumerativas (por ejemplo, la CDD) y las facetadas (por ejemplo, la Colon Classification o Clasificación Facetada de Ranganathan), enumera el conocimiento en diez grandes grupos, combinándolos con subdivisiones auxiliares, comunes y especiales, para hacer posibles combinaciones de conceptos.

El sistema de notación de la CDU se construye mediante numeración arábica con valor decimal. Los grupos de tres cifras se separan mediante un punto, y la notación presenta una estructura jerárquica, de forma que cuanto mayor sea su longitud, más concreto y específico es el concepto que expresa. A cada número de la CDU le corresponde un concepto o grupo de conceptos de forma que puedan ser reflejados los documentos complejos y también aquellos que contienen varios conceptos.

Esta notación se puede observar en la figura 1, de acuerdo con el sitio web de CDU.

Universal Decimal Classification

summary

ARRIBA SIGNOS AUXILIARES 0 1 2 3 4 5 6 7 8 9

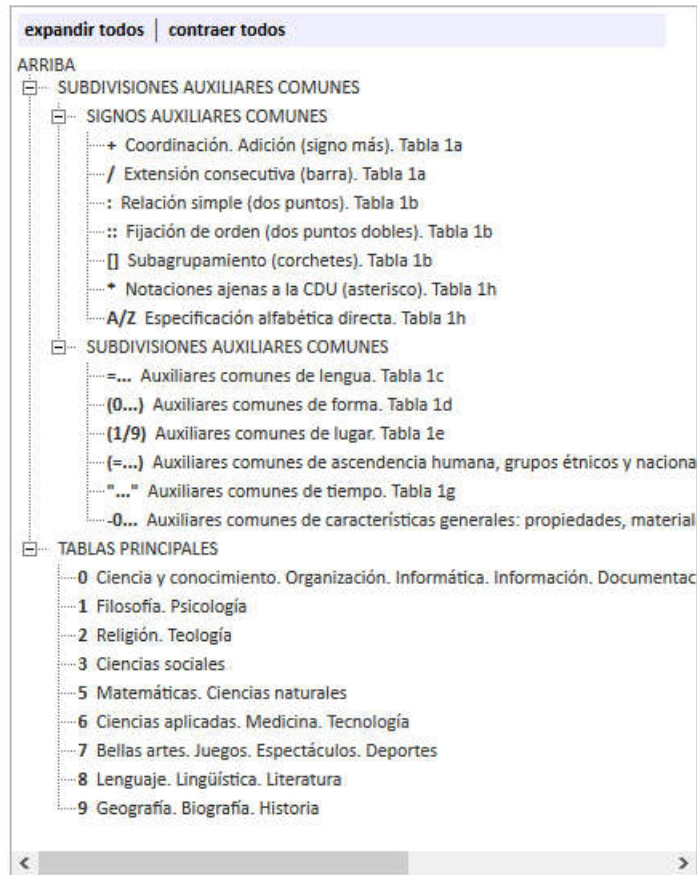


Figura 1. Sistema de clasificación universal

Fuente: <http://www.udcc.org/>

- **Nomenclatura UNESCO.** Se establece una clasificación de los diferentes campos de la Tecnología y la Ciencia, fue creado y desarrollado por la UNESCO, básicamente es un sistema de clasificación utilizado para proyectos de investigación y de tesis doctorales.[5]

Se define los niveles de clasificación de las disciplinas y sub-disciplinas, que son codificados y agrupados con dos, cuatro y seis dígitos.

- Campos (dos dígitos): Definen los campos del conocimiento principales de diferentes áreas.

- Disciplinas (cuatro dígitos): Se establece y se explica los grupos de ramas asociadas a los campos.
- Sub-disciplinas (seis dígitos): representan a las categorías que se hallan dentro de una disciplina.

Esta clasificación se muestra en la tabla 1.

Tabla 1: Clasificación de los campos en la nomenclatura de la UNESCO

Campo amplio	Campo específico	Campo detallado
11	Lógica	Categoría: Lógica
12	Matemáticas	Categoría: Matemática
21	Astronomía y Astrofísica	Categoría: Astronomía y Categoría: Astrofísica
22	Física	Categoría: Física
23	Química	Categoría: Química
24	Ciencias de la vida	Categoría: Ciencias de la vida
25	Ciencias de la Tierra y el Espacio	Categoría: Ciencias de la Tierra
31	Ciencias agronómicas	Categoría: Agronomía
32	Ciencias Médicas	Categoría: Medicina
33	Ciencias tecnológicas	Categoría: Tecnología
51	Antropología	Categoría: Antropología
52	Demografía	Categoría: Demografía
53	Ciencias Económicas	Categoría: Economía
54	Geografía	Categoría: Geografía
55	Historia	Categoría: Historia
56	Ciencias Jurídicas y Derecho	Categoría: Derecho
57	Lingüística	Categoría: Lingüística
58	Pedagogía	Categoría: Pedagogía

59	Ciencia Política	Categoría: Ciencia política
61	Psicología	Categoría: Psicología
62	Ciencias de las Artes y las Letras	
63	Sociología	Categoría: Sociología
71	Ética	Categoría: Ética
72	Filosofía	Categoría: Filosofía

Fuente: Tesista

- **Joint Academic Coding System.** El sistema de Sistema de Codificación Común Académico (JACS) es utilizado por la Agencia de Educación Superior de Estadística (HESA) y las Universidades y el Servicio de Admisiones de Colegios y Universidades (UCAS) en el Reino Unido para clasificar las materias académicas.[6]

Un código JACS para una sola materia consta de una letra y tres números. La letra representa la clasificación temática amplia y números subsiguientes representan más detalles, similar al sistema decimal Dewey. Por ejemplo, F representa las Ciencias Físicas, F300 Física, F330 Física Ambiental y F331 Física Atmosférica.

JACS se utiliza actualmente para codificar los temas de los cursos de educación superior y los módulos individuales en el Reino Unido, estos códigos se encuentran detallados en la Tabla 2.

Tabla 2: Sistema de clasificación JACS

Código	Grupo de sujetos	Subgrupos principales
A	Medicina y Odontología	Medicina A100 Preclínica, Odontología A200 preclínica, clínica A300 Medicina, Odontología Clínica A400
B	Temas relacionados con la medicina	B100 anatomía, fisiología y patología, B200 Farmacología, Toxicología y Farmacia, Medicina Complementaria B300, B400 Nutrición, B500 Oftalmología, B600 Aural y Ciencias orales, B700 Enfermería, Tecnología Médica B800
C	Ciencias biológicas	Biología C100, C200 Botánica, Zoología C300, C400 Genética, Microbiología C500, C600 Ciencias del Deporte, Psicología C800

D	Ciencias Veterinarias, Agricultura y disciplinas relacionadas	Medicina Pre-D100 Clínica Veterinaria, Agricultura D400, D500 Forestal, D600 Alimentos y Bebidas Estudios
F	Ciencias físicas	Química F100, F200 Ciencia de los Materiales, Física F300, F400 forense y arqueología, geología F600, F700 Ciencias Oceánicas, F800 Físicas y geográficas terrestres y Ciencias Ambientales, Geografía física F840
G	Ciencias Matemáticas e Informática	Matemáticas G100, G300 Estadística, Ciencias de la Computación G400, G600 Ingeniería de Software
H	ingeniería	H200 Civil, H300 Mecánica, Aeroespacial H400, H500 arquitectura naval, producción y fabricación H700, H800 Química
J	tecnologías	J200 Metalurgia, cerámicas y vidrios, J300, J400 Polímeros y textiles, materiales de J500 Tecnología
K	Arquitectura, Construcción y Planificación	Arquitectura K100, K200 K400 Edificio Planificación
L	estudios sociales	L100 Economía, Política L200, L300 Sociología, Política Social L400, L500 Trabajo Social, Antropología L600, L700 Geografía Humana y Social
M	ley	Ley M100 por áreas geográficas, Ley M200 por tema
N	Empresariales y Administrativas	N100 Estudios Empresariales, Gestión N200, N300 Finanzas, Contabilidad N400, N500 de Marketing
P	Comunicación y Documentación de masas	P300 Ciencias de la Información, Periodismo P500
Q	Lingüística, obras clásicas y temas relacionados	La lingüística Q100, Q500 Estudios Celtas
R	Idiomas europeos, literatura y temas relacionados	R100 Estudios francés, alemán Estudios R200, R300 Estudios Italianos, Estudios Hispánicos R400, R600 Estudios escandinavos, Rusia Estudios R700
T	Oriental, asiático, África, América y Australasia Lenguas, Literatura y temas relacionados	Estudios Chinos T100, T500 Estudios Africanos
V	Estudios históricos y filosóficos	Historia V100 por periodo, V200 Historia por zona, Historia del Arte V350, V400 Arqueología, V500 Filosofía, Teología y Estudios Religiosos V600

W	Arte Creativo y Diseño	W100 bellas artes, Diseño W200, W300 música, drama W400, W500 de baile, y Cinematics Fotografía W600, W700 manualidades, escritura creativa W800
X	Educación	X100 la formación de profesores
Y	no se utiliza	no se utiliza
Z	no se utiliza	no se utiliza

Fuente: Tesista

- **European Training Thesaurus.** Es un sistema creado de la derivación del entrenamiento profesional y asegura que los conceptos utilizados signifiquen lo mismo, el tesoro europeo de formación profesional pretende ser una herramienta de clasificación del conocimiento para todo lo que tienen que ver con la educación, la formación y el empleo. Incluye una selección de 1207 términos y conceptos que se utilizan con frecuencia en la literatura en el contexto de la educación y la formación europeas [7].

Cada término se presenta en 11 idiomas: danés, holandés, inglés, estonio, finlandés, francés, alemán, italiano, polaco, portugués y sueco. El ReferNet hizo una contribución particularmente importante para la determinación de las traducciones para las diferentes versiones lingüísticas del tesoro.

Cubre temas que incluyen las áreas de las necesidades de cualificación y de formación, el aprendizaje permanente y la calidad de la educación, las políticas y los sistemas de formación profesional, evaluación, reconocimiento y certificación de las cualificaciones y los resultados de aprendizaje y orientación profesional.

1.2.5. Clasificación temática de recursos.

La clasificación temática de recursos permite encontrar temas (tópicos) en colecciones de documentos, en otras palabras, es la extracción de la información de las bases de conocimiento de una manera eficiente y ágil en grandes volúmenes de datos en muchos dominios.

Cada dominio tiene su propia definición de un concepto el cual puede variar en múltiples dominios, para esto se hace necesario introducir una clasificación temática de los recursos, y puesto que la cantidad de información es abundante es necesario comprimir estos en unidades de conocimiento de manera útil y manejable [8].

Para esto se han definido varios métodos para la clasificación temática de recursos los cuales mencionaremos a continuación [9]:

- **El análisis semántico latente (LSA)** [Deerwester, 1990] utiliza una descomposición de valor singular de la matriz de una colección, formando un sub-espacio lineal reducido que captura las características más significativas de la colección.
- **El modelo Probabilístico LSA (PLSA)** [Hofmann, 1999], que es un modelo generativo de datos que puede proporcionar una sólida base estadística. En este modelo de mezcla estadística, cada palabra en un documento como una muestra de un modelo de mezcla, donde los componentes de la mezcla son variables aleatorias multinomiales que pueden ser vistas como representaciones de temas. Así, cada documento independiente está representado por una lista de proporciones de temas latentes, y cada uno de los temas se representa mezclando palabras de variables aleatorias multinomiales.
- **La asignación latente de Dirichlet (LDA)** [Blei, 2012] es una técnica de modelado de temas y la herramienta de modelado tópico más común actualmente en uso puede descubrir los temas ocultos en colecciones de documentos de las palabras que aparecen en aquellos
- **El modelo n-Gram (TNG)** [Wang, 2007], este modelo TNG descubre automáticamente y simultáneamente temas y extractos de tópicos con frases pertinentes. Se ha integrado perfectamente en el modelado del lenguaje basado en tareas extracción de información.
- **Etiquetado del modelo de tema** [Zhai, 2008], la distribución multinomial basada en palabras se utiliza para representar temas basados en la estadística, pero funciona menos bien al interpretar explícitamente la semántica de los temas. Normalmente las palabras con alta probabilidad de un tema tienden a sugerir el significado del tema, pero las palabras simples tienen los problemas de la polisemia (una palabra tiene varios significados) y la sinonimia. Así, la gente tiende a etiquetar los temas con frases semánticas. Al aplicar este modelo se realiza normalmente en dos pasos. En primer lugar, se genera un conjunto de frases candidatas, ya sea analizando la colección de texto o utilizando medidas estadísticas tales como la información mutua. En segundo lugar, estas frases candidatas se clasifican en función de una medida probabilística, que indica lo bien que una frase puede caracterizar un modelo de tema. Finalmente, algunas frases de alto rango se elegirían como etiquetas para

un modelo de tema. Las etiquetas seleccionadas pueden ser diversificadas eliminando la redundancia. Es el caso de DBPedia Spotlight.

1.2.6. Vocabulario SKOS.

Los vocabularios SKOS (Simple Knowledge Organization System), es una familia de lenguajes formales, que representan al conocimiento y que se encuentra definido por: tesauros, listas de encabezamientos de materia, taxonomías, esquemas de clasificación, folksonomías y otros vocabularios controlados.

SKOS tiene su origen en RDF (Resource Description Framework) y RDFS, y su objetivo principal es permitir la publicación fácil de vocabularios controlados y estructurados para la Web semántica.

La web semántica es una web que contiene un mayor significado respecto al conocimiento de cada elemento, de tal manera que un usuario haciendo uso del Internet puede encontrar respuesta a sus preguntas e inquietudes de una manera más rápida y dinámica partiendo de una base de conocimiento formal, ya que la información de un tema se halla interrelacionado, mejor definido y estructurado [10], con la cual posible compartir, analizar y transferir información de forma sencilla, de tal modo que los SKOS permiten la creación y publicación en la Web de los conceptos del conocimiento humano, realizar los vínculos hacia otros conceptos asociados e integrarlos de una manera eficiente.

Los SKOS permiten identificar los recursos conceptuales (conceptos) mediante URIs, etiquetarlos con literales de uno o varios idiomas, documentarlos con diversos tipos de notas, relacionarlos entre sí mediante estructuras jerárquicas o redes asociativas, y agregarlos a esquemas de conceptos, de este modo los conceptos pueden relacionarse semánticamente entre sí [11].

RDF, el Resources Description Framework es una estructura que permite representar la información en la Web. Permite realizar afirmaciones sobre recursos tanto cosas concretas como abstractas. Como ejemplos tenemos una empresa determinada, una persona, una página Web, un sentimiento, o un color, también son considerados recursos.

Las URIs, constituye una manera más simple de representar los conceptos con el uso de una URL, de tal modo que se puede de definir e identificar cosas, a diferencia de una URL que es una dirección para observar el contenido de un tema a manera de documento.

Los elementos integrales de SKOS son los siguientes [12]:

- Concept: es el elemento principal dentro de SKOS, ya que define que un concepto de un determinado recurso, ejemplo:

@prefix ex:.

ex:animales rdf:type skos:Concept.

- prefLabel, altLabel: son etiquetas que hacen referencia a las definiciones en lenguaje natural: "prefLabel" es una etiqueta preferida a ser mostrado de un concepto mientras que "altLabel" es una etiqueta alternativa utilizada; por ejemplo, para conceptos sinónimos tenemos:

ex:animales rdf:type skos:Concept ;

skos:prefLabel "animales" ;

skos:altLabel "criaturas".

En este ejemplo el concepto animal también puede asociarse a criaturas

- broader, narrower: con ellos se puede establecer que el significado de un concepto está asociado con otros conceptos del vocabulario; "broader", indica que un concepto es más amplio que otro ya que incluye el concepto definido, "narrower" es lo contrario de "broader" contiene un concepto que es parte de otro, ejemplo:

ex:animales rdf:type skos:Concept ;

skos:prefLabel "animales" ;

skos:narrower ex:mamiferos.

ex:mamiferos rdf:type skos:Concept ;

skos:prefLabel "mamíferos" ;

skos:broader ex:animales.

- related: indica que hay relación entre dos conceptos, así tenemos, por ejemplo:

ex:entomologia rdf:type skos:Concept ;

skos:prefLabel "entomología".

ex:aracnidos rdf:type skos:Concept;

skos:prefLabel "arácnidos";
skos:related ex:entomologia

- note: es una nota asociada al concepto, se puede establecer otras anotaciones con el uso de otros tipos, y tenemos: "scopeNote", "editorialNote" y "changeNote", que especifican el alcance y/o historia, anotaciones editoriales y cambios realizados, respectivamente, por ejemplo:

ex:frecuenciasMicroondas

skos:scopeNote "utilizado para frecuencias entre 1GHz to 300Ghz"@es.

ex:tomates

skos:changeNote "Movido desde 'frutas' a 'vegetales'".

- definition: es la definición del concepto, por ejemplo:

ex:documentacion rdf:type skos:Concept ;

skos:definition "el proceso de almacenar y extraer la información en todos los campos del conocimiento".

- ConceptScheme: los conceptos pueden utilizarse como elementos independientes; sin embargo, los conceptos forman parte de vocabularios organizados como tesauros o esquemas de clasificación. Estos esquemas pueden representarse con la clase "ConceptScheme", por ejemplo:

ex:tesauroAnimal rdf:type skos:ConceptScheme ;

dct:title "Tesauro Animal simple" ;

dct:creator ex:antoinelsaac.

ex:mamiferos rdf:type skos:Concept ;

skos:inScheme ex:tesauroAnimal.

ex:vaca rdf:type skos:Concept;

skos:broader ex:mamiferos ;

skos:inScheme ex:tesauroAnimal.

ex:pez rdf:type skos:Concept ;

skos:inScheme ex:tesauroAnimal.

1.2.7. OWL.

OWL es una forma de desarrollar temas o vocabularios específicos para asociar los recursos, de tal manera que OWL proporciona un lenguaje para definir ontologías organizadas y que se utilizan en diferentes sistemas, de tal modo que se puede representar y describir un área de conocimiento. Estos OWL son utilizadas por los usuarios, las aplicaciones y bases de datos que necesitan compartir información específica que incluyen las definiciones de los conceptos campo definido y sus relaciones [10].

1.2.8. SPARQL.

Como se ya se ha indicado en la Web Semántica los conceptos son representados utilizando el modelo de datos de RDF y OWL, los cuales están almacenados en bases de datos en forma de tripletas (t<sujeito> <predicado> <objeto>), RDF expresa una relación entre dos elementos el sujeto y el objeto mediante un predicado. Por ejemplo:

Libro -> Título -> Cien años de soledad

SPARQL (SPARQL Protocol and Query Language), es el lenguaje de consulta de la Web Semántica con el uso de las tripletas mencionadas anteriormente. SPARQL es similar lenguaje SQL (Structure Query Language) que se utiliza para realizar consultas en las bases de datos relacionales, SPARQL tiene su definición propia para realizar consultas en un conjunto de tripletas RDF. [13]

Por ejemplo, si tenemos un conjunto de tripletas como el siguiente:

```
<http://example.org/book/book1> <http://purl.org/dc/elements/1.1/title>  
  "Tutorial RDF".  
<http://example.org/book/book2> <http://purl.org/dc/elements/1.1/title>  
  "Tutorial SPARQL ".
```

La consulta:

```
SELECT ?title  
WHERE
```

```

{
  <http://example.org/book/book2>
  <http://purl.org/dc/elements/1.1/title>
  ?title .
}

```

mostraría:

```

title
"Tutorial SPARQL"

```

Las variables en SPARQL comienzan con un carácter "?" y pueden estar en cualquiera parte de la tripleta ya sea como sujeto, predicado y/u objeto dentro de un RDF. Las tripletas de la cláusula SELECT son iguales a las tripletas normales. SELECT devuelve una matriz de variables con los valores que satisfacen la consulta.

En SELECT es posible utilizar filtros y operadores que ayudan a realizar consultas complejas. Las bases de datos de tripletas ofrecen puntos de acceso por vía Web, las cuales permiten utilizar SPARQL los cuales reciben el nombre de endpoints, los cuales aceptan consultas y devuelve los resultados vía HTTP [14].

En cláusula SELECT se puede especificar el grafo RDF a ser consultado por medio de la palabra clave FROM. En el ejemplo siguiente, se muestran las páginas de acceso en la web de las personas conocidas ("foaf:knows") por Steve Jobs, en el grafo RDF localizado en <http://dig.csail.mit.edu/2008/webdav/timbl/foaf.rdf>.

```

PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX card: <http://www.w3.org/People/SteveJobs/card#>
SELECT ?homepage
FROM <http://dig.csail.mit.edu/2008/webdav/timbl/foaf.rdf>
WHERE {
  card:i foaf:knows ?known
  ?known foaf:homepage ?homepage . }

```

Resultado:


```
homepage
http://purl.org/net/tim/
http://www.stevebosniac.com/
...
```

Las consultas SPARQL devuelven como resultado un conjunto de tuplas (similares a líneas de una tabla), que puede mostrarse en varios formatos como son: HTML, XML, JSON, CSV, etc, el formato se especifica en la consulta, como ejemplo mostramos el resultado en formato JSON de la consulta anterior:

```
{ "head": { "link": [], "vars": ["homepage"] },
  "results": { "distinct": false, "ordered": true, "bindings": [
    { "homepage": { "type": "uri", "value": "http://purl.org/net/tim/" } },
    { "homepage": { "type": "uri", "value": "http://www.stevebosniac.com/" } },
    ... ] } }
```

SELECT acepta los comandos DESCRIBE, CONSTRUCT y ASK.

El comando DESCRIBE devuelve un grafo RDF simple donde los datos devueltos son triplas en las cuales el recurso de consulta aparece como sujeto u objeto, o bien también el recurso aparece solamente como sujeto, como ejemplo tenemos que el recurso (":SteveJobs") aparece como sujeto y como objeto.

```
PREFIX dbpedia-owl:
DESCRIBE ?stevejobs WHERE {
  ?timbl dbpedia-owl:alias "SteveJobs".
}
```

Resultado:

s	p	o
:SteveJobs	dc:description	"American computer scientist, best know as Apple CEO"

```

:SteveJobs      dbpedia2:occupation  :Computer_scientist
:SteveJobs      foaf:name           "Jobs, Steve"
:SteveJobs      foaf:surname        "Jobs"
:SteveJobs      dbpedia2:birthPlace  USA
:SteveJobs      dbpedia:ontology/employer  :Apple_Computer_INC
...

```

Construct, produce como resultado tripletas no encontradas en el store de tripletas que se está consultando. Un ejemplo observamos a continuación:

```

construct {
  <http://dbpedia.org/resource/SteveJobs> ?p ?o .
  ?s ?p2 <http://dbpedia.org/resource/SteveJobs> .
}
where {
  { <http://dbpedia.org/resource/SteveJobs> ?p ?o }
  union
  { ?s ?p2 <http://dbpedia.org/resource/SteveJobs> }
}

```

Resultado:

Prefix	Namespace IRI
dbo	http://dbpedia.org/ontology/
foaf	http://xmlns.com/foaf/0.1/
rdfs	http://www.w3.org/2000/01/rdf-schema#
rdf	http://www.w3.org/1999/02/22-rdf-syntax-ns#
n4	http://en.wikipedia.org/wiki/SteveJobs?oldid=
wikipedia-en	http://en.wikipedia.org/wiki/
prov	http://www.w3.org/ns/prov#
xsdh	http://www.w3.org/2001/XMLSchema#

dbr <http://dbpedia.org/resource/>

Subject Item

dbr:SteveJobs

rdfs:label

SteveJobs

prov:wasDerivedFrom

[n4:464371503](#)

foaf:isPrimaryTopicOf

[wikipedia-en:SteveJobs](#)

dbo:wikiPageID

33972732

dbo:wikiPageRevisionID

464371503

dbo:wikiPageRedirects

[dbr:Steve_Jobs](#)

Subject Item

wikipedia-en:SteveJobs

foaf:primaryTopic

[dbr:SteveJobs](#)

Ask, es utilizado para probar si un patrón de consulta tiene o no una solución, no devuelve resultados.

Con SPARQL se puede utilizar etiquetas para obtener los resultados de una manera conveniente y orientada al usuario, para esto se puede utilizar:

- ORDER – ordena el resultado.
- FILTER – aplica filtros sobre los valores buscados en los elementos de la tripleta.
- LIMIT – limita el número de líneas.

- DISTINCT – excluye las líneas duplicadas.
- OFFSET – realiza paginación.

SPARQL permite también el uso de operadores lógicos (“!”, “&&”, “||”), operadores matemáticos (“+”, “-”, “*”, “/”), operadores comparativos (“=”, “>”, “<”, ...), operadores de testeo (“isLiteral”, isURI”, ...) y funciones de manipulación de cadenas de caracteres (“STRLEN”, “SUBSTR”, “UCASE”, “CONCAT”, ...).

Se puede utilizar la guía en el link <https://www.w3.org/TR/rdf-sparql-query/> para manejar más a profundidad los conceptos de SPARQL.

CAPITULO II

ENTORNOS VIRTUALES DE APRENDIZAJE, MOODLE Y EL PROCESO DE CREACION DE PLUGINS

2.1. Resumen.

En este capítulo se realiza una revisión de todos los conceptos asociados los entornos virtuales de aprendizaje como la base para la enseñanza y aprendizaje a distancia utilizando herramientas automatizadas manejadas por un software y que estos recursos pueden ser utilizados en cualquier momento tanto por los profesores como por los estudiantes.

Así mismo se realiza un estudio de Moodle que es una plataforma utilizada en los EVA de libre distribución y muy popular en el medio de la enseñanza remota por su funcionalidad y características de escalabilidad. También se aborda el tema de construcción de plugins para esta plataforma con el objetivo de añadir nuevas funcionalidades que ayuden a mejorar la interactividad y uso de los diferentes usuarios del sistema.

2.2. Entornos virtuales de aprendizaje.

Un entorno virtual de aprendizaje (EVA en español, VLE Virtual Learning Environment en inglés) es un conjunto de herramientas de enseñanza y aprendizaje diseñadas para mejorar la experiencia de adquisición de conocimientos de un estudiante mediante el uso de computadores y/o dispositivos móviles, y el manejo de los contenidos de un curso mediante una interfaz de software a través de Internet o intranet.

Los componentes principales de un paquete EVA tomando los lineamientos establecidos por un modelo pedagógico de enseñanza son los siguientes [15]:

- Mapeo curricular de un curso (dividir el elemento de estudio en secciones de tal manera que se puedan asignar y evaluar).
- Seguimiento de los estudiantes.
- Apoyo en línea para el profesor y el estudiante.
- Fuentes de comunicación (correo electrónico, foros de discusión, chat, publicaciones web).
- Enlaces a recursos externos del curso.

Para que los usuarios puedan hacer uso de un EVA se les asigna un ID de profesor o un ID de estudiante. El profesor ve lo que los estudiantes ven, pero el profesor tiene derechos de usuario adicionales para crear o modificar el contenido del plan de estudios de un curso y monitorear el desempeño de los estudiantes.

De la experiencia de la aplicación de los sistemas EVA en el medio académico y del análisis de sus resultados las ventajas de los EVA se detallan a continuación:

- **Comunicación:** Abre un número infinito de canales en el formato de foros, hilos de discusión, sondeos, encuestas y retroalimentación instantánea como grupo o individualmente.
- **Páginas de inicio dinámicas:** los profesores tienen la oportunidad de crear un espacio virtual llamativo para representar su tema y atraer la atención de los estudiantes.
- **Producción de trabajos:** Los estudiantes no tienen que encontrar físicamente a su maestro para entregar el trabajo, a la vez que es un método seguro y cuenta con ventanas de tiempo para su entrega.
- **Centro de recursos:** Los profesores tienen infinito espacio de almacenamiento en línea para ppts, documentos, hojas de trabajo, etc. que pueden ser propios del profesor o compartidos con los estudiantes.
- **Enlaces a fuentes externas:** Las vías de acceso a todos los demás espacios de aprendizaje en línea están vinculadas a través del EVA.
- **Contenido incrustado:** Como referencia a YouTube, periódicos y revistas y otros enlaces a página web asociadas a un tema que se esté estudiando.
- **Podcasts y videos:** los podcasts y videos producidos por profesores y estudiantes tienen una plataforma compartida que les permite interactuar a la vez consultar cuando sea requerido.
- **Evaluación:** permite evaluar el avance de los estudiantes en tiempo real y cuando sea requerido y reforzar en temas que no se han comprendido utilizando cualquiera de los métodos mencionados anteriormente.

Existen varias plataformas EVA disponibles en el mercado, unas de pago como son:

- WebCT o Blackboard
- First Class
- eCollege

Y otras Open Source (de libre uso) como son:

- ATutor
- Chamilo

- Claroline
- Docebo
- Dokeos
- LRN
- FLE3
- Moodle
- Olat
- Sakai

A continuación, se detallan ciertas características de la plataforma Moodle, puesto que es el medio de despliegue del EVA sobre el cual se centra la presente investigación.

2.3. Moodle.

Una de las plataformas más populares en entornos de aprendizaje es MOODLE es la abreviación de Modular Object-Oriented Dynamic Learning Environment (Entorno de Aprendizaje Dinámico Orientado a Objetos y Modular).

La principal característica de Moodle es que está hecho en base a la pedagogía social constructivista (el entorno social puede contribuir a mejorarlo). Adicionalmente se puede mencionar las siguientes características [16] :

- Posibilidad de personalizar la plataforma.
- Creación de tantos perfiles de usuario como sea necesario, por ejemplo, administrador, tutor, alumno.
- Creación de cursos virtuales y entornos de aprendizaje virtuales.
- Importación y exportación de datos en formato SCORM.
- Interfaz liviana, adopta las normas W3C.
- Accesibilidad y compatibilidad desde cualquier navegador web.
- Sistema escalable respecto a la cantidad de alumnos.
- Está en constante actualización y evolución.
- Complemento digital para cursos presenciales.
- Posibilidad de diversos métodos de evaluación y calificación.

2.4. Proceso de construcción de plugins en Moodle.

Antes de abordar el tema de construcción puglins en Moodle, es necesario definir la estructura del mismo. Moodle se basa en un núcleo (conocido como el Moodle core) y módulos que provee de la funcionalidad específica a la plataforma [17]. A este core se puede adicionar plugins los cuales dan características adicionales y específicas a Moodle, estos plugins son de varios tipos, por ejemplo: autenticación, bloques, matriculación, temas, actividades, etc.

La estructura de los módulos en Moodle está integrado por un directorio con archivos PHP, JavaScript, XML y/o CSS y puntos de entrada definidos en varios archivos clave como locallib.php, db/install.xml o lib.php que permiten la comunicación con el core.[18].

Los componentes principales del core del Moodle son los siguientes:

- Cursos y Actividades, estos están agrupados en secciones son de varios tipos y flexibles. Los cursos se organizan en categorías jerárquicas dentro de la plataforma.
- Usuarios y Grupos, Los usuarios son básicamente los estudiantes, los estudiantes pueden asociarse a un grupo por ejemplo a un grupo de estudio o curso de una asignatura los cuales cuentan con roles, permisos, y log de transacciones para control de los mismos.
- Matriculación, permite realizar la gestión para que los alumnos puedan inscribirse a un determinado curso o entrenamiento

A estos componentes se les puede adicionar plugins los cuales son elementos y/o aplicaciones que permiten cambio de apariencia, herramientas para un mejor desarrollo del curso como es la interacción del tutor y el alumno, complementos que añade características y funcionalidades adicionales tales como nuevas actividades, nuevas preguntas para examen (cuestionario), nuevos reportes, integración con otros sistemas, etc.

Los plugins de Moodle se clasifican en las siguientes categorías:

- Actividades
- Bloques
- Temas
- Condiciones de disponibilidad
- Usuarios
- Formatos de Curso
- Salidas de mensajería

- Repositorios
- Portafolios
- Plagio
- Protocolos de servicios web
- Herramientas de administración
- Calendarios
- Filtros
- Informes
- Libro de calificaciones
- Plugins generales (locales)
- Editores
- Cache
- Otros

Existen un sinnúmero de plugins elaborados por terceros para el fortalecimiento de la misma, en el sitio <https://moodle.org/plugins/> a la fecha (Enero/2017) existe un total de 1291 plugins.

Cualquier persona u organización que desee contribuir con la elaboración de plugins puede hacerlo para ello se debe seguir los pasos recomendados en la página <https://docs.moodle.org/dev/Tutorial> donde existe una guía de cómo se deben crear plugins. En el Anexo 1 se muestra esta guía como referencia extraída del sitio web mencionada a manera de instructivo.

CAPITULO III

DESARROLLO DEL PROCESO DE PREPARACIÓN DEL PLUGIN PARA LA CLASIFICACIÓN TEMÁTICA.

3.1. Resumen.

En este capítulo se realiza una revisión del marco teórico y elementos que ayudarán a la construcción del plugin Moodle para la clasificación temática de recursos Web.

Para este fin se examinará a profundidad DBPedia como fuente de descripciones de todo tipo de recursos. También se enfocará el estudio en la herramienta desarrollada por DBPedia denominada Spotlight utilizada para la extracción de las palabras claves o tags de un texto, su valoración dentro del contexto del tema; para lo cual se aplica las ontologías de DBpedia en tareas de marcado semántico basándose en técnicas enfocadas a reconocer entidades, anotar un texto o desambiguar un término.

Para la clasificación temática se hace referencia a un método de clasificación, que está conformado por una base de conocimiento la cual es creada en base a un proceso iterativo cuyo objetivo es enriquecer un tesoro controlado definido por la Nomenclatura de la UNESCO (un esquema de conocimiento cerrado) a través de las entidades de un sistema de conocimiento abierto (DBPedia).

3.2. Proceso de clasificación temática de recursos web.

La clasificación temática de recursos web se basa en identificar los temas vinculados al contenido de un recurso. Un recurso puede cualquier documento como una página HTML, un archivo PDF, presentación que estén basados en texto. Por esta razón se requiere realizar un proceso de anotación del texto consultado.

El proceso de anotación no es otra cosa que tomar las palabras claves del texto y asociarlas a un recurso externo para su identificación como concepto. Para este proyecto los conceptos asociados a una palabra clave son tomados desde DBPedia por la relevancia que tiene actualmente. para el proceso de anotación vamos a utilizar la herramienta DBPedia Spotlight ya que de los estudios realizados es una de las más eficientes para manejo de anotaciones como se puede observar en la Figura 2. [19]

<i>System</i>	<i>F1</i>
DBpedia Spotlight (best configuration)	56.0%
DBpedia Spotlight (no configuration)	45.2%
The Wiki Machine	59.5%
Zemanta	39.1%
Open Calais+Naïve	16.7%
Alchemy	14.7%
Ontos+Naïve	10.6%
Open Calais	6.7%
Ontos	1.5%

Figura 2. Porcentaje de precisión respecto al proceso de anotación
Fuente: <http://www.dbpedia-spotlight.org/>

Una vez que se ha anotado el contenido basado en texto, los recursos semánticos son vinculados hacia los temas asociados definidos o clasificados por algún sistema formal como el tesoro definido por la UNESCO para la ciencia y la tecnología., para este proyecto vamos a asociarlos a clasificación internacional de Unesco para la ciencia y tecnología. Los tópicos que tenga mayor relación con el contenido analizado serán presentados al usuario mediante una interfaz Web o como el resultado de llamar a un servicio.

A continuación, se describe el proceso con los conceptos y temas asociados a él para una comprensión exacta del alcance del presente proyecto.

3.3. DBPedia.

DBPedia es un repositorio de datos enlazados que permite el acceso a los datos creados colaborativamente y publicados en la Wikipedia. En otras palabras, es una base de datos enlazada que se mantiene dinámicamente actualizada. Todos los recursos en DBPedia están descrito al menos por una etiqueta, un resumen y un enlace directo a su conceptualización en una página en Wikipedia.

Puesto que DBPedia basa su naturaleza en Wikipedia posibilita acceder a un conjunto ilimitado de recursos de tal manera que puedan ser extraídos y manipulados mediante la relación o la recomendación hacia nuevos temas.

Todas las entradas a Wikipedia están disponibles para todas las personas, además estas páginas pueden ser editadas o modificadas sin ningún tipo de restricción, sin embargo, existen personas llamadas moderadores o editores voluntarios que realizan la función de

proteger el contenido de las páginas para evitar que sean distorsionadas, que exista falsedad a la vez que se comprueba la calidad y exactitud de la información.

En Wikipedia todos los datos se encuentran relacionados de tal manera que una entrada o nodo se relaciona por medio de propiedades (arcos) predefinidos, por ejemplo: birthDate, birthName, birthPlace, deathDate, author of, basedOn of, influenced of, etc. con otros nodos, de este modo se convierte en un mapa conceptual de contenidos semántico y disponible.

DBPedia maneja un importante volumen de datos semánticos contenida en las fichas de Wikipedia con alrededor de 4 millones entradas generando alrededor de 2460 millones de datos semánticos (tripletras RDF) en 119 idiomas.[20]

3.4. DBPedia Spotlight.

DBpedia Spotlight es una herramienta desarrollada para realizar anotaciones automáticamente utilizando los recursos de DBpedia asociados a un texto de tal manera que cada anotación de una palabra o un concepto se asocian a los disponibles en su base de datos. Spotlight puede ser utilizado como un servicio gratuito proporcionado por DBPedia, también es posible instalarlo en una máquina para ser utilizado particularmente. Fue desarrollado por la universidad de Berlín (WBSG at Freie Universität Berlin). [21]

El software está disponible bajo una licencia Apache gratuita, con esta licencia es posible utilizar el código modificarlo y utilizarlo según sea la necesidad.

A continuación, se describirá el proceso que utiliza esta herramienta con respecto al proceso de anotación, es decir el modelo con el cual se reconocen las palabras o conceptos en un texto definido por un usuario [22]:

- **Spotting:** Es el mecanismo con el cual se toma el texto deseado y reconoce formas superficiales como por ejemplo nombres y conceptos, se puede utilizar varias técnicas de detección tales como la búsqueda en diccionario y el reconocimiento de entidad con nombre.
- **Desambiguar:** Una vez que se han reconocido las entidades y conceptos se realiza la desambiguación que mediante técnicas de procesamiento del lenguaje natural y ontologías. En este paso se busca de cada palabra su sentido o significado dentro de la oración, en este caso las entidades y conceptos se encuentren dentro del

contexto de la frase analizada. La desambiguación se realiza utilizando un modelo probabilístico generativo basado en el contexto léxico de la entidad o concepto analizado.

- Para este proceso de desambiguación se parte de TF (Term Frequency) que se utiliza comúnmente para medir la relevancia local de un término en un documento. Para el modelo de Spotlight, TF representa la relevancia de una palabra para un recurso dado para esto se toma en cuenta el peso de la Frecuencia Inversa en el Documento (IDF) que representa la importancia general de la palabra en la colección de recursos DBpedia. Adicionalmente se introduce el peso de la Frecuencia del Candidato Inverso (ICF) donde el poder discriminativo de una palabra es inversamente proporcional al número de entradas en DBpedia asociada a la palabra. Finalmente se calcula el producto de $TF \cdot ICF$ para obtener el peso final de la asociación a DBpedia de la palabra o concepto. Dado que puede existir más de una asociación con el modelo expuesto se puede extraer varios de ellos con un ranking en función del peso final.
- **Anotación:** Es un proceso que implica la ejecución de los dos pasos anteriores, es decir, un texto de entrada reconoce los conceptos y entidades asociados en DBpedia. Se realiza un proceso de desambiguación para que aquellos estén dentro del contexto de la frase y los marca con su referencia hacia DBpedia.
- **Candidatos:** Es un proceso similar a la anotación, pero se devuelve una lista clasificada de los candidatos asociados a cada concepto y entidad en lugar de sólo uno como el caso anterior.

Las anotaciones en DBpedia Spotlight pueden ser de tipo HTML, RDF, XML o JSON, puede filtrarse utilizando el contenido de ciertos tipos desde la ontología de DBpedia o utilizando filtros a partir de SPARQL.

DBpedia Spotlight puede realizar anotaciones para los siguientes lenguajes: inglés, holandés, francés, alemán, húngaro, italiano, portugués, ruso, español y turco.

Además, Spotlight puede discernir los conceptos en un gran volumen de texto plano de manera rápida y efectiva (56.9 GB aproximadamente), a la vez que los desambigua tomando en consideración que en Wikipedia los términos son ambiguos en un 58.8 por ciento.

3.5. Modelo de desarrollo del plugin.

Para el desarrollo del plugin como se ha mencionado se hará uso de la herramienta Spotlight y el uso del tesoro de la UNESCO, para este fin se ha tomado como referencia a todas las disciplinas que están definidas bajo el área de Ciencia de los Ordenadores (Clasificación 1203 [23]).

Se ha tomado como referencia la área mencionada, debido a la disponibilidad de una base de conocimiento generada como resultado de un trabajo previo titulado “Domain categorization of Open Educational Resources based on Linked Data” [24]. Esta base de conocimiento contiene las entradas a DBPedia de los conceptos asociados a la Ciencia de los Ordenadores, que servirá como modelo para realizar las anotaciones de textos asociados a este universo (actualmente en inglés). A su vez las anotaciones basadas en DBPedia están enlazados a los conceptos asociados al tesoro de la UNESCO. El proceso general se observa en la Figura 3.

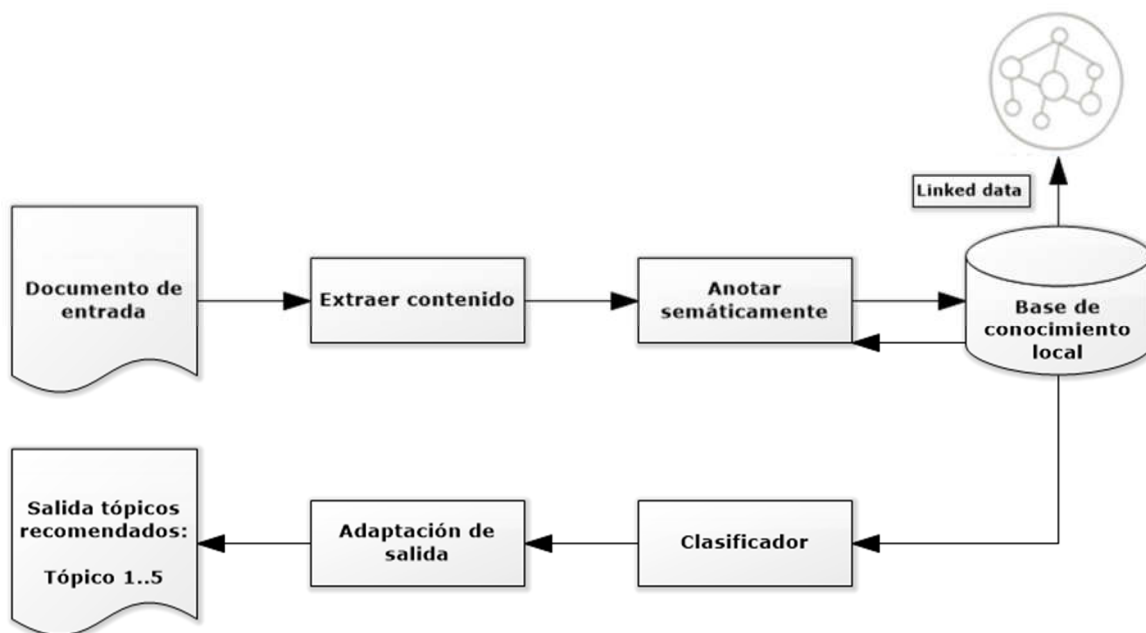


Figura 3. Proceso general de clasificación temática.
Fuente: Tesista.

La base de conocimiento local asociada al tema de la Ciencia de los Ordenadores tiene un total de 576762 entradas a DBPedia. La asociación hacia el tesoro de la UNESCO tiene un total de 622427 entradas desde la base de conocimiento basada en DBPedia.

La clasificación de la Ciencia de los Ordenadores existentes en el tesoro de la UNESCO se puede observar en la tabla 3.

Tabla 3. Clasificación de la Ciencia de los Ordenadores tesoro de la UNESCO

UNESCO URI	Clasificación
http://skos.um.es/unesco6/12	Matemáticas
http://skos.um.es/unesco6/1203	Ciencia de los ordenadores
http://skos.um.es/unesco6/120301	Contabilidad
http://skos.um.es/unesco6/120302	Lenguajes algorítmicos
http://skos.um.es/unesco6/120303	Cálculo analógico
http://skos.um.es/unesco6/120304	Inteligencia artificial
http://skos.um.es/unesco6/120305	Sistemas automatizados de producción
http://skos.um.es/unesco6/120306	Sistemas automatizados de control de calidad
http://skos.um.es/unesco6/120307	Modelos causales
http://skos.um.es/unesco6/120308	Código y sistemas de codificación
http://skos.um.es/unesco6/120309	Diseño con ayuda de ordenador
http://skos.um.es/unesco6/120310	Enseñanza con ayuda de ordenador
http://skos.um.es/unesco6/120311	Lógica de ordenadores
http://skos.um.es/unesco6/120312	Bancos de datos
http://skos.um.es/unesco6/120313	Cálculo digital
http://skos.um.es/unesco6/120314	Sistemas de control del entorno
http://skos.um.es/unesco6/120315	Heurística
http://skos.um.es/unesco6/120316	Cálculo híbrido
http://skos.um.es/unesco6/120317	Informática
http://skos.um.es/unesco6/120318	Sistemas de información
http://skos.um.es/unesco6/120319	Control de inventarios
http://skos.um.es/unesco6/120320	Sistemas de control médico
http://skos.um.es/unesco6/120321	Sistemas de navegación y telemetría del espacio
http://skos.um.es/unesco6/120322	Sistemas de control de producción
http://skos.um.es/unesco6/120323	Lenguajes de programación
http://skos.um.es/unesco6/120324	Teoría de la programación
http://skos.um.es/unesco6/120325	Diseño de sistemas sensores
http://skos.um.es/unesco6/120326	Simulación
http://skos.um.es/unesco6/120399	Otras (especificar)

Fuente: https://www.upct.es/estudios/doctorado/documentos/codigos_unesco_7809.pdf

Para presentar el resultado de los temas (tópicos) se realiza el siguiente proceso: Al peso obtenido por la palabra anotada se multiplicará por el peso de la referencia de DBPedia a la entrada de la UNESCO, si existe varias anotaciones que hacen referencia a la misma entrada de la UNESCO se sumarán para totalizar sobre esta referencia. Se ordenarán las

entradas de la UNESCO de acuerdo con la ponderación total a cada uno según el cálculo descrito y se mostrarán los cinco mayores pesos obtenidos.

Una vez que se ha descrito el proceso de desarrollo de la herramienta, en el siguiente punto se describirá el modelo de implementación del plugin Moodle.

3.6. Modelo de implementación del plugin en Moodle

El plugin Moodle ha sido implementado siguiendo el modelo establecido en los pasos descritos en el Anexo 1. El objetivo es integrar el plugin a través de un servicio web genérico desarrollado, de tal manera que pueda integrarse fácilmente, al mismo tiempo a futuro si existiese alguna mejora en el servicio el plugin no se verá afectado, por ello será independiente.

El servicio web genérico desarrollado basado en DBPedia Spotlight hace uso de las herramientas proporcionadas como librerías y llamadas a funciones para efectuar las anotaciones a un texto ingresado por el usuario, el servicio web ha sido desarrollado para que sea albergado por un servidor web con extensiones PHP que es un lenguaje de código abierto diseñado para el desarrollo web y que puede estar contenido en HTML. Las librerías y funciones de Spotlight se hallan escritas en javascript que es un lenguaje de programación que no requiere de compilación por lo tanto el lenguaje funciona del lado del cliente ya que los navegadores web son los encargados de interpretar y las instrucciones del programa escrito en este lenguaje. Estas librerías se adaptarán para las necesidades de la aplicación que se ha desarrollado.

Como se mencionó en el acápite anterior se contrastará los resultados con una base de conocimiento previamente construida y almacenada en una base de datos de MySQL sistema de gestión de bases de datos relacional que está sobre todo orientada para entornos de desarrollo web. Por último, en el procesamiento basado en pesos y haciendo uso de la clasificación del conocimiento de la UNESCO mostrar los resultados de los tópicos asociados al contenido analizado. Para este fin el servicio web devolverá el resultado en un formato de datos denominado JSON (JavaScript Object Notation - Notación de Objetos de JavaScript), el cual es un formato de intercambio de datos que puede ser leído tanto por los humanos e interpretado por los diferentes lenguajes de programación ya que su estándar es universal. Los resultados pueden ser interpretados por el plugin Moodle y se presenta al usuario final.

Respecto a la construcción del plugin, su definición y los pasos que se siguieron para su construcción se describirá en el siguiente capítulo a detalle, así como la codificación del mismo.

CAPITULO IV

CONSTRUCCIÓN DEL PLUGIN MOODLE PARA LA CLASIFICACIÓN TEMÁTICA.

4.1. Resumen.

En este capítulo se realiza una revisión del proceso de construcción del plugin Moodle, así como los servicios asociados en la clasificación temática, se describen las diferentes fases en el proceso de creación y despliegue del plugin en Moodle.

Como primer paso se detallará el entorno de desarrollo para la construcción del plugin, luego se continuará con la descripción del diseño de la base de datos respecto a la base de conocimiento y su información asociada a los términos de la Ciencia de los Ordenadores, ya que con aquella información se realizarán las anotaciones de los conceptos definidos en DBPedia, así como la clasificación proporcionada por el tesoro de la UNESCO.

Se detallará la estructura de las aplicaciones tanto del plugin de Moodle como de sitio web con los servicios correspondientes de acuerdo al marco teórico detallado en el capítulo anterior; y, su interacción con DBPedia y el tesoro de la UNESCO.

Finalmente se explicará los pasos para instalar el plugin en Moodle, y la publicación del mismo en el entorno Moodle con su respectiva parametrización y despliegue.

4.2. Entorno de desarrollo del plugin Moodle.

El entorno de desarrollo de plugin de Moodle basado en una aplicación web se halla estructurado de la siguiente manera:

- PHP versión 7, como lenguaje de programación de la aplicación web.
- JavaScript como lenguaje de programación del componente de anotación DBPedia como servicio web que interactúa con las extensiones proporcionadas por DBPedia.
- Apache 2.4 como servidor web.
- HTML como lenguaje de presentación de la aplicación web.
- Se utiliza extensiones de jQuery 1.6 para el manejo de Ajax y JSON tanto en la programación del plugin Moodle con las extensiones de DBPedia spotlight como la presentación de la aplicación web.
- Para el desarrollo del plugin Moodle se utilizan las extensiones definidas para la versión 3.3 de Moodle.
- MySQL 5.7 se utiliza para albergar la base de datos de conocimiento, así como también la base de datos de Moodle.

4.3. Diseño de la base de conocimiento.

Como se ha mencionado el plugin tiene por objetivo clasificar el contenido textual de acuerdo con las diferentes disciplinas que están bajo Ciencia de los Ordenadores, Para validar los resultados obtenidos de la anotación realizada utilizando el servicio de DBPedia Spotlight se contrasta con la base de datos respecto al conocimiento de los términos empleados en la ciencia de los ordenadores. Para este fin se ha creado una base de datos en MySql con el nombre de clasificadorbdc, la cual está compuesta por las tablas mostradas en la Figura 4.

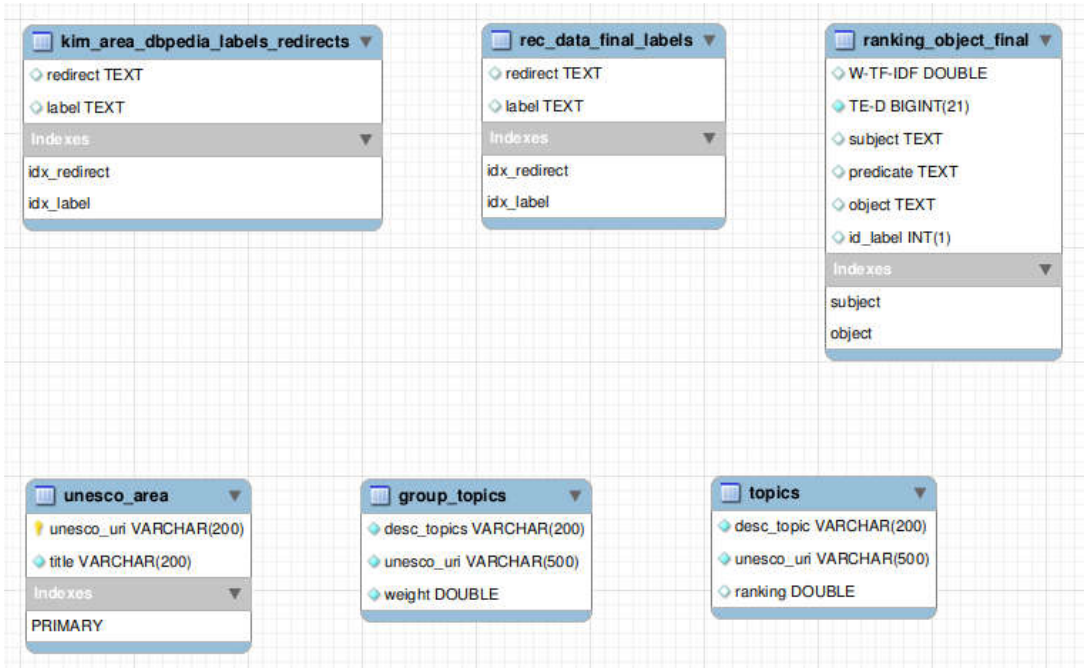


Figura 4. Estructura de las tablas de la Base de Conocimiento Fuente: Tesista

Las tablas y sus campos son los siguientes:

- **kim_area_dbpedia_labels_redirect**, es la primera tabla donde se almacena las direcciones url de los términos asociados a la ciencia de los computadores y que se encuentran en la base de conocimiento de DBPedia. Los campos son los siguientes:
 - o redirect de tipo TEXT, es la url del término asociado en DBPedia.
 - o label de tipo TEXT, es el término que se encuentra definido en la base de conocimiento de DBPedia.

La tabla se halla indexada por el campo redirect y label

- **rec_data_final_labels**, es la segunda tabla donde se almacena las direcciones url de los términos asociados a la ciencia de los computadores y que se encuentran en la base de conocimiento de DBPedia. Los campos son los siguientes:
 - o redirect de tipo TEXT, es la url del término asociado en DBPedia.
 - o label de tipo TEXT, es el término que se encuentra definido en la base de conocimiento de DBPedia.

La tabla se halla indexada por el campo redirect y label

- **ranking_object_final**, es una tabla que asocia mediante el uso de triplas la dirección url de termino encontrado en DBPedia y la definición de este dentro del conocimiento clasificado por la UNESCO. Los campos son los siguientes:
 - o w-tf-idf es de tipo DOUBLE, y es el peso que tiene una palabra dentro de un documento o texto, es utilizada como un factor de ponderación en la recuperación de información y la minería de textos, de este modo se determina que algunas palabras son más comunes que otras.
 - o te-d, es de tipo BIGINT, que determina el peso que tiene la palabra como frecuencia de uso en un texto.
 - o subject, es de tipo TEXT y corresponde al url de la clasificación del término dentro de la clasificación de la Ciencia de los Ordenadores realizado por la UNESCO.
 - o predicate, es de tipo TEXT y determina la clasificación RDF establecida por W3.
 - o object, es de tipo TEXT y corresponde a la url del término dentro de la base de conocimiento de DBPedia.
 - o Id_label, es de tipo INT, es una identificación asignada a la etiqueta.

La tabla se halla indexada por subject y object.

- **unesco_area**, es una tabla que contiene las definiciones de la clasificación de las Ciencia de los Ordenadores establecido por la UNESCO. Los campos son los siguientes:
 - o unesco_uri, de tipo VARCHAR, que es la dirección URL de la definición de la clasificación definida por la UNESCO

- title, de tipo VARCHAR, es el título asignado al campo de clasificación de la UNESCO
- **group_topics**, es una tabla de paso que se utiliza para determinar el peso asignado a cada término anotado y agrupado de un texto y de esta manera determinar los cinco temas más relevantes dentro del texto analizado. Los campos son los siguientes:
 - desc_topics, de tipo VARCHAR, la descripción del tópico encontrado.
 - unesco_uri, de tipo VARCHAR, la dirección URL de la clasificación realizada por la UNESCO.
 - weight, de tipo DOUBLE, y es el peso de todos los términos u ocurrencias que tiene el término dentro del texto analizado.
- **topics**, es una tabla de paso que se utiliza para determinar el peso asignado a cada término anotado de un texto y de esta manera determinar los cinco temas más relevantes dentro del texto analizado. Los campos son los siguientes:
 - desc_topics, de tipo VARCHAR, la descripción del tópico encontrado.
 - unesco_uri, de tipo VARCHAR, la dirección URL de la clasificación realizada por la UNESCO.
 - ranking, de tipo DOUBLE, y es el peso que tiene el término dentro del texto analizado.

4.4. Estructura de los procesos desarrollados para la clasificación temática.

La estructura desarrollada para para la clasificación temática es modular y plasmando lo descrito en el marco teórico del capítulo anterior, los siguientes son los componentes que se han creado:

- Lectura del contenido textual ingresado por el usuario.
- Extracción del contenido y anotación semántica.
- Proceso de anotación con las entradas obtenidas en el punto anterior en la base de conocimiento local.
- Proceso de clasificación temática.
- Adaptación de la salida y presentación al usuario final.

Estos componentes, su arquitectura y elementos empleados se definen de manera detallada a continuación.

4.4.1. Lectura del documento o texto de entrada.

Para este proceso se ha desarrollado una interfaz en el lenguaje PHP en combinación con JQuery orientado al usuario final, para este fin se creó el módulo view.php. Este módulo permite crear la interfaz de usuario final. El código fuente para mostrar la interfaz de ingreso para el usuario del documento o texto de entrada se muestra en el Anexo 2, como se observa en el anexo se encuentran las declaraciones para acoplarse como plugin a Moodle (en color rojo) y las declaraciones para realizar la clasificación temática.

En la Figura 5 se muestra la salida de la interfaz de usuario.



Figura 5. Interfaz de usuario para la lectura del documento o texto de entrada.
Fuente: Tesista.

El usuario ingresará el contenido o texto de un tema por el momento asociado a la ciencia de los ordenadores y en idioma inglés, dado que la construcción del presente plugin se orientó con este criterio. Una vez que el usuario ingresa el texto presiona el botón procesar para continuar con la extracción del contenido y la anotación semántica.

4.4.2. Extracción del contenido y anotación semántica.

Para el proceso de extracción del contenido y la anotación semántica se utiliza el servicio web disponible por DBpedia con el uso del recurso de Spotlight, para ello se realizó una

adaptación a la librería para conseguir el objetivo de la clasificación temática. Se utiliza programación basada en javascript. El método que se utiliza es el de anotación como se puede observar en el código del Anexo 3. Como se definió en el capítulo anterior este proceso realiza: extracción del contenido, selección de candidatos, desambiguación y filtrado, con lo cual se tiene las anotaciones respectivas.

El llamado al servicio de Spotlight de DBpedia (<http://model.dbpedia-spotlight.org/en>) se lo realiza como se puede observar en el siguiente ejemplo:

```
GET
"http://model.dbpedia-
spotlight.org/en/annotate?text=A%20computer%20is%20a%20device%20that%20accepts%20information%20(in%20the
%20form%20of%20digitalized%20data)%20and%20manipulates%20it%20for%20some%20result%20based%20on%20a
%20program&confidence=0.35" -H "accept: application/json"
```

Donde los parámetros que se utilizan son:

- *Annotate*, llama a un proceso de anotación propio de Spotlight
- *text*="texto a anotar"
- *confidence=0.35*, el grado de aplicación del desambiguador entre 0 y 1, mientras más bajo es el valor es mayor el proceso de desambiguación.
- *H "accept:application/json"*, quiere decir que el resultado se devolverá en formato json.

El resultado del ejemplo anterior se puede observar a continuación y corresponde a las anotaciones encontradas dentro del texto:

```
{
  "@text": "A computer is a device that accepts information (in the form of digitalized data) and manipulates it for some
result based on a program",
  "@confidence": "0.33",
  "@support": "0",
  "@types": "",
  "@sparql": "",
  "@policy": "whitelist",
  "Resources": [
    {
      "@URI": "http://dbpedia.org/resource/Computer",
      "@support": "10094",
      "@types": "",
      "@surfaceForm": "computer",
```

```

"@offset": "2",
"@similarityScore": "0.9994450311809745",
"@percentageOfSecondRank": "2.909000535767863E-4"
},
{
"@URI": "http://dbpedia.org/resource/Data",
"@support": "7263",
"@types": "",
"@surfaceForm": "data",
"@offset": "76",
"@similarityScore": "0.8420999583126315",
"@percentageOfSecondRank": "0.18147048293649692"
}
]
}

```

En el texto del ejemplo se anotaron dos palabras: *computer* y *data* que se asocia al contexto de la frase que está analizando, el resultado tiene los siguientes elementos:

- *URI*, dirección donde se halla el concepto de la palabra en DBPedia.
- *support*, es un valor numérico que indica el número de entradas en Wikipedia.
- *types*, el tipo de entrada en DBPedia por ejemplo: DBpedia:Place.
- *surfaceForm*, la palabra anotada.
- *offset*, es la posición de la palabra dentro del contexto del texto analizado.
- *similarityScore*, es un valor numérico que determina la similitud de la palabra anotada y la palabra en los registros de DBPedia
- *percentageOfSecondRank*, es un valor numérico que expresa la posición del segundo candidato encontrado asociado a la palabra dentro de DBPedia

En el proceso de anotación se procede a contrastar con las palabras en la base de conocimiento cuyas referencias se hallan en las entradas a las tablas *kim_area_dbpedia_labels_redirect* y *rec_data_final_labels* las cuales forman parte de la base de conocimiento y que se detalla en el siguiente acápite.

4.4.3. Proceso de anotación con las entradas de la base de conocimiento local.

En el proceso de anotación de Spotlight se realiza una contrastación de las entradas existentes en la base de conocimiento local para este fin se ha elaborado dos servicios

adicionales web en la aplicación; el primero obtiene la entrada desde la tabla *kim_area_dbpedia_labels_redirect*; y, el segundo desde la tabla *rec_data_final_labels* como se puede observar en el código detallado en el Anexo 4.

Si la entrada se encuentra en la base de conocimiento local es tomada en cuenta para realizar el proceso de clasificación y se ingresa la entrada en la tabla de procesamiento topics. La llamada a la función anterior se realiza desde el proceso de anotación definida en el paso anterior 4.4.2. utilizando Ajax de JQuery, para esto primero se borra las entradas en la base de datos para un nuevo proceso de clasificación como se observa en el código detallado en el Anexo 5 y luego se contrasta como se puede observar de manera detallada en el código del Anexo 6.

4.4.4. Proceso de clasificación temática.

El proceso de clasificación consiste en tomar las entradas encontradas en la anotación y procesarla para encontrar los temas más destacados vinculados al texto que se está procesando, para este fin la url de DBPedia obtenida en el proceso de anotación y que fue contrastada con la base de conocimiento local se procede a vincular con la clasificación de la palabra que para este referente a Ciencia de los Ordenadores establecida por la UNESCO, como resultado de este proceso se obtiene la asociación a la url respectiva de la base de conocimiento SKOS en el tesoro de la UNESCO, que como se indicó anteriormente reside en la base de conocimiento local como tripletas en la tabla llamada *ranking_object_final*.

Para este fin se ha creado una función como servicio web, cuyo código se halla detallado en el Anexo 7. En este proceso se toma en cuenta el valor del campo *w-tf-idf* de la tabla *ranking_object_final* y es el peso que tiene una palabra dentro de un documento o texto, este valor se multiplica por el factor de similitud de la palabra en las entradas de DBPedia que es devuelto en el proceso de anotación denominado *similarityScore*, el valor obtenido se utilizada como un factor de ponderación en la clasificación temática ranking. Este servicio es invocado como un servicio web desde la función detallada en el paso 4.4.2. utilizando Ajax de JQuery, como se muestra a continuación:

```
....  
  
// Contrast with knowledge database local  
$.ajax({  
  type: "POST",  
  url: currurl+"reconocimiento.php",  
  data: "subject="+uri+"&ranking="+ranking,
```

```
    async: false,
    dataType: 'json',
    success: function(data)
    {
        label = data[1];
    },
    error: function(xhr, desc, err){
        alert("Hay un error en validación tripletas:" + xhr.responseText);
    }
    });
....
```

4.4.5. Adaptación de la salida y presentación al usuario final.

Para presentar la clasificación temática al usuario final se seleccionan las cinco de mayor relevancia encontradas en el documento, para este fin se parte de las entradas asociadas a la clasificación temática y su representación URI según el tesoro de la UNESCO.

Dado que se conoce el valor *w-tf-idf* de cada entrada para obtener las cinco de mayor importancia se realiza el siguiente proceso:

- Se agrupan las entradas encontradas y clasificadas temáticamente con la misma agrupación en el tesoro de la UNESCO.
- Se suman los pesos, ranking, el producto de *w-tf-idf* y *similarityScore* de todas las entradas que corresponden al mismo grupo.
- Se toman cinco grupos cuya sumatoria sean las mayores, estas serán las clasificaciones temáticas encontradas en el texto que el usuario ingresó y que nuestro sistema de clasificación sugiere al usuario.

El resultado que se muestra al usuario son las anotaciones realizadas en el texto, sus enlaces a DBPedia y las clasificaciones temáticas asociadas al texto ingresado por el cliente y sus enlaces al tesoro de la UNESCO. Un ejemplo de la salida que se muestra al usuario de observa en la Figura 6.



Clasificador temático recursos web

Versión: Ciencia de los Ordenadores inglés

Idioma: Confianza:

Los tópicos relacionados al texto seleccionado asociados al tesoro de la UNESCO son:

- Inteligencia artificial
- Matemáticas
- Sistemas de información
- Heurística
- Modelos causales

Computer simulations reproduce the behavior of a system using a mathematical model. Computer simulations have become a useful tool for the mathematical modeling of many natural systems in physics (computational physics), astrophysics, climatology, chemistry and biology, human systems in economics, psychology, social science, and engineering. Simulation of a system is represented as the running of the system's model. It can be used to explore and gain new insights into new technology and to estimate the performance of systems too complex for analytical solutions.[1]

Computer simulations are computer programs that can be either small, running almost instantly on small devices, or large-scale programs that run for hours or days on network-based groups of computers. The scale of events being simulated by computer simulations has far exceeded anything possible (or perhaps even imaginable) using traditional paper-and-pencil mathematical modeling. Over 10 years ago, a desert-battle

Figura 6. Detalle de la clasificación temática mostrada al usuario.
Fuente: Tesista.

A continuación, se muestra todos los resultados de la anotación asociados a la clasificación temática del texto del ejemplo de la Figura 6, clasificados por sus pesos y que se encuentran en la base de datos en la Figura 7.

desc_topics	unesco_uri	weight
Inteligencia artificial	http://skos.um.es/unesco6/120304	38.83593656
Matemáticas	http://skos.um.es/unesco6/12	38.53710688
Sistemas de información	http://skos.um.es/unesco6/120318	34.68718833
Heurística	http://skos.um.es/unesco6/120315	34.11949733
Modelos causales	http://skos.um.es/unesco6/120307	33.32820886
Ciencia de los ordenadores	http://skos.um.es/unesco6/1203	31.76887054

Simulación	http://skos.um.es/unesco6/120326	24.55624052
Sistemas de control del entorno	http://skos.um.es/unesco6/120314	23.96182465
Bancos de datos	http://skos.um.es/unesco6/120312	20.89946603
Código y sistemas de codificación	http://skos.um.es/unesco6/120308	19.21271819
Diseño con ayuda de ordenador	http://skos.um.es/unesco6/120309	18.41391812
Teoría de la programación	http://skos.um.es/unesco6/120324	13.68017512
Sistemas de control de producción	http://skos.um.es/unesco6/120322	10.64138362
Sistemas de navegación y telemetría del espacio	http://skos.um.es/unesco6/120321	8.484519133
Lenguajes algorítmicos	http://skos.um.es/unesco6/120302	6.592965612
Lenguajes de programación	http://skos.um.es/unesco6/120323	5.655348201
Cálculo digital	http://skos.um.es/unesco6/120313	5.055278758
Cálculo analógico	http://skos.um.es/unesco6/120303	4.744113389
Contabilidad	http://skos.um.es/unesco6/120301	3.689294581
Control de inventarios	http://skos.um.es/unesco6/120319	2.987880937
Sistemas automatizados de control de calidad	http://skos.um.es/unesco6/120306	2.909348839
Sistemas automatizados de producción	http://skos.um.es/unesco6/120305	2.127588922
Cálculo híbrido	http://skos.um.es/unesco6/120316	2.010215773

Figura 7. Pesos de la anotación semántica encontrados del ejemplo.

Fuente: Tesista.

4.5. Construcción del plugin Moodle.

Para la construcción del plugin Moodle del clasificador temático se debe seguir las pautas proporcionadas para este fin y que se halla detallado en el Anexo 1 y se siguió el siguiente proceso.

- Como primer paso se debe crear la estructura de directorios base para la construcción del plugin, que para el proyecto del clasificador se estructuró de la manera mostrada en la figura 8.

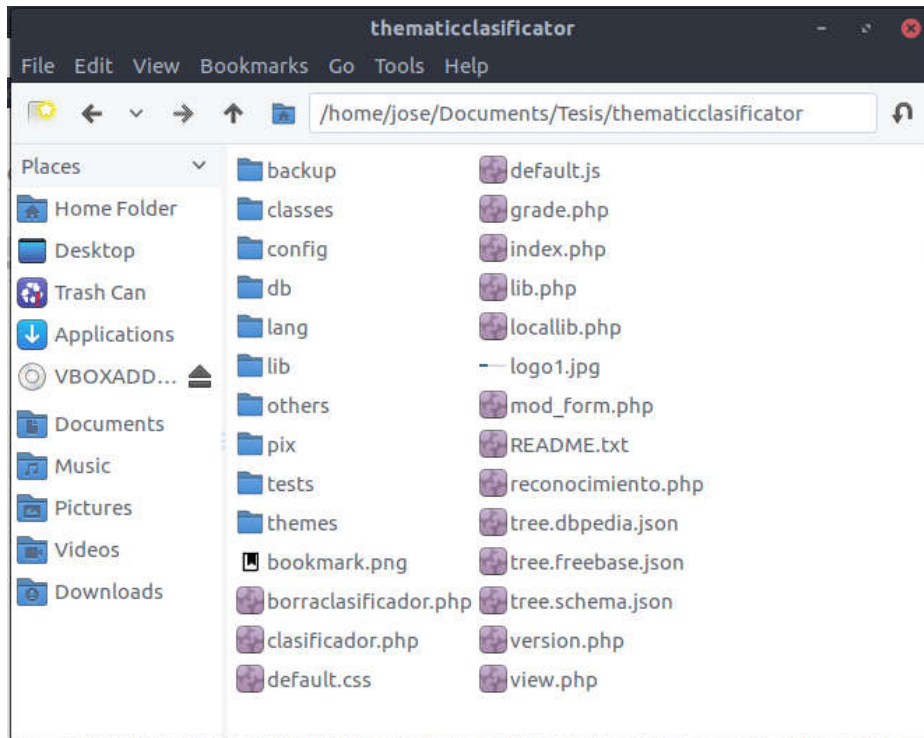


Figura 8. Estructura de directorios para la construcción del plugin.
Fuente: Tesista.

- El siguiente paso es definir la versión del plugin, aquí se indica el nombre del plugin, que se llamará **thematicclasificator**, esto está definido en el archivo *version.php* cuyo código fuente se detalla en el Anexo 8.
- Luego se define la lógica del contenido del plugin que ya se definió y explicó en el punto 4.4.1, y que es básicamente la aplicación del clasificador temático, con sus enlaces y definiciones de los estándares de Moodle.
- Se establece para quienes estará disponible el plugin dentro de Moodle, un parámetro denominado *grade*, esto se halla establecido en el archivo *grade.php*, cuyo código fuente se detalla en el Anexo 9.
- Se define como se presentará el plugin dentro del entorno de Moodle para ello se utiliza el archivo *mod_form.php* cuyo código fuente se detalla en el Anexo 10.
- Se identifica las librerías Moodle a utilizar en el plugin, esto se encuentra en el archivo *locallib.php*, cuyo código fuente se detalla en el Anexo 11.
- Se define las entradas del idioma del plugin para Moodle esto se encuentra dentro del directorio *Lang*, se está utilizando el idioma inglés, por ello se tiene el directorio

en, allí se coloca el archivo con el nombre del plugin y extensión php, para el proyecto es *thematicclasificador.php*, el código fuente se detalla en el Anexo 12.

- Se define las clases que se utilizan para mostrar el plugin en este caso utilizaremos la vista como lista y como módulo dentro de Moodle, para esto se utilizan los archivos *course_module_instance_list_viewed.php* y *course_module_viewed.php* que se encuentran en el directorio *classes/event*, los códigos fuentes respectivamente se muestran en el Anexo 13.
- Se debe definir las entradas a la base de datos de Moodle del plugin cuyas definiciones se encuentran en el directorio **db**, se define el acceso, la instalación, la desinstalación y actualización del plugin. Los archivos para estas tareas respectivamente son *access.php*, cuyo código fuente se muestra en el Anexo 14.
- Se define el proceso de instalación dentro de Moodle, para ello se define el proceso con las directivas de Moodle, para esto se utiliza el archivo *install.php* e *install.xml*, cuyos códigos fuentes se muestran respectivamente en el Anexo 15.
- También se debe definir el proceso de desinstalación del plugin en Moodle para ello se utiliza el archivo *uninstall.php*, su código fuente se muestra en el Anexo 16.
- Se define el proceso de actualización que debe seguir el plugin de Moodle para ello se utiliza el archivo *upgrade.php*, donde se define las estructuras a modificar, librerías, entradas a la base de datos, etc., el código fuente se muestra en el Anexo 17.
- Finalmente se define el proceso de respaldo y restauración del plugin en el entorno Moodle esto se halla en el directorio *backup*, el proceso de respaldo y restauración se define con tareas y pasos, los archivos donde se define aquello son los siguientes:
backup_thematicclasificador_activity_task.class,
backup_thematicclasificador_stepslib.php
restore_thematicclasificador_activity_task.class.php y
restore_thematicclasificador_stepslib.php, cuyo código fuente respectivamente se puede observar en el Anexo 18.

Con esta estructura y definiciones se ha creado el plugin, el último paso es comprimir toda esta estructura en un archivo zip con el nombre del plugin que este caso es *thematicclasificador.zip*, el cual es utilizado para la instalación del plugin en el entorno Moodle.

4.6. Proceso de instalación del plugin Moodle clasificador temático.

Para realizar el proceso de instalación del plugin Moodle desarrollado se siguen los siguientes pasos:

1. Instalar la plataforma Moodle de acuerdo con lo establecido en la página web de la aplicación, como se anota la aplicación requiere de los siguientes componentes:

Hardware, computador requerimientos mínimos:

- Procesador de 2 Ghz con dos núcleos o superior
- 5Gb. en disco disponibles o superior
- Memoria RAM de 1 Gb. o superior

Software uno de los siguientes elementos:

- Sistema operativo Linux, Windows o MacOS
- Base de datos MySQL, Postgres, SQLServer, Oracle
- PHP 5 o superior.
- Apache web server.

Para las pruebas de funcionamiento y desarrollo se ha optado por la siguiente configuración:

Hardware:

- Máquina virtual en vmware de la distribución Linux Ubuntu.
- Memoria RAM 4 Gb.
- Procesador de Intel I5 de 2.5 Ghz de 4 núcleos
- Disco de 40 Gb.

Software:

- Linux Ubuntu 16
- PHP 7.0.18.
- MySQL Server 5.7.
- Apache server 2.4.18.
- Moodle versión 3.3.

2. Ingresar a Moodle como administrador.
3. Seleccionar la opción Administración del Sitio del panel derecho como se observa en la Figura 9.

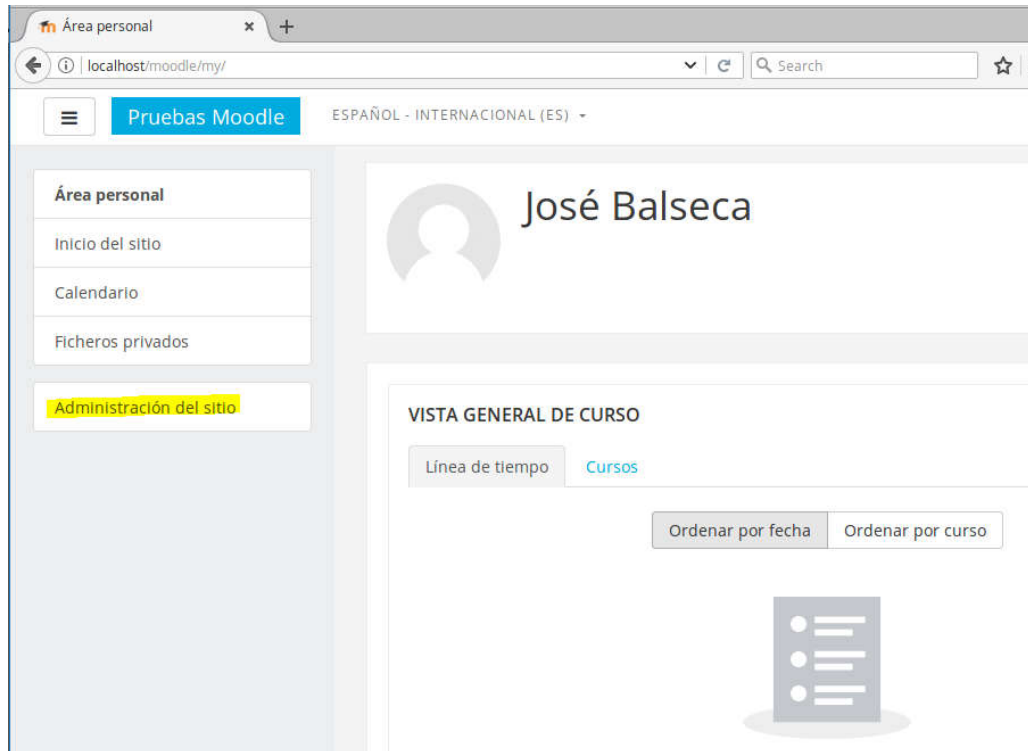


Figura 9. Administración del sitio Moodle.
Fuente: Tesista.

4. Seleccionar la opción Extensiones como se observa en la Figura 10.

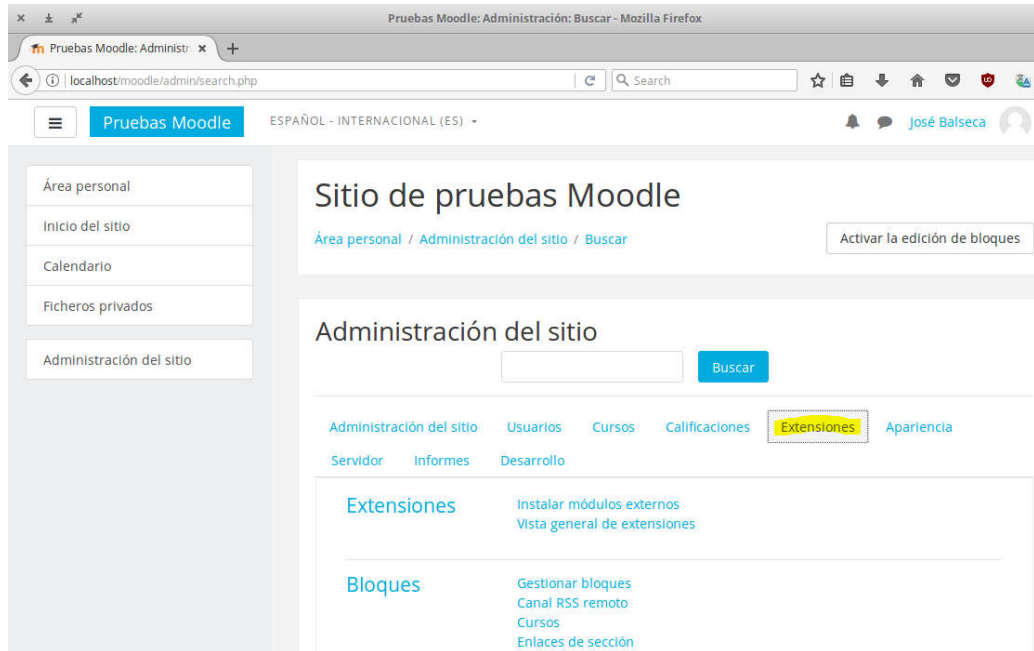


Figura 10. Selección de extensiones Moodle.
Fuente: Tesista.

5. Dentro de la sub-opción Extensiones seleccionar la opción Instalar módulos externos y se mostrará el contenido como en la Figura 11.

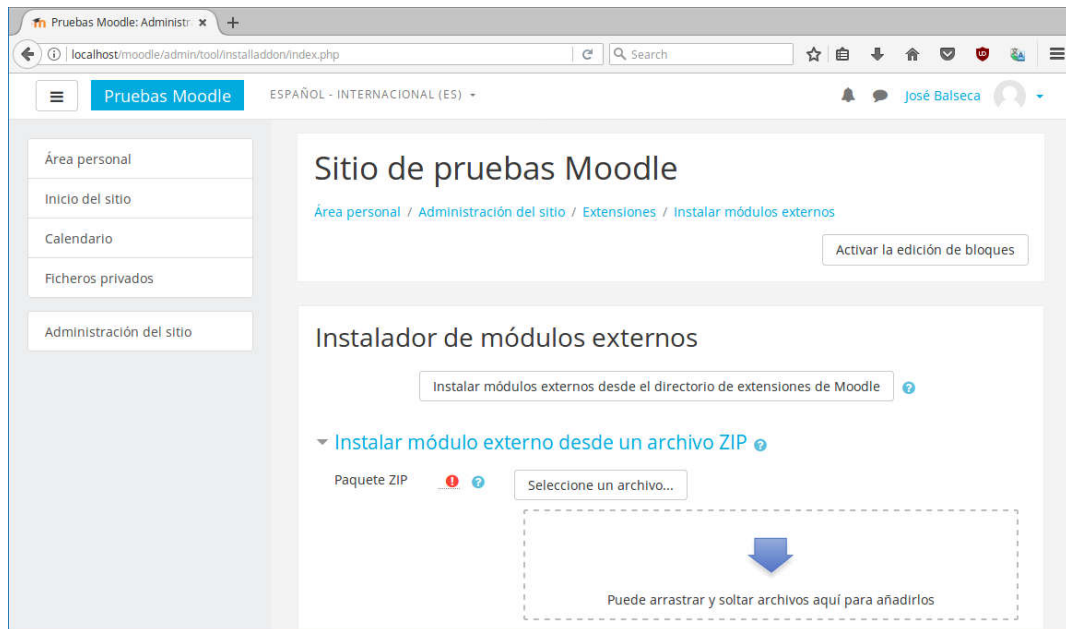


Figura 11. Instalación de extensiones Moodle.
Fuente: Tesista.

Con lo cual se debe seleccionar el botón “Seleccione un archivo” o arrastrar el archivo del plugin desarrollado para Moodle el cual se llama thematicclasificador.zip. Al realizar esto aparece el archivo seleccionado y seleccionar el botón “Instalar módulo externo desde un archivo zip”, como se muestra en la Figura 12.

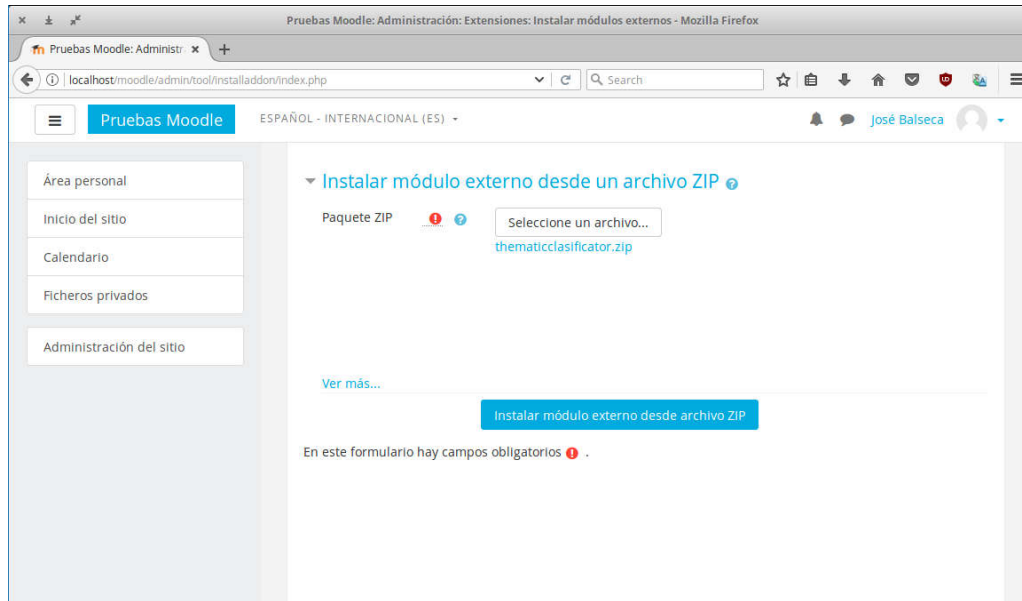


Figura 12. Botón del modo de instalación de plugin Moodle.
Fuente: Tesista.

Luego de esto la plataforma Moodle realiza una revisión del paquete zip donde revisa si cumple todos los estándares establecidos para crear un plugin, si todo está correcto aparece el contenido mostrado en la Figura 13.



Figura 13. Resultado del análisis del paquete de instalación del plugin Moodle.
Fuente: Tesista.

Se continuará presionando el botón “Continuar” y aparecerá el contenido mostrado en la Figura 14.

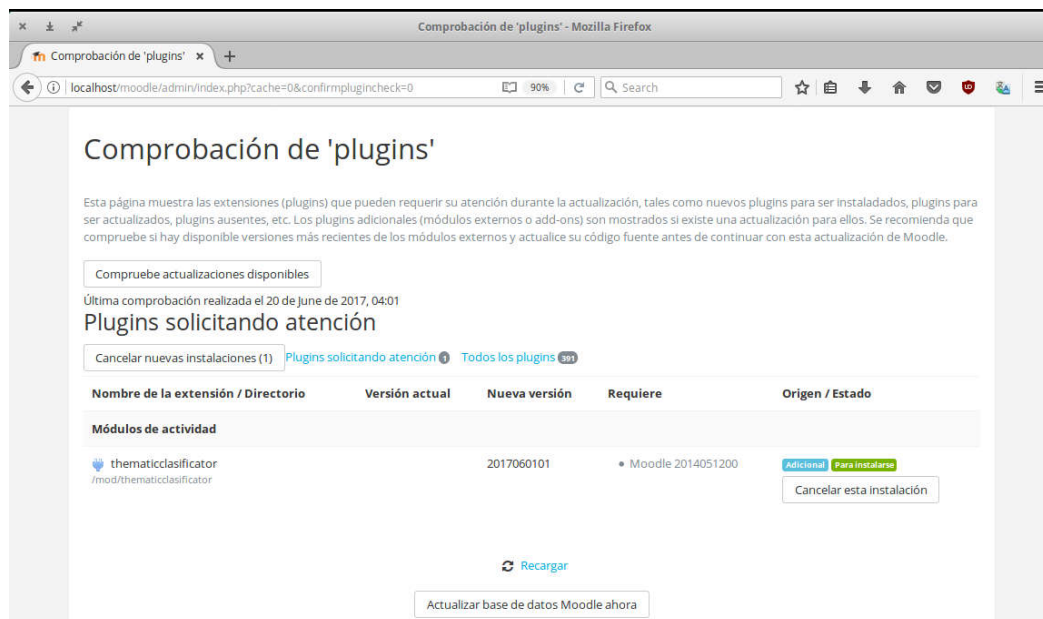


Figura 14. Proceso de instalación del plugin Moodle.
Fuente: Tesista.

Y se dará clic en el botón “Actualizar base de datos Moodle ahora”, con lo cual aparecerá el contenido de la Figura 15.

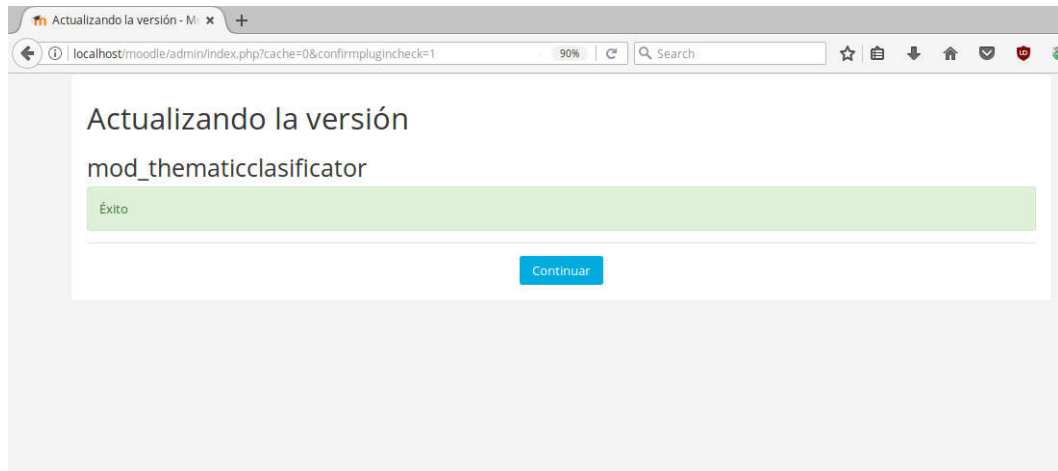


Figura 15. Resultado del proceso de instalación del plugin Moodle.
Fuente: Tesista.

Con esto se muestra que el plugin desarrollado se instaló con éxito.

6. Para mostrar el contenido del plugin en la página deseada de Moodle como administrador se va por la opción “Inicio de sitio” y sobre el ícono de rueda dentada se selecciona la opción “Activar edición” como se muestra en la figura 16.

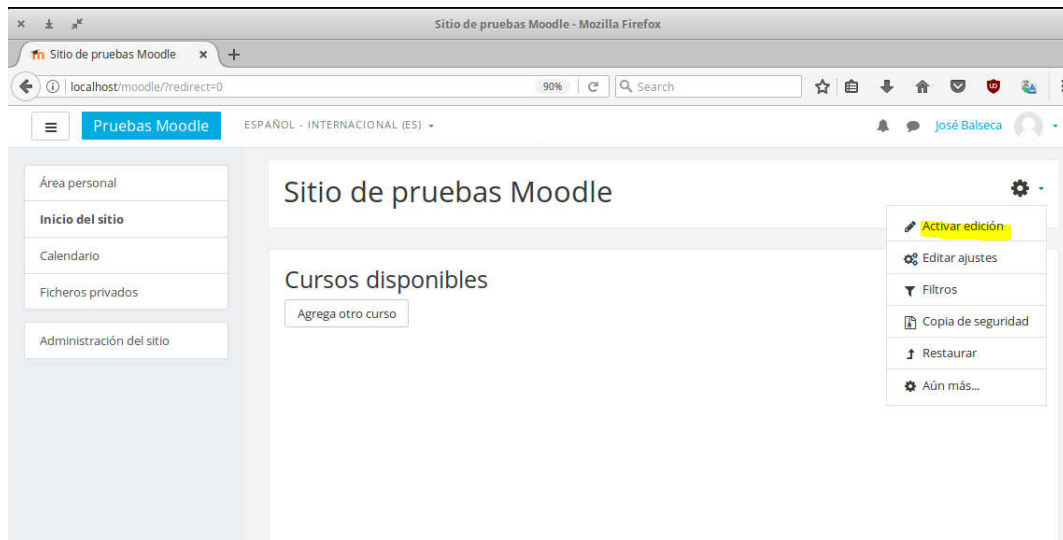


Figura 16. Proceso de gestión del plugin Moodle parte 1.
Fuente: Tesista.

Con esto es posible añadir una actividad o recurso, en este caso el contenido de nuestro plugin de clasificador temático de recursos web, al hacer clic sobre el link de “Añadir una actividad o recurso” aparece el contenido de la Figura 17.

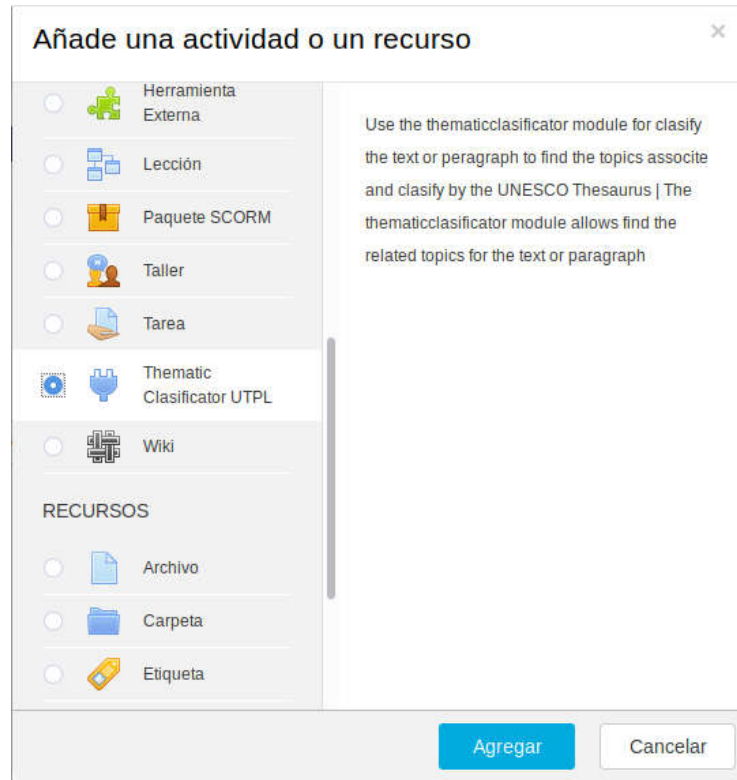


Figura 17. Proceso de gestión del plugin Moodle parte 2.
Fuente: Tesista.

Como se puede observar se tiene en la lista el plugin del clasificador temático, se selecciona y se da clic en el botón Agregar. Luego aparece el formulario para determinar el nombre y parámetros del plugin dentro del entorno de Moodle como se observa en la Figura 18.

Figura 18. Parametrización del plugin Moodle parte 1.
Fuente: Tesista.

En el área general se ingresa el nombre y la descripción del plugin de acuerdo al criterio del administrador de Moodle.

En el área de Ajustes comunes del módulo se define quienes tienen acceso al componente y/o que grupos, se tiene lo que se muestra en la Figura 19.

Figura 19. Parametrización del plugin Moodle parte 2.
Fuente: Tesista.

En el área de Restricciones de acceso se define las posibilidades que se puede especificar para acceder al componente esto se muestra en la Figura 20.

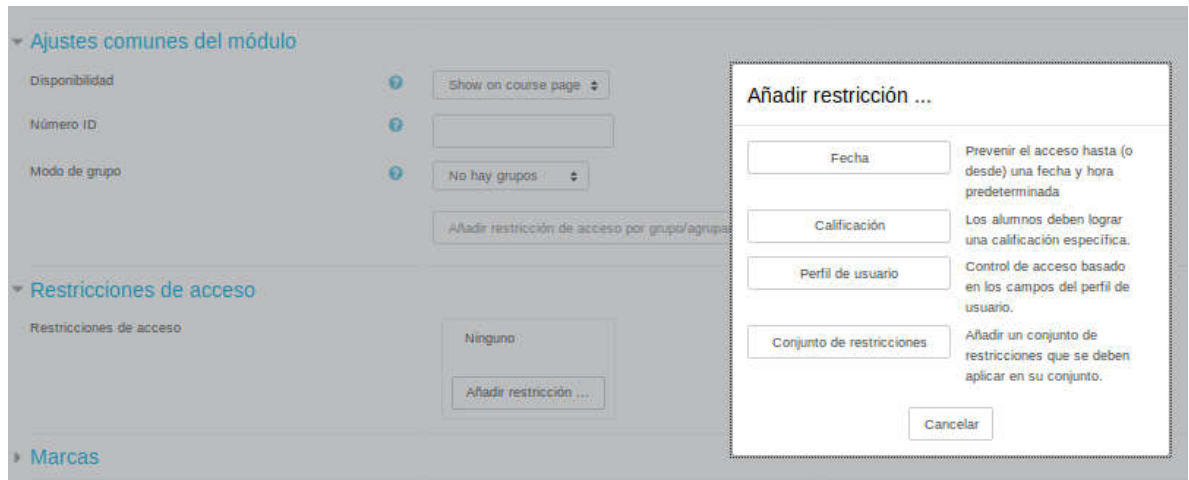


Figura 20. Parametrización del plugin Moodle parte 3.
Fuente: Tesista.

En el área de marcas y competencia se establece parámetros de cumplimiento que para el caso del clasificador temático no se hará uso.

Para terminar, se hará clic en el botón Guardar cambios, finalmente el componente se muestra cómo se observa en la Figura 21.

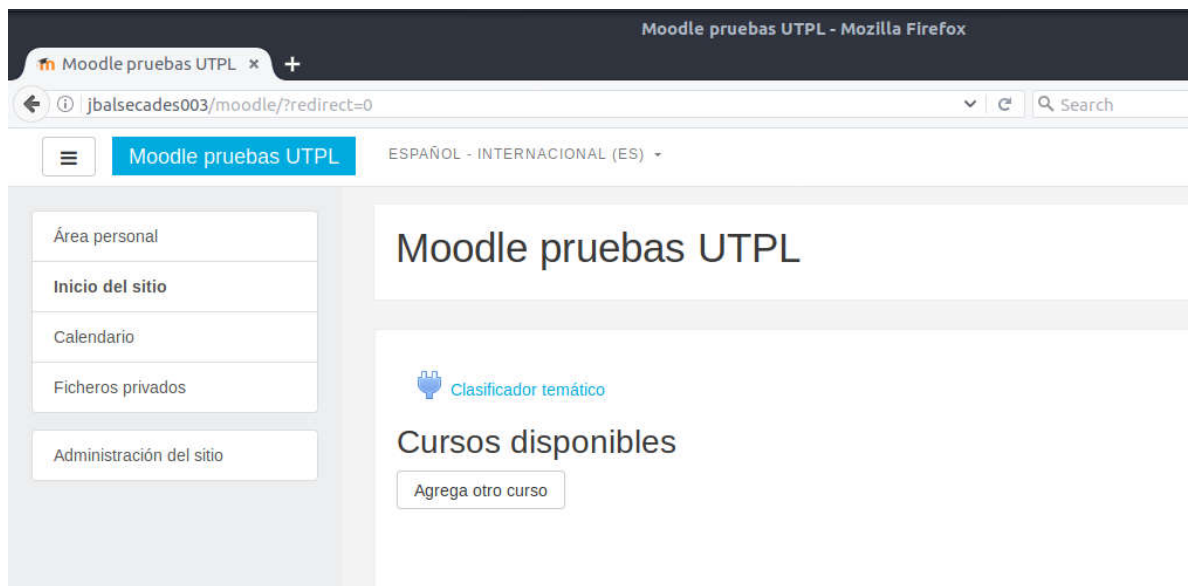


Figura 21. Resultado de la gestión del plugin en el entorno Moodle.
Fuente: Tesista.

Al acceder al link del componente se tiene el entorno de trabajo del clasificador temático como se observa en la figura 22.

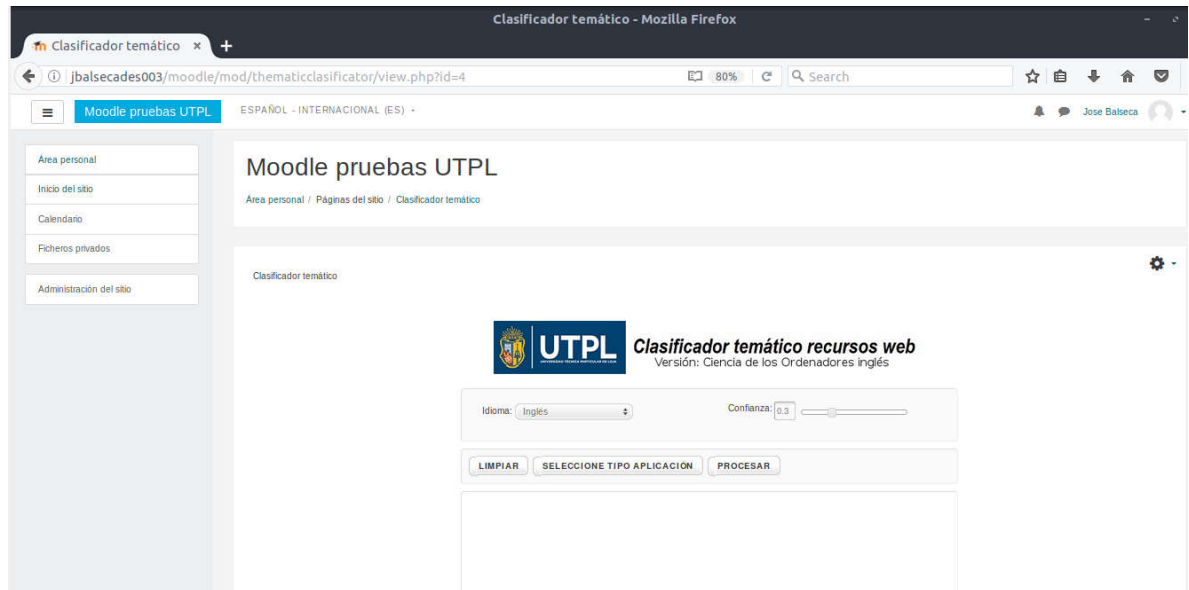


Figura 22. Entorno de trabajo del usuario del Clasificador Temático en Moodle.
Fuente: Tesista.

Como experiencia en la construcción del plugin clasificador temático en Moodle se puede resaltar los siguiente:

- DBPedia ofrece un muy buen servicio para realizar el proceso de anotación en todas sus instancias, la invocación al servicio es sencillo, la respuesta es rápida y detallada en el contexto de análisis. Las herramientas proporcionadas por Spotlight en el ambiente de desarrollo es de fácil aplicación.
- Se pudo aplicar los conceptos respecto al diseño de base de datos adquiridos en la carrera para la estructuración e implementación de la base de conocimiento albergada en MySQL.
- La construcción del plugin permitió el uso de las diferentes herramientas de programación disponibles en la actualidad en entornos Web, así como también mediante la investigación de su funcionamiento permitió aplicarlas de la mejor manera para un buen desempeño del plugin en este tipo de entornos.
- A pesar de que existe la documentación de la elaboración de plugins para Moodle resultó una tarea bastante ardua ya que hay que conocer todas las

posibilidades de parametrización de componentes Moodle y sus diferentes aspectos para poder implementarlo ya que si no se reúne todos estos elementos resulta algo complicado que Moodle reconozca a un plugin. Se encontró que hay muy poca documentación respecto a plugins de la índole del clasificador temático. Se realizó un proceso de causa y efecto para llegar a la construcción final del plugin. Sin embargo, al final resultó un proceso enriquecedor ya que afianzó los diversos conocimientos que ya se tenía así como también permitió conocer nuevos conceptos en el desarrollo de aplicaciones web de manera general.

CAPITULO V

VALIDACIÓN DEL PLUGIN MOODLE PARA LA CLASIFICACIÓN TEMÁTICA.

5.1. Resumen.

En este capítulo se presenta la validación de los resultados obtenidos por el clasificador temático como plugin de Moodle para esto se describen diferentes pruebas de funcionalidad sus resultados, tiempos de procesamiento y calidad de los temas que son mostrados al usuario final, así como la eficiencia del plugin dentro del entorno Moodle.

5.2. Pruebas de funcionalidad.

En las pruebas de funcionalidad se comprobará el comportamiento del clasificador ante la posibilidad de fallas de conectividad a los servicios o base de datos donde se halla la base de conocimiento.

5.2.1. Comportamiento cuando no se encuentra disponible la base de conocimiento local.

En esta prueba se detendrá el servicio de MySQL como se observa en la Figura 23, y es donde reside la base de conocimiento. Para comprobar el comportamiento del clasificador, se realizó un proceso de clasificación luego se realizó esta prueba se obtuvo el resultado mostrado en la Figura 24.

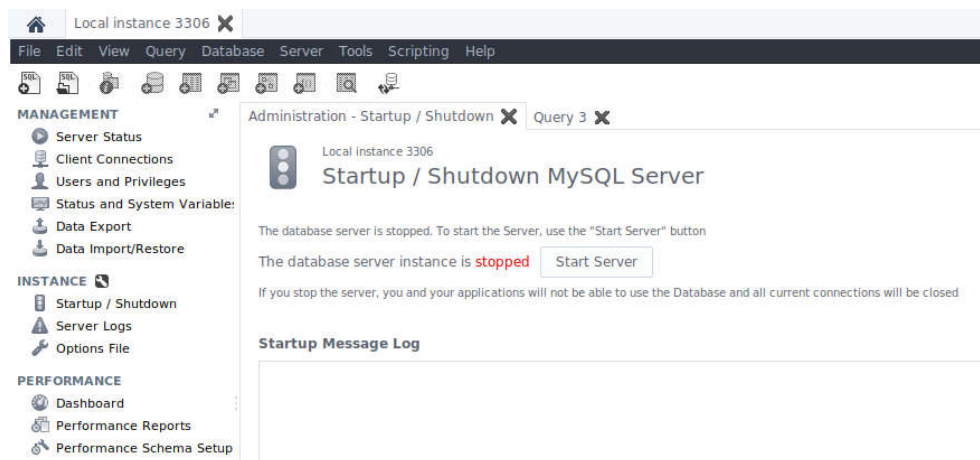


Figura 23. Detención del servicio de MySQL.
Fuente: Tesista.

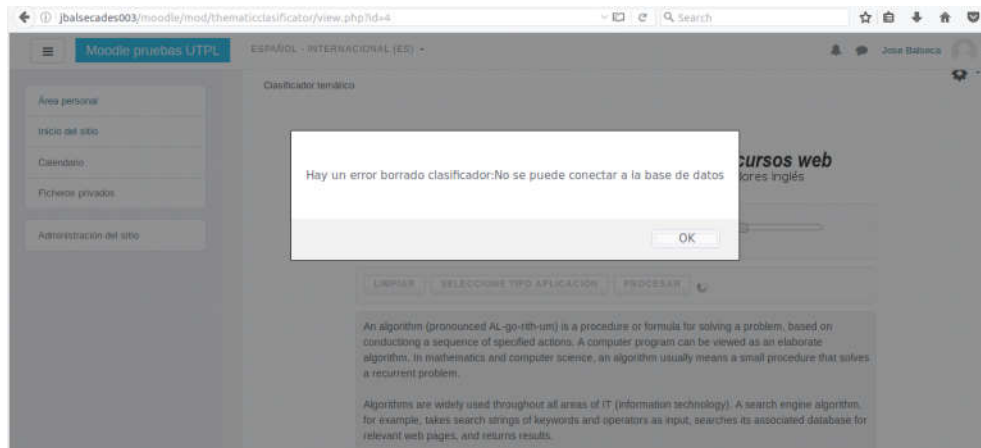


Figura 24. Resultado del plugin Moodle cuando se detiene el servicio MySQL.
Fuente: Tesista.

Con este mensaje el administrador de Moodle puede tomar los correctivos necesarios, nótese que el medio ambiente de Moodle sigue funcionando ya que la base de conocimiento es independiente de la base de datos de Moodle.

5.2.2. Comportamiento cuando se intenta realizar una clasificación con un contenido diferente al idioma inglés.

Como se ha especificado el clasificador está orientado a la Ciencia de los ordenadores y contenidos escritos en inglés, se mostrará entonces cómo el clasificador se comporta con un contenido diferente al idioma inglés como se muestra en la Figura 25 y su resultado como se muestra en la Figura 26.



Clasificador temático recursos web

Versión: Ciencia de los Ordenadores inglés

Idioma: Inglés Confianza: 0.6

La inteligencia artificial es considerada una rama de la computación y relaciona un fenómeno natural con una analogía artificial a través de programas de computador. La inteligencia artificial puede ser tomada como ciencia si se enfoca hacia la elaboración de programas basados en comparaciones con la eficiencia del hombre, contribuyendo a un mayor entendimiento del conocimiento humano.

Si por otro lado es tomada como ingeniería, basada en una relación deseable de entrada-salida para sintetizar un programa de computador. "El resultado es un programa de alta eficiencia que funciona como una poderosa herramienta para quien la utiliza."

A través de la inteligencia artificial se han desarrollado los sistemas expertos que pueden imitar la capacidad mental del hombre y relacionan reglas de sintaxis del lenguaje hablado y escrito sobre la base de la experiencia, para luego hacer juicios acerca de un problema, cuya solución se logra con mejores juicios y más rápidamente que el ser humano. En la medicina tiene gran utilidad al acertar el 85 % de los casos de diagnóstico

Figura 25. Texto en español introducido por el usuario del plugin Moodle.
Fuente: Tesista.



Clasificador temático recursos web

Versión: Ciencia de los Ordenadores inglés

Idioma: Inglés Confianza: 0.6

No se encontraron tópicos relacionados al texto seleccionado.

La inteligencia artificial es considerada una rama de la computación y relaciona un fenómeno natural con una analogía artificial a través de programas de computador. La inteligencia artificial puede ser tomada como ciencia si se enfoca hacia la elaboración de programas basados en comparaciones con la eficiencia del hombre, contribuyendo a un mayor entendimiento del conocimiento humano.

Si por otro lado es tomada como ingeniería, basada en una relación deseable de entrada-salida para sintetizar un programa de computador. "El resultado es un programa de alta eficiencia que funciona como una poderosa herramienta para quien la utiliza."

A través de la inteligencia artificial se han desarrollado los sistemas expertos que pueden imitar la capacidad mental del hombre y relacionan reglas de sintaxis del lenguaje hablado y escrito sobre la base de la experiencia, para luego hacer juicios acerca de un problema, cuya solución se logra con mejores juicios y más rápidamente que el ser humano. En la medicina tiene gran utilidad al acertar el 85 % de los casos de diagnóstico

Figura 26. Resultado devuelto luego de aplicar la clasificación temática.
Fuente: Tesista.

5.2.3. Comportamiento cuando se intenta realizar una clasificación con un contenido diferente al de la ciencia de los ordenadores.

Como se ha especificado el clasificador está orientado a la Ciencia de los ordenadores y contenidos escritos en inglés, se mostrará entonces lo que el clasificador muestra con un contenido diferente a la ciencia de los ordenadores como se muestra en la Figura 27 y su resultado como se muestra en la Figura 28. Hay que anotar que el usuario puede hacer uso del valor de confianza con lo cual el proceso de desambiguación puede evitar falsos positivos al aumentar el valor del mismo.

UTPL *Clasificador temático recursos web*
Versión: Ciencia de los Ordenadores inglés

Idioma: Inglés Confianza: 0.8

LIMPIAR SELECCIONE TIPO APLICACION PROCESAR

Miami (/maɪˈæmiː; Spanish pronunciation: [miˈami]) is a major port city on the Atlantic coast of south Florida in the southeastern United States. As the seat of Miami-Dade County, the municipality is the principal, central, and the most populous city of the Miami metropolitan area and part of the second-most populous metropolis in the southeastern United States.[8] According to the U.S. Census Bureau, Miami's metro area is the eighth-most populous and fourth-largest urban area in the U.S., with a population of around 5.5 million.[9][10]

Miami is a major center, and a leader in finance, commerce, culture, media, entertainment, the arts, and international trade.[11][12] In 2012, Miami was classified as an Alpha-World City in the World Cities Study Group's inventory.[13] In 2010, Miami ranked seventh in the United States and 33rd among global cities in terms of business activity, human capital, information exchange, cultural experience, and political engagement.[14][15] In 2008, Forbes magazine ranked Miami "America's Cleanest City", for its year-round good air quality, vast green spaces, clean drinking water, clean streets, and citywide recycling programs.[16] According to a 2009 a study of 73 world cities, Miami was ranked as the richest city in the United States, and the world's seventh-richest city in terms of purchasing power.[17] Miami is nicknamed the "Capital of Latin America"[1] and is the largest city with a Cuban-American plurality.[18]

Figura 27. Contenido ingresado por el usuario del plugin Moodle diferente a Ciencia de los Ordenadores

Fuente: Tesista.

Idioma: Inglés Confianza: 0,8

No se encontraron tópicos relacionados al texto seleccionado.

Miami (/maɪˈæmi/; Spanish pronunciation: [miˈami]) is a major port city on the Atlantic coast of south Florida in the southeastern United States. As the seat of Miami-Dade County, the municipality is the principal, central, and the most populous city of the Miami metropolitan area and part of the second-most populous metropolis in the southeastern United States.[8] According to the U.S. Census Bureau, Miami's metro area is the eighth-most populous and fourth-largest urban area in the U.S., with a population of around 5.5 million.[9][10]

Miami is a major center, and a leader in finance, commerce, culture, media, entertainment, the arts, and international trade.[11][12] In 2012, Miami was classified as an Alpha-World City in the World Cities Study Group's inventory.[13] In 2010, Miami ranked seventh in the United States and 33rd among global cities in terms of business activity, human capital, information exchange, cultural experience, and political engagement.[14][15] In 2008, Forbes magazine ranked Miami "America's Cleanest City", for its year-round good air quality, vast green spaces, clean drinking water, clean streets, and citywide recycling programs.[16] According to a 2009 a study of 73 world cities, Miami was ranked as the richest city in the United States, and the world's seventh-richest city in terms of purchasing power.[17] Miami is nicknamed the "Capital of Latin America"[1] and is the largest city with a Cuban-American plurality.[18]

Figura 28. Resultado devuelto luego de aplicar la clasificación temática.
Fuente: Tesista.

5.3. Validación de los resultados proporcionados por el clasificador.

En las pruebas de los resultados se mostrará la efectividad del clasificador temático al contrastar la información desde diferentes puntos de vista.

5.3.1. Comportamiento del clasificador al variar el nivel de confianza (desambiguación).

A continuación, se mostrarán los resultados del clasificador temático al ir variando el valor del nivel de confianza o desambiguación según las necesidades del usuario final. Para esto se ha tomado como referencia el siguiente texto, que es la definición de inteligencia dada por Wikipedia en inglés:

"Artificial intelligence (AI, also machine intelligence, MI) is intelligence exhibited by machines, rather than humans or other animals (natural intelligence, NI). In computer science, the field of AI research defines itself as the study of "intelligent agents": any device that perceives its environment and takes actions that maximize its chance of success at some goal.[1] Colloquially, the term "artificial intelligence" is applied when a machine

mimics "cognitive" functions that humans associate with other human minds, such as "learning" and "problem solving".[2]

The scope of AI is disputed: as machines become increasingly capable, task considered as requiring "intelligence" are often removed from the definition, a phenomenon known as the AI effect, leading to the quip "AI is whatever hasn't been done yet." [3] For instance, optical character recognition is frequently excluded from "artificial intelligence", having become a routine technology. [4] Capabilities generally classified as AI, as of 2017, include successfully understanding human speech, [5] competing at a high level in strategic game systems (such as chess and Go [6]), autonomous cars, intelligent routing in content delivery networks, military simulations, and interpreting complex data.

AI research is divided into subfields [7] that focus on specific problems, approaches, the use of a particular tool, or towards satisfying particular applications.

The central problems (or goals) of AI research include reasoning, knowledge, planning, learning, natural language processing (communication), perception and the ability to move and manipulate objects. [8] General intelligence is among the field's long-term goals. [9] Approaches include statistical methods, computational intelligence, and traditional symbolic AI. Many tools are used in AI, including versions of search and mathematical optimization, logic, methods based on probability and economics. The AI field draws upon computer science, mathematics, psychology, linguistics, philosophy, neuroscience, artificial psychology and many others.

The field was founded on the claim that human intelligence "can be so precisely described that a machine can be made to simulate it". [10] This raises philosophical arguments about the nature of the mind and the ethics of creating artificial beings endowed with human-like intelligence, issues which have been explored by myth, fiction and philosophy since antiquity. [11] Some people also consider AI a danger to humanity if it progresses unabatedly. [12] Attempts to create artificial intelligence have experienced many setbacks, including the ALPAC report of 1966, the abandonment of perceptrons in 1970, the Lighthill Report of 1973, the second AI winter 1987–1993 and the collapse of the Lisp machine market in 1987.

In the twenty-first century, AI techniques, both hard (using a symbolic approach) and soft (sub-symbolic), have experienced a resurgence following concurrent advances in computer power, sizes of training sets, and theoretical understanding, and AI techniques have become an essential part of the technology industry, helping to solve many challenging problems in computer science. [13] Recent advancements in AI, and specifically in machine learning, have contributed to the growth of Autonomous Things such as drones and self-driving cars, becoming the main driver of innovation in the automotive industry."

En la Tabla 4 se muestra los resultados del clasificador al variar el valor de confianza:

Tabla 4. Resultados del clasificador al variar el valor de confianza

Nivel de confianza	Clasificación temática sugerida
1	<ul style="list-style-type: none"><li data-bbox="591 1621 850 1650">- Inteligencia artificial<li data-bbox="591 1671 902 1701">- Sistemas de información<li data-bbox="591 1722 834 1751">- Modelos causales<li data-bbox="591 1772 932 1801">- Ciencia de los ordenadores<li data-bbox="591 1822 980 1852">- Diseño con ayuda de ordenador

0.9	<ul style="list-style-type: none"> - Ciencia de los ordenadores - Inteligencia artificial - Sistemas de información - Sistemas de navegación y telemetría del espacio - Matemáticas
0.8	<ul style="list-style-type: none"> - Matemáticas - Ciencia de los ordenadores - Sistemas de navegación y telemetría del espacio - Inteligencia artificial - Sistemas de información
0.7	<ul style="list-style-type: none"> - Matemáticas - Ciencia de los ordenadores - Sistemas de navegación y telemetría del espacio - Inteligencia artificial - Sistemas de información
0.6	<ul style="list-style-type: none"> - Matemáticas - Ciencia de los ordenadores - Sistemas de navegación y telemetría del espacio - Inteligencia artificial - Código y sistemas de codificación
0.5	<ul style="list-style-type: none"> - Matemáticas - Ciencia de los ordenadores - Sistemas de navegación y telemetría del espacio - Inteligencia artificial - Código y sistemas de codificación
0.4	<ul style="list-style-type: none"> - Matemáticas - Ciencia de los ordenadores - Sistemas de navegación y telemetría del espacio - Inteligencia artificial - Código y sistemas de codificación
0.3	<ul style="list-style-type: none"> - Ciencia de los ordenadores - Matemáticas

	<ul style="list-style-type: none"> - Inteligencia artificial - Sistemas de navegación y telemetría del espacio - Sistemas de información
0.2	<ul style="list-style-type: none"> - Ciencia de los ordenadores - Matemáticas - Inteligencia artificial - Sistemas de navegación y telemetría del espacio - Sistemas de información
0.1	<ul style="list-style-type: none"> - Ciencia de los ordenadores - Matemáticas - Inteligencia artificial - Sistemas de navegación y telemetría del espacio - Sistemas de información

Fuente: Tesista

Como se puede observar cuando se tiene un nivel de confianza 1 el resultado mostrado trata de enmarcarse en el concepto de inteligencia artificial dado que existe pocas anotaciones por la exigencia del algoritmo de desambiguación. Sin embargo, entre 0.9 y 0.1 la orientación de los temas sugeridos al usuario final es casi la misma solo que varía la presentación por la ponderación de las anotaciones encontradas, sin embargo, es una guía eficaz al usuario para buscar temas asociados al texto que se está analizando.

5.4. Comportamiento del clasificador con diferentes textos analizados.

A continuación, en la Tabla 5 se muestra los resultados de diferentes textos analizados y las clasificaciones temáticas mostradas al usuario con un nivel de confianza de 0.7.

Tabla 5. Resultados del clasificador con diferentes textos con un nivel de confianza de 0.7

Contenido	Clasificación temática
Concepto de algoritmo según el sitio: http://www.businessdictionary.com "Step by step procedure designed to perform an operation, and which (like a map or flowchart) will lead to the sought result if followed correctly. Algorithms have a definite beginning and a definite end, and a finite number of steps. An algorithm produces the same output information given the same input information, and several short algorithms can be combined to perform complex tasks such as writing a computer program. A cookbook recipe, a	<ul style="list-style-type: none"> - Heurística - Código y sistemas de codificación - Matemáticas - Sistemas automatizados de control de calidad

<p>diagnosis, a problem-solving routine, are some common examples of simple algorithms. Suitable for solving structured problems (amenable to sequential analysis) algorithms are, however, unsuitable for problems where value judgments are required.”</p>	<ul style="list-style-type: none"> - Inteligencia artificial
<p>Concepto de redes neuronales según el sitio: www.doc.ic.ac.uk</p> <p>“An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems. ANNs, like people, learn by example. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons. This is true of ANNs as well.”</p>	<ul style="list-style-type: none"> - Código y sistemas de codificación - Inteligencia artificial - Modelos causales - Matemáticas - Bancos de datos
<p>Concepto de redes Wireless según el sitio: http://www.home-network-help.com</p> <p>“Wireless network is a network set up by using radio signal frequency to communicate among computers and other network devices. Sometimes it’s also referred to as WiFi network or WLAN. This network is getting popular nowadays due to easy to setup feature and no cabling involved. You can connect computers anywhere in your home without the need for wires. Here is simple explanation of how it works, let say you have 2 computers each equipped with wireless adapter and you have set up wireless router. When the computer send out the data, the binary data will be encoded to radio frequency and transmitted via wireless router. The receiving computer will then decode the signal back to binary data. It doesn’t matter you are using broadband cable/DSL modem to access internet, both ways will work with wireless network. If you heard about wireless hotspot, that means that location is equipped with wireless devices for you and others to join the network. The two main components are wireless router or access point and wireless clients. If you have not set up any wired network, then just get a wireless router and attach it to cable or DSL modem. You then set up wireless client by adding wireless card to each computer and form a simple wireless network. You can also cable connect computer directly to router if there are switch ports available.”</p>	<ul style="list-style-type: none"> - Lenguajes algorítmicos - Cálculo analógico - Bancos de datos - Sistemas de navegación y telemetría del espacio - Sistemas de control del entorno
<p>Concepto de diseño asistido por computadores (CAD) según el sitio: https://cadsetterout.com</p> <p>“What is CAD to a Setter out? Once upon a time a craftsman would make his product and that would be the end of it. In order to increase production he might take on some Apprentices who would learn by copying the original. The making of copies would be aided by marking dimensions on off cuts of timber or ‘Rods’.</p>	<ul style="list-style-type: none"> - Sistemas automatizados de producción - Diseño con ayuda de ordenador - Sistemas de información

<p>This technique began with the Medieval master Masons and became increasingly sophisticated, with full size sectional drawings of components being marked out on planks of timber (also known as 'Rods'). In more recent times plywood panels or sheets of lining paper were used.</p> <p>Drawing out items by hand has largely been superseded by Computer Aided Design. Drawings are created in 2D or 3D using the computer, they can be printed out full size or to scale. Increasingly geometry can be passed straight to a CAM* programmer who will use the geometry to create a program which will run on a CNC* machine."</p>	
<p>La explicación de CPU según el sitio: www.techopedia.com</p> <p>"The central processing unit (CPU) is the unit, which performs most of the processing inside a computer. To control instructions and data flow to and from other parts of the computer, the CPU relies heavily on a chip set, which is a group of microchips located on the motherboard.</p> <p>The CPU has two typical components:</p> <p>Control Unit: extracts instructions from memory and decodes and executes them.</p> <p>Arithmetic Logic Unit (ALU): handles arithmetic and logical operations.</p> <p>To function properly, the CPU relies on the system clock, memory, secondary storage, and data and address buses.</p> <p>This term is also known as a central processor, microprocessor or chip.</p> <p>The CPU is the heart and brain of a computer. It receives data input, executes instructions, and processes information. It communicates with Input/Output (I/O) devices, which send and receive data to and from the CPU. Additionally, the CPU has an internal bus for communication with the internal cache memory, called the backside bus. The main bus for data transfer to and from the CPU, memory, chipset, and AGP socket is called the front side bus.</p> <p>CPU contains internal memory units, which are called registers. These registers contain data, instructions, counters, and addresses used in the ALU information processing.</p> <p>Some computers have dual or multiple processors. These consist of two or more separate physical CPUs located side-by-side on the same board or on separate boards. Each CPU has an independent interface, separate cache, and individual paths to the system front-side bus. Multiple processors are ideal for intensive parallel tasks requiring multitasking."</p>	<ul style="list-style-type: none"> - Cálculo digital - Código y sistemas de codificación - Bancos de datos - Cálculo analógico - Sistemas de control del entorno

Fuente: Tesista.

Como se pueden observar los resultados del clasificador temático, guían al usuario de la mejor manera para buscar información de los temas asociados el texto analizado, de tal

manera que el usuario puede realizar una investigación más profunda del tema que se está tratando.

Los resultados devueltos por el clasificador son coherentes con el texto analizado como se observa en las diferentes pruebas que se muestran en la Tabla 5, es más efectivo si las palabras contenidas en el texto analizado están en relación directa con la Ciencia de los Ordenadores. En el caso que no exista esta relación el clasificador tratará de enfocar los términos hacia la Ciencia de los Ordenadores buscando las palabras que tengan similitud o sean homónimas a las relacionadas con este tema obviamente la clasificación tendrá una desviación que finalmente será muy baja por el cálculo de pesos que se realiza en el algoritmo implementado.

5.5. Comportamiento del clasificador respecto a los tiempos de respuesta con textos de diferentes tamaños.

En la tabla 6 se muestra el tiempo de procesamiento del texto con un nivel de confianza 0.3 (para aumentar las anotaciones). Para este fin se ha tomado partes del siguiente texto que se halla en el sitio <https://www.pcmag.com/article2/0,2817,2390698,00.asp>, sobre un artículo de los 30 años de los sistemas operativos de los computadores personales. Este texto se encuentra en idioma inglés y tiene un total de 1730 palabras, hace referencia a la evolución de los computadores personales y se observa que en su narración se va incorporando varios términos asociados a la Ciencia de los Ordenadores relacionados directamente con la evolución que se ha tenido lo cual hace que la tarea de clasificación temática sea bastante compleja.

En la Tabla 6 se muestra el tiempo de procesamiento y respuesta por parte del clasificador según se va incrementando el número de palabras a ser analizadas que se hallan en las frases del texto referenciado.

Tabla 6. Tiempo de respuesta del clasificador en función de la longitud de texto analizado.

Longitud del texto (palabras)	Tiempo de procesamiento (segundos)
316	1.1
518	2.1
635	2.5
791	2.9

1129	3.7
------	-----

Fuente: Tesista.

Como se observa el tiempo de procesamiento y entrega de resultados al usuario final es rápido tomado en cuenta que se analiza cada palabra, el contexto, la desambiguación, el contraste con la base conocimiento interna cuyo tamaño incluye varias decenas de miles de tripletas llegando a más de 1'199.189 entradas.

A continuación, en la Figura 29 se muestra la carga del procesador y recursos de red cuando se realiza la clasificación de las 1129 palabras de la Tabla 6.



Figura 29. Carga del procesador al aplicar al clasificador un procesamiento de 1129 palabras.

Fuente: Tesista.

Como se observa en la Figura 29, entre el segundo 16 y 20 de la gráfica se observa la carga de CPU y red que no afecta a los recursos del Sistema Operativo, la carga en la memoria es imperceptible.

5.6. Conclusiones sobre los resultados obtenidos.

Las conclusiones sobre los resultados obtenidos son los siguientes:

- El proceso de clasificación temático está optimizado y presenta al usuario temas congruentes con el texto analizado.
- Se consideran los posibles eventos ante entradas no coherentes y se muestra al usuario resultados reales y correctos.
- Los tiempos de respuesta son aceptables a pesar de la carga de análisis que se debe realizar.
- La carga sobre los recursos del computador donde reside la aplicación Moodle y por ende el clasificador temático son aceptables para este tipo de procesamiento.

CONCLUSIONES

Las conclusiones sobre el desarrollo del presente trabajo se resumen en los siguientes puntos:

- Con la aplicación del presente trabajo se puede observar claramente en que consiste las redes semánticas y como se asocia el conocimiento a través de los grafos para la representación de mapas conceptuales y mentales que ayudan a las personas que estudian en las diferentes áreas del conocimiento humano a manejar toda esta gran cantidad de información y sobre todo a entender en qué consisten y en la actualidad como extraerlas aprovechando las herramientas tecnológicas existentes.
- En la construcción del clasificador se aprendió en que consiste un proceso de anotación de un texto cualquiera y asociar estas a las entradas en una base de conocimiento como DBpedia. Y como podemos aplicar estos principios para encontrar como se halla estructurada la semántica y estructura del lenguaje humano. Finalmente, como este proceso de anotación sirve para aplicarla en varias áreas de la ciencia de los computadores en la actualidad por ejemplo los buscadores como es el caso de Google.
- La experiencia al trabajar con diferentes tecnologías para el desarrollo de aplicaciones web y servicios web ha permitido enriquecer lo aprendido y palpar cómo se aplican los protocolos y estándares para el intercambio de información en diversos formatos, también la manera de optimizarlos para obtener resultados con tiempos de respuesta aceptables.
- El clasificador en su naturaleza de manera general sirve para cualquier audiencia que desee utilizarlo como fuente referencia en sus investigaciones y analizar el contenido de sus estudios para indagar a que áreas de la ciencia se orienta el mismo y determinar su área de influencia.
- Y finalmente con los conocimientos adquiridos en la carrera de Ingeniería en Informática se puede generar aplicaciones de este tipo donde se reúne todos los aspectos cubiertos por la malla curricular de la carrera.

RECOMENDACIONES Y TRABAJOS FUTUROS

Las recomendaciones y trabajos futuros se resumen en los siguientes puntos:

- Crear una interfaz para agregar más entradas a la base de conocimiento local, también se puede incluir entradas a otros idiomas como es el caso de español.
- Buscar e incorporar otros métodos de anotación diferentes a DBPedia Spotlight.
- Hacer uso de los servicios desarrollados en el proyecto de clasificación temática para aplicar en otros entornos diferentes a los de Moodle.
- Extender la clasificación temática a otras áreas del conocimiento humano diferente a la Ciencia de los ordenadores.

BIBLIOGRAFÍA

- [1] “1. Knowledge Organization Systems: An Overview — Council on Library and Information Resources”, 2016. [En línea]. Disponible en:
<http://www.clir.org/pubs/reports/pub91/contents.html/1knowledge.html>.
- [2] G. Araceli, T. Vargas, B. J. Santamaría, y A. Resumen, “el manejo de contenidos digitales Los sistemas de organización del conocimiento”, vol. 11, núm. 1, pp. 3–12, 2008.
- [3] “Sistema de Clasificación Dewey | Biblioteca Pública de Denver”, 2016. [En línea]. Disponible en: <https://bibliotecadenver.org/sistema-de-clasificacion-dewey>.
- [4] “El sistema de clasificación decimal universal”, 2016. [En línea]. Disponible en:
<http://www.taranco.eu/cdu/cdu-breve.htm>.
- [5] “SKOS: Nomenclatura de Ciencia y Tecnología de la UNESCO”, 2016. [En línea]. Disponible en: <http://skos.um.es/unesco6/>.
- [6] “Joint Academic Classification of Subjects : définition de Joint Academic Classification of Subjects et synonymes de Joint Academic Classification of Subjects (anglais)”, 2016. [En línea]. Disponible en:
[http://dictionnaire.sensagent.leparisien.fr/Joint Academic Classification of Subjects/en-en/](http://dictionnaire.sensagent.leparisien.fr/Joint+Academic+Classification+of+Subjects/en-en/).
- [7] European Union. European Centre for the Development of Vocational Training., *European training thesaurus : a multilingual synopsis*. EUR-OP, 2012.
- [8] S. Wang, Z. Chen, G. Fei, B. Liu, y S. Emery, “Targeted Topic Modeling for Focused Analysis”.
- [9] Z. Hu, G. Luo, M. Sachan, E. Xing, y Z. Nie, “Grounding Topic Models with Knowledge Bases”.
- [10] “Guía Breve de Web Semántica”. [En línea]. Disponible en:
<http://www.w3c.es/Divulgacion/GuiasBreves/WebSemantica>. [Consultado: 06-mar-2017].
- [11] “Manual de SKOS (Simple Knowledge Organization System, Sistema Simple de

- Organización del Conocimiento)", 2016. [En línea]. Disponible en:
<http://skos.um.es/TR/skos-primer/>.
- [12] "6. Vocabularios y Ontologías | Guía de Web Semántica", 2015. [En línea].
Disponible en: <http://ceweb.br/guias/web-semantica//es/capitulo-6/>.
- [13] "SPARQL 1.1 Query Language", 2013. [En línea]. Disponible en:
<https://www.w3.org/TR/sparql11-query/>.
- [14] "SparqlEndpoints - W3C Wiki", 2015. [En línea]. Disponible en:
<https://www.w3.org/wiki/SparqlEndpoints>.
- [15] M. Jimmy y H. López López, "DISEÑO DE UN AMBIENTE VIRTUAL DE
APRENDIZAJE COMO ESTRATEGIA PARA LA ENSEÑANZA DE LAS CIENCIAS
NATURALES". [En línea]. Disponible en:
http://www.virtualeduca.red/documentos/23/proyecto_virtual_educa.pdf.
- [16] "¿Qué es Moodle? | Entornos Educativos". [En línea]. Disponible en:
<http://www.entornos.com.ar/moodle>.
- [17] "MoodleDocs", 2016. [En línea]. Disponible en:
https://docs.moodle.org/dev/Main_Page.
- [18] "Arquitectura de Moodle - MoodleDocs", 2016. [En línea]. Disponible en:
https://docs.moodle.org/all/es/Arquitectura_de_Moodle.
- [19] P. N. Mendes, M. Jakob, A. García-Silva, y C. Bizer, "DBpedia Spotlight: Shedding
Light on the Web of Documents".
- [20] "esDBpedia". [En línea]. Disponible en: <http://mayor2.dia.fi.upm.es/oeg-upm/index.php/es/services/322-spanish-chapter-dbpedial/>. [Consultado: 12-mar-2017].
- [21] "DBpedia Spotlight - DigitWiki". [En línea]. Disponible en: http://succeed-project.eu/wiki/index.php/DBpedia_Spotlight.
- [22] "DBpedia Spotlight - Semantic Web Standards". [En línea]. Disponible en:
https://www.w3.org/2001/sw/wiki/DBpedia_Spotlight.
- [23] "SKOS: Nomenclatura de Ciencia y Tecnología de la UNESCO". [En línea].
Disponible en: <http://skos.um.es/unesco6/1203/html?l=es>.

- [24] J. Chicaiza, N. Piedra, J. Lopez-Vargas, y E. Tovar-Caro, "Domain Categorization of Open Educational Resources Based on Linked Data", Springer, Cham, 2014, pp. 15–28.
- [25] "Tutorial - MoodleDocs", 2016. [En línea]. Disponible en: <https://docs.moodle.org/dev/Tutorial>.
- [26] "Plugin types - MoodleDocs". [En línea]. Disponible en: https://docs.moodle.org/dev/Plugin_types.

ANEXOS

ANEXO 1. Creación de módulos (plugins) en Moodle.

Para crear plugins en Moodle se deben seguir los siguientes pasos los cuales son detallados en inglés en la página web de Moodle [25]:

1. Preparar el ambiente de desarrollo:
 - Instalar Git
 - Clonar Moodle
 - Crear la estructura y un repositorio público con GitHub
2. Crear una estructura Git para generar un seguimiento de cambios del plugin, para esto se debe realizar lo siguiente:
 - Ingrese a su cuenta GitHub
 - Cree un nuevo repositorio, se tiene una guía en el enlace: <https://help.github.com/articles/create-a-repo/>
 - Haga un enlace del directorio de su plugin a crear hacia git
 - Genere el primer commit
3. Definir el tipo de plugin Moodle a desarrollar, para ello se tiene un total de 24 tipos los cuales se pueden observar en la siguiente tabla [26]:

Tabla tipo de plugins.

Plugin type	Component name (Frankenstyle)	Moodle path	Description	Moodle versions
Activity modules	mod	/mod	Activity modules are essential types of plugins in Moodle as they provide activities in courses. For example:	1.0+

			Forum, Quiz and Assignment.	
Antivirus plugins	antivirus	/lib/antivirus	Antivirus scanner plugins provide functionality for virus scanning user uploaded files using third-party virus scanning tools in Moodle. For example: ClamAV.	3.1+
Assignment submission plugins	assignsubmission	/mod/assign/submission	Different forms of assignment submissions	2.3+
Assignment feedback plugins	assignfeedback	/mod/assign/feedback	Different forms of assignment feedbacks	2.3+
Book tools	booktool	/mod/book/tool	Small information-displays or tools that can be moved around pages	2.1+
Database fields	datafield	/mod/data/field	Different types of data that may be added to the Database activity module	1.6+
Database presets	datapreset	/mod/data/preset	Pre-defined templates for the Database activity module	1.6+
LTI sources	ltisource	/mod/lti/source	LTI providers can be added to external tools easily through the external tools interface see Documentation on External Tools . This type of plugin is specific to LTI providers that need a plugin that can register custom handlers to process LTI messages	2.7+

<u>LTI services</u>	ltiservice	/mod/lti/service	Allows the implementation of LTI services as described by the IMS LTI specification	2.8+
<u>Quiz reports</u>	quiz	/mod/quiz/report	Display and analyse the results of quizzes, or just plug miscellaneous behaviour into the quiz module	1.1+
<u>Quiz access rules</u>	quizaccess	/mod/quiz/accessrule	Add conditions to when or where quizzes can be attempted, for example only from some IP addresses, or student must enter a password first	2.2+
<u>SCORM reports</u>	scormreport	/mod/scorm/report	Analysis of SCORM attempts	2.2+
<u>Workshop grading strategies</u>	workshopform	/mod/workshop/form	Define the type of the grading form and implement the calculation of the grade for submission in the <u>Workshop</u> module	2.0+
<u>Workshop allocation methods</u>	workshopallocation	/mod/workshop/allocation	Define ways how submissions are assigned for assessment in the <u>Workshop</u> module	2.0+
<u>Workshop evaluation methods</u>	workshopeval	/mod/workshop/eval	Implement the calculation of the grade for assessment (grading grade) in the <u>Workshop</u> module	2.0+
<u>Blocks</u>	block	/blocks	Small information-displays or tools that can be moved around pages	2.0+

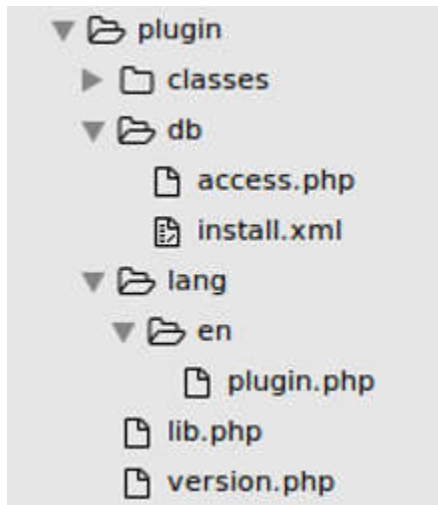
<u>Question types</u>	qtype	/question/type	Different types of question (e.g. multiple-choice, drag-and-drop) that can be used in quizzes and other activities	1.6+
<u>Question behaviours</u>	qbehaviour	/question/behaviour	Control how student interact with questions during an attempt	2.1+
<u>Question import/export formats</u>	qformat	/question/format	Import and export question definitions to/from the question bank	1.6+
<u>Text filters</u>	filter	/filter	Automatically convert, highlight, and transmogrify text posted into Moodle.	1.4+
<u>Editors</u>	editor	/lib/editor	Alternative text editors for editing content	2.0+
<u>Atto editor plugins</u>	atto	/lib/editor/atto/plugins	Extra functionality for the Atto text editor	2.7+
<u>TinyMCE editor plugins</u>	tinymce	/lib/editor/tinymce/plugins	Extra functionality for the TinyMCE text editor.	2.4+
<u>Enrolment plugins</u>	enrol	/enrol	Ways to control who is enrolled in courses	2.0+
<u>Authentication plugins</u>	auth	/auth	Allows connection to external sources of authentication	2.0+
<u>Admin tools</u>	tool	/admin/tool	Provides utility scripts useful for various site administration and maintenance tasks	2.2+
<u>Log stores</u>	logstore	/admin/tool/logstore	Event logs storage back-ends	2.7+
<u>Availability conditions</u>	availability	/availability/condition	Conditions to restrict user access to activities and sections.	2.7+
<u>Calendar types</u>	calendartype	/calendar/type	Defines how dates are displayed throughout Moodle	2.6+

<u>Messaging consumers</u>	message	/message/output	Represent various targets where messages and notifications can be sent to (email, sms, jabber, ...)	2.0+
<u>Course formats</u>	format	/course/format	Different ways of laying out the activities and blocks in a course	1.3+
<u>Data formats</u>	dataformat	/dataformat	Formats for data exporting and downloading	3.1+
<u>User profile fields</u>	profilefield	/user/profile/field	Add new types of data to user profiles	1.9+
<u>Reports</u>	report	/report	Provides useful views of data in a Moodle site for admins and teachers	2.2+
<u>Course reports</u>	coursereport	/course/report	Reports of activity within the course	Up to 2.1 (for 2.2+ see <u>Reports</u>)
<u>Gradebook export</u>	gradeexport	/grade/export	Export grades in various formats	1.9+
<u>Gradebook import</u>	gradeimport	/grade/import	Import grades in various formats	1.9+
<u>Gradebook reports</u>	gradereport	/grade/report	Display/edit grades in various layouts and reports	1.9+
<u>Advanced grading methods</u>	gradingform	/grade/grading/form	Interfaces for actually performing grading in activity modules (eg Rubrics)	2.2+
<u>MNet services</u>	mnet-service	/mnet/service	Allows to implement remote services for the <u>MNet</u> environment (deprecated, use web services instead)	2.0+
<u>Webservice protocols</u>	webservice	/webservice	Define new protocols for web service communication (such as	2.0+

			SOAP, XML-RPC, JSON, REST ...)	
Repository plugins	repository	/repository	Connect to external sources of files to use in Moodle	2.0+
Portfolio plugins	portfolio	/portfolio	Connect external portfolio services as destinations for users to store Moodle content	1.9+
Search engines	search	/search/engine	Search engine backends to index Moodle's contents.	3.1+
Media players	media	/media/player	Pluggable media players	3.2+
Plagiarism plugins	plagiarism	/plagiarism	Define external services to process submitted files and content	2.0+
Cache store	cachestore	/cache/stores	Cache storage backends.	2.4+
Cache locks	cachelock	/cache/locks	Cache lock implementations.	2.4+
Themes	theme	/theme	Change the look of Moodle by changing the the HTML and the CSS.	2.0+
Local plugins	local	/local	Generic plugins for local customisations	2.0+
Legacy assignment types	assignment	/mod/assignment /type	Different forms of assignments to be graded by teachers	1.x - 2.2
Legacy admin reports	report	/admin/report	Provides useful views of data in a Moodle site, for admins only.	Up to 2.1 (for 2.2+ see Reports)

Fuente: www.moodle.org

4. Construir la estructura base del plugin, mostrada en la siguiente figura:



Los elementos de esta estructura son:

- **Plugin**, es el nombre del plugin. Esto también se aplica al archivo en "lang / en /".
- **Classes**, contiene clases usadas en el plugin.
- **Db**, tiene archivos relacionados con la base de datos y también capacidades (access.php).
- **Lang**, El directorio de idiomas.
- **Lib.php**, archivo de biblioteca base para algunos ganchos y devoluciones de llamada.
- **Version.php**, contiene información básica sobre el plugin.

5. Definir la estructura básica de la página principal del plugin, la cual consta de las siguientes secciones:

- Required files.

- POST and GET variables retrieved using `required_param` and `optional_param`.
- context.
- navigation setup.
- title, headings, headers.
- content.
- footers.

Esto se observa en el siguiente ejemplo de código:

```
// Required files.
require_once(__DIR__ . '/config.php');
// Required and optional parameters.
$id = required_param('id', PARAM_INT);
$text = optional_param('text', '', PARAM_ALPHA);

// Setting context for the page.
$PAGE->set_context(context_system::instance());
// URL is created and then set for the page navigation.
$url = new moodle_url('/test.php');
$PAGE->set_url($url);
// Heading, headers, page layout.
$PAGE->set_heading(get_string('hello'));
$PAGE->set_pagelayout('standard');
echo $OUTPUT->header();
// Displaying basic content.
echo html_writer::tag('p', $text);
// Display the footer.
echo $OUTPUT->footer();
```

6. Definir el o los lenguajes que manejará el plugin, por ejemplo, inglés, español, alemán, etc., para esto refiérase al link https://docs.moodle.org/31/en/Language_customisation

7. Añadir el contenido a la página principal del plugin, para esto hay que tener en cuenta las siguientes consideraciones:

- El contenido de una página se agrega a través de renderizadores.
- Los procesadores normalmente se almacenan en el directorio "classes / output".
- Poner contenido en renderers permite a los usuarios anular la visualización visual del contenido.
- La información muy básica se presenta usando la clase `html_writer`.
- En la mayoría de los casos se deben usar plantillas.
- Las plantillas se almacenan en el directorio "plantillas". Las plantillas usan archivos de bigotes.
- Los archivos de bigotes permiten html más genérico con marcadores de posición insertados, que inserta los datos (contexto) en tiempo de ejecución.

Para mayor información refiérase a los siguientes links:

- https://docs.moodle.org/dev/Output_API
- https://docs.moodle.org/dev/Output_functions
- <https://docs.moodle.org/dev/Templates>
- https://docs.moodle.org/dev/Output_renderers
- <https://docs.moodle.org/dev/Renderer>
- https://docs.moodle.org/dev/Overriding_a_renderer

8. Añadir el plugin al sistema de navegación de Moodle, con las siguientes consideraciones:

- El sistema de navegación Moodle tiene ganchos que permite a los complementos añadir enlaces al menú de navegación.
- Los ganchos se encuentran en lib.php. Trate de mantener lib.php lo más pequeño posible, ya que este archivo se incluye en cada página. Poner clases y funciones en otros lugares.

9. Utilizar las extensiones si es necesario para el manejo de base de datos con las siguientes consideraciones:

- Moodle tiene una biblioteca genérica de consultas de bases de datos. Detrás de esta biblioteca hay bibliotecas adicionales que permiten a Moodle trabajar con MySQL, PostgreSQL, Oracle, SQL Server y Maria DB.
- Cuando sea posible, es aconsejable utilizar las funciones predefinidas en lugar de escribir SQL. Escribir SQL tiene una mayor probabilidad de no trabajar con una de las bases de datos admitidas.
- El retorno de las funciones selectas tiende a ser un objeto o una matriz de objetos.

Para mayor información refiérase a los siguientes links:

- <https://docs.moodle.org/dev/Database>
- https://docs.moodle.org/dev/Data_manipulation_API

10. Crear definiciones propias de tablas si es requerido, para ello se debe tener en cuenta lo siguiente:

- Crear las tablas de base de datos en Moodle usando el editor XMLDB. Esto se encuentra en el bloque de administración "Administración del sitio | Desarrollo | Editor XMLDB".

- El directorio {plugin} \ db necesita tener acceso de escritura para que el editor XMLDB sea más efectivo.
- El editor XMLDB crea un archivo install.xml en el directorio db. Este archivo se cargará durante la instalación para crear las tablas.
- El editor XMLDB producirá código de actualización de php para agregar y actualizar tablas de base de datos de Moodle.

Para mayor información refiérase a los siguientes links:

- https://docs.moodle.org/dev/XMLDB_editor
- https://docs.moodle.org/dev/Using_XMLDB
- https://docs.moodle.org/dev/XMLDB_Documentation
- https://docs.moodle.org/dev/Upgrade_API
- https://docs.moodle.org/dev/XMLDB_introduction

11. Definir el control de accesos del plugin (privilegios), hay que tomar en cuenta los siguientes criterios:

- Los privilegios se controlan en "access.php" en el directorio "db".
- Este archivo tiene una serie de privilegios para lo siguiente:
 - Nombre
 - Posibles riesgos de seguridad detrás de dar un privilegio.
 - El contexto en el que el privilegio funciona.
- Los roles predeterminados (maestro, administrador, estudiante, etc.) que tienen los privilegios.
- Otra información.
- Estas posibilidades de privilegios se comprueban en código para permitir el acceso a páginas, secciones y capacidades (guardar, borrar, etc.).

Para mayor información refiérase a los links:

- https://docs.moodle.org/dev/Access_API
- <https://docs.moodle.org/dev/Roles#Context>
- <https://docs.moodle.org/31/en/Category:Capabilities>
- https://docs.moodle.org/31/en/Roles_and_permissions

12. Añadir Web Forms si es necesario para el ingreso de información por parte del usuario del plugin, tomar en cuenta que:

- Moodle tiene su propia biblioteca de formularios.
- El lib de formularios incluye una gran cantidad de código de accesibilidad y comprobación de errores de forma predeterminada.
- Los formularios Moodle se pueden mostrar en JavaScript usando 'fragmentos'

Para mayor información refiérase a los links:

- https://docs.moodle.org/dev/Form_API
- https://docs.moodle.org/dev/lib/formslib.php_Form_Definition
- <https://docs.moodle.org/dev/Fragment>

13. Mantener una buena seguridad:

- Utilice el sesskey al dirigir a las páginas para hacer las acciones.
- Utilice los filtros apropiados al recuperar los parámetros

14. Para el manejo de archivos hay que tomar en cuenta que:

- Los archivos se almacenan conceptualmente en áreas de archivo.
- Los plugins sólo pueden acceder a los archivos desde su propio componente.

15. Utilizar javascript si es requerido con las siguientes consideraciones:

- Moodle está utilizando jquery y AMD (Asynchronous Module Definition).
- Los archivos JavaScript se encuentran en el directorio "amd / src".
- Usa grunt para construir tu JavaScript.
- Incluya su JavaScript en archivos php de la siguiente manera:

```
$this->page->requires->js_call_amd('{JScriptfilename}');
```

- También se puede incluir en plantillas de bigote.

Para mayor referencia refiérase a los links:

- https://docs.moodle.org/dev/Javascript_Modules
- <https://docs.moodle.org/dev/jquery>
- https://docs.moodle.org/dev/Javascript_FAQ
- https://docs.moodle.org/dev/JavaScript_guidelines
- <https://docs.moodle.org/dev/Grunt>

16. Utilizar el manejo de eventos y log de transacciones si es necesario, con los siguientes criterios:

- Todo el registro en Moodle se realiza a través del sistema de eventos.
- Los nuevos eventos deben estar ubicados en el directorio "classes/events".
- Es posible crear observadores y suscribirse a eventos.

Para mayor información refiérase a los links:

- https://docs.moodle.org/dev/Event_2
- https://docs.moodle.org/dev/Logging_2

17. Utilizar webservices y Ajax si es necesario, para lo cual considerar que:

- Los servicios web de Moodle utilizan la API de funciones externas.
- La forma recomendada de realizar solicitudes AJAX es utilizar el archivo ajax AMD que utiliza las funciones externas API.
- Las funciones externas deben estar ubicadas en el archivo "classes / external.php".
- Una lista de servicios debe incluirse en "db / services.php". Este archivo se requiere para registrar los servicios web con Moodle.
- La lista de servicios es una matriz que contiene:
 - classname: Nombre de la clase externa.
 - methodname: Nombre de la función externa.
 - classpath: ruta del sistema al archivo de función externa.
 - description: Descripción de la función
 - type: Crear, leer, actualizar, eliminar

- ajax: ¿Se puede usar esta función con ajax?
- capabilities: Capacidades necesarias para utilizar esta función.

Para mayor información refiérase a los links:

- https://docs.moodle.org/dev/External_functions_API
- https://docs.moodle.org/dev/Adding_a_web_service_to_a_plugin
- https://docs.moodle.org/dev/Web_services_API

18. Es recomendable utilizar cache para mejorar el rendimiento del plugin, para esto:

- El método utilizado por Moodle para el cache es: Moodle Universal Cache (MUC).
- MUC tiene las siguientes definiciones:
 - Request cache
 - Session cache
 - Application

Para mayor referencia refiérase a los links:

- [https://docs.moodle.org/dev/The_Moodle_Universal_Cache_\(MUC\)](https://docs.moodle.org/dev/The_Moodle_Universal_Cache_(MUC))
- https://docs.moodle.org/31/en/MUC_FAQ
- <https://docs.moodle.org/31/en/Caching>

19. Establecer soporte para backup y restore:

- El respaldo de copia de seguridad y restauración requiere crear varios archivos en el directorio 'backup / moodle2'.

- Restore requiere una clase para extender la tarea de copia de seguridad de algún tipo. Puede que exista una tarea de complemento específica.
- Restore requiere una clase para extender la restore_task de algún tipo. Puede que exista una tarea de complemento específica.
- Los pasos de restauración lib definen la estructura del complemento que se va a restaurar.
- Los pasos de copia de seguridad lib definen pasos, configuraciones, atributos, etc.

Para mayor información refiérase al link:

- https://docs.moodle.org/31/en/Backup_and_restore_FAQ

20. Establecer soporte para pruebas de testeo, para ello tomar en cuenta que:

- Moodle tiene dos tipos de pruebas automatizadas: pruebas de unidad de php, y pruebas de behat.
- Las pruebas unitarias son para probar funciones.
- Las pruebas de Behat se ejecutan a través de escenarios.
- Las pruebas de Behat siguen un guión y navegan a través de las páginas de moodle.
- Las pruebas unitarias deben ubicarse en el directorio "tests".
- Las pruebas de behat deben ubicarse en el directorio "tests / behat".
- Las pruebas ubicadas en estos directorios se ejecutarán cuando se inicie una prueba completa.
- Las pruebas del behat son realmente archivos de la característica y terminan con la extensión ".feature".

Para mayor información refiérase a los links:

- https://docs.moodle.org/dev/Writing_PHPUnit_tests
- <https://docs.moodle.org/dev/PHPUnit>
- https://docs.moodle.org/dev/Acceptance_testing
- https://docs.moodle.org/dev/Behat_integration

21. Añadir el plugin a Moodle.org, para ello realizar lo siguiente:

- Publicar sus plugins en <https://moodle.org/plugins/>
- Los plugins de publicación en el sitio de moodle se realiza a través de una serie de pasos que deben completarse para que el complemento sea aprobado y publicado.
- Los plugins se ejecutarán a través de un pre-checker para dar sugerencias sobre posibles problemas con el código.

22. Mantener un canal de soporte sobre el plugin.

ANEXO 2. Código para mostrar la interfaz de usuario para la lectura del documento o texto de entrada.

El siguiente es el código que se desarrolló para mostrar la interfaz de usuario para la lectura del documento o texto de entrada por parte del usuario para obtener la clasificación temática:

```
<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
```

```

//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * Prints a particular instance of thematicclasificator
 *
 * You can have a rather longer description of the file as well,
 * if you like, and it can span multiple lines.
 *
 * @package mod_thematicclasificator
 * @copyright 2017 Jose Balseca <jbalsecac@gmail.com>
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

// Replace thematicclasificator with the name of your module and remove this line.

require_once(dirname(dirname(dirname(__FILE__))).'/config.php');
require_once(dirname(__FILE__).'./lib.php');

$id = optional_param('id', 0, PARAM_INT); // Course_module ID, or
$n = optional_param('n', 0, PARAM_INT); // ... thematicclasificator instance ID - it should be named as the first
character of the module.

if ($id) {
    $cm = get_coursemodule_from_id('thematicclasificator', $id, 0, false, MUST_EXIST);
    $course = $DB->get_record('course', array('id' => $cm->course), '*', MUST_EXIST);
    $thematicclasificator = $DB->get_record('thematicclasificator', array('id' => $cm->instance), '*', MUST_EXIST);
} else if ($n) {
    $thematicclasificator = $DB->get_record('thematicclasificator', array('id' => $n), '*', MUST_EXIST);
    $course = $DB->get_record('course', array('id' => $thematicclasificator->course), '*', MUST_EXIST);
    $cm = get_coursemodule_from_instance('thematicclasificator', $thematicclasificator->id, $course->id, false,
MUST_EXIST);
} else {
    error('You must specify a course_module ID or an instance ID');
}

require_login($course, true, $cm);

$event = \mod_thematicclasificator\event\course_module_viewed::create(array(
    'objectid' => $PAGE->cm->instance,

```

```

'context' => $PAGE->context,
));
$event->add_record_snapshot('course', $PAGE->course);
$event->add_record_snapshot($PAGE->cm->modname, $thematicclasificator);
$event->trigger();

// Print the page header.

$PAGE->set_url('/mod/thematicclasificator/view.php', array('id' => $cm->id));
$PAGE->set_title(format_string($thematicclasificator->name));
$PAGE->set_heading(format_string($course->fullname));

/*
 * Other things you may want to set - remove if not needed.
 * $PAGE->set_cacheable(false);
 * $PAGE->set_focuscontrol('some-html-id');
 * $PAGE->add_body_class('thematicclasificator-'. $somevar);
 */

// Output starts here.
echo $OUTPUT->header();

// Conditions to show the intro can change to look for own settings or whatever.
if ($thematicclasificator->intro) {
    echo $OUTPUT->box(format_module_intro('thematicclasificator', $thematicclasificator, $cm->id), 'generalbox
mod_introbox', 'thematicclasificatorintro');
}

// Replace the following lines with you own code.
//echo $OUTPUT->heading('Yay! It works!');
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">

<head><meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Clasificación Temática Recursos Web</title>

    <script src="lib/selector.js" type="text/javascript"></script>
    <script src="lib/jquery-1.6.1.min.js" type="text/javascript"></script>
    <script src="lib/dbpedia-spotlight-0.3.js" type="text/javascript" charset="utf-8"></script>
    <script src="lib/jquery.getUrlParam.js" type="text/javascript"></script>
    <script src="lib/ui/jquery.ui.core.js" type="text/javascript"></script>
    <script src="lib/ui/jquery.ui.widget.js" type="text/javascript"></script>
    <script src="lib/ui/jquery.ui.mouse.js" type="text/javascript"></script>
    <script src="lib/ui/jquery.ui.slider.js" type="text/javascript"></script>

```

```

<script src="lib/ui/jquery.ui.dialog.js" type="text/javascript"></script>
<script src="lib/ui/jquery.ui.tabs.js" type="text/javascript"></script>
<script src="lib/ui/jquery.ui.position.js" type="text/javascript" charset="utf-8"></script>
<script src="lib/ui/jquery.ui.draggable.js" type="text/javascript" charset="utf-8"></script>
<script src="lib/jquery.uniform/jquery.uniform.min.js" type="text/javascript"></script>
<script src="lib/jquery.json-2.2.js" type="text/javascript"></script>
<script src="lib/jstree/jquery.jstree.js" type="text/javascript" charset="utf-8"></script>
<script src="lib/autoresize.jquery.min.js" type="text/javascript" charset="utf-8"></script>
<link rel="stylesheet" href="themes/base/jquery.ui.all.css">
<link rel="stylesheet" href="lib/jquery.uniform/css/uniform.default.css" type="text/css" media="screen" charset="utf-8"
/>
<link rel="stylesheet" href="default.css">
<script charset="UTF-8" src="default.js" type="text/javascript"></script>
</head>

<body>
<div class="main">

<div id="action_container" class="container">
  <div id="action_options">
    <div id="slider_container" class="container">
      <div class="option">
        <table>
          <tr>
            <td> <label for="confidence">Confianza:</label></td>
            <td><input type="text" id="confidence" name="confidence" value="0.3" style="margin: 0px"></td>
            <td><div id="confidence-slider" class="slider"></div></td>
          </tr>
        </table>
      </div>
    </div>
    <!--<input type="checkbox" id="nbest" /><label for="nbest">mejores resultados</label> -->
    <!--<select id="disambiguator" class="action_button">
      <option value="Default" selected>Default Disambiguation</option>
      <option value="Occurrences">Occurrence-centric</option>
      <option value="Document">Document-centric</option>
    </select>
    <br />
  </div>
  <div id="action_language">
    <label for="annotation_language">Idioma:</label>
    <select id="annotation_language" class="action_button">
    </select>
  </div>
</div>
<div id="options_container" class="container">

```

```

<div id="action_buttons">
  <input class="action_button" id="clear" type="button" value="Limpiar"/>
  <input class="action_button" id="types" type="button" value="Seleccione tipo aplicación"/>
  <input class="action_button" id="annotate" type="button" value="Procesar"/>
  <div id="status_indicator">&nbsp;</div>
</div>
</div>
<div id="text_container" class="container">
  <textarea rows="10" id="text" cols="1"></textarea>
  <div id="edit_text_wrapper">
    <input id="edit_text" type="button" value="Regresar al texto"/>
  </div>
</div>
<div id="filter_infobox" class="container" style="display: none;">
</div>

<!-- TODO input id="view_source" type="button" value="View Source (RDFa annotations)"/-->
<div id=suggestdialog></div>
<div id="instructions" class="container">
  <h2>Importante:</h2>
  <ul>
    <li>Es posible utilizar el clasificador como <a id="bookmarklet_link"
href="#bookmarklet_container">bookmark</a>, para esto siga las instrucciones del link</li>
  </ul>
</div>

<div id="bookmarklet_container" class="container">
  <div>
    <br/>
    Para utilizar el clasificador como bookmark<br/>
    Mueva el botón "Clasificador temático recursos web" de abajo a la barra de bookmarks (O botón derecho y
seleccione "Bookmark this link").
    Después de esto, simplemente seleccione el texto deseado en cada página y haga clic sobre el bookmark
creado, con ello se puede realizar la clasificación.
    <div class="bookmarklet">
      <p>
        <a href="javascript:function se(d) {return d.selection ? d.selection.createRange().text : d.getSelection()} s =
se(document); for (i=0; i<frames.length && !s; i++) s = se(frames[i].document); if (!s || s=="") s =
prompt("Enter%20text%20to%20annotate%20with%20DBpedia%20Spotlight",""); open('<?php echo
!empty($_SERVER["HTTPS"]) ? 'https' : 'http' : '//'. $_SERVER["HTTP_HOST"] .
$_SERVER["REQUEST_URI"]?>index.php' + (s ? '?execute=yes&client=bookmarklet&text=' + encodeURIComponent(s)
: ")).focus();">
          Clasificador temático recursos web
        </a>
      </p>
    </div>
  </div>
</div>

```

```

</div>
</div>

<div id="type_selector">
  <div id="filterPolicy">
    <input id="filterPolicyWL" type="radio" name="filterPolicy" value="whitelist" checked>Solamente estos
tipos</input>
    <input id="filterPolicyBL" type="radio" name="filterPolicy" value="blacklist">Ninguno de estos tipos</input>
  </div>

  <div class="tabs">
    <ul>
      <li><a href="#tabs-1">DBpedia</a></li>
      <li><a href="#tabs-2">Freebase</a></li>
      <li><a href="#tabs-3">Schema.org</a></li>
      <li><a href="#tabs-4">Consulta (SPARQL)</a></li>
    </ul>

    <div id="tabs-1">
      <ul>
        <li id="tree_dbpedia" class="tree" />
      </ul>
    </div>

    <div id="tabs-2">
      <ul>
        <li id="tree_freebase" class="tree" />
      </ul>
    </div>

    <div id="tabs-3">
      <ul>
        <li id="tree_schema" class="tree" />
      </ul>
    </div>

    <div id="tabs-4">
      <textarea id="sparql_type" rows="8" cols="1"></textarea>
      <div id="sparqlExamples">
        <strong>Ejemplo de consultas:</strong>
        <a href="#" onclick="sparqlExample(1);">Políticos de Chicago</a>,
        <a href="#" onclick="sparqlExample(2);">Personas nacidas en Berlín antes de 1900</a>,
        <a href="#" onclick="sparqlExample(3);">Películas francesas</a>.
      </div>
    </div>
  </div>

```

```

</div>
</div>

<div class=copyright>
  Este trabajo utiliza:
  <p><a href=http://plugins.jquery.com/project/dbpedia-spotlight>DBpedia Spotlight jQuery Plugin v0.3</a>.
  <br/> Para la últimas versiones visite: <a href=http://spotlight.dbpedia.org>http://spotlight.dbpedia.org</a>
  <br/>Se utiliza los servicios web de DBPedia <span id="webservice">http://spotlight.sztaki.hu:2222/rest</span>.
  </p>
</div>

<br/>

</body>
</html>

<?php
// Finish the page.
  echo $OUTPUT->footer();
?>

```

ANEXO 3. Código para realizar el proceso de extracción de contenido y anotación semántica.

El siguiente es el código que se desarrolló para realizar el proceso de extracción de contenido y anotación semántica utilizando los servicios de DBpedia Spotlight:

```

/*
 * Copyright 2011 Pablo Mendes, Max Jakob
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 *
 * Check our project website for information on how to acknowledge the authors and how to contribute to the project:
 * http://spotlight.dbpedia.org
 */

/**
 * If you use this script, please give credit to DBpedia Spotlight by:
 * - adding "powered by DBpedia Spotlight" somewhere in the page.
 * - add a link to http://spotlight.dbpedia.org.
 *
 * TODO people can also want to change the boundaries of the spotting

```

```

* TODO showScores = [ list of score names to display ]
*
* @author pablomendes
* @adapded by Jose Balseca
*/

// adding this in case we forget to remove some console.log statement that would make it break on firefox without firebug.

/*if (console === undefined) {
  console = {
    "log": function () {
    }
  }
}*/

//from http://www.wrichards.com/blog/2009/02/jquery-sorting-elements/

jQuery.fn.sort = function () {
  return this.pushStack([].sort.apply(this, arguments), []);
};

function sortOffset(a, b) {

  if (a["@offset"] == b["@offset"]) {
    return 0;
  }
  return parseInt(a["@offset"]) > parseInt(b["@offset"]) ? 1 : -1;
}

(function ($) {
  var powered_by = "<div style='font-size: 9px; float: right'><a href='http://spotlight.dbpedia.org'>Powered by DBpedia Spotlight</a></div>";

  var settings = {
    'endpoint': 'http://localhost:2223/rest',
    'confidence': 0.5, //
    'support': 0,
    'powered_by': 'yes', // yes or no
    'showScores': 'yes', // yes or no
    'types': "",
    'policy': 'whitelist',
    'spotter': 'LingPipeSpotter', // one of: LingPipeSpotter,AtLeastOneNounSelector,CoOccurrenceBasedSelector
    'disambiguator': 'Default' // one of: LingPipeSpotter,AtLeastOneNounSelector,CoOccurrenceBasedSelector
  };

  function getScoreDOMElements(data) {
    var li = "";
    for (var key in data)
      li += "<span class='hidden " + key + "'>" + data[key] + "</span>";
    return li;
  }

  var Parser = {
    getSelectBox: function (resources, className) {
      var ul = $("<ul class='" + className + "s'></ul>");
      $.each(resources, function (i, r) {
        var li = "<li class='" + className + " " + className + "-" + i + "'><a href='" + res_prefix + r["@uri"] + "' about='" + r["@uri"] + "'>" + r["@label"] + "</a>";

        //TODO settings.showScores = ["finalScore"] foreach showscores, add k=v

        if (settings.showScores == 'yes') {

```



```

    li += " (<span class='finalScoreDisplay'>" + parseFloat(r["@finalScore"]).toFixed(3) + "</span>");
  }

  li += getScoreDOMElements({
    "finalScore": parseFloat(r["@finalScore"]).toFixed(3),
    "contextualScore": parseFloat(r["@contextualScore"]),
    "percentageOfSecondRank": parseFloat(r["@percentageOfSecondRank"]),
    "support": parseFloat(r["@support"]),
    "priorScore": parseFloat(r["@priorScore"])
  });

  li += "</li>";
  var opt = $(li);
  $(ul).append(opt);
});

return ul;
},

getAnnotatedText: function (response) {
  var json;
  if (typeof response == 'object') {
    json = response;
  } else {
    json = $.parseJSON(response);
  }

  var text = json["annotation"]["@text"];
  var start = 0;
  var annotatedText = text;
  var annotations = [];

  if (json.annotation['surfaceForm'] != undefined) {

    annotations = annotations.concat(json.annotation.surfaceForm).sort(sortOffset); // deals with the case of only one
    surfaceForm returned (ends up not being an array)

  }
  annotatedText = annotations.map(function (e) {

    if (e == undefined) return "";
    //console.log(e);
    var sfName = e["@name"];
    var offset = parseInt(e["@offset"]);
    var sfLength = parseInt(sfName.length);
    var snippet = text.substring(start, offset);
    var surfaceForm = text.substring(offset, offset + sfLength);
    start = offset + sfLength;

    var classes = "annotation";

    snippet += "<div id='" + (sfName + offset) + "' class='" + classes + "'><a class='surfaceForm'>" + sfName + "</a>";
    var ul = Parser.getSelectBox($(e.resource), 'candidate');

    //ul.children().each(function() { console.log($.data(this), "testProp"); });

    snippet += "<ul class='candidates'>" + ul.html() + "</ul>"; //FIXME this wrapper is a repeat from getSelectBox
    snippet += "</div>";
    return snippet;

  }).join("");

  //snippet after last surface form
  annotatedText += text.substring(start, text.length);

  //console.log(annotatedText);
  return annotatedText.replace(/\n/g, "<br />");
}

```

```

},

getAnnotatedTextFirstBest: function (response) {

    var json;
    if (typeof response == 'object') {
        json = response;
    } else {
        json = $.parseJSON(response);
    }

    var text = json["@text"];
    var start = 0;
    var annotatedText = text;
    var annotations = [];

    if (json['Resources'] != undefined) {
        annotations = annotations.concat(json['Resources']); // deals with the case of only one surfaceFrom returned
        (ends up not being an array)
    }

    var currurl=window.location.href.match(/^.*/);

    // clear entries for a new UNESCO classification process
    $.ajax({
        type: "POST",
        url: currurl+"borraclasificador.php",
        async: false,
        dataType: 'json',
        success: function(data)
        {
            resultado = data[0];
        },
        error: function(xhr, desc, err){
            alert("Hay un error borrado clasificador:" + xhr.responseText);
        }
    });
    annotatedText = annotations.map(function (e) {
        if (e == undefined) return "";
        var sfName = e["@surfaceForm"];
        var offset = parseInt(e["@offset"]);
        var uri = e["@URI"];

        var sfLength = parseInt(sfName.length);
        var snippet = text.substring(start, offset);
        var surfaceForm = text.substring(offset, offset + sfLength);
        var ranking = e["@similarityScore"];
        start = offset + sfLength;

        var support = parseInt(e["@support"]);
        var confidence = 0.5;
        if (settings.its == 'yes') {
            snippet += "<span id=\"" + (sfName + offset) +
                "\" its-ta-ident-ref=\"" + uri +
                "\" its-ta-confidence=\"" + parseFloat(e["@similarityScore"]).toFixed(3) + "\">" + sfName;
            snippet += "</span>";
        } else {
            var label="";

            // Contrast with knowledge database local
            $.ajax({
                type: "POST",
                url: currurl+"reconocimiento.php",
                data: "subject="+uri+"&ranking="+ranking,
                async: false,
                dataType: 'json',
                success: function(data)
                {
                    label = data[1];
                }
            });
        }
    });
}

```

```

    },
    error: function(xhr, desc, err){
        alert("Hay un error en validación tripletas:" + xhr.responseText);
    }
});

if(label.length > 0) {
    var classes = "annotation";
    snippet += "<a id=\"" + (sfName + offset) + "\" class=\"" + classes + "\" about=\"" + uri + "\" href=\"" + uri + "\" title=\""
+ uri + "\">" + sfName;

    snippet += getScoreDOMElements({
        "finalScore": parseFloat(e["@similarityScore"]).toFixed(3),
        "contextualScore": parseFloat(e["@similarityScore"]).toFixed(3), //TODO send contextualScore from
server and grab here
        "percentageOfSecondRank": parseFloat(e["@percentageOfSecondRank"]),
        "support": parseFloat(e["@support"])
    });

    snippet += "</a>";
} else {
    snippet += sfName;
}

return snippet;
}).join("");
//snippet after last surface form
annotatedText += text.substring(start, text.length);

var topics="";

// classification process using UNESCO entries
$.ajax({
    type: "POST",
    url: currurl+"clasificador.php",
    async: false,
    dataType: 'json',
    success: function(data)
    {
        for ( i = 0; i < data.length; i++ )
        {
            if(data[i].topic != "") {
                topics += " " + "<a href=\"" + data[i].unesco_uri + "\" about=\"" + data[i].unesco_uri + "\">" + data[i].topic +
"</a>" + "\n";
            }
        }
    },
    error: function(xhr, desc, err){
        alert("Hay un error en validación tripletas:" + xhr.responseText);
    }
});

if (topics.length > 0){
    annotatedText="Los tópicos relacionados al texto seleccionado asociados al tesoro de la UNESCO son: \n\n"
+ topics + "\n\n" + annotatedText;
} else {
    annotatedText = "No se encontraron tópicos relacionados al texto seleccionado. \n\n" + annotatedText;
}
//console.log(annotatedText);
return annotatedText.replace(/\n/g, "<br />\n");
},
getSuggestions: function (response, targetSurfaceForm) {
    var json;
    if (typeof response == 'object') {
        json = response;
    } else {
        json = $.parseJSON(response);
    }
}

```

```

    }
    var suggestions = "";
    if (json.annotation['surfaceForm'] != undefined) {
        var annotations = [].concat(json.annotation.surfaceForm); // deals with the case of only one surfaceForm returned
        (ends up not being an array)
        suggestions = annotations.filter(function (e) {
            return (e["@name"] == $.trim(targetSurfaceForm));
        }).map(function (e) {
            return Parser.getSelectBox($(e.resource), 'none');
        }).join("");
    }
    return suggestions;
}
};

var ajaxRequest;
var methods = {
    init: function (options) {
        // If options exist, lets merge them with our default settings
        if (options) {
            $.extend(settings, options);
        }
    },
    best: function (options) {
        //init(options);
        function update(response) {

            var content = "<div>" + Parser.getAnnotatedTextFirstBest(response) + "</div>";
            if (settings.powered_by == 'yes') {
                $(content).append($(powered_by));
            }
            $(this).html(content);

            if (settings.callback != undefined) {
                settings.callback(response);
            }
        }

        return this.each(function () {
            //console.log($.quoteString($(this).text()));
            var params = {
                'text': $(this).text(),
                'confidence': settings.confidence,
                'support': settings.support,
                'spotter': settings.spotter,
                'disambiguator': settings.disambiguator,
                'policy': settings.policy
            };
            if ("types" in settings && settings["types"] != undefined)
                params["types"] = settings.types;
            if ("sparql" in settings && settings["sparql"] != undefined)
                params["sparql"] = settings.sparql;

            ajaxRequest = $.ajax({
                'url': settings.endpoint + "/annotate",
                'data': params,
                'context': this,
                'headers': {'Accept': 'application/json'},
                'success': update,
                'error': function (response) {
                    if (settings.callback != undefined) {
                        settings.callback(response);
                    }
                }
            });
        });
    },
    candidates: function (options) {
        //init(options);

```

```

function update(response) {
    var content = "<div>" + Parser.getAnnotatedText(response) + "</div>";

    if (settings.powered_by == 'yes') {
        $(content).append($(powered_by));
    }
    $(this).html(content);

    if (settings.callback != undefined) {
        settings.callback(response);
    }
}

return this.each(function () {
    var params = {
        'text': $(this).text(),
        'confidence': settings.confidence,
        'support': settings.support,
        'spotter': settings.spotter,
        'disambiguator': settings.disambiguator,
        'policy': settings.policy
    };
    if ("types" in settings && settings["types"] != undefined)
        params["types"] = settings.types;
    if ("sparql" in settings && settings["sparql"] != undefined)
        params["sparql"] = settings.sparql;

    ajaxRequest = $.ajax({
        'url': settings.endpoint + "/candidates",
        'data': params,
        'context': this,
        'headers': {'Accept': 'application/json'},
        'success': update,
        'error': function (response) {
            if (settings.callback != undefined) {
                settings.callback(response);
            }
        }
    });
});
},
suggest: function (options) {
    //init(options);
    function update(response) {
        var keywords = Kolich.Selector.getSelected();
        var suggestion = Parser.getSuggestions(response, keywords);
        //console.log('keywords:'+keywords);
        //console.log('suggestion:'+suggestion);
        var content = "<div style='overflow: auto>" + suggestion + "</div>";
        if (settings.powered_by == 'yes') {
            $(content).append($(powered_by));
        }
        $(this).html(content);
    }
}

return this.each(function () {
    return;
    //console.log('suggest');
    var keywords = $.trim("") + Kolich.Selector.getSelected();
    var params = {
        'text': $(this).text(),
        'confidence': settings.confidence,
        'support': settings.support,
        'spotter': settings.spotter,
        'disambiguator': settings.disambiguator,
        'surfaceForm': keywords
    };
    if ("types" in settings && settings["types"] != undefined) {

```

```

        params['types'] = settings.types;
    }

    ajaxRequest = $.ajax({
        'url': settings.endpoint + "/candidates",
        'data': params,
        'context': this,
        'headers': {'Accept': 'application/json'},
        'success': update,
        'error': function (response) {
            if (settings.callback != undefined) {
                settings.callback(response);
            }
        }
    });
});
};
}
};

$.fn.annotate = function (method) {
    //console.log('method:',method);
    // Method calling logic
    if (methods[method]) {
        return methods[method].apply(this, Array.prototype.slice.call(arguments, 1));
    } else if (typeof method === 'object' || !method) {
        return methods.init.apply(this, arguments);
    } else {
        $.error("Method " + method + " does not exist on jQuery.annotate (DBpedia Spotlight)");
    }
};

$.fn.cancelAnnotation = function () {
    ajaxRequest.abort();
}
})(jQuery);

```

ANEXO 4. Código para la anotación con las entradas en la base de conocimiento local.

El siguiente es el código que se desarrolló para realizar el proceso de anotación con las entradas en la base de conocimiento local.

```

<?php
/**
 * Created by PhpStorm.
 * User: Jose Balseca
 * Date: 3/2/2017
 * Time: 8:17 AM
 * Function: reconocimiento.php
 */

//-----
// Php script for fetching data from mysql database
//-----

$myurlvariable = $_POST['subject'];
$ranking_dbpedia = $_POST['ranking'];
$host = "localhost";
$user = "clasificador";
$pass = "clasificador";

```

```

$databaseName = "clasificadorbdc";
$tableName1 = "rec_data_final_labels";
$tableName2 = "kim_area_dbpedia_labels_redirects";
$tableName3 = "topics";
$tableName4 = "unesco_area";
$tableName5 = "ranking_object_final";

//-----
// Connect to mysql database
//-----

$con = mysqli_connect($host, $user, $pass, $databaseName);

if (!$con)
{
    die('No se puede conectar a la base de datos');
}

//-----
// Query database for data
//-----

$result = mysqli_query($con, "SELECT * FROM $tableName1 where redirect = '$myurlvariable'");
if ($result->num_rows <= 0) {
    $result = mysqli_query($con, "SELECT * FROM $tableName2 where redirect = '$myurlvariable'");
}

if ($result->num_rows > 0) {

    //echo 'query';
    //echo "INSERT INTO $tableName3 values (select title from $tableName4 where unesco_uri IN (select subject from
    $tableName5 where object = '$myurlvariable'))";

    $stmt = mysqli_query($con, "INSERT INTO $tableName3 select title, subject, ('w-tf-idf' * $ranking_dbpedia) as ranking
    from ranking_object_final, unesco_area where unesco_uri = subject and object = '$myurlvariable'");
}

$array = mysqli_fetch_array($result);

//-----
// echo result as json
//-----
echo json_encode($array);

```

ANEXO 5. Código para realizar el proceso de borrado y preparación para una nueva clasificación.

El siguiente es el código que se desarrolló para realizar el proceso de borrada y preparación para una nueva clasificación:

```

<?php
/**
 * Created by PhpStorm.
 * User: Jose Balseca
 * Date: 3/2/2017
 * Time: 8:17 AM
 * Function: borraclasificador.php
 */

//-----
// Php script for fetching data from mysql database
//-----

```

```

$host = "localhost";
$user = "clasificador";
$pass = "clasificador";

$databaseName = "clasificadorbdc";
$tableName1 = "topics";

//-----
// Connect to mysql database
//-----

$con = mysqli_connect($host, $user, $pass, $databaseName);
if (!$con)
{
    die('No se puede conectar a la base de datos');
}

//-----
// Query database for data
//-----

$result = mysqli_query($con, "DELETE FROM $tableName1");

//-----
// echo result as json
//-----
echo json_encode($result);

```

ANEXO 6. Código para contrastar las entradas de la base de conocimiento local.

El siguiente es el código que se desarrolló para contrastar las entradas de la base de conocimiento local:

```

// Clear entries for a new UNESCO classification process
$.ajax({
    type: "POST",
    url: currurl+"borraclasificador.php",
    async: false,
    dataType: 'json',
    success: function(data)
    {
        resultado = data[0];
    },
    error: function(xhr, desc, err){
        alert("Hay un error borrado clasificador:" + xhr.responseText);
    }
});

.....

// Contrast with knowledge database local
$.ajax({
    type: "POST",
    url: currurl+"reconocimiento.php",
    data: "subject="+uri+"&ranking="+ranking,
    async: false,
    dataType: 'json',
    success: function(data)
    {
        label = data[1];
    },
    error: function(xhr, desc, err){
        alert("Hay un error en validación tripletas:" + xhr.responseText);
    }
});

```



```
    }  
  });  
  
.....
```

ANEXO 7. Código para realizar la clasificación temática.

El siguiente es el código que se desarrolló para realizar la clasificación temática en función de las entradas de la base de conocimiento local asociada al tesoro de la UNESCO:

```
<?php  
/**  
 * Created by PhpStorm.  
 * User: Jose Balseca  
 * Date: 3/2/2017  
 * Time: 8:17 AM  
 * Function: reconocimiento.php  
 */  
  
//-----  
// Php script for fetching data from mysql database  
//-----  
  
$myurlvariable = $_POST['subject'];  
$ranking_dbpedia = $_POST['ranking'];  
$host = "localhost";  
$user = "clasificador";  
$pass = "clasificador";  
  
$databaseName = "clasificadorbdc";  
$tableName1 = "rec_data_final_labels";  
$tableName2 = "kim_area_dbpedia_labels_redirects";  
$tableName3 = "topics";  
$tableName4 = "unesco_area";  
$tableName5 = "ranking_object_final";  
  
//-----  
// Connect to mysql database  
//-----  
  
$con = mysqli_connect($host, $user, $pass, $databaseName);  
if (!$con)  
{  
    die('No se puede conectar a la base de datos');
```

```

}

//-----
// Query database for data
//-----

$result = mysqli_query($con, "SELECT * FROM $tableName1 where redirect = '$myurlvariable'");
if ($result->num_rows <= 0) {
    $result = mysqli_query($con, "SELECT * FROM $tableName2 where redirect = '$myurlvariable'");
}
if ($result->num_rows > 0) {
    //echo 'query';
    //echo "INSERT INTO $tableName3 values (select title from $tableName4 where unesco_uri IN (select subject from
    $tableName5 where object = '$myurlvariable'))";
    $stmt = mysqli_query($con, "INSERT INTO $tableName3 select title, subject, ('w-tf-idf' * $ranking_dbpedia) as ranking
    from ranking_object_final, unesco_area where unesco_uri = subject and object = '$myurlvariable'");
}
$array = mysqli_fetch_array($result);

//-----
// echo result as json
//-----
echo json_encode($array);

```

ANEXO 8. Código para definir el detalle y versión del plugin desarrollado Moodle.

```

<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * Defines the version and other meta-info about the plugin
 *
 * Setting the $plugin->version to 0 prevents the plugin from being installed.
 * See https://docs.moodle.org/dev/version.php for more info.

```

```

*
* @package mod_thematicclasificador
* @copyright 2016 Your Name <your@email.address>
* @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
*/

defined('MOODLE_INTERNAL') || die();

$plugin->component = 'mod_thematicclasificador';
$plugin->version = 2017060101;
$plugin->release = 'v1.0';
$plugin->requires = 2014051200;
$plugin->maturity = MATURITY_STABLE;
$plugin->cron = 0;
$plugin->dependencies = array();

```

ANEXO 9. Código para definir la disponibilidad del plugin Moodle.

```

<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * Redirect the user to the appropriate submission related page
 */
* @package mod_thematicclasificador
* @category grade
* @copyright 2016 Your Name <your@email.address>
* @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
*/

require_once(__DIR__ . '/../../config.php');

$id = required_param('id', PARAM_INT); // Course module ID.
// Item number may be != 0 for activities that allow more than one grade per user.
$itemnumber = optional_param('itemnumber', 0, PARAM_INT);
$userid = optional_param('userid', 0, PARAM_INT); // Graded user ID (optional).

// In the simplest case just redirect to the view page.
redirect('view.php?id='.$id);

```

ANEXO 10. Código para definir la presentación del plugin dentro del entorno Moodle.

```

<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or

```

```

// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * The main thematicclasificator configuration form
 *
 * It uses the standard core Moodle formslib. For more info about them, please
 * visit: http://docs.moodle.org/en/Development:lib/formslib.php
 *
 * @package mod_thematicclasificator
 * @copyright 2016 Your Name <your@email.address>
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

defined('MOODLE_INTERNAL') || die();

require_once($CFG->dirroot.'/course/moodleform_mod.php');

/**
 * Module instance settings form
 *
 * @package mod_thematicclasificator
 * @copyright 2016 Your Name <your@email.address>
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */
class mod_thematicclasificator_mod_form extends moodleform_mod {

    /**
     * Defines forms elements
     */
    public function definition() {
        global $CFG;

        $mform = $this->_form;

        // Adding the "general" fieldset, where all the common settings are showed.
        $mform->addElement('header', 'general', get_string('general', 'form'));

        // Adding the standard "name" field.
        $mform->addElement('text', 'name', get_string('thematicclasificatorname', 'thematicclasificator'), array('size' => '64'));
        if (!empty($CFG->formatstringstriptags)) {
            $mform->setType('name', PARAM_TEXT);
        } else {
            $mform->setType('name', PARAM_CLEANHTML);
        }
        $mform->addRule('name', null, 'required', null, 'client');
        $mform->addRule('name', get_string('maximumchars', '', 255), 'maxlength', 255, 'client');
        $mform->addHelpButton('name', 'thematicclasificatorname', 'thematicclasificator');

        // Adding the standard "intro" and "introformat" fields.
        if ($CFG->branch >= 29) {
            $this->standard_intro_elements();
        } else {
            $this->add_intro_editor();
        }

        // Adding the rest of thematicclasificator settings, spreading all them into this fieldset
        // ... or adding more fieldsets ('header' elements) if needed for better logic.
        // $mform->addElement('static', 'label1', 'thematicclasificatorsetting1', 'Your thematicclasificator fields go here. Replace
        // me!');
    }
}

```

```

        //mform->addElement('header', 'thematicclasificatorfieldset', get_string('thematicclasificatorfieldset',
'thematicclasificator'));
        //mform->addElement('static', 'label2', 'thematicclasificatorsetting2', 'Your thematicclasificator fields go here. Replace
me!');

        // Add standard grading elements.
        //$this->standard_grading_coursemodule_elements();

        // Add standard elements, common to all modules.
        $this->standard_coursemodule_elements();

        // Add standard buttons, common to all modules.
        $this->add_action_buttons();
    }
}

```

ANEXO 11. Código para identificar las librerías utilizadas en la creación del plugin Moodle.

```

<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * Internal library of functions for module thematicclasificator
 *
 * All the thematicclasificator specific functions, needed to implement the module
 * logic, should go here. Never include this file from your lib.php!
 *
 * @package mod_thematicclasificator
 * @copyright 2016 Your Name <your@email.address>
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

defined('MOODLE_INTERNAL') || die();

/**
 * Does something really useful with the passed things
 *
 * @param array $things
 * @return object
 */
function thematicclasificator_do_something_useful(array $things) {
    return new stdClass();
}

```

ANEXO 12. Código para especificar las entradas del idioma para el plugin Moodle.

```

<?php
// This file is part of Moodle - http://moodle.org/

```

```

//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * English strings for thematicclasificator
 *
 * You can have a rather longer description of the file as well,
 * if you like, and it can span multiple lines.
 *
 * @package mod_thematicclasificator
 * @copyright 2016 Your Name <your@email.address>
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

defined('MOODLE_INTERNAL') || die();

$string['modulename'] = 'Thematic Clasificator UTPL';
$string['modulenameplural'] = 'Thematic Clasificator UTPL';
$string['modulename_help'] = 'Use the thematicclasificator module for clasify the text or peragraph to find the topics associate and clasify by the UNESCO Thesaurus | The thematicclasificator module allows find the related topics for the text or paragraph';
$string['thematicclasificator:addinstance'] = 'Add a new thematicclasificator';
$string['thematicclasificator:submit'] = 'Submit thematic clasificator';
$string['thematicclasificator:view'] = 'View thematic clasificator';
$string['thematicclasificatorfieldset'] = 'Custom example fieldset';
$string['thematicclasificatorname'] = 'Name';
$string['thematicclasificatorname_help'] = 'This is the content of the help tooltip associated with the thematicclasificatorname field. Markdown syntax is supported.';
$string['thematicclasificator'] = 'thematicclasificator';
$string['pluginadministration'] = 'thematicclasificator administration';
$string['pluginname'] = 'thematicclasificator';

```

ANEXO 13. Código para definir las clases utilizadas en el plugin Moodle.

```

<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * The mod_thematicclasificator instance list viewed event.
 *
 * @package mod_thematicclasificator

```

```

* @copyright 2016 Your Name <your@email.address>
* @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
*/

namespace mod_thematicclasificator\event;

defined('MOODLE_INTERNAL') || die();

/**
 * The mod_thematicclasificator instance list viewed event class.
 *
 * @package mod_thematicclasificator
 * @copyright 2016 Your Name <your@email.address>
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */
class course_module_instance_list_viewed extends \core\event\course_module_instance_list_viewed {
}

```

```

<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * Defines the view event.
 *
 * @package mod_thematicclasificator
 * @copyright 2016 Your Name <your@email.address>
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

namespace mod_thematicclasificator\event;

defined('MOODLE_INTERNAL') || die();

/**
 * The mod_thematicclasificator instance viewed event class
 *
 * If the view mode needs to be stored as well, you may need to
 * override methods get_url() and get_legacy_log_data(), too.
 *
 * @package mod_thematicclasificator
 * @copyright 2016 Your Name <your@email.address>
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */
class course_module_viewed extends \core\event\course_module_viewed {

    /**
     * Initialize the event
     */
    protected function init() {
        $this->data['objecttable'] = 'thematicclasificator';
        parent::init();
    }
}

```

ANEXO 14. Código para definir las entradas a la base de datos de Moodle asociadas al plugin creado.

```
<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * Capability definitions for the thematicclasificator module
 *
 * The capabilities are loaded into the database table when the module is
 * installed or updated. Whenever the capability definitions are updated,
 * the module version number should be bumped up.
 *
 * The system has four possible values for a capability:
 * CAP_ALLOW, CAP_PREVENT, CAP_PROHIBIT, and inherit (not set).
 *
 * It is important that capability names are unique. The naming convention
 * for capabilities that are specific to modules and blocks is as follows:
 * [mod/block]/<plugin_name>:<capabilityname>
 *
 * component_name should be the same as the directory name of the mod or block.
 *
 * Core moodle capabilities are defined thus:
 * moodle/<capabilityclass>:<capabilityname>
 *
 * Examples: mod/forum:viewpost
 *           block/recent_activity:view
 *           moodle/site:deleteuser
 *
 * The variable name for the capability definitions array is $capabilities
 *
 * @package   mod_thematicclasificator
 * @copyright 2016 Your Name <your@email.address>
 * @license   http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

defined('MOODLE_INTERNAL') || die();

// Modify capabilities as needed and remove this comment.
$capabilities = array(
    'mod/thematicclasificator:addinstance' => array(
        'riskbitmask' => RISK_XSS,
        'captype' => 'write',
        'contextlevel' => CONTEXT_COURSE,
        'archetypes' => array(
            'editingteacher' => CAP_ALLOW,
            'manager' => CAP_ALLOW
        ),
        'clonepermissionsfrom' => 'moodle/course:manageactivities'
    ),

    'mod/thematicclasificator:view' => array(
        'captype' => 'read',
        'contextlevel' => CONTEXT_MODULE,
        'archetypes' => array(
```



```

        'guest' => CAP_ALLOW,
        'user' => CAP_ALLOW,
    )
),

'mod/thematicclasificator:submit' => array(
    'riskbitmask' => RISK_SPAM,
    'captype' => 'write',
    'contextlevel' => CONTEXT_MODULE,
    'archetypes' => array(
        'student' => CAP_ALLOW
    )
),
);

```

ANEXO 15. Código para definir las entradas de la instalación del plugin Moodle.

```

<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * Provides code to be executed during the module installation
 *
 * This file replaces the legacy STATEMENTS section in db/install.xml,
 * lib.php/modulename_install() post installation hook and partially defaults.php.
 *
 * @package   mod_thematicclasificator
 * @copyright 2016 Your Name <your@email.address>
 * @license   http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

/**
 * Post installation procedure
 *
 * @see upgrade_plugins_modules()
 */
function xmldb_thematicclasificator_install() {
}

/**
 * Post installation recovery procedure
 *
 * @see upgrade_plugins_modules()
 */
function xmldb_thematicclasificator_install_recovery() {
}

```

```
<?xml version="1.0" encoding="UTF-8" ?>
```

```

<XMLDB PATH="mod/thematicclasificator/db" VERSION="20101203" COMMENT="XMLDB file for Moodle
mod/thematicclasificator"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="../../lib/xmldb/xmldb.xsd"
>
<TABLES>
<TABLE NAME="thematicclasificator" COMMENT="Default comment for thematicclasificator, please edit me">
  <FIELDS>
    <FIELD NAME="id" TYPE="int" LENGTH="10" NOTNULL="true" UNSIGNED="true" SEQUENCE="true"/>
    <FIELD NAME="course" TYPE="int" LENGTH="10" NOTNULL="true" UNSIGNED="true" SEQUENCE="false"
COMMENT="Course thematicclasificator activity belongs to"/>
    <FIELD NAME="name" TYPE="char" LENGTH="255" NOTNULL="true" SEQUENCE="false" COMMENT="name field
for moodle instances"/>
    <FIELD NAME="intro" TYPE="text" NOTNULL="true" SEQUENCE="false" COMMENT="General introduction of the
thematicclasificator activity"/>
    <FIELD NAME="introformat" TYPE="int" LENGTH="4" NOTNULL="true" UNSIGNED="true" DEFAULT="0"
SEQUENCE="false" COMMENT="Format of the intro field (MOODLE, HTML, MARKDOWN...)/>
    <FIELD NAME="timecreated" TYPE="int" LENGTH="10" NOTNULL="true" UNSIGNED="true" SEQUENCE="false"/>
    <FIELD NAME="timemodified" TYPE="int" LENGTH="10" NOTNULL="true" UNSIGNED="true" DEFAULT="0"
SEQUENCE="false"/>
    <FIELD NAME="grade" TYPE="int" LENGTH="10" NOTNULL="true" DEFAULT="100" SEQUENCE="false"
COMMENT="The maximum grade. Can be negative to indicate the use of a scale."/>
  </FIELDS>
  <KEYS>
    <KEY NAME="primary" TYPE="primary" FIELDS="id"/>
  </KEYS>
  <INDEXES>
    <INDEX NAME="course" UNIQUE="false" FIELDS="course"/>
  </INDEXES>
</TABLE>
</TABLES>
</XMLDB>

```

ANEXO 16. Código para definir las entradas para el proceso de desinstalación del plugin Moodle.

```

<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * Provides code to be executed during the module uninstallation
 *
 * @see uninstall_plugin()
 *
 * @package mod_thematicclasificator
 * @copyright 2016 Your Name <your@email.address>
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

/**
 * Custom uninstallation procedure
 */
function xmldb_thematicclasificator_uninstall() {
    return true;
}

```

```
}
```

ANEXO 17. Código para definir las entradas para el proceso de actualización del plugin Moodle.

```
<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * This file keeps track of upgrades to the thematicclasificator module
 *
 * Sometimes, changes between versions involve alterations to database
 * structures and other major things that may break installations. The upgrade
 * function in this file will attempt to perform all the necessary actions to
 * upgrade your older installation to the current version. If there's something
 * it cannot do itself, it will tell you what you need to do. The commands in
 * here will all be database-neutral, using the functions defined in DLL libraries.
 *
 * @package mod_thematicclasificator
 * @copyright 2016 Your Name <your@email.address>
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

defined('MOODLE_INTERNAL') || die();

/**
 * Execute thematicclasificator upgrade from the given old version
 *
 * @param int $oldversion
 * @return bool
 */
function xmldb_thematicclasificator_upgrade($oldversion) {
    global $DB;

    $dbman = $DB->get_manager(); // Loads ddl manager and xmldb classes.

    /*
     * And upgrade begins here. For each one, you'll need one
     * block of code similar to the next one. Please, delete
     * this comment lines once this file start handling proper
     * upgrade code.
     *
     * if ($oldversion < YYYYMMDD00) { //New version in version.php
     * }
     *
     * Lines below (this included) MUST BE DELETED once you get the first version
     * of your module ready to be installed. They are here only
     * for demonstrative purposes and to show how the thematicclasificator
     * itself has been upgraded.
     *
     * For each upgrade block, the file thematicclasificator/version.php
     * needs to be updated . Such change allows Moodle to know
     * that this file has to be processed.
     */
}
```

```

* To know more about how to write correct DB upgrade scripts it's
* highly recommended to read information available at:
* http://docs.moodle.org/en/Development:XMLDB\_Documentation
* and to play with the XMLDB Editor (in the admin menu) and its
* PHP generation possibilities.
*
* First example, some fields were added to install.xml on 2007/04/01
*/
if ($oldversion < 2007040100) {

    // Define field course to be added to thematicclasificator.
    $table = new xmldb_table('thematicclasificator');
    $field = new xmldb_field('course', XMLDB_TYPE_INTEGER, '10', XMLDB_UNSIGNED, XMLDB_NOTNULL, null, '0',
'id');

    // Add field course.
    if (!$dbman->field_exists($table, $field)) {
        $dbman->add_field($table, $field);
    }

    // Define field intro to be added to thematicclasificator.
    $table = new xmldb_table('thematicclasificator');
    $field = new xmldb_field('intro', XMLDB_TYPE_TEXT, 'medium', null, null, null, null, 'name');

    // Add field intro.
    if (!$dbman->field_exists($table, $field)) {
        $dbman->add_field($table, $field);
    }

    // Define field introformat to be added to thematicclasificator.
    $table = new xmldb_table('thematicclasificator');
    $field = new xmldb_field('introformat', XMLDB_TYPE_INTEGER, '4', XMLDB_UNSIGNED, XMLDB_NOTNULL, null,
'0',
    'intro');

    // Add field introformat.
    if (!$dbman->field_exists($table, $field)) {
        $dbman->add_field($table, $field);
    }

    // Once we reach this point, we can store the new version and consider the module
    // ... upgraded to the version 2007040100 so the next time this block is skipped.
    upgrade_mod_savepoint(true, 2007040100, 'thematicclasificator');
}

// Second example, some hours later, the same day 2007/04/01
// ... two more fields and one index were added to install.xml (note the micro increment
// ... "01" in the last two digits of the version).
if ($oldversion < 2007040101) {

    // Define field timecreated to be added to thematicclasificator.
    $table = new xmldb_table('thematicclasificator');
    $field = new xmldb_field('timecreated', XMLDB_TYPE_INTEGER, '10', XMLDB_UNSIGNED, XMLDB_NOTNULL,
null, '0',
    'introformat');

    // Add field timecreated.
    if (!$dbman->field_exists($table, $field)) {
        $dbman->add_field($table, $field);
    }

    // Define field timemodified to be added to thematicclasificator.
    $table = new xmldb_table('thematicclasificator');
    $field = new xmldb_field('timemodified', XMLDB_TYPE_INTEGER, '10', XMLDB_UNSIGNED, XMLDB_NOTNULL,
null, '0',
    'timecreated');

    // Add field timemodified.
    if (!$dbman->field_exists($table, $field)) {

```

```

    $dbman->add_field($table, $field);
}

// Define index course (not unique) to be added to thematicclasificator.
$table = new xmldb_table('thematicclasificator');
$index = new xmldb_index('courseindex', XMLDB_INDEX_NOTUNIQUE, array('course'));

// Add index to course field.
if (!$dbman->index_exists($table, $index)) {
    $dbman->add_index($table, $index);
}

// Another save point reached.
upgrade_mod_savepoint(true, 2007040101, 'thematicclasificator');
}

// Third example, the next day, 2007/04/02 (with the trailing 00),
// some actions were performed to install.php related with the module.
if ($oldversion < 2007040200) {

    // Insert code here to perform some actions (same as in install.php).

    upgrade_mod_savepoint(true, 2007040200, 'thematicclasificator');
}

/*
 * And that's all. Please, examine and understand the 3 example blocks above. Also
 * it's interesting to look how other modules are using this script. Remember that
 * the basic idea is to have "blocks" of code (each one being executed only once,
 * when the module version (version.php) is updated.
 *
 * Lines above (this included) MUST BE DELETED once you get the first version of
 * your module working. Each time you need to modify something in the module (DB
 * related, you'll raise the version and add one upgrade block here.
 *
 * Finally, return of upgrade result (true, all went good) to Moodle.
 */
return true;
}

```

ANEXO 18. Código para definir el proceso de respaldo y restauración del plugin Moodle.

```

<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * Defines backup_thematicclasificator_activity_task class
 *
 * @package mod_thematicclasificator
 * @category backup
 * @copyright 2016 Your Name <your@email.address>
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

```

```

defined('MOODLE_INTERNAL') || die;

require_once($CFG->dirroot . '/mod/thematicclasificator/backup/moodle2/backup_thematicclasificator_stepslib.php');

/**
 * Provides the steps to perform one complete backup of the thematicclasificator instance
 *
 * @package mod_thematicclasificator
 * @category backup
 * @copyright 2016 Your Name <your@email.address>
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */
class backup_thematicclasificator_activity_task extends backup_activity_task {

    /**
     * No specific settings for this activity
     */
    protected function define_my_settings() {
    }

    /**
     * Defines a backup step to store the instance data in the thematicclasificator.xml file
     */
    protected function define_my_steps() {
        $this->add_step(new backup_thematicclasificator_activity_structure_step('thematicclasificator_structure',
'thematicclasificator.xml'));
    }

    /**
     * Encodes URLs to the index.php and view.php scripts
     *
     * @param string $content some HTML text that eventually contains URLs to the activity instance scripts
     * @return string the content with the URLs encoded
     */
    static public function encode_content_links($content) {
        global $CFG;

        $base = preg_quote($CFG->wwwroot, '/');

        // Link to the list of thematicclasificators.
        $search = '/('.$base.'/mod/thematicclasificator/index.php?id=)[0-9]+';
        $content = preg_replace($search, '@THEMATICCLASIFICATORINDEX*$2@$', $content);

        // Link to thematicclasificator view by moduleid.
        $search = '/('.$base.'/mod/thematicclasificator/view.php?id=)[0-9]+';
        $content = preg_replace($search, '@THEMATICCLASIFICATORVIEWBYID*$2@$', $content);

        return $content;
    }
}

```

```

<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

```

```

/**
 * Define all the backup steps that will be used by the backup_thematicclasificator_activity_task
 *
 * @package mod_thematicclasificator
 * @category backup
 * @copyright 2016 Your Name <your@email.address>
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

defined('MOODLE_INTERNAL') || die;

/**
 * Define the complete thematicclasificator structure for backup, with file and id annotations
 *
 * @package mod_thematicclasificator
 * @category backup
 * @copyright 2016 Your Name <your@email.address>
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */
class backup_thematicclasificator_activity_structure_step extends backup_activity_structure_step {

    /**
     * Defines the backup structure of the module
     *
     * @return backup_nested_element
     */
    protected function define_structure() {

        // Get know if we are including userinfo.
        $userinfo = $this->get_setting_value('userinfo');

        // Define the root element describing the thematicclasificator instance.
        $thematicclasificator = new backup_nested_element('thematicclasificator', array('id'), array(
            'name', 'intro', 'introformat', 'grade'));

        // If we had more elements, we would build the tree here.

        // Define data sources.
        $thematicclasificator->set_source_table('thematicclasificator', array('id' => backup::VAR_ACTIVITYID));

        // If we were referring to other tables, we would annotate the relation
        // with the element's annotate_ids() method.

        // Define file annotations (we do not use itemid in this example).
        $thematicclasificator->annotate_files('mod_thematicclasificator', 'intro', null);

        // Return the root element (thematicclasificator), wrapped into standard activity structure.
        return $this->prepare_activity_structure($thematicclasificator);
    }
}

```

```

<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

```

```

/**
 * Provides the restore activity task class
 *
 * @package mod_thematicclasificator
 * @category backup
 * @copyright 2016 Your Name <your@email.address>
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

defined('MOODLE_INTERNAL') || die();

require_once($CFG->dirroot . '/mod/thematicclasificator/backup/moodle2/restore_thematicclasificator_stepslib.php');

/**
 * Restore task for the thematicclasificator activity module
 *
 * Provides all the settings and steps to perform complete restore of the activity.
 *
 * @package mod_thematicclasificator
 * @category backup
 * @copyright 2016 Your Name <your@email.address>
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */
class restore_thematicclasificator_activity_task extends restore_activity_task {

    /**
     * Define (add) particular settings this activity can have
     */
    protected function define_my_settings() {
        // No particular settings for this activity.
    }

    /**
     * Define (add) particular steps this activity can have
     */
    protected function define_my_steps() {
        // We have just one structure step here.
        $this->add_step(new restore_thematicclasificator_activity_structure_step('thematicclasificator_structure',
'thematicclasificator.xml'));
    }

    /**
     * Define the contents in the activity that must be
     * processed by the link decoder
     */
    static public function define_decode_contents() {
        $contents = array();

        $contents[] = new restore_decode_content('thematicclasificator', array('intro'), 'thematicclasificator');

        return $contents;
    }

    /**
     * Define the decoding rules for links belonging
     * to the activity to be executed by the link decoder
     */
    static public function define_decode_rules() {
        $rules = array();

        $rules[] = new restore_decode_rule('THEMATICCLASIFICATORVIEWBYID',
'/mod/thematicclasificator/view.php?id=$1', 'course_module');
        $rules[] = new restore_decode_rule('THEMATICCLASIFICATORINDEX', '/mod/thematicclasificator/index.php?id=$1',
'course');

        return $rules;
    }
}

```



```

/**
 * Define the restore log rules that will be applied
 * by the {@link restore_logs_processor} when restoring
 * thematicclasificator logs. It must return one array
 * of {@link restore_log_rule} objects
 */
static public function define_restore_log_rules() {
    $rules = array();

    $rules[] = new restore_log_rule('thematicclasificator', 'add', 'view.php?id={course_module}', '{thematicclasificator}');
    $rules[] = new restore_log_rule('thematicclasificator', 'update', 'view.php?id={course_module}', '{thematicclasificator}');
    $rules[] = new restore_log_rule('thematicclasificator', 'view', 'view.php?id={course_module}', '{thematicclasificator}');

    return $rules;
}

/**
 * Define the restore log rules that will be applied
 * by the {@link restore_logs_processor} when restoring
 * course logs. It must return one array
 * of {@link restore_log_rule} objects
 *
 * Note this rules are applied when restoring course logs
 * by the restore final task, but are defined here at
 * activity level. All them are rules not linked to any module instance (cmid = 0)
 */
static public function define_restore_log_rules_for_course() {
    $rules = array();

    $rules[] = new restore_log_rule('thematicclasificator', 'view all', 'index.php?id={course}', null);

    return $rules;
}
}

```

```

<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * Define all the restore steps that will be used by the restore_thematicclasificator_activity_task
 *
 * @package mod_thematicclasificator
 * @category backup
 * @copyright 2016 Your Name <your@email.address>
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

/**
 * Structure step to restore one thematicclasificator activity
 *
 * @package mod_thematicclasificator
 * @category backup
 * @copyright 2016 Your Name <your@email.address>
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

```

```

*/
class restore_thematicclasificator_activity_structure_step extends restore_activity_structure_step {

    /**
     * Defines structure of path elements to be processed during the restore
     *
     * @return array of {@link restore_path_element}
     */
    protected function define_structure() {

        $paths = array();
        $paths[] = new restore_path_element('thematicclasificator', '/activity/thematicclasificator');

        // Return the paths wrapped into standard activity structure.
        return $this->prepare_activity_structure($paths);
    }

    /**
     * Process the given restore path element data
     *
     * @param array $data parsed element data
     */
    protected function process_thematicclasificator($data) {
        global $DB;

        $data = (object)$data;
        $oldid = $data->id;
        $data->course = $this->get_courseid();

        if (empty($data->timecreated)) {
            $data->timecreated = time();
        }

        if (empty($data->timemodified)) {
            $data->timemodified = time();
        }

        if ($data->grade < 0) {
            // Scale found, get mapping.
            $data->grade = -($this->get_mappingid('scale', abs($data->grade)));
        }

        // Create the thematicclasificator instance.
        $newitemid = $DB->insert_record('thematicclasificator', $data);
        $this->apply_activity_instance($newitemid);
    }

    /**
     * Post-execution actions
     */
    protected function after_execute() {
        // Add thematicclasificator related files, no need to match by itemname (just internally handled context).
        $this->add_related_files('mod_thematicclasificator', 'intro', null);
    }
}

```