



UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA

La Universidad Católica de Loja

ESCUELA DE CIENCIAS DE LA COMPUTACIÓN

“DESARROLLO E IMPLEMENTACIÓN DE UNA BASE DE DATOS DE LOS RECURSOS LICENCIADOS CON CC EN ECUADOR”

*Tesis de Grado previa a la obtención
del título de Ingeniero en
Sistemas Informáticos y Computación*

Autora:

Susana Aguilar Quezada.

Directora:

Ing. Irma Cadme Samaniego.

Loja - Ecuador

2011

CERTIFICACIÓN

Ing.

Irma Cadme Samaniego.

DIRECTORA DE TESIS

CERTIFICA:

Que el presente trabajo de tesis realizado por la señora Susana Aguilar Quezada, cuyo tema es **“DESARROLLO E IMPLEMENTACIÓN DE UNA BASE DE DATOS DE LOS RECURSOS LICENCIADOS CON CC EN ECUADOR”**, ha sido dirigido, orientado y evaluado en todas sus fases, habiendo constatado que cumple con los requisitos de fondo y forma exigidos por la Escuela de Ciencias de la Computación, en consecuencia autorizo su presentación, sustentación y defensa.

Loja, abril del 2010

.....
Ing. Irma Cadme Samaniego
Directora de Tesis

CESIÓN DE DERECHOS

Yo, Susana Aguilar Quezada, declaro ser autora del presente trabajo y eximo expresamente a la Universidad Técnica Particular de Loja y a sus representantes legales de posibles reclamos o acciones legales.

Adicionalmente, declaro conocer y aceptar la disposición del Art. 67 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 que se realicen a través, o con el apoyo financiero, académico o institucional (operativo) de la universidad".

Loja, abril del 2011

.....

Susana Aguilar Quezada

AUTORÍA

Los conceptos, ideas y opiniones desarrolladas en el presente trabajo son de exclusiva responsabilidad del autor.

Loja, abril del 2011

.....

Susana Aguilar Quezada

AGRADECIMIENTO

Agradezco a Dios por haberme dado la vida y permitirme a través de mis padres, familiares, profesores y amigos adquirir valores como la honestidad, honradez, justicia, humildad, responsabilidad, sacrificio y respeto para conmigo mismo y quienes me rodean.

Agradezco de manera especial a mi directora y codirectora de tesis, Ing. Irma Cadme Samaniego e Ing. Samanta Cueva, respectivamente, por su colaboración durante el tiempo de realización de la presente tesis para guiar y conducir pensamientos y técnicas que permitieron desarrollar adecuadamente el presente trabajo.

Susana

DEDICATORIA

A Dios, a mi madre, a mi esposo y a mi tierna hija Anghelet.

Dedicado también a todas las personas que luchan día a día contra cualquier tipo de adversidad, siempre convencidos y llenos de esperanza de que sus sueños no morirán jamás mientras se tenga coraje y valentía para seguir adelante y vencer cualquier obstáculo que se presente al caminar.

Susana

ÍNDICE GENERAL

Índice de figuras	VIII
Introducción	1
Objetivos	2
Organización del informe	3
1. Situación actual de las licencias Creative Commons	5
1.1. Fuentes de información	5
1.1.1. Clasificación de las Fuentes	6
1.1.2. Tratamiento de las fuentes	6
1.2. Tipos de licencias	8
1.2.1. Creative Commons	8
1.2.2. Licencias — Copyleft	9
1.2.3. Clasificación	10
1.2.4. Licencia Arte Libre	11
1.2.5. Licencia Aire Incondicional	11
1.3. Análisis de la situación actual	11
1.4. Indicadores	12
1.5. Definición de los requerimientos	13
1.5.1. Elementos arquitectónicos y de diseño	13
1.5.2. Dominio	14
1.5.3. Entorno de ejecución	14

1.5.4. Rendimiento	15
1.5.5. Seguridad	15
1.6. Definición de las estadísticas a obtener	15
1.7. Definición de los indicadores a obtener	16
2. Web Crawling: Situación actual	17
2.1. Web Crawler	18
2.2. Características del Web Crawler	19
2.3. Técnicas de búsqueda	20
2.3.1. Búsqueda primero en anchura	20
2.3.2. Búsqueda primero el mejor	21
2.4. Arquitectura del Web Crawler	21
2.4.1. Ejemplos de Crawlers	22
3. Análisis y Diseño de la aplicación	27
3.1. Arquitectura de la aplicación	27
3.2. Definición de las entradas	30
3.3. Definición de las salidas	30
3.4. Metodología y herramientas utilizadas	31
3.4.1. Análisis del sistema	31
3.4.2. Diseño del sistema	32
3.4.3. Desarrollo del software	32
3.4.4. Prueba del software	32
3.4.5. Implantación y evaluación	33
3.5. Especificación de los casos de uso	33
3.6. Diagrama de clases de la solución	34
3.7. Diseño de la base de datos	36
3.7.1. Modelo entidad-relación de la base de datos	36
3.7.2. Modelo relacional de la base de datos	37
3.8. RDF (Resource Description Framework)	38
3.8.1. Definición	38
3.8.2. Descripción de RDF utilizado en la aplicación	40
4. Desarrollo e Implementación de la aplicación	42
4.1. Construcción de la aplicación	42
4.2. Codificación del módulo de actualización de datos	44

Índice general

4.3. Codificación del módulo de estadística	45
4.4. Implementación de la aplicación	47
4.4.1. Configuración de la base de datos	48
4.4.2. Instalación de la aplicación	49
4.4.3. Implementación de la interfaz de usuario	49
5. Pruebas y resultados	51
5.1. Objetivo	51
5.2. Audiencia	51
5.3. Plan de Pruebas	52
5.4. Pruebas de funcionalidad y usabilidad	52
5.5. Pruebas de compatibilidad	62
5.6. Pruebas de robustez	64
5.7. Validación del portal	66
Discusión	69
Conclusiones y recomendaciones	73
Conclusiones	73
Recomendaciones	74
Trabajos futuros	76
Apéndices	77
A. Casos de uso	78
A.1. Diagramas de los casos de uso	78
A.2. Casos de uso de alto nivel	79
A.2.1. Caso de Uso: Lanzar el crawler	79
A.2.2. Caso de Uso: Estadísticas de la UTPL	80
A.2.3. Caso de Uso: Estadísticas de otras instituciones	81
A.2.4. Caso de Uso: Estadística consolidada (Ecuador)	82
A.2.5. Caso de Uso: Creación del repositorio local	83
A.2.6. Caso de Uso: Clasificación de documentos	84
A.2.7. Caso de Uso: Exploración Web personalizada	84

B. Evaluación de Estadística	86
C. Instalación y Configuración de Estadística	89
C.1. Requerimientos básicos	89
C.2. Instalación de Apache	90
C.3. Instalación de MySQL	91
D. Manual del usuario	94
D.1. Uso de Estadística	95
D.2. Uso de Licenciamiento	96
E. Manual del programador	98
E.1. Requerimientos básicos	98
E.2. Configuración e inicialización de la BDD	99
E.2.1. Script de instalación de la base de datos	99
E.2.2. Script de configuración de la base de datos	104
E.3. Código de la aplicación	106
E.3.1. Codificación del Web Crawler	106
E.3.2. Codificación del módulo de actualización de datos	109
E.3.3. Codificación del módulo de estadística	110
Bibliografía	118

Título de la Tesis:

Desarrollo e implementación de una base de datos de los recursos licenciados con CC en Ecuador.

Autora:

Susana Aguilar Quezada

Resumen:

El objetivo de este trabajo de investigación es desarrollar e implementar una base de datos de los recursos licenciados con CC en Ecuador y proveer estadísticas sobre la utilización de las mismas; esto con el fin de medir el impacto de la utilización de las licencias CC en el país y conocer la producción intelectual. En la implementación de la aplicación se construyó un crawler, utilizado para recolectar los recursos licenciados con CC Ecuador.

ÍNDICE DE FIGURAS

1.1. Clasificación de las fuentes.	6
1.2. Ciclo de tratamiento de las fuentes.	7
1.3. Licencias Copyleft.	10
2.1. Diagrama de actividad del Crawler.	18
2.2. Comportamiento del Web Crawler.	20
2.3. Exploración primero en anchura.	20
2.4. Exploración primero el mejor.	21
2.5. Arquitectura de un Web Crawler.	22
3.1. Modelo lógico de la aplicación.	28
3.2. Modelo físico de la aplicación.	29
3.3. Diagrama de secuencia del sistema.	29
3.4. Ciclo de vida clásico.	31
3.5. Diagrama de casos de uso de la aplicación.	33
3.6. Diagrama de casos de uso del Crawler.	34
3.7. Modelo de diseño de alto nivel de paquetes.	35
3.8. Modelo de clases de la capa de entidad.	35
3.9. Modelo E-R, BDD de la aplicación.	37
3.10. Modelo relacional, BDD Estadísticas CC.	37
3.11. Grafo dirigido que ilustra la estructura RDF.	39
3.12. Grafo dirigido (con URI's) que ilustra la estructura RDF.	39
3.13. Grafo dirigido (con URI's) del RDF de la aplicación.	41

Índice de figuras

4.1. Modelo de implementación de la aplicación.	43
4.2. Interfaz gráfica de usuario (GUI) de la aplicación.	44
4.3. Presentación de los datos.	45
4.4. Presentación de la gráfica.	46
4.5. Integración de los datos con la gráfica correspondiente. . . .	47
4.6. phpMyAdmin pantalla principal.	48
4.7. Interfaz gráfica de usuario (GUI) de la aplicación.	49
4.8. Interfaz gráfica de usuario (GUI) de Licenciamiento.	50
5.1. ¿El acceso al portal Estadísticas Creative Commons UTPL se realizó en un tiempo razonable?.	56
5.2. ¿Es aceptable el tiempo de respuesta al realizar la petición de un gráfico al portal?.	56
5.3. ¿Es aceptable el tiempo de respuesta al realizar la petición de un reporte al portal?.	57
5.4. ¿Es aceptable el tiempo de respuesta al realizar la petición de una licencia al portal?.	57
5.5. ¿Es fácil la navegación en el portal?	58
5.6. ¿En todo momento se sabe en qué parte de la aplicación nos encontramos?	59
5.7. ¿Es amigable la interfaz del portal?	59
5.8. ¿Se puede obtener la información fácilmente?	60
5.9. ¿Permite obtener reportes fácilmente?	60
5.10. La estructura de la información es	61
5.11. La estructura del contenido del portal es	61
5.12. Vista de la aplicación en el navegador Mozilla Firefox	63
5.13. Vista de la aplicación en el navegador Google Chrome	63
5.14. Vista de la aplicación en el navegador Internet Explorer	64
5.15. Gráfica del tiempo promedio del crawler, pruebas exhaustivas. .	65
5.16. Gráfica del tiempo promedio del crawler, pruebas leves. . . .	66
5.17. Validador W3C de documentos web.	66
5.18. Validación del portal.	67
5.19. Validación del portal.	68
5.20. Validación del portal.	68
A.1. Diagrama de casos de uso de la aplicación.	78

Índice de figuras

A.2. Diagrama de casos de uso del Crawler.	79
B.1. Encuesta de evaluación de la funcionalidad y usabilidad de Estadística (parte 1).	87
B.2. Encuesta de evaluación de la funcionalidad y usabilidad de Estadística (parte 2).	88
D.1. Pantalla de ingreso al portal.	94
D.2. Pantalla de Estadísticas.	95
D.3. Gráfica trimestral de recursos CC.	96
D.4. Reporte generado por la aplicación.	96
D.5. Pantalla de licenciamiento.	97
E.1. Presentación de los datos.	111
E.2. Presentación de la gráfica.	113
E.3. Integración de los datos con la gráfica correspondiente. . . .	114

INTRODUCCIÓN

Hoy en día, el crecimiento exponencial de la Web tiene, entre otras, dos implicaciones: existe más información disponible, pero el obtener información, de acuerdo a las necesidades del usuario, se vuelve una tarea compleja. Los motores de búsqueda existentes son, de forma general, buenas soluciones pero, existen enfoques alternativos a éstas soluciones que pueden mejorar los resultados de las búsquedas en la Web.

Esta tesis desarrolla e implementa una base de datos de los recursos licenciados con CC en Ecuador y provee estadísticas sobre la utilización de las mismas; esto con el fin de medir el impacto de la utilización de las licencias en el país y conocer la producción intelectual. Además, se aborda el tema del web crawling, proceso utilizado por los motores de búsqueda para obtener las páginas de la red.

Este esfuerzo se enmarca entre los diversos proyectos que buscan promover el uso de las licencias Creative Commons en Latinoamérica y especialmente en el Ecuador. La difusión de este tipo de licencias permite compartir información bajo la filosofía de “algunos derechos reservados” escogidos éstos de acuerdo a los intereses del autor.

Finalmente, se detalla el proceso de construcción de una aplicación web que permite:

- Obtener la licencia Creative Commons Ecuador y almacenar la información en una base de datos local.

- Determinar las estadísticas de las obras licenciadas con Creative Commons en Ecuador provenientes de la UTPL como de otras fuentes del país.
- Proveer una interfaz para que los usuarios de CC Ecuador licencien sus obras y almacenen estos datos.
- Implementar un web crawler que busque recursos Creative Commons cuya licencia tenga la jurisdicción del Ecuador en la internet.

Objetivos

El objetivo general del presente proyecto es:

- *Desarrollar e implementar una base de datos de los recursos licenciados con CC en Ecuador.*

Para su cumplimiento demanda la realización de los siguientes objetivos particulares:

- Analizar la situación actual de las licencias Creative Commons en el Ecuador.
- Desarrollar e implementar una base de datos de los recursos licenciados en el Ecuador con CC.
- Desarrollar e implementar el mecanismo de comunicación de datos entre el sitio principal y la base de datos desarrollada.
- Desarrollar e implementar mecanismos de obtención de estadísticas de datos de los recursos licenciados y almacenados en la base de datos.
- Establecer indicadores para la medición e impacto de las licencias CC Ecuador.
- Desarrollar e implementar un sistema informático (frontal) para la base de datos.
- Vincular los sistemas informáticos entre los portales CC Ecuador y Propiedad intelectual de la UTPL.

Organización del informe

Para una mejor comprensión del tema de investigación, se ha organizado este informe de la manera siguiente:

Capítulo 1: Situación actual de las licencias Creative Commons

En este capítulo se realiza una breve descripción de la situación actual de las licencias Creative Commons en el Ecuador. Además, se definen los tipos de licenciamiento alternativos existentes y el proceso de adaptación legal de las licencias Creative Commons al marco jurídico del país.

Capítulo 2: Web Crawling-Situación actual

Este apartado describe cómo la búsqueda de información en línea se convierte en una parte primordial de cualquier actividad investigativa, debido a la necesidad de información relevante y actualizada por parte de los usuarios. Además, se describen los componentes relevantes de la construcción de un crawler, así como sus características y arquitectura.

Capítulo 3: Análisis y diseño de la aplicación

Este capítulo describe los procesos de análisis y diseño de la aplicación. Se detallan: las entradas y salidas del sistema, la especificación de los casos de uso más relevantes, el diagrama de clases y el diseño de la base de datos mediante los modelos de entidad-relación y relacional de la misma.

Capítulo 4: Desarrollo e implementación de la aplicación

Este apartado describe los procesos de desarrollo e implementación de la aplicación. Se describe el proceso de construcción y codificación de la aplicación, así como los pasos a seguir para la implementación de la misma.

Capítulo 5: Pruebas y resultados

En este capítulo se detalla el plan de pruebas realizadas al software desarrollado. Se incluyen las pruebas de funcionalidad, de compatibilidad y de robustez realizadas a la aplicación con sus respectivos resultados.

Finalmente se detallan las conclusiones de este trabajo de investigación y el desarrollo de la aplicación, así como recomendaciones para su aplicación en proyectos futuros. Como resultado de esta tesis se obtuvo un crawler flexible y robusto que puede ser modificado o actualizado de acuerdo a las necesidades del usuario.

CAPÍTULO 1

SITUACIÓN ACTUAL DE LAS LICENCIAS CREATIVE COMMONS

En este apartado se ofrece una visión general acerca de la organización Creative Commons y su implicación dentro del desarrollo y creación de contenidos en la sociedad de la información. Se inicia presentando la organización y las iniciativas que propone para ayudar a los creadores de contenidos a seleccionar la versión legal bajo la que desean publicar sus creaciones. Además, se ofrece una visión de la situación de Creative Commons en el Ecuador.

1.1. Fuentes de información

Una fuente de información es todo aquello (documento, persona u otro objeto) que nos proporciona datos para el análisis y tratamiento de un problema de investigación planteado [1].

Las fuentes pueden ser escritas y no escritas. Las escritas o documentales son todos los documentos impresos que nos proveen información (fuentes bibliográficas); las no escritas provienen de la tradición oral de los pueblos (en especial aborígenes) recogidas por el autor de la investigación. Se debe tomar en cuenta que, en determinados contextos, los términos autor y fuente son sinónimos [1].

1.1.1. Clasificación de las Fuentes

Según su originalidad u orden de precedencia las fuentes pueden clasificarse en [1]:

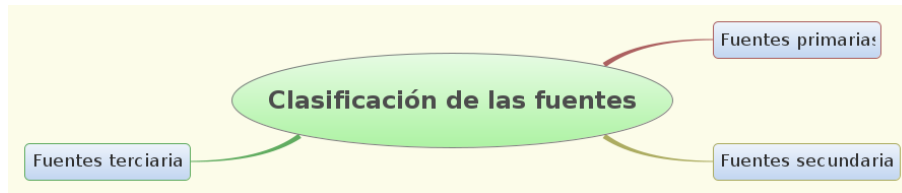


Figura 1.1.: Clasificación de las fuentes (22).

- Fuentes primarias: se incluyen en esta categoría a todos los documentos que proporcionan información directa y original, no abreviada ni traducida, de los hechos que son objeto de estudio.

En otras palabras, la información que ofrecen no ha sufrido ningún proceso de reelaboración, síntesis o interpretación. Como ejemplos se citan: monografías, artículos de revista, poemas, novelas, cartas e informes [1].

- Fuentes secundarias: en esta categoría se encuentran los documentos que han sufrido un proceso de reelaboración por parte de personas distintas a los autores. Como ejemplos se citan: los comentarios a las obras, la interpretación de las leyes, reglamentos y discursos [1].
- Fuentes terciarias: en esta categoría se engloban las selecciones y recopilaciones de fuentes primarias y secundarias. Por ejemplo: bibliografías, catálogos de bibliotecas, directorios, y artículos sobre encuestas realizadas [1].

Los trabajos de investigación, los libros de texto, las enciclopedias y la propia Wikipedia suelen reunir los tres tipos de fuentes citados anteriormente.

En el desarrollo de esta tesis se utiliza esta clasificación para discernir la relevancia de las fuentes utilizadas en esta investigación; además, en la elaboración de las citas o referencias bibliográficas de los documentos relacionados al tema de estudio.

1.1.2. Tratamiento de las fuentes

El ciclo de tratamiento de las fuentes se describe a continuación [2]:

1. Situación actual de las licencias Creative Commons

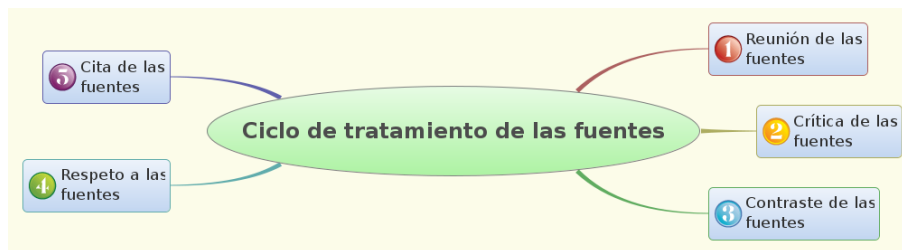


Figura 1.2.: Ciclo de tratamiento de las fuentes (22).

Reunión de fuentes

Como primer paso en el proceso de investigación se deben reunir todas las fuentes de información relevantes sobre el tema de estudio. Esto para tener una idea de todas las aristas que tiene el tema escogido y de los diversos puntos de vista de diversos autores sobre el mismo [2].

Crítica de las fuentes

Reunidas las fuentes, el investigador debe realizar un análisis concienzudo sobre la validez del material recogido. Es de especial interés que las fuentes recogidas se correspondan con la realidad y la reflejen de forma veráz [2].

Contraste de fuentes

El contraste de las fuentes permite encontrar discrepancias y coincidencias entre ellas. Si el grado de discrepancia es alto deberá aplicarse un método estadístico apropiado para obtener conclusiones valederas de su análisis [2].

Respeto a las fuentes

Aunque las opiniones vertidas en los documentos utilizados como fuentes para una investigación no sean del agrado del investigador, éste debe transcribirlas con fidelidad sin tergiversarlas de manera que las conclusiones obtenidas sean consistentes y valederas [2].

Cita de las fuentes

No se debe ocultar las fuentes en las que se basa la investigación. Tampoco se debe abusar de las citas, pues si la mayor parte del texto se encuentra entre comillas no existirá ninguna aportación original sino un simple plagio [2].

En toda investigación objetiva debe reconocerse el aporte de las fuentes en el texto mismo, en el pie de página o al final del capítulo o de la obra misma; reconociendo así que no todo lo escrito es producción propia.

Al desarrollar el presente trabajo se utilizó el ciclo de tratamiento de las fuentes citado anteriormente para reunir una base documental que permita la gestión de la información textual así como respaldar y documentar adecuadamente el trabajo de investigación complejo respetando la autoría de las diversas fuentes.

1.2. Tipos de licencias

1.2.1. Creative Commons

Creative Commons (CC) es una organización sin fines de lucro fundada en el año 2001 en la Stanford Law School (Facultad de Derecho de la Universidad de Stanford). Nace a partir de los esfuerzos de una serie de productores y estudiosos de la propiedad intelectual y el mundo digital en diferentes universidades estadounidenses [7].

Los esfuerzos de todos estos participantes están encaminados a combatir la situación de amenaza mediática y legal generada respecto al tema de los derechos de autor y la propiedad intelectual, sobre todo en lo que concierne al papel que juega el ciberespacio en este terreno. De ahí que, frente al popular *all rights reserved* (todos los derechos reservados) que usualmente se adjunta en las notas de copyright de muchas creaciones, CC propone como lema el *some rights reserved* (algunos derechos reservados).

Con este fin, CC ofrece distintos tipos de licencias que los creadores pueden utilizar para acompañar a sus obras, aclarando que no por ello se compromete en modo alguno a establecer ningún tipo de relación jurídica con los usuarios de estas licencias, o que asuma entre sus objetivos el ofrecer a éstos asesoramiento o representación legal [7].

Tipos de licencias

Básicamente, el hecho de que un autor decida publicar su trabajo mediante una licencia CC significa que debe escoger, de entre los derechos que posee, aquellos que está dispuesto a ceder. Notemos por tanto cómo desde el principio el planteamiento es radicalmente diferente al tradicional copyright, en el que el autor reserva absolutamente todos sus derechos sobre la obra.

Es importante señalar que en todos los casos de licencias CC se presupone que el autor otorga los derechos de copia y distribución de la obra. Partiendo de esa base, se toman en consideración los siguientes derechos que el autor puede elegir reservarse o no [7]:

- Reconocimiento (Attribution BY): Se permite la copia, distribución y presentación pública de la obra y trabajos derivados de la misma siempre que se reconozca y cite adecuadamente al autor original.
- No comercial (Non-commercial NC): Se permite la copia, distribución y presentación de la obra y trabajos derivados de la misma siempre que se realice con fines no comerciales.
- Prohibición de obras derivadas (No Derivative ND): Se permite la copia, distribución y presentación de la obra en su versión original, pero se prohíbe la realización de trabajos derivados de la misma.
- Redistribución bajo la misma licencia (Share Alike SA): Se permite la distribución de trabajos derivados de la obra siempre que se realice bajo una licencia idéntica a la que ampara la obra original.

Además, se encuentra en marcha el proyecto Icommons (International Commons)[16] cuyo objetivo principal es adaptar las licencias CC a la jurisprudencia y parámetros de los países que deseen adoptarla como parte de su legislación [7].

1.2.2. Licencias — Copyleft

Definición

Copyleft es un grupo de licencias cuyo objetivo es garantizar que cada persona que recibe una copia de una obra pueda a su vez usar, modificar y redistribuir el propio trabajo y las versiones derivadas del mismo. Unas veces se permite el uso

1. Situación actual de las licencias Creative Commons

comercial de dichos trabajos y en otras ocasiones no, dependiendo de los derechos que quiera ceder el autor [8].

1.2.3. Clasificación

Las licencias que engloba Copyleft son las siguientes:

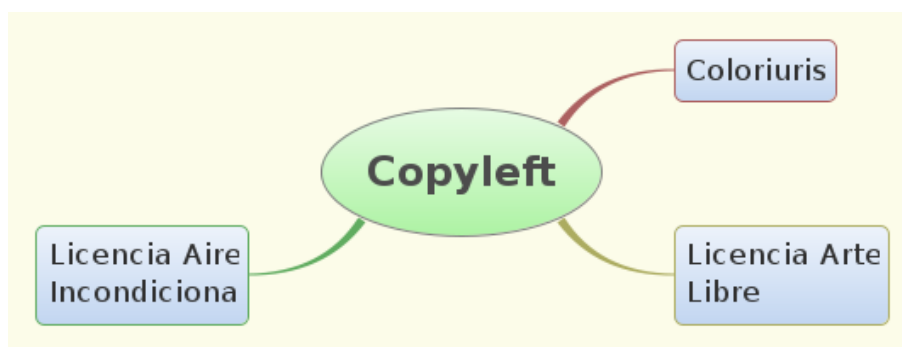


Figura 1.3.: Licencias Copyleft (22).

A continuación se realiza una breve descripción de cada una de ellas.

Coloriuris

ColorIURIS es un sistema internacional de gestión y cesión de derechos de autor con efectos legales en 25 países (Argentina, Bolivia, Brasil, Chile, Colombia, Costa Rica, Cuba, Ecuador, El Salvador, España, Estados Unidos, Francia, Guatemala, Honduras, Inglaterra, Irlanda, México, Nicaragua, Panamá, Paraguay, Perú, Portugal, República Dominicana, Uruguay, y Venezuela) [17].

Este sistema garantiza, mediante un contrato legalmente celebrado entre las dos partes, que de darse una utilización indebida de los contenidos objeto de la cesión, su titular tendrá una herramienta legal ante la justicia ordinaria.

Coloriuris fue creado para los creadores de contenido — libros, música, audio y video y fotografías — que utilizan la World Wide Web para su difusión y/o publicación; que desean ceder los derechos sobre sus creaciones dentro y fuera de la red [17].

1.2.4. Licencia Arte Libre

Bajo el principio de que el saber y la creatividad son recursos que deben permanecer libres para seguir siendo lo que son: conocimientos y creación, conceptualizados como una labor de investigación fundamental no determinada por sus aplicaciones en la vida real [18].

Se desarrolló la Licencia Arte Libre con un sólo fin: proteger las prácticas artísticas y liberarlas de las reglas del neoliberalismo.

1.2.5. Licencia Aire Incondicional

Esta licencia, creada por el abogado Abel Garriga, fue diseñada para la exposición "Aire Incondicional" llevada a cabo en el Centro de Arte Shedhalle (Zürich) y aplicada a una serie de contenidos dentro de la exposición y a otros realizados durante el tour de presentaciones en Suiza [19].

La licencia Aire Incondicional fue concebida y redactada en idioma español de acuerdo a las normas vigentes en España; de tal forma que pueda ser modificada y aplicada por todos los hispanohablantes que deseen hacerlo. Además, se debe tomar en cuenta que no existen este tipo de iniciativas en idioma español sino solamente en lengua inglesa [19].

1.3. Análisis de la situación actual

El proceso seguido en nuestro país para la implementación de las licencias Creative Commons es similar al desarrollado en otras regiones del mundo que adaptaron las licencias CC a su legislación. Como primer paso del proceso se conformó grupos de expertos en Derecho e Informática quienes colaboran con el equipo de CC para elaborar un documento borrador que se somete a debates y modificaciones hasta llegar a la redacción del documento definitivo que se utiliza como base para la elaboración de las licencias.

En relación con la adaptación de las licencias en nuestro país, contamos con una completa adecuación de las mismas a nuestro marco legislativo. Se encuentra disponible una traducción de las licencias al castellano gracias al esfuerzo de un equi-

1. Situación actual de las licencias Creative Commons

po de voluntarios de la Universidad Técnica Particular de Loja con el que también colaboraron el equipo de servicios jurídicos y de traducción de dicha institución.

Fue necesario adaptar las licencias Creative Commons a Ecuador porque, si bien el derecho de autor es uno de los regímenes jurídicos más estandarizados a nivel internacional, las diferencias entre sistemas y regulaciones siguen existiendo por lo que la idea es mantener un modelo común de licencia pero ajustado a la legislación de cada país y además, utilizando su terminología de manera que facilite su comprensión.

El objetivo de este trabajo es contar con una herramienta legal útil y adecuada a los adelantos tecnológicos y las herramientas informáticas que la UTPL utiliza como medios de aprendizaje; aunque, una vez obtenidas las licencias estas son para uso de todo el país y de todas aquellas personas que deseen licenciar sus creaciones bajo esta jurisdicción.

1.4. Indicadores

“Un indicador numérico es una relación entre dos o más datos significativos que proporcionan información sobre aspectos críticos o de importancia vital para la toma de decisiones.”[15]

Dentro de los indicadores numéricos encontramos diferentes clases:

- Indicadores aritméticos:
 - Razones: expresan la relación entre dos datos, se obtiene de dividir un dato (numerador) para la base (denominador). Tanto numerados como denominador deben ser unidades de medida de la misma naturaleza.
 - Porcentajes: muestra el valor de una parte con relación al todo.
- Indicadores estadísticos:
 - Promedio: este tipo de indicador a su vez se subdivide en:
 - Media: suma de un conjunto de valores dividido entre la cantidad de valores sumados.
 - Mediana: es el valor que ocupa la posición central cuando los datos son ordenados de menor a mayor.

1. Situación actual de las licencias Creative Commons

- Moda: representa el mayor número de veces que se repite un valor en la muestra tomada.

En su calidad de medida todo indicador debe cumplir las siguientes condiciones:

Observable: Debe referenciar las características, manifestaciones o elementos del objeto que puedan ser analizadas y constatadas. Debe permitir replicación y verificación, por tanto no debe hacer referencia a hechos ocasionales o meramente circunstanciales.

Aplicable: Los medios y recursos que se utilicen para realizar la observación y recoger la información deben encontrarse a disposición del investigador.

Codificación: La información proporcionada por los indicadores puede convertirse en índices (valores numéricos que reflejan tasa, ratio o frecuencia) o en descriptores (que reflejen la presencia/ausencia de un elemento o el grado de consecución del mismo). El tratamiento de tales valores numéricos debe respetar algunas reglas:

- No deben sumarse indicadores cuando uno es causa de otro.
- No deben sumarse indicadores cuando uno forma parte del otro
- Si los objetivos o finalidades de cada indicador son diferentes, no se los debe combinar [15].

1.5. Definición de los requerimientos

1.5.1. Elementos arquitectónicos y de diseño

1. Uso de PHP para aceptar y construir las solicitudes de página por las razones siguientes:
 - a) Bajo costo: el costo de utilización de este lenguaje de programación es cero (PHP es gratuito).
 - b) Portabilidad: está disponible para todas las plataformas existentes (Windows, Mac OSX, Linux, etc.).
 - c) Bases de datos: tiene soporte nativo para diversas bases de datos relacionales (MySQL, PostgreSQL, etc.).

1. Situación actual de las licencias Creative Commons

- d) Librerías: posee una gran variedad de librerías incluídas con la versión estándar entre ellas la GD que permite realizar gráficos de todo tipo.
2. La aplicación Web utilizará los siguientes patrones de diseño.
- a) Modelo Vista Controlador: Separa, elementos discretos para administrar la lógica de negocio, construcción y presentación de la interfaz de usuario, y coordinar y controlar sus actividades.
 - b) Controladores controlados: Definición de varios controladores concurrentes en la aplicación.
 - c) Lista paginada: Una manera de presentar las grandes listas de datos al usuario una página a la vez. El usuario podrá desplazarse a través de las páginas de datos una a la vez para examinar la lista. Esto mantiene el tamaño máximo de página y el tiempo de respuesta en un nivel manejable.

1.5.2. Dominio

1. La aplicación debe seguir los estándares de las aplicaciones comerciales. Su interfaz debe ser simple sin contener muchas especificaciones.
2. El dominio del problema es un sitio que provee datos e información estadística sobre las licencias Creative Commons en Ecuador.
3. Las licencias Creative Commons serán categorizadas en grupos. El enfoque inicial considera los conjuntos de licencias de la UTPL y aquellos de otras fuentes procedentes de Ecuador.
 - a) La clasificación licencias CC de la UTPL las agrupa por tipo de licencia y por tipo de recurso.
 - b) La clasificación licencias CC de otras universidades, también las agrupa por tipo de licencia y por tipo de recurso.

1.5.3. Entorno de ejecución

1. La aplicación debe ser capaz de ejecutar tanto el cliente como el servidor en la misma máquina para facilitar el estudio y la experimentación. La aplicación

1. Situación actual de las licencias Creative Commons

- también puede ser desplegada en un nodo independiente, sin cambios en el diseño.
2. El uso de una entidad formal y la capa de datos debe reducirse al mínimo o eliminarse, si es posible. El enfoque de la técnica de modelado se encuentra en la capa de presentación. Los demás niveles se mantendrán ocultos mientras sea posible.

1.5.4. Rendimiento

1. Cada servidor Web en el sistema deberá ser capaz de manejar al menos 150 sesiones de usuarios simultáneos.
2. El tiempo de respuesta debe ser de 7 segundos para todas las páginas durante la operación normal.
3. El sistema requerirá no más de 5 segundos para recuperar y responder a la petición de un cliente de una página Web estática.
4. El sistema requerirá no más de 10 segundos para responder a una página dinámica.

1.5.5. Seguridad

1. No hay requisitos especiales (o políticas) de seguridad para esta aplicación ya que los datos utilizados no son de carácter confidencial sino de uso público y general.

1.6. Definición de las estadísticas a obtener

1. Cantidad de recursos licenciados con CC por la UTPL por año.
2. Cantidad de recursos licenciados con CC por la UTPL por mes.
3. Cantidad de recursos licenciados con CC por la UTPL por tipo de licencia.

1.7. Definición de los indicadores a obtener

1. Número (máximo y mínimo) de recursos licenciados por la UTPL año.
2. Número (máximo y mínimo) de recursos licenciados por la UTPL por mes.
3. Número (máximo y mínimo) de recursos licenciados por la UTPL por tipo de licencia.

A modo de conclusión se debe acotar que la utilización de las licencias creative commons es incipiente en nuestro país y en toda latinoamérica. Sin embargo, la información sobre los tipos de licencias y el análisis de la situación actual permitieron la definición de los requerimientos, las estadísticas e indicadores que dirigirán el desarrollo de la aplicación.

Los requerimientos de la aplicación planteados en las secciones 1.5, 1.6 y 1.7 fueron escogidos de acuerdo a las necesidades planteadas por la UTPL de manera que permitan obtener tanto datos como información confiables al momento de evaluar el impacto y la utilización de este tipo de licencias en el Ecuador. La aplicación desarrollada facilitará esta evaluación poniendo a disposición del usuario la información actualizada y resumida del uso de las licencias CC debidamente categorizada.

CAPÍTULO 2

WEB CRAWLING: SITUACIÓN ACTUAL

En este capítulo, se estudia algunos trabajos relacionados cuya información es relevante para esta tesis. Especial atención reciben las características y la arquitectura del web crawler.

En la mayoría de publicaciones se enfatiza que el análisis de los links (enlaces) es un tema de investigación actual para los interesados en el desarrollo de algoritmos de recuperación de la información. La importancia de la web, hoy en día, radica en dos factores: es la piedra angular de la sociedad de la información y es utilizada, cada día, por millones de personas alrededor del mundo; naturalmente, ofrece oportunidades únicas de negocios y de investigación. El análisis de los enlaces es un tema relativamente nuevo en las investigaciones realizadas sobre la web.

Por otro lado, el tema del diseño de un crawler no tiene un lugar preponderante entre los investigadores ya que existen pocas publicaciones del tema disponibles. La investigación en el campo del crawling se ve afectada por el secreto empresarial ya que los motores de búsqueda sirven de mediadores entre los usuarios y los sitios web y se han convertido en la clave del éxito de muchos portales.

2.1. Web Crawler

Un Web Crawler (robot web o araña web) es, esencialmente, un agente explorador; un programa diseñado para explorar internet de forma organizada y metódica aprovechando las conexiones existentes entre las páginas web para moverse de una a otra [23].

El propósito del Crawler es recolectar las páginas web y añadirlas a un repositorio local para su posterior utilización; entre los posibles usos se encuentran los siguientes: proveer datos para motores de búsqueda, validar código HTML o verificar la estructura de un sitio web.

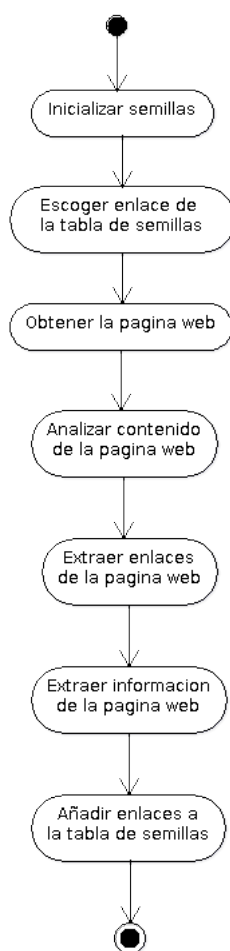


Figura 2.1.: Diagrama de actividad del Crawler (22).

La figura 2.1 muestra el diagrama de actividades de un Web Crawler. La primera actividad a realizar es mantener una lista de páginas web denominadas semillas;

éstas son obtenidas a partir de un directorio web. Cada ejecución del Crawler implica escoger el siguiente enlace de las semillas, obtener la página web, analizar el contenido de la página para extraer los hipervínculos y la información relevante para la aplicación y, añadir los vínculos nuevos a la tabla de semillas.

Si observamos detenidamente el diagrama de actividades del Web Crawler (figura 2.1) una de las actividades principales es “escoger enlace de la tabla de semillas”; parte esencial de esta actividad es el criterio utilizado para escoger el siguiente enlace a ser visitado. En el apartado siguiente se realiza un breve análisis de los algoritmos de búsqueda más utilizados.

2.2. Características del Web Crawler

Hay dos características importantes de la Web que generan un escenario en el que el web crawling es muy difícil: el gran volumen de contenido y la tasa de cambio (existe gran cantidad de páginas que se agregan, cambian y quitan todos los días).

Además, la velocidad de la red ha tenido una leve mejora frente a las actuales velocidades de procesamiento y capacidad de almacenamiento. El gran volumen implica que el crawler sólo puede descargar una fracción de las páginas web en un tiempo determinado, por lo que debe dar prioridad a su descarga. La alta tasa de cambio implica que en el momento en que el crawler está descargando las últimas páginas de un sitio, es muy probable que otras páginas se han añadido al sitio, o que las páginas se actualizaron o incluso se suprimieron.

El Crawler no es una foto "instantánea" de la Web, ya que no representa la Web en un momento dado. Las últimas páginas rastreadas probablemente se representan con exactitud, pero las primeras páginas que fueron descargadas tienen una alta probabilidad de haber cambiado.

El comportamiento de un rastreador web es el resultado de una combinación de políticas:

- Una política de selección que indica que páginas serán descargadas.
- Una política de re-visitar los enlaces para comprobar si hay cambios en las páginas.
- Una política de cortesía para evitar la sobrecarga de los sitios Web rastreados.

2. Web Crawling: Situación actual

- Una política de paralelización que indica la forma de coordinar los esfuerzos de los crawlers lanzados en paralelo.

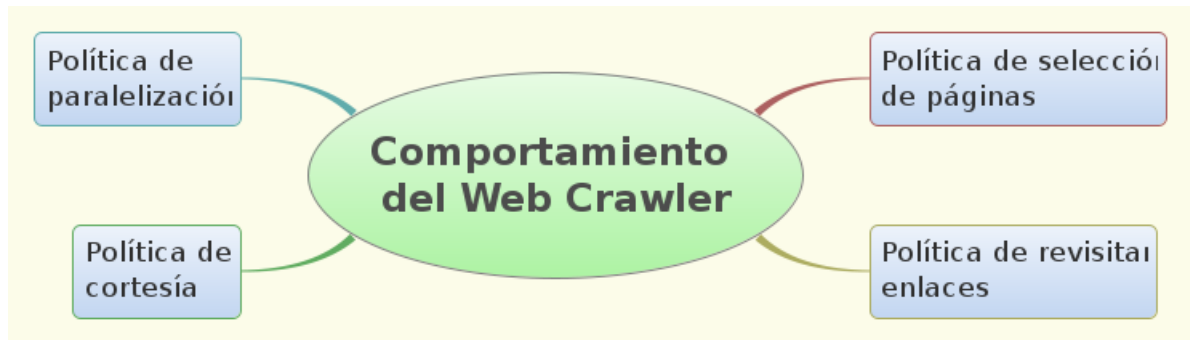


Figura 2.2.: Comportamiento del Web Crawler (22).

2.3. Técnicas de búsqueda

2.3.1. Búsqueda primero en anchura

La búsqueda del primero en anchura comprueba cada nodo sobre el mismo nivel antes de proceder a analizar un nivel más profundo. La figura 2.3 muestra un ejemplo de este tipo de búsqueda (utilizando grafos) teniendo el nodo C como meta [23].

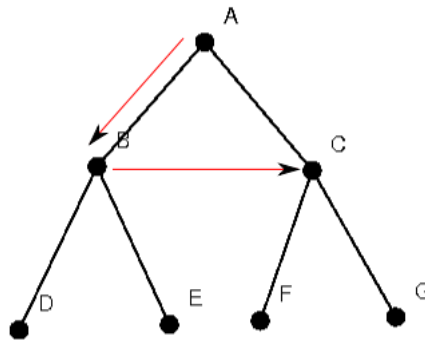


Figura 2.3.: Exploración primero en anchura (22).

La figura 2.3 muestra que la búsqueda comprueba los nodos ABC, esta búsqueda garantiza el encontrar una solución (si existe) aunque puede degenerar en una búsqueda exhaustiva.

2.3.2. Búsqueda primero el mejor

La búsqueda primero el mejor trata de expandir el nodo más cercano al objetivo, alegando que probablemente conduzca rápidamente a una solución. La figura 2.4 muestra un ejemplo de este tipo de búsqueda (utilizando grafos) teniendo el nodo C como meta [23].

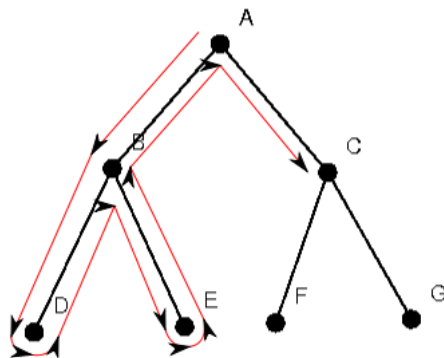


Figura 2.4.: Exploración primero el mejor (22).

La figura 2.4 muestra que este tipo de búsqueda se parece a la búsqueda primero en profundidad debido a que prefiere seguir un camino hacia el objetivo, aunque regresará si encuentra un callejón sin salida.

2.4. Arquitectura del Web Crawler

Todo crawler debe tener una buena estrategia de rastreo, como se señala en los apartados anteriores, pero también necesita una arquitectura altamente optimizado. Shkapenyuk y Suel [SS02] señalaron que:

"Si bien es fácil de construir un rastreador lento que descargue algunas páginas por segundo durante un corto período de tiempo, la construcción de un sistema de alto rendimiento que pueda descargar cientos de millones de páginas durante varias semanas se presenta una serie de retos en el sistema diseñado, E / S y eficiencia de la red, y la robustez y capacidad de gestión" [20].

Además, los web crawler son una parte central de los motores de búsqueda, y los detalles de sus algoritmos y arquitectura se mantienen como secretos comerciales. Cuando los diseños de crawlers se publican, a menudo hay una falta importante

2. Web Crawling: Situación actual

de detalle que impide que otros reproduzcan la obra. También hay nuevas preocupaciones acerca de "spam en los motores de búsqueda", que impiden que los principales motores de búsqueda publiquen sus algoritmos. La arquitectura típica (de alto nivel) de un crawler se muestra en la figura 2.5.

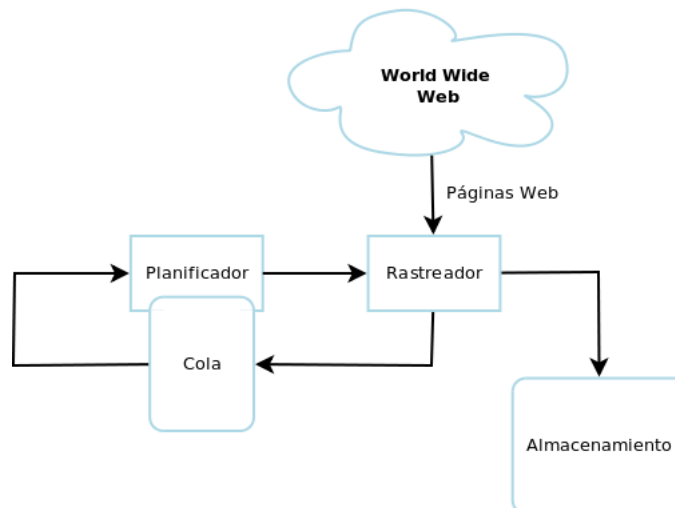


Figura 2.5.: Arquitectura de un Web Crawler(22).

2.4.1. Ejemplos de Crawlers

La siguiente es una lista de las arquitecturas de crawlers publicadas para los rastreadores de uso general (se excluye los crawlers web especializados), acompañada de una breve descripción que incluye los nombres dados a los diferentes componentes y las características excepcionales [24]:

RBSE fue el primer rastreador web publicado. El crawler RBSE es un mecanismo para explorar la estructura de la Web e indexar el material útil descubierto. Los objetivos planteados son los siguientes:

Separación de la estructura y del mecanismo de indexación; bajo impacto en la Web, la recuperación se limita únicamente a los documentos HTML; esfuerzo limitado en la obtención de resultados preliminares, para evaluar la utilidad del crawler.

Se basa en dos programas: el primero "spider" es el que crea y manipula una base de datos Oracle que contiene el grafo de la web utilizando los vínculos *.html o http:*/; y el segundo programa "mite", cuando recibe una url como

2. Web Crawling: Situación actual

argumento, recupera el documento en un archivo local e imprime las url encontradas.

WebCrawler se utilizó para construir el primer índice de texto completo a disposición del público de un subconjunto de la Web. Se basaba en lib-WWW para descargar las páginas, y otro programa para analizar y ordenar las URL para la exploración primero en profundidad de la red. También incluye un rastreador en tiempo real que siguió los enlaces basado en el algoritmo primero en anchura [25].

Este crawler está diseñado para identificar la intención de búsqueda del usuario y los resultados son listados generalmente en orden de importancia, ya que se reciben de varios motores de búsqueda. La clasificación y mezcla de los resultados depende de la naturaleza de la búsqueda

World Wide Web Worm fue un rastreador utilizado para construir un índice simple de los títulos de los documentos y URLs. El índice podría ser utilizado con el comando grep de UNIX.

Internet Archive Crawler es un rastreador diseñado con el propósito de archivar de forma periódica el estado de un subconjunto de la web. Utiliza varios procesos de forma distribuida, y asigna un número fijo de sitios Web a cada proceso. El intercambio entre procesos de las URL's se realiza en lotes con un intervalo de tiempo largo, debido a que este es un proceso costoso. El Internet Archive Crawler también tiene que lidiar con el problema de cambiar las DNS, por lo que mantiene un archivo histórico de los IP asignados.

WebSPHINX es una biblioteca de clases de Java y un entorno de desarrollo interactivo de crawlers para la web. Un crawler o spider es un programa que visualiza y procesa las páginas web automáticamente.

Consta de dos partes: el Crawler Workbench y la librería de clases WebSPHINX. El primero, tiene un interfaz que le permite al usuario configurar y controlar el crawler; el segundo provee múltiples características que facilitan el desarrollo de los crawlers en java.

La librería de clases es open source disponible bajo licencia Apache [26].

Google Crawler se describe con cierto detalle solamente la primera versión de su arquitectura basada en C++ y Python. El crawler se integró en el proceso de indexación, porque el análisis de texto se hizo para la indexación de texto

2. Web Crawling: Situación actual

completo y para la extracción de URL. Hay un servidor de URL's que envía listas de URL's utilizadas por los rastreadores que se ejecutan en paralelo.

El motor de búsqueda Google tiene dos características importantes que ayudan a la producción de resultados de gran precisión. El primero hace uso de la estructura de enlaces, para calcular una puntuación en un ranking de calidad para cada web. Este ranking se llama PageRank. En el segundo, Google utiliza los enlaces para mejorar los resultados de la búsqueda.

Aparte del PageRank y el uso de texto ancla, Google tiene otras características. Primero, guarda una copia local de la información de todos los resultados obtenidos, lo cual hace extender el uso de la proximidad de la búsqueda. Segundo, Google guarda algunos detalles de la presentación visual de las webs tales como el tamaño de la letras, pues este indica que información tiene mayor relevancia. Tercero, el contenido total del HTML de las webs es almacenado en un repositorio (localmente).

CobWeb utiliza un "planificador" centralizado y una serie de "collectors" distribuidos. Los colectores analizan las páginas web descargadas y envían las direcciones URL encontradas al planificador, que a su vez las reasigna a los colectores. El planificador ejecuta una búsqueda primero en amplitud con una política de cortesía para evitar la sobrecarga de los servidores Web. El crawler está escrito en Perl.

Mercator es un web crawler modular escrito en Java. Su modularidad se deriva del uso de conectores "módulos de protocolo" y "módulos de procesamiento". Los módulos de protocolo determinan la forma de búsqueda de las páginas web, y los módulos de procesamiento determinan la forma de procesar las páginas Web. El módulo de procesamiento estándar sólo analiza las páginas y extrae nuevas URL's, pero otros módulos de procesamiento se pueden utilizar para indexar el texto de las páginas, o para recoger las estadísticas de internet.

El rastreo se realiza por múltiples subprocesos de trabajo, por lo general varios cientos. Cada trabajador realiza varias veces los pasos necesarios para descargar y procesar un documento. El primer paso de este bucle es remover una dirección URL absoluta de la frontera compartida de URL's y descargarla.

En el esquema de la URL, el trabajador elige el protocolo adecuado para des-

2. Web Crawling: Situación actual

cargar el documento. A continuación, invoca el método buscar, que descarga el documento de Internet utilizando un hilo por `RewindInputStream`. La RIS es una abstracción de la E/S que se inicializa desde un flujo de entrada arbitrario, y que posteriormente permite que el contenido se vuelva a leer varias veces. Una vez que el documento ha sido guardado, el subproceso de trabajo determina si este documento (asociado a una dirección URL diferente) se ha procesado antes. Si es así, el documento no se procesa más, y el subproceso de trabajo elimina la siguiente URL de la frontera.

WebFountain es un sistema distribuido, modular similar a Mercator, pero escrito en C++. Cuenta con un "controlador" que coordina una serie de "hormigas". Después de repetir la descarga de páginas, una tasa de cambio se desprende de cada página y un método de programación no lineal se debe utilizar para resolver el sistema de ecuaciones para maximizar el refrescamiento. Los autores recomiendan utilizar este orden en las primeras etapas del rastreo, y luego cambiar a un orden uniforme de rastreo, en el que todas las páginas se visitan con la misma frecuencia.

Este proyecto representa uno de los primeros intentos integrales para catalogar e interpretar los datos no estructurados de la Web de forma continua. Para ello los investigadores de IBM han propuesto nuevos sistemas para la recuperación precisa de subsectores de información de la web, análisis de la tendencia en tiempo real y análisis de meta nivel de la información disponible en la Web.

Polybot es un crawler de alto rendimiento desarrollado en el departamento de Informática y Ciencias de la Información de la Universidad Politécnica como parte de un proyecto de investigación académica que explora nuevas técnicas para la búsqueda y análisis de la web.

Es un sistema distribuido escrito en C++ y Python, que se compone de un "crawl manager", uno o más "downloaders" y uno o más "traductores de DNS". Las URL's recopiladas se añaden a una cola en el disco, y se procesan (por lotes) en busca de direcciones URL [20].

La gran mayoría de los crawlers revisados son herramientas comerciales o tienen licencias restrictivas que no permiten realizar modificaciones en el código fuente, debido a ello y a que las soluciones open source estudiadas estaban desarrollados en lenguajes de programación no compatibles con la aplicación producto de esta

2. Web Crawling: Situación actual

tesis, se creyó conveniente realizar uno propio de acuerdo a las especificaciones recopiladas en la investigación previa.

Se debe puntualizar que el crawler que será construido en el desarrollo de esta investigación cubre solamente las características básicas señaladas en los apartados 2.2 y 2.4; ya que lo que se necesita obtener con esta herramienta es una lista de enlaces a los recursos creative commons. No contiene, por tanto, diversas herramientas especializadas propias de los buscadores web (Bing o Google por ejemplo) que se encuentran desarrolladas en la mayoría de crawlers mencionados anteriormente (otra razón para su no utilización en este trabajo).

Para conseguir un directorio de los recursos creative commons se debe recorrer la internet localizando páginas cuya jurisdicción sea Ecuador o cuyo contenido ostente una licencia creative commons ecuatoriana, con el objetivo de obtener los datos necesarios, tanto acerca de la obra como sobre su autor, para la generación de las estadísticas requeridas por la aplicación.

CAPÍTULO 3

ANÁLISIS Y DISEÑO DE LA APLICACIÓN

En este capítulo se realizará un análisis detallado de los diferentes módulos que conforman la aplicación, las funciones que puede ejecutar el usuario y el esquema de la base de datos a implementarse.

El diseño de software es el proceso de determinar una implementación efectiva y eficiente que realice las funciones y tenga la información del análisis de dominio. Las siguientes actividades se plantean en esta etapa: determinar la arquitectura inicial (decisiones acerca de recursos de implementación, categorías y prototipos a desarrollar); determinar el diseño lógico (detalle al diagrama de clases); implementación física (interfaz a dispositivos o características propias de la implementación); y, refinamiento del diseño (incorporar el aprendizaje debido a los prototipos y cumplir con requerimientos de desempeño).

3.1. Arquitectura de la aplicación

El modelo lógico de la arquitectura de la aplicación ilustra tanto el funcionamiento de la misma como las interacciones entre el usuario y el software.

3. Análisis y Diseño de la aplicación

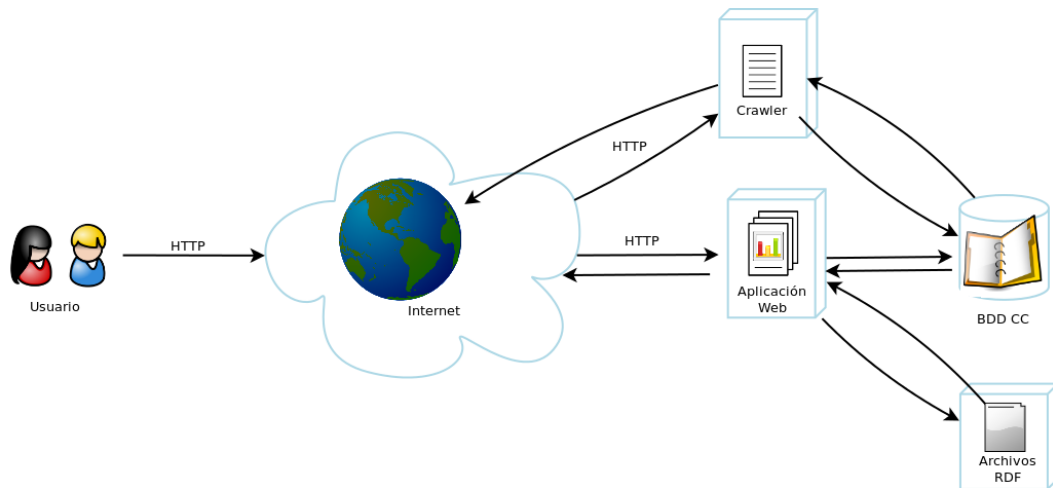


Figura 3.1.: Modelo lógico de la aplicación (22).

El software de la aplicación se instala como un script en el servidor Web. Éste produce las páginas que serán vistas mediante un navegador. Cuando se accede a una página de la aplicación, el navegador primero envía una consulta al servidor que administra los conjuntos de datos que contienen el software de la aplicación. Estos datos, son formateados para mostrarse en un navegador.

Para realizar esta tarea, el script traduce el contenido de la página al lenguaje HTML y lo embebe en una página web que es enviada al navegador. Por ejemplo, el script de la aplicación escrito en lenguaje PHP lee los datos de la base de datos MySQL para luego mostrarlos al usuario a través del navegador Firefox.

El contenido de la aplicación se visualiza a través de un navegador web (Firefox), bajo este entorno se realizan las operaciones de visualización y manipulación de los datos de la aplicación. El servidor web es el encargado de responder a todas las peticiones del navegador, para ello recurre al núcleo de la aplicación.

El núcleo de la aplicación (formado por los scripts y el código) es el encargado de transformar los datos obtenidos mediante el crawler en información estadística así como de generar las gráficas y los informes correspondientes. Además de proporcionar un interfaz (medio de comunicación) con la base de datos relacional. Finalmente, la base de datos relacional, es la encargada de almacenar la información de cada licencia registrada y de las páginas web con licencia creative commons encontradas por el crawler.

En cuanto a la plataforma, la aplicación está implementada con herramientas Open-Source, donde predomina la trilogía Linux-Apache-PHP utilizando como base de

3. Análisis y Diseño de la aplicación

datos relacional MySQL.

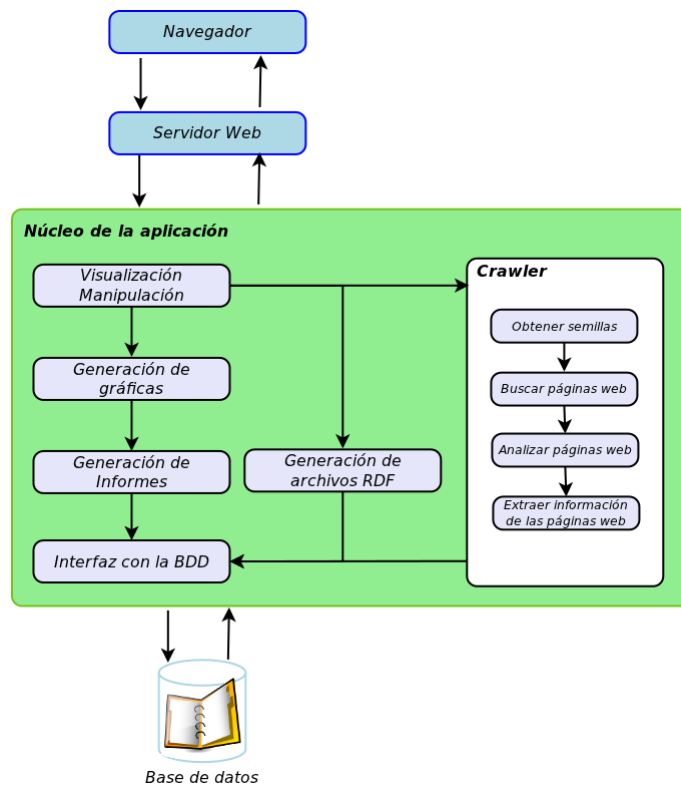


Figura 3.2.: Modelo físico de la aplicación (22).

El diagrama de secuencia de la aplicación que muestra en detalle las interacciones entre los componentes del software se muestra a continuación:

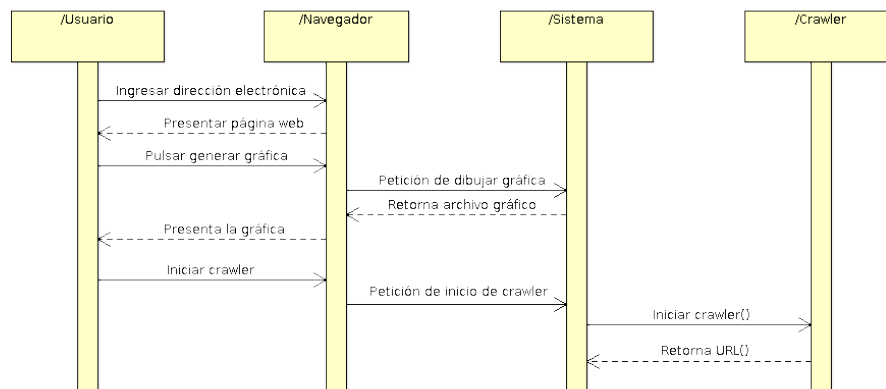


Figura 3.3.: Diagrama de secuencia del sistema (22).

3.2. Definición de las entradas

Los datos necesarios para el correcto funcionamiento de la aplicación son:

- Información de la obra licenciada.
 - Título
 - Autor
 - Formato (audio, video, imagen, texto, interactivo y otros)
 - Web del autor
 - Dirección de la obra
 - Más permisos
- Tipo de licencia CC utilizada.
 - BY
 - BY-NC
 - BY-SA
 - BY-ND
 - BY-NC-SA
 - BY-NC-ND
- Datos de las licencias registradas por día, semana, mes y año.

3.3. Definición de las salidas

Los resultados presentados al usuario se corresponden con las estadísticas siguientes:

- Cantidad de recursos registrados con licencias CC por la UTPL por año.
- Cantidad de recursos registrados con licencias CC por la UTPL por mes.
- Cantidad de recursos registrados con licencias CC por la UTPL por tipo de recurso.

3.4. Metodología y herramientas utilizadas

En el proyecto se utilizó el ciclo de vida en cascada, debido a la complejidad del proyecto y a la interdependencia de cada componente. Como complemento se utilizó las plantillas del Proceso Unificado de Desarrollo (RUP) mismas que se usaron para la especificación de los casos de uso, como lenguaje de modelado se trabajó con UML.

En la implementación del algoritmo se utilizó el lenguaje de programación PHP, para el modelo estadístico, y la programación de la interfaz grafica de usuario (GUI) se la realizó mediante Java Script. A continuación describimos las fases del ciclo de vida en cascada:

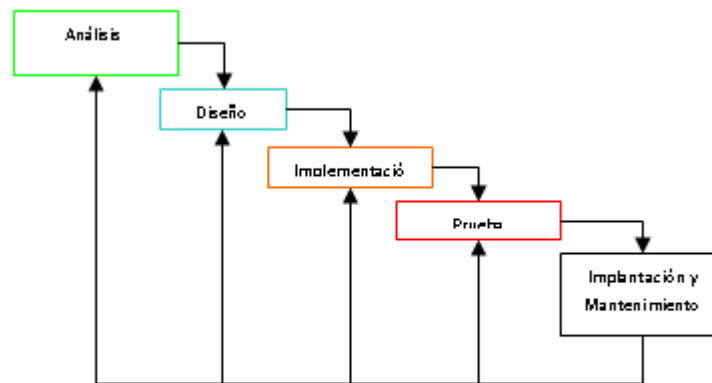


Figura 3.4.: Ciclo de vida clásico (22).

3.4.1. Análisis del sistema

El aspecto fundamental del análisis de sistemas es comprender todas las facetas importantes del problema que se encuentra bajo estudio. Los analistas, al trabajar con los usuarios finales, deben estudiar los procesos que intervienen en el problema para dar respuesta a las siguientes preguntas clave:

1. ¿Qué es lo que hace?
2. ¿Cómo se hace?

3. Análisis y Diseño de la aplicación

3. ¿Con que frecuencia se presenta?
4. ¿Qué tan grande es el volumen de transacciones o decisiones?
5. ¿Cuál es el grado de eficiencia con el que se efectúan las tareas?
6. ¿Existe algún problema?
7. Si existe un problema, ¿qué tan serio es?
8. Si existe un problema, ¿cuál es la causa que lo origina?

3.4.2. Diseño del sistema

El diseño de un sistema de información produce los detalles que establecen la forma en la que el sistema cumplirá con los requerimientos identificados durante la fase de análisis. Los especialistas en sistemas se refieren, con frecuencia, a esta etapa como diseño lógico en contraste con la del desarrollo del software, a la que denominan diseño físico.

3.4.3. Desarrollo del software

Los encargados de desarrollar software pueden instalar software comprobando a terceros o escribir programas diseñados a la medida del solicitante. La elección depende del costo de cada alternativa, del tiempo disponible para escribir el software y de la disponibilidad de los programadores. Por lo general, los programadores que trabajan en las grandes organizaciones pertenecen a un grupo permanente de profesionales.

3.4.4. Prueba del software

Durante la prueba de sistemas, el sistema se emplea de manera experimental para asegurarse de que el software no tenga fallas, es decir, que funciona de acuerdo con las especificaciones y en la forma en que los usuarios esperan que lo haga. Se alimentan como entradas conjunto de datos de prueba para su procesamiento y después se examinan los resultados.

3.4.5. Implantación y evaluación

La implantación es el proceso de verificar e instalar nuevo equipo, entrenar a los usuarios, instalar la aplicación y construir todos los archivos de datos necesarios para utilizarla.

Una vez instaladas, las aplicaciones se emplean durante muchos años. Sin embargo, las organizaciones y los usuarios cambian con el paso del tiempo, incluso el ambiente es diferente con el paso de las semanas y los meses. Por consiguiente, es indudable que debe darse mantenimiento a las aplicaciones.

3.5. Especificación de los casos de uso

El sitio web de estadísticas tiene sólo cuatro casos de uso, por lo que los cuatro son arquitectónicamente significativos. La figura 3.5 muestra el caso de uso de alto nivel de la aplicación.

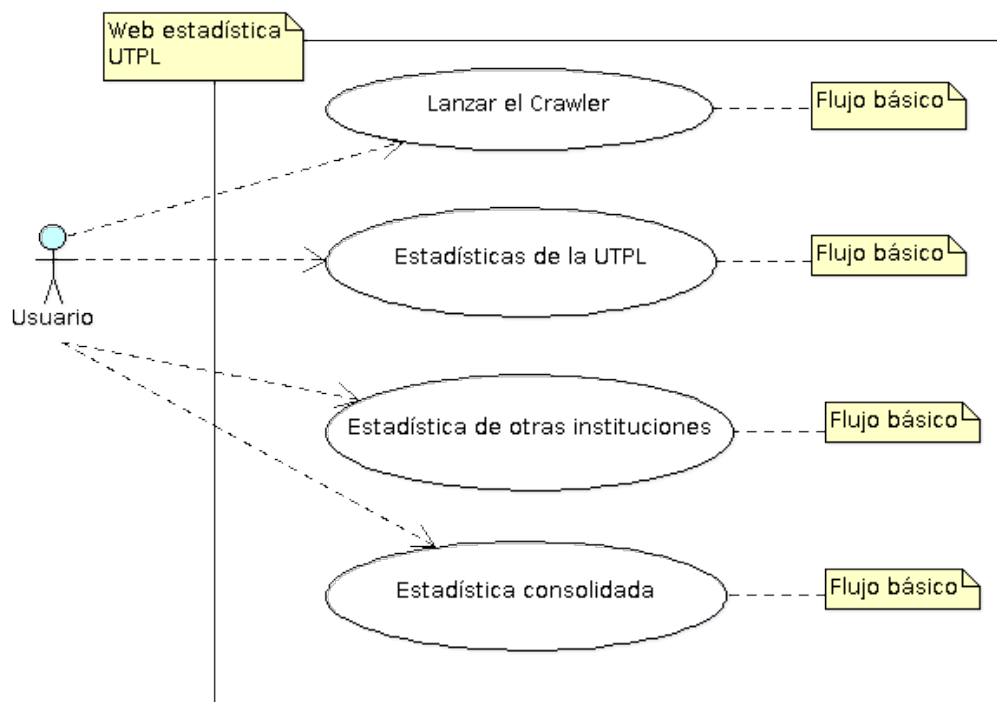


Figura 3.5.: Diagrama de casos de uso de la aplicación (22).

3. Análisis y Diseño de la aplicación

Esta aplicación cuenta con un solo actor: el usuario. El usuario utiliza la internet de forma anónima y navegará por las estadísticas de las licencias CC visualizándolas.

El software de estadística requiere la implementación de un crawler cuyas funciones son: exploración automática de páginas web, clasificación de documentos e, indexación de documentos.

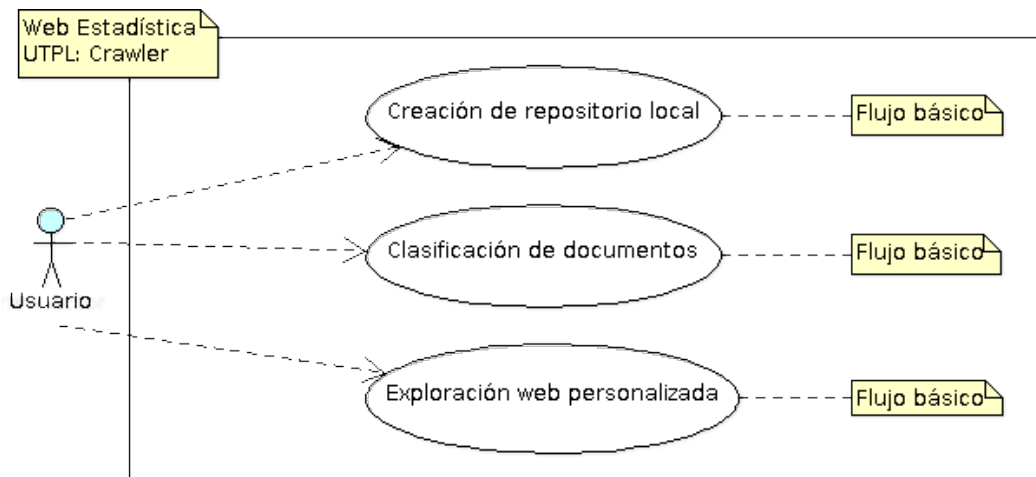


Figura 3.6.: Diagrama de casos de uso del Crawler (22).

El usuario es el único autor de estos casos de uso. El usuario utiliza de forma transparente el crawler, es decir, lo inicializa pero no puede observar de forma directa su funcionamiento. Los resultados de la ejecución del crawler solamente son visibles a nivel de la capa de datos.

3.6. Diagrama de clases de la solución

El diagrama de paquetes de alto nivel se detalla en la figura 3.7. Esta es una aplicación pequeña que se centra casi exclusivamente en el usuario y en la capa de presentación, por lo que el modelo incluye los elementos desarrollados para todas las capas en los mismos paquetes.

3. Análisis y Diseño de la aplicación

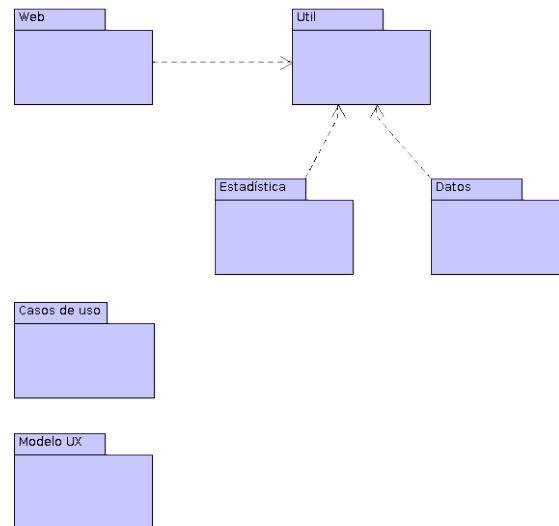


Figura 3.7.: Modelo de diseño de alto nivel de paquetes(22).

La capa de entidad se encuentra sobre la base de datos, es responsable de todas las reglas del negocio y la lógica del sistema. La figura 3.8 muestra las clases que componen la capa de entidad de la aplicación.

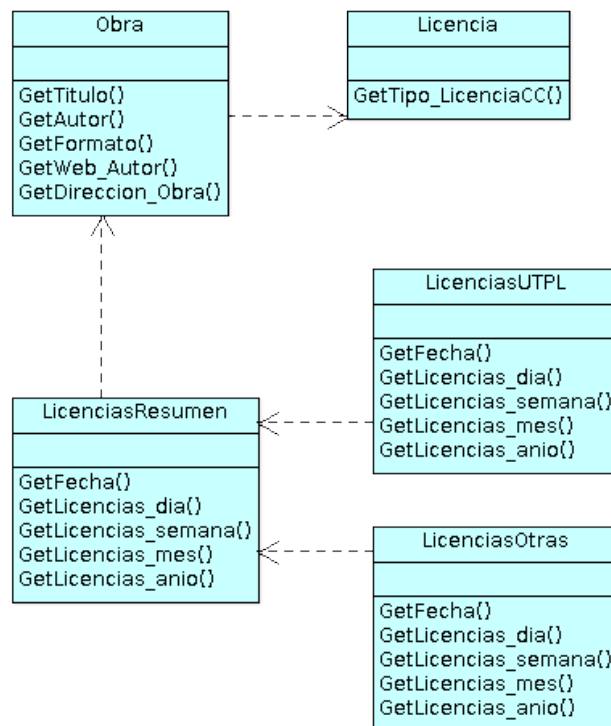


Figura 3.8.: Modelo de clases de la capa de entidad (22).

La capa de datos se ocupa de las operaciones de almacenamiento de la información del software desarrollado. En esta aplicación, la persistencia corresponde a la base de datos implementada en MySQL. La capa de datos se encarga de proporcionar los servicios de persistencia a la capa de entidad, es decir, almacena y extrae la información necesaria desde la base de datos detallada en los apartados 2.4 y 2.5 del presente capítulo.

3.7. Diseño de la base de datos

En este apartado se describe el modelo entidad-relación realizado para las bases de datos de licencias Creative Commons y de estadísticas; también se relata la transformación de éstos modelos en tablas y relaciones.

3.7.1. Modelo entidad-relación de la base de datos

El proceso de diseño se inicia tomando en cuenta los datos del formulario de licenciamiento proporcionado por Creative Commons Ecuador. Entre los elementos principales tenemos los tipos de obra (seis en total: audio, video, imagen, texto, interactivo y otros) y los tipos de licencia CC (seis en total: BY, BY-NC, BY-SA, BY-ND, BY-NC-SA, BY-NC-ND).

Para la información estadística que debe proporcionar la aplicación tuvieron especial importancia preguntas como ¿cuántos internautas obtuvieron su licencia?, ¿cuántas licencias Creative Commons obtuvo la UTPL durante el año 2010? deben obtener respuesta satisfactoria; además, debemos tener en cuenta los tipos de obra y los tipos de licencia CC para entregar información válida y detallada del proceso de licenciamiento.

El diagrama entidad relación de la figura 3.9 refleja todos los datos necesarios para realizar el licenciamiento de una obra y para obtener una estadística completa del licenciamiento de una obra bajo los parámetros de la Creative Commons.

3. Análisis y Diseño de la aplicación

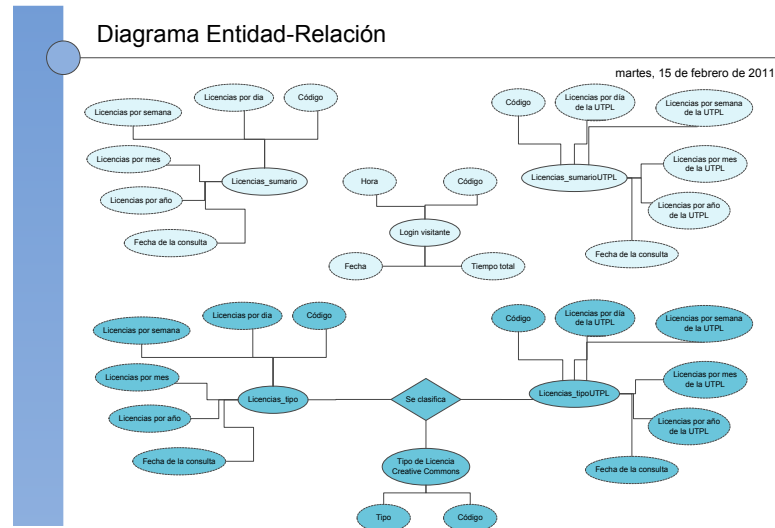


Figura 3.9.: Modelo E-R, BDD de la aplicación (22).

3.7.2. Modelo relacional de la base de datos

A continuación se traslada el modelo conceptual al modelo relacional en el que se incluye las relaciones (con su respectiva cardinalidad), las tablas con sus columnas respectivas y las claves (primarias y foráneas). La figura 3.10 muestra el modelo relacional de la base de datos que permitirá el funcionamiento tanto del licencia- miento como de las estadísticas de las licencias Creative Commons Ecuador.

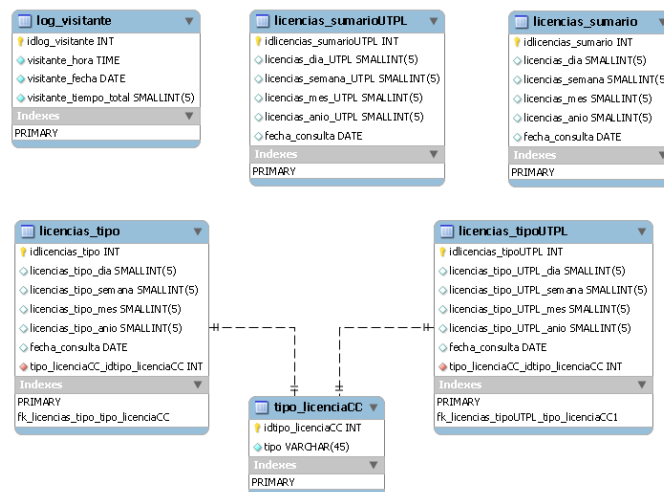


Figura 3.10.: Modelo relacional, BDD Estadísticas CC (22).

3.8. RDF (Resource Description Framework)

En esta sección se explica la utilización de RDF (Resource Description Framework) en la aplicación desarrollada. La inclusión de metadatos se realiza tanto para la descripción de los recursos creative commons como para la representación de los resultados de las estadísticas, esto con el fin de facilitar la posterior utilización de los mismos por otras aplicaciones informáticas.

3.8.1. Definición

Es un framework para metadatos, basado en XML, desarrollado por la W3C (World Wide Web Consortium) para el intercambio de datos en internet. Este modelo facilita la compartición de datos aunque los esquemas que los describen sean diferentes.

El significado de las siglas RDF se explica a continuación.

- Resource (recurso): páginas, imágenes, videos, ... cualquier objeto que tenga una URI.
- Description (describe): atributos, características y relaciones de los recursos.
- Framework (marco de trabajo): modelo, lenguajes y sintaxis para las descripciones.

RDF estructura las declaraciones de los recursos de la web en expresiones denominadas tripletas, las que a su vez son combinaciones sujeto-predicado-objeto. Este modelo permite que los datos puedan ser compartidos por diversas aplicaciones informáticas añadiendo significado a las páginas web (Web semántica).

Las tripletas pueden representarse gráficamente utilizando grafos dirigidos, donde los nodos representan el sujeto y los objetos y las relaciones del grafo simbolizan el predicado. Los componentes que describen el formato RDF son:

- Namespace XML.
- XML information set.
- XML base.

Cada triplete RDF desglosa el conocimiento en (sujeto, predicado, objeto), si tomamos como ejemplo el siguiente fragmento del conocimiento:

3. Análisis y Diseño de la aplicación

- La página *doc.html* tiene por autor a Fabián y por tema la música puede descomponerse en:
 - doc.html tiene por autor a Fabián.
 - doc.html tiene por tema la música.
- Las respectivas tripletas RDF serían:
 - (sujeto, predicado, objeto)
 - (doc.html, autor, Fabián)
 - (doc.html, tema, música)

El grafo dirigido se ilustra en la gráfica siguiente:

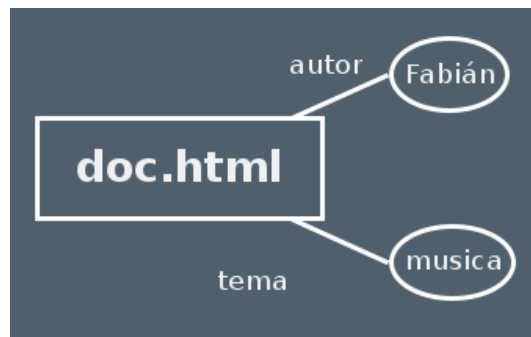


Figura 3.11.: Grafo dirigido que ilustra la estructura RDF (22).

Utilizando URI's para identificar los recursos y las propiedades tenemos:

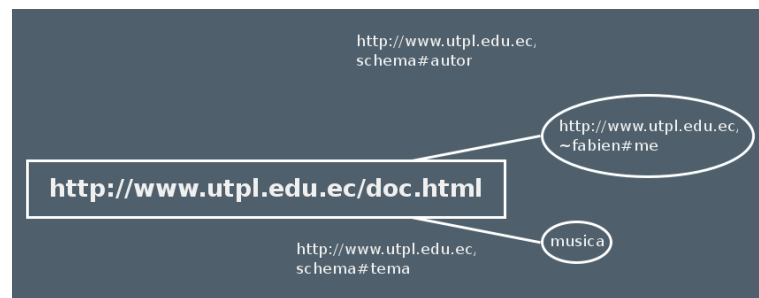


Figura 3.12.: Grafo dirigido (con URI's) que ilustra la estructura RDF (22).

El archivo RDF generado para el ejemplo tendría la estructura siguiente:

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:utpl="http://www.utpl.edu.ec/schema#" >
<rdf:Description rdf:about="http://www.utpl.edu.ec/doc.html">
<utpl:autor rdf:resource="http://www.utpl.edu.ec/~fabien#me" />
<utpl:tema>Musica</utpl:tema> </rdf:Description>
</rdf:RDF>
```

3.8.2. Descripción de RDF utilizado en la aplicación

En la aplicación se utiliza RDF tanto para los recursos creative commons encontrados en la web como para las estadísticas obtenidas a partir de estos recursos. El modelo utilizado varía de acuerdo al elemento descrito pero en general se utilizarán los metadatos DC (Dublin Core) cuando sea pertinente y para aquellos datos no descritos aún por este estándar se usará términos propios.

Para la descripción de los recursos se utilizó el modelo siguiente, basado totalmente en metadatos DC:

- Título *dc.title*
- Autor *dc.contributor.author*
- Idioma *dc.language.iso*
- Tipo *dc.type*
- Fecha *dc.date.issued*

En la definición de las estadísticas se utilizó el modelo:

- Título *dc.title*
- Autor *dc.contributor.author*
- Año, mes o semana *creativecommons.ec/año mes semana*
- Cantidad *creativecommons.ec/cantidad*

A continuación se presenta el esquema RDF detallado utilizado en la aplicación:

3. Análisis y Diseño de la aplicación

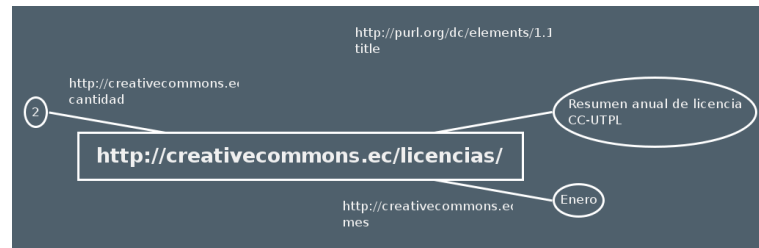


Figura 3.13.: Grafo dirigido (con URI's) del RDF de la aplicación (22).

CAPÍTULO 4

DESARROLLO E IMPLEMENTACIÓN DE LA APLICACIÓN

Una vez completados los casos de uso, realizado el análisis de la aplicación y definido el diseño adecuado, es hora de empezar el proceso de implementación. La principal actividad a realizarse es la implementación del sistema en términos de componentes, es decir, ficheros de código fuente, scripts, ficheros de código binario, etc., en el lenguaje de programación PHP.

4.1. Construcción de la aplicación

En esta sección se muestra la planificación del proceso de construcción de las páginas web que conforman la solución. Entre los criterios a considerar durante este proceso se tiene:

- La codificación de la funcionalidad completa de las páginas web.
- La implementación de los casos de uso determinados para la aplicación.

4. Desarrollo e Implementación de la aplicación

- La determinación de la trazabilidad de los casos de uso para su uso posterior en los escenarios de prueba.
- Determinar las estrategias de mantenimiento y actualización.

Para la implementación de esta aplicación cliente/servidor se utilizó el modelo de tres capas: capa de presentación, capa de negocios y capa de datos; distribuyéndose el código de la manera siguiente:

- Capa de presentación: Codificación de scripts utilizando el lenguaje JavaScript mediante el Dojo Toolkit.
- Capa de negocios: Codificación de scripts utilizando el lenguaje de programación PHP.
- Capa de datos: Implementación de la base de datos utilizando el motor MySQL.

La aplicación genera dinámicamente la información presentada al usuario final, para ello utiliza el modelo de implementación mostrado en la figura siguiente.

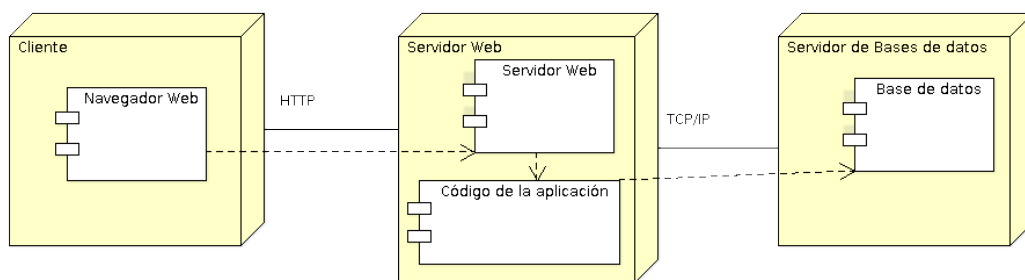


Figura 4.1.: Modelo de implementación de la aplicación (22).

La interfaz gráfica de usuario, implementada en la página principal de la aplicación, se presenta a continuación:

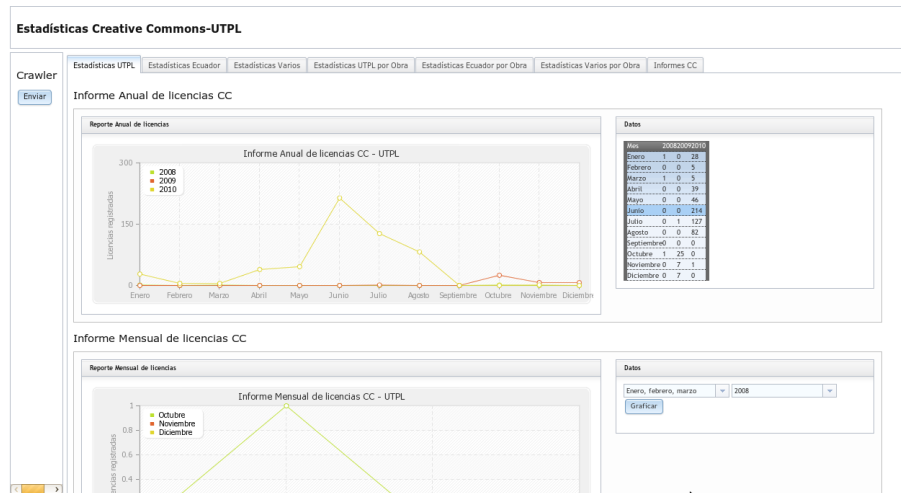


Figura 4.2.: Interfaz gráfica de usuario (GUI) de la aplicación (22).

4.2. Codificación del módulo de actualización de datos

La actualización de la información de la base de datos MySQL se realiza mediante el código siguiente:

```
<?php
if($link = pg_connect($connect_parameters)){
    $sSql = "select * from \"formato_obra\"";
    $sResult = pg_query($link, $sSql);
    if(pg_num_rows($sResult) > 0){
        print("<table border=\"1\">");
        print("<tr><th>ID</th><th>Formato</th></tr>");
        while($sRow = pg_fetch_object($sResult)){

            print("<tr><th>$sRow->idformato_obra</th>
            <td>$sRow->formato</td>");
            print("</tr>");
        }

        print("</table>");
    }
    ?>
```

Una descripción completa de la codificación del módulo de actualización de datos se encuentra en el apéndice B.

4.3. Codificación del módulo de estadística

El módulo de estadística maneja diversas funciones, entre ellas:

- Extracción de la información.
- Generación de la gráfica de acuerdo a los datos obtenidos.
- Presentación de la gráfica y de la información al usuario.

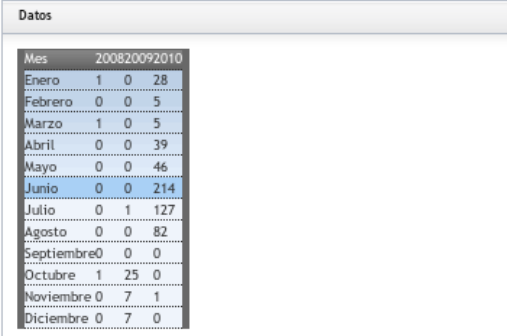
La extracción de la información a utilizarse en la creación de la gráfica estadística se realiza con el siguiente código PHP:

```
<?php
print("<table border=\"1\">");
print("<tr><th>Mes</th><th>2008</th><th>2009</th>

    <th>2010</th></tr>");

while (($datos = fgetcsv($fp, 50, ",", " ")) !== FALSE ) {
$numero = count($datos);
print ("<tr>");
for($c = 0; $c < $numero; $c++) {
print ("<td>$datos[$c]</td>");
}
print("</tr>");
}
print ("</table>");
?>
```

Los resultados de la ejecución del código anterior son los siguientes:



Mes	2008	2009	2010
Enero	1	0	28
Febrero	0	0	5
Marzo	1	0	5
Abril	0	0	39
Mayo	0	0	46
Junio	0	0	214
Julio	0	1	127
Agosto	0	0	82
Septiembre	0	0	0
Octubre	1	25	0
Noviembre	0	7	1
Diciembre	0	7	0

Figura 4.3.: Presentación de los datos (22).

La creación de la gráfica estadística se realiza mediante la librería GD de PHP, así:

4. Desarrollo e Implementación de la aplicación

```
<?php
// Definicion de datos
$DataSet->AddAllSeries(); $DataSet->SetAbsciseLabelSerie();
$DataSet->SetYAxisName("Licencias registradas");
$DataSet->SetYAxisUnit("");
// Configurar el grafico
$Test = new pChart(700,230);
$Test->drawScale($DataSet->GetData(),$DataSet->

    GetDataDescription(),SCALE_NORMAL,150,150,150,TRUE,0,2);

$Test->drawGrid(4,TRUE,230,230,230,50);
// Dibujar ejes
$Test->drawTreshold(0,143,55,72,TRUE,TRUE);
// Dibujar lineas
$Test->drawLineGraph($DataSet->GetData(),$DataSet->

    GetDataDescription());

$Test->drawPlotGraph($DataSet->GetData(),$DataSet->

    GetDataDescription(),3,2,255,255,255);

?>
```

Los resultados de la ejecución del código anterior son los siguientes:

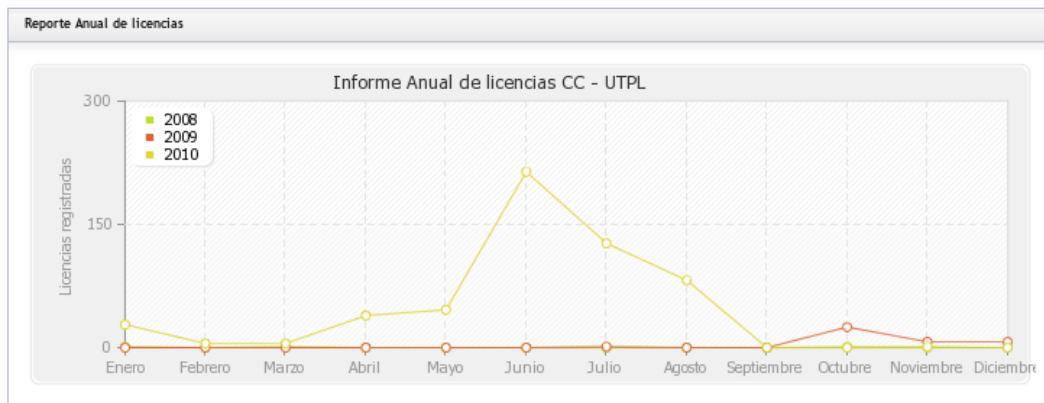


Figura 4.4.: Presentación de la gráfica (22).

La presentación tanto de la gráfica como de la información se realiza de la manera siguiente:

4. Desarrollo e Implementación de la aplicación

```
<div dojoType="dojox.widget.Portlet" title="Grafica">
<div>  </div>
</div>
<div dojoType="dojox.widget.Portlet" title="Datos">
<div> <?php require_once 'data/datacsv1.php'; ?>
</div>
</div>
```

Los resultados de la ejecución del código anterior son los siguientes:

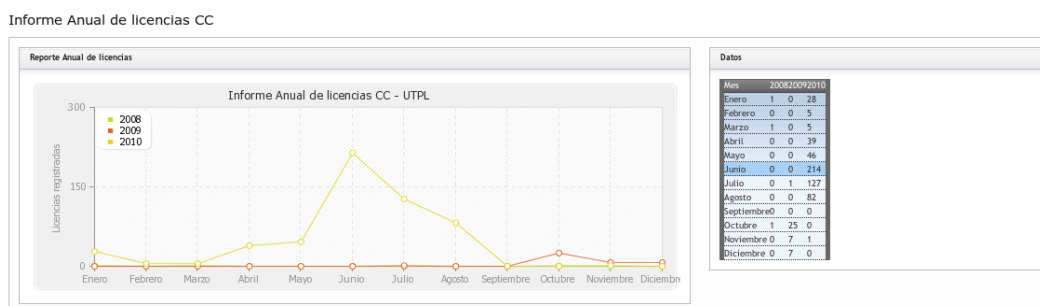


Figura 4.5.: Integración de los datos con la gráfica correspondiente (22).

Una descripción completa de la codificación del módulo de estadística se encuentra en el apéndice B.

4.4. Implementación de la aplicación

Durante el proceso de implementación de la aplicación se probaron varias alternativas; inicialmente se intentó instalar la aplicación en los servidores de la UTPL lo que acarreó varios problemas referidos a la versión del servidor web y a la instalación de las extensiones necesarias para el correcto funcionamiento del portal. Por lo que se optó por alquilar el hosting y el dominio en un servidor externo, en el que se instaló la aplicación sin problema alguno.

La dirección URL del portal es la siguiente <http://www.utplbdcc.com/> allí encontrará los accesos correspondientes a las aplicaciones de Estadística y de Licenciamiento gracias a sendos enlaces descriptivos. Si se desea acceder directamente a alguna de estas aplicaciones las direcciones son: para la aplicación de Estadística <http://www.utplbdcc.com/EstadisticaUTPL/> y para la de Licenciamiento <http://www.utplbdcc.com/LicenciasUTPL/>.

4. Desarrollo e Implementación de la aplicación

El código de la aplicación desarrollada puede encontrarse en el CD que acompaña a esta tesis. Lo que a continuación se describe son las partes críticas del proceso de instalación.

4.4.1. Configuración de la base de datos

Durante el proceso de configuración de la base de datos se realiza la creación de la base de datos de la aplicación: BDD de Estadísticas Creative Commons.

Para ello se utilizan el script (incluido en el CD de instalación y detallado en el manual del programador): *BDD StatisticsCC* para la base de datos de estadísticas.

Además, se necesita introducir en las base de datos información preliminar que permita el correcto funcionamiento de la aplicación:

- Tipos de formatos Creative Commons de las obras licenciadas.
- Tipos de licencias Creative Commons.
- Semillas utilizadas por el crawler.
- Páginas que contienen obras con licencias CC encontradas durante la fase de pruebas.

Para la configuración de las bases de datos se utilizan el script: *StatisticsCC data*, con los datos iniciales de las tablas *formatoObra*, *tipoLicenciaCC*, *urlSeed* y *urlUtpl*.

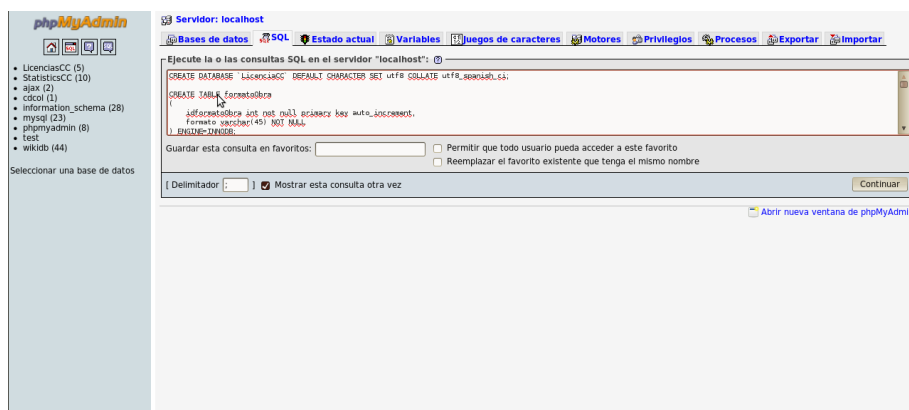


Figura 4.6.: phpMyAdmin pantalla principal (22).

Finalmente, se recomienda que para la configuración de la base de datos MySQL se utilice un front-end amigable para el usuario como phpMyAdmin.

4.4.2. Instalación de la aplicación

Para la correcta instalación de la aplicación se necesita tener, como requisitos, los siguientes programas: Apache (servidor web) con el lenguaje PHP habilitado (esto incluye las librerías GD y XML); y, MySQL (base de datos). El proceso completo de instalación y configuración de estos programas se detalla en el apéndice de esta tesis.

4.4.3. Implementación de la interfaz de usuario

Durante los procesos de diseño e implementación de la interfaz gráfica de usuario se ha seguido la política de mantener simple, sencilla, amigable e intuitiva la interacción entre el usuario y el sistema. Por este motivo, se ha buscado que toda la información posible esté al alcance del usuario en todo momento y que el aspecto visual de la aplicación web resulte agradable.

El usuario común dispondrá de una interfaz desde la cual podrá arrancar el crawler y acceder mediante el escogitamiento de la pestaña indicada a las diversas estadísticas disponibles sobre las licencias Creative Commons en el Ecuador. Adicionalmente, el usuario puede acceder al formulario de registro de obras y obtención de licencias Creative Commons Ecuador, la misma que se generará de acuerdo a las opciones escogidas por el usuario final.

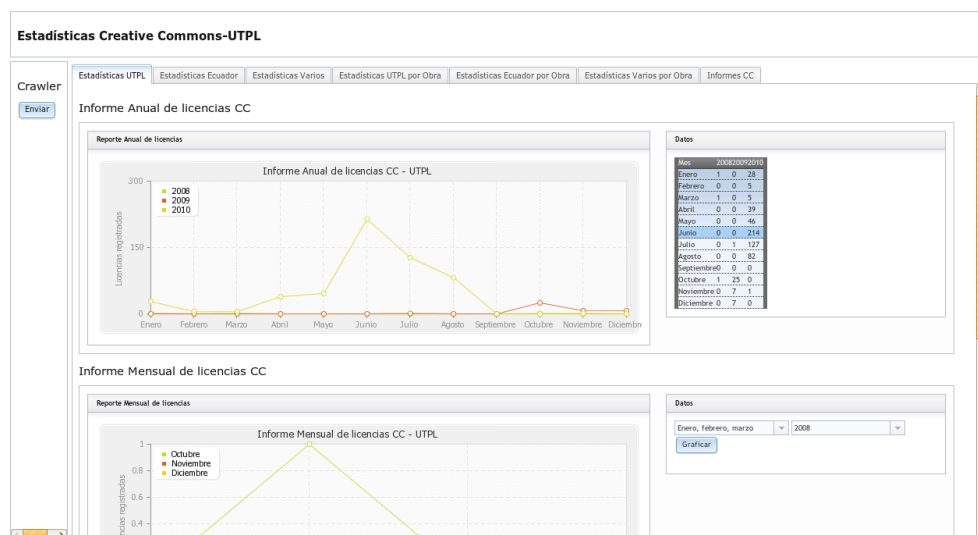


Figura 4.7.: Interfaz gráfica de usuario (GUI) de la aplicación (22).

4. Desarrollo e Implementación de la aplicación



The image shows a web interface for selecting a Creative Commons license in Ecuador. The header features the 'Creative Commons Ecuador' logo on a blue gradient background. Below the header, the main section is titled 'Ponga una licencia a su obra'. It contains two paragraphs of explanatory text about Creative Commons licenses. The first paragraph explains that using a license allows others to share the work while the creator retains their rights. The second paragraph states that selecting a license provides an HTML code for attribution on the user's website. There are three sections of form elements: 1. '¿Quiere permitir usos comerciales de su obra?' with radio buttons for 'Si' (selected) and 'No'. 2. '¿Quiere permitir modificaciones de su obra?' with radio buttons for 'Si' (selected), 'Si, mientras se comparta de la misma manera', and 'No'. 3. 'Jurisdicción de su licencia' with a text input field containing 'Ecuador'. Below these is an 'Información Adicional' section with a note about optional fields and a '¿Cuál es el formato de su obra?' section with a dropdown menu showing 'Otro'.

Creative Commons Ecuador

Ponga una licencia a su obra

Con una licencia de [Creative Commons](#), **mantienes tus derechos de autor** pero permites a otras personas [copiar y distribuir su obra](#), siempre y cuando reconozcan la correspondiente autoría – y solamente bajo las condiciones que especifiques aquí. Si las licencias Creative Commons son algo nuevo para ti, hemos preparado una lista de aspectos a tener en cuenta. Si prefieres ofrecer tu obra sin condiciones o quieres certificar que una obra está en el dominio público, escoge una de nuestras herramientas para dominio público.

Quando usted escoge una licencia, le entregamos el código HTML que puede usar para agregar información sobre la licencia en su sitio web e información sobre cómo elegir una licencia en alguno de los sitios de alojamiento gratuito que han incorporado Creative Commons. Esto no constituye un registro y no almacenamos su elección.

¿Quiere permitir usos comerciales de su obra?

☒ Si
☐ No

¿Quiere permitir modificaciones de su obra?

☒ Si
☐ Si, mientras se comparta de la misma manera
☐ No

Jurisdicción de su licencia

Ecuador

Información Adicional

Los campos siguientes son opcionales, pero serán incrustados en el código HTML generado para su licencia. Esto permite que los usuarios sepan cómo darle atribución o dónde ir para obtener más información sobre la obra.

¿Cuál es el formato de su obra?

Otro ▼

Figura 4.8.: Interfaz gráfica de usuario (GUI) de Licenciamiento (22).

CAPÍTULO 5

PRUEBAS Y RESULTADOS

En esta sección se describe el conjunto de pruebas aplicadas a Estadística y Licenciamiento para garantizar su buen funcionamiento, esto permitirá al usuario contar con una aplicación flexible y robusta. El plan de pruebas evalúa la herramienta desde diversos ámbitos; uno de ellos es la interacción con el usuario que permite obtener el nivel de satisfacción con respecto a su uso; otro es la correcta realización de los diversos procesos de la aplicación de manera eficiente y eficaz, y por último, la facilidad de uso que presta la herramienta al usuario.

5.1. Objetivo

El objetivo principal de la elaboración y aplicación del plan de pruebas es:

- Evaluar la confiabilidad, funcionalidad y usabilidad de Estadística y Licenciamiento.

5.2. Audiencia

El personal involucrado en el proyecto es: público general y profesores de la UTPL (que licencian sus obras con CC). Por la naturaleza del proyecto se han realizado

las pruebas con varios usuarios finales tanto de la aplicación de estadísticas como de la de licencias.

5.3. Plan de Pruebas

El plan de pruebas está dirigido a los diferentes tipos de usuarios que interactúan con Estadística y Licenciamiento, estos son: Usuarios de las estadísticas y usuarios de las licencias. Para cada tipo de usuario se tomará una muestra representativa que permita obtener resultados válidos de las diversas pruebas aplicadas. Los perfiles de usuario considerados para la etapa de pruebas son:

Cuadro 5.1.: Tipos de prueba por perfil de usuario

Perfil de usuario	Tipo de Prueba
Usuario final	Funcionalidad
Usuario final	Compatibilidad
Usuario final	Robustez

5.4. Pruebas de funcionalidad y usabilidad

De acuerdo con Pressman [Rog02], las pruebas de caja negra, llamadas también de comportamiento, se encuentran enfocadas en los requisitos funcionales del software y permiten al desarrollador centrarse en la coherencia de las entradas y salidas del sistema sin preocuparse de la estructura interna de la aplicación examinada.

Las pruebas de funcionalidad permiten localizar fallas funcionales en el sistema al identificar situaciones en las que las respuestas que éste presenta a determinadas acciones del usuario no se corresponden con las especificaciones establecidas durante el proceso de diseño del software.

Este proceso simulará varios escenarios para confirmar que todos los resultados satisfacen las expectativas establecidas. Durante esta evaluación, los usuarios prueban Estadísticas y Licencias Creative Commons UTPL y comentan los distintos problemas y dificultades con las que se encuentran al realizar las operaciones solicitadas.

5. Pruebas y resultados

Evaluar la usabilidad de un sitio web puede ser algo verdaderamente útil, ya que permite valorar la interacción con el usuario y obtener el nivel de satisfacción con respecto al uso. Estas pruebas nos permitirán medir, además, la facilidad de uso de Estadísticas y Licencias Creative Commons UTPL.

La evaluación de la funcionalidad y usabilidad consta de dos partes:

- Realizar las distintas operaciones disponibles en Estadísticas y Licencias Creative Commons UTPL.
- Contestar la encuesta sobre la funcionalidad y usabilidad de Estadísticas y Licencias Creative Commons UTPL.

Las operaciones utilizadas en la prueba son:

1. Acceder a una pestaña de contenido de Estadísticas Creative Commons UTPL.
2. Graficar datos de los informes mensuales de Estadísticas Creative Commons UTPL.
3. Graficar datos de los informes semanales de Estadísticas Creative Commons UTPL.
4. Generar dinámicamente los reportes de Estadísticas Creative Commons UTPL.
5. Obtener una licencia Creative Commons Ecuador de acuerdo al tipo de obra.

La encuesta aplicada a los usuarios consta de dos partes bien diferenciadas, la primera evalúa la funcionalidad y la segunda la usabilidad. Dentro de la funcionalidad (8 items) se consideran categorías como: accesibilidad, seguridad y navegabilidad; y, en cuanto a la usabilidad (11 items) se contempla las categorías siguientes: amigabilidad, calidad del entorno, calidad del contenido y potencialidad de los recursos didácticos. Éstas pruebas se realizaron a 20 usuarios potenciales del portal (entre ellos estudiantes de PHD de la UTPL).

A continuación se presenta la descripción de cada una de las categorías anteriormente mencionadas:

5. Pruebas y resultados

Cuadro 5.2.: Descripción de las categorías de la encuesta.

Tipo de Prueba	Categoría	Descripción
Funcionalidad	Accesibilidad	Facilidad de acceso.
	Seguridad	Control y administración de usuarios.
	Navegabilidad	Facilidad para la navegación dentro de Estadísticas Creative Commons UTPL.
Usabilidad	Amigabilidad	Facilidad de uso.
	Calidad del entorno	Uso adecuado de los elementos que conforman la parte estética.
	Calidad del contenido	Pertinencia de la información ingresada en Estadísticas Creative Commons UTPL.

Para la correcta estimación de los items que componen la encuesta se utilizó la siguiente escala de valores:

Cuadro 5.3.: Descripción de las escalas de valores.

Tipo de Prueba	Escala	Descripción
Funcionalidad	Siempre	75 % - 100 %, prueba satisfactoria.
	Casi siempre	50 % - 75 %, prueba aprobada.
	A veces	25 % - 50 %, prueba con errores.
	Nunca	0 % - 25 %, prueba fallida.
Usabilidad	Muy bueno	75 % - 100 %, prueba satisfactoria.
	Bueno	50 % - 75 %, prueba aprobada.
	Regular	25 % - 50 %, prueba con errores.
	Malo	0 % - 25 %, prueba fallida.

En cada una de estas categorías se establecieron variables, las que fueron planteadas como preguntas en la encuesta. Para el apartado de funcionalidad tenemos:

5. Pruebas y resultados

Cuadro 5.4.: Variables de la encuesta por categorías.

Categoría	Variable
Accesibilidad	¿El acceso al portal Estadísticas Creative Commons UTPL. se realizó en un tiempo razonable?
	¿Es aceptable el tiempo de respuesta al realizar la petición de un gráfico al portal?
	¿Es aceptable el tiempo de respuesta al realizar la petición de un reporte al portal?
	¿Es aceptable el tiempo de respuesta al realizar la petición de una licencia al portal?
Seguridad	¿Se permite al usuario acceder a los datos y modificarlos?
	¿Se puede conocer quién y cuándo ha actualizado la información del portal de Licencias Creative Commons UTPL?
Navegabilidad	¿Es fácil la navegación en el portal?
	¿En todo momento se sabe en qué parte de la aplicación nos encontramos?
Amigabilidad	¿Es amigable la interfaz del portal?
	¿Se puede obtener la información fácilmente?
	¿Permite obtener reportes fácilmente?
Calidad del entorno	El tipo de letra utilizado es. . .
	El tamaño de la fuente utilizada es. . .
	El uso de imágenes e iconos es. . .
	La calidad técnica de los enlaces es. . .
	La estética de los tabs es. . .
	El estilo del lenguaje utilizado es. . .
Calidad del contenido	La estructura de la información es. . .
	La estructura del contenido del portal es. . .

5. Pruebas y resultados

Los resultados obtenidos al realizar la prueba de forma concurrente, después de la tabulación y análisis respectivo, fueron los siguientes:

En cuanto a la funcionalidad:

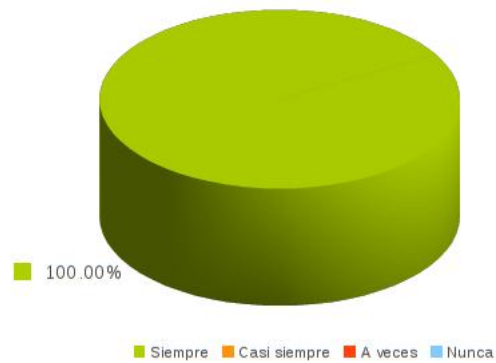


Figura 5.1.: ¿El acceso al portal Estadísticas Creative Commons UTPL se realizó en un tiempo razonable? (22).

El acceso concurrente al portal de Estadísticas Creative Commons UTPL se realizó en un tiempo razonable el 100 % de las veces en las que fue requerido, la prueba se realizó con la ayuda de los usuarios quienes utilizaron el portal al mismo tiempo. Se concluye que se completó esta parte de la prueba sin errores.

Se considera como “tiempo de acceso al portal” al tiempo de carga en el navegador de la página principal del portal web.

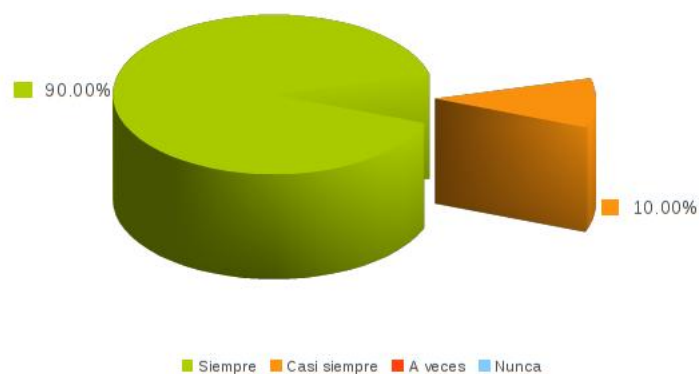


Figura 5.2.: ¿Es aceptable el tiempo de respuesta al realizar la petición de un gráfico al portal? (22).

5. Pruebas y resultados

El tiempo de respuesta, al realizar la petición de generación de una gráfica al portal, fue aceptable para el usuario el 90 % de las veces, sólo en el 10 % de los casos se presentaron dificultades menores. Se considera como “tiempo de respuesta al realizar la petición de generación de una gráfica” al tiempo que se tarda el navegador en presentar al usuario la gráfica generada dinámicamente por la aplicación web. Se concluye que se completó esta parte de la prueba sin errores.

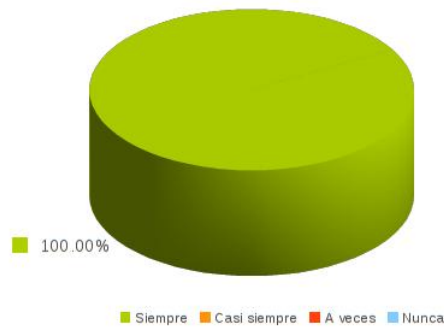


Figura 5.3.: ¿Es aceptable el tiempo de respuesta al realizar la petición de un reporte al portal?(22).

El tiempo de respuesta, al realizar la petición de un reporte al portal, fue aceptable para el usuario el 100 % de las veces en las que fue requerido. Se considera como “tiempo de respuesta al realizar la petición de un reporte” al tiempo que se tarda el navegador en presentar al usuario el reporte generado dinámicamente (consultando la base de datos) por la aplicación web. Se concluye que se completó esta parte de la prueba sin errores.

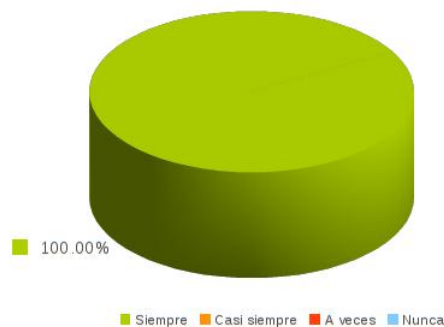


Figura 5.4.: ¿Es aceptable el tiempo de respuesta al realizar la petición de una licencia al portal? (22).

5. Pruebas y resultados

El tiempo de respuesta, al realizar la petición de una licencia al portal, fue aceptable para el usuario el 100 % de las veces en las que fue requerido. Se concluye que se completó esta parte de la prueba sin errores.

En el ámbito de la seguridad se comprobó si el usuario final tiene acceso a los datos o si puede modificarlos, el resultado de las pruebas realizadas fue concluyente, en el 100 % de las casos ningún usuario tuvo acceso a la información del portal, ni pudo modificarla.

Siguiendo en el ámbito de la seguridad se comprobó si el usuario final puede conocer quién y cuándo modificó la información del portal de Licencias Creative Commons UTPL, de acuerdo a las pruebas realizadas en el 100 % de los casos ningún usuario tuvo acceso a esta información dentro del portal.

Se considera como “tiempo de respuesta al realizar la petición de una licencia” al tiempo que se tarda el navegador en presentar al usuario la página con el tipo de licencia requerido generada dinámicamente (realizando inserciones y consultas en la base de datos) por la aplicación web.

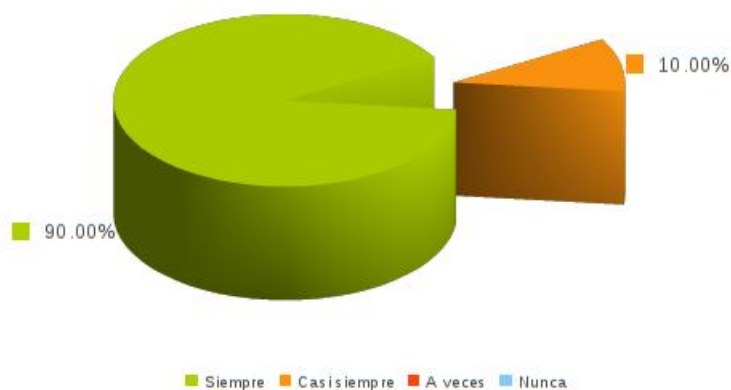


Figura 5.5.: ¿Es fácil la navegación en el portal? (22).

A la pregunta ¿es fácil la navegación en el portal? el 90 % de los usuarios respondieron siempre, sólo en el 10 % de los casos se presentaron dificultades menores. Se concluye que se completó esta parte de la prueba sin errores.

5. Pruebas y resultados

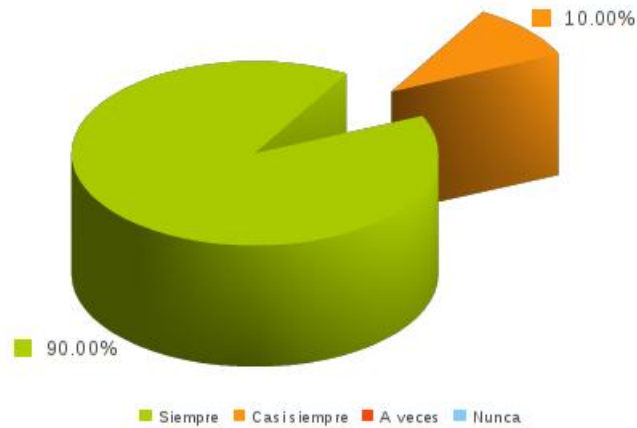


Figura 5.6.: ¿En todo momento se sabe en qué parte de la aplicación nos encontramos? (22).

A la pregunta ¿es todo momento se sabe en qué parte de la aplicación nos encontramos? el 90 % de los usuarios respondieron siempre, sólo en el 10 % de los casos se presentaron dificultades menores. Se concluye que se completó esta parte de la prueba sin errores.

En cuanto a la usabilidad:

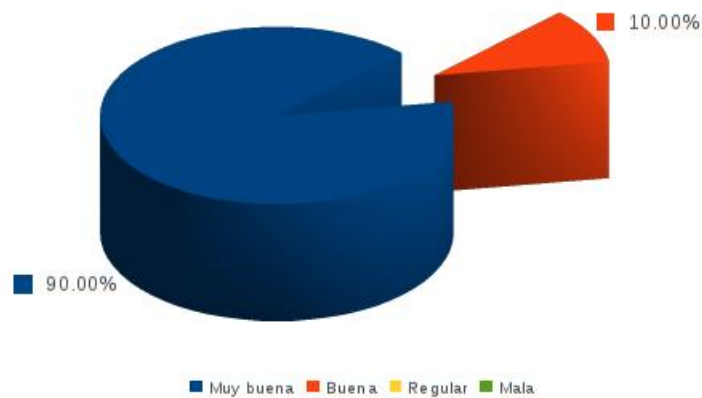


Figura 5.7.: ¿Es amigable la interfaz del portal? (22).

A la pregunta ¿es amigable la interfaz del portal? el 90 % de los usuarios la calificaron como muy buena, sólo en el 10 % de los casos se presentaron dificultades menores. Se concluye que se completó esta parte de la prueba sin errores.

5. Pruebas y resultados

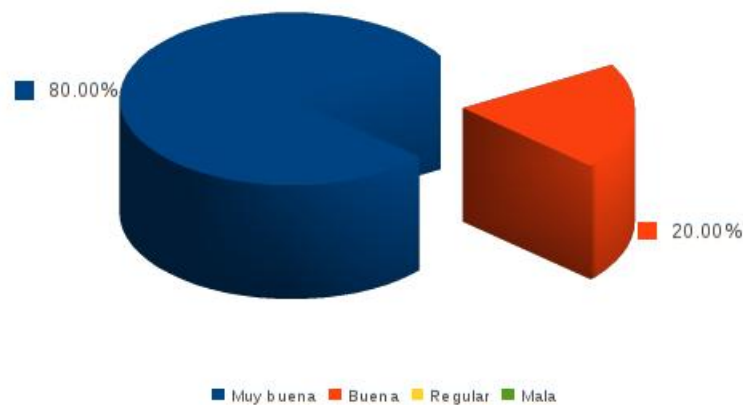


Figura 5.8.: ¿Se puede obtener la información fácilmente? (22).

A la pregunta ¿se puede obtener la información fácilmente? el 80 % de los usuarios la calificaron como muy buena, sólo en el 20 % de los casos se presentaron dificultades menores. Se concluye que se completó esta parte de la prueba sin errores.

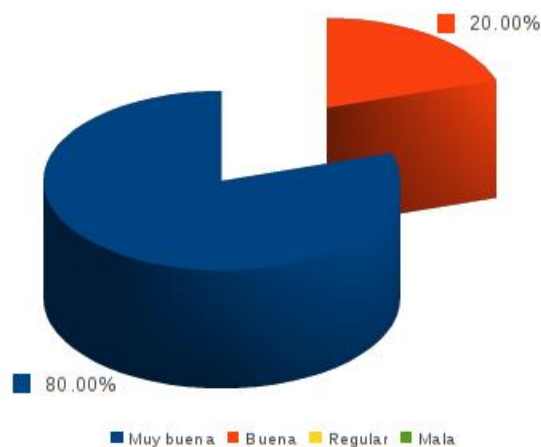


Figura 5.9.: ¿Permite obtener reportes fácilmente? (22).

A la pregunta ¿el portal permite obtener reportes fácilmente? el 80 % de los usuarios la calificaron como muy buena, sólo en el 20 % de los casos se presentaron dificultades menores. Se concluye que se completó esta parte de la prueba sin errores.

5. Pruebas y resultados

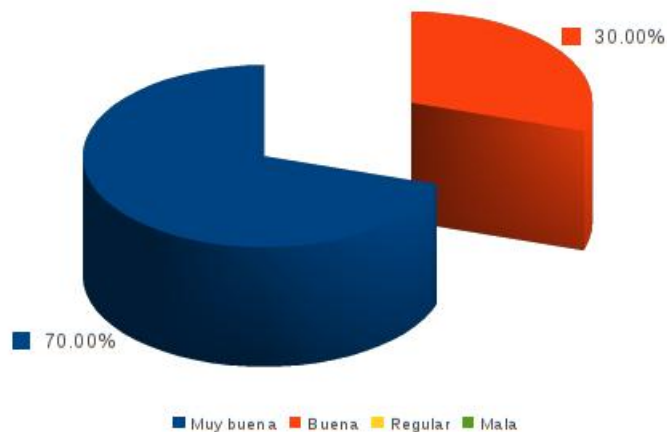


Figura 5.10.: La estructura de la información es (22).

Cuando se preguntó sobre la estructura de la información en el portal, el 80 % de los usuarios respondieron que era muy buena, sólo el 20 % de los casos la calificó de buena. Se concluye que se completó esta parte de la prueba sin errores.

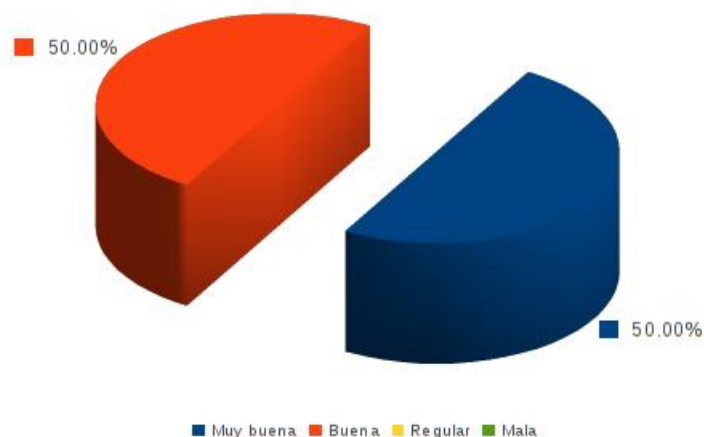


Figura 5.11.: La estructura del contenido del portal es (22).

Cuando se preguntó sobre la estructura del contenido del portal, el 50 % de los usuarios respondieron que era muy buena, el otro 50 % la calificó de buena. Se concluye que se completó esta parte de la prueba sin errores.

Como conclusión de las pruebas de usabilidad y funcionalidad, se puede afirmar que el uso de Estadísticas y Licencias Creative Commons UTPL ha tenido muy

buena acogida, y que en cuanto al diseño las decisiones tomadas fueron las correctas.

5.5. Pruebas de compatibilidad

Estas pruebas se realizan con el fin de comprobar la compatibilidad del sistema con los navegadores web más utilizados. Para que la aplicación sea considerada como compatible con un navegador, el diseño de su interfaz gráfica debe permanecer constante, sin sufrir grandes alteraciones o cualquier tipo de cambio que afecte o disminuya su funcionalidad. Asimismo el usuario debe poder realizar las operaciones que ofrece el sistema de manera fluida, sin la presencia de mensajes de error por parte del navegador. A continuación se presenta una tabla con los resultados de las pruebas de compatibilidad aplicadas con los navegadores web más utilizados en la actualidad y considerando los sistemas operativos Windows (XP y 7) y Linux (10.04, 10.10 y 11.04):

Cuadro 5.5.: Compatibilidad con navegadores web.

Navegador	Versión	Compatible
Mozilla Firefox	3.0.3	Si
Mozilla Firefox	3.6	Si
Internet Explorer	8	Si
Google Chrome	2	Si

Cabe mencionar que el sistema no fue probado bajo la plataforma Macintosh debido a que durante la etapa de pruebas no se tuvo acceso a ningún equipo que contara con esta plataforma. Por otra parte, todas las pruebas en la plataforma PC resultaron exitosas, debido a que el sistema no presentó alteración alguna en su interfaz o funcionamiento con ninguno de los navegadores probados. De esta forma, se puede concluir que el sistema es compatible con los navegadores Internet Explorer (versión 8.0.x y 9.0.x), Mozilla Firefox (versión 3.x y 4.x) y Google Chrome (versión 10.x y 11.x).

5. Pruebas y resultados



Figura 5.12.: Vista de la aplicación en el navegador Mozilla Firefox (22).



Figura 5.13.: Vista de la aplicación en el navegador Google Chrome (22).

5. Pruebas y resultados

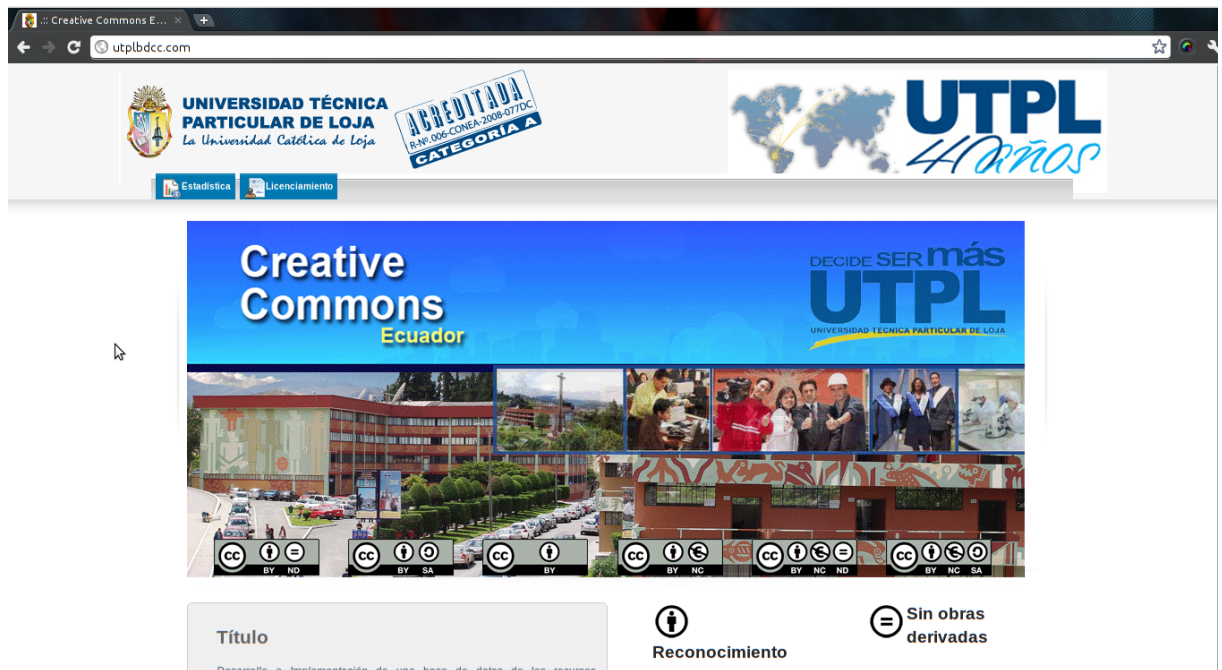


Figura 5.14.: Vista de la aplicación en el navegador Internet Explorer (22).

5.6. Pruebas de robustez

Con el objetivo de comprobar la capacidad del sistema para soportar múltiples accesos concurrentes sin sufrir una disminución considerable en el rendimiento, se han realizado pruebas de stress con la ayuda de la herramienta Apache JMeter (<http://jakarta.apache.org/jmeter>). Este software está diseñado para realizar pruebas de carga sobre un sistema mediante la simulación de múltiples hilos o usuarios.

En la herramienta JMeter, para simular la interacción del usuario con el sistema, es necesario programar cada operación que se desea efectuar durante la prueba, indicando la ruta en el servidor para acceder al recurso, los parámetros que deben ser enviados, el tipo de método que se utiliza para realizar la petición (GET o POST) y la respuesta que se espera del sistema.

Se debe destacar que cada prueba con la simulación de un determinado número de usuarios se ha repetido 10 veces para mejorar el número de muestras e incrementar la validez de los resultados obtenidos.

A continuación se presentan una tabla y una gráfica con los resultados de las prue-

5. Pruebas y resultados

bas realizadas para el crawler con una cantidad grande de semillas:

Cuadro 5.6.: Tiempo promedio del crawler, pruebas exhaustivas.

Usuarios	Semillas utilizadas	Tiempo promedio
1	10	17753 ms
5	10	29588 ms
10	10	50176 ms
15	10	80764 ms
20	10	118352 ms



Figura 5.15.: Gráfica del tiempo promedio del crawler, pruebas exhaustivas (22).

A continuación se muestra una tabla y una gráfica con los resultados de las pruebas realizadas para el crawler con pocas semillas:

Cuadro 5.7.: Tiempo promedio del crawler, pruebas leves.

Usuarios	Semillas utilizadas	Tiempo promedio
1	4	7250 ms
5	4	10083 ms
10	4	18166 ms
15	4	29249 ms
20	4	37332 ms

5. Pruebas y resultados

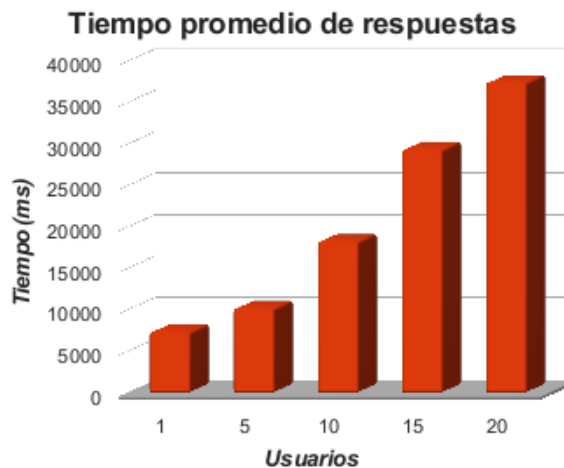


Figura 5.16.: Gráfica del tiempo promedio del crawler, pruebas leves (22).

Se puede concluir que el deterioro en los tiempos de respuesta aumenta con el número de semillas utilizadas por el crawler durante el proceso de rastreo.

5.7. Validación del portal

Interfaz de usuario del W3C Markup Validation Service. El encabezado muestra el logo W3C y el título "Markup Validation Service" con el subtítulo "Check the markup (HTML, XHTML) of Web documents". Hay tres pestañas: "Validate by URI" (seleccionada), "Validate by File Upload" y "Validate by Direct Input". La sección "Validate by URI" contiene el texto "Validate a document online:", un campo de entrada etiquetado "Address:" y un botón "Check". Hay un enlace "More Options" debajo del campo de entrada.

Figura 5.17.: Validador W3C de documentos web (22).

5. Pruebas y resultados

La validación es el proceso que asegura que una página web (escrita utilizando el lenguaje HTML) cumple con las normas y restricciones impuestas por el lenguaje HTML. La realización de este proceso con sus resultados respectivos no tiene impacto alguno en la funcionalidad, usabilidad, compatibilidad o robustez del portal desarrollado, pues existen muchas páginas web que no han pasado la prueba de validación y, a pesar de ello funcionan correctamente en varios navegadores y con tiempos de respuesta razonables; un ejemplo de ello es la página principal de Google.

Para este proceso se utilizó el validador del W3C que permite tres formas de validación directa, la primera mediante el ingreso del URI (una vez que la página se encuentre en la web), la segunda cargando el archivo html que se desea validar y, la tercera insertando en un campo de texto el código HTML que se quiere validar.

W3C[®] Markup Validation Service
Check the markup (HTML, XHTML, ...) of Web documents

Jump To: [Validation Output](#)

Errors found while checking this document as HTML 4.01 Transitional!

Result:	5 Errors, 6 warning(s)
Address:	<input type="text" value="http://utplbdcc.com/"/>
Encoding:	utf-8 (detect automatically)
Doctype:	HTML 4.01 Transitional (detect automatically)
Root Element:	HTML

The W3C validators are hosted on server technology donated by HP, and supported by community donations. [Donate](#) and help us build better tools for a better web.

Options

☐ Show Source ☐ Show Outline ☒ List Messages Sequentially ☐ Group Error Messages by Type

☐ Validate error pages ☐ Verbose Output ☐ Clean up Markup with HTML-Tidy

[Help](#) on the options is available. [Revalidate](#)

Figura 5.18.: Validación HTML del portal web (22).

La gráfica anterior muestra la validación HTML realizada mediante la herramienta proporcionada por la W3C. Este resultado se debe a que al verificar la estructura del HTML se encuentran varias inconsistencias con el estándar; ya que la utilización del framework Dojo permite interoperabilidad en los diversos navegadores existentes sacrificando los estándares de cada uno de estos lenguajes.

5. Pruebas y resultados

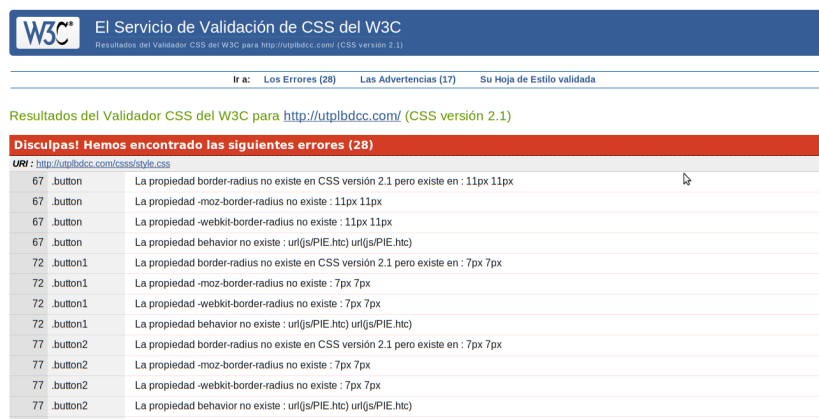


Figura 5.19.: Validación CSS del portal web (22).

Los resultados obtenidos se explican debido a que, la utilización del framework open source Dojo para javascript facilita la interoperabilidad de la aplicación en los diversos navegadores existentes, pero para ello debe incluir diversas propiedades y métodos JS que son válidos solamente en un navegador determinado; esto sumado a que la versión de CSS implementada (2.1) es diferente de la utilizada por el validador (2) da como resultado los mensajes mostrados en el gráfico anterior.



Figura 5.20.: Validación accesibilidad del portal web (22).

La gráfica anterior muestra la calificación C obtenida al realizar la validación de la accesibilidad en el portal web. Este resultado se debe a que se verifica tanto la estructura del HTML como la de CSS para asignar la calificación, de nuevo, la utilización del framework Dojo permite interoperabilidad en los diversos navegadores existentes sacrificando los estándares de cada uno de estos lenguajes.

DISCUSIÓN

El desarrollo e implementación de una base de datos de los recursos licenciados con Creative Commons en Ecuador, requirió la realización de dos aplicaciones web diferentes pero a la vez complementarias, la de licenciamiento -cuyo objetivo es el de otorgar una licencia creative commons de acuerdo al tipo de obra y al tipo de licencia- y la de estadísticas de las licencias CC -cuyo fin es el de realizar un compendio de todas las licencias con jurisdicción Ecuador y presentar estos datos en forma de gráficas e informes que faciliten su lectura y comprensión-.

La creación del sitio web de licenciamiento requirió el estudio del sistema de licencias utilizado por Creative Commons; se presenta al usuario un entorno fácil de utilizar y que, además, le permite escoger fácilmente el tipo de obra que desea licenciar, los permisos que desea otorgar a terceros, todo esto considerando siempre como jurisdicción Ecuador. Se puso especial atención a los tipos de licencias reconocidos por la legislación ecuatoriana y a los tipos de obras que pueden ser licenciados.

Para la aplicación de estadísticas de las licencias Creative Commons del Ecuador se consideraron varios factores entre ellos: la construcción de un crawler -cuyo único objetivo es encontrar los recursos ecuatorianos licenciados mediante creative commons-; la implementación de un módulo de generación de gráficas -con el fin de generar las gráficas de barras verticales-; la fabricación de un módulo de generación de informes; y, el establecimiento de un módulo de generación de archivos RDF.

Fue necesaria la **construcción de un crawler** ya que no existía otra forma de recolectar la información referente a los recursos ecuatorianos existentes con licencias creative commons (tanto si la jurisdicción era Ecuador como si no lo fuera) que recorrer toda la internet (o gran parte de ella) con una herramienta de búsqueda automática. Para su implementación se han considerando las siguientes etapas: Inicialización de las semillas url utilizadas por el crawler; selección de las semillas; obtención de la página web correspondiente; análisis del contenido de las páginas web; extracción de los enlaces de la página web; extracción de la información pertinente (licencias CC Ecuador) de la página web, y; adición de los enlaces encontrados a la base de datos.

Este elemento (web crawler) es una de las partes críticas de la aplicación, sin él no sería posible ni la búsqueda de recursos Creative Commons ni la indización de parte del contenido de la web y consecuentemente, no se podrían realizar los resúmenes gráficos ni los informes pormenorizados que necesitan de esta información.

Una vez obtenidos los enlaces, se procede a la clasificación de los recursos y a la obtención de las estadísticas de los recursos encontrados.

Para la clasificación de las páginas encontradas se consideran dos grandes grupos: los recursos licenciados por la UTPL y aquellos licenciados por otras instituciones o personas naturales, manteniendo siempre el criterio de que las licencias pertenezcan a la jurisdicción de Ecuador o en su defecto, que las páginas indizadas tengan en su dominio el sufijo ec. Esta clasificación se puede observar claramente en los diagramas de diseño de la base de datos de datos de la sección 3.7. También se realiza una consolidación de los recursos a nivel de país, esto con el fin de tener datos consolidados sobre los recursos CC licenciados en Ecuador.

El **módulo de generación de gráficos** se encarga además del cálculo de las estadísticas, para ello se realiza un conteo de las licencias obtenidas -clasificadas por tipo de obra y de licencia-, y se generan dinámicamente archivos de resumen anual, mensual y semanal (almacenados dinámicamente como CSV y RDF) que servirán de materia prima en la construcción tanto de los gráficos como de las tablas de datos desplegadas al usuario.

La elaboración de las gráficos de barras verticales exige la utilización de la librería GD del lenguaje de programación PHP, ésta permite la generación dinámica de archivos gráficos (png o jpg) que luego serán desplegadas en pantalla mediante el navegador web. También fue necesaria la utilización de la librería opensource

javascript Dojo para el diseño e implementación de la interfaz gráfica de usuario de la aplicación con el fin de cumplir con los criterios de accesibilidad y usabilidad que una aplicación web debe tener.

La fabricación de un **módulo de generación de informes** se realizó con el fin de tener listas de resumen de las obras debidamente clasificadas por su tipo y por el tipo de licencia que ostentan, ésto se pone de manifiesto en el interfaz gráfico de usuario de la aplicación ya que esta característica cuenta con su propia pestaña de visualización y con varios elementos gráficos que permiten la generación dinámica de los resúmenes de acuerdo a los requerimientos del usuario.

Además, se estableció un **módulo de generación de archivos RDF** para facilitar la reutilización tanto de los datos como de los resultados de las estadísticas obtenidas. En su creación se utilizó la librería XMLWriter que facilitó los elementos necesarios para la correcta estructuración de este tipo de archivos, además de utilizar varias librerías de metadatos existentes como la DublinCore (DC) para la descripción de la información manejada por la aplicación. El modelo completo de implementación de las estructuras RDF se detalla en la sección 3.8 mediante una descripción completa tanto de la estructura como de la implementación en XML.

RDF es una implementación de metadatos que utiliza XML con el fin de proporcionar un marco estándar para la interoperabilidad en la descripción de los contenidos web; no es más que la infraestructura que permite esa restricción gracias a la codificación, reutilización e intercambio de metadatos estructurados. Con estas prerrogativas interoperabilidad y estructuración, RDF es el mejor modelo para asociar información sobre el contenido de las páginas web con licencia Creative Commons, y además, permite optimizar la búsqueda y recuperación de la información de la aplicación en la web.

RDF es también el mejor esquema para la interoperabilidad entre formatos ya que utiliza metadatos estándar facilitando que la metainformación sea legible por el ordenador. Además, permite el uso coherente de los metadatos así como la aplicación de esquemas jerárquicos lo que preserva el conocimiento electrónico distribuido mediante la internet.

Entre las **dificultades encontradas** durante el desarrollo de esta investigación, he considerado como las más importantes las siguientes: existe poca información con respecto al desarrollo del crawler lo que extendió los tiempos de consulta de la información (realización del estado del arte del crawler) y la implementación

de este componente; durante la implementación, la falta de comprensión del administrador web de los requisitos mínimos que exige la aplicación acarreó un importante despilfarro de tiempo y esfuerzo, ya que éste (el administrador) se limitó a proporcionar los “errores” de la aplicación sin mencionar que ninguno de los servidores disponibles tenía instaladas las versiones necesarias de las librerías GD y XMLWriter establecidas en los requisitos de instalación de la aplicación.

La aplicación desarrollada permite llevar un control estadístico de las obras licenciadas con Creative Commons desarrolladas por la UTPL, realizar análisis comparativos de la producción de estos recursos entre diversos períodos de tiempo (años, trimestres, meses) y obtener la información necesaria para desarrollar estrategias o políticas de desarrollo de recursos en general.

Es necesario añadir que como **trabajos futuros** se puede considerar la realización de detalles exhaustivos de los recursos licenciados por la UTPL clasificando la información de acuerdo al CITTE que patrocinó una publicación o de acuerdo al autor; también es posible realizar una indexación más detallada de la información encontrada utilizando para ello la institución que patrocina una obra licenciada; todo esto siempre dentro de la jurisdicción del Ecuador.

En cuanto al crawler varias características deberían implementarse para aumentar su utilidad, como: la capacidad de realizar búsquedas y de relacionar las páginas web encontradas mediante conceptos y significados; y, la utilización de métodos de interpretación del lenguaje natural y tesauros de sinónimos en los procesos de indexación, extracción y búsqueda del contenido dentro de cada página encontrada.

Con este pequeño análisis se demuestra plausiblemente que los objetivos planteados al inicio de este trabajo de investigación se cumplieron de forma satisfactoria.

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

- La eficacia del crawler depende en gran medida de la calidad de las semillas escogidas para la inicialización de la aplicación.
- El tiempo de ejecución del crawler depende tanto de la implementación realizada como de la cantidad de semillas analizadas.
- La recuperación de la información es un elemento esencial ya que permite compartir la información generada en diversas plataformas web de la UTPL.
- Los ficheros CSV y RDF (XML) generados facilitan tanto la generación de las gráficas como la portabilidad de los resultados de las estadísticas obtenidas por la aplicación hacia otros entornos colaborativos.
- La utilización de RDF ayudó a cumplir con los objetivos planteados, en cuanto a la reutilización de la información generada por la aplicación, y permitirá la utilización y recuperación automática de los datos generados.
- La librería XMLWriter de PHP facilita enormemente la generación de archivos XML, en el presente trabajo se la adaptó para generar los archivos RDF necesarios para la estandarización de los resultados.

- La aplicación desarrollada permite conocer la cantidad de recursos Creative Commons desarrollados por la UTPL, así como determinar su evolución a través del tiempo indicando aquellos períodos de mayor o menor producción.
- La utilización del portal web permite conocer la cantidad de recursos Creative Commons desarrollados por otras instituciones, así como determinar su evolución a través del tiempo indicando aquellos períodos de mayor o menor producción.
- Mediante esta aplicación se puede conocer la cantidad de recursos Creative Commons desarrollados en el Ecuador, así como determinar su evolución a través del tiempo indicando aquellos períodos de mayor o menor producción.
- Este software permite realizar un análisis comparativo entre la cantidad de recursos Creative Commons UTPL y de otras instituciones, en un período de tiempo determinado, señalando las fortalezas y debilidades de la universidad.

Recomendaciones

- Existe mucha redundancia de contenido en la Internet, es decir, existen páginas web con contenidos iguales o similares y diferentes URIs. Esta característica es perjudicial para la búsqueda porque tiende a encontrar resultados relevantes pero repetidos. Por esto, es recomendable considerar a futuro identificadores únicos para cada sitio o página web que evite la redundancia de datos.
- La aplicación de búsqueda desarrollada puede ser fácilmente integrada en sistemas de manejo de contenido, donde este agente explorador puede obtener nuevos recursos automáticamente, para que los usuarios los revisen y los indexen en una base de datos interna.
- El crawler desarrollado es flexible y puede fácilmente actualizarse y adecuarse a las necesidades del usuario. Puede utilizarse en aplicaciones como DSpace o similares para indexar automáticamente los documentos contenidos en este CMS.

Conclusiones y recomendaciones

- Promover entre la comunidad estudiantil de la UTPL el uso y desarrollo de herramientas alternativas para la exploración y búsqueda de documentos en la [www](#).

TRABAJOS FUTUROS

En esta sección se pretende esbozar algunas características adicionales para este software, encontradas durante el proceso de desarrollo del mismo, que serían de mucha utilidad al implementarse.

- Clasificación de las licencias CC Ecuador especificando las universidades y/o centros de investigación patrocinadores (no solamente la UTPL).
- Determinación de estándares adecuados para una universidad con énfasis en la investigación científica y la publicación de esos resultados.
- Determinación de la cantidad de publicaciones realizadas por los integrantes de los CITTES de la UTPL.
- Determinación de la cantidad de publicaciones realizadas por los estudiantes (o graduados) de PHD de la UTPL.

En cuanto al crawler desarrollado para este trabajo de investigación, algunas características adicionales serían:

- Capacidad de buscar y relacionar páginas web por conceptos y significados, utilizando para ello tecnología semántica.
- Mejorar los procesos de indexación, extracción y búsqueda del contenido utilizando métodos de interpretación del lenguaje natural y tesauros que incorporen sinónimos.

Apéndices

APÉNDICE A

CASOS DE USO

A.1. Diagramas de los casos de uso

El sitio web de estadísticas tiene sólo cuatro casos de uso, por lo que los cuatro son arquitectónicamente significativos. La figura A.1 muestra el caso de uso de alto nivel de la aplicación.

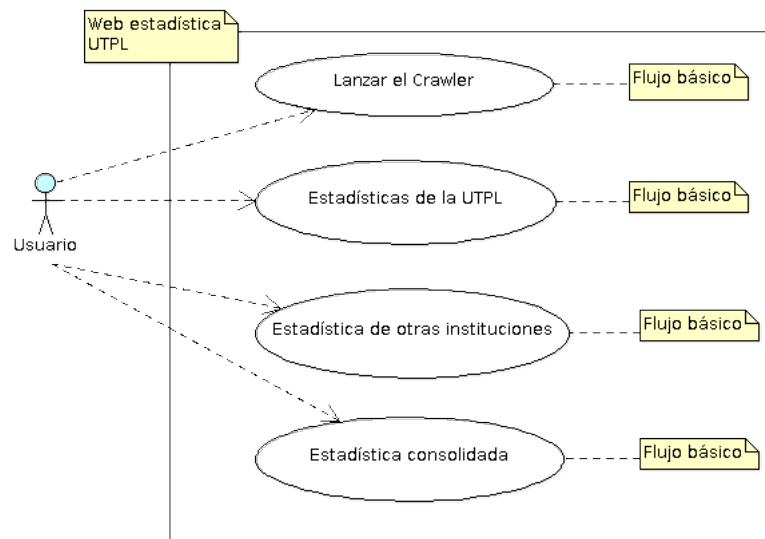


Figura A.1.: Diagrama de casos de uso de la aplicación (22).

A. Casos de uso

Esta aplicación cuenta con un solo actor: el usuario. El usuario utiliza la internet de forma anónima y navegará por las estadísticas de las licencias CC visualizándolas.

El software de estadística requiere la implementación de un crawler cuyas funciones son: exploración automática de páginas web, clasificación de documentos e, indexación de documentos.

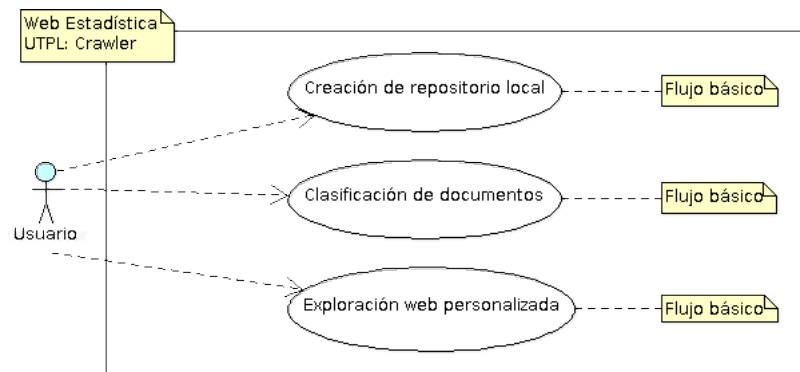


Figura A.2.: Diagrama de casos de uso del Crawler (22).

El usuario es el único autor de estos casos de uso. El usuario utiliza de forma transparente el crawler, es decir, lo inicializa pero no puede observar de forma directa su funcionamiento. Los resultados de la ejecución del crawler solamente son visibles a nivel de la capa de datos.

A.2. Casos de uso de alto nivel

A.2.1. Caso de Uso: Lanzar el crawler

Propósito: Explorar la web en busca de recursos Creative Commons y obtener los resultados.

Descripción: Este caso de uso especifica la puesta en marcha del crawler por parte del usuario. El usuario accede a la página de la aplicación y pulsa el botón de lanzamiento del crawler. Como resultado obtiene una lista de enlaces a las páginas que contienen recursos Creative Commons.

Flujo Básico:

A. Casos de uso

1. El caso de uso comienza cuando el usuario común accede a la página de la aplicación.
2. Se muestra un formulario donde el usuario puede presionar el botón “Enviar” del apartado crawler para poner en marcha la búsqueda de recursos Creative Commons en la web.
3. Se ingresa en la base de datos un listado de enlaces a las páginas web que contienen recursos Creative Commons como resultado de la búsqueda del crawler.

Flujo Alternativo:

- *En el punto 3* Si no se ha establecido la conexión con la BDD se genera un error.

A.2.2. Caso de Uso: Estadísticas de la UTPL

Propósito: Generar las gráficas estadísticas (por año, mes y semana) que resuma los recursos Creative Commons de la UTPL.

Descripción: Este caso de uso especifica la generación de las gráficas de la aplicación de acuerdo a los parámetros establecidos (año, mes, día, tipo de recurso). Para ello el usuario debe presionar el botón del formulario correspondiente al período del que se requiere la información, la aplicación recolecta los datos y los presenta en forma de gráfica (con su respectiva tabla de datos) para facilitar la comprensión de la información por parte del usuario.

Flujo Básico:

1. El caso de uso comienza cuando el usuario común accede a la página de la aplicación.
2. Se muestra el formulario con diferentes pestañas de acuerdo al ámbito de despliegue de la información (Estadísticas de la UTPL, del Ecuador o de otras fuentes).
3. El usuario selecciona la pestaña correspondiente a la información que busca.
4. La aplicación muestra tres secciones (cada una de ellas contiene gráfica y datos) que corresponden al año, mes y semana.

A. Casos de uso

5. El usuario selecciona el período de tiempo correspondiente (trimestre y año o mes y año) dependiendo de la sección y presiona el botón graficar.
6. La aplicación recolecta los datos de la BDD correspondiente al período ingresado, los clasifica, genera la gráfica y los archivos de respaldo (CSV y RDF) y, finalmente, presenta la gráfica al usuario.

Flujo Alternativo:

- *En el punto 5* Si el usuario no selecciona alguno de los parámetros necesarios para la consulta, se toman los valores por defecto de los campos de la aplicación.

A.2.3. Caso de Uso: Estadísticas de otras instituciones

Propósito: Generar las gráficas estadísticas (por año, mes y semana) que resuma los recursos Creative Commons de otras instituciones.

Descripción: Este caso de uso especifica la generación de las gráficas de la aplicación de acuerdo a los parámetros establecidos (año, mes, día, tipo de recurso). Para ello el usuario debe presionar el botón del formulario correspondiente al período del que se requiere la información, la aplicación recolecta los datos y los presenta en forma de gráfica (con su respectiva tabla de datos) para facilitar la comprensión de la información por parte del usuario.

Flujo Básico:

1. El caso de uso comienza cuando el usuario común accede a la página de la aplicación.
2. Se muestra el formulario con diferentes pestañas de acuerdo al ámbito de despliegue de la información (Estadísticas de la UTPL, del Ecuador o de otras fuentes).
3. El usuario selecciona la pestaña correspondiente a la información que busca.
4. La aplicación muestra tres secciones (cada una de ellas contiene gráfica y datos) que corresponden al año, mes y semana.
5. El usuario selecciona el período de tiempo correspondiente (trimestre y año o mes y año) dependiendo de la sección y presiona el botón graficar.

A. Casos de uso

6. La aplicación recolecta los datos de la BDD correspondiente al período ingresado, los clasifica, genera la gráfica y los archivos de respaldo (CSV y RDF) y, finalmente, presenta la gráfica al usuario.

Flujo Alternativo:

- *En el punto 5* Si el usuario no selecciona alguno de los parámetros necesarios para la consulta, se toman los valores por defecto de los campos de la aplicación.

A.2.4. Caso de Uso: Estadística consolidada (Ecuador)

Propósito: Generar las gráficas estadísticas (por año, mes y semana) que resuma los recursos Creative Commons del Ecuador.

Descripción: Este caso de uso especifica la generación de las gráficas de la aplicación de acuerdo a los parámetros establecidos (año, mes, día, tipo de recurso). Para ello el usuario debe presionar el botón del formulario correspondiente al período del que se requiere la información, la aplicación recolecta los datos y los presenta en forma de gráfica (con su respectiva tabla de datos) para facilitar la comprensión de la información por parte del usuario.

Flujo Básico:

1. El caso de uso comienza cuando el usuario común accede a la página de la aplicación.
2. Se muestra el formulario con diferentes pestañas de acuerdo al ámbito de despliegue de la información (Estadísticas de la UTPL, del Ecuador o de otras fuentes).
3. El usuario selecciona la pestaña correspondiente a la información que busca.
4. La aplicación muestra tres secciones (cada una de ellas contiene gráfica y datos) que corresponden al año, mes y semana.
5. El usuario selecciona el período de tiempo correspondiente (trimestre y año o mes y año) dependiendo de la sección y presiona el botón graficar.

A. Casos de uso

6. La aplicación recolecta los datos de la BDD correspondiente al período ingresado, los clasifica, genera la gráfica y los archivos de respaldo (CSV y RDF) y, finalmente, presenta la gráfica al usuario.

Flujo Alternativo:

- *En el punto 5* Si el usuario no selecciona alguno de los parámetros necesarios para la consulta, se toman los valores por defecto de los campos de la aplicación.

A.2.5. Caso de Uso: Creación del repositorio local

Propósito: Crear la Base de Datos y las tablas necesarias para el crawler.

Descripción: Este caso de uso especifica la configuración de la base de datos de la aplicación web. El administrador crea la BDD y las tablas necesarias para el correcto funcionamiento del crawler.

Flujo Básico:

1. El caso de uso empieza cuando el administrador selecciona la opción de “Crear base de datos” de la página principal de PHPMyAdmin.
2. La pantalla muestra el entorno de creación de las tablas que forman parte de la BDD.
3. El administrador introducirá los scripts de creación de las tablas de la BDD mediante la pestaña “Importar”.
4. El administrador introducirá los scripts de inserción de datos primarios en las tablas de la BDD.
5. Se le muestra el mensaje que indica la creación satisfactoria de las tablas y la inserción de los datos primarios en ellas.

Flujo Alternativo:

- *En el punto 4* Si no se ha creado la BDD con la configuración requerida, se produce un error al insertar los datos primarios.

A.2.6. Caso de Uso: Clasificación de documentos

Propósito: Clasificar las páginas web que contienen recursos Creative Commons de acuerdo a su procedencia (UTPL u otros).

Descripción: Este caso de uso especifica la clasificación de las páginas web encontradas gracias al crawler de acuerdo a su procedencia; es decir si provienen de la UTPL o de otras fuentes.

Flujo Básico:

1. El caso de uso comienza cuando el usuario común accede a la página de la aplicación.
2. Se muestra un formulario donde el usuario puede presionar el botón "Enviar" del apartado crawler para poner en marcha la búsqueda de recursos Creative Commons en la web.
3. Se ingresa en la base de datos un listado de enlaces a las páginas web que contienen recursos Creative Commons como resultado de la búsqueda del crawler.
4. A continuación y de forma automática se clasifican las páginas web de acuerdo a su procedencia, si son de la UTPL se guardan en la tabla `urltpl` de la base de datos, si provienen de otra fuente se guardan en la tabla `urlotros`.

Flujo Alternativo:

- *En el punto 3* Si no se ha establecido la conexión con la BDD se genera un error.

A.2.7. Caso de Uso: Exploración Web personalizada

Propósito: Explorar la web en busca de recursos Creative Commons y obtener los resultados.

Descripción: Este caso de uso especifica la puesta en marcha del crawler por parte del usuario. El usuario accede a la página de la aplicación y pulsa el botón de lanzamiento del crawler. Como resultado obtiene una lista de enlaces a las páginas que contienen recursos Creative Commons.

A. Casos de uso

Flujo Básico:

1. El caso de uso comienza cuando el usuario común accede a la página de la aplicación.
2. Se muestra un formulario donde el usuario puede presionar el botón “Enviar” del apartado crawler para poner en marcha la búsqueda de recursos Creative Commons en la web.
3. Se ingresa en la base de datos un listado de enlaces a las páginas web que contienen recursos Creative Commons como resultado de la búsqueda del crawler.

Flujo Alternativo:

- *En el punto 3* Si no se ha establecido la conexión con la BDD se genera un error.

APÉNDICE B

EVALUACIÓN DE ESTADÍSTICA

Las encuestas se realizaron a los usuarios finales de la aplicación, al personal encargado del desarrollo y de la administración de los portales web de la UTPL. Para tener un conocimiento completo del tema la evaluación del portal se desarrolló en dos partes:

- Realizar las distintas operaciones disponibles en el portal.
- Contestar la encuesta sobre la funcionalidad y usabilidad del portal.

La encuesta aplicada a los usuarios consta de dos partes bien diferenciadas, la primera evalúa la funcionalidad y la segunda la usabilidad. Dentro de la funcionalidad (8 ítems) se consideran categorías como: accesibilidad, seguridad y navegabilidad; y, en cuanto a la usabilidad (11 ítems) se contempla las categorías siguientes: amigabilidad, calidad del entorno y calidad del contenido.

Con la aplicación de esta entrevista se busca evaluar los aspectos básicos relacionados con los portales web, entre ellos: accesibilidad (facilidad de acceso), seguridad (control y administración de usuarios), navegabilidad (facilidad para la navegación dentro del portal), amigabilidad (facilidad de uso), calidad del entorno (uso adecuado de los elementos que conforman la parte estética) y, calidad del contenido (pertinencia de la información del portal).

B. Evaluación de Estadística

Datos informativos	Datos del usuario:			
	Fecha:			
	Nota: Marque con una X la opción que le parezca correcta			
Funcionalidad				
Accesibilidad	<i>Siempre</i>	<i>Casi siempre</i>	<i>A veces</i>	<i>Nunca</i>
1. ¿El acceso al portal Estadísticas Creative Commons UTPL se realizó en un tiempo razonable?				
2. ¿Es aceptable el tiempo de respuesta al realizar la petición de un gráfico al portal?				
3. ¿Es aceptable el tiempo de respuesta al realizar la petición de un reporte al portal?				
4. ¿Es aceptable el tiempo de respuesta al realizar la petición de una licencia al portal?				
Seguridad	<i>Siempre</i>	<i>Casi siempre</i>	<i>A veces</i>	<i>Nunca</i>
5. ¿Se permite al usuario acceder a los datos y modificarlos?				
6. ¿Se puede conocer quién y cuándo ha actualizado la información del portal de Licencias Creative Commons UTPL?				
Navegabilidad	<i>Siempre</i>	<i>Casi siempre</i>	<i>A veces</i>	<i>Nunca</i>
7. ¿Es fácil la navegación en el portal de Licencias Creative Commons UTPL?				
8. ¿En todo momento se sabe en qué parte de la aplicación nos encontramos?				

Figura B.1.: Encuesta de evaluación de la funcionalidad y usabilidad de Estadística (parte 1) (22).

B. Evaluación de Estadística

Usabilidad				
Amigabilidad	<i>Muy Buena</i>	<i>Buena</i>	<i>Regular</i>	<i>Mala</i>
1. ¿Es amigable la interfaz del portal?				
2. ¿Se puede obtener la información fácilmente?				
3. ¿Permite obtener reportes fácilmente?				
Calidad del entorno	<i>Muy Buena</i>	<i>Buena</i>	<i>Regular</i>	<i>Mala</i>
4. El tipo de letra empleado es:				
5. El tamaño de la fuente es:				
6. El uso de imágenes e iconos es:				
7. La calidad técnica de los enlaces es:				
8. La estética de los tabs es:				
9. El estilo del lenguaje utilizado es:				
Calidad del contenido	<i>Muy Buena</i>	<i>Buena</i>	<i>Regular</i>	<i>Mala</i>
10. La estructura de la información es:				
11. La estructura del contenido del portal es:				

Figura B.2.: Encuesta de evaluación de la funcionalidad y usabilidad de Estadística (parte 2) (22).

APÉNDICE C

INSTALACIÓN Y CONFIGURACIÓN DE ESTADÍSTICA

El objetivo de este manual es explicar la instalación de Estadística en un servidor apache. Sin ser el objetivo de este manual, en la primera parte se dan una serie de nociones en la instalación y configuración de todas las aplicaciones necesarias para el correcto funcionamiento de Estadística. Como siguiente punto se explicará como instalar la aplicación en el servidor apache.

C.1. Requerimientos básicos

- Computadora Pentium IV, Memoria 512 MB. (Consideración mínima).
- Se debe permitir el tráfico TCP y UDP en la red.
- Instalar el servidor web Apache 2.0 Handler.
- Lenguaje de programación PHP con las librerías GD y XMLWriter activadas.
- MySQL 5.1

- Tener instalado un navegador de internet que soporte javascript, el más común es Mozilla Firefox también Internet Explorer 6 o superior.
- El navegador debe tener activos las cookies.

C.2. Instalación de Apache

El proceso para instalar Apache en Windows es muy sencillo. Es posible que la configuración del servidor se complique un poco, sin embargo, si deseamos utilizarlo con las funciones habituales no tendremos mayores dificultades.

Descarga de Apache. El servidor web Apache se puede descargar de la página oficial <http://www.apache.org/>. Este sitio posee una sección particular para el servidor HTTP (protocolo de transmisión utilizado en la web) en la que se busca el enlace para la descarga.

Instalación. Para la instalación en Windows siga los pasos siguientes:

- Inicie el instalador del Apache haciendo doble clic sobre el icono del archivo `apache_X.X.XX-win32-x86-no_ssl.msi`.
- Deberá escoger diversos valores para la configuración del servidor entre ellas: dominio de red, nombre del servidor y e-mail del administrador. Puede utilizar `localhost` para los dos primeros items y colocar cualquier e-mail para el tercero. Todos estos datos son almacenados en el archivo `httpd.conf`. También se debe escoger si se instalará Apache como un servicio para todos los usuarios o no.
- A continuación deberá escoger el tipo de configuración: típica o personalizada. Se recomienda la configuración típica.
- En la ventana siguiente se pregunta por el directorio de instalación; por defecto es `C:\Program Files\Apache Group` pero se puede cambiar al valor siguiente `C:\Apache2\` o cualquier otro destino preferido.
- Finalmente escoja la opción `Install` para completar el proceso de instalación de Apache.

Nota: No se debe tener otro servidor configurado para trabajar en el puerto 80 (por ejemplo IIS que viene en las versiones profesionales de Windows). En caso de tener en marcha otro servidor web la instalación puede fallar y mostrarnos

un mensaje de error que indique que no podemos tener dos servidores a la vez escuchando en el puerto 80. Para solucionar el problema basta con que paremos el IIS u otro servidor activo.

Una vez instalado, el Apache se pone en funcionamiento. Para controlar el Apache podemos encontrar en el botón de inicio, sección programas, un nuevo grupo llamado "Apache HTTP Server", con iconos para detener el servidor, ponerlo en marcha, editar el archivo de configuración `httpd.conf` o para ver los logs de acceso.

También podemos ver en la barra de tareas un nuevo icono que indica que el Apache está funcionando y que ofrece opciones para controlarlo si pulsamos sobre él. Para comprobar que el servidor está activo y funciona correctamente podemos abrir un explorador y acceder a la URL `http://localhost`, que es nuestro propio servidor. Si todo ha ido bien observaremos el mensaje de bienvenida del servidor con algo como "Funciono! ¡El servidor Apache ha sido instalado en este equipo!".

C.3. Instalación de MySQL

MySQL 5.0 para Windows está disponible en tres formatos de distribución:

- La distribución binaria contiene un programa de instalación que instala cada elemento necesario para iniciar el servidor inmediatamente.
- La distribución de código fuente contiene todo el código y ficheros de soporte para generar ejecutables utilizando el compilador de VC++ 6.0

Siempre que sea posible debería emplearse la distribución binaria. Es más simple que las otras y no se necesita ninguna herramienta adicional para poner en funcionamiento el servidor MySQL.

1. Inicialmente se debe dar doble clic en el archive `mysql-essential-5.0.51-win32.msi`

El servidor MySQL Server 5.0 – Setup Wizard aparece con el mensaje Welcome to the Setup Wizard for MySQL Server 5.0. Clic en siguiente.

2. Tipo de instalación

Verifique que "Típica" esté seleccionada y continuar.

3. Listo para instalar el programa. Clic en Instalar.

C. Instalación y Configuración de Estadística

4. Instalado MySQL Server 5.0, Espere mientras el asistente instala MySQL
5. MySQL.com Sign-Up. Verifique que “Skip Sign-Up” está seleccionado
6. Asistente Completado.
Verifique que “Configurar El Servidor MySQL Server ahora” está activado.
Clic Finalizar.
7. El asistente de La instancia del Servidor MySQL Server aparece con el mensaje Bienvenido al asistente de la configuración de la Instancia de MySQL Server. Clic en siguiente.
8. Por favor seleccionar un tipo de configuración.
Verifique que “Configuración Estándar” está seleccionada.
Clic siguiente.
9. Por favor establezca las opciones.
Verifique que “Install as a Windows Service” está seleccionada.
Verifique que “MySQL” es el nombre del servicio de Windows a utilizar.
Verificar que “Launch the MySQL Server automatically” está marcada.
Verificar que “Include Bin Directory in Windows PATH” está seleccionado.
Clic siguiente.
10. Por favor configure las opciones de seguridad
Ingrese y confirme (reingrese) su contraseña
Clic siguiente
11. Listo para ejecutar... Clic en Ejecutar.
12. Procesando configuración...
Espere por favor mientras el asistente de configuración configure el Servidor MySQL.
Clic en Finalizar.

Connection Error

En algunos casos (usualmente cuando MySQL está siendo instalado sobre versiones previas), la instancia de configuración del Servidor MySQL dice un mensaje similar a “Connection Error”.

En este caso se debe ingresar al menú Todos los programas y reconfigurar la instancia de MySQL Server.

APÉNDICE D

MANUAL DEL USUARIO

A modo de introducción, el manual del usuario de Estadística le ofrece una visión amplia y genérica pero suficiente para utilizar correctamente esta aplicación.



Figura D.1.: Pantalla de ingreso al portal (22).

D.1. Uso de Estadística

Esta aplicación permite el cálculo y la visualización gráfica de las estadísticas de los recursos Creative Commons licenciados en la jurisdicción del Ecuador.

Al ingresar se despliega la pantalla siguiente:

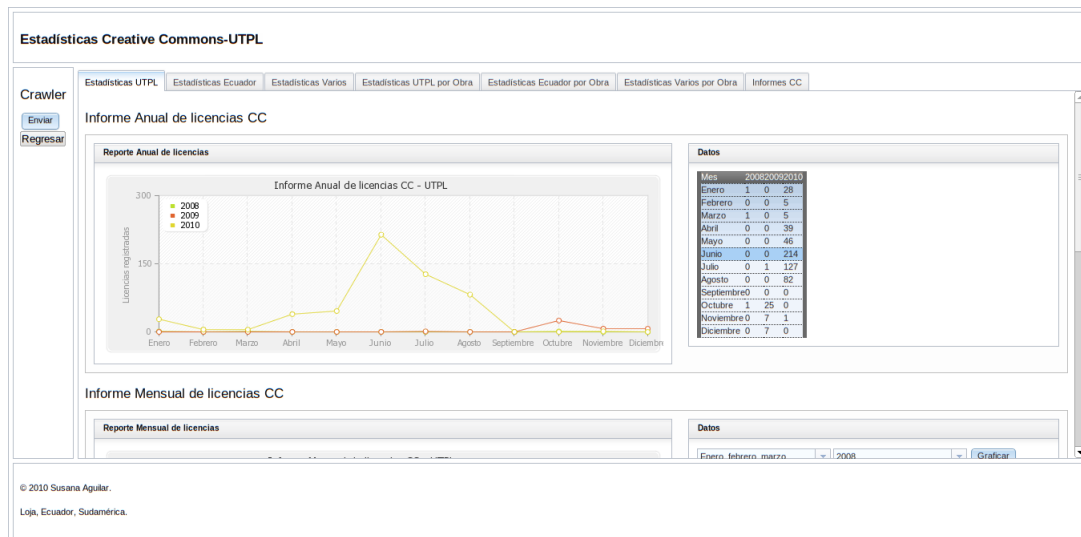


Figura D.2.: Pantalla de Estadísticas (22).

En ella se debe escoger una de las pestañas de acuerdo a la información requerida por el usuario. Las tres primera pestañas presentan la información consolidada respecto de la cantidad de licencias registradas en un período de tiempo determinado; las tres siguientes presentan la información categorizada de acuerdo el tipo de obra registrada en un período de tiempo determinado y la última permite obtener una amplia variedad de informes de acuerdo a los parámetros ingresados por el usuario.

Para la obtención de los informes trimestrales, el usuario deberá escoger el período y el año respectivos; luego pulsará el botón “Graficar” y, automáticamente se generará la gráfica correspondiente al período requerido.

Para la obtención de los informes semanales, el usuario deberá escoger el período y el año respectivos; luego pulsará el botón “Graficar” y, automáticamente se generará la gráfica correspondiente al período requerido.

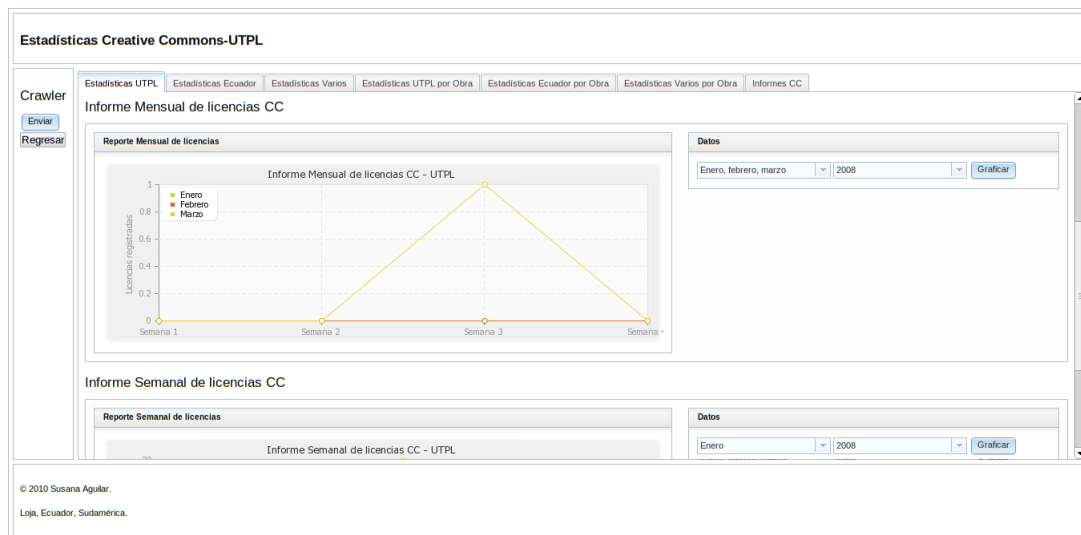


Figura D.3.: Gráfica trimestral de recursos CC (22).

En el apartado correspondiente a los informes se puede ingresar cuatro opciones diferentes: la primera permite seleccionar el ámbito del informe (recursos de la UTPL o de fuentes externas); la segunda permite escoger el tipo de obra (audio, video, texto, etc.); la tercera y cuarta el período de tiempo del informe. Luego, se debe pulsar el botón “Generar reporte” para obtener los resultados respectivos.

Reporte de licencias tipo Texto de Febrero del 2010

Título	Autor	Tipo de Licencia	Tipo de Obra
Elaboración del plan estratégica y rediseño organizacional para el caso de la empresa de transporte pesado Transgalsant cia. Ltda con sede en la ciudad de Quito	Galánaga Santamaria, Pablo Washington	BY-NC-ND	texto
Plan de negocios para la empresa Optimiza y su aplicación en la ciudad de Loja julio 2008-junio 2009	Guarinda Castillo, Juan Carlos	BY-NC-ND	texto
"Plan de negocios para la empresa Ail Wilib en la ciudad de Loja, marzo 2008 – marzo 2009"	González González, José Alberto	BY-NC-ND	texto
Estudio del banco de semillas herbáceas en la reserva "La Ceiba", Zapotillo, Ecuador	Soto Arias Johana Elizabeth	BY-NC-SA	texto
"Determinación de la presencia y germinabilidad de semillas de especies herbáceas en heces de ganado caprino, en un bosque seco del sur del Ecuador"	Guaman Ponce, Ximena Yadira	BY-NC-SA	texto

[Regresar](#)

Figura D.4.: Reporte generado por la aplicación (22).

D.2. Uso de Licenciamiento

Esta aplicación permite el licenciamiento de varios tipos de recursos mediante creative commons con la restricción de que la licencia generada pertenecerá siem-

pre a la jurisdicción del Ecuador.

Al ingresar se despliega la pantalla siguiente:

Creative Commons Ecuador

Ponga una licencia a su obra

Con una licencia de Creative Commons, **mantienes tus derechos de autor** pero permites a otras personas **copiar y distribuir su obra**, siempre y cuando reconozcan la correspondiente autoría -- y solamente bajo las condiciones que especifiques aquí. Si las licencias Creative Commons son algo nuevo para ti, hemos preparado una lista de aspectos a tener en cuenta. Si prefieres ofrecer tu obra sin condiciones o quieres certificar que una obra está en el dominio público, escoge una de nuestras herramientas para dominio público.

Cuando usted escoge una licencia, le entregamos el código HTML que puede usar para agregar información sobre la licencia en su sitio web e información sobre cómo elegir una licencia en alguno de los sitios de alojamiento gratuito que han incorporado Creative Commons. Esto no constituye un registro y no almacenamos su elección.

¿Quiere permitir usos comerciales de su obra?

☒ Sí
☐ No

¿Quiere permitir modificaciones de su obra?

☒ Sí
☐ Sí, mientras se comparta de la misma manera
☐ No

Jurisdicción de su licencia

Ecuador

Información Adicional

Los campos siguientes son opcionales, pero serán incrustados en el código HTML generado para su licencia. Esto permite que los usuarios sepan cómo darle atribución o dónde ir para obtener más información sobre la obra.

¿Cuál es el formato de su obra?

Otro ▼

Título de la obra:

Atribuir la obra a:

Atribuir la obra a (URL):

URL de la fuente de la obra:

Más permisos sobre la obra (URL):

Escoja una licencia Regresar

Figura D.5.: Pantalla de licenciamiento (22).

En ella se debe escoger los usos comerciales de la obra, si se permite o no modificaciones, el formato, el título, la atribución, los URL de la obra y, si se desea se pueden añadir algunos permisos más. Luego se pulsa el botón “Escoja una licencia” y se generará automáticamente la licencia creative commons Ecuador para el recurso respectivo.

APÉNDICE E

MANUAL DEL PROGRAMADOR

El manual del programador incluye el código desarrollado para la creación del software de Estadísticas de los licenciamientos CC con la documentación respectiva. A continuación se describe el código de cada uno de los módulos de la aplicación.

E.1. Requerimientos básicos

- Computadora Pentium IV, Memoria 512 MB. (Consideración mínima).
- Se debe permitir el tráfico TCP y UDP en la red.
- Instalar el servidor web Apache 2.0 Handler.
- Lenguaje de programación PHP con las librerías GD y XMLWriter activadas.
- MySQL 5.1
- Tener instalado un navegador de internet que soporte javascript, el más común es Mozilla Firefox también Internet Explorer 6 o superior.
- El navegador debe tener activos las cookies.

E.2. Configuración e inicialización de la BDD

A continuación se describe los scripts utilizados tanto en la instalación como en la configuración de la base de datos. Estos archivos se encuentran en el CD adjunto a la tesis.

E.2.1. Script de instalación de la base de datos

El script de instalación de la base de datos se corresponden con el nombre de archivo siguiente BDD StatisticsCC.sql, a continuación el contenido del mismo.

```
----- Creacion de la base de datos -----
CREATE DATABASE 'StatisticsCC' DEFAULT CHARACTER SET utf8
COLLATE utf8_spanish_ci;
----- Creación de la tabla formatoObra -----
CREATE TABLE formatoObra
(
  idformatoObra int not null primary key auto_increment,
  formato varchar(45) NOT NULL
)ENGINE=INNODB;
----- Creación de la tabla tipoLicenciaCC -----
CREATE TABLE tipoLicenciaCC
(
  idtipoLicenciaCC int not null primary key auto_increment,
  tipo varchar(45) NOT NULL
) ENGINE=INNODB;
----- Creación de la tabla obra -----
CREATE TABLE obra
(
  idobra int not null primary key auto_increment,
  titulo varchar(45),
  autor text,
  webAutor text,
  direccionObra text,
  masPermisos text,
```

E. Manual del programador

```
formatoObraId int,
INDEX (formatoObraId),
FOREIGN KEY (formatoObraId) REFERENCES formatoObra(idformatoObra)
) ENGINE=INNODB;
----- Creación de la tabla licenciaCC -----
CREATE TABLE licenciaCC
(
idlicenciaCC int not null primary key auto_increment,
comercial tinyint(1) unsigned NOT NULL,
modificar varchar(45) NOT NULL,
jurisdiccion varchar(45) NULL,
tipoLicenciaId int,
INDEX (tipoLicenciaId),
FOREIGN KEY (tipoLicenciaId)
REFERENCES tipoLicenciaCC(idtipoLicenciaCC)
) ENGINE=INNODB;
----- Creación de la tabla obraLicenciaCC -----
CREATE TABLE obraLicenciaCC
(
idobraLicenciaCC int not null primary key auto_increment,
fecha date,
hora time,
obraId int,
INDEX (obraId),
FOREIGN KEY (obraId) REFERENCES obra(idobra),
licenciaCCId int,
INDEX (licenciaCCId),
FOREIGN KEY (licenciaCCId) REFERENCES licenciaCC(idlicenciaCC)
) ENGINE=INNODB;
----- Creación de la tabla tipoLicenciaCC -----
CREATE TABLE tipoLicenciaCC
(
idtipoLicenciaCC int not null primary key auto_increment,
tipo varchar(45) not null
) ENGINE=INNODB;
```

E. Manual del programador

```
----- Creación de la tabla licenciasSumario -----
CREATE TABLE licenciasSumario
(
  idlicenciasSumario int not null primary key auto_increment,
  dia varchar(10) not null,
  semana varchar(10) not null,
  mes varchar(10) not null,
  anio varchar(10) not null,
  cantidad int not null
) ENGINE=INNODB;
----- Creación de la tabla licenciasSumarioUTPL -----
CREATE TABLE licenciasSumarioUTPL
(
  idlicenciasSumarioUTPL int not null primary key auto_increment,
  dia varchar(10) not null,
  semana varchar(10) not null,
  mes varchar(10) not null,
  anio varchar(10) not null,
  cantidad int not null
) ENGINE=INNODB;
----- Creación de la tabla licenciasSumarioOtros -----
CREATE TABLE licenciasSumarioOtros
(
  idlicenciasSumarioOtros int not null primary key auto_increment,
  dia varchar(10) not null,
  semana varchar(10) not null,
  mes varchar(10) not null,
  anio varchar(10) not null,
  cantidad int not null
) ENGINE=INNODB;
----- Creación de la tabla licenciasTipo -----
CREATE TABLE licenciasTipo
(
  idlicenciasTipo int not null primary key auto_increment,
  dia varchar(10) not null,
```

```
semana varchar(10) not null,
mes varchar(10) not null,
anio varchar(10) not null,
cantidad int not null,
tipoLicenciaId integer NOT NULL,
INDEX (tipoLicenciaId),
FOREIGN KEY (tipoLicenciaId)
REFERENCES tipoLicenciaCC(idtipoLicenciaCC)
) ENGINE=INNODB;
----- Creación de la tabla licenciasTipoUTPL -----
CREATE TABLE licenciasTipoUTPL
(
idlicenciasTipoUTPL int not null primary key auto_increment,
dia varchar(10) not null,
semana varchar(10) not null,
mes varchar(10) not null,
anio varchar(10) not null,
cantidad int not null,
tipoLicenciaId int NOT NULL,
INDEX (tipoLicenciaId),
FOREIGN KEY (tipoLicenciaId)
REFERENCES tipoLicenciaCC(idtipoLicenciaCC)
) ENGINE=INNODB;
----- Creación de la tabla licenciasTipoOtros -----
CREATE TABLE licenciasTipoOtros
(
idlicenciasTipoOtros int not null primary key auto_increment,
dia varchar(10) not null,
semana varchar(10) not null,
mes varchar(10) not null,
anio varchar(10) not null,
cantidad int not null,
tipoLicenciaId int NOT NULL,
INDEX (tipoLicenciaId),
FOREIGN KEY (tipoLicenciaId)
```

E. Manual del programador

```
REFERENCES tipoLicenciaCC(idtipoLicenciaCC)
) ENGINE=INNODB;
----- Creación de la tabla urlOtros -----
CREATE TABLE urlOtros
(
  idurlOtros int not null primary key auto_increment,
  url text,
  fecha date,
  tipoLicenciaId int,
  INDEX (tipoLicenciaId),
  FOREIGN KEY (tipoLicenciaId)
  REFERENCES tipoLicenciaCC(idtipoLicenciaCC)
) ENGINE=INNODB;
----- Creación de la tabla urlSeed -----
CREATE TABLE urlSeed
(
  idurlSeed int not null primary key auto_increment,
  url text,
  visitado tinyint(1) unsigned
) ENGINE=INNODB;
----- Creación de la tabla urlUtpl -----
CREATE TABLE urlUtpl
(
  idurlUtpl int not null primary key auto_increment,
  url text,
  fecha date,
  tipoLicenciaId int,
  INDEX (tipoLicenciaId),
  FOREIGN KEY (tipoLicenciaId)
  REFERENCES tipoLicenciaCC(idtipoLicenciaCC)
) ENGINE=INNODB;
```

Debe tomarse en cuenta la codificación utilizada en la creación de la base de datos ya que esta permite una fácil gestión de los datos en idioma español, esto especialmente si se crea la base de datos de forma manual sin utilizar el script.

E.2.2. Script de configuración de la base de datos

El script de configuración de la base de datos se corresponden con el nombre de archivo siguiente StatisticsCC data.sql, a continuación el contenido del mismo.

```
INSERT INTO formatoObra (formato)
VALUES ('otro');
INSERT INTO formatoObra (formato)
VALUES ('audio');
INSERT INTO formatoObra (formato)
VALUES ('video');
INSERT INTO formatoObra (formato)
VALUES ('imagen');
INSERT INTO formatoObra (formato)
VALUES ('texto');
INSERT INTO formatoObra (formato)
VALUES ('interactivo');
INSERT INTO tipoLicenciaCC (tipo)
VALUES ('BY');
INSERT INTO tipoLicenciaCC (tipo)
VALUES ('BY-NC');
INSERT INTO tipoLicenciaCC (tipo)
VALUES ('BY-SA');
INSERT INTO tipoLicenciaCC (tipo)
VALUES ('BY-ND');
INSERT INTO tipoLicenciaCC (tipo)
VALUES ('BY-NC-SA');
INSERT INTO tipoLicenciaCC (tipo)
VALUES ('BY-NC-ND');
INSERT INTO tipoLicenciaCC (tipo)
VALUES ('BY');
INSERT INTO tipoLicenciaCC (tipo)
VALUES ('BY-NC');
INSERT INTO tipoLicenciaCC (tipo)
VALUES ('BY-SA');
INSERT INTO tipoLicenciaCC (tipo)
```

E. Manual del programador

```
VALUES ('BY-ND');
INSERT INTO tipoLicenciaCC (tipo)
VALUES ('BY-NC-SA');
INSERT INTO tipoLicenciaCC (tipo)
VALUES ('BY-NC-ND');
INSERT INTO urlSeed(url, visitado)
VALUES ('http://repositorio.utpl.edu.ec/tesis.jsp', 0);
INSERT INTO urlSeed(url, visitado)
VALUES ('http://bibdigital.epn.edu.ec/handle/15000/1020', 0);
INSERT INTO urlSeed(url, visitado)
VALUES ('http://www.flacsoandes.org/dspace/handle/10469/546', 0);
INSERT INTO urlUtpl(url, fecha, tipoLicenciaId, visitado)
VALUES ('http://repositorio.utpl.edu.ec/handle/123456789/994',
'2008-01-22', 5, 0);
INSERT INTO urlUtpl(url, fecha, tipoLicenciaId, visitado)
VALUES ('http://repositorio.utpl.edu.ec/handle/123456789/1178',
'2008-03-12', 5, 0);
INSERT INTO urlUtpl(url, fecha, tipoLicenciaId, visitado)
VALUES ('http://repositorio.utpl.edu.ec/handle/123456789/1206',
'2008-10-14', 1, 0);
INSERT INTO urlUtpl(url, fecha, tipoLicenciaId, visitado)
VALUES ('http://repositorio.utpl.edu.ec/handle/123456789/1985',
'2009-07-23', 5, 0);
INSERT INTO urlUtpl(url, fecha, tipoLicenciaId, visitado)
VALUES ('http://repositorio.utpl.edu.ec/handle/123456789/2240',
'2009-10-01', 5, 0);
INSERT INTO urlUtpl(url, fecha, tipoLicenciaId, visitado)
VALUES ('http://repositorio.utpl.edu.ec/handle/123456789/2286',
'2009-10-21', 6, 0);
```

E.3. Código de la aplicación

E.3.1. Codificación del Web Crawler

La extracción de la información de internet se realiza utilizando un Web Crawler, el código es el siguiente:

```
<?php
// Extrae links de la url mediante expresiones regulares.
class Crawler{
// Busca links en la página mediante expresiones regulares.
function crawl( $url ){

    try{ $fp = fopen( $url, "r" );
    if ( $fp === false ){

        throw new Exception( "URL no encontrada" );
    }else{

        // Se extra el contenido HTML de una url.
        $page_contents = file_get_contents( $url );
        // Expresiones regulares para buscar etiquetas a href.
        $match_result = preg_match_all( '/<\s*A\s*HREF=
"([^\"]+)\s*>([^\>]*)</A>/i',
        $page_contents, $match_array1, PREG_SET_ORDER );
        $match_result1 = preg_match_all( '/<\s*A\s*HREF=
"([^\"]+)\s*TARGET=" [^\"]+\s*>/i',
        $page_contents, $match_array2, PREG_SET_ORDER );
        $match_array = array_merge( (array)$match_array1,
        (array)$match_array2 );
        return $match_array;
    }

}

}catch ( Exception $exception ){

    $error = "Error: ".$exception->getMessage();
    print( "$error <br>" );

}
```



```
} // fin de la funcion crawl
// Imprime el resultado respetando la variable set_order.
function printer_set_order( $url, $match_array ){

    foreach( $match_array as $entry ){

        $href = $entry[1];
        $anchortext = $entry[2];
        print( "<b>HREF</b>: $entry[1];
        <b>ANCHORTEXT</b>: $entry[2]<br>" );

    }

} // fin de la funcion printer_set_order
// Imprime el resultado respetando la variable pattern_order.
function printer_pattern_order( $url, $match_array ){

    $items = count( $match_array );
    if ( $items>0 ){

        foreach( $match_array as $entry ){

            print( "<b>Property</b>: $entry[0];<br>" );

        }

    }

} // fin de la funcion printer_pattern_order
// Extrae links Creative Commons internacional
// utiliza expresiones regulares.
function parser_CCint( $match_array ){

    $space = " ";
    $crawl_contents = "";
    foreach( $match_array as $entry ){

        $href = $entry[1];
        $anchortext = $entry[2];
        $crawl_contents .= $href.$space.$anchortext.$space;

    }

    // Expresiones regulares que buscan etiquetas
    // en cuyo contenido se encuentre la cadena
```

```
// creative commons.
$match_result = preg_match_all(
    '/http:\\\\creativecommons.org\\/licenses\\/([^\\/]+)\\/
    (\\d\\.\\d)\\/\\/i',$crawl_contents,
    $match_array_result, PREG_PATTERN_ORDER );
if ( $match_result == 1 )
    return $match_array_result;

} // fin de la funcion parser_CCint
// Extrae links Creative Commons Ecuador
function parser_CCec( $match_array ){

    $space = " ";
    $crawl_contents = "";
    foreach( $match_array as $entry ){
        $href = $entry[1];
        $anchortext = $entry[2];
        $crawl_contents .= $href.$space.$anchortext.$space;
    }
    // Expresiones regulares que buscan etiquetas
    // en cuyo contenido se encuentre la cadena
    // creative commons.
    $match_result = preg_match_all(
        '/http:\\\\creativecommons.org\\/licenses\\/([^\\/]+)\\/
        (\\d\\.\\d)\\/ec\\/\\/i', $crawl_contents,
        $match_array_result, PREG_PATTERN_ORDER );
    if ( $match_result == 1 )
        return $match_array_result;

} // fin de la funcion parser_CCec
// Completa los links internos al sitio
// para presentarlos al usuario de forma
// comprensible
function parser_Links( $url, $match_array ){

    $domain = $this->getDomain( $url );
    //echo "dominio: $domain[0]<br>";
```

```
$i = 0;
foreach( $match_array as $entry ){
    $href = $entry[1];
    //echo "href: $href <br>";
    $position = strpos( $href, "/" );
    $link = "";
    if ( $position == 0 ){

        $substring = substr( $domain[0], 0, -1 );
        $link .= $substring.$href;
        $links[$i] = $link;
        //echo "links: $links[$i]<br>";
        $i = $i+1;
    }else{
        ;//echo "<b>Link:</b> $href;
        <b>Posicion barra:</b> $position<br>";
    }
}
return $links;
} // fin de la funcion parser_Links
// Obtiene el dominio de una url
function getDomain( $url ){

    $match_result = preg_match( '/http:\\\\\[^\\/]+\\\/i',
    $url, $match_array_result );
    return $match_array_result;
} // fin de la funcion getDomain
}
?>
```

E.3.2. Codificación del módulo de actualización de datos

La actualización de la información de la base de datos MySQL se realiza gracias al código siguiente:

```
<?php
$connect_parameters = "host=localhost dbname=LicenciaCC
user=susana password=susana";
if($link = pg_connect($connect_parameters)){
    $sSql = "select * from \"formato_obra\"";
    $sResult = pg_query($link, $sSql);
    if(pg_num_rows($sResult) > 0){
        print("<table border=\"1\">");
        print("<tr><th>ID</th><th>Formato</th></tr>");
        while($sRow = pg_fetch_object($sResult)){

            print("<tr><th>$sRow->idformato_obra</th>
                <td>$sRow->formato</td>");
            print("</tr>");
        }

        print("</table>");
    }else{
        print("<p> No hay registros en la base de datos"); }
    }else{
        print("<p> La coneccion con la base de datos LicenciaCC

            ha fallado <p>");
    }
?>
```

E.3.3. Codificación del módulo de estadística

El módulo de estadística maneja diversas funciones, entre ellas:

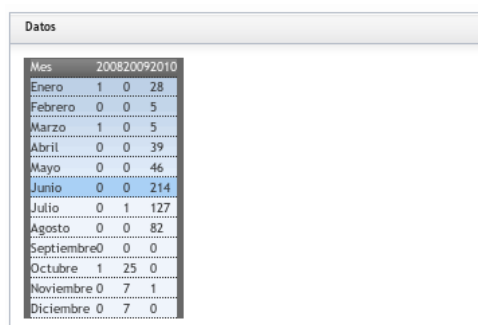
- Extracción de la información.
- Generación de la gráfica de acuerdo a los datos obtenidos.
- Presentación de la gráfica y de la información al usuario.

La extracción de la información a utilizarse en la creación de la gráfica estadística se realiza con el siguiente código PHP:

```
<?php
// Apertura del archivo CSV que contiene la información para
// elaborar el grafico.
if (($fp = fopen("graph/tmp/grafico1anioutpl.csv", "r")) !== FALSE) {
```

```
// Visualización en pantalla de la tabla de datos del gráfico.
print("<table>");
print("<thead><tr><th>Mes</th>
<th>2008</th><th>2009</th><th>2010</th></tr></thead>");
print("<tbody>");
$i = 0;
while (($datos = fgetcsv($fp, 50, ",", " ")) != FALSE ) {
    $numero = count($datos);
    if ( $i == 5 ){
        print ("<tr bgcolor=\"#a9d0f5\">");
    } else {
        print ("<tr>");
    }
    for($c = 0; $c < $numero; $c++){
        print ("<td>$datos[$c]</td>");
    }
    print("</tr>");
    $i = $i + 1;
}
print ("</tbody></table>");
fclose ($fp);
}
?>
```

Los resultados de la ejecución del código anterior son los siguientes:



Mes	2008	2009	2010
Enero	1	0	28
Febrero	0	0	5
Marzo	1	0	5
Abril	0	0	39
Mayo	0	0	46
Junio	0	0	214
Julio	0	1	127
Agosto	0	0	82
Septiembre	0	0	
Octubre	1	25	0
Noviembre	0	7	1
Diciembre	0	7	0

Figura E.1.: Presentación de los datos (22).

La creación de la gráfica estadística se realiza utilizando la librería GD de PHP, así:

```
<?php
// Definición de datos
$DataSet = new pData; $DataSet->ImportFromCSV("graph/tmp/

    grafico1anioutpl.csv","",array(1,2,3),FALSE,0);

$DataSet->AddAllSeries(); $DataSet->SetAbsciseLabelSerie();
$DataSet->SetSerieName("2009","Serie1");
$DataSet->SetSerieName("2010","Serie2");
$DataSet->SetSerieName("2011","Serie3");
$DataSet->SetYAxisName("Licencias registradas");
$DataSet->SetYAxisUnit("");
// Configuración del grafico
$Test = new pChart(700,230);
$Test->setFontProperties("graph/Fonts/tahoma.ttf",8);
$Test->setGraphArea(70,30,680,200);
$Test->drawFilledRoundedRectangle(7,7,693,223,5,240,240,240);
$Test->drawRoundedRectangle(5,5,695,225,5,230,230,230);
$Test->drawGraphArea(255,255,255,TRUE);
$Test->drawScale($DataSet->GetData(),$DataSet->

    GetDataDescription(),SCALE_NORMAL,150,150,150,TRUE,0,2);

$Test->drawGrid(4,TRUE,230,230,230,50);
// Dibujar ejes
$Test->setFontProperties("graph/Fonts/tahoma.ttf",6);
$Test->drawTreshold(0,143,55,72,TRUE,TRUE);
// Dibujar líneas
$Test->drawLineGraph($DataSet->GetData(),$DataSet->

    GetDataDescription());

$Test->drawPlotGraph($DataSet->GetData(),$DataSet->

    GetDataDescription(),3,2,255,255,255);

// Finalizar el grafico
$Test->setFontProperties("graph/Fonts/tahoma.ttf",8);
$Test->drawLegend(75,35,$DataSet->GetDataDescription(),

    255,255,255);
```

```
$Test->setFontProperties("graph/Fonts/tahoma.ttf",10);  
$Test->drawTitle(60,22,"Informe Anual de licencias  
CC",50,50,50,585); $Test->Render("graph/ejemplo1anioutpl.png");  
?>
```

Los resultados de la ejecución del código anterior son los siguientes:

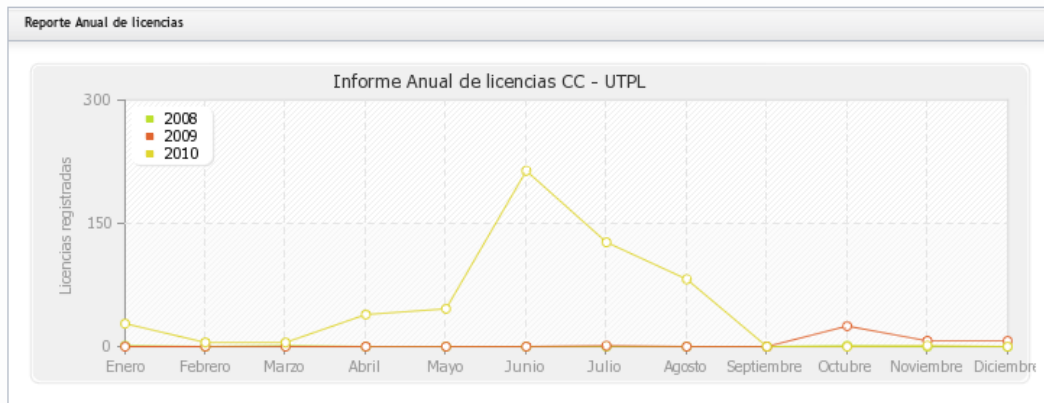


Figura E.2.: Presentación de la gráfica (22).

La presentación tanto de la gráfica como de la información al usuario se realiza mediante el framework javascript Dojo de la manera siguiente:

```
<div dojoType="dojox.widget.Portlet" title="Grafica">  
<div dojoType="dojox.widget.PortletSettings">  
Put whatever settings you like in here  
</div>  
<div>  </div>  
</div>  
<div dojoType="dojox.widget.Portlet" title="Datos">  
<div dojoType="dojox.widget.PortletSettings">  
Put whatever settings you like in here  
</div>  
<div> <?php require_once 'data/datacsv1.php'; ?>  
</div>  
</div>
```

Los resultados de la ejecución del código anterior son los siguientes:

Informe Anual de licencias CC

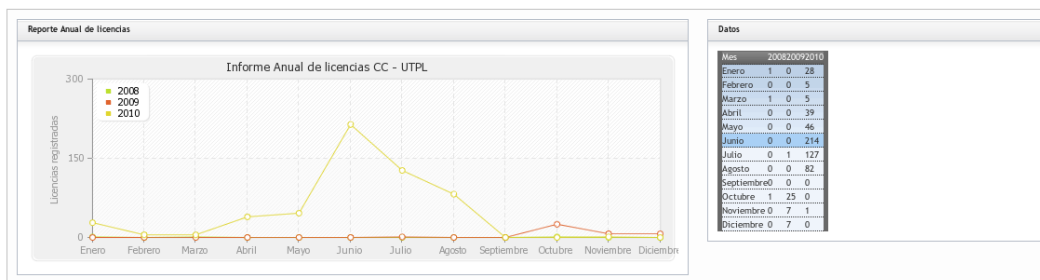


Figura E.3.: Integración de los datos con la gráfica correspondiente (22).

GLOSARIO

Administrador. Admin. Usuario de software con derechos especiales. Tiene la tarea de instalar, mantener y actualizar el software y de establecer las condiciones que el resto de usuarios necesitan para realizar sus actividades diarias.

API. Interfaz de comunicación entre componentes de software, es un conjunto de llamadas a ciertas bibliotecas que ofrecen acceso a ciertos servicios desde los procesos.

ASP. Sirve para generar páginas web dinámicas del tipo lado del servidor.

Back end. Se refiere a la parte de una aplicación de software que se ejecuta en el servidor y administra los datos. En comparación, el programa que muestra los datos al cliente que se conoce como la front end. El back end es sólo accesible a los administradores.

Creative Commons. Licencia libre que regula el uso de un documento en varios niveles. El autor tiene la opción de estipular si su nombre es citado o no, así como permitir el uso comercial o no.

CSS. Es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML.

Directorio raíz. Root directory. Punto de partida (raíz) de una estructura de directorios.

Editor WYSIWYG. Lo que ves es lo que obtienes. Se refiere a los editores en los que el texto puede ser editado en el formato deseado. Así, mientras se escribe el texto, se puede ver como lucirá en forma impresa o en la pantalla.

Front end. Parte de una red, pieza de software que es accesible a todos los usuarios. Es lo opuesto a Back End.

GNU GPL. General Public License. Licencia de software que establece el principio de derecho de copia. Garantiza que el programa sujeto de esta licencia puede ser copiado y distribuido con la condición de que no será alterado.

Hash. El símbolo #.

Hipertexto. En lugar de un texto único largo, se conectan varias partes del texto mediante enlaces.

HTML. lenguaje de marcado predominante para la construcción de páginas web.

HTTPS. Protocolo de red basado en HTTP, destinado a la transferencia segura de datos de hipertexto.

Implementación. La ejecución concreta de un programa en un lenguaje de programación.

JPEG. Es un formato de compresión, considerado como un formato de archivo.

Licencia. La concesión y la regulación de los derechos del usuario a una obra o documento por el autor. Las licencias de los documentos es un fenómeno nuevo vinculado a la existencia de documentos digitales y su capacidad para ser copiado.

Open source. El código fuente de un programa es libre y puede ser modificado y distribuido por cualquiera.

Plantilla. Diseño prototipo en el que los contenidos es integrado por el software (wiki).

Plugin. Componente del programa que puede ser acoplado a un software existente a fin de dotarlo con funciones complementarias. Ver Add-on.

Página principal. En una wiki, la página desde la que se genera otra página. Puede ser editada manualmente a fin de establecer una jerarquía.

RDBMS. Relational Database Management Systems. Sistemas de Gestión de Bases de Datos Relacionales.

Rollback. Retroceso. Restaurar la versión previa de una página.

Script. Un pequeño programa que usualmente se encuentra en código fuente y puede ser ejecutado con ayuda de un programa intérprete.

SQL. Structured Query Language. Lenguaje de Búsqueda Estructurado.

Tag. Tipo de etiqueta con la que se indica la información sobre un área determinada de un documento. En HTML, las etiquetas se utilizan para indicar los formatos.

URL. Es una secuencia de caracteres, de acuerdo a un formato estándar, usado para nombrar recursos como documentos e imágenes en internet.

Web host. Alojamiento. Proveedor de servicios que hace que el espacio de almacenamiento y la dirección de la página estén disponibles.

Web. Es una abreviatura de la World Wide Web (WWW).

BIBLIOGRAFÍA

- [1] *Las Fuentes de Información*. Obtenido agosto 5, 2010 del sitio: <http://www.monografias.com/trabajos59/fichas-fuentes-informacion/fichas-fuentes-informacion.shtml>.
- [2] *Fuente documental*. Obtenido agosto 12, 2010 del sitio: http://es.wikipedia.org/wiki/Fuentes_de_informaci%C3%B3n.
- [3] JOHNSON James. (2000). *Bases de datos: Modelos, lenguajes, diseño*. Oxford University Press. 1028 p.
- [4] HANSEN Gary. (1997). *Diseño y administración de bases de datos*. Prentice Hall, 2da edición. 592 p.
- [5] SILBERSCHATZ A., KORTH H., SUDARSHAN S. (1998). *Fundamentos de bases de datos*. McGraw-Hill, Madrid, 3ra Edición, 641 p.
- [6] ULLMAN J., WIDOM J. (1999). *Introducción a los sistemas de bases de datos*. Prentice Hall, México, 1ra Edición, 488 p.
- [7] *Manual para licenciamiento de obras con licencias Creative Commons Ecuador 3.0 en la UTPL*. Obtenido julio 10, 2010 del sitio web de Creative Commons Ecuador <http://creativecommons.ec/>.
- [8] Fundación Copyleft. Obtenido agosto 15, 2010 del sitio: <http://fundacioncopyleft.org/>.

- [9] *Reconocimiento No Comercial Sin Obra Derivada 3.0 Ecuador*. Obtenido julio 12, 2010 del sitio web de Creative Commons <http://creativecommons.org/licenses/by-nc-nd/3.0/ec/>.
- [10] ARÓSTEGUI, Julio (2001). *La investigación histórica: teoría y método*. Barcelona.
- [11] TUÑÓN DE LARA, Manuel (1985). *Por qué la historia*. Barcelona: Aula Abierta Salvat.
- [12] JACOBSON Ivar, BOOCH Grady, RUMBAUGH James. (2005). *El Proceso Unificado de Desarrollo de Software*. Pearson Educación S. A., Madrid, 464p.
- [13] JACOBSON Ivar, BOOCH Grady, RUMBAUGH James. (2005). *El Lenguaje Unificado de Modelado*. Pearson Educación S. A., Madrid, 464p.
- [14] PENDER Thomas. (2004). *UML Weekend Crash Course*. Wiley Publishing, Inc., Indiana, Indianápolis, 358p.
- [15] Trabajo de grado especialización en gerencia de recursos humanos. Obtenido diciembre 10, 2010 del sitio web <http://www.gestiopolis.com/recursos5/docs/rrhh/matrizde.htm>.
- [16] iCommons. Obtenido diciembre 15, 2010 del sitio web <http://icommons.org/>.
- [17] Licencias libres para obras. Obtenido diciembre 15, 2010 del sitio web <https://merchandlinux.wordpress.com/2009/04/04/licencias-libres-para-obras/>.
- [18] Licencia Arte Libre. Obtenido diciembre 15, 2010 del sitio web <http://artlibre.org/licence/lal/es>.
- [19] Licencia Aire Incondicional. Obtenido diciembre 15, 2010 del sitio web http://www.platoniq.net/aireincodicional_licencia.html.
- [20] Vladislav Shkapenyuk, Torsten Suel (2002). *Design and Implementation of a High-Performance Distributed Web Crawler*. ICDE 2002: 357-368.

- [21] High-Performance Web Crawler. Obtenido diciembre 15, 2010 del sitio web <http://cis.poly.edu/polybot/>.
- [22] Aguilar Susana (2010). Gráficas elaboradas para la tesis *“Desarrollo e implementación de una base de datos de los recursos licenciados con CC en Ecuador”*.
- [23] Luger, George (2005). *Inteligencia Artificial: Estructuras y Estrategias para resolver problemas complejos*. Addison-Wesley, quinta edición, México.
- [24] Web Crawler. Obtenido marzo 15, 2011 del sitio web http://en.wikipedia.org/wiki/Web_crawler.
- [25] Web Crawler Simplify. Obtenido marzo 15, 2011 del sitio web <http://en.wikipedia.org/wiki/User:SimonTrew/WebCrawlerSimplify>.
- [26] WebSPHINX: A Personal, Customizable Web Crawler. Obtenido marzo 15, 2011 del sitio web <http://www.cs.cmu.edu/~rcm/websphinx/>.

Desarrollo e implementación de una base de datos de los recursos licenciados con Creative Commons en Ecuador

Susana Aguilar Quezada

10 de septiembre de 2011

Resumen

El objetivo de este trabajo de investigación es desarrollar e implementar una base de datos de los recursos licenciados con CC en Ecuador y proveer estadísticas sobre la utilización de las mismas; esto con el fin de medir el impacto de la utilización de las licencias CC en el país y conocer la producción intelectual. En la implementación de la aplicación se construyó un crawler, utilizado para recolectar los recursos licenciados con CC Ecuador.

1. Tipos de licencias

1.1. Creative Commons

Creative Commons (CC) ofrece distintos tipos de licencias que los creadores pueden utilizar para acompañar a sus obras; frente al popular all rights reserved (todos los derechos reservados) que usualmente se adjunta en las notas de copyright de muchas creaciones, CC propone como lema el some rights reserved (algunos derechos reservados) [1].

Es importante señalar que en todos los

casos de licencias CC se presupone que el autor otorga los derechos de copia y distribución de la obra [1].

1.2. Licencias — Copyleft

Copyleft es un grupo de licencias cuyo objetivo es garantizar que cada persona que recibe una copia de una obra pueda a su vez usar, modificar y redistribuir el propio trabajo y las versiones derivadas del mismo. Unas veces se permite el uso comercial de dichos trabajos y en otras ocasiones no, dependiendo de los derechos que quiera ceder el autor [8].

Las licencias que engloba Copyleft son las siguientes:

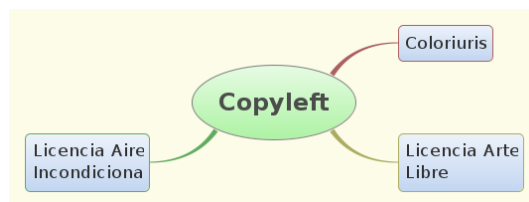


Figura 1: Licencias Copyleft [22].

1.3. Análisis de la situación actual

El proceso seguido en Ecuador para la implementación de las licencias Creative Commons es similar al desarrollado en otras regiones del mundo que adaptaron las licencias CC a su legislación. Como primer paso del proceso se conformó grupos de expertos en Derecho e Informática quienes colaboran con el equipo de CC para elaborar un documento borrador que se somete a debates y modificaciones hasta llegar a la redacción del documento definitivo que se utiliza como base para la elaboración de las licencias definitivas.

En relación con la adaptación de las licencias en Ecuador, se cuenta con una completa adecuación de las mismas al marco legislativo actual. Se encuentra disponible una traducción de las licencias al castellano gracias al esfuerzo de un equipo de voluntarios de la Universidad Técnica Particular de Loja con el que también colaboran el equipo de servicios jurídicos y de traducción de dicha institución.

Fue necesario adaptar las licencias Creative Commons a Ecuador porque, si bien el derecho de autor es uno de los regímenes jurídicos más estandarizados a nivel internacional, las diferencias entre sistemas y regulaciones siguen existiendo por lo que la idea es mantener un modelo común de licencia pero ajustado a la legislación de cada país y además, utilizando su terminología de manera que facilite su comprensión.

Una de las limitantes para la utilización de este licenciamiento en Ecuador es la falta de difusión y el desconocimiento de los potenciales usuarios. Además, en la actualidad no se cuenta con estadísticas sobre el uso de licencias CC o la cantidad de obras (debidamente categorizadas) que la utilizan.

2. Requerimientos de la Aplicación

2.1. Dominio de la aplicación

1. El dominio del problema es un sitio que provee datos e información estadística sobre las licencias Creative Commons en Ecuador.
2. Las licencias Creative Commons serán categorizadas en grupos. El enfoque inicial considera los conjuntos de licencias de la UTPL y luego, aquellos de otras fuentes procedentes de Ecuador.
 - a) La clasificación licencias CC de otras universidades, las agrupa por tipo de licencia y por tipo de recurso.
 - b) La clasificación licencias CC de la UTPL las agrupa por tipo de licencia y por tipo de recurso.

2.2. Indicadores a obtener

1. Recursos licenciados con CC por la UTPL en un año.
2. Recursos licenciados con CC por la UTPL en un mes.
3. Recursos licenciados con CC por la UTPL de acuerdo al tipo de licencia requerido.

3. Análisis y Diseño de la aplicación

Los datos necesarios para el correcto funcionamiento de la aplicación son:

- Información de la obra licenciada.
 - Título

- Autor
 - Formato (audio, video, imagen, texto, interactivo y otros)
 - Web del autor
 - Dirección de la obra
 - Más permisos
- Tipo de licencia CC utilizada.
 - BY
 - BY-NC
 - BY-SA
 - BY-ND
 - BY-NC-SA
 - BY-NC-ND

Los resultados presentados al usuario se corresponden con las estadísticas siguientes:

- Cantidad de recursos registrados con licencias CC por la UTPPL por año.
- Cantidad de recursos registrados con licencias CC por la UTPPL por mes.
- Cantidad de recursos registrados con licencias CC por la UTPPL por tipo de recurso.

3.1. Casos de uso

El sitio web de estadísticas tiene sólo cuatro casos de uso, por lo que los cuatro son arquitectónicamente significativos. La figura 2 muestra el caso de uso de alto nivel de la aplicación.

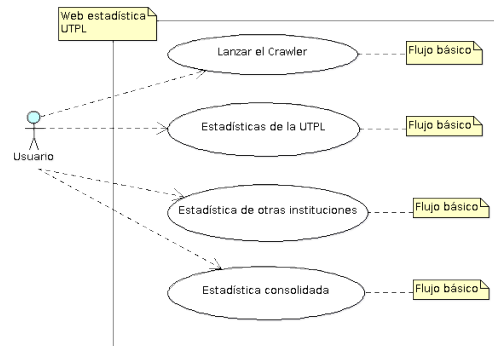


Figura 2: Diagrama de casos de uso de la aplicación [22].

Esta aplicación cuenta con un solo actor: el usuario. El usuario utiliza la internet de forma anónima y navegará por las estadísticas de las licencias CC visualizándolas.

El software de estadística requiere la implementación de un crawler cuyas funciones son: exploración automática de páginas web, clasificación de documentos e, indexación de documentos.

El usuario utiliza de forma transparente el crawler, es decir, lo inicializa pero no puede observar de forma directa su funcionamiento. Los resultados de la ejecución del crawler solamente son visibles a nivel de la capa de datos.

3.2. Diagrama de clases de la solución

La capa de entidad se encuentra sobre la base de datos, es responsable de todas las reglas del negocio y la lógica del sistema. La figura 3 muestra las clases que componen la capa de entidad de la aplicación.

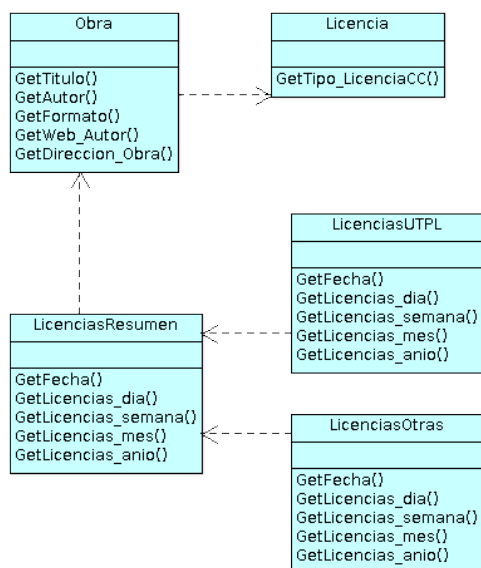


Figura 3: Modelo de clases de la capa de entidad [22].

La capa de datos se ocupa de las operaciones de almacenamiento de la información del software desarrollado. En esta aplicación, la persistencia corresponde a la base de datos implementada en MySQL.

4. Desarrollo de la aplicación

4.1. Arquitectura de la aplicación

Esta aplicación utiliza como frontal un navegador web (Firefox, IE, Opera, etc.) que permite acceder a la información y ejecutar el programa.

El modelo lógico se detalla en el diagrama que a continuación se muestra:

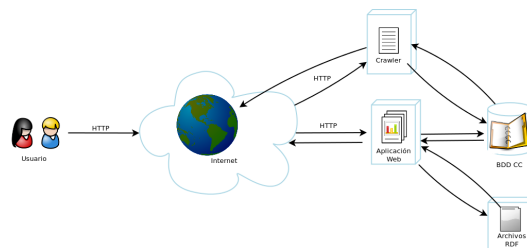


Figura 4: Modelo lógico de la aplicación [22].

Entre los elementos que conforman el software desarrollado tenemos:

Crawler se encarga de recolectar y clasificar la información proveniente de la internet. En este caso específico, el crawler recorre la web buscando páginas que contengan licencias CC, las clasifica y solamente las almacena si estas licencias tienen como jurisdicción Ecuador o están embebidas en páginas con el dominio ec.

Aplicación realiza el cálculo y la presentación de las estadísticas de licenciamiento Creative Commons, permite la generación de informes detallados, el licenciamiento de una obra y la generación de archivos RDF correspondientes a las estadísticas calculadas.

BDD la base de datos se encarga de la persistencia de los datos de la aplicación.

RDF incluídos en la aplicación para facilitar la tarea de importar/exportar la información relevante.

En cuanto a la plataforma, la aplicación está implementada con herramientas Open-Source, donde predomina la trilogía Linux-Apache-PHP utilizando como base de datos relacional MySQL.

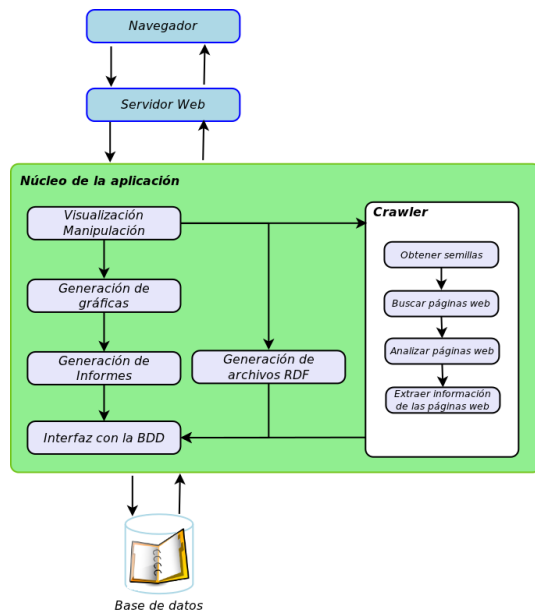


Figura 5: Modelo físico de la aplicación.

El diagrama de secuencia de la aplicación que muestra en detalle las interacciones entre los componentes del software se muestra a continuación:

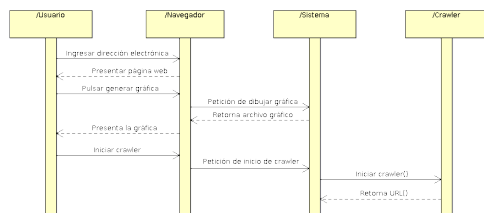


Figura 6: Diagrama de secuencia del sistema.

Para la implementación de esta aplicación cliente/servidor se utilizó el modelo de tres capas: capa de presentación, capa de negocios y capa de datos; distribuyéndose el código de la manera siguiente:

- Capa de presentación: Codificación de scripts utilizando el lenguaje JavaScript mediante el Dojo Toolkit.

- Capa de negocios: Codificación de scripts utilizando el lenguaje de programación PHP.
- Capa de datos: Implementación de la base de datos utilizando el motor MySQL.

La aplicación genera dinámicamente la información presentada al usuario final, para ello utiliza el modelo de implementación mostrado en la figura siguiente.

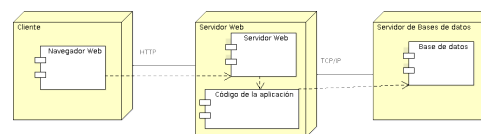


Figura 7: Modelo de implementación de la aplicación [22].

La interfaz gráfica de usuario, implementada en la página principal de la aplicación, se presenta a continuación:



Figura 8: Interfaz gráfica de usuario (GUI) de Licenciamiento.

5. Pruebas y resultados

El plan de pruebas está dirigido a los diferentes tipos de usuarios que interactúan

con Estadística y Licenciamiento, estos son: Usuarios de las estadísticas y usuarios de las licencias.

Cuadro 1: Tipos de prueba por perfil de usuario

Perfil de usuario	Tipo de Prueba
Usuario final	Funcionalidad
Usuario final	Compatibilidad
Usuario final	Robustez

5.1. Pruebas de funcionalidad y usabilidad

La evaluación de la funcionalidad y usabilidad consta de dos partes:

- Realizar las distintas operaciones disponibles en Estadísticas y Licencias Creative Commons UTPL.
- Contestar la encuesta sobre la funcionalidad y usabilidad de Estadísticas y Licencias Creative Commons UTPL.

La encuesta aplicada a los usuarios consta de dos partes bien diferenciadas, la primera evalúa la funcionalidad y la segunda la usabilidad. Dentro de la funcionalidad se consideran categorías como: accesibilidad, seguridad y navegabilidad; y, en cuanto a la usabilidad se contempla las categorías siguientes: amigabilidad, calidad del entorno, calidad del contenido y potencialidad de los recursos didácticos.

A continuación se presenta la descripción de cada una de las categorías anteriormente mencionadas:

Cuadro 2: Descripción de las categorías de la encuesta.

Tipo de Prueba	Categoría
Funcionalidad	Accesibilidad
	Seguridad
	Navegabilidad
Usabilidad	Amigabilidad
	Calidad del entorno
	Calidad del contenido

Como conclusión de las pruebas de usabilidad y funcionalidad, se puede afirmar que el uso de Estadísticas y Licencias Creative Commons UTPL ha tenido muy buena acogida, y que en cuanto al diseño las decisiones tomadas fueron las correctas.

5.2. Pruebas de compatibilidad

Estas pruebas se realizan con el fin de comprobar la compatibilidad del sistema con los navegadores web más utilizados. Para que la aplicación sea considerada como compatible con un navegador, el diseño de su interfaz gráfica debe permanecer constante, sin sufrir grandes alteraciones o cualquier tipo de cambio que afecte o disminuya su funcionalidad. Asimismo el usuario debe poder realizar las operaciones que ofrece el sistema de manera fluida, sin la presencia de mensajes de error por parte del navegador.

Cuadro 3: Compatibilidad con navegadores web.

Navegador	Versión	Compatible
Mozilla Firefox	3.0.3	Si
Mozilla Firefox	3.6	Si
Internet Explorer	8	Si
Google Chrome	2	Si

5.3. Pruebas de robustez

Con el objetivo de comprobar la capacidad del sistema para soportar múltiples accesos concurrentes sin sufrir una disminución considerable en el rendimiento, se han realizado pruebas de stress con la ayuda de la herramienta Apache JMeter (<http://jakarta.apache.org/jmeter>). Este software está diseñado para realizar pruebas de carga sobre un sistema mediante la simulación de múltiples hilos o usuarios.

A continuación se presenta la gráfica con los resultados de las pruebas realizadas para el crawler con una cantidad grande de semillas:

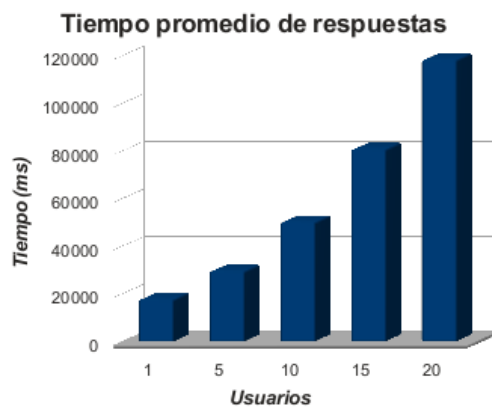


Figura 9: Gráfica del tiempo promedio del crawler, pruebas exhaustivas.

A continuación se muestra la gráfica con los resultados de las pruebas realizadas para el crawler con pocas semillas:



Figura 10: Gráfica del tiempo promedio del crawler, pruebas leves.

Se puede concluir que el deterioro en los tiempos de respuesta aumenta con el número de semillas utilizadas por el crawler durante el proceso de rastreo.

5.4. Validación del portal

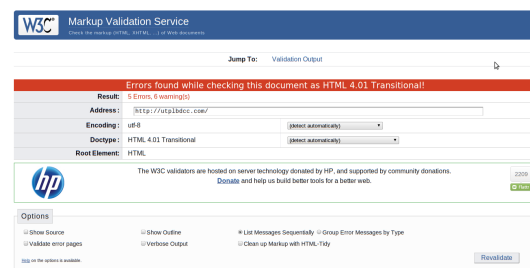


Figura 11: Validación HTML del portal web [22].

La validación es el proceso que asegura que una página web cumple con las normas y restricciones impuestas por el lenguaje HTML. Este proceso no tiene impacto alguno en el portal desarrollado, pues existen muchas páginas web que no han pasado la prueba de validación y funcionan correctamente en varios navegadores y con tiempos

de respuesta razonables, un ejemplo de ello es la página de búsqueda de Google.

Para este proceso se utilizó el validador del W3C.

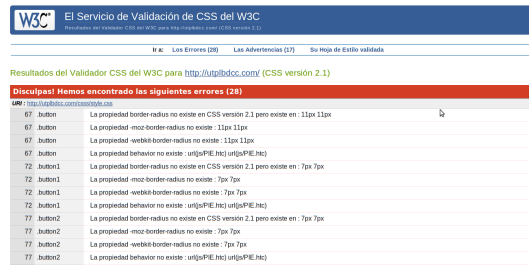


Figura 12: Validación CSS del portal web [22].



Figura 13: Validación accesibilidad del portal web [22].

Los resultados obtenidos se explican debido a que la utilización del framework open source Dojo para javascript facilita la interoperabilidad de la aplicación en los diversos navegadores existentes pero, para ello debe incluir diversas propiedades y métodos js que son válidos en un navegador determinado esto sumado a que la versión de CSS implementada (2.1) es diferente de la utilizada por el validador (2) dan como resultado los mensajes mostrados en los gráficos anteriores.

6. Discusión

El desarrollo e implementación de una base de datos de los recursos licenciados con Creative Commons en Ecuador, requirió la realización de dos aplicaciones web diferentes pero a la vez complementarias, la de licenciamiento y la de estadísticas.

La creación del sitio web de licenciamiento requirió el estudio del sistema de licencias utilizado por Creative Commons; se presenta al usuario un entorno que le permite escoger fácilmente el tipo de obra que desea licenciar y los permisos que desea otorgar a terceros.

Para la aplicación de estadísticas de las licencias Creative Commons del Ecuador se consideraron varios factores entre ellos: la construcción de un crawler; la implementación de un módulo de generación de gráficas; la fabricación de un módulo de generación de informes; y, el establecimiento de un módulo de generación de archivos RDF.

Fue necesaria la **construcción de un crawler** ya que no existía otra forma de recolectar la información referente a los recursos ecuatorianos existentes con licencias creative commons que recorrer toda la internet con una herramienta de búsqueda automática.

Para la clasificación de las páginas encontradas se consideran dos grandes grupos: los recursos licenciados por la UTPL y aquellos licenciados por otras instituciones o personas naturales, manteniendo siempre el criterio de que las licencias pertenezcan a la jurisdicción de Ecuador o en su defecto, que las páginas indizadas tengan en su dominio el sufijo ec.

El **módulo de generación de gráficas** se encarga además del cálculo de las estadísticas, para ello se realiza un conteo de las licencias obtenidas, y se generan dinámicamente archivos de resumen anual, mensual

y semanal que servirán de materia prima en la construcción tanto de los gráficos como de las tablas de datos desplegadas al usuario.

La fabricación de un **módulo de generación de informes** se realizó con el fin de tener listas de resumen de las obras debidamente clasificadas por su tipo y por el tipo de licencia que ostentan.

Además, se estableció un **módulo de generación de archivos RDF** para facilitar la reutilización tanto de los datos como de los resultados de las estadísticas obtenidas. En su creación se utilizó la librería XMLWriter además de utilizar varias librerías de metadatos existentes como la DublinCore (DC) para la descripción de la información manejada por la aplicación.

Entre las **dificultades encontradas** durante el desarrollo de esta investigación, he considerado como las más importantes las siguientes: existe poca información con respecto al desarrollo del crawler lo que extendió los tiempos investigación y desarrollo; durante la implementación, la falta de comprensión del administrador web de los requisitos mínimos que exige la aplicación acarrió un importante despilfarro de tiempo y esfuerzo, ya que éste (el administrador) se limitó a proporcionar los “errores” de la aplicación sin mencionar que ninguno de los servidores disponibles tenía instaladas las versiones necesarias de las librerías GD y XMLWriter establecidas en los requisitos de instalación.

Con este pequeño análisis se demuestra plausiblemente que los objetivos planteados al inicio de este trabajo de investigación se cumplieron de forma satisfactoria.

7. Conclusiones

- La eficacia del crawler depende en gran medida de la calidad de las semillas escogidas para la inicialización de la aplicación.
- El tiempo de ejecución del crawler depende tanto de la implementación realizada como de la cantidad de semillas analizadas.
- La recuperación de la información es un elemento esencial ya que permite compartir la información generada en diversas plataformas web de la UTPL.
- Los ficheros CSV y RDF (XML) generados facilitan tanto la generación de las gráficas como la portabilidad de los resultados de las estadísticas obtenidas por la aplicación hacia otros entornos colaborativos.
- La utilización de RDF ayudó a cumplir con los objetivos planteados, en cuanto a la reutilización de la información generada por la aplicación, y permitirá la utilización y recuperación automática de los datos generados.
- La librería XMLWriter de PHP facilita enormemente la generación de archivos XML, en el presente trabajo se la adaptó para generar los archivos RDF necesarios para la estandarización de los resultados.
- La aplicación desarrollada permite conocer la cantidad de recursos Creative Commons desarrollados por la UTPL, así como determinar su evolución a través del tiempo indicando aquellos períodos de mayor o menor producción.
- La utilización del portal web permite conocer la cantidad de recursos Crea-

tive Commons desarrollados por otras instituciones, así como determinar su evolución a través del tiempo indicando aquellos periodos de mayor o menor producción.

- Mediante esta aplicación se puede conocer la cantidad de recursos Creative Commons desarrollados en el Ecuador, así como determinar su evolución a través del tiempo indicando aquellos periodos de mayor o menor producción.
- Este software permite realizar un análisis comparativo entre la cantidad de recursos Creative Commons UTPL y de otras instituciones, en un periodo de tiempo determinado, señalando las fortalezas y debilidades de la universidad.

Referencias

- [1] *Las Fuentes de Información*. Obtenido agosto 5, 2010 del sitio: <http://www.monografias.com/trabajos59/fichas-fuentes-informacion/fichas-fuentes-informacion.shtml>.
- [2] *Fuente documental*. Obtenido agosto 12, 2010 del sitio: http://es.wikipedia.org/wiki/Fuentes_de_informaci%C3%B3n.
- [3] JOHNSON James. (2000). *Bases de datos: Modelos, lenguajes, diseño*. Oxford University Press. 1028 p.
- [4] HANSEN Gary. (1997). *Diseño y administración de bases de datos*. Prentice Hall, 2da edición. 592 p.
- [5] SILBERSCHATZ A., KORTH H., SUDARSHAN S. (1998). *Fundamentos de bases de datos*. McGraw-Hill, Madrid, 3ra Edición, 641 p.
- [6] ULLMAN J., WIDOM J. (1999). *Introducción a los sistemas de bases de datos*. Prentice Hall, México, 1ra Edición, 488 p.
- [7] *Manual para licenciamiento de obras con licencias Creative Commons Ecuador 3.0 en la UTPL*. Obtenido julio 10, 2010 del sitio web de Creative Commons Ecuador <http://creativecommons.ec/>.
- [8] Fundación Copyleft. Obtenido agosto 15, 2010 del sitio: <http://fundacioncopyleft.org/>.
- [9] *Reconocimiento No Comercial Sin Obra Derivada 3.0 Ecuador*. Obtenido julio 12, 2010 del sitio web de Creative Commons <http://creativecommons.org/licenses/by-nc-nd/3.0/ec/>.
- [10] ARÓSTEGUI, Julio (2001). *La investigación histórica: teoría y método*. Barcelona.
- [11] TUÑÓN DE LARA, Manuel (1985). *Por qué la historia*. Barcelona: Aula Abierta Salvat.

- [12] JACOBSON Ivar, BOOCH Grady, RUMBAUGH James. (2005). *El Proceso Unificado de Desarrollo de Software*. Pearson Educación S. A., Madrid, 464p.
- [13] JACOBSON Ivar, BOOCH Grady, RUMBAUGH James. (2005). *El Lenguaje Unificado de Modelado*. Pearson Educación S. A., Madrid, 464p.
- [14] PENDER Thomas. (2004). *UML Weekend Crash Course*. Wiley Publishing, Inc., Indiana, Indianápolis, 358p.
- [15] Trabajo de grado especialización en gerencia de recursos humanos. Obtenido diciembre 10, 2010 del sitio web <http://www.gestiopolis.com/recursos5/docs/rrhh/matrizde.htm>.
- [16] iCommons. Obtenido diciembre 15, 2010 del sitio web <http://icommons.org/>.
- [17] Licencias libres para obras. Obtenido diciembre 15, 2010 del sitio web <https://merchandlinux.wordpress.com/2009/04/04/licencias-libres-para-obras/>.
- [18] Licencia Arte Libre. Obtenido diciembre 15, 2010 del sitio web <http://artlibre.org/licence/lal/es>.
- [19] Licencia Aire Incondicional. Obtenido diciembre 15, 2010 del sitio web http://www.platoniq.net/aireincodicional_licencia.html.
- [20] Vladislav Shkapenyuk, Tors-ten Suel (2002). *Design and Implementation of a High-Performance Distributed Web Crawler*. ICDE 2002: 357-368.
- [21] High-Performance Web Crawler. Obtenido diciembre 15, 2010 del sitio web <http://cis.poly.edu/polybot/>.
- [22] Aguilar Susana (2010). Gráficas elaboradas para la tesis “Desarrollo e implementación de una base de datos de los recursos licenciados con CC en Ecuador”.