UNIVERSIDAD
DE LOS ANDES
VENEZUELA

Universidad de Los Andes

Escuela de Ingeniería de Sistemas

Repositorios de Bajo Costo

por

Br. julio Jaimes Rodríguez

Tutor: Jacinto Dávila

2016

Departamento de Sistemas Computacionales

Repositorios de Bajo Costo

Br. julio Jaimes Rodríguez
Proyecto de grado, Escuela de Ingeniería de Sistemas, Departamento de Sistemas Computacionales, Universidad de Los Andes, 2016

Resumen: Se propuso un repositorio de bajo costo, el cual sea un almacen electrónico accesible vía internet. Entre los objetivos se extendió una herramienta ya desarrollada [3] CEA, para el manejo de información, el cual permite la búsqueda y recuperación de la información a través del catálogo electrónico ceaWeb.

Esta implementación fue de bajo costo con la finalidad de ser más liviano, es decir, con baja cohesión y un alto acoplamineto dependiente de menos herramientas externas, ya que el mismo software tiene su propio administrador de metadatos, no necesita de la instalación de un servidor externo para el manejo de toda la información. Esta implementación incorporó [7] proveer datos, así como también se necesitó probar su accesibilidad vía internet con un despliegue desde cualquier sitio web, y el reúso de herramientas existentes en [8] software libre.

Este software permitió crear repositorios desde casi cualquier computador en cualquier sitio a través de internet, ya que esto es de gran ayuda para los usuarios. Se facilitó esta metadata a travéz del protocolo sitemap, permitiendo asi mediante un archivo xml, en el cual se enumeraron todos los metadatos del software ceaWeb, para informar a Google y a otros motores de búsqueda sobre la organización del contenido del mismo y su difusión y cosechado.

Palabras clave: Repositorios, Acceso abierto, protocolo, sitemap, metadatos.

Este trabajo está dedicado a Dios,
mi papá Julio Jaimes y mi mamá Belén Rodríguez,
que aunque no está en este mundo desde
el cielo me cuidó de la mano de Dios.

Agradecimientos

Sinceramente son muchas las personas que me colaborarón con esta larga trayectoria de vida, lucha, constancia y dedicación de las cuales estoy inmensamente agradecido.

Primeramente le doy las gracias y honra a mi Dios todo poderoso, por permitirme llegar a esta meta y cumplir mi sueño, ya que en todo momento siempre estuvo guiando mis pasos para llevarme por el buen camino.

A mi mamá Belen Rodriguez, que desde el cielo me acompañó de la mano de Dios.

A mi papá Julio Jaimes pilar fundamental para mi vida, por su apoyo y esfuerzo para ayudarme en este sueño.

A mi hermano Jhon J. Jaimes ya que siempre me dió ánimos para seguir luchando.

A mi compañera de vida Yohana Terán por su amor, paciencia, consejos durante mis estudios.

Al profesor y tutor Jacinto Dávila por su paciencia, y excelente atención que me brindó día a día para hacer posible este trabajo.

A todos los profesores que me ayudaron en mi formación académica, en especial a los profesores Gerard Paez, Demián Gutierrez, Rodolfo Sumoza y Junior Altamiranda.

A la ilustre Universidad de Los Andes por abrir sus puertas y ser pilar durante mi formación.

A mi amigo Ronald Sulbaran por sus consejos y apoyo durante toda mi formación académica.

A mi amigo y hermano Jośe Rodriguez por su apoyo incondicional en todo momento ya que siempre estuvo cuando más necesité.

A mi amigo Luis Contreras, por su motivación a incentivarme a seguir mis estudios desde el inicio.

A mis amigos Roger Cuevas, Lenin Roa, Yosephaider Vanegas porqué compartimos momentos de lucha y dedicación durante esta formación.

Por último a todas aquellas personas que de una u otra manera me ayudaron aún sin darse cuenta.

Índice general

Capítulo 1

Introducción.

Para la realización de este proyecto, se propuso como objetivo desarrollar e implementar un repositorio de bajo costo, el cual fuese un almacen electrónico accesible vía internet. Se extendió una herramienta ya desarrollada [3] CEA, para el manejo de información, el cual permite la búsqueda y recuperación de la información a través del catálogo electrónico ceaWeb.

Hemos procurado un repositorio de bajo costo, en el sentido de que sea más liviano para la máquina que lo hospeda, teniendo de esta manera baja cohesión y un alto acoplamiento dependiente de menos herramientas externas, con su propio administrador de metadatos. Mediante esta implementación ha sido posible proveer datos, así como también probar su accesibilidad vía internet con un despliegue desde ULA-Mérida y el reúso de herramientas existentes en [8] software libre.

Este repositorio facilita el acceso a la metadata de los archivos que contiene, mediante un recurso (también archivo) xml que sirve para informar a los robot crawlers y a buscadores como Google sobre tales contenidos. Tambien se ofrece un servicio para que los usuarios puedan acceder a toda la metada desde el sistema web.

1.1. Antecedentes.

Se conoce que un repositorio [9] es un sitio o contenedor electrónico donde se pueden depositar, almacenar, gestionar, preservar y facilitar el acceso a los objetos digitales normalmente vía internet. Las iniciativas relacionadas a lo que hoy conocemos como "movimiento Open Access (OA)", se remontan a los años 60; pero no es sino hasta la década de los '90 con la aparición de la www que emergen proyectos como la creación de reconocidos repositorios tales como: Arxiv [http://arxiv.org/] en áreas relacionadas con la física, matemáticas y ciencias de la computación, creado en 1991 por Paul Ginspard.

El inicio del siglo XXI marcó una nueva etapa y la expansión del movimiento Open Access [9] (OA) empezó a ser exponencial. Y no sólo se realizan proyectos sino que hay un compromiso social avalado por declaraciones de ámbito internacional que sostienen y perfilan la definición de OA.

Las tres declaraciones más importantes han sido:

- Declaración de Budapest (BOAI, 2002)
- Declaración de Bethesda (2003)
- Declaración de Berlín (2003)

Este movimiento persigue el acceso gratuito mediante internet, sin barreras económicas, legales ni técnicas a la información científica. La declaración de la Budapest Open Access Initiative (BOAI 2002), sugirió dos estrategias para lograr el acceso abierto:

- La publicación de artículos en revistas de acceso abierto (ruta dorada).
- El depósito en repositorios institucionales de los artículos por parte de los autores, o sea el autoarchivo (Ruta Verde)

Los repositorios son sistemas de información que reúnen, preservan, divulgan y dan acceso a la producción intelectual de una comunidad, contribuyendo a aumentar su visibilidad y promoviendo la divulgación de los resultados de su actividad. En la búsqueda y recuperación de información existió un variado número de mecanismos para la generación y obtención de metadatos en internet. El más utilizado es el

protocolo OAI-PMH (Open Archives Iniciative Protocol for Metadata Harvesting), ya que sirve para almacenar y obtener cualquier tipo de información que se encuentre en cualquier formato electrónico.

El protocolo OAI-PMH [7] proporciona una sencilla interfaz que permitió el acceso a los metadatos de contenidos en formato XML proveniente de distintas fuentes, plataformas y repositorios. Con el avance del protocolo OAI, se ofrecierón herramientas "técnicas y se pusó de manifiesto que la disponibilidad libre de documentos en línea facilitó el acceso en múltiples modos, desde la obtención de los archivos, pasando por la conexión directa entre los científicos, además de la creación de grupos de trabajo y de discusión, la indexación por medio de motores de búsqueda en internet y la creación de nuevos servicios tecnológicos".

Este protocolo generó y promovió estándares de interoperabilidad que facilitarón la difusión, intercambio y accesibilidad a documentos de diferente naturaleza. Además, OAI – PMH [5] permitió almacenar en un solo lugar los metadatos y es allí en donde se realizan las diferentes consultas, el protocolo no definió la creación de los metadatos, ni da los parámetros para realizar una consulta, únicamente se ocupó de la gestión de información. El protocolo de cosecha de metadatos utilizó transacciones HTTP (Hyper Text Transfer Protocol) usado en la transferencia de información de contenido Web, en donde está definida la sintaxis y la semántica que utilizan los clientes y servidores para comunicarse entre sí.

El protocolo OAI [11] está basado en un modelo cliente/servidor que transmite preguntas y respuestas entre un proveedor de datos, y un proveedor de servicios. Un sencillo ejemplo es la búsqueda que un usuario realiza en un servidor Web, el usuario envía una petición a un proveedor de servicios, el cual solicita a un proveedor de datos que le envié registros de metadatos de diferentes recursos con que este disponga.

El proveedor de datos envía la respuesta al proveedor de servicios, como un conjunto de registro de metadatos en formato XML, el cual mostrará los resultados al usuario, por medio de una interfaz. La ventaja de la búsqueda es que el usuario selecciona los documentos de su interés a través de los registros de los metadatos que los describen.

En sí, el usuario no utiliza el protocolo OAI-PMH [2], los encargados de utilizarlo para comunicarse son los proveedores de datos y los proveedores de servicios, el

usuario tendrá contacto con el proveedor de servicios, el que estará encargado de resolver las necesidades de información que el usuario tenga. Es allí donde dicho proveedor de servicios tendrá que hacer contacto por medio del protocolo OAI-PMH, a uno o varios proveedores de datos, los cuales disponen de la información que el usuario necesita.

Caracteristicas del protocolo OAI.

- Su funcionamiento se basa en una arquitectura cliente-servidor en la que un servicio recolector de metadatos pide información a un proveedor de datos.
- Las peticiones se expresan en HTTP, utilizando únicamente los métodos GET o POST.
- Fechas y tiempo se codifican mediante la ISO 8601 y se expresan en UTC.
- Soporta la difusión de registros en diversos formatos de metadatos.
- Tiene control de flujo.
- Cuando hay un error o una excepción los repositorios deben indicarlos distinguiéndolos de los códigos de estado HTTP por incluir uno o más elementos de error en la respuesta.

Los metadatos [6] tienen sus raíces en el catálogo, probablemente inventado poco después del comienzo de la historia por parte de los sumerios. A lo largo de los siglos las tabletas de arcilla utilizadas evolucionaron hasta listas manuscritas y posteriormente a catálogos de libros después de la invención de la imprenta. Estos primeros catálogos de libros eran impresos y eran listas ordenadas alfabéticamente sin criterios de clasificación sofisticados.

Un avance importante en cuanto a esquemas de clasificación se desarrolla alrededor del 1900 cuando los catálogos de libros son reemplazados completamente por tarjetas, las cuales entre otras cosas pueden ser actualizadas. En la década del sesenta los métodos de producción en masa (como los computadores) hacen necesario disponer de múltiples copias de los catálogos existentes, surgen masivas colecciones distribuidas de libros y los catálogos de tarjetas no logran satisfacer los nuevos

requerimientos. Es necesario entonces desarrollar una referencia que explica un determinado dato, llamados hoy en día metadatos.

Un sistema de educación abierta es aquel en el cual los controles sobre los estudiantes se revisan continuamente y se eliminan cuando sea necesario. Se utiliza una gran variedad de estrategias pedagógicas, especialmente las empleadas en el aprendizaje independiente e individual, según Coffey en el año 1977. Tomando en cuenta todos los antecedentes mencionados sugió el problema de este trabajo, el cual se divide en varios subproblemas específicos para llevar a cabo el desarrollo del mismo.

1.2. Planteamiento del problema.

Un repositorio es un almacén electrónico, normalmente accesible vía internet, como el que se desarrolló en este trabajo de tesis. Se propone aprovechar una herramienta ya desarrollada, el catálogo electrónico [3] CEA, para permitir la búsqueda y recuperación de documentos desde cualquier parte del mundo, y así poder crear repositorios de bajo costo y agregar nuevas funcionalidades a esta herramienta. Este tipo de repositorio está dirigido a organizaciones o individuos que no movilicen gigantescos volúmenes de datos pero, no obstante, quieran ofrecer los servicios modularmente, de un repositorio estándar basado en sitemaps.

1.3. Justificación.

Nos hemos dado cuenta que la información es un recurso necesario y vital para mejorar la educación y la investigación, se incentiva el aprendizaje de los individuos dentro de las organizaciones y fuera de ellas, también ayuda a contribuir a elevar los conocimientos y por ende ayuda a que tomen mejores decisiones, planteen soluciones y propuestas innovadoras que conllevan al desarrollo y éxito.

El internet es una de las redes fundamentales para propagar está información, pero muchas veces su acceso es limitado y en algunos casos costoso, por esta razón se propone desarrollar un sistema web que nos permita almacenar, distribuir y gestionar todo tipo de información, teniendo en cuenta los aspectos y caracteristicas fundamentales de las máquinas, para que en ellas se pueda utilizar dicho sistema.

1.4. Alcance.

El proyecto termina con el diseño e implementación de un sistema web para extender la herramienta existente Catalago Elecrtonico Autonomo CEA. Se desarrolló la termerización de html e implementación de todos los módulos los cuales están compuestos por toda la arquitectura, para así llevar a cavo esté software. El sistema se puede instalar en cualquier máquina de bajos recursos, el proceso de instalación es muy sencillo y no requiere de conocimientos específicos o complicados. El usuario puede administrar y gestionar su información, asi como también todo el sistema es software libre, permitiendo de esta manera contribuir con la educación y el desarrollo de herramientas. El protocolo implementado está basado en sitemap el cual es de alto rendimiento y amigable a la web.

1.5. Objetivos.

1.5.1. Objetivo General.

Diseñar e implementar un repositorio de bajo costo desde cualquier sitio, basado en un sistema al que llamaremos ceaWeb, el cual maneje datos en formato pdf, permitiendo el acceso a los mismos desde cualquier parte del mundo, vía internet y así contribuir con la educación abierta.

1.5.2. Objetivos Específicos.

- Evaluar la herramienta preexistente CEA.
- Evaluar la plataforma Prolog para acceso Web y el despliegue basado en html termerizado.
- Diseñar la arquitectura del Sistema ceaWeb.
- Desarrollar los componentes de ceaWeb.
- Probar el despliegue de ceaWeb.

- Implementar la difusión de metadatos del repositorio como un archivo xml basado en el protocolo sitemap.
- Dar funcionalidad como proveedor de metadatos amigable a la web.

1.6. Metodología.

1.6.1. Iterativa e incremental.

Para el desarrollo de esta tesis se usaron métodos los cuales se fundamentaron en el desarrollo ágil de software, ya que permitió un enfoque para la toma de decisiones sobre el proyecto a desarrollar, el cual se refiere a uno o más métodos de ingeniería del software basados en el desarrollo iterativo e incremental, donde los requisitos y soluciones para el proyecto evolucionaron con el tiempo según la necesidad del mismo. Permitiendo así que el trabajo fuese realizado mediante la ayuda del profesor tutor Jacinto Dávila y Julio Jaimes R, los cuales estuvieron íntimamente relacionados con los procesos para así proveer soluciones y toma de decisiones a corto plazo.

Se implementaron iteraciones las cuales estuvieron acompañadas de una planificación, análisis de requisitos, diseño, codificación, documentación y pruebas. Esto nos permitió incrementar el valor del desarrollo del proyecto a medida que se fue avanzando en las iteraciones. Para solucionar el problema principal se dividieron en distintos bloques temporales que se le denominaron iteraciones. Lo que se buscó es que en cada iteración los componentes lograrán evolucionar al proyecto dependiendo de los objetivos completados de las iteraciones antecesoras, agregando más opciones de requisitos y logrando así un mejoramiento mucho más completo.

1.6.2. Ventajas de la metodología iterativa e incremental.

- Permitió la retroalimentación con los usuarios de forma inmediata.
- Permitió separar la complejidad del proyecto, gracias a su desarrollo por parte de cada iteración o bloque.
- Se llevó a cabo un proyecto puntual y consistente en el desarrollo.

- El proyecto tuvo una menor probabilidad de fallar ya que se evaluó a medida que se desarrolló.
- Después de cada iteración se obtuvo un aprendizaje para ser aplicado en el desarrollo del proyecto y así aumentar la experiencia.

1.7. Plan de Trabajo

Este plan se dividió estructuró en 6 fases de desarrollo, ya que nos permitió facilitar el proceso de desarrollo, se tomaron en cuenta los recursos, fechas, relaciones de dependencias entre cada tarea y tiempos para su ejecución. Definición de las 6 fases de desarrollo.

- Fase 1. Recolección de información: se hizó un estudio de la información relacionada con el desarrollo de este proyecto y las herramientas existentes.
- Fase 2. Prácticas y estudios: se realizó un curso previo de Prolog, las capacidades del protocolo sitemap y un análisis y estudio sobre la herramienta CEA.
- Fase 3. Almacenamiento de metadatos e indexación: se utilizó metadata de Educere y se implementó la herramienta swish-e para la indexación.
- Fase 4. Diseño e implementación: se realizaron los diseños de la arquitectura y a su vez la implementación de la misma, el despliegue experimental desde ULA-Mérida, la difusión y cosechado ligero.
- Fase 5. Pruebas del Proyecto: se realizaron las pruebas internas del sistema, su accesibilidad desde internet y todas las pruebas finales.
- Fase 6. Preparación y entrega final: se realizaron las últimas correcciones para la preparación y la entrega final del proyecto.

1.8. Estructura del documento

Capítulo I. Introducción.

En esta sección se dan a conocer las características del problema que se desarrolla en este proyecto, se realiza un estudio y análisis de los antecedentes, concretanto los objetivos y estableciendo la metodología con que se realizó todo el estudio para dar solución a este proyecto.

Capítulo II. Marco teórico-funcional.

En esta sección se describen conceptos relacionados con el proyecto que se desarrollará. También se exponen las herramientas utilizadas para llevar a cabo el mismo.

Capítulo III. ceaWeb: Catálogo Electrónico Autónomo web..

Se describe todo el diseño en general del sistema ceaWeb, así como la arquitectura del sistema web, sus componentes, relaciones, características y funcionalidades del software.

Capítulo IV. Difusión y cosechado de metadata, una propuesta amigable a la web..

Contiene la definición final arquitectura y su implementación, así como el desarrollo e integración con el protocolo sitemap.

Capítulo V. Pruebas del sistema ceaWeb.

En esta sección se exponen las distintas pruebas realizadas a la aplicación ceaWeb, definición, implementación y el resultado, as'ı mismo se describe cada una de ellas.

Capítulo VI. Conclusiones y Recomendaciones.

Conclusiones y Recomendaciones, contiene las conclusiones obtenidas a partir del diseño e implementación del sistema, así como recomendaciones y posibles mejoras e implementaciones futuras.

Capítulo 2

Marco Teórico-Funcional.

En esta investigación la teoría constituyó la base donde se sustentarón los análisis, experimentos o propuestas de desarrollo de este trabajo de grado. Es por esto que se mencionó un estudio de conceptos y nociones sobre los repositorios, y a su vez de las bases fundamentales en la herramienta Catálogo Electrónico Autónomo CEA y su funcionamiento.

2.1. Repositorio.

Es un sistema de información que reúne, preserva, divulga y da acceso a la producción intelectual de las comunidades, permitiendo asi contribuir al aumento de la visibilidad de la información, promoviendo la divulgación de los resultados de su actividad, preservar su memoria intelectual y facilitar su gestión de información.

2.2. Open Access (OA)

Este movimiento persigue el acceso abierto mediante internet, sin barreras económicas, legales ni técnicas a la información científica.

2.3. Open Archive Initiative (OAI).

Es una organización para desarrollar y aplicar las normas de interoperabilidad técnica de archivos para compartir información de catálogo (metadatos). Se trata de construir una barrera baja de interoperabilidad, para los archivos y repositorios institucionales, que contienen los datos digitales. Permite a las personas proveedores de servicios, la extracción de metadatos de los proveedores de datos. Estos metadatos se utilizan para proporcionar servicios de valor añadido, a menudo mediante la combinación de diferentes conjuntos de datos.

La interoperabilidad de los repositorios recibe un fuerte impulso con la Open Archive Initiative (OAI).

Las tres declaraciones más importantes sobre OAI son:

- Declaración de Budapest (BOAI, 2002)
- Declaración de Bethesda (2003)
- Declaración de Berlín (2003)

2.4. La declaración de la Budapest Open Access Initiative (BOAI) (2002).

Esta delaración sugirió dos estrategias para lograr el acceso abierto:

1. La publicación de artículos en revistas de acceso abierto (Ruta Dorada).
2. El depósito en repositorios de los artículos por parte de los autores, o sea el autoarchivo (Ruta Verde).

2.5. ¿Por qué es importante la Educación Abierta?.

Vivimos en un mundo en donde las personas quieren aprender. Al proporcionar acceso libre a la educación y el conocimiento, ayudamos a crear un mundo donde la gente puede cumplir este deseo. Los estudiantes pueden obtener información

adicional, puntos de vista y materiales para ayudarles a tener éxito. Los trabajadores pueden aprender cosas que les ayudarán en el trabajo. Los profesores pueden aprovechar los recursos de todo el mundo, encontrar nuevas maneras de ayudar a los estudiantes a aprender. Los investigadores pueden compartir datos y desarrollar nuevas redes.

La gente puede conectarse con otros que de otro modo no se reunirían para compartir ideas e información. Los materiales pueden ser traducidos, mezclados, separados y compartidos otra vez en abierto, incrementando así el acceso y promoviendo nuevos enfoques. Cualquier persona puede acceder a los materiales educativos, artículos académicos y las comunidades de aprendizaje de apoyo en cualquier momento que lo deseen. La educación es entonces accesible, disponible, modificable y libre.

2.6. Metadatos.

Son datos altamente estructurados que describen información, describen el contenido, la calidad, la condición y otras características de los datos. Es ̈Información sobre información.º "datos sobre los datos". Algunos ejemplos de información que se puede describir usando metadatos son: impresa, audiovisual, geoespacial, digital, etc.

2.7. Estructura de los metadatos

Los metadatos están estructurados por un mínimo de elementos tales como: título, autor, fecha de creación, etc. Típicamente, los elementos que conforman un metadato están definidos por algún estándar, donde los usuarios que deseen compartir metadatos están de acuerdo con un significado preciso de cada elemento.

2.8. Estándares y su importancia de los metadatos.

Los estándares se han definido para determinar qué debe documentarse de la base de datos, proveen una terminología común y un conjunto de definiciones para

la documentación de los datos geoespaciales. Los estándares del Comité Federal de Datos Geográficos (FGDC) de los Estados Unidos recomienda documentar los siguientes elementos de cada base de datos:

- Título: Nombre del conjunto de datos o del mapa/imagen.
- Área geográfica: Cobertura espacial de la base de datos.
- Descripción de los datos: Resumen que indica el propósito o uso para el cual fue elaborado el set de datos.
- Temporalidad de los datos: Fecha en que fue elaborado el set de datos.
- Normas para obtener y utilizar los datos: Indicar como se puede obtener una copia de la base de datos y cuáles son las condiciones que regulan su uso.
- Contacto: Dirección física y electrónica de la persona que puede proveer acceso a los datos, incluyendo horas de oficina.
- Fecha y nombre de la persona que elaboró los metadatos. Indicar la fecha y la persona responsable por elaborar la descripción del set de datos.
- Información sobre la calidad de los datos.

2.9. Herramienta swish-e.

En las unidades de información se presenta el reto de minimzar la brecha digital que ha supeditado a los países en vía de deasarrollo a través de la recuperación de contenidos digitales en la Web, para ello se hace necesaria la vinculación de herramientas tecnológicas que faciliten la administración y la recuperación de estos contenidos de una forma accesible a todas la comunidades. Por esta razón es ineludible la disposición de software libre que permita la realización de dichas labores enfocadas a la preservación y disposición del contenido en una civilización. Se presenta entonces, como opción a esta problemática el sistema de indexación [10] Swish-e como alternativa de búsqueda y recuperación de contenidos completos.

Desarrollado por Kevin Hughes, es un sistema de indexación Simple Web para los seres humanos. Se utiliza para las colecciones de índice de documentos que van hasta un millón de documentos en tamaño e incluye filtros de importación para muchos tipos de documentos.

Roy Tennant (entonces en la Universidad de California, Berkeley Library) pidió a mediados de los años 1990 asumir la responsabilidad de desarrollar aún más esta herramienta de indexación web, en su actualidad es mantenido por equipos voluntarios.

2.10. Ventajas y Desventajas de Swish-e

Ventajas: Entre las principales ventajas al utilizar Swish-e encontramos:

- Se puede indexar cualquier tipo de archivo que contenga texto.
- Es licencia de software libre, permite su portabilidad a cualquier sistema operativo, para los cuales se encuentran. múltiples versiones en el sitio web de Swish-e.
- Es una herramienta muy rápida en Indexación y Búsqueda.
- Permite indizar colecciones enormes de documentos.

Desventajas: las principales desventajas al utilizar el programa Swish-e se encontraron las siguientes:

- No es un sistema de Indexación llave en mano.
- No permite la indexación optima de archivos semi-estructurados que no manejen las marcas de HTML/XML.
- No tiene interfaz para usuarios no expertos.

2.11. Catálogo Electrónico Autónomo.

Es el software para gestionar una colección electrónica de documentos sobre el que se desarrolla esta propuesta. Permite la recuperación de información para usuarios

autónomos, estos datos son colecciones de archivos en formato pdf de revistas y periódicos.

2.11.1. ¿Por qué un repositorio CEA?.

Principalmente porque:

- Acceso público.
- Permite el acceso abierto a documentos en formato pdf.
- Preservación de información digital.

2.12. Educere.

Es la revista venezolana de educación de la cual se tomó los metadatos y datos en formato pdfs para el sistema ceaWeb.

Figura 2.1: Revistas de Educere.

2.13. Arquitectura de cea.

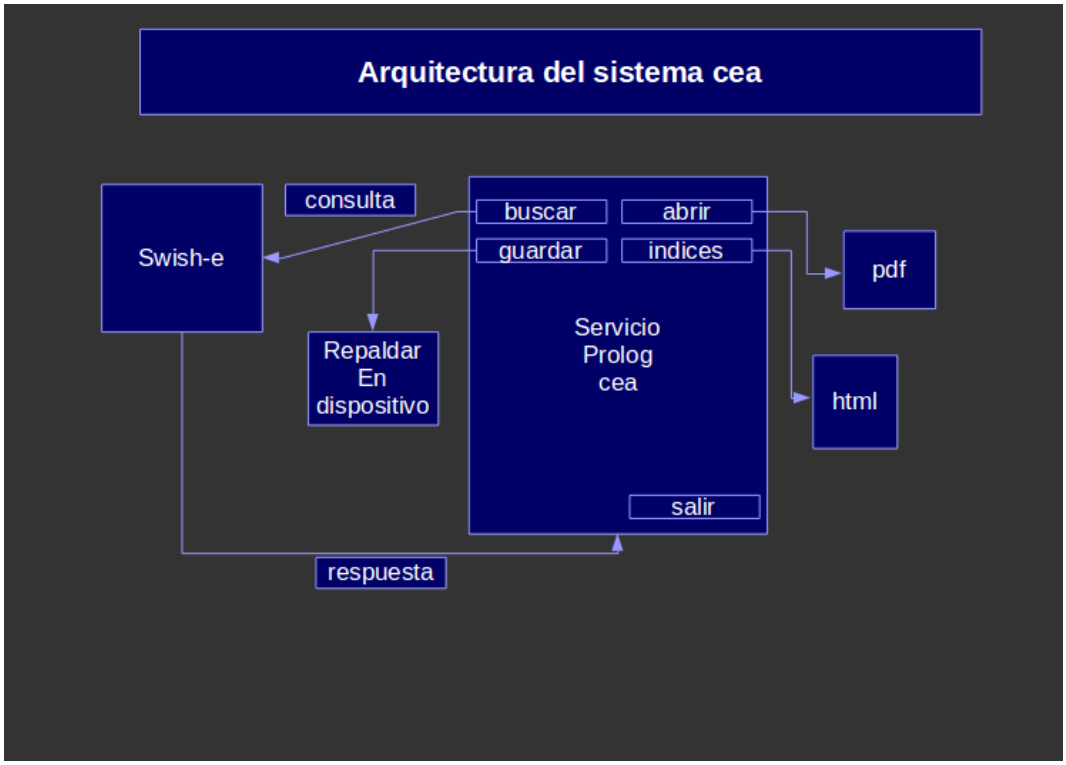

Figura 2.2: Arquitectura de la herramienta cea.

2.13.1. Descripción.

Esta arquitectura consiste en: el cliente realiza una consulta al sistema, ésta es enviada y recibida por el indexador swish-e, para así procesarla mediante ésta herramienta y generar una respuesta al cliente, la cual se mostrará en un módulo de resultados.

Estos resultados contienen todos los urls relacionados a la petición, el cliente puede abrir, guardar y cerrar estos documentos pdfs. También tiene un módulo de indices el cual consiste en generar un indice de autores, títulos y temas sobre toda la metadata almacenada mediante una pagina en html.

2.14. Funcionamiento de la herramienta CEA.

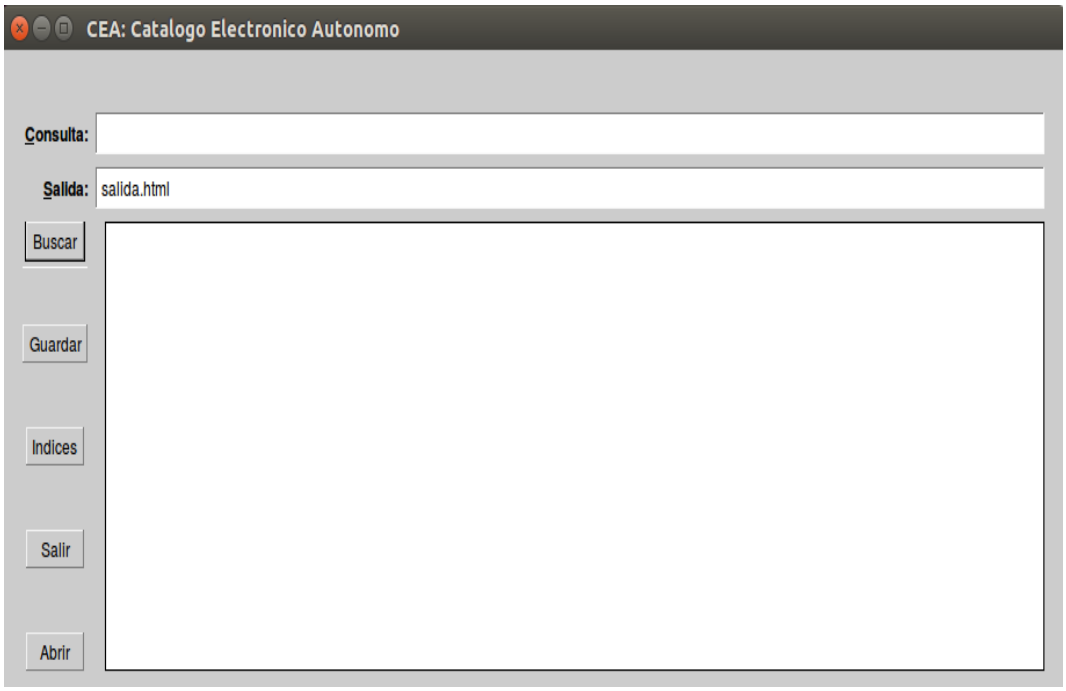

Figura 2.3: Interfaz inicial de cea.

- Inicio. Cuenta con una interfaz de escritorio amigable al usuario la cual consta de varios items los mismos nos permitiran administar el repositorio.
- Buscar: desde aquí puede ingresar a lo que desea y así conseguir tus resultados de manera instantanea.
- Seleccionar. Podrá seleccionar cualquier documento de búsqueda.
- Abrir. Se mostrará el documento pdf.
- Guardar. Permite guardar documentos de búsqueda en donde tu elijas.
- Salir. Cerrar el sistema CEA.

Todos estos conceptos fundamentales y herramientas funcionales, fueron necesarios para la realización de este proyecto; en especial la herramienta Catálogo Electrónico Autónomo cea, la cuel es base fundamental en el desarrollo del mismo. En el siguiente capítulo se presentará el trabajo propiamente dicho.

Capítulo 3

ceaWeb: Catálogo Electrónico Autónomo web.

3.1. CeaWeb.

Es una extención del software existente [3] CEA, cuya implementación está basada en la web y su despliegue podría ser desde cualquier sitio, su información y sus códigos son software libre. Esta plataforma está comprometida con el acceso abierto.

3.2. Caracteristicas ceaWeb.

- Implementación dinámica.
- Datos livianos, es dicir no son pesados.
- Gestionar los datos (ver, guardar, imprimir).
- Acceso vía internet.
- Colaborar con la educación abierta.
- Generador de archivos sitemap.
- Servicio sitemap de todo el sistema.

3.3. ¿Por qué Repositorios de bajo costo?.

Es un sistema con alta cohesión y bajo acoplamiento en sus módulos, es liviano y no dependió de otros sistemas externos, debido a esto se comportó con un alto rendimiento y eficiencia. Este sistema es fácil de instalar en cualquier máquina personal económicas o de cierta generación antigua, ya que no consume muchos recursos en su ejecución e instalación, tiene la capacidad de manejar archivos ligeros permitiendo acceder a ellos desde cualquier parte haciendo uso del internet.

Todo el sistema está basado en software libre, permitiendo de esta manera a cualquier persona poder descargarlo del repositorio en github sin ningún costo alguno. De esta forma podrás adaptarlo para el uso propio y seguir realizando mejoras según la necesidad de cada desarrollador con toda libertad, ya que también el sistema no tiene dependencias privativas, es decir, el software de soporte es software libre.

El sistema ceaWeb se puede adaptar según la necesidad de cada usuario sin la necesidad de realizar la instalación de un servidor externo, el cual en la mayoría de los casos es complicado y requiere de caracteristicas muy costosas.

El sistema web está implementado bajo la termerización del html, esto permite que la página se construyá dinámicamente cada vez que se le invoque. En lugar de tener una página web con código imbuido, es código Prolog con el html allí representado. El módulo basado en sitemap, permite que la metadata sea encontrada y rastreada por los buscadores.

3.4. Arquitectura del sistema ceaWeb.

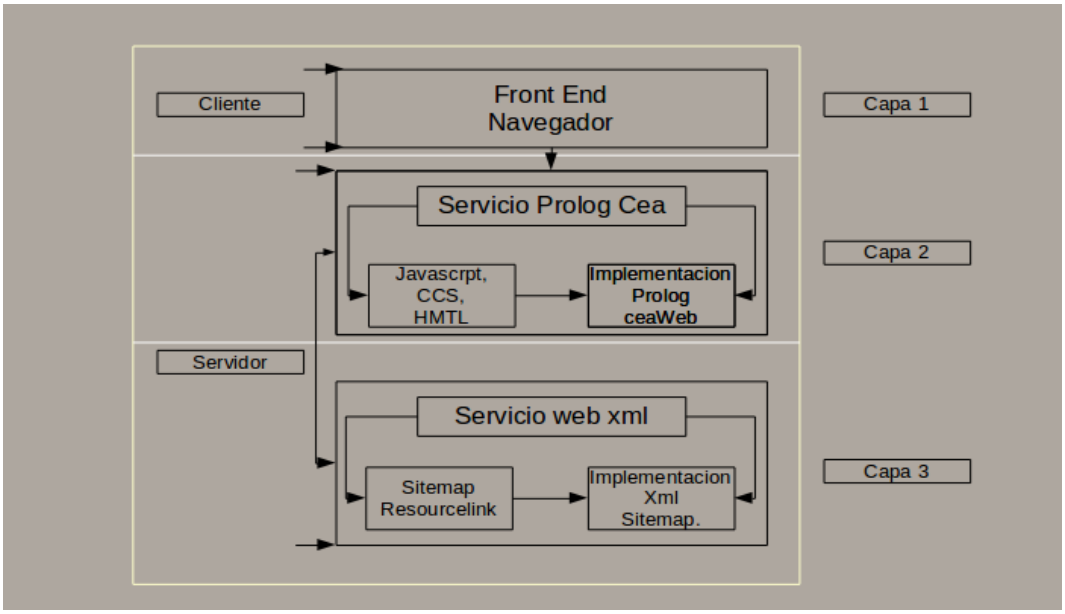

Figura 3.1: Diseño Arquitectónico ceaWeb.

El sistema está estructurado en una arquitectura cliente servidor, la cual consta de tres, la capa Front End se encuentra relacionada con la presentación a través del navegador ceaWeb, por parte del cliente; la capa Servicio Prolog Cea, esta consta de la implementación del sistema, la cual está relacionada con las herramientas Javascript, CCS y html; la ultima capa es la que provee el Servicio web xml, se relaciona con la generación del sitemap y su implementación; todas estas se describen a continuación:

3.4.1. Capa 1: Front End, Navegador.

Esta capa se basa en la interfaz gráfica del sistema ceaWeb. Ella permite la interacción del usuario con el sistema Catálogo Electrónico Autónomo web ceaWeb. La misma consiste en generar todas las vistas a través de los navegadores web, para así por medio de ellas el usuario pueda realizar todas sus consultas y peticiones al servidor.

3.4.2. Capa 2: Servicio Prolog Cea.

Está basado en la implementación mediante Prolog, este módulo es el encargado de procesar la búsqueda este recibe la consulta la cual es enviada por el cliente, la procesa y luego invoca a la plataforma previamente configurada swish-e, y a partir de ahí genera los resultados ya ordenados y con su respecivos urls asociados a los pdfs, almacenados en el repositorio.

3.4.3. Capa 2: Javascript, Cea, Navegador html.

El sistema está implemtado mediante html termerizado, también utiliza Javascript en sus módulos, cuando el sistema es iniciado, se escribe en tiempo de compilación la página inicial del sistema por medio de la termerización. Los resultados del sistema son reflejados en este módulo.

3.4.4. Capa 2: Implementación Prolog ceaWeb.

Entre los módulos más importantes tenemos: buscar, guardar, abrir, cerrar, indices, contactos y la información asociada a todos los metados del repositorio. Es importante resaltar que los indices de autores, títulos y temas pueden ser creados e indexados cada vez que el usuario lo desee ya que el sistema, tiene un módulo para generarlos, esto es importante para cuando el sistema sea actualizado.

La otra mitad de la arquitectura es la capa 3 del sistema ceaWeb, la misma se describirá en el siguiente capítulo, pero consiste en el protocolo sitema el cual es amigable a la web y permite que la metadata del repositorio sea rastreada y monitoreada por los buscadores web.

3.5. Funcionamiento de ceaWeb.

Su funcionamiento inicia mediante una página web, la cual es creada a través de Prolog con la termerización del html, entre las funcionalidades más destacadas tenemos:

- Inicio.
- Buscar.
- Resultados.
- Seleccionar.
- Abrir.
- Guardar.
- Imprimir.

3.5.1. Inicio.

El sistema arranca y muestra su interfaz inicial, en donde se despliega una barra de navegación, ésta nos permite poner en funcionamiento el sistema web, según la necesidad del usuario.

3.5.2. Buscar.

Este funcionamiento, se activa cuando el usuario inserta una consulta, esta es enviada a swish-e para ser procesada y ejecutada.

3.5.3. Resultados, seleccionar, abrir, guardar e imprimir.

El sistema genera y muestra los resultados en un módulo en donde permite, seleccionar, abrir, imprimir y guardar los resultados de la búsqueda asociada. También es posible regresar al modulo de buscar para volver a realizar otra consulta si el usuario así lo desea.

3.6. Indexación de los archivos de ceaWeb.

En el software ceaWeb consiste en analizar los documentos pdfs, ordenar y extraer toda la información necesaria que permita crear la base de datos de índices. Y así poder facilitar su consulta y análisis desde el buscador de ceaWeb, utilizando la herramienta swish-e, a continuación se describen los pasos para indexar los archivos.

3.7. Pasos para indexar los archivos:

1. Desde el archivo donde desea indexar, cree el siguiente archivo llamado pdfs.conf

2. Copie, pegue y guarde en el archivo pdfs.conf lo siguiente:

   # swish-e -S fs -c pdfs.conf
   IndexDir ./pdfs
   FileFilterMatch pdftotext
   "WordCharacters äöüéèàabcdefghijklmnopqrstuvwxyz0123456789.-
   IgnoreFirstChar .-
   IgnoreLastChar .-
   BeginCharacters
   äöüéèàabcdefghijklmnopqrstuvwxyz0123456789
   EndCharacters
   äöüéèàabcdefghijklmnopqrstuvwxyz0123456789
   MinWordLimit 2

   Nota: Usted puede cambiar este archivo según sea su necesidad.

3. Luego ejecute el siguiente comando desde el directorio en donde tiene el archivo pdfs.conf:

   swish-e -S fs -c pdfs.conf

   Ya con esto habrá indexado los archivos.

   Para realizar la búsqueda desde la consola inserte el siguiente comando:

   swish-e -w "la frase a buscar"

Observaciones:

Hasta este punto en este documento, sólo está definida la mitad del sistema e incluyendo su arquitectura. La difusión y cosechado de ceaWeb es parte de esta otra mitad, la cual está basada en el protocolo sitemap, y se describe más adelante en el capitulo 4.

Capítulo 4

Difusión y cosechado de metadata, una propuesta amigable a la web.

4.1. ¿Qué es un sitemap?.

Un sitemap [4] es un archivo en el que se pueden enumerar las páginas de tu sitio web para informar a Google y a otros motores de búsqueda sobre la organización del contenido del mismo. Los rastreadores web de los motores de búsqueda, por ejemplo, el robot de Google, leen este archivo para rastrear el sitio de forma más inteligente.

Además, tu sitemap te puede proporcionar valiosos metadatos asociados a las páginas que enumeras en él. Los metadatos son información sobre una página web, como, por ejemplo, cuándo se ha actualizado por última vez, con qué frecuencia se cambia y la importancia de esta en relación con otras URL del sitio web.

También los sitemaps [1] son una forma fácil que tienen los webmasters para informar a los motores de búsqueda de las páginas que se pueden rastrear en sus sitios web. Un Sitemap, en su forma más sencilla, es un archivo XML que enumera las URL de un sitio junto con metadatos adicionales acerca de cada una de ellas: la última actualización, frecuencia de modificación, importancia, en relación con las demás URL del sitio; así, los motores de búsqueda pueden llevar a cabo rastreos del sitio de una forma más inteligente.

Los rastreadores web suelen encontrar páginas a partir de vínculos del sitio y a partir de otros sitios. Sitemaps ofrece estos datos para que los rastreadores compati-

bles puedan seleccionar todas las URL del sitemap y obtengan información de ellas mediante los metadatos asociados. El uso del protocolo sitemaps no garantiza que las páginas web se incluyan en los motores de búsqueda, pero proporciona sugerencias para mejorar el trabajo de los rastreadores web al rastrear su sitio.

4.2. Etiquetas obligatorias del sitemap ceaWeb.

Para el archivo sitemap se tomó en cuenta las siguientes especificaciones:

- Comenzó con una etiqueta de apertura <urlset> y terminó con una de cierre </urlset>.
- Se especificó el espacio de nombres (protocolo estándar) en la etiqueta urlset.
- Se incluyó una entrada <url> para cada dirección URL como una etiqueta XML principal.
- Se incluyó unas entrada secundarias la cuales son: <title>,<author>,<subtitle>, <lastmod> y <loc> para cada etiqueta principal <url>.

4.2.1. Carácteres de escape de entidad sitemap.

Carácter		Código de caracteres de escape
Símbolo de unión	&	&amp;
Comillas simples	'	&apos;
Comillas	"	&quot;
Mayor que	>	&gt;
Menor que	<	&lt;

Figura 4.1: Carácteres de escape de entidad tomado de [1]

4.3. Protocolo del sitemap para weaWeb.

El formato del protocolo Sitemap para nuestro ceaWeb constó de etiquetas XML. Todos los valores incluyeron caracteres de escape de entidad. El archivo se codificó en formato UTF-8.

4.4. Capa 3: Servicio web xml.

A continuación se describirá la última capa de la arquitectura del sistema ceaWeb, esta se estructura por todos los módulos del protocolo sitemap y la generacion del xml. Permitiendo de esta manera la difusión y cosechado de toda la metadata en nuestro servidor.

4.4.1. Formato del sitemap para ceaWeb.

```
<?xml version="1.0" encoding="UTF-8"?>
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">

<url>
    <title> </title>
    <author> </author>
    <subtitle> </subtitle>
    <lastmod> </lastmod>
    <loc> </loc>
</url>

</urlset>
```

Figura 4.2: Arquitectura

4.4.2. Definición de las etiquetas usadas.

4.4.3. title:

Esta contiene el título de cada documento almacenado en el repositorio.

4.4.4. author:

Posee todos los autores involucrados en el título de cada documento pdf.

4.4.5. subtitle:

Algunos temas contienen subtítulos los cuales se describen en esta etiqueta.

4.4.6. lastmod:

Esta contiene la fecha de modificación de la metadata al cual está relacionada.

4.4.7. loc:

Es la más importante ya que contiene la ruta exacta de la metadata la cual es usada para que los motores de búsqueda logren proporcionar a otros usuarios.

4.4.8. Creación de un archivo robots.txt para mejorar la difusión y cosechado.

Una vez que se creó el archivo sitemap.xml se especificó la ubicación del Sitemap utilizando un archivo robots.txt. Para ello, tan solo se añadió la línea siguiente:

Sitemap: http//localhost:5000/f/indices/sitemap.xml

Para informar a los motores de búsqueda compatibles con el protocolo acerca de su ubicación y así permitir la difusión y cosechado.

4.4.9. Creación del módulo XSL para proveer el sitemap xml de ceaWeb.

Se creó una funcionalidad mediante un archivo xsl para proveer a los motores de búsqueda el sitemap xml, el cual es una forma de mostrar toda la metadata a los humanos. Esto se hizo mediante una lectura de todos los metadatos de nuestro ceaWeb para asi proporcionar un entorno amigable al usuario. Permitiendo de esta manera la difusión y cosechado de metadata a nivel mundial mediante el servicio sitemap del sistema ceaWeb.

Todos estos elementos explicados formaron parte de todo el desarrollo de la mitad del sistema ceaWeb, estos estuvieron relacionados con toda la difusión y consechado de la información del sistema. El protocolo sitemap nos generó resultados muy eficientes y una interfaz amigable al usuario; en el siguiente capítulo se describirá todas las pruebas realizadas con sus respectivos análisis.

Capítulo 5

Pruebas del sistema ceaWeb.

5.1. Inicio.

En este momento la herramienta ceaWeb se inicia mediante el puerto 5000.

Figura 5.1: Inicio del sistema ceaWeb.

5.2. Buscar en ceaWeb.

El usuario ingresa una consulta en el sistema ceaWeb mediante la pestaña buscar como se observa en la figura.

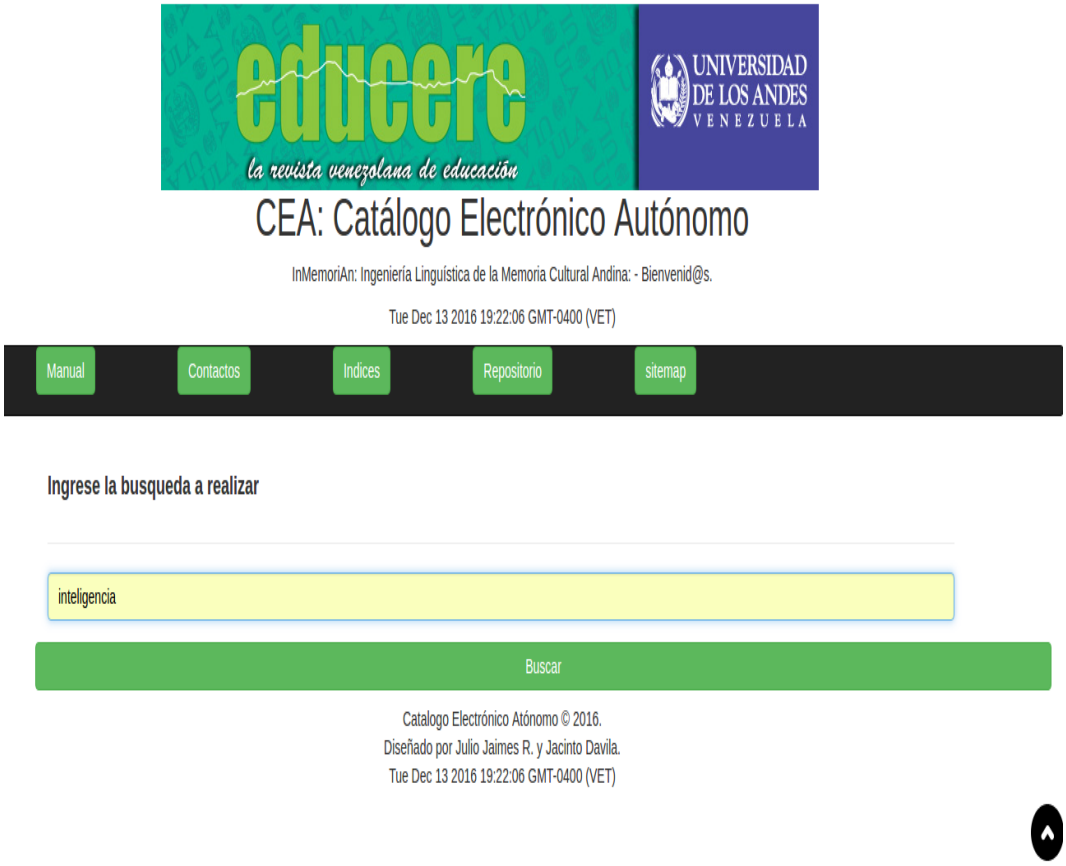

Figura 5.2: El usuario ingresa una búsqueda.

El usuario a continuación ingresa la consulta que desea, en este caso ingresó *inteligencia*, y luego presionó el botón buscar.

5.3. Resultados en ceaWeb.

El sistema envia una consulta según el usuario, luego el buscador con la herramienta swish-e genera la respuesta la cual es el resultado.

Relevancia	Archivo (URL implicito)	Tamano (en Bytes)
1000	articulo9.pdf	*421353*
913	articulo16.pdf	*358462*
824	articulo12.pdf	*765696*
776	articulo4-10-1.pdf	*347919*
670	articulo13.pdf	*377675*
659	articulo7.pdf	*566658*
659	articulo4.pdf	*147283*
648	articulo2.pdf	*797631*
613	art08.pdf	*513951*
600	articulo2.pdf	*1289391*

Figura 5.3: Resultados.

Se puede observar que el sistema ofrece los archivos según la consulta hecha por el usuario, esta se muestra según la relevancia, el url y el tamaño del archivo.

5.4. Seleccionar en ceaWeb.

El usuario selecciona un documento deseado a partir de su búsqueda y el sistema le provee la información.

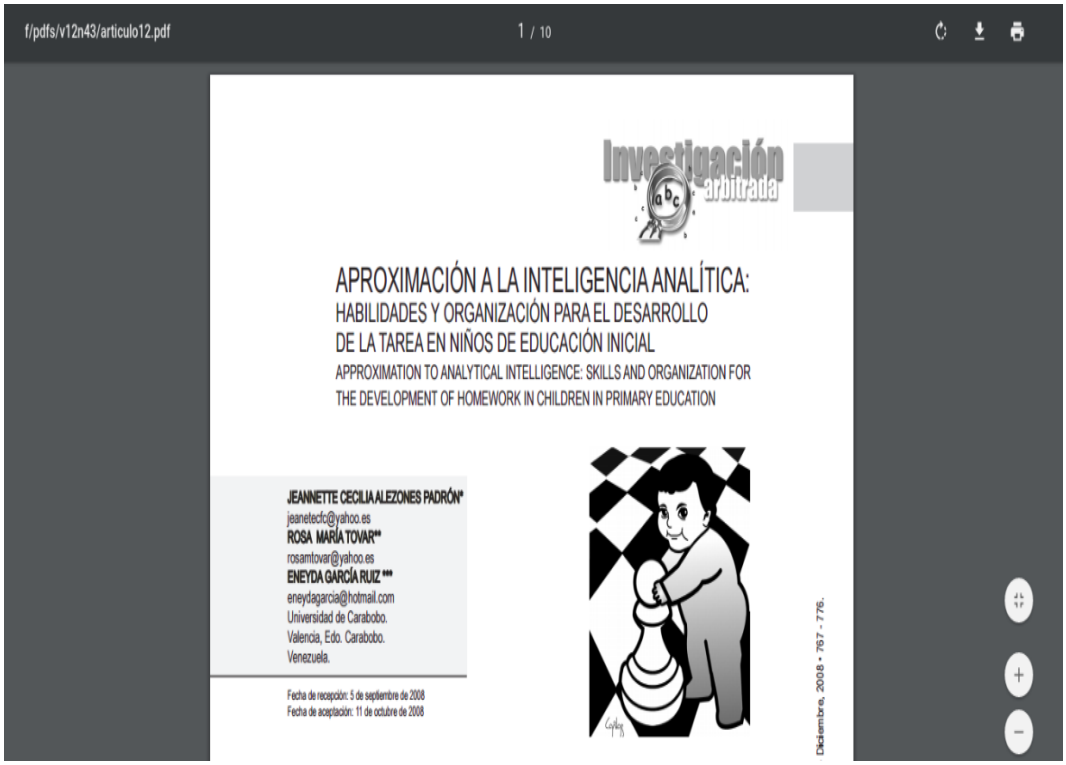

Figura 5.4: Prueba de selección de un archivo.

Es importante mencionar que en este momento el usuario puede guardar o imprimir el documento seleccionado para así tenerlo en algún directorio de su preferencia.

5.5. Servicio sitemap en ceaWeb.

El usuario selecciona en la barra de navegación el botón sitemap y este le llevará a toda la metadata que existe en su repositorio.

Repositorio completo basada en sitemap

Titulo	Autor	url	Fecha de modificaicon
"Bin laden es el hijo extraviado de la versión saudí del islam"	[[Yassin, Nadia]]	http://localhost:5000/indices/f/pdfs/v6n18/articulo15.pdf	2002-07-01
"Pistolas para el Niño, Muñecas para la Niña"	[[Gianini Belotti, Elena]]		2001-04-01
2007, año de consolidación de un proyecto editorial de una revista gestada desde la autonomía universitaria.	[[Rivas, Pedro José]]		2007-10-01
50 años de la escuela de educación. Discurso de orden	[[Aguaded Gómez, Ignacio],[Fandos Igado, Manuel]]		1997-01-01
A contracorriente: la socialización de los estudiantes en su camino a la universidad.	[[Velázquez Reyes, Luz María],[Pérez González, Julia]]	http://localhost:5000/indices/f/pdfs/v11n36/articulo14.pdf	2007-01-01
A diez años del golpe. Una tragicomedia en dos actos, con un epílogo inesperado	[[Mora García, José Pascual]]		1997-01-01
A la escucha de la infancia	[[Angulo, Lilian Nayive],[León Salazar, Aníbal]]		1997-01-01
A treinta años del 11 de septiembre chileno, nos ronda el olor a pólvora de la Moneda	[[Rivas, Pedro José]]	http://localhost:5000/indices/f/pdfs/v7n22/editorial.pdf	2003-07-01
Abordaje constructivista en educación especial: la construcción de significado en padres de niños con Síndrome de Down	[[Romero, Rosalinda]]		2003-07-01

Figura 5.5: metadata del sitemap.

En esta sección se ofrece al usuario, rastreadores y motores de búsqueda toda la metadata del sistema ceaWeb, y dar al usuario de una manera amigable todos los títulos, autores, url y fechas de modificaciones de cada metadata lo cual es muy importante.

Capítulo 6

Conclusiones y recomendaciones.

6.1. Conlusiones.

En el transcurso de este trabajo, se lograron los resultados esperados, se desarrolló el software Catálogo Electrónico Autónomo ceaWeb, consiste en un repositorios de bajo costo, el cual permitió extender la herramienta cea existente desarrollada por Jacinto Dávila quien es fue el tutor de este trabajo y Yaritza Vargas.

Se implementó una pagina web mediante el leguaje de programación Prolog, esta página se crea cada vez que se ejecuta el código con el html termerizado, de esta manera se termerizó el código html necesario para el diseño de la misma, esto hizó que la herramienta ceaWeb sea más eficiente, y a su vez dar funcionalidades a los usuarios para que tuvieran una interfaz amigable.

Para la indexación de toda la metadata se utilizó swish-e, una herramienta basada en software libre, permitiendo así arrojar buenos resultados al momento de la indexación por títulos, autores y búsqueda de información en toda la metada, para dar funcionalidades importantes en el sistema ceaWeb, teniendo una velocidad de respuesta muy rápida y eficiente.

Se realizó toda la documentación necesaria, para dar una orientación al usuario, permitiendo que todo el sistema en general tuviera su manual de instalación, y así tener en cualquier computador con requerimientos bajos.

Para la difusión y cosechado de la metada se desarrolló un servicio basado en sitemap, este consistió en generar un archivo xml con toda la metadata más importante del repositorio, teniendo en su interior etiquetas las cuales fueron título, autor, subtitulo, fecha de modificación y la dirección de cada metadata, este documento se incorporó a la herramienta ceaWeb permitiendo asi ofrecerla a todos los usuarios y buscadores web. Los sitemaps resultaron ser muy poderosos al momento de proveer los metadatos a los rastreadores y motores de búsqueda.

La funcionalidad del buscador de Google mediante el protocolo sitemap, consistió en enlazar todas las páginas web, en nuestro caso es toda la metadata de nuestros archivos pdfs, permitiendo de esta manera rastrear y monitorear todo estos datos, este enlace se realizó con un archivo robots.txt, el mismo estuvo en el directorio raiz del sistema, siendo así una de las herramientas fundamentales para la difusión y cosechado del sistema. Utilizar un sitemap no garantiza que todos los elementos del sitemap se vayan a rastrear e indexar, ya que los procesos de Google se basan en algoritmos complejos. Sin embargo, en la mayoría de los casos, tu sitio web se beneficiará de tener un sitemap y en ningún caso se verá perjudicado.

6.2. Recomendaciones.

El trabajo al ser un repositorio de bajo costo, nos ofrece oportunidades de desarrollo e implementación, ya que se pueden incorporar nuevas características y mejoras. Entre las cuales cabe destacar las siguientes:

- Realizar una red de intercomunicación entre usuarios.
- Implementar la herramienta en universidades u otras entidades de investigación para promover la ciencia e información.
- Ofrecer información a los rastreadores de búsqueda con otras implementaciones.
- Realizar un estudio sobre los sitemaps en otras herramientas para obtener características y relaciones más importantes en su implementación y arquitectura.
- Enviar mediante una interfaz al motor de búsqueda todos los sitemaps.

- Realizar envíos de solicitudes HTTP para que así los motores de búsqueda puedan realizar una difusión y cosechado.

  Observaciones:

  Puede enviar la solicitud HTTP utilizando wget, curl o el mecanismo que prefiera. Si la solicitud se procesa correctamente, recibirá un código de respuesta HTTP 200. Si recibe una respuesta diferente, debe volver a enviar la solicitud. El código de respuesta HTTP 200 sólo indica que el motor de búsqueda ha recibido su Sitemap, no que el Sitemap o las URL que incluye sean válidas. Una forma fácil de hacerlo es configurar una tarea automatizada que genere y envíe Sitemaps periódicamente.

  Nota: Si proporciona un archivo del índice de Sitemap, bastará con que envíe una solicitud HTTP que incluya la ubicación del archivo del índice de Sitemap; no es necesario que envíe solicitudes para cada Sitemap que se especifica en el índice.

Para el desarrollo de todas estas mejoras se debe tomar en cuenta algunos aspectos fundamentales con el buscador de Google:

- Si el xml del sitemap es muy grande, es más probable que los rastreadores web de Google se olviden de rastrear algunas páginas nuevas o actualizadas recientemente.

- Si tiene un gran archivo de páginas de contenido que están aisladas o no están bien enlazadas entre sí. Si las páginas del sitio no se hacen referencia de forma natural, puedes enumerarlas en un sitemap para asegurarte de que Google no pase por alto algunas de ellas.

- Es nuevo y es destinatario de pocos enlaces externos. El robot de Google y otros rastreadores web siguen los enlaces de una página a otra para rastrear la Web. Por eso, es posible que Google no detecte las páginas si no son destinatarias de enlaces de otros sitios.

Permitiendo de esta forma a los motores de búsqueda obtener su Sitemap y poner las URL a disposición de sus rastreadores de diferentes maneras, como recomendación

al desarrollar estas implementaciones mantener la característica más importante de esté trabajo y es que sea de bajo costo.

Apéndice A

Pasos para la instalación de ceaWb y swish-e.

A.1. Instalación de swish-e.

.

Existen dos formas para la instalación de swish-e en linux:

Instalación 1:

- Abrir la terminal (presionar Ctrl+T).
- Desde la terminal ejecutar el siguiente comando: sudo apt-get update
- sudo apt-get install swish-e
- Para verificar que está instalado presione el siguiente comando:

  swish-e -V
- Deberá tener algo parecido a esto:

  SWISH-E 2.4.7

Instalación 2:

1. Bajar el directorio swish-e-2.4.7 de github donde está el proyecto.

   Puedes descargarlo desde www.swish-e.org.

2. Ingresar al directorio con el siguiente comando:

   cd swish-e-2.4.7 (this directory will depend on the version of Swish-e)

3. Ejecutar el siguiente comando: ./configure make make check

4. Ejecutar el siguiente comando como root:

   make install

5. Si todo vá bien ejecute el siguiente comando:

   swish-e -V

6. Deberá ver:

   SWISH-E 2.4.7

A.2. Pasos para la instalación del sistema ceaWeb

1. Descargar el proyecto completo en github.

2. Realizar la indexación de los documentos.

3. Entrar en el directorio del proyecto y abrir la terminal desde allí.

4. Ejecutar el siguiente comando:

   swipl

5. Desde swipl ejecute el siguiente comando [ceaWeb]. será de la siguiente forma en la terminal:

   Welcome to SWI-Prolog (Multi-threaded, 64 bits, Version 7.2.3) Copyright (c) 1990-2015 University of Amsterdam, VU Amsterdam SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software, and you are welcome to redistribute it under certain conditions. Please visit http://www.swi-prolog.org for details.

   For help, use ?- help(Topic). or ?- apropos(Word).

   ?- [ceaWeb].

6. Aparecerá esto en la consola:

   true.

   ?-

7. Dar click desde la consola a : http://localhost:5000/ o http://nux.ula.ve:5000/ceaweb/

   Otra forma es abrir su navegador y colocar la siguiente dirección url y abrir : http://localhost:5000/

   Nota: como el sistema estará alojado en CESIMO perteneciente a la Universidad de Los Andes, se mostrará de la siguiente manera: http://nux.ula.ve:5000/ceaweb/

Observaciones.

Por defecto el sistema se ejecuta en el puerto 5000, pero usted puede cambiarlo al ejecutar:

?- server[AquiVaElNumeroDePruerto].

un ejemplo es:

?- server[8000].

Luego seguir los pasos 6 y 7.

A.3. Pasos para generar el archivo sitemap xml de ceaWeb.

Se creó un modulo el cual permite crear el archivo sitema.xml de manera automática del ceaWeb.

1. Abrir la terminal de tu sistema operativo
2. Ingresar en la terminal swipl
3. carga_xml(A).
4. genera_xml_site("sitemap.xml").

Bibliografía

[1] *Sitemaps*. `https://www.sitemaps.org/es/`, 18 de Marzo del 2016.

[2] Arencibia y Jorge Ricardo. *Las iniciativas para el acceso abierto a la información científica en el contexto de la web semántica*. En: Biblos Vol. 7:25, 2006.

[3] Jacinto Davila y Yaritza Vargas. *Catalogo Electronico Autonomo CEA*. `https://github.com/jacintodavila/catalogoelectronicoautonomo/`, 20 de Enero del 2016.

[4] Ayuda de Search Console. *Más información sobre sitemaps*. `https://support.google.com/webmasters/answer/156184?hl=es`, 15 de Septiembre del 2016.

[5] Gómez Duenas y Laureano Felipe. *Las nuevas tecnologías en los procesos de cooperación documental: Aumento de la visibilidad para REDINED.*

[6] Metadata Principles y Anne J. Gilliland-Swetland Practicalities, Erik Duval et al. .Setting the Stage.

[7] the Digital Library Federation the National Science Foundation (IIS-9817416 IIS-0430906) Support for Open Archives Initiative activities has come from the Andrew W. Mellon Foundation, the Coalition for Networked Information y from the Alfred P. Sloan Foundation. *Open Archives Initiative OAI*. `https://www.openarchives.org/pmh`, 20 de Enero del 2016.

[8] the Digital Library Federation the National Science Foundation (IIS-9817416 IIS-0430906) Support for Open Archives Initiative activities has come from the Andrew W. Mellon Foundation, the Coalition for Networked Information

y from the Alfred P. Sloan Foundation. *Open Archives Initiative OAI.* `https://www.openarchives.org/pmh/tools/tools.php`, 20 de Enero del 2016.

[9] the Digital Library Federation the National Science Foundation (IIS-9817416 IIS-0430906) Support for Open Archives Initiative activities has come from the Andrew W. Mellon Foundation, the Coalition for Networked Information y from the Alfred P. Sloan Foundation. *Open Archives Initiative OAI.* `http://www.openarchives.org/`, 20 de Enero del 2016.

[10] Swish-e. `http://swish-e.org`. consultado el 25 de Octubre de 2016.

[11] la enciclopedia libre Wikipedia. *Hypertext Transfer Protocol.* `http://es.wikipedia.org/wiki/Http`, 20 de Enero del 2016.