



**UNIVERSIDAD AUTÓNOMA
DE AGUASCALIENTES**

Centro de ciencias básicas

Tesis

**Modelo de Recuperación de Recursos Educativos con
Tecnología Libre e Implementando una Arquitectura en Capas
Orientada a Servicios**

PRESENTA

Mario Alberto Luna Preciado

**PARA LA OBTENER EL GRADO DE MAESTRÍA EN CIENCIAS DE LA
COMPUTACIÓN**

Director de tesis:

Dr. Francisco Javier Álvarez Rodríguez

Comité tutorial:

Dr. Francisco Javier Álvarez Rodríguez

Dr. Jaime Muñoz Arteaga

Dr. Miguel Vargas Martin

Aguascalientes, Ags., 05 de agosto del 2016



UNIVERSIDAD AUTÓNOMA
DE AGUASCALIENTES

CENTRO DE CIENCIAS BÁSICAS

MARIO ALBERTO LUNA PRECIADO
MAESTRÍA EN CIENCIAS: CON OPCIÓN A LA COMPUTACIÓN
PRESENTE.

Estimado alumno:

Por medio de este conducto me permito comunicar a Usted que habiendo recibido los votos aprobatorios de los revisores de su trabajo de tesis y/o caso práctico titulado: **“Modelo de recuperación de recursos educativos con tecnología libre implementando una arquitectura en capas orientada a servicios”**, hago de su conocimiento que puede imprimir dicho documento y continuar con los trámites para la presentación de su examen de grado.

Sin otro particular me permito saludarle muy afectuosamente.

ATENTAMENTE

Aguascalientes, Ags., a 2 de agosto de 2016

“Se lumen proferre”

EL DECANO

M. en C. **JOSÉ DE JESÚS RUIZ GALLEGOS**





UNIVERSIDAD AUTÓNOMA
DE AGUASCALIENTES

M.C. JOSÉ DE JESÚS RUIZ GALLEGOS
DECANO DEL CENTRO DE CIENCIAS BÁSICAS

PRESENTE

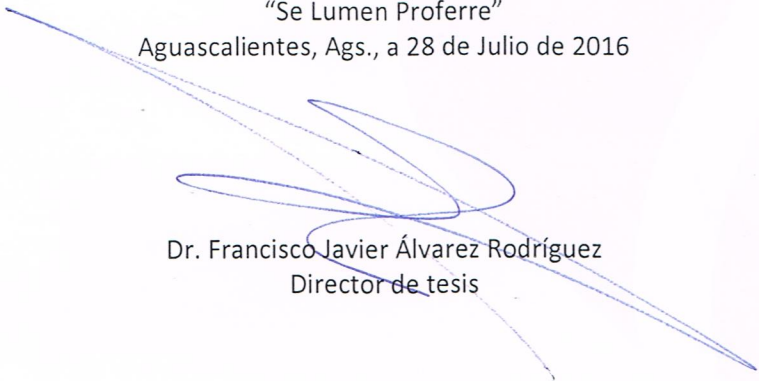
Por medio de la presente como Co-Tutor asignado del estudiante **MARIO ALBERTO LUNA PRECIADO** con ID 187578 quien realizó el trabajo de tesis titulado: **MODELO DE RECUPERACIÓN DE RECURSOS EDUCATIVOS CON TECNOLOGÍA LIBRE IMPLEMENTANDO UNA ARQUITECTURA EN CAPAS ORIENTADA A SERVICIOS**, y con fundamento en el Artículo 175, Apartado II Del Reglamento General de Docencia, me permito emitir el **VOTO APROBATORIO**, para que él pueda proceder a imprimirla, y así como continuar con el procedimiento administrativo para la obtención del grado.

Pongo lo anterior a su digna consideración y sin otro particular por el momento, me permito enviarle un cordial saludo.

ATENTAMENTE

"Se Lumen Proferre"

Aguascalientes, Ags., a 28 de Julio de 2016



Dr. Francisco Javier Álvarez Rodríguez
Director de tesis

c.c.p. – Interesado
c.c.p. – Secretaría de Investigación y Posgrado
c.c.p. – Consejero Académico
c.c.p. – Minuta Secretario Técnico



UNIVERSIDAD AUTÓNOMA
DE AGUASCALIENTES

FORMATO DE CARTA DE VOTO APROBATORIO

M.C. JOSÉ DE JESÚS RUIZ GALLEGOS
DECANO DEL CENTRO DE CIENCIAS BÁSICAS

PRESENTE

Por medio de la presente como Co-Tutor asignado del estudiante **MARIO ALBERTO LUNA PRECIADO** con ID 187578 quien realizó el trabajo de tesis titulado: **MODELO DE RECUPERACIÓN DE RECURSOS EDUCATIVOS CON TECNOLOGÍA LIBRE IMPLEMENTANDO UNA ARQUITECTURA EN CAPAS ORIENTADA A SERVICIOS**, y con fundamento en el Artículo 175, Apartado II Del Reglamento General de Docencia, me permito emitir el **VOTO APROBATORIO**, para que él pueda proceder a imprimirla, y así como continuar con el procedimiento administrativo para la obtención del grado.

Pongo lo anterior a su digna consideración y sin otro particular por el momento, me permito enviarle un cordial saludo.

ATENTAMENTE

“Se Lumen Proferre”

Aguascalientes, Ags., a 28 de Julio de 2016


Dr. Jaime Muñoz Arteaga
Co-Tutor de tesis

c.c.p. – Interesado
c.c.p. – Secretaría de Investigación y Posgrado
c.c.p. – Consejero Académico
c.c.p. – Minuta Secretario Técnico

FORMATO DE CARTA DE VOTO APROBATORIO

M.C. JOSÉ DE JESÚS RUIZ GALLEGOS
DECANO DEL CENTRO DE CIENCIAS BÁSICAS

PRESENTE

Por medio de la presente como Co-Tutor asignado del estudiante **MARIO ALBERTO LUNA PRECIADO** con ID 187578 quien realizó el trabajo de tesis titulado: **MODELO DE RECUPERACIÓN DE RECURSOS EDUCATIVOS CON TECNOLOGÍA LIBRE IMPLEMENTANDO UNA ARQUITECTURA EN CAPAS ORIENTADA A SERVICIOS**, y con fundamento en el Artículo 175, Apartado II Del Reglamento General de Docencia, me permito emitir el **VOTO APROBATORIO**, para que él pueda proceder a imprimirla, y así como continuar con el procedimiento administrativo para la obtención del grado.

Pongo lo anterior a su digna consideración y sin otro particular por el momento, me permito enviarle un cordial saludo.

ATENTAMENTE

Ontario, Canadá, a 1 de agosto de 2016

Digitally signed by Miguel Vargas Martin
DN: cn=Miguel Vargas Martin,
o=University of Ontario Institute of
Technology, ou=Faculty of Business
and IT,
email=miguel.vargasmartin@uoit.ca,
c=CA
Date: 2016.08.01 17:59:24 -04'00'

Dr. Miguel Vargas Martin
Co-Tutor de tesis

- c.c.p. – Interesado
- c.c.p. – Secretaría de Investigación y Posgrado
- c.c.p. – Consejero Académico
- c.c.p. – Minuta Secretario Técnico

AGRADECIMIENTOS

Al terminar un trabajo arduo y lleno de complicaciones como el desarrollo y escritura de tesis, es inevitable que te asalte un muy humano egocentrismo que te lleva a concentrar el mérito en el aporte que has realizado. Sin embargo, el análisis objetivo te muestra inmediatamente que dicho aporte no hubiese sido posible sin la participación de personas e instituciones que han facilitado las cosas para que este trabajo llegue a un feliz término. Por ello, es para mí un verdadero placer utilizar este espacio para ser justo y consecuente con ellas, expresándoles mis agradecimientos.

Debo agradecer de manera especial al Dr. Francisco Álvarez Rodríguez, mi director de tesis, por aceptarme para realizar este trabajo de tesis de maestría bajo su dirección. Su apoyo y confianza en este trabajo, y su capacidad para guiar las ideas han sido invaluable, no solamente en el desarrollo de la tesis, sino también en mi formación como investigador. Le agradezco también el haberme facilitado los medios suficientes para llevar a cabo las actividades propuestas. Quiero agradecerle también, por ofrecerme su apoyo y consejos durante el peor momento que he pasado de mi vida. Muchas gracias Dr. Álvarez, espero volver a verlo pronto.

Quiero agradecer a mis Co-tutores de tesis al Dr. Jaime Muñoz y al Dr. Miguel Vargas Martín, por el apoyo y conocimiento que me ofrecieron durante el desarrollo de esta tesis. Sus ideas y conocimientos impartidos también fueron importantes para la finalización de este trabajo.

Agradezco también a mi familia, que sin su apoyo no hubiera podido seguir con esta formación. Aunque estuvimos separados sé que de alguna manera estuvieron conmigo ofreciéndome su apoyo. En especial a mi padre y madre, que primeramente me regalaron la vida. A mis hermanos que aunque son muchos y no todos estamos ya, sepan que los quiero mucho y les agradezco el apoyo brindado. A mi padre por haberme ofrecido su conocimiento y apoyo para realizar este proyecto. A mi madre le agradezco por escucharme y guiarme durante el trayecto de mi vida. Yasher te agradezco el apoyo moral que me ofreciste en mis peores momentos, sé que

tu nueva manera de ver la vida te llevara por buen camino. Claudia, no pude estar para ti en tus peores momentos, pero me escuchaste en el momento más clave de mi vida. Sin esa platica, no hubiera podido culminar este proyecto. Javier, eres el hermano más pequeño que tengo y aun así me ofreciste apoyo, te quiero mucho. Sé que culminaras tus proyectos de vida, y serás una persona feliz.

A mis amigos les agradezco también, en especial a los de la maestría, José, Juan, Miguel, Humberto y Alma. Les agradezco también el apoyo brindado durante estos dos años, espero haber sido reciproco. A mis amigos de mi tierra natal, que los visito cada tanto tiempo. Les agradezco también por el simple hecho de ser mis amigos.

Te agradezco Pyle, por apoyarme durante estos años, hemos sido amigos por un largo tiempo, por el momento nos separamos del camino, pero sé que culminaras todos tus proyectos de vida, muchas gracias “bro”. Muchas suerte en Alaska y Canada, espero volver a verte pronto.

Te agradezco de manera especial a Diego, José por estar conmigo en esos peores momentos y los mejores momentos de mi vida. Nos la pasamos genial cuando estamos juntos, ¡gracias!

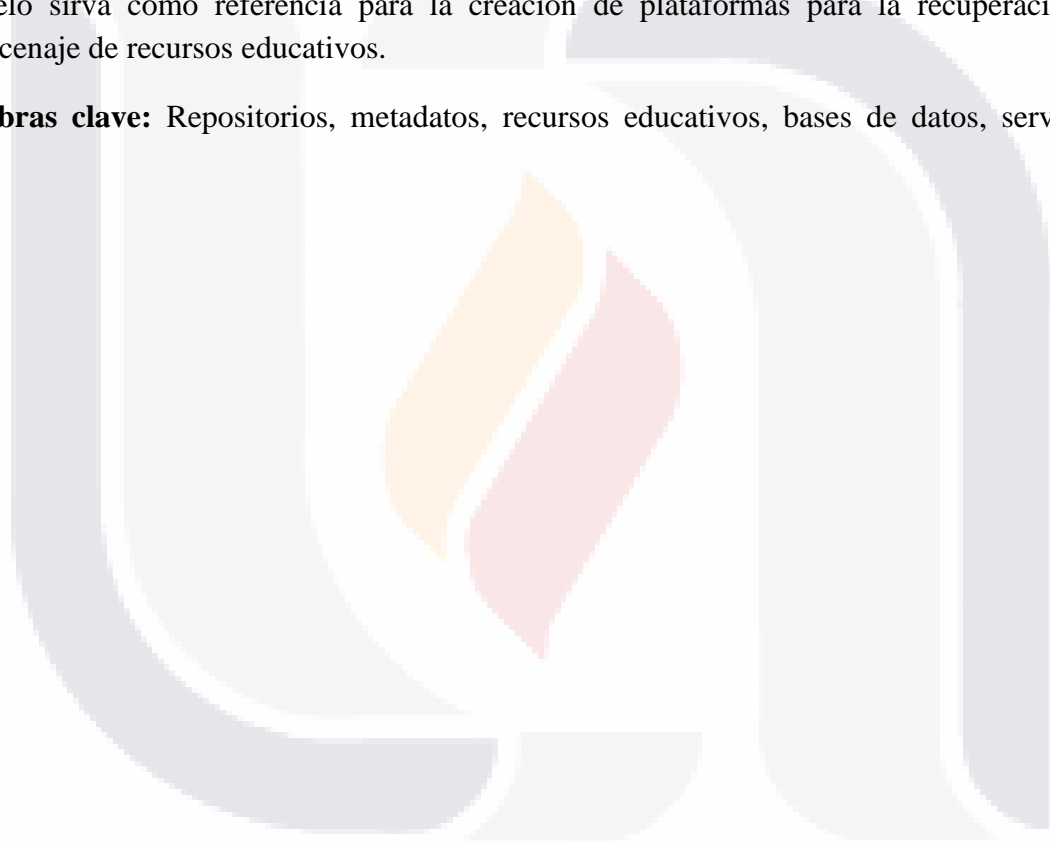
Gracias también a todas esas personas que me cruce durante mi camino de vida estos dos últimos años, que aunque no puedo nombrar a todos, sepan que también les agradezco la amistad brindada.

Por último, pero no menos importante. Gracias Edith por el apoyo brindado durante la mayor parte de la maestría. Gracias por los momentos tan agradables que pasamos. Sé que cumplirás todas tus metas y objetivos de vida. ¡Mucha suerte!

Resumen

El día de hoy se cuentan con varias técnicas para la recuperación de recursos educativos en diversos sistemas de administración. Una alternativa es utilizar repositorios de recursos educativos para que estos puedan ser recuperados por medio de un buscador utilizando ciertos criterios del usuario usando metadatos. Este documento propone un modelo para recuperar y almacenar recursos educativos en un repositorio. Dicho modelo implementa: servicios web, metadatos y criterios de búsqueda del usuario para recuperar los recursos educativos que más se aproximen a los requerimientos del usuario. La idea es que este modelo sirva como referencia para la creación de plataformas para la recuperación y almacenaje de recursos educativos.

Palabras clave: Repositorios, metadatos, recursos educativos, bases de datos, servicios web.



Índice de figuras

Figura 1. Proceso de desarrollo de un recurso educativo ----- 4

Figura 2. Situación actual de los recursos educativos abiertos, objetos de aprendizaje y recursos educativos ----- 5

Figura 3. Esquema de elementos del estándar DCMI ----- 24

Figura 4. Esquema de elementos del estándar LOM ----- 25

Figura 5. Esquema de metadatos propuesto ----- 26

Figura 6. Estructura de elementos básicos de una arquitectura SOA ----- 34

Figura 7. Colaboraciones de una arquitectura SOA ----- 35

Figura 8. Diagrama de casos de uso ----- 49

Figura 9. Arquitectura de MTG-DB: A Repository for Music Audio Processing ----- 60

Figura 10. Arquitectura de An Infrastructure for Open-Architecture Digital Libraries ---- 62

Figura 11. Arquitectura de I2Cnet: content-based similarity search in geographically distributed repositories of medical images ----- 64

Figura 12. Arquitectura de una tienda de aplicaciones ----- 66

Figura 13. Arquitectura de un Sistema Interactivo Distribuido de Repositorios de Objetos de Aprendizaje MatEmáticos ----- 69

Figura 14. Tabla Comparación de requerimientos propuestos sobre las arquitecturas analizadas ----- 71

Figura 15. Diagrama de objetos que representa de manera general el repositorio ----- 72

Figura 16. Diagrama de componentes e interacción entre los mismos del repositorio de recursos educativos ----- 72

Figura 17. Diagrama general de la arquitectura propuesta ----- 73

Figura 18. Diagrama detallado de la arquitectura propuesta para el repositorio ----- 75

Figura 19. Capa 1 denominada como cliente de la arquitectura propuesta ----- 76

Figura 20. Capa 2 denominada como Interfaz de usuario de la arquitectura propuesta ----- 77

Figura 21. Interfaz de reproducción de audio ----- 81

Figura 22. Interfaz de visualización de documentos en línea ----- 82

Figura 23. Interfaz de reproducción de recursos del tipo video ----- 83

Figura 24. Capa 3 denominada como Servicios para el usuario de la arquitectura propuesta ----- 84

Figura 25. Capa 4 denominada como Seguridad de la arquitectura propuesta ----- 88

Figura 26. Capa 5 denominada como Almacenaje de la arquitectura propuesta ----- 90

Figura 27. Diagrama de la estructura general del repositorio de la Universidad Autónoma de Aguascalientes ----- 94

Figura 28. Diagrama general del comportamiento del repositorio de recursos educativos --- 94

Figura 29. Interfaz para búsqueda recursos implementada en el repositorio ----- 102

Figura 30. Interfaz implementada para el registro de recursos educativos en el repositorio 106

Figura 31. Gráfica de resultados de la prueba de registro en la primera iteración ----- 112

Figura 32. Gráfica de resultados de la prueba de registro en la segunda iteración ----- 112

Figura 33. Gráfica de resultados de la prueba de búsqueda en la primera iteración ----- 112

Figura 34. Gráfica de resultados de la prueba de búsqueda en la segunda iteración ----- 112

Figura 35. Síntesis de análisis de tecnologías para el desarrollo de un repositorio de recursos educativos ----- 115

Figura 36. Resultados obtenidos de las pruebas realizadas al prototipo ----- 118

Índice de tablas

Tabla 1. Listado de requerimientos para el desarrollo del repositorio de recursos de aprendizaje ----- 48

Tabla 2. Definición de servicios y nombre de la implementación ----- 92

Tabla 3. Lista de dispositivos a implementar en las pruebas ----- 96

Índice de anexos

Anexo 1 Código de funcionalidad de la capa 1 ----- 123

Anexo 2 Código de para interfaz de búsqueda de recursos ----- 124

Anexo 3 Código de para interfaz de listado de recursos ----- 124

Anexo 4 Código implementado para controles de archivos de audio ----- 125

Anexo 5 Código implementado para lector de documentos -----	125
Anexo 6 Código implementado para ver imágenes en el repositorio -----	125
Anexo 7 Código implementado para la reproducción de video -----	125
Anexo 8 Código para realizar la búsqueda de recursos -----	125
Anexo 9 Código para realizar el registro de recursos -----	126
Anexo 10 Código implementado para la descarga de recursos -----	127
Anexo 11 Código implementado para el registro de metadatos -----	127
Anexo 12 Código implementado para evitar SQL -----	128
Anexo 13 Formato de prueba de identificador de plataforma -----	129
Anexo 14 Formato de prueba búsqueda de recursos -----	130
Anexo 15 Formato de prueba de registro de recursos -----	131
Anexo 16 Formato de prueba de descarga de recursos -----	132
Anexo 17 Formato de prueba de visualización de recurso -----	133
Anexo 18 Carta aceptación a conferencia Ccita 2015 -----	134
Anexo 19 Artículo Ccita 2015: Una Arquitectura en Capas Orientada a Servicios para el Desarrollo de un Repositorio de Recursos de Aprendizaje ArCaOrSe -----	135
Anexo 20 Carta aceptación a XI Conferencia latinoamericana de objetos y tecnologías de aprendizaje -----	143
Anexo 21 Artículo LACLO 2016: Implementación de Arquitectura ArCaOrSe de un Repositorio de Recursos Educativos y Obtención de Resultados Preliminares de Pruebas Realizadas a la Implementación -----	144

INDICE GENERAL

1 – Introducción a la investigación

1.	Introducción a la investigación -----	1
1.1	Contexto de la investigación -----	1
1.2	Definición de la problemática de la investigación -----	5
1.2.1	Antecedentes de la investigación -----	8
1.2.2	Relevancia de la investigación -----	12
1.3	Justificación de la investigación -----	14
1.4	Descripción del reporte de Tesis -----	15

2 – Formulación de la investigación

2.	Formulación de la investigación -----	17
2.1	Preguntas de la investigación -----	18
2.1.1	Pregunta general de la investigación -----	18
2.1.2	Preguntas específicas de la investigación -----	18
2.2	Objetivos de la investigación -----	19
2.2.1	Objetivo general de la investigación -----	19
2.2.2	Objetivo específicos de la investigación -----	19

3 – Marco teórico

3.	Marco teórico -----	20
3.1	Contenido educativo -----	20
3.1.1	Recursos educativos -----	21
3.1.2	Formatos de recursos educativos -----	22
3.2	Metadatos -----	23
3.3	Servicios Web -----	26
3.3.1	Big web services -----	27
3.3.2	Web services description language -----	27
3.3.3	Simple Object Access Protocol -----	28

3.3.4	RESTful web services -----	28
3.3.5	Comparación big web services vs RESTful web services -----	29
3.4	Repositorios de recursos educativos -----	30
3.4.1	Importancia de los repositorios de recursos educativos -----	31
3.4.2	Tipos de recursos educativos -----	31
3.5	Estilos de arquitectura -----	32
3.5.1	Arquitectura orientada a servicios -----	33
3.5.2	Arquitectura en capas -----	38
3.6	Visualización de recursos educativos en línea -----	42
3.6.1	Visualización de diferentes tipos de recursos educativos -----	43
4	Desarrollo del repositorio de recursos educativos	
4.	Desarrollo del repositorio de recursos educativos -----	46
4.1	Diseño de arquitectura para su implementación -----	46
4.1.1	Objetivos de una arquitectura -----	46
4.1.2	Requerimientos de la arquitectura -----	47
4.1.3	Diagrama de casos de uso y actores -----	48
4.1.4	Casos de uso -----	49
4.1.5	Comparación de arquitecturas analizadas como soluciones parciales -----	58
4.1.5.1	MTG-DB un repositorio para el procesamiento de audio música ---	59
4.1.5.2	Una infraestructura para arquitecturas abiertas de librerías visuales-- -----	61
4.1.5.3	I2Cnet: Búsquedas basadas en contenido similar en repositorios geográficamente distribuidos de imágenes médicas -----	63
4.1.5.4	Habilitando agilidad a través de la arquitectura -----	65
4.1.5.5	Sistema interactivo distribuido de repositorios de objetos de aprendizaje matemáticas -----	67
4.1.6	Justificación de la arquitectura -----	70
4.2	Definición arquitectura para su implementación -----	71

4.2.1	Capa 1: Cliente	75
4.2.2	Capa 2: Interfaz de usuario	77
4.2.3	Capa 3: Servicios para el usuario	83
4.2.4	Capa 4: Seguridad	88
4.2.5	Capa 5: Almacenamiento	89

5 – Validación de la arquitectura con la implementación de un prototipo real

5	Validación de la arquitectura con la implementación de un prototipo real	91
5.1	Objetivos y servicios	91
5.2	Implementación	92
5.3	Validación de servicios y perfil de usuario	95
5.3.1	Prueba de identificador de cliente	97
5.3.2	Prueba de búsqueda de recursos	100
5.3.3	Prueba de registro de recursos	103
5.3.4	Prueba de descarga de recursos	107
5.3.5	Prueba de visualización de recursos en línea	109
5.3.6	Pruebas realizadas	111
5.4	Conclusiones	113

6 – Conclusiones y futuras líneas de investigación

6.	Conclusiones y futuras líneas de investigación	114
6.1	Conclusiones, objetivos y aportaciones	114
6.2	Futuras líneas de investigación	117

Referencias

Referencias	119
-------------------	-----

Anexos

Anexos	123
--------------	-----

1 Introducción a la investigación

El presente documento es la memoria de la tesis de maestría “Modelo de Recuperación de Recursos educativos de Tecnología Libre e Implementando una Arquitectura en Capas Orientada a Servicios para Múltiples Plataformas” del departamento de ciencias básicas de la Universidad Autónoma de Aguascalientes.

Este primer capítulo tiene como finalidad introducir los conceptos previos relevantes para la propuesta y la Aplicación, así como describir el propio documento para facilitar su lectura.

1.1 Contexto de la investigación

El contexto de la investigación gira entorno a los repositorios de recursos educativos y la utilización de estos para la recuperación y almacenamiento de los recursos educativos. Es importante mencionar que la finalidad del repositorio que se pretende desarrollar es una arquitectura capaz de almacenar recursos educativos y no será el caso para los recursos educativos abiertos. Esto se debe a que los recursos que se almacenaran son parte de los recursos educativos generados en la Universidad Autónoma de Aguascalientes.

Las instituciones educativas en todo el mundo buscan gestionar su educación, investigación y recursos de forma efectiva y transparente, esto se logra haciendo que la investigación y la producción científica se encuentre disponible, se difunda en el ánimo de entablar relaciones entre académicos y centros de investigación tanto nacionales como internacionales, lo que traería aparejado estímulo económico, desarrollo social y por sobre todo el desarrollo de un entorno pedagógico rico en información, que tendrá un enfoque centrado en el estudiante.

Las universidades hoy en día cuentan con repositorios de recursos educativos para que sus alumnos tengan acceso a diversos materiales educativos. Cada vez son más las universidades que cuentan con repositorios de recursos educativos y abren sus repositorios a nivel tanto nacional como internacional (Barrueco, 2009). Este panorama en plena

transformación y potenciación merece que por un momento fijemos en ellos nuestra atención, examinando detalladamente que características conllevan, en qué circunstancias están creciendo y con qué singularidades. De esta manera se puede orientar las próximas creaciones de repositorios de forma que verdaderamente sirvan para que la producción científica y académica de los miembros de una institución pueda ser gestionada, organizada y preservada que por un medio de acceso libre a la misma, se pueda promocionar tanto en beneficio de la cultura, como del propio investigador y de la institución que lo soporta.

En la actualidad existen diversas técnicas para la recuperación de recursos educativos de la web, una de las técnicas para ello es la utilización de repositorios de recursos educativos. Los repositorios de recursos educativos facilitan el almacenamiento y recuperación de los recursos educativos.

Se han hecho grandes inversiones en el desarrollo de los repositorios de recursos educativos en el sector académico en años recientes como el programa e-learning de EduSourceCanda, en el cual se invirtieron. Esto se debe a que la importancia de los repositorios de recursos educativos ha ido en aumento dado que el desarrollo de recursos educativos ha ido en aumento.

Los repositorios de recursos educativos pueden ser comparados con bibliotecas digitales combinados con un buscador de recursos en él. Para que un repositorio cumpla su objetivo se debe contar con recursos educativos debidamente etiquetados para que estos puedan ser identificados de manera única.

En los últimos años se comenzó un movimiento que inicio con el desarrollo de Software de Código Abierto, continuó con la formación de diferentes estándares de licenciamiento diferentes a las leyes que contempla el derecho internacional y remato con la creación y provisión de contenidos abiertos para cursos; Como resultado de la evolución surge una iniciativa con una idea simple pero poderosa “El conocimiento es un bien público y tanto la tecnología general, como internet en particular, ofrecen una oportunidad extraordinaria para cualquiera, desde cualquier sitio, comparta, use y aproveche este conocimiento”.

En el año 2002 la UNESCO se convirtió en la organización anfitriona de la discusión internacional en torno a esta iniciativa, cuando en el “Foro sobre Impacto de los Cursos

Abiertos para Educación Superior en los países en desarrollo” se adoptó la sigla OER (del inglés Open Educational Resources) y cuya traducción al español fue REA (Recursos educativos abiertos).

Una definición de los REA es: “Los recursos digitales que apoyan la educación y pueden reutilizarse constantemente, indicando las características de una mínima estructura como el objetivo, la actividad de aprendizaje y el sistema de evaluación” (Enríquez, 2004). Los recursos educativos abiertos son de tres tipos: contenidos educativos, herramientas y recursos de implementación.

Si desglosamos cada uno de los tres tipos de recursos mencionados, encontramos que los Recursos educativos pueden estar compuestos por (EDUTEKA, 2008):

- **Contenidos educativos:** Cursos completos (programas educativos), materiales para cursos, módulos de contenido, objetos de aprendizaje, libros de texto, materiales multimedia (texto, sonido, video, imágenes, animaciones), exámenes, compilaciones, publicaciones periódicas (diarios y revistas) etc.
- **Herramientas:** Software para apoyar la creación, entrega (acceso), uso y mejoramiento de contenido educativos abiertos. Esto incluye herramientas y sistemas para: crear contenido, registrar y organizar contenido; gestionar el aprendizaje y desarrollar comunidades de aprendizaje en línea.
- **Recursos de implementación:** Licencias de propiedad intelectual que promuevan la publicación abierta de materiales, principios de diseño, adaptación y localización de contenido y materiales o técnicas para apoyar el acceso al conocimiento. Por general, quienes crean REA permiten que cualquier persona use sus materiales, los modifique, los traduzca o los mejore y, además que lo comparta con otros. Se debe tener en cuenta que algunas licencias restringen las modificaciones (obras derivadas) o el uso comercial.

El tema de recursos educativos se aborda más adelante, pero los recursos educativos son entidades digitales que pueden ser utilizados para un aprendizaje de un tema en específico apoyado en esta tecnología. Considerando la definición de los REA y la breve definición de los recursos educativos, la diferencia consiste en que los recursos educativos pueden consistir en una simulación o en otro material educativo muy puntual; por su parte, los recursos educativos abiertos son algo más integral, como un curso. Un recurso educativos

es muy preciso, orientado a un tema específico que el docente debe tratar en un curso (EDUTEKA, 2007).

Las directrices que guían el proceso de diseño de un recurso educativo se muestra en la figura 1:

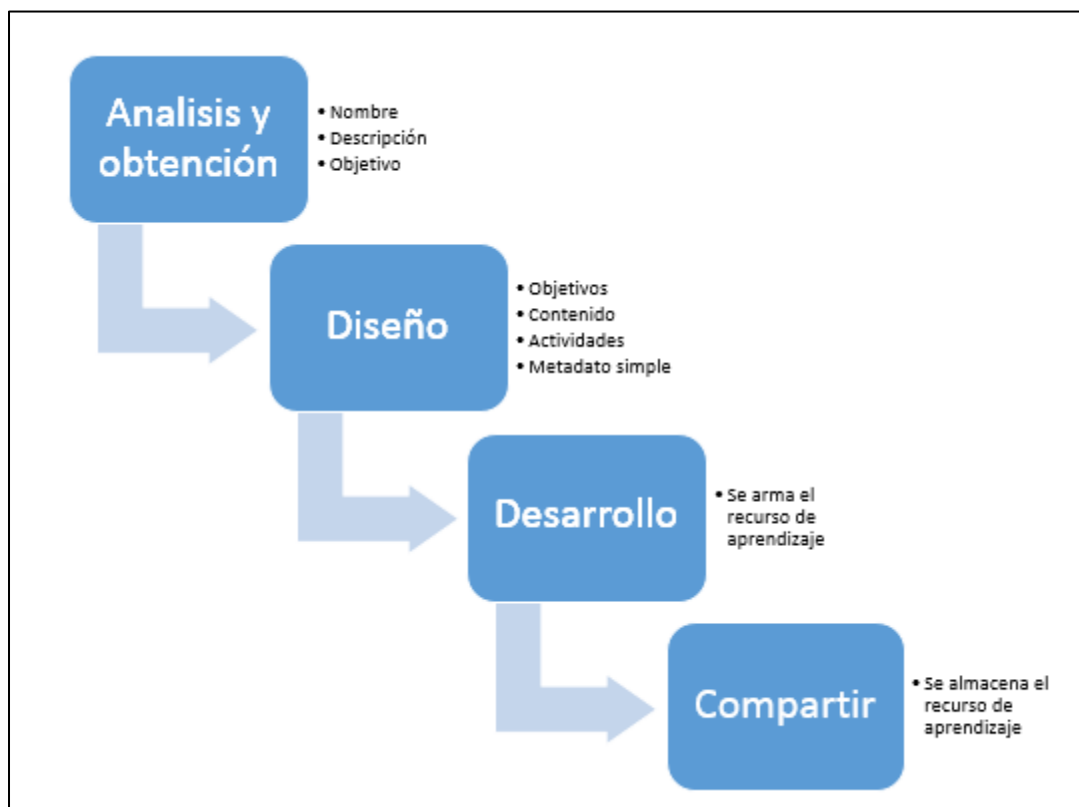


Figura 1. Proceso de desarrollo de un recurso educativo

Este proceso de desarrollo de recursos educativos permite tener una idea clara de cómo crear recursos educativos. Los recursos educativos se diferencian principalmente de REA dado el proceso de creación del mismo. Ya que los REA llevan un proceso iterativo (Maina et Al, 2012) que envuelven una serie de revisiones del recurso para la creación y validación del mismo y no es así para un recurso educativo. Esto se debe a que los recursos educativos suelen ser desarrollados o producidos de manera independiente en las universidades, y debido a que el proceso para el desarrollo de un REA es más complejo, estos recursos educativos nunca llegan a convertirse en REA por lo que estos recursos educativos se pierden y no pueden ser recuperados, por lo que estos recursos educativos suelen quedarse únicamente con el autor (UNESCO, 2012).

1. INTRODUCCIÓN

Los recursos educativos creados en las universidades no suelen ser almacenados en un repositorio por lo que como se mencionó, estos suelen quedarse con su autor o en su defecto se pierden. En la figura 2 podemos observar la situación actual de recursos educativos abiertos, objetos de aprendizaje y los recursos educativos.

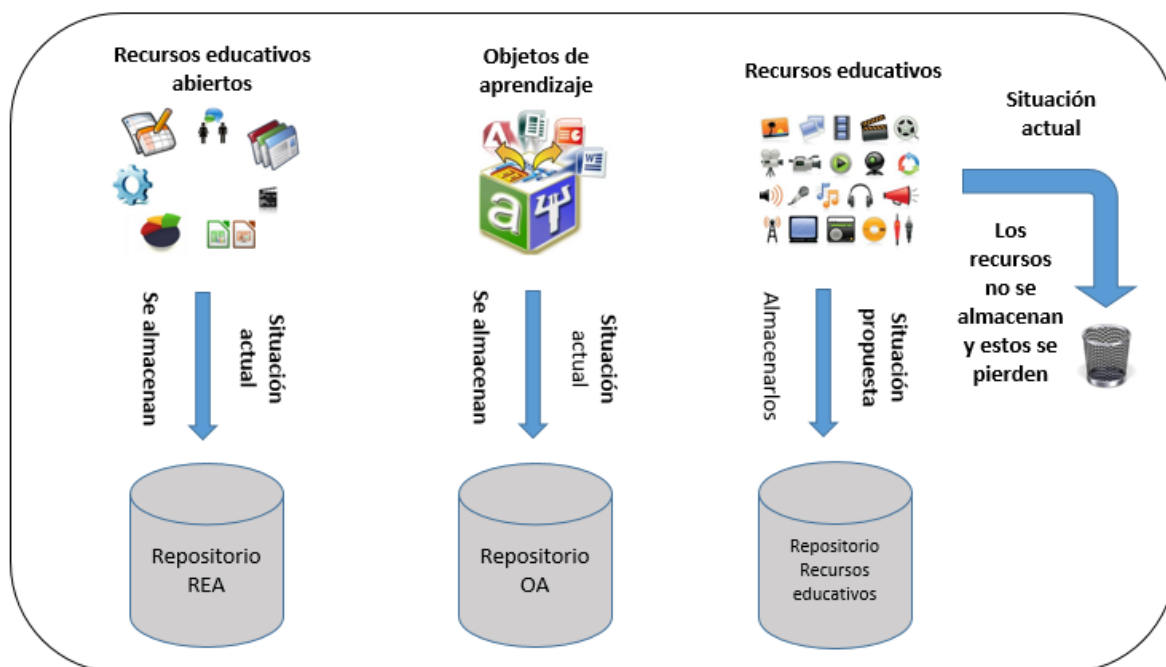


Figura 2. Situación actual de los recursos educativos abiertos, objetos de aprendizaje y recursos educativos

Como se aprecia en la imagen anterior, la situación actual de los recursos educativos por lo general termina en que estos se pierden. Por lo que la situación propuesta para estos recursos educativos es poder almacenarlos para que estos después puedan ser recuperados a través de un sistema de búsqueda implementado en un repositorio de recursos educativos.

1.2 Definición de la problemática de la investigación

Hoy en día se cuenta con diversas técnicas para la recuperación de recursos educativos en diversos sistemas de administración. Una alternativa para la recuperación de recursos educativos es la utilización de repositorios de recursos educativos, ya que con la implementación de un buscador, los recursos pueden ser recuperados utilizando metadatos.

Dados los avances tecnológicos en la actualidad, se ha incrementado la posibilidad de recurrir al aprendizaje a través del uso del internet. Muchas universidades cuentan con repositorios de recursos educativos para el almacenamiento y recuperación de los recursos educativos, ya sea que estos fueron desarrollados en la universidad o simplemente recuperados a través de internet. Estando estos disponibles a través de los servicios web que ofrecen las universidades en sus estudiantes.

Algunos de estos repositorios existentes en la actualidad son: Centro de recursos para la Enseñanza y el Aprendizaje (CREA) desarrollado por la Universidad de Guadalajara, el cual está orientado al fortalecimiento del proceso de enseñanza-aprendizaje y a la formación integral de los estudiantes de nivel superior y medio superior mediante el almacenamiento de objetos de aprendizaje (OA). Otro de los repositorios se llama UOC OpenCourseWare es un sitio a través del cual la Universidad Oberta de Catalunya (UOC) ofrece sus materiales docentes en la comunidad de internet: profesores, estudiantes y autodidactas. La Universidad Nacional de Educación a Distancia (UNED) también cuenta con un repositorio en línea. Dicho repositorio de la UNED se encuentra alojado en su portal que facilita la búsqueda de contenidos educativos abiertos que la misma UNED ofrece. Por último se encontraron dos repositorios más de la comunidad española, uno de ellos es el Proyecto Agrega, el cual es un proyecto de la administración públicas españolas y el sector privado creado con el objetivo de ofrecer a la comunidad educativa un repositorio de contenidos digitales curriculares en línea y el otro repositorio es Alejandría, un repositorio de materiales creado por el profesorado de los centros educativos catalanes.

Los repositorios mencionados solo son algunos encontrados en la literatura pero aún hay más de los mencionados. Es importante señalar que estos repositorios mencionados recuperan y almacenan únicamente recursos educativos abiertos, los cuales no son el objetivo de esta tesis sino la ofrecer la capacidad de recuperar y almacenar recursos educativos.

La siguiente lista de repositorios de recursos educativos que si permiten la recuperación y almacenamiento de recursos educativos, se encuentran algunos como los siguientes enlistados:

1. INTRODUCCIÓN

- Desarrollo de un repositorio de objetos de aprendizaje usando DSpace (Chazarra Bernabé, 2005)
- Repositorio de Objetos de Apoyo al aprendizaje colaborativo (González, L. Á., 2003)
- Desarrollo de repositorios de objetos de aprendizaje a través de la reutilización de los metadatos de una colección digital: de Dublin Core a IMS (Guzmán, C., 2005)
- Sistema Interactivo Distribuido de Repositorios de Objetos de Aprendizaje MatEmáticos (SINDROME) (López-Morteo, 2007)
- Un Repositorio basado en Servicios Web para el Sistema Generador de Ambientes de Aprendizaje AMBAR (López, M, 2008)
- Diseño de un repositorio de objetos de aprendizaje implementado con servicios web (Montilva, J., 2010)
- Diseño de repositorios de objetos de aprendizaje como estrategia de reutilización e integración de contenidos en modelos de educación virtual (Rojas, M, 2013)

Los repositorios de recursos educativos mencionados ofrecen la capacidad de recuperación y almacenamiento de recursos educativos, pero además de esto se encontró una problemática dentro de estos tipos de repositorios. En la literatura se encontró que las arquitecturas implementadas de repositorios de recursos educativos únicamente cubren los recursos educativos del tipo documento de texto (Hernández, 2001, Macano, 2008), es decir no soportan la recuperación o el almacenaje de otro tipo de recursos educativos como los repositorios de los REA (recursos multimedia e hipermedia)

Los recursos educativos abiertos, tienen la característica de que son recuperados y almacenados en diferentes formatos. Los formatos con los que se trabajan los recursos educativos abiertos son: documentos de texto, audios, videos, imágenes, animaciones y videojuegos educativos. Estos formatos como se mencionó anteriormente no son soportados por los repositorios de recursos educativos y esta es una de las problemáticas encontradas en la literatura.

Adicionalmente a la problemática principal mencionada anteriormente considerando los diferentes formatos posibles en los que se pueden almacenar los recursos educativos (video, audio, documento de texto, imágenes, aplicaciones, video juegos) se encontró en la literatura que no se ha implementado en los repositorios de recursos educativos la capacidad para poder visualizar dichos recursos en línea (Álvarez Et al, 2011), es decir que el usuario del repositorio pueda ver el video en línea o que pueda escuchar un recurso guardado en formato de audio o ver las imágenes sin la necesidad de tener que descargar el

recurso en cuestión por lo que también se tomará en consideración como parte de la problemática.

Con los avances de la tecnología móvil, hoy en día los dispositivos móviles tienen la facilidad para poder ingresar a navegar la web con mucha facilidad, pero aún siguen existiendo muchos inconvenientes durante la navegación desde un dispositivo móvil. La capacidad hardware de los dispositivos móviles actuales es bastante alta comparada con la de hace varios años. Dichos móviles el día de hoy cuentan con procesadores que superan las capacidades de muchas computadoras del año 2005, pueden ofrecer grandes capacidades como mostrar modelos 3D, o realizar cálculos muy largos o complejos. Pero aun así con todo es poder de procesamiento siguen existiendo varios inconvenientes cuando se navega por la web.

Los principales inconvenientes encontrados al navegar por la web tienen que ver con la incompatibilidad debido a la estructura de la página web y el tamaño de la pantalla del dispositivo móvil en cuestión (Hernández Et al, 2009). Al combinar estos elementos nos encontramos con páginas web que no pueden mostrar correctamente todos los elementos o componentes de la página debido a la pantalla del dispositivo que navega dicha página web. Este mismo problema se ve puede ver reflejado al querer usar un repositorio de recursos educativos desde un dispositivo móvil, y debido al uso de los dispositivos móviles se puede considerar como buena opción la implementación de alguna tecnología que permita visualizar correctamente todos los elementos del repositorio visitado desde cualquier dispositivo móvil.

1.2.1 Antecedentes de la investigación

Los repositorios de recursos educativos no es algo nuevo en estos días. Se han desarrollado diversas arquitecturas por todo el mundo. Actualmente muchas universidades cuentan con repositorios de recursos educativos propios. Según el directorio de repositorios abiertos (OpenDOAR, 2007), se encuentran registrados 853 repositorios académicos distribuidos alrededor del mundo. Se analizarán algunos repositorios para analizar las características que estos tienen.

Los autores Jonas Montilva, Ailin Orjuela y Mauricio Rojas realizaron un documento explicando la creación de un modelo repositorio de recursos educativos. Este documento tiene el nombre de DISEÑO DE UN REPOSITORIO DE OBJETOS DE APRENDIZAJE IMPLEMENTADO CON SERVICIOS WEB. En este documento los autores desarrollan los siguientes temas:

- Definición de objetos de aprendizaje
- Características de los objetos de aprendizaje
- La especificación SCORM
- Repositorios de objetos de aprendizaje (ROA)
- Tipos y características de los ROA
- Arquitecturas orientadas a servicios (SOA)
- Tecnologías que soportan a los servicios web
- XML (Extendible Markup Language)
- WSDL (Web Services Description Language)
- SOAP (Simple Object Access Protocol)
- UDDI (Universal Description, Discovery and Integration)
- Arquitectura de repositorio de objetos de aprendizaje con servicios web
- Descripción de las capas
- Funcionamiento de las capas
- Conclusiones

Durante el desarrollo de estos temas, los autores se adentran en cada uno de los temas para definir cuál será la estructura necesaria para la arquitectura de su repositorio. Los autores principalmente indagan en una característica especial. Esta característica es la llamada “Interoperabilidad”, gracias a esta característica los repositorios de recursos de aprendizaje ya no dependen de la plataforma en la que fueron creados. Es decir no importa si el sistema operativo (SO) es Windows, Linux o MAC, el repositorio tendrá la misma funcionalidad sin importar la plataforma en la que se use. Esto se debe específicamente gracias a la interoperabilidad lograda utilizando SOA.

Los autores implementaron un servicio web junto con una interfaz para que los usuarios puedan acceder a su repositorio a través de internet. Logrando así la deseada interoperabilidad, además de implementar las siguientes características para su modelo:

1. INTRODUCCIÓN

- Servicio de registro de recursos educativos
- Servicio de descarga de recursos educativos
- Servicio de búsqueda de recursos educativos

Los autores Juan Chazarra Bernabé, Víctor Manuel Requena López y Sergio Valverde Jerónimo publicaron un documento llamado DESARROLLO DE UN REPOSITORIO DE OBJETOS DE APRENDIZAJE USANDO DSPACE. En este documento se desarrollan los siguientes capítulos.

- Resumen
- Introducción
- Repositorios de contenidos digitales
- DSpace
- Configuración de Dspace
- Repositorio de material didáctico para los alumnos de informática
- Dificultades, conclusiones y trabajo futuro

El repositorio de recursos de aprendizaje creado por los autores es desarrollado usando una tecnología llamada Dspace. Este software permite crear repositorios de tecnología abierta ofreciendo diversas razones del porque es bueno utilizar dicha tecnología, tales como:

- Comunidad de usuario muy grande
- Es gratis
- Permite ajustes personalizables
- Es para todo tipo de institución
- Diferentes a otros tipos de repositorios de recursos educativos
- Permite almacenaje de cualquier tipo de contenido digital

Los autores comienzan con la personalización de la interfaz para que esta se ajuste a las necesidades del modelo que desarrollaron. Después continuaron con la creación de un esquema de metadatos que se ajuste a las necesidades de su modelo. Después crearon el servidor y por último montaron el repositorio en el servidor para que se pudiera acceder a las funcionalidades del repositorio.

Este modelo de repositorio ofrece dos grandes cualidades necesarias para otras arquitecturas de repositorios:

- Personalización de interfaz
- Personalización de esquema de metadatos

Estas dos características permiten una configuración muy personalizada de la arquitectura de repositorio lo cual facilita bastante el desarrollo de la misma, así como facilita que los usuarios accedan de una manera sencilla a los servicios ofrecidos por el repositorio de recursos educativos. Por ello es importante tomar en cuenta durante la creación del repositorio considerar las características mencionadas.

Para el almacenamiento de recursos educativos del tipo video, existen dos proyectos los cuales proveen la capacidad de poder almacenar videos en repositorios. El primero de ellos se llama “The open video digital library” y “Terabyte digital video library”.

“The open video digital library” se creó a partir de las siguientes características:

- Crear una biblioteca digital de video que pudieran ser descargados
- Crear sustitutos útiles para contenido de video
- Desarrollo de interfaces interactivas para los servicios digitales
- Desarrollar herramientas para que los usuarios finales puedan usar el video
- Desarrollar métodos y métricas para la evaluación del usuario interactuando con los sustitutos de video

“Terabyte digital video library” se concentró en cubrir los siguientes aspectos

- Extracción automática de información de videos digitalizados
- Creación de interfaces que permitieran a los usuarios la búsqueda y recuperación de videos
- Validar el sistema en base a un banco de pruebas del usuario

También existen documentos que hablan de otros tipos de recursos educativos, los archivos de tipo de audio. Los factores que abordan sobre los recursos educativos del tipo audio son:

- El costo de producción
- El método de publicación en una web con ambiente educativos

Con el avance de la tecnología los dispositivos móviles han ido cambiando el estilo de vida tanto de los adultos como de los niños a un nivel mundial. Por lo que se realizó un estudio sobre cómo utilizar las aplicaciones de dispositivos móviles para avanzar en el tema de aprendizaje. Existe un trabajo llamado repositorio GREDOS el cual se dedica a almacenar aplicaciones de dispositivos móviles. Este trabajo hace énfasis a los siguientes aspectos:

Estudio de los estándares de metadatos para describir los recursos educativos

- Almacenaje de metadatos en Dspace
- Creación de etiquetas visuales para nuevos metadatos
- Creación de plantillas de descripción en cada colección
- Almacenaje de recursos educativos en GREDOS de acuerdo a las competencias de los estilos educativos
- Descripción de las aplicaciones de dispositivos móviles en GREDOS

1.2.2 Relevancia de la investigación

A fin de aprovechar al máximo el potencial de los contenidos digitales, en el ámbito educativo ha surgido un nuevo concepto que está causando cambios radicales en la forma de conceptuar y hacer contenidos, al cual se le ha llamado recurso educativos. Este tipo de recursos tienen características particulares que dan capacidades y funcionalidades a los sistemas de gestión de aprendizaje, principalmente desde el punto de vista de la organización y reutilización de recursos. No se puede pensar en los repositorios sin estos tipos de recursos, ya que algunas de las características de estos repositorios están dadas por las características intrínsecas de los recursos educativos.

En primera instancia, el uso de los recursos educativos debe responder a una necesidad y a un deseo de mejorar las prácticas de la elaboración de recursos educativos. De esa forma, se pueden aprovechar producciones previas y que no se tengan que diseñar materiales que ya están elaborados.

La enseñanza en la virtualidad experimenta problemáticas como la "ausencia de unas metodologías técnicas, documentales y psicopedagógicas comunes y aceptadas que garanticen los objetivos, interoperabilidad, durabilidad y reutilización de los materiales curriculares basados en las redes". (Zapata, 2005). Tales problemáticas se originan por el modo en que se presentan los contenidos de un curso. Por tanto, un recurso educativo elaborado adecuadamente, puede facilitar, paulatinamente, mejoras en la educación virtual.

Muchos autores plantean que la idea de repositorio es intrínseca a los recursos educativos. No es posible pensar en recursos educativos si no se los concibe albergados en repositorios. Como recursos aislados no tienen ninguna relevancia ni significado real. Una manera de comprender los repositorios, es imaginar una combinación entre una biblioteca digital y un buscador como Yahoo o Google, pero mucho más sofisticado que ambos. Por una parte, los recursos son de naturaleza diversa (al contrario que en una biblioteca) y por otra parte, los criterios de búsqueda deben considerar bastante más que títulos, autores o palabras claves. El tipo de componentes albergados en un repositorio, que deben tener sus propias identidades y ser por lo tanto localizables, son tan variados como gráficos, imágenes, textos, "applets", videos, documentos e integración de ellos como capítulos de un curso o hasta cursos completos. Un aspecto muy importante de los repositorios es que no necesariamente albergan físicamente los recursos que contienen; les basta con "apuntar" a ellos.

El repositorio de recursos educativos está ligado directamente con la naturaleza de un recurso educativo. De hecho, no se visualizan de forma independiente, sino que la existencia del uno justifica la del otro. El repositorio almacenará de forma digital los objetos de aprendizaje, de forma debidamente organizada, ya que la búsqueda debe ser sencilla.

La importancia de un banco o repositorio de recursos educativos radica en el acceso y que se puedan encontrar los recursos para una determinada temática. Muchas veces la información existe, pero la falta de etiquetas o una correcta identificación limitan que los objetos elaborados de manera previa sean aprovechados.

El repositorio de recursos educativos sirve para que las instituciones puedan compartir recursos y así puedan ahorrar costos en la elaboración de material didáctico digital. Para

ello, "es necesario generar un almacén de los recursos educativos con una herramienta de búsqueda que permita una localización fácil, rápida y ajustada a descriptores de interés para el usuario; esto es lo que se conoce como repositorio". (Martínez et Al, 2007).

1.3 Justificación de la investigación

En la actualidad existe una gran cantidad de contenido educativo ya sea físico y/o digital. El contenido educativo es usado para propósitos de aprendizaje ya sea por las personas o por docentes que utilizan dicho contenido educativo para impartirlo a los interesados en aprender del dicho contenido.

Hablando más específicamente sobre los contenidos educativos digitales, existen los recursos educativos los cuales siendo derivados del contenido educativo, tienen la misma finalidad. Estos recursos aprendizaje los podemos encontrar ya sea en alguna base de datos o esparcidos a través de internet o los mismos recursos educativos que se generan en las universidades y podemos acceder a ellos descargándolos desde el mismo repositorio.

Los recursos educativos se pueden encontrar en diferentes formatos. Existen diversos formatos en los que podemos encontrar algún recurso educativo como: documento de texto, imágenes, aplicaciones, videos, audios o juegos serios. Todos estos tipos de archivos están en las siguientes categorías (Hernández, 2001, Marcano, 2008).

- Audio
- Video
- Documento
- Imagen
- Juegos serios

Existe una vasta variedad de recursos educativos de cada categoría. Muchas páginas de internet ofrecen acceso a sus colecciones de recursos educativos usando un repositorio de recursos educativos.

Se han creado diversos modelos de repositorios de recursos educativos que permiten la recuperación de los recursos educativos utilizando metadatos como recurso descriptivo de los recursos educativos. Otros modelos definen la necesidad de una interfaz que facilite la interacción entre el usuario y el repositorio de recursos educativos. En si diversos modelos ofrecen diferentes características para sus repositorios de recursos educativos pero no hay un modelo que contenga un conjunto de características que permita al usuario visualizar los recursos educativos sin descargarlos, que el repositorio ofrezca interoperabilidad o que ofrezca la posibilidad de dar poder contener diversos tipos de archivos para los recursos educativos.

Las universidades hoy en día cuentan con repositorios de recursos educativos para que sus alumnos tengan acceso a diversos materiales de aprendizaje. Los repositorios institucionales como lugar para la organización, preservación y difusión de la producción digital de las universidades es un tema candente en la actualidad. Cada vez son más las universidades que cuentan con repositorios de recursos educativos y abren sus repositorios a nivel tanto nacional como internacional (Barrueco, 2009). Este panorama en plena transformación y potenciación merece que por un momento fijemos en ellos nuestra atención, examinando detalladamente que características conllevan, en qué circunstancias están creciendo y con qué singularidades.

De esta manera se puede orientar las próximas creaciones de repositorios de forma que verdaderamente sirvan para que la producción científica y académica de los miembros de una institución pueda ser gestionada, organizada y preservada y que por un medio de acceso libre a la misma, se pueda promocionar tanto en beneficio de la cultura, como del propio investigador y de la institución que lo soporta.

Debido a lo mencionado anteriormente es la razón por la cual se desea abordar en el mundo de los repositorios de recursos educativos para desarrollar una arquitectura que cumpla con las características mencionadas y resuelva las problemáticas abordadas en el documento.

1.4 Descripción del reporte de tesis

El documento en el que se recoge el resultado de la investigación llevada a cabo consta de 7 capítulos, estructurándose tal y como se describe a continuación.

En el **Capítulo 1** que incluimos en este apartado, se incluye una introducción con la motivación y justificación de la investigación que se ha realizado, así como se incluye algunos antecedentes a la investigación y la descripción de la problemática que involucra la investigación.

En el **Capítulo 2** se desarrolla la formulación de la investigación, así como los objetivos y preguntas de investigación a responder que motivaron al desarrollo de esta tesis.

En el **Capítulo 3** se comienza el desarrollo del “estado del arte” del tema de investigación. Este apartado también incluye información los contenidos de aprendizaje, los recursos educativos y los tipos de estos. Se habla también sobre el significado de los metadatos para los recursos educativos, así como también los métodos para la recuperación de dichos recursos usando servicios web. Se aborda también el tema correspondiente a los repositorios de recursos educativos así como los estilos de arquitectura en los cuales se planea desarrollar la arquitectura y por ultimo una introducción al tema de la visualización de los recursos en línea y la definición de la misma.

En el **Capítulo 4** se habla sobre las metodologías a implementar para el desarrollo del repositorio de recursos educativos así como los lenguajes de programación que se planean utilizar. También se aborda las tecnologías disponibles para utilizar como base de datos y que tecnología se planea implementar. Por último se trabaja en el diseño de la arquitectura a implementar en el repositorio de recursos educativos, sus requerimientos, comparación con otras arquitecturas que solucionan parcialmente el problema y una justificación de la arquitectura que se diseña.

En el **Capítulo 5** se incluye una validación de la arquitectura usando un prototipo real, así como la justificación de los objetivos y servicios. Se habla también de la implementación de la arquitectura, el diseño de la base de datos, las pruebas que se le harán al repositorio de recursos educativos y una conclusión del tema.

En el **Capítulo 6** se abordan las conclusiones de la investigación, los objetivos y las aportaciones de así como también las líneas de investigación futuras.

El **Capítulo 7** contiene las referencias bibliográficas citadas a lo largo de la tesis.

2 Formulación de la investigación

La investigación aborda en este documento tesis tiene que ver con los repositorios de recursos educativos tal y como se mencionó en el capítulo anterior. Esta investigación comienzo debido a que la Universidad Autónoma de Aguascalientes genera recursos educativos y no se tiene un modelo para el almacenamiento de estos recursos. Además del almacenamiento d estos recursos también es importante mencionar que no se tiene un método para poder compartir dichos recursos educativos con la comunidad de la universidad. Dado esto se comenzó a investigar las diferentes formas para almacenar, recuperar y compartir los recursos educativos generados por la universidad. Durante la investigación se encontró que los repositorios de recursos educativos son una buena opción para realizar esta actividad y por ello se comienza con la investigación.

En ese capítulo veremos las preguntas de investigación alineados a las problemáticas abordadas en este documento, así como también veremos los objetivos de la investigación.

2.1 Preguntas de la investigación

Dada la problemática aborda en este documento las preguntas de la investigación circundaran temáticas como: la generación de arquitecturas, definición de tecnologías a implementar para el desarrollo de repositorios de recursos educativos como: tecnologías referentes a bases de datos, y lenguajes de programación. Se considera también los diferentes tipos de formatos en los que encontramos los diferentes tipos de recursos educativos. También se considera la idea de definir o generar una tecnología que permita la visualización de recursos educativos desde el mismo repositorio (archivos de audio, video, imágenes, documentos de texto), así como también se considera que la interoperabilidad es necesaria para el desarrollo de repositorios de recursos educativos.

Dentro las preguntas de investigación también es necesario considerar los temas relacionados a los diferentes estilos arquitectónicos disponibles para el desarrollo de la arquitectura del repositorio que se pretende desarrollar y por ultimo considerar los

estándares existentes de metadatos para poder definir como almacenar los recursos educativos en el repositorio de recursos educativos.

2.1.1 Pregunta general de la investigación

La pregunta general de la investigación se define a continuación:

- ¿Qué elementos deberá contener una arquitectura en capas orientada a servicios para que un repositorio de recursos educativos soporte el almacenaje/recuperación de recursos educativos en múltiples formatos?

2.1.2 Preguntas específicas de la investigación

Las preguntas específicas de la investigación se definen a continuación:

- ¿Qué tecnología se deberá utilizar para implementar una arquitectura en capas orientada a servicios para el desarrollo de un repositorio de recursos educativos se logre utilizando software libre?
- ¿Qué elementos debería contener el diseño arquitectónico planteado como solución?
- ¿Qué elementos deberá contener una estructura de metadatos simple para el almacenaje de recursos educativos en un repositorio?

2.2 Objetivos de la investigación

Los objetivos de la investigación deben girar en torno a las preguntas de la investigación. Condicionando esto, logramos que los objetivos de la investigación logren satisfacer la problemática abordada en el documento.

2.2.1 Objetivo general de la investigación

El objetivo general de la investigación se define a continuación:

- Desarrollar una arquitectura en capas orientada a servicios a un repositorio de recursos educativos que permita la recuperación y almacenaje de recursos educativos de diferentes formatos de archivos.

2.2.2 Objetivos específicos de la investigación

Los objetivos específicos de la investigación se definen a continuación:

- Analizar las tecnologías de software libre existentes que permitan la implementación de una arquitectura en capas orientada a servicios en el desarrollo de un repositorio de recursos educativos y seleccionar una que se adecue a los propósitos de esta tesis.
- Definir los elementos arquitectónicos que deberá contener el diseño que funcionara como solución al problema planteado.
- Definir una estructura de metadatos para el almacenaje de recursos educativos en un repositorio.

3 Marco Teórico

En este capítulo se abordaran los temas concernientes al tema de tesis. Se relata sobre los contenidos educativos, los recursos educativos y su importancia. También veremos información acerca de los metadatos y su importancia para los recursos educativos. Se habla también sobre los servicios web y la importancia de estos en los repositorios de recursos educativos. Más adelante veremos la definición de los repositorios de recursos educativos, su importancia y los tipos de repositorios de recursos educativos. Por ultimo para el desarrollado de la arquitectura del repositorio que se propone en este documento de tesis se habla sobre diferentes estilos arquitectónicos, las arquitecturas que se analizaron para el desarrollo de la arquitectura propuesta y la definición sobre la visualización de recursos educativos.

3.1 Contenido educativo

Los contenidos educativos han logrado un mayor esparcimiento desde que el internet se volvió más comercial. Esto se debe a que cualquier persona que desarrolla contenido educativo lo comparte a través de internet. En este capítulo se habla sobre los contenidos educativos y su definición. Además se presenta una clasificación de los diferentes tipos de contenidos educativos. Más adelante se presenta algunas de las definiciones sobre los recursos educativos y la importancia de los mismos.

Los contenidos educativos digitales, son materiales de carácter didáctico, basados en la investigación documental, experiencial o de ambas fuentes, originados del tratamiento pedagógico de la temática seleccionada y constituida en guion, instruccional para su conversión en formato multimedia.

Desde el punto de vista técnico, los contenidos educativos son unidades de significado en formato multimedia (video, audio, texto e imagen) o hipermedia (multimedia interactivo), que estructuradas en objetos temáticos cumplen con el propósito informativo y didáctico (Hernández M., 2011).

Cabe mencionar que en la actualidad los avances tecnológicos aparecen para dar respuesta a las necesidades de estar en continua conexión con la información y las comunicaciones. El uso de dispositivos móviles en educación es un elemento fundamental en la construcción de conocimiento, ya que con la utilización de estas tecnologías se incrementan las posibilidades de interactuar con los miembros del grupo, se mejora la comunicación; por lo tanto, se difumina la barrera que separa a los docentes y discentes. La tendencia actual hacia el uso de dispositivos móviles en educación está enfocada a que, en el futuro, cada vez más se utilicen estos aparatos en las aulas y en los centros educativos y culturales (Valero Et al, 2012). Todo material utilizado para la educación se puede considerar como contenido educativo, para ser más específico recursos educativos. Por ello se presentan las siguientes definiciones para recursos educativos.

3.1.1 Recursos educativos

Abordar una definición de recurso educativo, es un tanto complicado, dado que existe una amplia discusión respecto del término y más. Si tomamos en cuenta que el término ha estado evolucionando y adaptándose a la necesidad educativa y tecnológica. Aun así se toman en cuenta las siguientes definiciones encontradas en la literatura. “Los recursos educativos se definen como cualquier entidad, digital o no digital, que puede ser utilizada, reutilizada o referenciada durante un aprendizaje apoyado en esta tecnología” (Islas, 2010). Otra definición es “Una unidad independiente y auto contenible de contenido educativos que está predispuesto a ser reutilizado en múltiples contextos instruccionales” (Polsani, 2006). Otra definición dice que “cualquier recurso digital que pueda ser usado para ofrecer soporte al aprendizaje” (Wiley, 2003), esta definición incluye cualquier objeto digital que pueda ser esparcido a través de la red, ya sea pequeño o grande.

Como ejemplos de recursos educativos se incluyen los contenidos multimedia, el contenido instruccional, los objetivos educativos, el software instruccional y las herramientas de software. Así como documentos digitales en los formatos anteriormente mencionados más las aplicaciones para celulares.

La promesa más poderosa que ofrecen las tecnologías de la información es su habilidad para capturar conocimiento tal que este pueda ser analizado, reusado, compartido por otros y usado para crear nuevo conocimiento (Hodgins, 2002).

Hoy en día constantemente se genera información, e internet es hoy una gran fuente de información, se sabe esta llegar a ser tal, gracias a la organización y ensamblaje oportuno y adecuado de los datos. Estos datos, debidamente combinados producen unidades de

información, pero combinados de otra manera o con otros datos podrían llegar a generar unidades diferentes (Aretio, 2005).

.Las tecnologías de recursos educativos han sido ampliamente utilizadas para apoyar las iniciativas de educación digital y online en el ámbito internacional, y muchos las consideran la piedra angular para proyectos amplios de desarrollo y para la adopción del e-learning a través de internet (Alvares Et al, 2011).

Un recurso educativo es caracterizado por la creencia de que podemos crear trozos de contenido educacional independientes. Que pueden proveer experiencia educacional para algunos propósitos pedagógicos (Quinn Et al, 2000).

En conclusión los recursos educativos pueden ser cualquier entidad digital o no digital que facilite el aprendizaje al usuario. Estos recursos educativos es importante recopilarlos y almacenarlos en un repositorio de recursos educativos para que estos puedan ser reutilizados. Existen diferentes formatos en los que podemos encontrar los recursos educativos, entre ellos encontramos juegos, documentos, videos, imágenes y audios, pero esto lo veremos en el siguiente apartado.

3.1.2 Formatos de recursos educativos

La gran diversidad de los recursos educativos permite que los recursos educativos se puedan encontrar en diversos formatos. Los diferentes formatos en los que podemos encontrar los recursos educativos son los siguientes (Hernández, 2011, Aretio, 2005):

- Documentos: Estos recursos educativos usualmente se encuentran en formatos de documentos de texto, presentaciones y por ultimo también los encontraremos en formatos PDF.
- Audio: Estos recursos educativos son aquellos archivos de audio, usualmente en formato MP3 que contienen algún conocimiento, grabado por algún dispositivo de voz.
- Video: Estos recursos educativos son aquellos archivos de video los presentan algún curso, demostraciones, o explicación sobre algún tema específico.
- Imagen: Estos recursos educativos suelen ser imágenes que nos demuestran algún diagrama, alguna explicación o simplemente ejemplos de algún tema en específico.

- Aplicaciones: Estos recursos educativos son aquellas aplicaciones de escritorio o de dispositivos móviles. Entre este tipo de recursos educativos podemos encontrar la categoría de juegos serios.

Los diferentes formatos en los que encontramos un recurso educativo nos permite tener una mayor variedad de recursos en el repositorio, por lo que se consideran estos cinco formatos para que estos sean almacenados en el repositorio.

3.2 Metadatos

Los recursos educativos son almacenados en repositorios, esto lo hemos mencionado durante esta tesis, pero una vez almacenados se necesita una manera para que estos puedan ser diferenciados unos de otros, es decir tener una clave única para estos puedan ser identificados de manera única, para ello se emplean los metadatos.

Los metadatos ofrecen una facilidad para la recuperación de los recursos educativos. Además estos ofrecen la posibilidad de que el objeto de aprendizaje perdure y pueda ser reutilizado, por ello es importante tener en claro que son los metadatos y para que nos sirven. En este capítulo se habla sobre los metadatos. Se ofrece una definición sobre el tema en cuestión. Más adelante se habla sobre los diferentes estándares que existen para definir la estructura de los metadatos de los recursos educativos y se presentan los elementos que existen los estándares ya definidos.

El término de metadato se refiere a la información de la información o, su equivalente datos sobre datos (Amy Brand 2003). En la práctica, el término significa información estructurada que alimenta un proceso automático. Y esta es la mejor forma de darse una idea de que significa metadato.

Para ilustrar mejor que son los metadatos, es necesario ver un estándar de metadatos simple, el llamado “Dublin Core”. La iniciativa “Dublin Core Metadata Initiative” (DCMI) comenzó en 1995 como una unión de un esfuerzo entre profesionales de comunidades de publicadores, bibliotecarios y académicos. El resultado de este esfuerzo fue “The Dublin Core Metadata Element Set” el cual se convirtió en un estándar de NISO en 2001 (ANSI/ISO Z39.85-2001) y en un estándar internacional (ISO 15836) en 2003.

El estándar DCMI se puede apreciar en la figura 3:

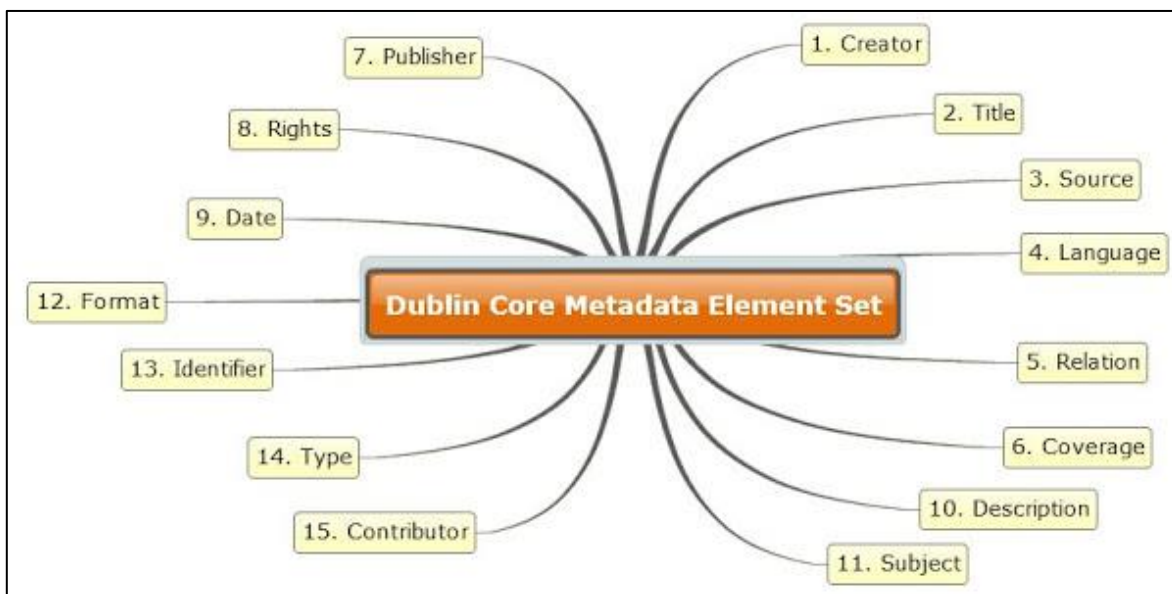


Figura 3. Esquema de elementos del estándar DCMI (ANSI/ISO, 2001)

Otro estándar para metadatos que se mencionara es el llamado “Learning object Metada” (LOM). El estándar LOM permite describir las características de cualquier objeto educativos, digital o no, mediante una serie de metadatos agrupados en nueve categorías (CanCore, 2004). Los metadatos más empleados en la práctica son de tipo genérico: título, descripción del objeto, formato, idioma... Estos coinciden con los elementos del estándar DCMI en cambio los elementos de LOM específicamente educativos se encuentran entre los menos utilizados. Los elementos del estándar de metadatos LOM se pueden observar en la figura 4.

Como podemos observar ambos estándares son bastante complejos y completos, ya que reúnen bastantes características para poder describir y almacenar los REA u objetos de aprendizaje (OA). Pero dado que para el desarrollo del repositorio de esta tesis no utilizaremos REA u OA no es necesario un estándar de metadatos tan complejo como estos. Ya que los recursos educativos no llevan un proceso de desarrollo tan complejo como los OA o REA por lo que la estructura de metadatos que se implementara será algo más sencillo y simple. Los elementos de metadatos que se utilizaran para almacenar los recursos educativos en un repositorio de recursos educativos serán los siguientes:

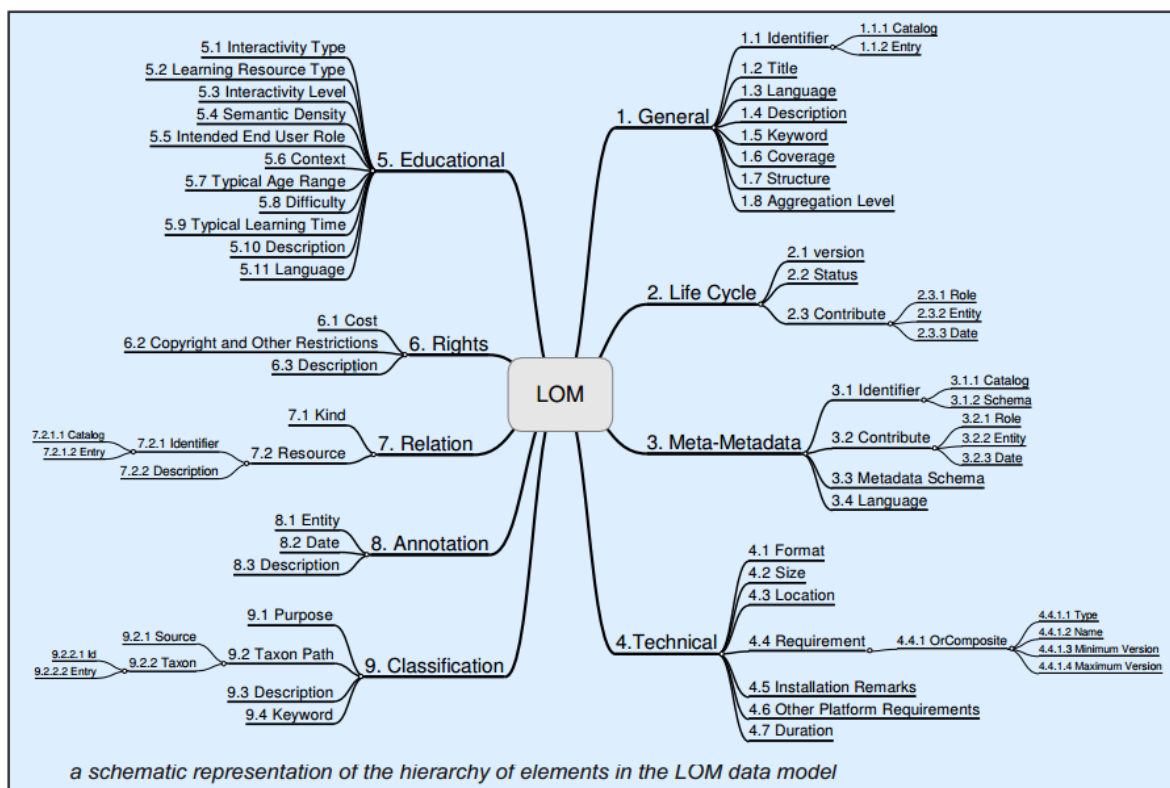


Figura 4. Esquema de elementos del estándar LOM (CanCore, 2004)

- Autor
- Título
- Descripción
- Tema de cobertura
- Formato del recurso
- Palabras clave

En la figura 5 se muestra el esquema de metadatos a implementar

Por conclusión en cuanto los diferentes esquemas o estructuras de metadatos, se tiene que no es necesario utilizar una estructura de metadatos tan compleja. Esto se debe principalmente a que los recursos educativos almacenados pueden ser recuperados con una implementación de un sistema de búsqueda simple y no será necesario tener un sistema de búsqueda de recursos educativos tan complejo.

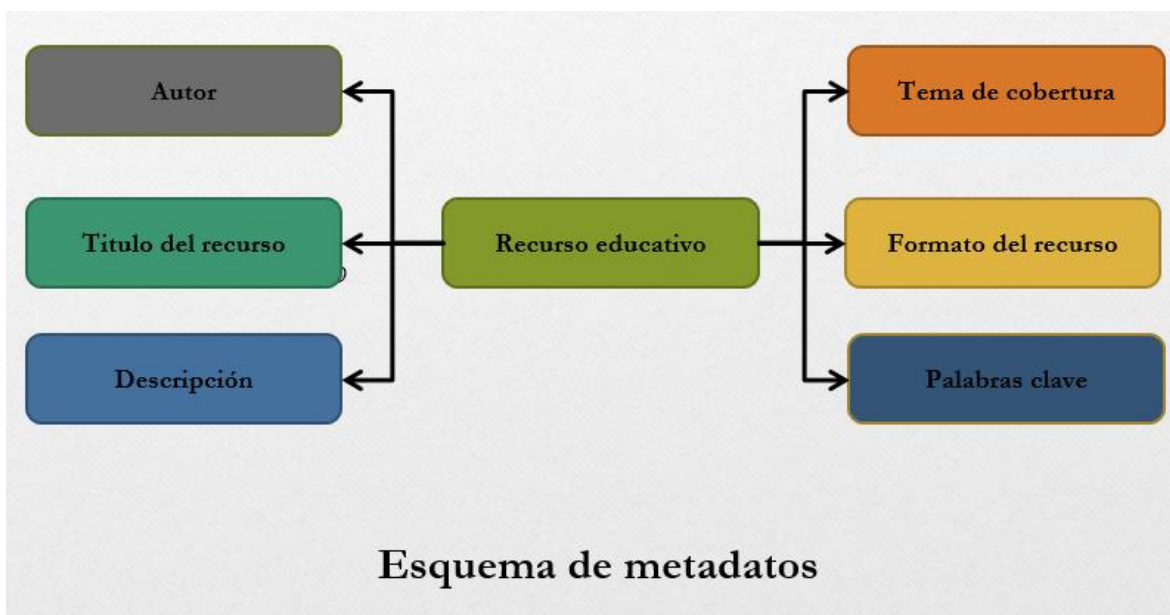


Figura 5. Esquema de metadatos propuesto

3.3 Servicios Web

En las tecnologías de la información existen las tecnologías orientadas a servicios. Estas tecnologías ofrecen la posibilidad de generar servicios a través de internet que no dependan de una tecnología única y que puedan operar bajo diversas plataformas. Esta capacidad nos ofrece la posibilidad de la interoperabilidad entre plataformas. En este capítulo se habla sobre los servicios web. Se inicia con la definición de servicio web. Después se habla sobre la clasificación de los servicios web actuales. Más adelante se definen los tipos de servicios web, se desarrolla el tema y ofrece una lista de fortalezas que ofrecen dichos servicios web y se concluye con la elección de cual tecnología es la adecuada a utilizar para el beneficio del proyecto.

La World Web Wide Consortium lo define como “... un sistema de software diseñado para soportar interacción interoperable maquina a máquina sobre una red” (Morales, 2005). Otra definición de servicios web es un lenguaje agnóstico basado en estándares que acepta solicitudes en formatos especiales de otras entidades de software de máquinas remotas a través de un proveedor y transporte de comunicación neutral de protocolos, produciendo respuestas de aplicaciones (Ioaniss, 2005).

Los servicios web pueden ser implantados de diversas maneras. Los servicios web pueden ser clasificados como “Big Web Services” y “RESTful Web Services” (Pautasso, 2008).

3.3.1 Big web services

Los “Big Web Services” están basados en Simple Object Access Protocol y generalmente contiene Web Services Description Language para describir la interface que el servicio web ofrece.

Proporcionar interoperabilidad entre la tecnología Middleware Stack Ks y el consumo de servicios (solicitante. cliente) son los principales objetivos de la arquitectura Service Oriented Architecture (SOA) y Web Services Technologies.

En un nivel conceptual un servicio es un componente de software proveído a través un punto final de una red accesible (Pautasso et Al, 2008). El cliente que consume el servicio y el proveedor del mismo intercambian mensajes para solicitar y recibir información en forma de documentos auto contenidos.

Dado que los “Big Web Services” están basados tanto en Simple Object Access Protocol y Web Services Description Language, es necesario definir estos conceptos y conocer más a fondo sobre ellos.

3.3.2 Web services description language

Web Services Description Language (WSDL) es un formato basado en XML para describir un servicio web. Los clientes que desean acceder a un servicio web pueden leer e interpretar el archivo WSDL para aprender acerca de la localización del servicio y las operaciones disponibles. De esta manera la definición de WSDL funciona como la interfaz inicial del servicio web, proveyendo a los clientes con toda la información para interactuar con el servicio (Cavanaugh, E., 2006). Cabe mencionar que un documento WSDL que describe un servicio web actúa como un contrato entre el cliente y el proveedor del servicio web.

Un archivo WSDL es un documento XML que describe un servicio web usando 6 elementos principales:

- Tipo de puerto: Agrupa y describe las operaciones realizadas por el servicio a través de una interfaz definida
- Puerto: Especifica una dirección para el enlazamiento
- Mensaje: Describe los nombres y el formato del mensaje que son soportados por el servicio
- Tipos: Define los tipos de datos (tal y como se definen en un documento XML) usados por el servicio para el envío de mensajes entre cliente y servidor.
- Enlazamiento (Binding): Define los protocolos de comunicación soportados por las operaciones provistas por el servicio.
- Servicio: Especifica la dirección (URL) para acceder al servicio.

3.3.3 Simple object access protocol

Simple Object Access Protocol (SOAP) es un protocolo basado en XML de W3C para intercambiar información a través de HTTP. Provee un método simple para el intercambio de mensajes entre aplicaciones. Los servicios web usan SOAP para el envío de mensajes entre servicio y cliente. Esto es dado HTTP es soportado por todos los servidores web y exploradores de internet, los mensajes SOAP pueden ser enviados entre aplicaciones sin importar la plataforma o el lenguaje de programación. Esta cualidad (enviar mensaje sin importar la plataforma o lenguaje de programación) provee la característica de interoperabilidad (Cavanaugh, E., 2006).

A nivel tecnología SOAP es un lenguaje XML que define la arquitectura y el formato de mensajes. El documento SOAP define un elemento XML a un alto nivel un “sobre” (Envelope) que contiene un “encabezado” (Header) y un “cuerpo” (Body). El “encabezado” SOAP es un contenedor extensible que sirve de infra estructura para la capa mensaje que puede ser usado para el enrutamiento (dirección) y para la configuración de QoS (Quality of Service). El cuerpo (Body) contiene los datos del mensaje. El esquema XML es usado para describir la estructura de del mensaje SOAP, para que los motores SOAP en ambos puntos de la conexión puedan descifrar el contenido del mensaje y direccionarlo apropiadamente (Pautasso et Al, 2008).

3.3.4 RESTful web services

REpresentational State Transfer (REST) fue originalmente introducido como un estilo de arquitectura para la construcción a gran escala sistemas distribuidos hipermedia. Este estilo de arquitectura es mejor dicho una entidad, cuyos principios han sido usados para explicar

la excelente escalabilidad del protocolo HTTP 1.0 por lo tanto el termino REST es usado frecuentemente en conjunto con HTTP.

La arquitectura REST está basada en 4 principios:

- Identificación de recursos a través de URI: Un servicio web RESTful expone un conjunto de recursos para identificar los objetivos de la interacción con sus clientes.
- Interface uniforme: Los recursos son manipulados ando un conjunto fijo de cuatro operaciones, crear, leer, actualizar y borrar (PUT, GET, POST, DELETE). PUT crea un recurso nuevo que puede ser borrado usando DELETE. GET recupera el estado actual del recurso y POST transfiere un nuevo estado a un recurso.
- Mensajes auto descriptivo: Los recursos son separados de su representación para que se pueda acceder al contenido en diversos formatos (HTML, XML, texto plano, PDF, JPG, etc.).
- Interacciones a través de los hipervínculos: Cada interacción con un recurso permanece sin estado, por ejemplo los mensajes solicitados están auto contenidos. Las interacciones están basadas sobre el concepto de transferencia explícita de estado. Existen diversas técnicas para cambiar el estado por ejemplo: re escribir una URI, cookies, y campos ocultos.

3.3.5 Comparación de “Big web services” vs “RESTful web services”

Se presenta la siguiente lista de puntos fuertes de cada una de las tecnologías (Pautasso et Al, 2008).

“Big Web Service”

- El formato de mensaje SOAP y la definición de la interfaz WSDL ha ganado una adaptación amplia
- Transparencia e independencia en el protocolo
- El mismo mensaje en el mismo formato puede ser enviado a través de una variedad de sistemas middleware
- Se pueden modelar interfaces de servicio para sistemas con patrones basados en interacciones asíncronas y síncronas

- Motores maduros SOAP y herramientas WSDL efectivamente esconden la complejidad percibida de la aplicación

“RESTful Web Services”

- Son percibidos por su simpleza dado que existen estándares muy bien conocidos (W3C, IETF, HTML, XML) y la infraestructura necesaria es muy bien conocida.
- Los clientes y servidores HTTP están disponibles para la mayoría de los lenguajes de programación y los diferentes sistemas operativos (interoperabilidad) y el puerto por defecto de HTTP 80 es comúnmente dejado abierto en la mayoría de las configuraciones de los firewall.
- El esfuerzo requerido para construir un servicio web REST es mínimo dado que los programadores pueden comenzar las pruebas desde un explorador web ordinario sin tener la necesidad de desarrollar un software para realizar las pruebas.
- Dada la posibilidad de elegir formatos de mensaje poco pesados por ejemplo JavaScript Object Notation (JSON) o en un extremo texto plano para tipo de datos simples. REST ofrece una gran libertad de acción para optimizar el desempeño de los servicios web.
- Dada la infraestructura poco pesada, los servicios web pueden ser construidos con pocas herramientas, es costo es relativamente bajo y por lo tanto las barreras para la adaptación son muy bajas.

Una vez analizados los puntos fuertes de dos tipos de servicios web, debido a la sencillez de los servicios web REST, es más recomendable utilizar este tipo de servicios web. Dado que no habrá necesidad de desarrollar software para poder realizar pruebas y las herramientas para desarrollar son de nulo o bajo costo.

3.4 Repositorios de recursos educativos

Los repositorios de recursos educativos nos facilitan el almacenaje de los recursos educativos. Si a esto sumamos la posibilidad de integrar un sistema de búsqueda a nuestro almacenaje, obtenemos un servicio de búsqueda de recursos educativos. Existen diversas tecnologías que se pueden utilizar para creación de repositorios por ello, durante este capítulo se desarrolla el tema de repositorios. Se ofrece una serie de definiciones. Más adelante se habla sobre la importancia de los repositorios de recursos educativos. Se clasifican los tipos de repositorios y se mencionan las diferentes tecnologías y lenguajes de programación disponibles para la creación de los repositorios y se concluye con la elección

de una tecnología y lenguaje de programación. Se ofrece una serie de fortalezas para cada tecnología y lenguaje de programación mencionado y se concluye con la elección de la tecnología que beneficiara al proyecto.

Un repositorio digital es un sistema que “permite el almacenaje, descubrimiento y recuperación de metadatos y/o de objetos electrónicos/digitales almacenados a un nivel distribuido o local” (The JORUM Team, 2006, p. 8). Más específicamente un repositorio de recursos educativos (Learning Object Repository (LOR)) es un sistema que administra el acceso a contenido de aprendizaje reutilizable (Monge, S. Et al, 2008). Sin embargo los límites de lo que puede ser llamado LOR permanece borroso dado que no existe un acuerdo común acerca de la naturaleza de los recursos educativos (McGreal, 2004).

3.4.1 Importancia de los repositorios de recursos educativos

Se han hecho grandes inversiones internacionales para el desarrollo y el despliegue de los repositorios para el sector académico en años recientes. Por ejemplo, en julio del 2002, \$9.4 millones fueron invertidos en el programa e-learning de EduSourceCanada. En marzo del 2005, Joint Information Systems Committee (JISC) solicitó diseños de repositorios digitales e invirtió £4 millones.

Actualmente los repositorios son usados comúnmente para guardar contenido educativo digital por lo que el desarrollo de esta tecnología es importante. En teoría los repositorios facilitan tanto el almacenaje más eficientemente como la administración de los recursos. Los repositorios permiten a los usuarios compartir recursos y descubrir recursos compartidos por otros (Lorna M et Al, 2004).

3.4.2 Tipos de repositorios de recursos educativos

Los recursos educativos son agrupados y almacenados en repositorios de recursos educativos (Guzmán Et al, 2005). De los repositorios de recursos educativos existen dos tipos (Downes, 2002)

- Contenedores de recursos educativos y metadatos
- Contenedores de metadatos

Los primeros contienen los recursos educativos, y sus descriptores se encuentran dentro del mismo sistema e incluso dentro de un mismo servidor. Los segundos el repositorio contiene solo los descriptores de los recursos educativos y se accede el recurso educativo a través de una referencia a su ubicación física. Esta referencia se encuentra en otro repositorio o en el mismo.

3.5 Estilos de arquitectura

En los inicios de la programación, los programadores se dieron cuenta que escribir código se convertía en una tarea cada vez más y más compleja. Cuando los investigadores se dieron cuenta de esto, se introdujo el concepto de diseño modular. Con los principios de diseño modular, los programadores podían escribir sub rutinas, funciones y re utilizar el código. Esto fue una gran idea por un tiempo. Después los desarrolladores se dieron cuenta que estaban copiando y pegando módulos en otras aplicaciones, lo cual genero un gran problema en el área de mantenimiento. Cuando se encontraba un error en el módulo, se tenía que rastrear todas las aplicaciones que contenían dicho modulo y corregir el error.

Los investigadores, propusieron clases y software orientado a objetos para resolver este y muchos otros problemas. Pero de nuevo, la complejidad del software se incrementó. Los investigadores ofrecieron una nueva capa de abstracción para manejar esta complejidad en el software. Esta nueva capa se llama Software basado en componentes. El software basado en componentes es una buena solución para la re utilización y el mantenimiento de software, pero no es la solución a todos los problemas que enfrentan los desarrolladores hoy en día. Las complejidades que se enfrentan hoy en día los desarrolladores tienen que ver con software distribuido, integración de la aplicación, plataformas variantes, protocolos variantes, diferentes dispositivos, el internet, etc.

La arquitectura de software es una pieza central del desarrollo de sistema de software moderno. El objetivo de la arquitectura consiste en desarrollar sistemas de software grandes de forma eficiente, estructura y con capacidades de reusó. La arquitectura forma parte del proceso de diseño de software el cual también forma parte del proceso de desarrollo de software que comprende, requerimientos, diseño, implementación, prueba y desarrollo (Tahuiton, 2011).

3.5.1 Arquitectura orientada a servicios

La arquitectura orientada a servicios, es un concepto de arquitectura de software que define la utilización de servicios para dar soporte a los requisitos del negocio.

A continuación se ofrecen una serie de definiciones de arquitectura SOA:

- La World Wide Web Consortium (W3C) define a SOA como; “Un conjunto de componentes que puede ser invocado y cuyas descripciones de interfaz puede ser publicado y descubierto”
- Según Everware – CBDI: “Estilo resultante de políticas, prácticas y frameworks que permiten que la funcionalidad de una aplicación se pueda proveer y consumir como conjuntos de servicios, con una granularidad relevante para el consumidor...” (John, 2006).
- OASIS definió SOA como: “SOA es un paradigma para organizar y utilizar capacidades distribuidas que pueden estar bajo el control de diferentes dominios. Provee una manera uniforme de ofrecer, descubrir, interactuar con ellos y sus capacidades de uso para producir el efecto deseado consistente en precondiciones y expectativas medibles” (OASIS, 2006).
- IBM define SOA como: “SOA representa una forma de construir sistemas distribuidos que permite ofrecer las funcionalidades de una aplicación como servicios tanto para aplicaciones de usuario final como para otros servicios”. IBM se refiere a SOA como “la plataforma que alinea el Negocio con Tecnología” (Cuomo, 2005).

Una arquitectura SOA es compuesta principalmente por elementos funcionales y elementos relacionados a la calidad del servicio como se puede ver en la siguiente figura.

A continuación se definen cada uno de los elementos de una arquitectura orientada a servicios

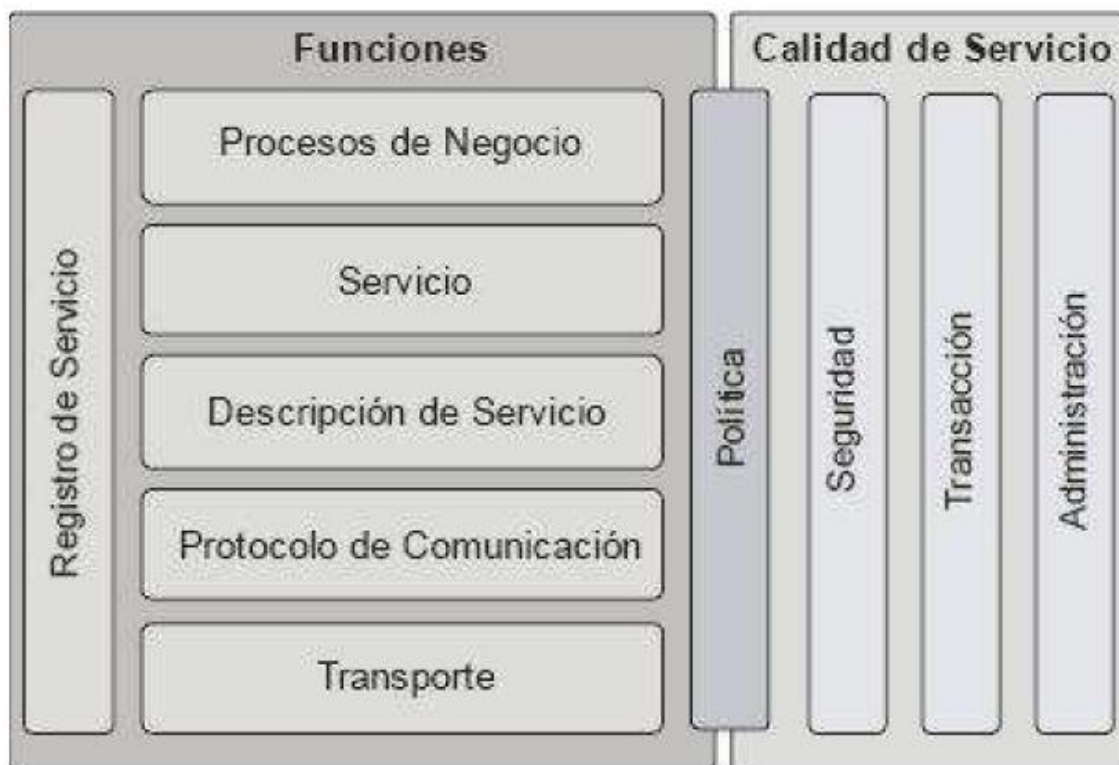


Figura 6. Estructura de elementos básicos de una arquitectura SOA (W3C, 2004)

Funciones:

- **Transporte:** es el mecanismo utilizado para llevar las demandas de servicio desde un consumidor de servicio hacia un proveedor de servicio, y las respuestas desde el proveedor hacia el consumidor.
- **Protocolo de comunicación de servicios:** es un mecanismo acordado a través del cual un proveedor de servicios y un consumidor de servicios comunican qué está siendo solicitado y qué está siendo respondido.
- **Descripción de servicio:** es un esquema acordado para describir qué es el servicio, cómo debe invocarse, y qué datos requiere el servicio para invocarse con éxito.
- **Servicio:** describe un servicio actual que está disponible para utilizar.
- **Procesos de Negocio:** es una colección de servicios, invocados en una secuencia particular con un conjunto específico de reglas, para satisfacer un requisito de negocio.
- **Registro de Servicios:** es un repositorio de descripciones de servicios y datos que pueden utilizar los proveedores de servicios para publicar sus servicios, así como los consumidores de servicios para descubrir o hallar servicios disponibles.

Calidad de Servicio:

- **Política:** es un conjunto de condiciones o reglas bajo las cuales un proveedor de servicio hace el servicio disponible para consumidores.
- **Seguridad:** es un conjunto de reglas que pueden aplicarse para la identificación, autorización y control de acceso a consumidores de servicios.
- **Transacciones:** es el conjunto de atributos que podrían aplicarse a un grupo de servicios para entregar un resultado consistente.
- **Administración:** es el conjunto de atributos que podrían aplicarse para manejar los servicios proporcionados o consumidos.

Colaboraciones en SOA

Una arquitectura SOA está compuesta por un servicio, un proveedor de servicios y un directorio de servicios (Mabrouk, 2008). Se puede observar la estructura de estos elementos en la siguiente figura



Figura 7. Colaboraciones de una arquitectura SOA (W3C, 2004)

A continuación se da una definición de los elementos mencionados:

- **Proveedor de servicios:** Es una entidad accesible a través de la red que acepta y ejecuta consultas de consumidores, y publica sus servicios y su contrato de interfaces en el registro de servicios para que el consumidor de servicios pueda descubrir y acceder al servicio.
- **Consumidor de servicios:** es una aplicación, un módulo de software u otro servicio que demanda la funcionalidad proporcionada por un servicio y la ejecutara de acuerdo a un contrato de interfaz.
- **Registro de servicio:** Es el encargado de hacer posible el descubrimiento de servicios, conteniendo un repositorio de servicios disponibles y permitiendo visualizar las interfaces de los proveedores de servicios a los consumidores interesados.

También es necesario definir las operaciones que ocurren:

- **Acceso (Access):** Después de recoger la descripción del servicio, el consumidor del servicio puede invocar el servicio de acuerdo con la información de la propia descripción del servicio.
- **Publicación (Publish):** Para que los servicios puedan ser accesibles, es necesario ofrecer una descripción del servicio para que esta pueda ser invocada por un consumidor de servicios.
- **Localización (Locate):** Un consumidor de servicios puede localizar un servicio realizando una búsqueda sobre el registro de servicios que cumpla algún criterio.

Y por último, los artefactos los artefactos en una arquitectura orientada a servicios son:

- **Servicio:** Un servicio que está disponible para el uso a través de una interfaz publicada y que permite ser invocado por un consumidor de servicios.
- **Descripción del servicio:** Una descripción del servicio específica la forma en que un consumidor de servicios interactuara con el proveedor de servicios. Especificando el formato de consultas y respuestas desde el servicio.

Requisitos de una arquitectura orientada a servicios:

No existe una definición estándar de cuáles son los Principios de la Orientación a Servicios, por lo tanto, lo único que se puede proporcionar es un conjunto de Principios que estén muy asociados con la Orientación a Servicios (OPOSICIONES, 2012): Estos principios según Thomas Erl son:

- **Los Servicios deben ser reutilizables:** Todo servicio debe ser diseñado y construido pensando en su reutilización dentro de la misma aplicación, dentro del dominio de aplicaciones de la empresa o incluso dentro del dominio público para su uso masivo.
- **Los Servicios deben proporcionar un contrato formal:** Todo servicio desarrollado, debe proporcionar un contrato en el cual figuren: el nombre del servicio, su forma de acceso, las funcionales que ofrece, los datos de entrada de cada una de las funcionalidades y los datos de salida. De esta manera, todo consumidor del servicio, accederá a este mediante el contrato, logrando así la independencia entre el consumidor y la implementación del propio servicio. En el caso de los Servicios Web, se logrará mediante la definición de interfaces con WSDL.
- **Los Servicios deben tener bajo acoplamiento:** Los servicios tienen que ser independientes los unos de los otros. Para lograr ese bajo acoplamiento, lo que se hará es que cada vez que se vaya a ejecutar un servicio, se accederá a él a través del contrato, logrando así la independencia entre el servicio que se va a ejecutar y el que lo llama. Si se consigue este bajo acoplamiento, entonces los servicios podrán ser totalmente reutilizables.
- **Los Servicios deben permitir la composición:** Todo servicio debe ser construido de tal manera que pueda ser utilizado para construir servicios genéricos de alto nivel a partir de servicios de bajo nivel. En el caso de los Servicios Web, esto se logrará mediante el uso de los protocolos para orquestación (WS-BPEL) y coreografía (WS-CDL).
- **Los Servicios deben ser autónomos:** Todo servicio debe tener su propio entorno de ejecución. De esta manera el servicio es totalmente independiente y nos podemos asegurar que así podrá ser reutilizable desde el punto de vista de la plataforma de ejecución.
- **Los Servicios no deben tener estado:** Un servicio no debe guardar ningún tipo de información. Esto es así porque una aplicación está formada por un conjunto de servicios, lo que implica que si un servicio almacena algún tipo de información, se pueden producir problemas de inconsistencia de datos. La solución, es que un

servicio sólo contenga lógica, y que toda información esté almacenada en algún sistema de información sea del tipo que sea.

- **Los Servicios deben poder ser descubiertos:** Todo servicio debe poder ser descubierto de alguna forma para que pueda ser utilizado, consiguiendo así evitar la creación accidental de servicios que proporcionen las mismas funcionalidades. En el caso de los Servicios Web, el descubrimiento se logrará publicando sus interfaces en registros UDDI.

El estilo de arquitectura a implementar para el desarrollo del repositorio de recursos educativos es la arquitectura orientada a servicios (SOA por sus siglas en inglés). Un estilo de arquitectura SOA se basa en interacciones de componentes de software de acoplamiento débiles que proveen un servicio. Un servicio es una pieza de funcionalidad creada para su disponibilidad por un proveedor de servicios para entregar resultados a un consumidor de servicios. Un consumidor de servicios envía la solicitud del servicio al proveedor de servicios. El proveedor de servicios regresa la respuesta al consumidor de servicios conteniendo los resultados esperados (Hashimi, 2003).

Un servicio SOA es una pieza de funcionalidad expuesta con tres propiedades:

1. La interfaz del servicio es independiente de la plataforma
2. El servicio puede ser dinámicamente localizado e invocado
3. El servicio es auto contenido. Esto es, el servicio mantiene su propio estado

3.5.2 Arquitectura en capas

Los servicios son puestos en la red y operan de manera cooperativa para dar soporte a uno o más procesos de negocios. En este modelo, una aplicación se convierte en un conjunto de servicios de usuario, negocios y datos que satisface las necesidades de los procesos de negocios o procesa su soporte. Como los servicios están diseñados para el uso general y siguen lineamientos de interfaz publicados, pueden ser reutilizados y compartidos entre múltiples aplicaciones.

Es necesario puntualizar las siguientes características que traen consigo esta forma de arquitectura: Utilización de esquemas más complejos.

- Los datos y los servicios web aparecen separados.
- Facilidad para separar datos de la “lógica de negocio”.
- Mayor seguridad en los “datos corporativos”.
- El cliente recibe los datos y la información de forma indirecta través servidor

Funcionalidad de las capas

Capa de Presentación o Interfaz de Usuario: Esta capa, está formada por los formularios y los controles que se encuentran en los formularios. Es la capa con la que interactúa el usuario.

Capa de Negocio: Está formada por las entidades, que representan objetos que van a ser manejados o utilizados por toda la aplicación. En este caso, están representados por clases y “DataTables” que se crean.

Capa de Acceso a Datos: Contiene clases que interactúan con la base de datos, éstas clases altamente especializadas se encuentran en la arquitectura del sistema y permiten, utilizando los procedimientos almacenados generados, realizar todas las operaciones con la base de datos de forma transparente para la capa de negocio.

Arquitectura en cuatro capas

El uso de cuatro capas permite una mayor extensibilidad en caso de que existan también clientes no web en el sistema, que trabajarían directamente contra el servidor del modelo. Los desarrollos más recientes empiezan a experimentar con una capa adicional

Funcionalidad de las capas

Capa de Presentación: Esta capa no se hace cálculos, consultas o actualizaciones sobre el dominio, de hecho ni siquiera se tiene visibilidad sobre la capa del dominio. La idea básica es separar todo lo que es programación Web de la aplicación.

Capa de Aplicación: Es la encargada de acceder la capa al dominio, simplificar la información del dominio convirtiéndolo a los tipos de datos que entiende la interfaz como son: enteros, reales, cadenas de caracteres, fecha y clases contenedoras (container, collection). Una forma de organizar esta nueva capa de la aplicación es considerarla una fachada al dominio.

Capa de Dominio de la Aplicación: Muestra el diseño de la arquitectura que facilita a que las aplicaciones se desarrollen.

Capa de Repositorio: En esta capa se tiene en cuenta la infraestructura como el hardware o las redes físicas.

Esta subdivisión en cuatro capas facilita determinar el papel que desempeña la arquitectura dentro de una organización. Cada capa actúa como cliente de la capa inferior a ella y como servidor de la capa superior. Los arquitectos no deben de malgastar su tiempo en temas relacionados con la infraestructura, tales como el sistema operativo. La mejor forma de separar la arquitectura de la infraestructura es tener en mente el esquema de cuatro capas antes mencionado: la infraestructura debe de dar soporte a la arquitectura. Mezclar erróneamente conceptos de una y otra capa es un error muy común en muchas organizaciones.

Arquitectura en siete capas

Partiendo de que un servicio Web o Web Service es una colección de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Los estándares abiertos y el foco en la comunicación y colaboración entre las personas y aplicaciones han creado un entorno donde los servicios Web se están convirtiendo en la plataforma para la integración de aplicaciones. Estas aplicaciones de software pueden ser desarrolladas en lenguajes de programación diferente y ejecutada sobre cualquier plataforma, y mediante los servicios poder intercambiar datos entre ellas a través de la Internet.

Funcionalidad de las capas

Capa de Presentación: Consiste en las interfaces o pantallas mostradas en los navegadores o browsers de los usuarios finales de la aplicación. Esta capa de presentación hace referencia a la interfaz del servicio Web y envía tipos de datos encapsulados en mensajes que son de conocimiento único de la interfaz. Esta capa debe conocer los tipos de datos y tipos de mensajes que puede recibir la capa de interfaz de servicio.

Capa de Interfaz de Servicio: Es la que se expone fuera de la aplicación. Contiene los tipos de mensajes utilizados por los métodos del servicio Web. Permite la comunicación de la aplicación Web y la aplicación móvil con la lógica de negocios.

Capa de Implementación del Servicio: En esta capa se tienen la clase adapter que es la encargada de intermediar la comunicación entre la capa de presentación y la de lógica de negocio. Como la lógica de negocio solo maneja entidades de negocio y la capa de presentación solo maneja tipos de datos, existe una clase traductor que se encarga de traducir las entidades en tipos de datos y viceversa.

Capa de Tipos de Datos: Contiene toda la colección de tipos de datos manejados por la capa de presentación y que se transmiten hasta la implementación de servicio donde tendrán que convertirse en entidades del negocio.

Capa Lógica del Negocio: Contiene el corazón del sistema. Es un conjunto de subrutinas que regulan la acción del usuario enviándolas a la capa de repositorio de datos. Esta información es enviada y encapsulada en entidades de negocio.

Capa Entidades del Negocio: Es la representación de los objetos manejados en el sistema y también de las tablas de la base de datos. Permiten el transporte de los datos desde fuera hacia la base de datos y viceversa. Maneja el principio de programación con objetos los cuales contienen atributos que representaran datos físicos. Sirve de intermediario entre la capa de implementación y el repositorio de datos.

Capa Repositorio de Datos: Es la capa encargada de la inserción y extracción de datos en la base de datos. El repositorio se comunica únicamente con la base de datos y con la lógica de negocio.

Únicamente se comunicará una capa con su capa superior inmediata y su capa inferior inmediata, de lo contrario se rompería el principio de comunicación entre capas.

3.6 Visualización en línea

Los avances de la tecnología hoy en día habilitan diversas capacidades para los usuarios del internet. Hoy en día es posible ver videos, ejecutar aplicaciones, leer documentos, escuchar audios y visualizar imágenes en línea. Lo mencionado anteriormente se concibe como recursos y aún más pueden ser considerados como recursos educativos (Hernández, 2001, Marcano, 2008).

Los repositorios de recursos de aprendizaje tienen por objetivo almacenar/recuperar/distribuir los recursos educativos como parte de sus funciones básicas (ADL, 2002 Rosanigo Et al, 2012). Los repositorios de recursos educativos hoy en día tienden a solo dar soporte a un tipo de archivo, siendo este los documentos de texto (PDF, TXT, DOCS). Por lo que ofrecer soporte a múltiples de tipos de archivos en los repositorios se convierte en una necesidad (Alvarez Et al, 2011).

Dados los diferentes tipos de recursos educativos se considera como una capacidad extra en los repositorios que los usuarios puedan visualizar dichos recursos en línea. De acuerdo a diversos diccionarios por visualizar se entiende:

- Representar por medio de imágenes
- Representar visualmente algo abstracto
- Ver la información de una pantalla

Esto quiere decir que la capacidad que se habilitar en los repositorios es que los usuarios pueden visualizar los diferentes tipos de recursos en línea.

3.6.1 Visualización de diferentes tipos de recursos educativos

La capacidad que se pretende habilitar en los repositorios, tiene que ver con la visualización de recursos en línea. Para lograr esto como se mencionó anteriormente se piensa trabajar con diferentes tipos de archivos y estos son archivos de imágenes, audios, videos, aplicaciones y documentos de texto.

Para lograr esto, la idea de visualizar los recursos educativos debe ser aclarada dado que el concepto de visualizar no se puede aplicar a los 5 tipos de archivos mencionados anteriormente. Para esto en la siguiente lista se explica a que refiere con visualizar los recursos en línea:

- Archivos de imágenes: Las imágenes como recursos educativos pueden contener información de aprendizaje por lo que este tipo de recurso se pretende almacenar. En cuanto a la visualización, se refiere a que el usuario encuentre el recurso de este tipo y cuando acceda a él la imagen se muestre para él pueda verla sin la necesidad de descargarla.
- Archivos de audio: Los archivos de audio pueden ser utilizados como recursos educativos, ya sea de pláticas sobre algún tema o de conferencias o simplemente algún audio que contenga alguna información sobre algún tema en particular que le interese al usuario. En cuanto a la visualización, se refiere a que el usuario pueda reproducir el archivo de audio desde el mismo repositorio sin la necesidad de descargar el archivo para poder reproducirlo.
- Archivos de video: Los archivos de video pueden ser utilizadas como recursos educativos. Estos archivos pueden contener información sobre algún tema en particular que al usuario le interese. En cuanto a la visualización, se refiere a que el usuario pueda reproducir el archivo de video desde el repositorio para que este no tenga la necesidad de descargarlo.
- Archivos de documentos de texto: Los archivos de texto como recurso educativos son los más comunes en la actualidad. Los repositorios de recursos educativos que se encuentran en línea solo contienen este tipo de archivo. En cuanto a la visualización, se refiere a que el usuario pueda leer este documento desde el

repositorio sin la necesidad de descargarlo. Cabe mencionar que esta capacidad ya se encuentra habilitada en varios repositorios de recursos educativos.

- Archivos de aplicaciones: Los archivos de aplicaciones también pueden ser utilizadas como recursos educativos, es importante mencionar que las aplicaciones pueden ser para computadoras de escritorio o para aplicaciones de aprendizaje. Debido a esto la visualización de estas aplicaciones se complica para poder ser ejecutadas desde un repositorio de recursos educativos. Por lo que en cuanto a visualización en este caso particular de archivos no aplica.

Lo explicado anteriormente es la manera que se define la visualización de los recursos educativos. Como se mencionó la visualización de los recursos educativos pretende ofrecer mayor interactividad con los usuarios de este repositorio.

Cabe mencionar que la manera en que se pretende manipular la visualización con cada tipo de archivo requiere el diseño de ciertas tecnologías o la implementación de algunas como se explica en la siguiente lista.

- Archivos de imágenes: Para la visualización de imágenes se propone utilizar una interfaz cuando se acceda al recurso del tipo imagen del repositorio que permita visualizar claramente la imagen.
- Archivos de audio: Para la visualización de archivos de audio se propone utilizar una interfaz de reproducción de archivos de audio para que el usuario seleccione desde que punto quiere escuchar el audio.
- Archivos de video: Para la visualización de archivos de video se propone utilizar una interfaz de reproducción de archivos de video para que el usuario seleccione desde que punto quiere ver el archivo de video.
- Archivos de documento de texto: Para la visualización de archivos de texto se propone utilizar una interfaz que permita la lectura de dichos archivos. Cabe mencionar que actualmente todos los navegadores cuentan con un lector de archivos del tipo PDF.
- Archivos de aplicaciones: Como se mencionó este tipo de archivo no es posible ejecutar la aplicación desde el repositorio dado que se pretende almacenar aplicaciones de dispositivos móviles y aplicaciones de computadora por lo que poder ejecutarlas desde el repositorio no será posible.

Conclusión

Con el concepto de visualización definido para cada uno de los tipos de recursos educativos que se pretenden almacenar se puede comenzar a trabajar en las tecnologías que permitan dicha visualización de los recursos. Es necesario mencionar que esta característica no ha sido implementada en ningún otro repositorio. Por lo que se le añade al repositorio de recursos educativos un mayor nivel de interactividad con el usuario.



4 Desarrollo del repositorio de recursos educativos

En este capítulo se mencionan los objetivos de una arquitectura de software, los requerimientos que necesita cumplir la arquitectura de software que se pretende desarrollar, así como también mostrar algunos elementos referentes al diseño de la arquitectura de software como los actores que envuelven la arquitectura y los casos de uso. Además de esto veremos una comparación que se realizó de arquitecturas que resolvían parcialmente la problemática, en base a la cual surge la arquitectura planteada en el documento y por último se habla de la justificación de la arquitectura desarrollada.

4.1 Diseño de arquitectura para su implementación

Para el desarrollo del repositorio de recursos educativos en este documento, se utilizan ciertos lenguajes de programación que le ofrecen al repositorio cierta flexibilidad y facilidad de manteniendo al repositorio. Además de integrar la capacidad de interoperabilidad, ya que para la implementación del repositorio se plantea utilizar tecnología web. Con la tecnología web, se facilita que dicho repositorio pueda ser accedido desde diferentes dispositivos tales como: Dispositivos móviles, laptops, computadores de escritorio, etc. Sin comprometer la integridad en cuanto a la parte de visual de la aplicación web (debido al tamaño de pantalla).

4.1.1 Objetivos de una arquitectura

Según Joe Batman, del Software Engineering Institute [Batman, 1999], los objetivos que debe cumplir una arquitectura de software efectiva son los siguientes:

- La arquitectura se debe usar como un plan descriptivo de la construcción del sistema, estructurando su resultado

- Mediante la arquitectura debe ser fácil identificar los modelos estructurales del dominio de aplicación, intentando proporcionar la solución más simple.
- La arquitectura debe identificar las partes del sistema, ordenándolo mediante su descomposición.
- La arquitectura debe aislar los componentes del sistema, presentando las instrucciones y líneas de acción necesarias para llegar al detalle de la descomposición.
- La arquitectura debe identificar las relaciones con sistemas externos.
- El entorno de desarrollo del sistema debe ser soportado por la arquitectura y esta debe facilitar su conocimiento en cuanto a sus servicios y utilidades.
- La arquitectura debe soportar los incrementos adicionales de capacidad del sistema.
- La arquitectura debe utilizar estándares para uniformizar problemas y soluciones en cuanto a nomenclatura, estilo, planificación, procedimientos, etc.
- La arquitectura debe proporcionar la documentación adecuada.
- Para el uso efectivo de la arquitectura debe realizarse entrenamientos y dar soporte a los cambios culturales que conlleva.
- El control de calidad debe ser integrado en la arquitectura.
- La arquitectura debe proporcionar propiedades estructurales y de calidad que permitan un proceso de prueba eficiente.

Cuando se realiza un estudio de una arquitectura, se debe verificar que un arquitecto, para cumplir con su cometido debe, en primera fase, basándose en los requisitos del sistema diseñar un esbozo de la arquitectura y establecer los objetivos que debe cumplir para satisfacer estos requerimientos, a continuación, establecer los principios que debe seguir para cumplir los objetivos definidos e integrar esos principios en el diseño de la arquitectura y, y por último, evaluar y redefinir de acuerdo a los resultados de la arquitectura para todos los casos de uso, midiendo parámetros de facilidad de adaptación al tamaño de los sistemas, presentaciones, integración etc. [Allamaraju, 1998]

4.1.2 Requerimientos de una arquitectura

Como se mencionó en el apartado anterior, una de las tareas que primero se deben realizar a la hora de definir la arquitectura de un sistema que incluya arquitectura SOA, es establecer la especificación de sus requisitos. Mediante estos requerimientos podremos conocer los

procesos que se llevan a cabo en el dominio del problema y establecer los objetivos que debe cumplir la arquitectura.

A continuación se muestra una tabla con el resumen de requisitos funcionales generales, que debe cumplir la arquitectura:

Requisitos	Descripción
R1	Servicio de identificador de plataforma de acceso del cliente
R2	Navegador web/Navegador dispositivo móvil
R3	Administración de sesión de usuario
R6	Visualización de recursos en línea
R7	Visualización de recursos almacenados
R8	Descarga de recursos educativos
R9	Servicio de búsqueda de recursos educativos
R10	Servicio de registro de recursos educativos en el repositorio
R11	Servicio catalogador de archivos registrados en el repositorio
R12	Servicio web para interoperabilidad del repositorio
R13	Seguridad interna del repositorio
R14	Almacenaje de archivo de aplicaciones
R15	Almacenaje de archivo de audio
R16	Almacenaje de archivo de documentos
R17	Almacenaje de archivo de imágenes
R18	Almacenaje de archivo de metadatos
R19	Almacenaje de archivo de video

Tabla 1. Listado de para el desarrollo del repositorio de recursos de aprendizaje.

4.1.3 Diagrama casos de uso y Actores

Antes de escribir cada uno de los requisitos en forma de casos de uso, es necesario detallar el diagrama de casos de uso (Figura 8) y cada uno de los actores que intervienen en los casos de uso y que utilizaran el sistema. Para ello se definirán los roles los siguientes roles:

- **Usuario:** Este rol es del usuario que tienen la capacidad de buscar, registrar (aplicaciones, audio, imágenes, documento y videos), descargar y visualizar recursos. Además de esto, tiene la capacidad de almacenar los metadatos de dichos recursos.

- **Sistema:** Este rol está representado en el repositorio, ya que tiene la responsabilidad de gestionar la identificación de la plataforma así como encargarse de la seguridad de inyección SQL.

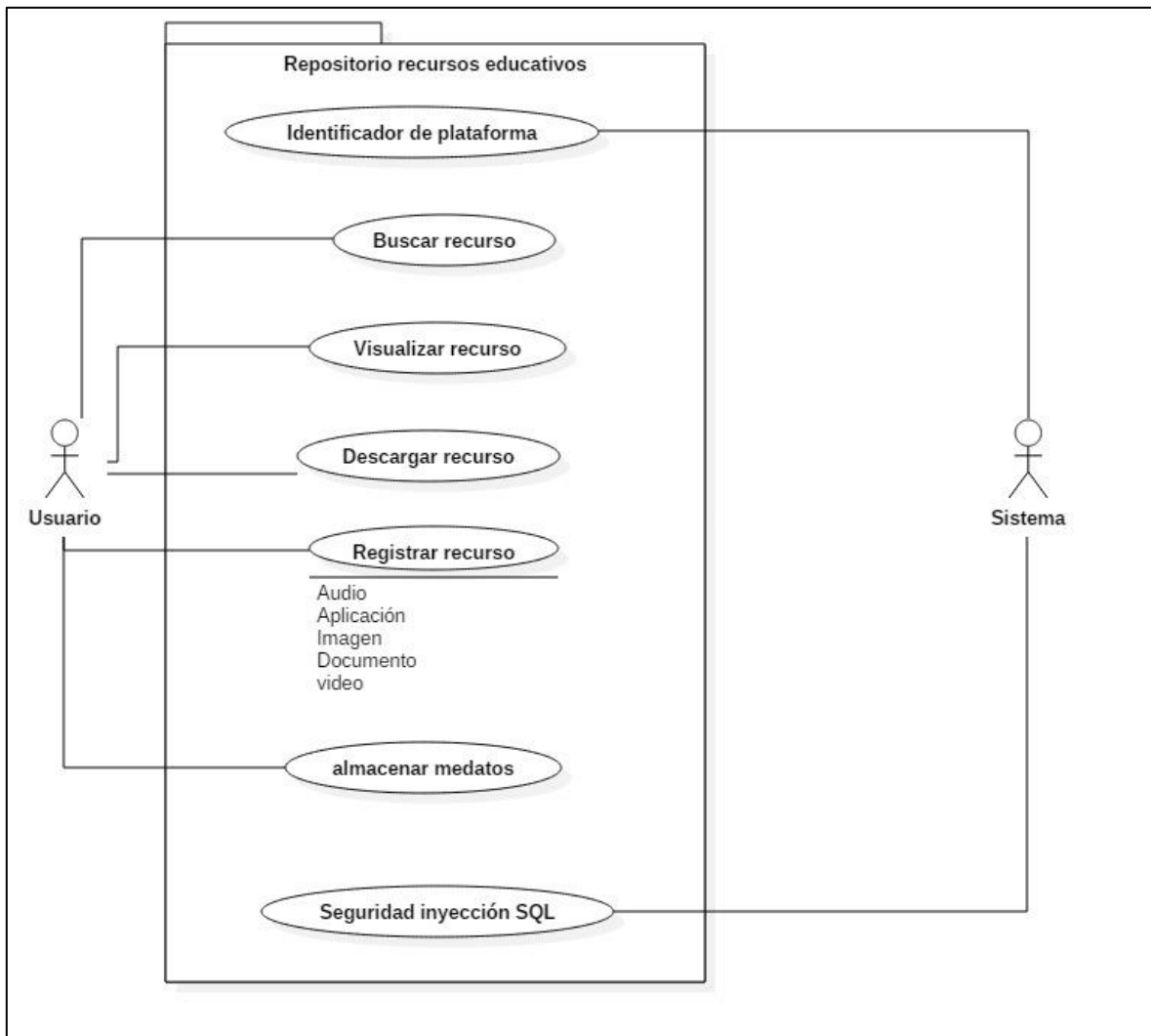


Figura 8. Diagrama de casos de uso

4.1.4 Casos de uso

Para describir cada uno de los casos de uso correspondientes a los requisitos funcionales presentados anteriormente, usaremos la siguiente plantilla para la definición de cada uno de los casos de uso:

Funcionalidad	Nombre del caso de uso
Objetivo	Descripción informal de los objetivos
Actores	Actores que intervienen: principales y secundarios
Precondiciones	Condiciones que deben cumplirse para que pueda llevarse a cabo
Pasos (Flujo básico)	Secuencia de pasos necesarios para que se desarrolle con éxito
Extensiones (Flujos alternativo)	Si se produce un error interno durante el proceso, indicar el proceso alternativo a seguir.

Usando la plantilla anterior, se procede a documentar cada uno de los casos de uso del sistema, que pertenecen a las funcionalidades descritas en los requerimientos:

Funcionalidad	Identificador de plataforma
Objetivo	Indicar de que plataforma viene el usuario para mostrar una interfaz correcta y adecuada
Actores	
Precondiciones	Iniciar la aplicación web del repositorio de recursos educativos
Pasos (Flujo básico)	1. Iniciar la aplicación web del repositorio de recursos educativos
Extensiones (Flujos alternativo)	Si se produce un error de identificación de plataforma del usuario, es decir no se logra definir qué plataforma está usando el usuario, se muestra una interfaz genérica

Funcionalidad	Visualización de recursos en línea
Objetivo	El poder visualizar un recurso en línea, es una capacidad que se le da al repositorio y al usuario de por visualizar un recurso educativo en línea (Refiérase al capítulo 3.6)
Actores	
Precondiciones	Iniciar aplicación web del repositorio e iniciar sesión de usuario y realizar una búsqueda de recursos educativos
Pasos (Flujo básico)	<ol style="list-style-type: none"> 1 Entrar a la página web donde se aloja la aplicación web del repositorio 2 Entrar en el área de inicio de sesión de usuario 3 Rellenar los campos con la información correcta 4 Iniciar la sesión 5 Realizar una búsqueda de un recurso educativo en el repositorio

4 DESARROLLO DEL REPOSITORIO DE RECURSOS EDUCATIVOS

	<ol style="list-style-type: none"> 6 Seleccionar un recurso educativo encontrado por el servicio de búsqueda, el cual envía a una página nueva 7 Seleccionar la opción de visualizar el recurso educativo (en caso de estar disponible)
Extensiones (Flujos alternativo)	En caso de error de carga del recurso educativo para su apropiada visualización, se le indica al usuario de dicho problema. Se le indica al usuario que intente más tarde.

Funcionalidad	Visualización de recursos almacenados en el repositorio
Objetivo	Después de utilizar el servicio de búsqueda del repositorio, se le muestra una lista de los recursos que se encontraron en el repositorio,
Actores	
Precondiciones	Iniciar aplicación web, iniciar sesión de usuario, realizar una búsqueda de recursos educativos en el repositorio.
Pasos (Flujo básico)	<ol style="list-style-type: none"> 1 Entrar a la página web donde se aloja la aplicación web del repositorio 2 Entrar en el área de inicio de sesión de usuario 3 Rellenar los campos con la información correcta 4 Realizar una búsqueda de un recurso educativo en el repositorio
Extensiones (Flujos alternativo)	En caso de encontrar ningún recurso educativo que se adecue a las condiciones de búsqueda del usuario, se le indica al usuario de que dicho evento ocurrió.

Funcionalidad	Descarga de recursos educativos
Objetivo	Permitir al usuario descargar un recurso educativo del repositorio
Actores	
Precondiciones	Iniciar aplicación web, iniciar sesión de usuario, realizar una búsqueda de recursos educativos en el repositorio.
Pasos (Flujo básico)	<ol style="list-style-type: none"> 1 Entrar a la página web donde se aloja la aplicación web del repositorio 2 Entrar en el área de inicio de sesión de usuario 3 Rellenar los campos con la información correcta 4 Realizar una búsqueda de un recurso educativo en el repositorio 5 Seleccionar un recurso educativo encontrado por el servicio de búsqueda, el cual envía a una página nueva 6 Seleccionar la opción de descargar el recurso educativo (en caso de estar disponible)
Extensiones (Flujos alternativo)	En caso de generarse un error al seleccionar la opción de descargar recurso educativo, se le indica al usuario de dicho evento.

Funcionalidad	Servicio de búsqueda de recursos educativos
Objetivo	Facilitar la capacidad de búsqueda de recursos en el repositorio a los usuarios del mismo
Actores	
Precondiciones	Iniciar aplicación web, iniciar sesión de usuario
Pasos (Flujo básico)	<ol style="list-style-type: none"> 1 Entrar a la página web donde se aloja la aplicación web del repositorio 2 Entrar en el área de inicio de sesión de usuario 3 Rellenar los campos con la información correcta 4 Seleccionar la opción de realizar búsqueda de recursos educativos
Extensiones (Flujos alternativo)	<p>Si se produce un error en el proceso interno de búsqueda del recurso, se le comunicara al usuario.</p> <p>En caso de encontrar ningún recurso educativo que se adecue a las condiciones de búsqueda del usuario, se le indica al usuario de que dicho evento ocurrió.</p>

Funcionalidad	Servicio de registro de recursos educativos en el repositorio
Objetivo	Facilitar la capacidad de registro de recursos en el repositorio a los usuarios del mismo.
Actores	
Precondiciones	Iniciar aplicación web, iniciar sesión de usuario
Pasos (Flujo básico)	<ol style="list-style-type: none"> 1 Entrar a la página web donde se aloja la aplicación web del repositorio 2 Entrar en el área de inicio de sesión de usuario 3 Rellenar los campos con la información correcta 4 Seleccionar la opción de registrar recursos educativos en el repositorio 5 Rellenar los campos solicitados 6 Presionar botón de registrar recurso
Extensiones (Flujos alternativo)	Si se produce un error en el proceso interno de registro del recurso educativo, se le comunicara al usuario.

Funcionalidad	Seguridad del repositorio
Objetivo	Ofrecer ciertos niveles de seguridad al repositorio para evitar inyección SQL
Actores	
Precondiciones	Iniciar aplicación web, iniciar sesión de usuario
Pasos (Flujo básico)	<ol style="list-style-type: none"> 1 Entrar a la página web donde se aloja la aplicación web del repositorio 2 Entrar en el área de inicio de sesión de usuario 3 Rellenar los campos con la información correcta 4 Seleccionar la opción de registrar recursos educativos en el repositorio 5 Rellenar los campos solicitados 6 Presionar botón de registrar recurso
Extensiones (Flujos alternativo)	No aplica

Funcionalidad	Almacenaje de archivo de aplicaciones
Objetivo	Ofrecer la capacidad del repositorio de almacenar/almacenar archivos de aplicaciones
Actores	
Precondiciones	Iniciar aplicación web, iniciar sesión de usuario
Pasos (Flujo básico)	<p>----- En caso de búsqueda de recursos -----</p> <ol style="list-style-type: none"> 1 Entrar a la página web donde se aloja la aplicación web del repositorio 2 Entrar en el área de inicio de sesión de usuario 3 Rellenar los campos con la información correcta 4 Seleccionar la opción de realizar búsqueda de recursos educativos <p>----- En caso de registro de recursos -----</p> <ol style="list-style-type: none"> 1 Entrar a la página web donde se aloja la aplicación web del repositorio 2 Entrar en el área de inicio de sesión de usuario 3 Rellenar los campos con la información correcta 4 Seleccionar la opción de registrar recursos educativos en el repositorio 5 Rellenar los campos solicitados 6 Presionar botón de registrar recurso
Extensiones (Flujos alternativo)	No aplica

Funcionalidad	Almacenaje de archivo de audio
Objetivo	Ofrecer la capacidad del repositorio de almacenar/almacenar archivos de audio
Actores	
Precondiciones	Iniciar aplicación web, iniciar sesión de usuario
Pasos (Flujo básico)	<p>----- En caso de búsqueda de recursos -----</p> <ol style="list-style-type: none"> 1. Entrar a la página web donde se aloja la aplicación web del repositorio 2. Entrar en el área de inicio de sesión de usuario 3. Rellenar los campos con la información correcta 4. Seleccionar la opción de realizar búsqueda de recursos educativos <p>----- En caso de registro de recursos -----</p> <ol style="list-style-type: none"> 1. Entrar a la página web donde se aloja la aplicación web del repositorio 2. Entrar en el área de inicio de sesión de usuario 3. Rellenar los campos con la información correcta 4. Seleccionar la opción de registrar recursos educativos en el repositorio 5. Rellenar los campos solicitados 6. Presionar botón de registrar recurso
Extensiones (Flujos alternativo)	No aplica

Funcionalidad	Almacenaje de archivo de documentos
Objetivo	Ofrecer la capacidad del repositorio de almacenar/almacenar archivos de aplicaciones
Actores	
Precondiciones	Iniciar aplicación web, iniciar sesión de usuario
Pasos (Flujo básico)	<p>----- En caso de búsqueda de recursos -----</p> <ol style="list-style-type: none"> 1. Entrar a la página web donde se aloja la aplicación web del repositorio 2. Entrar en el área de inicio de sesión de usuario 3. Rellenar los campos con la información correcta 4. Seleccionar la opción de realizar búsqueda de recursos educativos

	<p>----- En caso de registro de recursos -----</p> <ol style="list-style-type: none"> 1. Entrar a la página web donde se aloja la aplicación web del repositorio 2. Entrar en el área de inicio de sesión de usuario 3. Rellenar los campos con la información correcta 4. Seleccionar la opción de registrar recursos educativos en el repositorio 5. Rellenar los campos solicitados 6. Presionar botón de registrar recurso
Extensiones (Flujos alternativo)	No aplica

Funcionalidad	Almacenaje de archivo de imágenes
Objetivo	Ofrecer la capacidad del repositorio de almacenar/almacenar archivos de aplicaciones
Actores	
Precondiciones	Iniciar aplicación web, iniciar sesión de usuario
Pasos (Flujo básico)	<p>----- En caso de búsqueda de recursos -----</p> <ol style="list-style-type: none"> 1. Entrar a la página web donde se aloja la aplicación web del repositorio 2. Entrar en el área de inicio de sesión de usuario 3. Rellenar los campos con la información correcta 4. Seleccionar la opción de realizar búsqueda de recursos educativos <p>----- En caso de registro de recursos -----</p> <ol style="list-style-type: none"> 1. Entrar a la página web donde se aloja la aplicación web del repositorio 2. Entrar en el área de inicio de sesión de usuario 3. Rellenar los campos con la información correcta 4. Seleccionar la opción de registrar recursos educativos en el repositorio 5. Rellenar los campos solicitados 6. Presionar botón de registrar recurso
Extensiones (Flujos alternativo)	No aplica

Funcionalidad	Almacenaje de metadatos
Objetivo	Ofrecer la capacidad del repositorio de almacenar/almacenar de metadatos
Actores	
Precondiciones	Iniciar aplicación web, iniciar sesión de usuario
Pasos (Flujo básico)	<p>----- En caso de búsqueda de recursos -----</p> <ol style="list-style-type: none"> 1. Entrar a la página web donde se aloja la aplicación web del repositorio 2. Entrar en el área de inicio de sesión de usuario 3. Rellenar los campos con la información correcta 4. Seleccionar la opción de realizar búsqueda de recursos educativos <p>----- En caso de registro de recursos -----</p> <ol style="list-style-type: none"> 1. Entrar a la página web donde se aloja la aplicación web del repositorio 2. Entrar en el área de inicio de sesión de usuario 3. Rellenar los campos con la información correcta 4. Seleccionar la opción de registrar recursos educativos en el repositorio 5. Rellenar los campos solicitados 6. Presionar botón de registrar recurso
Extensiones (Flujos alternativo)	No aplica

Funcionalidad	Almacenaje de archivo de video
Objetivo	Ofrecer la capacidad del repositorio de almacenar/almacenar archivos de aplicaciones
Actores	
Precondiciones	Iniciar aplicación web, iniciar sesión de usuario
Pasos (Flujo básico)	<p>----- En caso de búsqueda de recursos -----</p> <ol style="list-style-type: none"> 1. Entrar a la página web donde se aloja la aplicación web del repositorio 2. Entrar en el área de inicio de sesión de usuario 3. Rellenar los campos con la información correcta 4. Seleccionar la opción de realizar búsqueda de recursos educativos <p>----- En caso de registro de recursos -----</p> <ol style="list-style-type: none"> 1. Entrar a la página web donde se aloja la aplicación web del repositorio 2. Entrar en el área de inicio de sesión de usuario 3. Rellenar los campos con la información correcta 4. Seleccionar la opción de registrar recursos educativos en el repositorio 5. Rellenar los campos solicitados 6. Presionar botón de registrar recurso
Extensiones (Flujos alternativo)	No aplica

4.1.5 Comparación de arquitecturas analizadas como soluciones parciales

Para poder proponer una arquitectura que pueda cumplir con la problemática que este documento aborda, es necesario analizar algunas arquitecturas existentes de repositorios los cuales resuelven parcialmente la problemática. En la actualidad se han definido arquitecturas que soportan otro tipo de recursos educativos como muestra la siguiente lista:

- Repositorio de archivos de audio, MTG-DB: A Repository for Music Audio Processing (Cano Et al, 2004)
- Repositorio de archivos de video, An Infrastructure for Open-Architecture Digital Libraries (Lagoze Et al, 1998)
- Repositorio de archivos de imágenes, I2Cnet: content-based similarity search in geographically distributed repositories of medical images (Orphanoudakis Et al, 1996)
- Repositorios de archivos de aplicaciones móviles, Enabling agility through architecture (Brown Et al, 2010)
- Repositorio de archivos de documentos, Sistema INteractivo Distribuido de Repositorios de Objetos de Aprendizaje MatEmáticos (SINDROME) (Lopez Et al, 2007)

A continuación se analizara cada una de las arquitecturas implementadas en los repositorios mencionados.

4.1.5.1 MTG-DB: Un repositorio para el procesamiento del audio de música

Una gran cantidad de investigación de procesamiento de audio de música se dedica actualmente a la recuperación de información musical. Algunos algoritmos extraen descripciones sonológicas y musicales de audio que se puede utilizar para proporcionar nuevas formas de interactuar con grandes colecciones de música.

El MTG-DB proporciona un marco para buscar, navegar y visualizar resultados. En este marco, un usuario se registra una descripción compatible extractor, el extractor se ejecuta en una selección de archivos de audio, los descriptores extraídos se almacenan en el repositorio y, automáticamente, las herramientas están disponibles para:

- Consulta por Descripción: Se aplica cuando los metadatos está en forma textual, que incluye sellos como el estado de ánimo, género, dupla / metro triple o clave.
- Consulta por valores numéricos: Hay una interfaz de control deslizante de descriptores numéricos como latidos por minuto
- Consulta por ejemplo: El usuario sube un archivo de audio y pide audio similar artículos.

MTG-DB es un repositorio a gran escala para el procesamiento basado en el contenido de audio de música que centraliza contenido de audio, metadatos, ontologías y algoritmos. Su objetivo es estimular la investigación de mayor calidad, mediante la evaluación de algoritmos en un corpus grande y diverso.

El repositorio permite la duplicación de experimentos, así como compartir tanto los resultados como los procedimientos experimentales.

La figura 9 muestra la arquitectura implementada en el repositorio de audio.

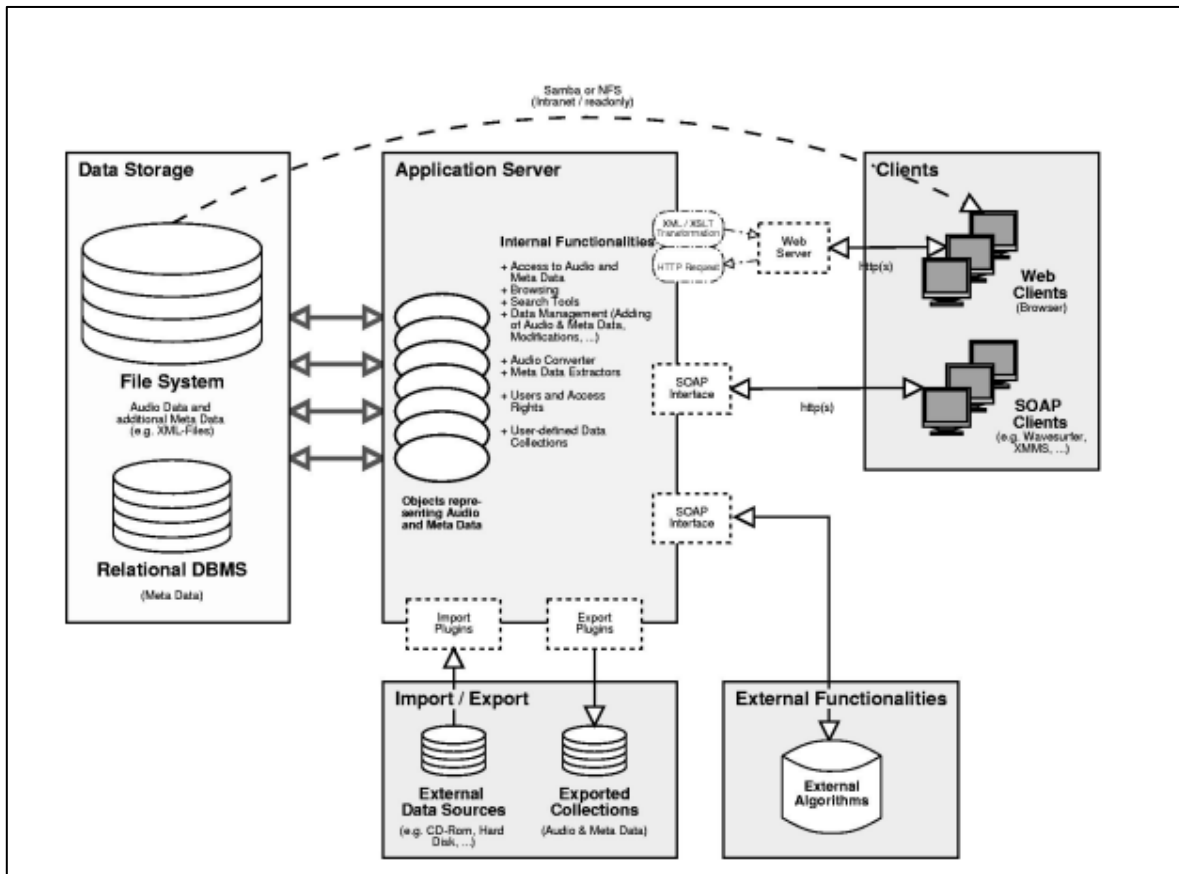


Figura 9. Arquitectura de MTG-DB: A Repository for Music Audio Processing (Cano Et al, 2004).

Conclusión

Esta arquitectura resuelve en términos generales el cómo almacenar archivos de audio en un repositorio y aún más allá como añadir diversas funciones al repositorio. Es esta la razón por la cual se decide añadir como arquitectura de referencia para este documento.

En general la arquitectura cuenta con fortalezas y debilidades. Entre las fortalezas encontramos la posibilidad de almacenaje de archivos de audio, las funcionalidades de los servicios web que incorpora y por último el hecho de que permite agregar funcionalidades externas.

Debido al diseño de esta arquitectura es posible continuar añadiendo módulos en esta arquitectura lo que la convierte en una arquitectura flexible. Cabe decir que existen elementos que pueden ser tomados para la propuesta de la arquitectura que este documento pretende.

4.1.5.2 Una infraestructura para arquitecturas abiertas de librerías digitales

Un requisito fundamental de una arquitectura para las bibliotecas digitales es lograr un medio fiable y seguro para el almacenamiento y el acceso a los contenidos digitales. La arquitectura del repositorio debe ser capaz de:

- Apoyar los tipos de contenido heterogéneos;
- Agregado mixto, posiblemente distribuida, contenido en objetos complejos;
- Proporcionar mecanismos para la gestión del acceso de los contenidos digitales.

La extensibilidad es de suma importancia en la arquitectura de un repositorio. Ya hay un sinnúmero de formas de contenido y nuevos que inevitablemente aparecerán. La arquitectura debe integrar sin problemas los nuevos formas de contenido y los mecanismos de difusión y presentación de los mismos.

Del mismo modo, las instalaciones para la gestión de acceso deben ser inclusivas y no normativo - acomodar la variedad de mecanismos de gestión de los derechos existentes y nuevos en lugar de intentar definir un mecanismo a nivel mundial incluyente. Estas dos dimensiones de extensibilidad sugieren que la arquitectura debe acomodar el "plug in" de nuevos mecanismos de manejo y gestión de los derechos de contenido, en lugar de depender únicamente de las instalaciones que están conectados a la implementación de repositorios específicos.

La arquitectura se construye lógicamente en tres capas. Cada capa se caracteriza por la opacidad decreciente del objeto digital, y su resultante aumentar la funcionalidad desde la perspectiva de los clientes y agentes.

Capa 1 - Gestión de objetos digitales opacos

Esta capa permite el depósito, la gestión y el acceso de los objetos digitales en repositorios. Estos objetos digitales, en esta capa, son totalmente opaco - no hay solicitudes de servicio que permiten "mirar en" la envoltura objeto digital.

Capa 2 - manipulación estructural de objetos digitales

Esta capa proporciona abstracciones y mecanismos para la construcción y la descomposición de los objetos digitales. Considerando que en la capa 1 un objeto digital es sólo una envoltura opaca llamado, esta capa expone las abstracciones que son la estructura (a diferencia de significado) de la envoltura.

Capa 3 - manipulación contenido dependiente de objetos digitales

Esta capa proporciona mecanismos para apoyar la especificidad del contenido en DigitalObjects. Desde un punto de vista operativo (y desde el punto de vista del cliente), la especificidad del contenido está determinado por el conjunto de comportamientos (solicitudes de servicio) asociados con un DigitalObect.

Por último se muestra la arquitectura implementada en el repositorio de archivos de video.

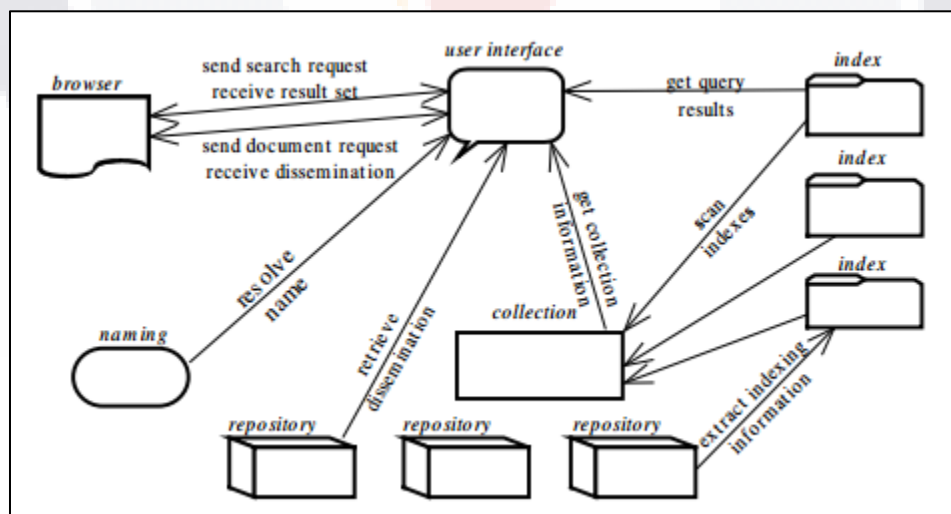


Figura 10. Arquitectura de An Infrastructure for Open-Architecture Digital Libraries (Lagoze Et al, 1998).

Conclusión

Esta arquitectura en términos generales cumple con la función principal, que es almacenar archivos de video. A demás de permitir las funciones básicas de un repositorio (Búsqueda/almacenaje) no ofrece funcionalidades extra como reproducción de video en línea, edición de video, conversión de formato, etc. En sí las capacidades de esta arquitectura están limitadas. Pero bien se puede reutilizar su método para almacenar archivos videos.

4.1.5.3 I2Cnet: Búsquedas basadas en contenido similar en repositorios geográficamente distribuidos de imágenes médicas.

I2C es un sistema de administración de imágenes el cual ha sido desarrollado como parte de un proyecto en marcha para el diseño, implementación, y evaluación de representación de contenido de imágenes y contenido basado en estrategias similares para la recuperación. Diferentes acercamientos son implementados e investigados usando description types. Además, una jerarquía de clases de imágenes definidas por el usuario ayuda al usuario a dirigir sus consultas al compuesto de imágenes apropiado, de esta manera se incrementa la eficiencia de la recuperación a través de la reducción del espacio de búsqueda.

La arquitectura de I2C ha sido desarrollada en base a los principios de la generalidad, modularidad, extensibilidad, arquitectura abierta y conformidad de estándares. La arquitectura abierta permite la incorporación en línea de nuevos algoritmos para la caja de herramientas de I2C.

La figura 10 se presenta la arquitectura implementada en el repositorio de imágenes

Conclusión

La funcionalidad principal de esta arquitectura, en sí no es el almacenaje de archivos del tipo imagen. Si no que ofrece otra serie de funcionalidades distintas a las básicas de un repositorio (Búsqueda/almacenaje) como mejoramiento de imágenes, y descriptores de imágenes. Cabe mencionar que no se aprecia en la arquitectura la existencia de interoperabilidad.

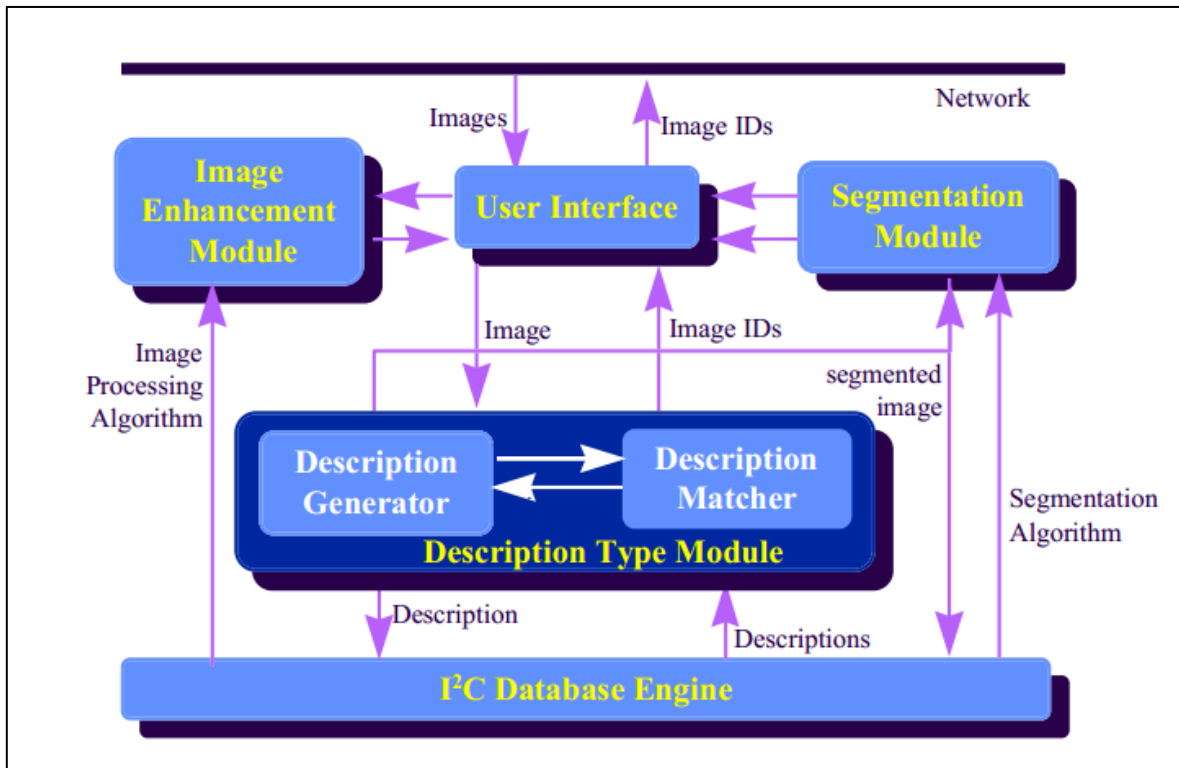


Figura 11. Arquitectura de I2Cnet: content-based similarity search in geographically distributed repositories of medical images (Orphanoudakis Et al, 1996).

Se puede decir que existen partes reutilizables que esta arquitectura ofrece en cuanto al almacenamiento de imágenes que puede ser reutilizado.

4.1.5.4 Habilitando agilidad a través de la arquitectura

Agilidad arquitectural aborda las deficiencias que se producen tanto en la cascada y los modelos de ciclo de vida. La agilidad arquitectural permite el desarrollo arquitectónico de seguir un modelo de "just-in-time".

La entrega de las características del cliente no se retrasa en espera de la finalización de los requisitos exhaustivos y actividades de diseño y revisiones. Al mismo tiempo, la agilidad arquitectural mantiene un enfoque constante y consistente en continua evolución arquitectónica en respaldo de los servicios del cliente emergente.

La agilidad arquitectural evita las trampas de un enfoque miope en historias de usuario, que con el tiempo puede conducir a un aumento de la complejidad y la tortura de la implementación, ya que los desarrolladores buscan incorporar características que no han sido diseñadas para ser soportadas por la arquitectura.

El procedimiento debajo de los paradigmas que llevan a la situación "Todo muy familiar" en la cual las características gradualmente se hacen más y más deficientes, y eventualmente la administración es informada de que el sistema debe ser borrado y re-escrito desde el cero.

La arquitectura conceptual de una tienda de aplicaciones se muestra en la figura 11.

La arquitectura conceptual describe los elementos esenciales de una arquitectura de alto nivel tales como administración de contenido, administración de servicio, acceso a los datos, seguridad y un rango de dispositivos externos destinados que puedan acceder/manipular las aplicaciones.

Usando un acercamiento ágil de comienzo pequeño y crecimiento del sistema, el equipo selecciona las capacidades que soportaran un pequeño número de aplicaciones determinadas en las primeras iteraciones.

Esto requiere identificar cuales elementos arquitecturas dentro de la lógica del negocio, acceso de datos, y los componentes de la administración de servicios que soportan estas capacidades. En posteriores iteraciones, el equipo deberá hacer foco en escalar el sistema en el número de aplicaciones y usuarios, mejorando la seguridad, y permitiendo que los usuarios contribuyan con sus propias aplicaciones.

Los elementos arquitecturales dentro de la seguridad, administraciones de contenido, y publicaciones de componentes debe ser escudriñado para ver cuáles son las necesidades de soportar estas capacidades adicionales.

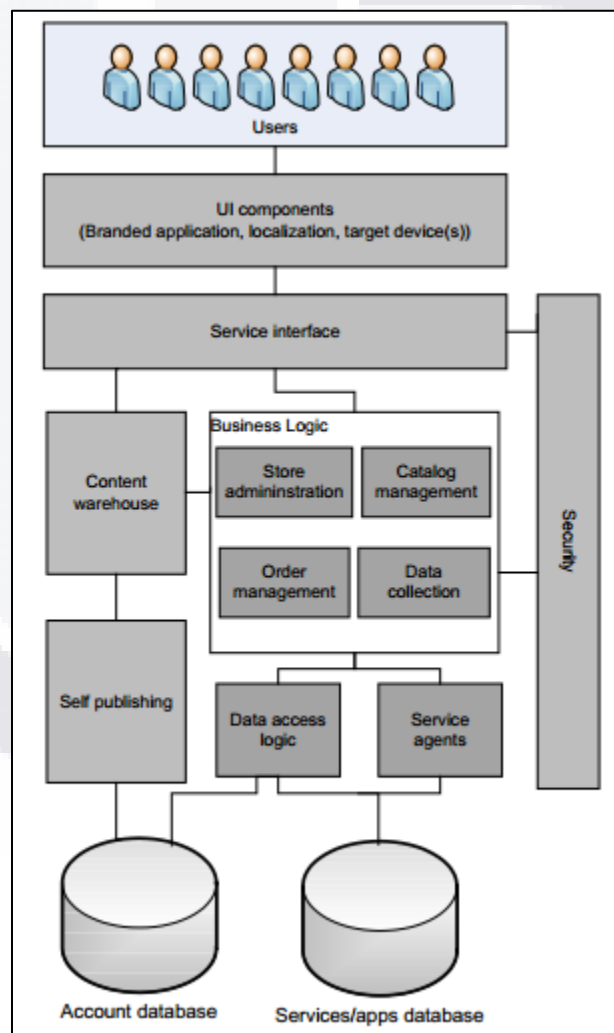


Figura 12. Arquitectura de una tienda de aplicaciones (Brown Et al, 2010).

Conclusión

Como se puede apreciar en la imagen de la arquitectura, se requiere una compleja cantidad de módulos para el almacenaje de aplicación móviles. Esto no quiere decir que no se puedan tomar algunos conceptos que maneja esta arquitectura para implementarlos en el diseño de arquitecturas que pretendan ofrecer el almacenaje de aplicaciones móviles.

Esta arquitectura ofrece elementos clave para el almacenaje de aplicaciones móviles que deben ser tomados en cuenta tales como; administración de catálogo, agentes de servicio, administración de almacén, colección de datos y la lógica de acceso. Estos elementos deben ser considerados en las arquitecturas que pretendan almacenar aplicaciones móviles.

De manera general esta arquitectura cumple con la finalidad de almacenar y exponer las aplicaciones móviles almacenadas a los usuarios. Por lo que los elementos mencionados anteriormente deben ser tomados en cuenta para los propósitos de este documento.

4.1.5.5 Sistema interactivo distribuido de repositorios de objetos de aprendizaje matemáticos

El sistema SINDROME tiene una arquitectura base organizada en capas y basado en componentes, fundamentada en el modelo J2EE aunque incorporando WebServices como el mecanismo de acceso a componentes distribuidos.

Debido a que SINDROME es parte del AIIDM, delega a éste la responsabilidad del manejo del usuario, en lo que se refiere a su registro, administración de sesión y manejo de su espacio de trabajo personalizado. Además, es a través de los servicios del AIIDM, que se incorpora a su funcionalidad a los componentes que tienen que ejecutarse en el contexto del propio sistema, como lo son los mecanismos de control de la búsqueda dinámica y recuperación de objetos de aprendizaje, del modelado del usuario, y de la navegación a través de los resultados.

Estos mecanismos se encuentran distribuidos en los diversos componentes de SINDROME y cuya descripción se presenta a continuación.

Sistema de Almacenamiento y Recuperación de Objetos de Aprendizaje (SAROA).

La búsqueda de los objetos de aprendizaje adecuados al perfil del usuario, se realiza a partir de información específica del usuario obtenida a través de su modelado en el sistema. Dicho modelo incluye entre otras cosas sus preferencias asociadas al estilo de aprendizaje y cualquier elemento del estándar de metadatos de la IEEE LOM v1.0. El proceso incluye una búsqueda heurística de metadatos de objetos de aprendizaje contenidos en la base de datos (repositorio) de éste componente.

Generador de Perfiles (GP).

Este se encarga de crear, actualizar y administrar los perfiles de los usuarios, de tal manera que cada uno de ellos cuente con un perfil individual asignado. Este perfil contiene la referencia al registro de la información personal del usuario administrado por el AIIDM, para poder asociar si es necesario un registro con el otro, así como sus preferencias de estilos de aprendizaje, parametrizadas con los valores asociados al estilo de aprendizaje del usuario, mismos que se obtienen a partir de un cuestionario VARK. Adicionalmente, este componente mantiene el historial de creación y modificación a los registros de los perfiles. Aunque ésta información no se emplea en el presente trabajo, se consideró que podría ser de gran valía en el futuro para analizar el comportamiento evolutivo de los perfiles tanto a nivel grupal como individual.

Interfaz de Presentación y Acceso (IPA).

Este módulo se emplea en dos procesos fundamentales para el objetivo de este trabajo: la búsqueda de objetos de aprendizaje y su recuperación. El sistema presenta los resultados de la búsqueda heurística a través de interfaces de navegación que le permiten al usuario recuperar el objeto de aprendizaje a través del ambiente de aprendizaje (es decir, agregarlo a su espacio de trabajo); o bien visualizar la información detallada de la búsqueda para revisar, por ejemplo, los valores de los metadatos asociados al objeto de aprendizaje.

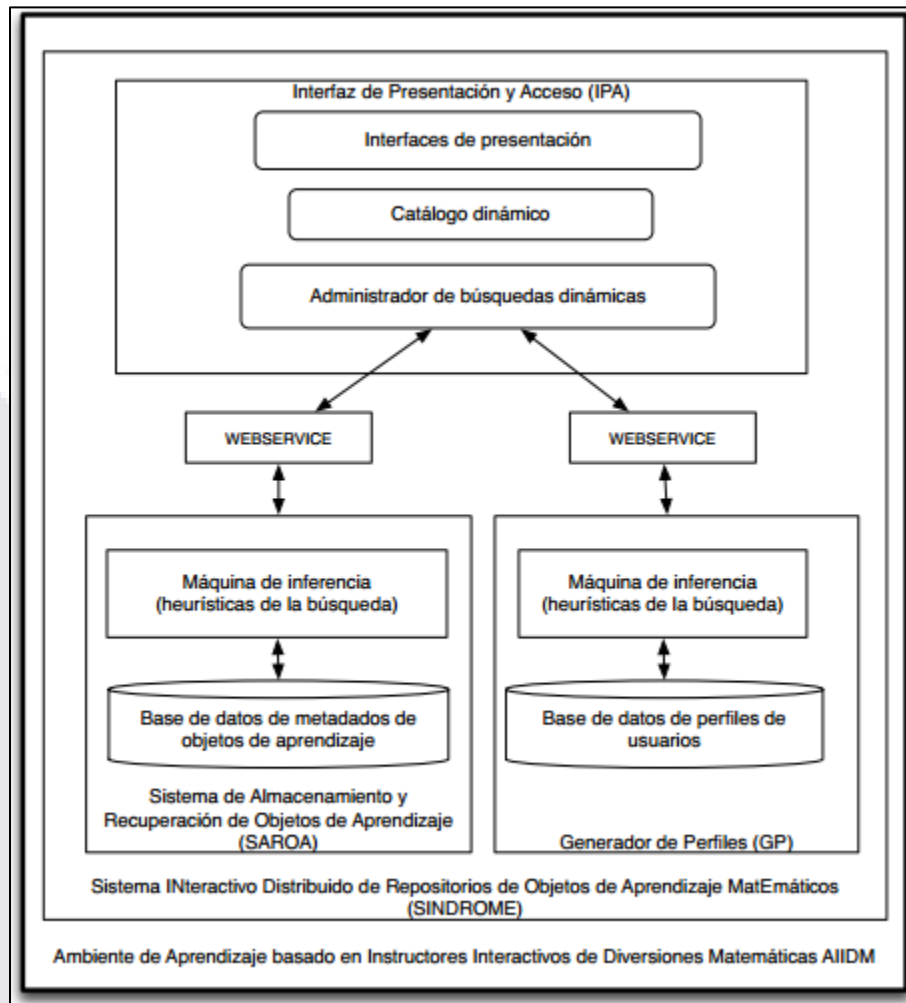


Figura 13. Arquitectura de un Sistema Interactivo Distribuido de Repositorios de Objetos de Aprendizaje MatEmáticos (Lopez Et al, 2007).

Conclusión

Esta arquitectura ofrece un diseño bastante completo para el almacenaje de archivos del tipo documento en este tipo de repositorios. Tiene cuenta con módulos interesantes para los desarrolladores como servicios web e interfaz de usuario. Estos dos elementos ofrecen a los usuarios la capacidad de interoperabilidad.

De manera general esta arquitectura cumple con la función principal que es almacenar objetos de aprendizaje por lo que es importante reutilizar los módulos principales que ofrece.

Cabe notar que es posible expandir esta arquitectura para añadir funcionalidades extra a este repositorio dado que ofrece una arquitectura del tipo basada en capas.

4.1.6 Justificación de la arquitectura

Como se mostró en el capítulo anterior, se realizó un análisis a varias arquitecturas que resuelven parcialmente la problemática planteada en este documento. En análisis de cada uno de las arquitecturas se basa principalmente en las características que esta ofrece como repositorio y los requerimientos necesarios para solventar la problemática.

En la siguiente figura se muestra una tabla comparativa de los requerimientos necesarios para desarrollar el repositorio que resuelva la problemática planteada en conjunto con las características principales de las arquitecturas analizadas.

Con la información anterior podemos validar cuáles son las áreas en las que se necesita enfocarse durante el desarrollo del repositorio de recursos educativos. Y en base a esto trabajar en el desarrollo de la arquitectura.

Requemientos	A Repository for Music Audio Processing	An Infrastructure for Open-Architecture Digital Libraries	I2Cnet	Enabling agility through architecture	SINDROME
Identificador de plataforma de acceso					
Interfaz de usuario	x	x	x	x	
Login de usuario					
Interoperabilidad	x			x	
Servicio de búsqueda	x	x	x	x	x
Servicio de registro de recursos	x	x	x	x	x
Servicios web	x			x	x
Seguridad				x	
Catalogador de archivos		x		x	x
Visualización de recursos en línea	x				
Almacenaje de archivos de audio	x				
Almacenaje de archivos de video		x			
Almacenaje de archivos de imágenes			x		
Almacenaje de archivos de documentos				x	
Almacenaje de archivos de aplicaciones					x
Almacenaje de archivos de metadatos	x	x	x	x	x

Figura 14. Tabla Comparación de requerimientos propuestos sobre las arquitecturas analizadas (Luna et AL, 2015)

4.2 Definición de la arquitectura para su implementación

En este apartado se expondrá la arquitectura propuesta que resolverá los problemas planteados en el capítulo 2. Para esto se toma la idea del apartado anterior, la justificación de la arquitectura para construir definirla la misma usando dos estilos arquitectónicos diferentes. Entre estos estilos arquitectónicos encontramos la arquitectura SOA para poder implementar la característica de interoperabilidad, la cual nos facilitara tener el repositorio de recursos educativos como aplicación web para que esta pueda ser accedida desde cual plataforma.

Como se ha indicado anteriormente, un repositorio de recursos educativos es una colección de recursos que son accesibles a través de una red de comunicaciones. No es necesario tener un conocimiento previo de la estructura de la colección, ya que el repositorio puede almacenar los metadatos y una referencia de donde se encuentra el recurso o almacenar los metadatos y los recursos de aprendizaje en el mismo. Como se mencionó anteriormente la

finalidad de un repositorio de recursos educativos es facilitar el acceso a recursos educativos para su reutilización.

Antes de pasar a presentar la arquitectura primero se presenta en la figura 17 el diagrama de objetos, esto con la finalidad de mostrar una vista parcial o completa de los objetos del sistema del repositorio de recursos educativos.

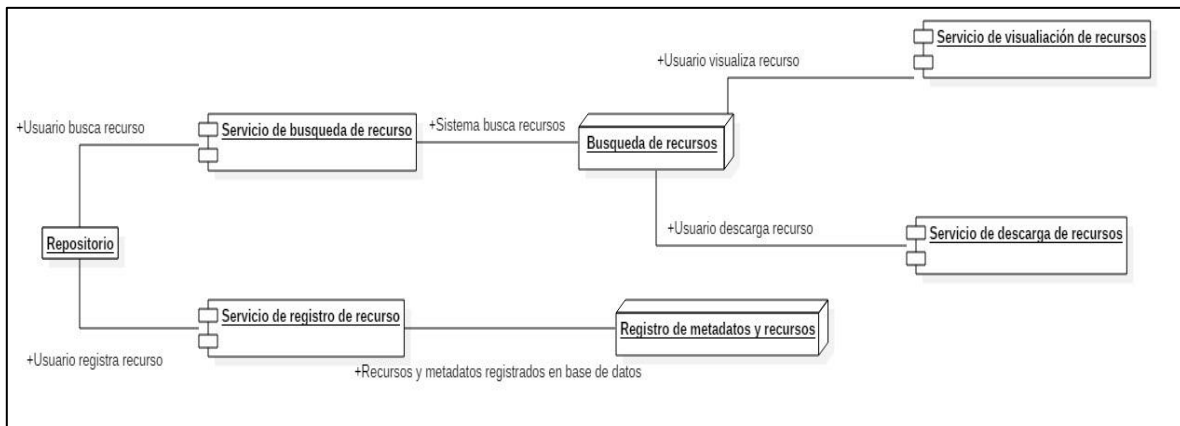


Figura 15. Diagrama de objetos que representa de manera general el repositorio.

Así como también en la figura 18 se presenta el diagrama del sistema dividido en componentes y muestra la dependencia entre estos componentes.

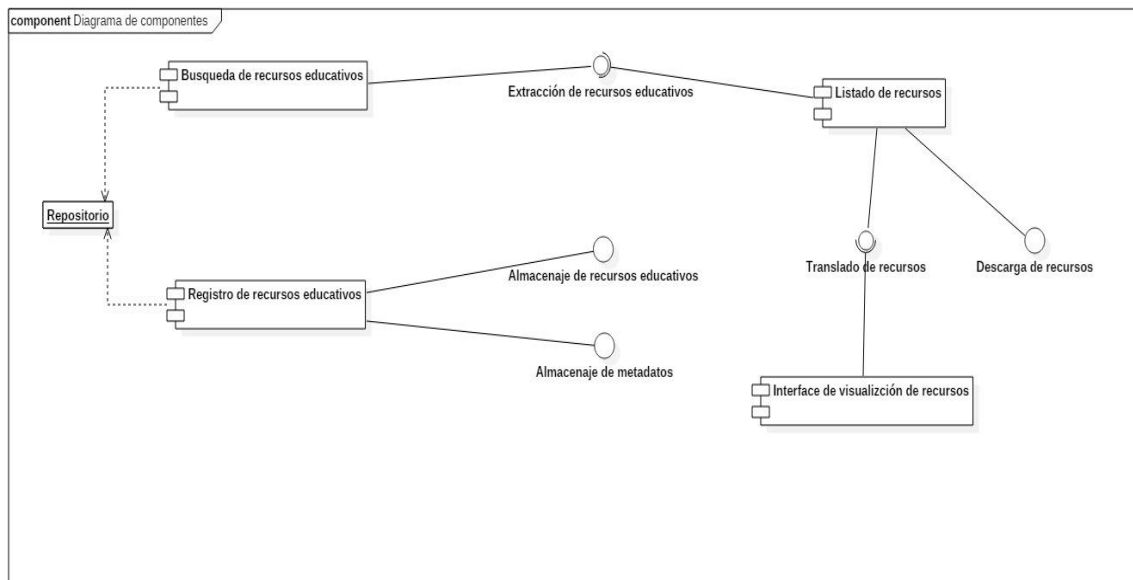


Figura 16. Diagrama de componentes e interacción entre los mismos del repositorio de recursos educativos.

En la figura 17 se presenta el diseño de la arquitectura de manera simple, dicho diseño abarca elementos de los estilos arquitectónicos orientado a servicios y basada en capas.

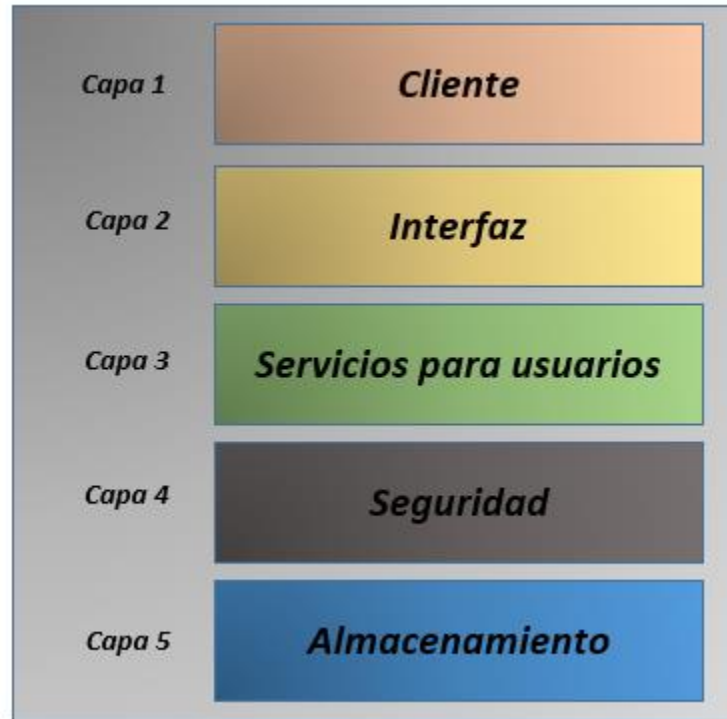


Figura 17. Diagrama general de la arquitectura propuesta.

La arquitectura se descompone en las siguientes capas:

- **Capa 1 Cliente:** La capa cliente es la parte de un sistema a través del cual, los usuarios pueden acceder al repositorio de recursos educativos. Además de esto, dicha capa permite al sistema poder identificar desde donde viene el usuario, es decir, desde que tipo de plataforma pretende acceder al sistema.
- **Capa 2 Interfaz de usuario:** La capa de interfaz de usuario es la parte del sistema que permite a los usuarios puedan interactuar con el sistema, en esta capa también se administra el inicio de sesión del usuario, así como también habilita la capacidad de mostrar la visualización en línea de los recursos, además de los recursos recuperados a través del buscador del repositorio

- **Capa 3 Servicios para el usuario:** En esta capa nos encontramos con los servicios que se pretenden ofrecer al usuario, entre ellos encontramos la descarga de los recursos educativos, el servicio de búsqueda de recursos que se utiliza para encontrar los recursos en el repositorio, el servicio de registro de recursos educativos que es una funcionalidad para que los usuarios puedan registrar los recursos educativos que deseen almacenar así como un sistema que cataloga dichos recursos, y por último nos encontramos el sistema que permite la interoperabilidad del repositorio de recursos educativos.
- **Capa 4 Seguridad:** En esta capa del repositorio nos encontramos con los servicios que trabajan en la seguridad del repositorio. Entre dichas tareas nos encontramos método de encriptación de contraseñas e inyección SQL.
- **Capa 5 Almacenamiento:** En esta capa del repositorio es donde se almacenan cada uno de los recursos educativos que los usuarios deciden almacenar en el repositorio. Los recursos educativos se almacenan de acuerdo al tipo al que pertenecen. Esto se puede debido al sistema catalogador de recursos educativos. Dichos recursos se separan para facilitar la recuperación de la misma dada la estructura del método de recuperación de recursos educativos del repositorio.

En la figura 18, se presenta la arquitectura de manera más detallada. La arquitectura orientada a servicios en capas, compuesta de 5 capas, cada una de las cuales tiene un papel determinante para el correcto funcionamiento del repositorio de recursos educativos presente en la universidad Autónoma de Aguascalientes.

En los siguientes apartados se describen con detalle los servicios más importantes que forman cada una de las capas.

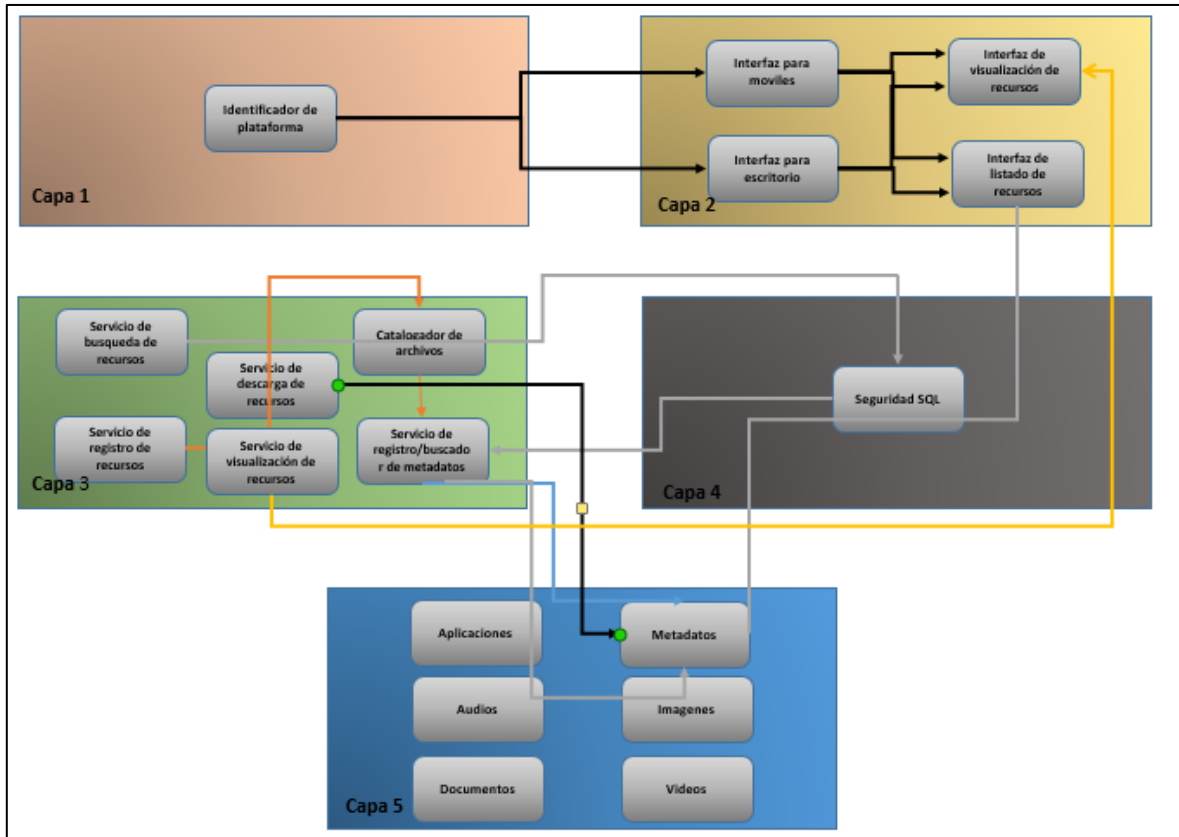


Figura 18. Diagrama detallado de la arquitectura propuesta para el repositorio.

4.2.1 Capa 1 Cliente

El objetivo de la arquitectura es generar un repositorio capaz de almacenar los recursos que le son administrados, además de poder visualizar dichos recursos en línea. Para lograr esto primero, se maneja una arquitectura orientada a servicios basada en capas.

En la primera capa nos encontramos con un elemento que se denomina Identificador de plataforma como podemos observar en la figura 16.



Figura 19. Capa 1 denominada como cliente de la arquitectura propuesta.

La definición de esta capa se debe a unas de las problemáticas a resolver, se trata de poder ofrecer a los usuarios tanto de dispositivos móviles, como de computadoras de escritorio una interfaz que facilite el uso de la aplicación.

Para lograr definir una interfaz, lo primero que debemos hacer es definir que plataforma se está usando para acceder a la aplicación web del repositorio. Para ello haremos uso del lenguaje PHP.

El código mostrado en el anexo 1 pertenece a la capa anteriormente mencionada, y la finalidad es definir desde que plataforma se accede a la aplicación web.

El código en el anexo 1, tiene como función primero determinar si el usuario accede desde una tableta, móvil o un pc de escritorio (Laptop y Pc de escritorio se considera lo mismo). Es importante mencionar que para la identificación de los dispositivos móviles como celulares, se genera en el código un arreglo de nombres de algunos dispositivos (Si el dispositivo móvil celular no se encuentra en la lista, se definirá como tableta), por lo que no siempre detectara correctamente si un dispositivo que accede es un celular o una tableta. Sí el código define que es una tableta o un móvil, lo llevara a una interfaz que se ajuste o se acerque a las necesidades del tamaño de pantalla del dispositivo. En caso de que el código anterior detecte que no es un dispositivo móvil (tableta o celular) automáticamente se define como una computadora de escritorio. Por lo que se envía al usuario a una interfaz definida para computadoras de escritorio.

Como se mencionó anteriormente, es importante para esta arquitectura definir desde donde se está accediendo a la aplicación web del repositorio. Por ello primero se identifica la plataforma de acceso, para poder mostrarle al usuario una interfaz adecuada.

4.2.2 Capa 2: Interfaz de usuario

En esta segunda capa de la arquitectura, se presentan los elementos necesarios para definir las interfaces de la aplicación del repositorio. En la figura 17 se puede observar la estructura de dicha capa.

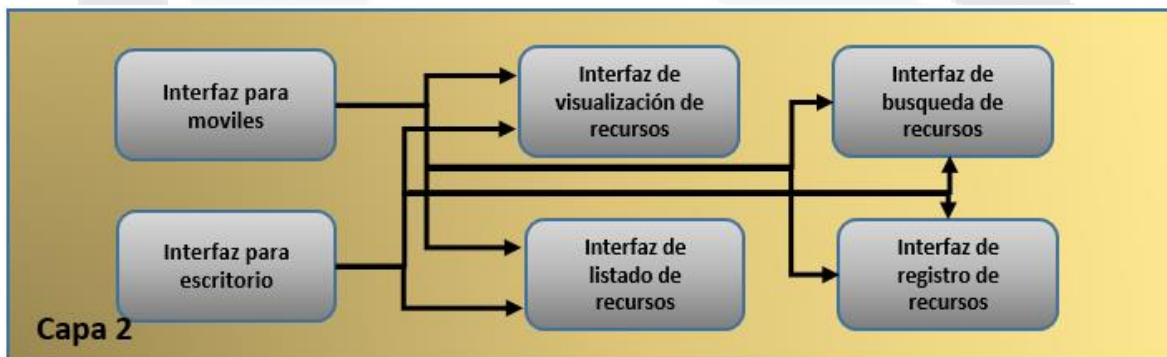


Figura 20. Capa 2 denominada como Interfaz de usuario de la arquitectura propuesta.

Como se puede apreciar en la figura anterior, la capa se compone de seis elementos. Estos elementos se definen a partir de la capa anterior. Esto se debe a que primero la arquitectura define desde que plataforma se accede, para después mostrar ya sea una interfaz para dispositivos móviles, o una para computadoras de escritorio.

A partir de lo anterior, las interfaces de toda la aplicación se definen en la capa 1. De esta manera aseguramos que la interfaz mostrada sea la correcta para la plataforma desde la cual accede el usuario.

Se definen 4 elementos importantes de la capa 2 y estos son:

- Interfaz de búsqueda de recursos
- Interfaz de listado de recursos
- Interfaz de registro de recursos
- Interfaz de visualización de recursos

Interfaz de búsqueda de recursos

Como se mencionó anteriormente, dependiendo de la respuesta de la capa 1 se define la interfaz a mostrar al usuario. Debido a esto es necesario definir una interfaz tanto para dispositivos móviles, como para computadoras de escritorio de cada una de las interfaces en esta capa.

Comenzando con la interfaz de búsqueda de recursos, como se muestra en la figura 19 muestra una interfaz.

Esta interfaz definida en lenguaje HTML, se pueden apreciar un “List box”, que nos permite definir los términos de búsqueda del usuario usando los siguientes posibles campos:

- Autor
- Descripción del recurso
- Nombre del recurso
- Palabras clave del recurso
- Temática del recurso

En la figura 19 se muestra la interfaz definida para la búsqueda de recursos que se implementó.

Esto nos permite definir los términos de búsqueda del usuario. De manera que de la búsqueda en la base de datos sea en base a ese término. Para definir dichos campos, se hizo uso del esquema de metadatos propuesto (Ver figura 5).

El código implementado en la interfaz de búsqueda de recursos, se puede apreciar en el anexo 2.

Interfaz de listado de recursos

En este elemento de la capa 2, se define la interfaz de listado de recursos cuando se realiza una búsqueda de recursos. Esto se logra en base a la información de recursos encontrados en base a la búsqueda. Es decir, se recupera de la base de datos la información relacionada a los recursos que cumplen con la característica de la búsqueda. En el caso práctico, es decir el prototipo implementado, la información que se recupera es la siguiente:

- Nombre del recurso
- Descripción del recurso
- Autor de recurso
- Tema del recurso
- Tipo de recurso (extensión del archivo)
- Link para visualizar en recurso en línea
- Link para descargar el recurso en línea

Cabe señalar que la información recuperada de la base de datos depende de lo que el programador desee. Por lo que no se limita únicamente a recuperar esta información, sino que se puede recuperar más información dependiendo de qué datos fueron almacenados durante el registro de un recurso.

El código implementado en la interfaz de listado de recursos, se puede apreciar en el anexo 3.

Interfaz de registro de recursos

La interfaz de registro de recursos, es uno de los elementos importantes del repositorio. Esta interfaz es la que interactúa con el usuario, ofreciéndole de manera visual los pasos para registrar un recurso en el repositorio.

Para registrar un recurso se requiere suministrar a la interfaz la siguiente información:

- Nombre del recurso
- Descripción del recurso
- Tipo del recurso
- Autor del recurso
- Tópico del recurso
- Palabras clave del recurso
- El recurso como archivo (se ofrece una interfaz proveída por HTML 5, para subir el archivo)

Una vez llenado el formulario, se procede a enviar dicha información, en caso de que el recurso sea guardado con éxito, este le avisara al usuario y lo mandara a la página de inicio.

Interfaz de visualización de recursos

La interfaz de visualización de recursos es el elemento encargado de definir la interfaz que se ofrece al usuario cuando este decide visualizar un recurso en línea. Para la visualización de recursos se tienen cuatro posibilidades siendo estas:

- Visualizar un recurso de audio
- Visualizar un recurso de documento
- Visualizar un recurso de imagen
- Visualizar un recurso de video

Esta interfaz de visualización se accede desde el listado de recursos, es decir cuando se realiza una búsqueda y el usuario decide visualizar uno de los recursos, se le muestra en dicha interfaz una link para dicha visualización.

Para la visualización de audio se muestra una interfaz, que cuenta con controles, para esto se implementó parte de las características de HTML (Ver anexo 4) la interfaz es la siguiente:

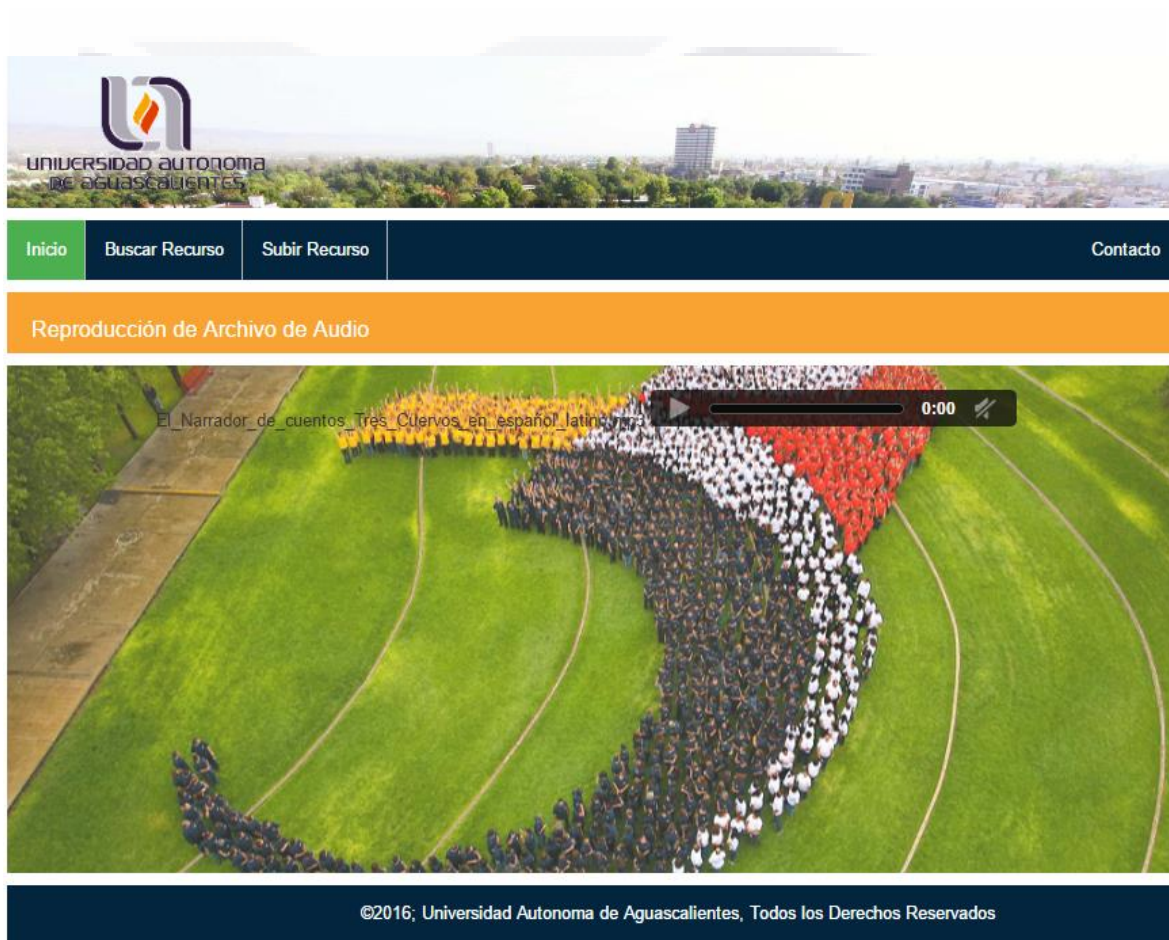


Figura 21. Interfaz de reproducción de audio.

Para la visualización de documentos, existen diferentes métodos para poder leer dichos documentos. Para el caso práctico, se utiliza una librería de Google Docs. La cual permite la visualización de dicho documento, solo con pasarle dicho documento (Ver anexo 5). La interfaz implementada es como un lector de documentos en línea, y se aprecia en la figura 19,

El siguiente tipo de archivo para visualizar es la imagen. Para este tipo de archivo, no es necesaria la implementación de ningún código especial. Simplemente se manda a llamar el archivo de imagen, y se abre en una ventana nueva (Ver anexo 6).

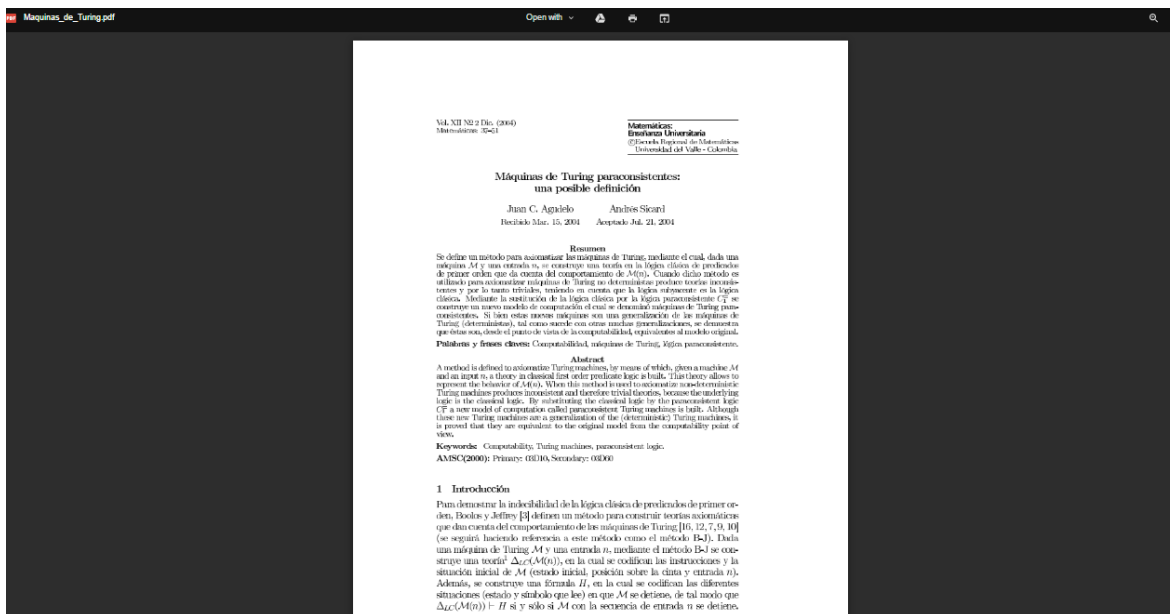


Figura 22. Interfaz de visualización de documentos en línea.

Por ultimo tenemos el recurso del tipo de video. Para esto se utiliza también una característica de HTML 5. Para la reproducción del archivo, primero se manda a llamar, y luego dicho archivo es pasado al reproductor de HTML 5, el cual incluye controles (Ver anexo 7). Para esto se implementó una mezcla de HTML 5 y PHP. La interfaz de reproducción de archivo implementada se muestra en la figura 20.

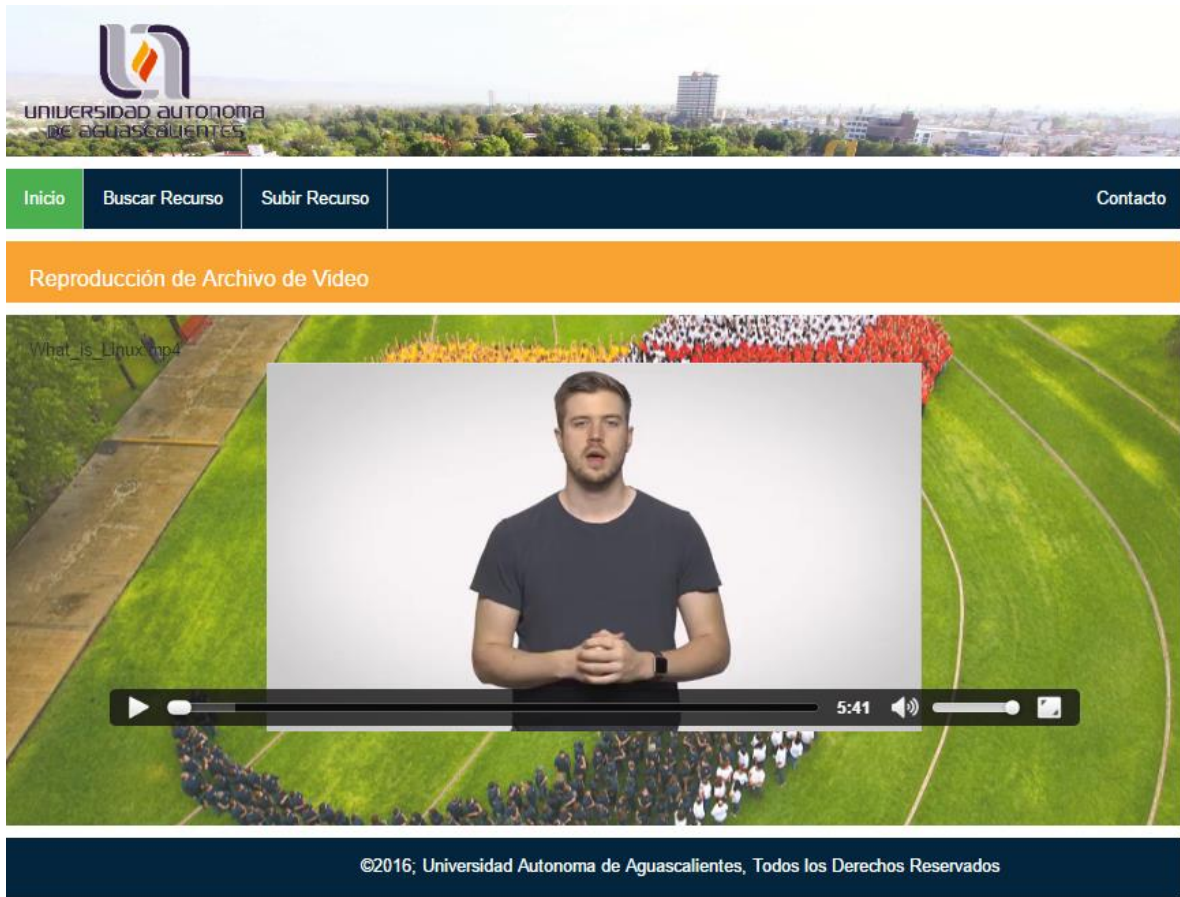


Figura 23. Interfaz de reproducción de recursos del tipo video.

De esta manera se concluye la visualización de recursos en línea. Y podemos pasar a la siguiente capa de la arquitectura.

4.2.3 Capa 3: Servicios para el usuario

Uno de los elementos más importantes para la arquitectura definida, es la capa 3. En esta capa se definen los servicios que se pretenden ofrecer, por lo que encontramos en esta capa es prácticamente la lógica de programación necesaria para que dichos servicios funcionen. En la siguiente figura podemos observar la estructura de dicha capa.



Figura 24. Capa 3 denominada como Servicios para el usuario de la arquitectura propuesta.

Los elementos que encontramos en esta capa son los siguientes:

- Servicio de búsqueda de recursos
- Servicio de registro de recursos
- Servicio de descarga de recursos
- Servicio de visualización de recursos
- Catalogador de archivos
- Servicio de registro/buscador de metadatos

Estos elementos son los encargados de ofrecer los servicios de la arquitectura propuesta. A continuación se hace una descripción de dichos elementos.

Servicio de búsqueda de recursos

El servicio de búsqueda, como se ha mencionado en anteriores ocasiones, es uno de los servicios más importantes de la arquitectura. Para realizar dicha función se utiliza la el código mostrado en el anexo 8.

La lógica detrás de sigue la siguiente estrategia:

1. Se define el tipo de búsqueda (Autor, descripción, palabras clave)
2. Se toma la información introducida en el paso anterior
3. Se compara dicha información con los metadatos almacenados en la base de datos

4. Los elementos que coinciden con lo anterior, se sustrae la información de dichos recursos.
5. Por último se muestra dicha información.

Servicio de registro de recursos

El servicio de búsqueda, como se ha mencionado en anteriores ocasiones, es uno de los servicios más importantes de la arquitectura. Para realizar dicha función se utiliza la el código mostrado en el anexo 9.

El proceso de registro de recursos sigue los siguientes pasos:

1. Se llena el formulario de registro de recursos
2. Se recupera dicha información que hace la función de metadatos
3. Se define el tipo de recurso que se está registrando en base a los metadatos
4. Se procede a almacenar los metadatos del recurso a la base de datos usando el catalogador de recursos
5. Se le avisa al usuario en caso éxito
6. Caso contrario se le comunica el error

El proceso de registro se puede observar en la figura 30. De esta manera el servicio de registros de recursos cumple su función.

Servicio de descarga de recursos

La descarga del recurso, es uno de los servicios que ofrece esta arquitectura, para lograr esto, en el anexo 10 se muestra el código implementado para que el usuario pueda descargar dichos recursos del repositorio.

El servicio de descarga de recursos permite al usuario obtener los siguientes tipos de recursos:

- Archivos de aplicaciones
- Archivos de audio
- Archivos de documentos
- Archivos de imágenes
- Archivos de video

Con lo anterior se cubren los cinco tipos de recursos que pretende almacenar este repositorio. Para lograr esto, se le presenta al usuario una opción después de realizar una búsqueda de recursos, descargar dicho recurso el cual el usuario está interesado. Por lo que de esta manera el servicio cumple con su función.

Servicio de visualización de recursos

El servicio de visualización de recursos es otro de los servicios que pretende ofrecer esta arquitectura. Para esto se implementa el código en el anexo 4, 5, 6 y 7. La lógica detrás para ofrecer dicha visualización se presentó en el apartado anterior.

Catalogador de archivos

Este elemento de la capa 3, es uno de los más importantes. Esto se debe a que el repositorio pretende almacenar cinco tipos de archivos (Aplicaciones, audio, documentos, imágenes y video), por lo que es necesario que durante su búsqueda y registro primero se defina el tipo de archivo para que dicho recurso sea buscado/registrado, en el lugar que le corresponde.

La función principal del catalogador de archivos es administrar los recursos, de esta manera no se tienen los recursos mezclados y permite una búsqueda más ordenada. El catalogador de archivos trabaja en base a los siguientes elementos:

- Archivos de aplicaciones
- Archivos de audio
- Archivos de documentos
- Archivos de imágenes
- Archivos de videos

Esto permite que los recursos sean administrados en base al tipo de archivos, lo cual permite una organización a un nivel mayor.

Servicio de registro/buscador de metadatos

El elemento principal para realizar la búsqueda de recursos en este repositorio, son los metadatos. Debido a esto, se implementa este servicio que es utilizado durante la búsqueda/registro de recursos en el repositorio.

Como se mencionó este servicio se usa en los siguientes elementos:

- Búsqueda de recursos
- Registro de recursos

Este elemento es la lógica que permite el almacenaje/búsqueda de los metadatos (Ver anexo 11). En el modo de registro, este almacena los metadatos en la base de datos. En el modo de búsqueda, este se encarga de buscar la coincidencia de las palabras clave con los metadatos almacenados para recuperar la información de los recursos que coincidan. Con esto se cubren los elementos de capa 3 y podemos pasar a la capa 4.

4.2.4 Capa 4: Seguridad

La capa 4 de la arquitectura tiene que ver con la seguridad del repositorio. La estructura de la capa 4 se presenta en la figura 21.



Figura 25. Capa 4 denominada como Seguridad de la arquitectura propuesta.

Como se puede observar, solo existe un elemento en esta capa. Este elemento pertenece a la seguridad en cuestión a las inyecciones SQL. Para esto primero hablaremos del por qué esta capa.

Los elementos necesarios encontrados para generar una arquitectura orientada a servicios son los siguientes:

- Registro de servicios
- Proceso de negocios
- Servicio
- Descripción del servicio
- Protocolo de comunicación
- Transporte
- Política
- Seguridad
- Transacción
- Administración

Dado que la mayoría de los elementos anteriormente mencionados han sido cumplidos en las capas anteriores, queda pendiente cumplir el elemento de seguridad, que es el elemento que se trabaja en esta capa.

La seguridad del repositorio trabaja en base a evitar un ataque muy común en las bases de datos. Este ataque se llama: Inyección SQL.

Inyección SQL es un método de infiltración de código intruso que se vale de una vulnerabilidad informática presente en una aplicación en el nivel de validación de las entradas para realizar operaciones sobre una base de datos (PHP Manual, 2004). Esta vulnerabilidad trabaja en base a la inyección de caracteres especiales en la base de datos que permiten la operación sobre la misma durante la consulta en la base de datos.

Para evitar este ataque se utiliza el código mostrado en el anexo 12. Para evitar este ataque, es necesario evitar que ciertos caracteres sean introducidos durante las consultas de la base de datos. Por ello se filtra que los únicos caracteres permitidos en las consultas sean los caracteres alfanuméricos, omitiendo símbolos que permitan dicho ataque.

4.2.5 Capa 5: Almacenamiento

La última capa de la arquitectura propuesta tiene que ver con el almacenaje de los recursos y los metadatos de dichos recursos. Los elementos incluidos en esta capa se pueden observar en la figura 26.



Figura 26. Capa 5 denominada como Almacenaje de la arquitectura propuesta.

Como se mencionó, esta capa se centra en el almacenaje de los siguientes elementos:

- Almacenaje de archivos de aplicaciones
- Almacenaje de archivos de audio
- Almacenaje de archivos de documentos
- Almacenaje de archivos de imágenes
- Almacenaje de archivos de video
- Almacenaje de los metadatos de los recursos

Esta capa hace uso de algunos de los elementos de la capa 3. Los servicios de búsqueda, registro y catalogador de archivos se implementan en esta capa. Esto se ve claramente dado que para registrar los recursos y los metadatos, es necesario tener donde depositarlos, y estos se almacenan en esta capa. Lo mismo pasa en la búsqueda de recursos. Se necesita dónde buscar donde almacenan estos recursos así como sus metadatos y también se encuentran en esta misma capa.

Para esta capa no se implementa ningún código en específico, simplemente se toma como referencia para no mezclar las funciones de las capas y tener definido en donde se almacenan los recursos y sus metadatos.

5 Validación de la arquitectura con la implementación de un prototipo real

En este capítulo se mencionan los objetivos y métodos que se implementaran para la validación de la arquitectura implementada. Esto se lograra a través de la implementación de un prototipo real en la red de la Universidad Autónoma de Aguascalientes para el uso de profesores y alumnos de la universidad. El resultado de la implementación ha sido un repositorio de recursos educativos.

5.1 Objetivos y servicios

Los objetivos a conseguir con el desarrollo del repositorio han sido los siguientes:

- Permitir las búsquedas de recursos educativos almacenados en el sistema usando el repositorio de recursos educativos.
- Permitir el registro de recursos educativos en el sistema usando el repositorio de recursos educativos usando metadatos.
- Permitir la descarga de los recursos educativos del sistema usando el repositorio de recursos educativos.
- Permitir la visualización de los recursos educativos en línea usando el repositorio de recursos educativos. Considerando que entre los recursos educativos que se pueden visualizar en línea serán los tipos: Imágenes, videos, audios y documentos.
- Definir un conjunto de metadatos sencillo para el almacenamiento e identificación de cada uno de los recursos educativos en el repositorio.

Los servicios implementados han sido los que se indican en la siguiente tabla. En la siguiente tabla se indican en que capa se encuentra cada servicio, el servicio y un nombre definido para el servicio.

Capa	Servicio	Nombre de la implementación
Capa 1 – Capa cliente	Identificación del cliente	Identificador de cliente
Capa 3 – Capa servicios para usuario	Visualizar recursos en línea	Visualizador de recursos
Capa 3 – Capa servicios para usuario	Descarga de recursos en línea	Descargador de recursos
Capa 3 – Capa servicios para usuario	Búsqueda de recursos	Búsqueda de recursos
Capa 3 – Capa servicios para usuario	Registro de recursos	Registro de recursos
Capa 3 – Capa servicios para usuario	Registro de metadatos	Registro de metadatos

Tabla 2. Definición de servicios y nombre de la implementación.

5.2 Implementación

Para el desarrollo del sistema, se ha elegido la plataforma PHP como lenguaje para definir toda la lógica detrás de los servicios ofrecidos por el repositorio, y para la parte de la base de datos, se elige MySQL. La elección de estas plataformas de desarrollo, se eligieron después de analizar los diferentes lenguajes que permitieran una interoperabilidad y flexibilidad para el desarrollo del sistema. Esto se debe a que el desarrollo web demanda cada vez más interactividad y capacidad de personalización de los sitios; y las limitaciones que imponen enfoques como el de HTML pueden ser superadas empleando tecnologías como PHP y MySQL (Weilling Et Al, 2005).

Para el desarrollo de las interfaces de la aplicación web del repositorio, se eligió HTML en conjunto con CSS. Esto ofrece al desarrollador las facilidades para incrustar formularios, elementos de HTML y personalizar dichos elementos y formularios usando CSS.

Ahora para la implementación del prototipo, se realizó una solicitud a la Universidad Autónoma de Aguascalientes espacio en sus servidores para montar esta aplicación.

Para comenzar la implementación, como se mencionó anteriormente, es necesario tener disponible las siguientes características:

- Servidor apache en su versión más actual 2.4.18 (con soporte a PHP y MySQL activado)
- PHP en su versión más actual (para el prototipo se pidió una versión 5.6)
- Administrador de base de datos MySQL versión 5.6
- Administrador PHPmyAdmin versión 4.5.3.1

Para montar este software en un equipo, se utilizó un servidor que cuenta con una serie de características que le permitan correr dichas aplicaciones. Dichas características son las siguientes:

- **Sistema operativo:** Linux distribución CentIOS 6.5
- **CPU:** Intel(R) Xeon(R) CPU E5310 @ 1.60GHz 8 core: x86_64
- **RAM:** 4GB
- **Tarjeta de red:** NetXtreme II BCM5708 Gigabit Ethernet
- **Disco duro:** Seagate: ST373455SS 70 GB

Siendo estas las características hardware que se tienen en el servidor. Dichas características no tienen que ser necesariamente estas, pueden ser menor o mayor dependiendo de la capacidad que se le quiera dar al repositorio.

Por ultimo lo que sigue en la implementación de la arquitectura del repositorio, es la implementación de la arquitectura mostrada en el capítulo 4.1 en la figura 18. Para lograr esto, se definen los siguientes pasos en base al diseño de la arquitectura:

- Implementación de los elementos de la capa 1
- Implementación de los elementos de la capa 2
- Implementación de los elementos de la capa 3
- Implementación de los elementos de la capa 4
- Implementación de los elementos de la capa 5

Para la descripción y ver los contenidos y código que se implementaron para el desarrollo de esta arquitectura, referirse al capítulo 4.2, en este apartado mencionado se definen los elementos desde la capa 1 hasta la capa 5 y se muestra el código implementado.

Siguiendo los pasos anteriormente mencionados, se implementó la arquitectura en el servidor de la Universidad Autónoma de Aguascalientes.

Para fines prácticos, en la figura 24, se presenta un diagrama de la estructura general del repositorio a nivel de desarrollo.

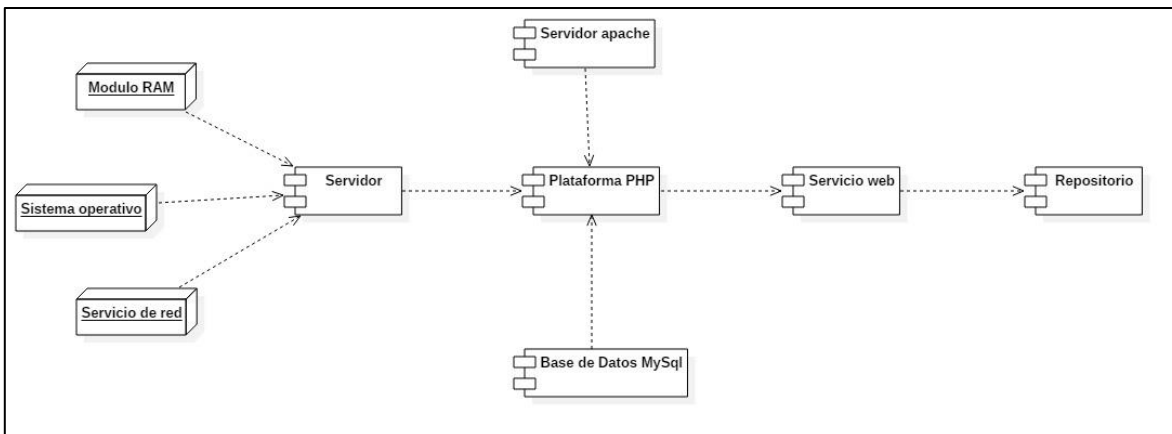


Figura 27. Diagrama de la estructura general del repositorio de la Universidad Autónoma de Aguascalientes

La figura anterior se presenta a un nivel abstracto, por lo que también es necesario presentar un diagrama del funcionamiento del sistema, para ello en la figura 25 podemos observar cómo se comporta el funcionamiento del repositorio de recursos educativos.

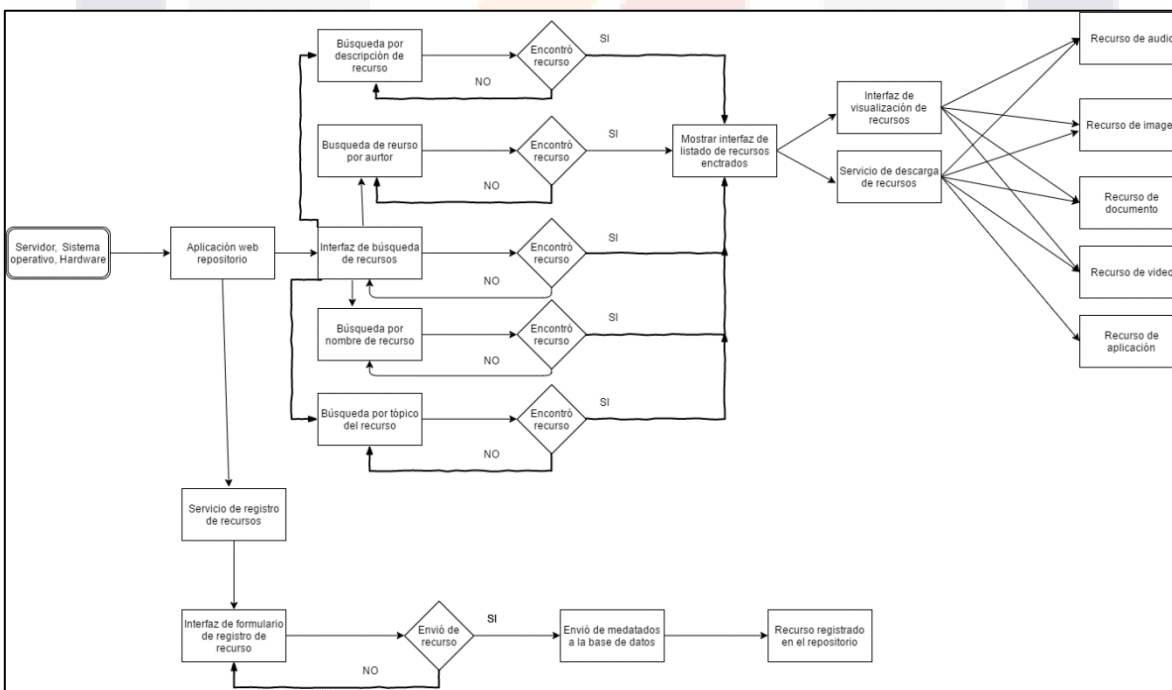


Figura 28. Diagrama general del comportamiento del repositorio de recursos educativos.

5.3 Validación de servicios y perfil del usuario

La variedad de servicios ofrecidos por el repositorio de recursos educativos necesitan ser validados, es decir para demostrar que dichos servicios pueden ser puestos en la práctica. Para ello se implementa una serie de pruebas para cada uno de los servicios ofrecidos por el repositorio de recursos educativos. Cada prueba implementada tendrá algún objetivo en específico que ayudara a validar la funcionalidad del servicio.

Las pruebas que se implementaran, abarcan los siguientes servicios:

- Identificador de plataforma
- Búsqueda de recursos educativos en el repositorio
- Registro de recursos educativos en el repositorio
- Prueba de descarga de recursos educativos del repositorio
- Prueba de visualización para cada uno de los tipos de recursos almacenados en el repositorio (imágenes, audio, video y documentos)

Como se mencionó, el objetivo de cada una de las pruebas mencionadas es para validar cada uno de los servicios que ofrece la arquitectura. A continuación se definirá a mayor detalle cada una de las pruebas. Para la aplicación de las pruebas primero es necesario definir las condiciones y los escenarios en los que aplicaran dichas pruebas.

Otro elemento importante a definir para la ejecución de las pruebas son los escenarios de pruebas. Para lograr esto se define que los escenarios de pruebas se conformaran de tres etapas:

- **Antes:** En esta parte del escenario de prueba, se definen las condiciones y elementos necesarios para la ejecución de la prueba (Servicios web, interfaces, dispositivos, etc.).
- **Durante:** en esta parte del escenario de prueba, se definen los pasos necesarios para ejecutar la prueba.
- **Después:** En esta parte del escenario de prueba, se define el formato para registrar los resultados de la prueba ejecutada.

Además de esto es necesario definir una serie de dispositivos a implementar para dichas pruebas, para el caso práctico se definen los siguientes dispositivos:

Lista de dispositivos y sus características			
Nombre	Sistema Operativo	Navegador web	Cantidad de memoria RAM
Galaxy Tab S2	Android 5.02	Google Chrome Build 49.0.2623.105	2 GB
Galaxy Tab 4	Android 5	Google Chrome Build 49.0.2623.105	2 GB
Nokia Lumia 920	Windows 8.1	Internet Explorer 11	1 GB
Laptop Lenovo Y510P	Windows 10	Google Chrome Build 49.0.2623.110 m	8 GB

Tabla 3. Lista de dispositivos a implementar en las pruebas.

Si bien no es necesario implementar los mismos dispositivos para la ejecución de estas pruebas en otros repositorios. Para el caso práctico se definió esta lista de dispositivos, tratando de abarcar dispositivos que usan diferentes plataformas y diferentes sistemas operativos.

De esa manera se define la metodología a implementar en las pruebas que se definen a continuación.

Perfil de usuario:

- Alumnos de la Universidad Autónoma de Aguascalientes.
 - ✓ Se incluirán todas las carreras que se encuentren dentro de dicha Universidad.
 - ✓ Rango de edad entre 18 a 40 años.
- Personas externas a la Universidad.
 - ✓ Personas con mínimo de preparatoria terminada o cursándola.
 - ✓ Rango de edad entre 18 a 40 años.

Justificación del perfil de usuario

Los humanos cambian y crecen en muy diversos aspectos durante el periodo de los 20 a los 40 años, edades límite aproximadas que la mayoría de los científicos han establecido para definir al joven adulto.

En el sentido común se señala que los adultos piensan de una manera diferente a los niños; pueden mantener sostener diferentes clases de conversaciones, comprender material mucho más complicado y resolver problemas más allá de las capacidades de la mayoría de los niños o aun de muchos adolescentes. (Fernández, 2006).

Decidimos por ello delimitar los perfiles de usuarios con estas edades, dando un rango de edad mínimo de 18 años, pensando en los usuarios que estudian la educación media superior, ya que presentan gran flexibilidad en el estudio-aprendizaje. (Ariño, 2008).

Determinamos de igual manera la edad máxima de 40 años porque la edad adulta intermedia, se define aproximadamente como a los años que van de los 40 hasta los 65, donde la inteligencia fluida, es decir, la habilidad de manejar nuevas situaciones, disminuye. Ya que están usualmente más preocupados en cómo utilizar el conocimiento ya adquirido para propósitos prácticos, que en adquirir nuevos conocimiento y habilidades. (Fernández, 2006).

5.3.1 Prueba de identificador de plataforma

Uno de los servicios mencionados consisten en que la aplicación web del repositorio pueda identificar correctamente el dispositivo desde el que se está accediendo a la aplicación. Logrando identificar el dispositivo, podemos mostrar una interfaz correcta para dicho dispositivo.

Objetivo:

Esta prueba permite validar la funcionalidad del servicio de identificador de cliente, es decir identificar si el usuario del repositorio, accede a la aplicación desde un dispositivo móvil, una Tablet, o una computadora de escritorio. Esto con la finalidad de mostrar una interfaz de tamaño adecuada para el usuario.

Para la implementación de la prueba, se plantea el siguiente escenario desarrollado en las siguientes etapas:

Antes:

Para la aplicación de la prueba, es necesario definir las condiciones en las que se plantea ejecutar dicha prueba para ello se define lo siguiente:

Los dispositivos que se planean usar para realizar las pruebas se muestran en la tabla 3. Estos dispositivos no serán cambiados durante las pruebas, esto es para que los resultados de dichas pruebas sean fieles.

Para realizar esta prueba, solo es necesario ingresar al sitio donde se encuentra alojada la aplicación web del repositorio. Automáticamente este detectara si el usuario está usando un dispositivo móvil, una Tablet o una computadora de escritorio.

Para lograr esta detección se muestra un fragmento del código utilizado en el anexo 1.

El código mostrado en el anexo mencionado, su función es verificar la respuesta del cliente (dispositivo), analiza si en la respuesta del cliente encuentra alguna marca de algún dispositivo móvil, para después comparar esta respuesta con un conjunto de marcas predefinidas. Si la comparación es válida, se dice que el usuario accede desde un dispositivo móvil. Sucede algo similar para las tabletas, y por último, el código define que el usuario accede desde un computador de escritorio, si la comparación de los dispositivos

móviles o tabletas fue cierta. Después de esta comparación, se reenvía al usuario a la interfaz más adecuada para el usuario.

Durante:

Para la ejecución de la prueba se definen los siguientes pasos para realizar esta prueba:

- Entrar a la aplicación web desde el dispositivo

Para la ejecución de esta prueba, no es necesario realizar una gran cantidad de pasos, ya que solo se necesita entrar a la aplicación web, y este automáticamente detectara y reenviara al usuario a la interfaz más adecuada.

Los pasos mencionados permiten tener un control preciso en la ejecución de la prueba, lo cual evita que el procedimiento para realizar dicha prueba no tenga variaciones entre las ejecuciones.

Es importante señalar que se deben tomar las siguientes consideraciones para ejecutar la prueba:

- Contar con una conexión disponible para los dispositivos
- Usar únicamente los dispositivos mencionados para la ejecución de la prueba
- Verificar que los recursos recuperados por la búsqueda tengan alguna relación con los términos de la búsqueda.

Después:

Al finalizar la prueba, es necesario obtener la evidencia de la ejecución de la prueba. Para ello se utiliza el formato del anexo 13.

Una vez lleno el formato, es necesario recolectar toda la información sobre las pruebas ejecutadas para tener toda la evidencia posible sobre la ejecución de dicha prueba. Esto con la finalidad de garantizar el funcionamiento del sistema.

5.3.2 Prueba de búsqueda de recursos educativos

Una de las principales funcionalidades de los repositorios es la búsqueda de recursos en el mismo. Por lo que para poder verificar que dicho servicio se encuentra en funcionamiento se implementara una prueba de búsqueda de recursos. En si prueba no se tratara de exactitud ni precisión sobre los términos de búsqueda, pero sí que por lo menos encuentre algún recurso relacionado con las palabras clave que el usuario implemento en su búsqueda.

El escenario de prueba se manejara por tres etapas. Cada una de las etapas tiene una funcionalidad distinta, la primera etapa denominada Condiciones de prueba, se definen los dispositivos que usaran para realizar dicha prueba. Además de esto también se definen las características de software así como también las características de hardware de estos dispositivos.

Además de esto para cada prueba se define lo necesario para realizar cada una de las pruebas, entre estas actividades se encuentran, despliegue de servicios web, definición de interfaces de usuario

Prueba: Búsqueda de recursos educativos

Objetivo:

Esta prueba permite validar la funcionabilidad del servicio de búsqueda de recursos, la prueba no está diseñada para medir la precisión ni la exactitud de la búsqueda de recursos. La finalidad de la prueba es verificar que se encuentre algún recurso relacionado con los términos definidos para realizar la búsqueda.

Para la implementación de la prueba, se plantea el siguiente escenario desarrollado en las siguientes etapas:

Antes:

Para la aplicación de la prueba, es necesario definir las condiciones en las que se plantea ejecutar, para ello se define lo siguiente:

Los dispositivos que se planean usar para realizar las pruebas se muestran en la tabla 3. Estos dispositivos no serán cambiados durante las pruebas, esto es para que los resultados de dichas pruebas sean fieles.

A demás de la lista de dispositivo también es necesario definir el servicio web necesario para la ejecución de la prueba.

El código mostrado en el anexo 8 sirve para realizar una búsqueda en la base de datos. Esta búsqueda se basa en dos elementos principales para la búsqueda, el tipo de búsqueda y las palabras clave. Con el tipo de búsqueda podemos definir el tipo de archivo, y las palabras clave para realizar dicha búsqueda. Con estos elementos el servicio web puede recuperar los recursos de aprendizaje para que el usuario pueda usarlos.

Además del servicio web para poder realizar esta prueba, es necesario también definir una interfaz para que el usuario realice la búsqueda en HTML.

Además del servicio web para poder realizar esta prueba, es necesario también definir una interfaz para definir esta prueba se utilizan dos lenguajes, PHP y HTML. Cabe mencionar que se utiliza una función en específico de PHP para poder generar los elementos HTML. Esta función es la de “echo”, la cual permite de alguna manera definir texto en el documento.

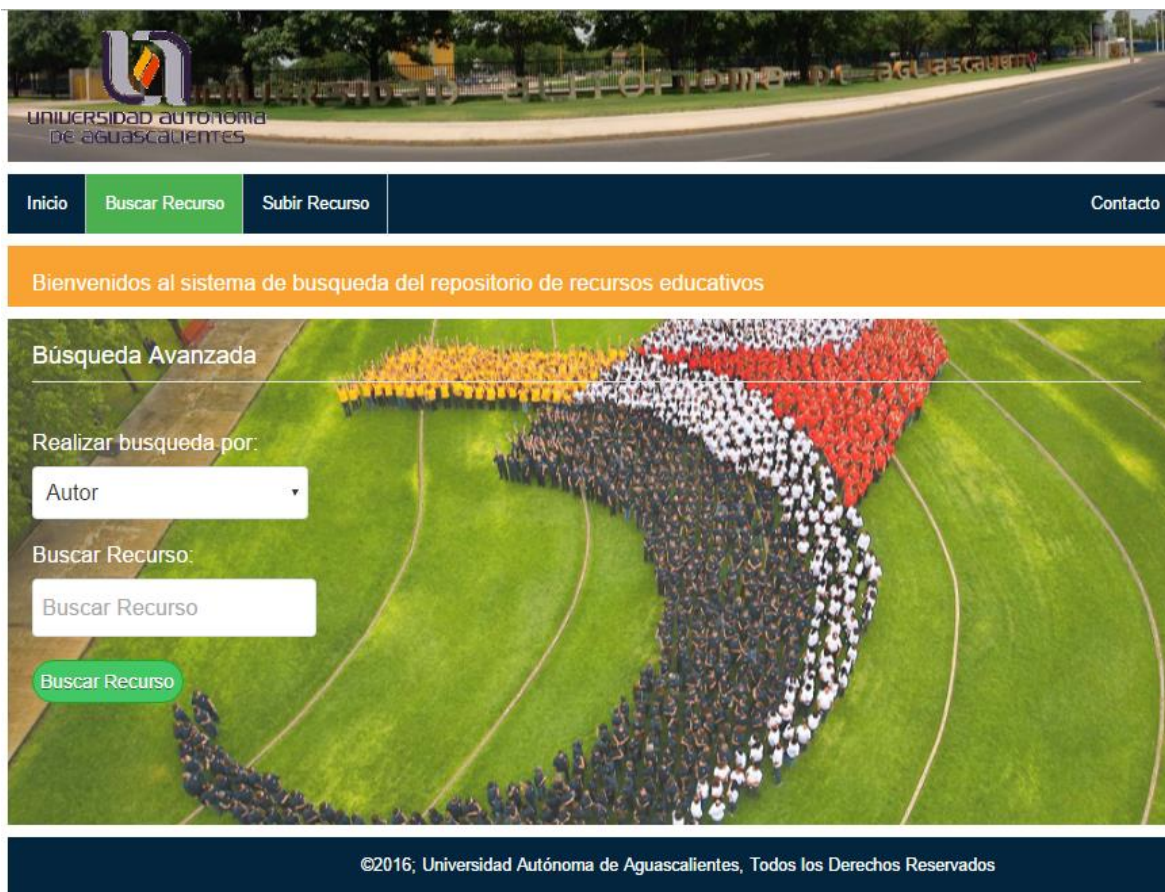


Figura 29. Interfaz para búsqueda recursos implementada en el repositorio.

Con el código mostrado en anexo 3 generamos una tabla para poder mostrar los resultados obtenidos de la búsqueda.

Durante:

Para la ejecución de la prueba se definen los siguientes pasos para realizar esta prueba.

1. Entrar a la aplicación web desde el dispositivo móvil
2. Seleccionar la opción de Búsqueda de recursos educativos
3. Llenar el formulario electrónico para realizar la búsqueda
4. Enviar el formulario para realizar la búsqueda

Los pasos mencionados permiten tener un control preciso en la ejecución de la prueba, lo cual evita que el procedimiento para realizar dicha prueba no tenga variaciones entre las ejecuciones.

Es importante señalar que se deben tomar las siguientes consideraciones para ejecutar la prueba:

- Contar con una conexión disponible para los dispositivos
- Usar únicamente los dispositivos mencionados para la ejecución de la prueba
- Verificar que los recursos recuperados por la búsqueda tengan alguna relación con los términos de la búsqueda.

Después:

Al finalizar la prueba, es necesario obtener la evidencia de la ejecución de la prueba. Para ello se utiliza el formato mostrado en el anexo 14.

Una vez lleno el formato, es necesario recolectar toda la información sobre las pruebas ejecutadas para tener toda la evidencia posible sobre la ejecución de dicha prueba. Esto con la finalidad de garantizar el funcionamiento del sistema.

5.3.3 Prueba de registro de recursos educativos

Los recursos educativos almacenados en un repositorio suelen ser cargados al mismo, ya sea por usuarios del repositorio o por los administradores del repositorio. Esta capacidad permite que el tamaño de la biblioteca pueda crecer y cada vez existan más recursos en el sistema. La prueba de registro de recursos educativos en el repositorio consiste en validar dicha capacidad. Esto quiere decir que todo usuario del repositorio de recursos educativos pueda registrar un recurso educativo, para ello se le presenta una plantilla con ciertos campos que funcionaran como metadatos del mismo recurso para que el recurso pueda ser almacenado.

Objetivo:

Esta prueba permite validar la funcionabilidad del servicio de registro de recursos en el repositorio. La finalidad de la prueba es verificar que los usuarios puedan efectivamente almacenar los recursos educativos en el repositorio.

Para la implementación de la prueba, se plantea el siguiente escenario desarrollado en las siguientes etapas:

Antes:

Para la aplicación de la prueba, es necesario definir las condiciones en las que se plantea ejecutar dicha prueba para ello se define lo siguiente:

Los dispositivos que se planean usar para realizar las pruebas se muestran en la tabla 3. Estos dispositivos no serán cambiados durante las pruebas, esto es para que los resultados de dichas pruebas sean fieles.

A demás de la lista de dispositivo también es necesario definir el servicio web necesario para la ejecución de la prueba. Para este caso se crea y se despliega un servicio web, el lenguaje utilizado para realizar el registro en la base de datos es PHP, dicho elemento se muestra en el anexo 9.

El código mostrado en el anexo mencionado, sirve para generar un registro en la base de datos del repositorio. Para lograr este registro se utiliza una palabra clave en el formulario de registro del recurso, esta palabra clave permite identificar el tipo de recurso que se pretende almacenar (por tipo nos referimos a la extensión del archivo). De esta manera podemos tener organizados todos los recursos educativos en diferentes direcciones, lo cual facilita la administración de los recursos.

Además de registrar los recursos que se pretenden almacenar, también se almacenan los metadatos correspondientes de dicho recurso educativo.

Además del servicio web para poder realizar esta prueba, es necesario también definir una interfaz para que el usuario pueda realizar el registro, para lo cual usamos la siguiente interfaz.

Usando la interfaz anterior facilitamos al usuario la tarea de poder registrar un recurso educativo en el repositorio.

Durante:

Para la ejecución de la prueba se definen los siguientes pasos para realizar esta prueba.

1. Entrar a la aplicación web desde el dispositivo móvil
2. Iniciar sesión en la aplicación web del repositorio
3. Seleccionar la opción de Registrar recurso educativo
4. Llenar el formulario electrónico para realizar el registro.
5. Enviar el formulario para realizar el registro
6. Los pasos mencionados permiten tener un control preciso en la ejecución de la prueba, lo cual evita que el procedimiento para realizar dicha prueba no tenga variaciones entre las ejecuciones.
7. Es importante señalar que se deben tomar las siguientes consideraciones para ejecutar la prueba:
 - Contar con una conexión disponible para los dispositivos
 - Usar únicamente los dispositivos mencionados para la ejecución de la prueba
 - Verificar que los recursos recuperados por la búsqueda tengan alguna relación con los términos de la búsqueda.

Después:

Al finalizar la prueba, es necesario obtener la evidencia de la ejecución de la prueba. Para ello se utilizara el formato anexo 15.

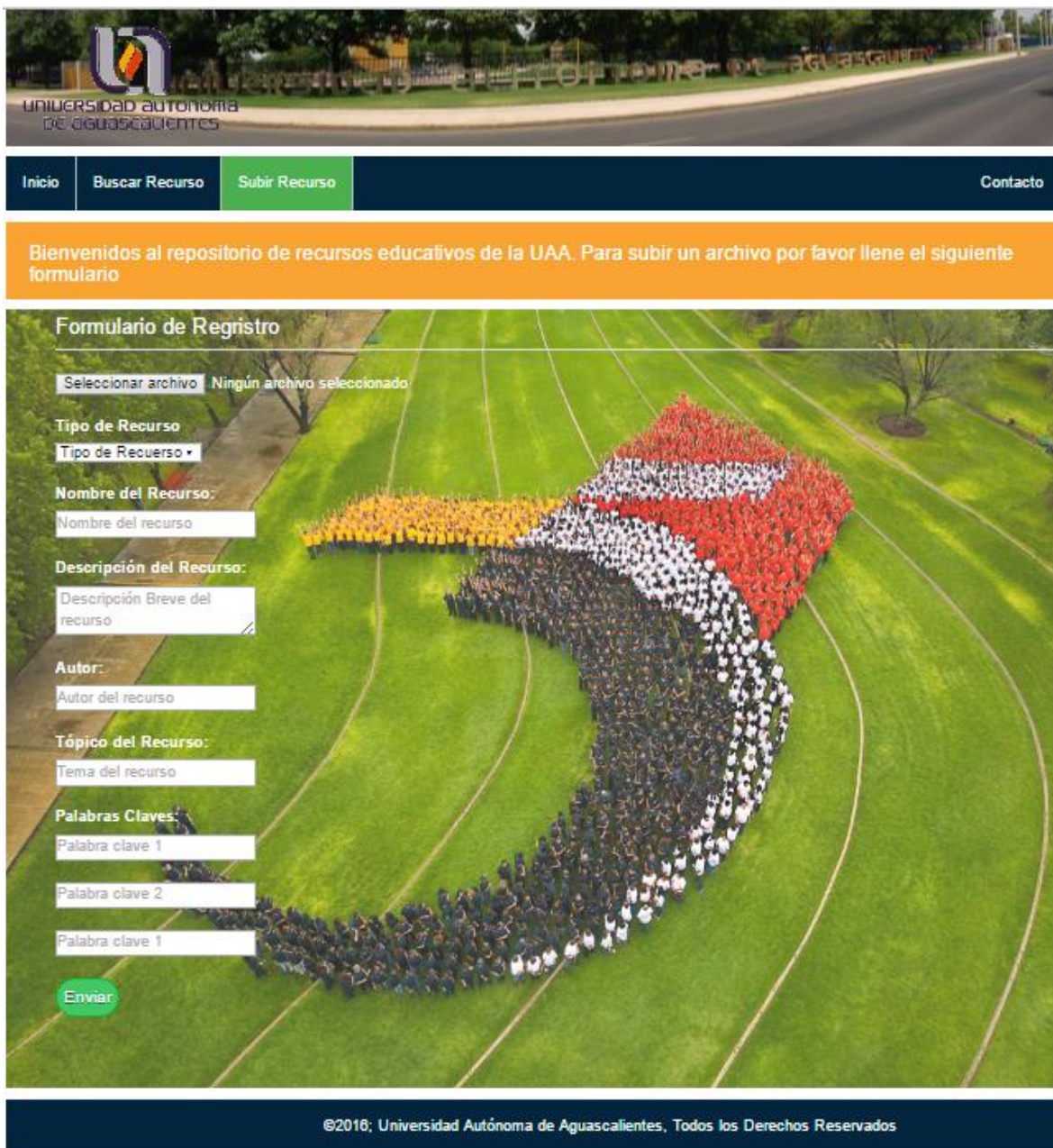


Figura 30. Interfaz implementada para el registro de recursos educativos en el repositorio.

Una vez lleno el formato, es necesario recolectar toda la información sobre las pruebas +ejecutadas para tener toda la evidencia posible sobre la ejecución de dicha prueba. Esto con la finalidad de garantizar el funcionamiento del sistema.

5.3.4 Prueba de descarga de recursos educativos

Cuando un usuario encuentra un recurso en específico que desea poseer en un repositorio, se encuentra la necesidad de poder descargar dicho recurso del repositorio. Par ello el usuario primero debe primero realizar una búsqueda de recurso, encontrar el recurso de que desea descargar y por último, seleccionar dicha opción. La prueba consiste en que los usuarios realicen una búsqueda de recursos, para que luego descarguen algún recurso en específico. Es importante validar este servicio del repositorio, dado que una de las ideas de los repositorios es almacenar y compartir recursos educativos.

Objetivo:

Esta prueba permite validar la funcionabilidad del servicio de descarga de recursos en el repositorio. La finalidad de la prueba es verificar que los usuarios puedan efectivamente además de recuperar dichos recursos educativos, estos puedan ser descargados del repositorio.

Para la implementación de la prueba, se plantea el siguiente escenario desarrollado en las siguientes etapas:

Antes:

Para la aplicación de la prueba, es necesario definir las condiciones en las que se plantea ejecutar dicha prueba para ello se define lo siguiente:

Los dispositivos que se planean usar para realizar las pruebas se muestran en la tabla 3. Estos dispositivos no serán cambiados durante las pruebas, esto es para que los resultados de dichas pruebas sean fieles.

Cabe mencionar que para la ejecución de esta prueba, primero es necesario realizar una búsqueda de recursos educativos en el repositorio. Ya que de esta manera podemos ofrecerle de manera visual al usuario, la opción de descargar dicho recurso.

Para este caso, no son necesarias muchas líneas de código, ya que solo es necesario apuntar al archivo en el repositorio, usando un hipervínculo para obtener dicho archivo y facilitarle la descarga al usuario.

El código anterior se puede implementar ya sea en un documento aparte, como alguna librería, o simplemente puede ser incorporado dentro del mismo servicio web para la búsqueda de recursos educativos del repositorio. No es la finalidad de este documento demostrar como añadirlo como parte de una librería añadida, por lo que simplemente se muestra lo que se utilizó para lograr el objetivo.

Como parte de la interfaz de esta funcionalidad del repositorio, la opción de la descarga del recurso educativo, se refleja dentro de la búsqueda del repositorio a través de un hipervínculo para que los usuarios puedan descargar los recursos desde la misma interfaz.

El código implementado se muestra en el anexo 10.

Durante:

Para la ejecución de la prueba se definen los siguientes pasos para realizar esta prueba.

1. Entrar a la aplicación web desde el dispositivo móvil
2. Seleccionar la opción de Buscar recurso educativo
3. Llenar el formulario electrónico para realizar la búsqueda.
4. Enviar el formulario para realizar la búsqueda
5. Seleccionar la opción de descargar recurso educativo.

Los pasos mencionados permiten tener un control preciso en la ejecución de la prueba, lo cual evita que el procedimiento para realizar dicha prueba no tenga variaciones entre las ejecuciones.

Es importante señalar que se deben tomar las siguientes consideraciones para ejecutar la prueba:

- Contar con una conexión disponible para los dispositivos
- Usar únicamente los dispositivos mencionados para la ejecución de la prueba
- Verificar que los recursos recuperados por la búsqueda tengan alguna relación con los términos de la búsqueda.

Después:

Al finalizar la prueba, es necesario obtener la evidencia de la ejecución de la prueba. Para ello se utiliza el formato mostrado en el anexo 16.

Una vez lleno el formato, es necesario recolectar toda la información sobre las pruebas ejecutadas para tener toda la evidencia posible sobre la ejecución de dicha prueba. Esto con la finalidad de garantizar el funcionamiento del sistema.

5.3.5 Prueba de visualización de recursos educativos

Una de las capacidades que se plantea implementar en el repositorio de recursos educativos, es la opción de que los usuarios puedan visualizar algún recurso encontrado en el mismo. Sin la necesidad de tener que descargarlo. Para esto, se plantea realizar una prueba que consista en validar que los recursos educativos en los formatos: imagen, audio, video y documentos, puedan ser visualizados en línea. Es importante validar este servicio, ya que es una de los servicios que se pretenden ofrecer.

Objetivo:

Esta prueba permite validar la funcionabilidad del servicio de visualización de recursos en el repositorio. La finalidad de la prueba es verificar que los usuarios puedan visualizar los recursos educativos desde el mismo repositorio sin la necesidad de descargarlos.

Para la implementación de la prueba, se plantea el siguiente escenario desarrollado en las siguientes etapas:

Antes:

Para la aplicación de la prueba, es necesario definir las condiciones en las que se plantea ejecutar dicha prueba para ello se define lo siguiente:

Estos son los dispositivos que se planean usar para realizar las pruebas. Estos dispositivos no serán cambiados durante las pruebas, esto es para que los resultados de dichas pruebas sean fieles.

Cabe mencionar que para la ejecución de esta prueba, primero es necesario realizar una búsqueda de recursos educativos en el repositorio. Ya que de esta manera podemos ofrecerle de manera visual al usuario, la opción de visualizar dicho recurso.

Para este caso, no son necesarias muchas líneas de código, ya que solo es necesario apuntar al archivo en el repositorio, usando un hipervínculo para obtener dicho archivo y proceder a visualizar dicho recurso desde el repositorio.

Toda la información anterior fue obtenida de <http://www.w3schools.com/>. Dicha página presenta tutoriales de todo tipo para que los desarrolladores aprendan diferentes lenguajes relacionados al desarrollo web.

Con las líneas de código anteriormente mostradas se implementaron en la aplicación para poder visualizar los recursos educativos. Es importante señalar que se puede añadir estilos CSS para darle un formato a estas reproducciones, pero no es el fin de este documento mostrar dichas posibilidades.

Durante:

Para la ejecución de la prueba se definen los siguientes pasos para realizar esta prueba.

1. Entrar a la aplicación web desde el dispositivo móvil
2. Seleccionar la opción de Buscar recurso educativo
3. Llenar el formulario electrónico para realizar la búsqueda.
4. Enviar el formulario para realizar la búsqueda
5. Seleccionar la opción de visualizar recurso educativo.

Los pasos mencionados permiten tener un control preciso en la ejecución de la prueba, lo cual evita que el procedimiento para realizar dicha prueba no tenga variaciones entre las ejecuciones.

Es importante señalar que se deben tomar las siguientes consideraciones para ejecutar la prueba:

- Contar con una conexión disponible para los dispositivos
- Usar únicamente los dispositivos mencionados para la ejecución de la prueba
- Verificar que los recursos recuperados por la búsqueda tengan alguna relación con los términos de la búsqueda.

Después:

Al finalizar la prueba, es necesario obtener la evidencia de la ejecución de la prueba. Para ello se utilizara el formato anexo 17.

Una vez lleno el formato, es necesario recolectar toda la información sobre las pruebas ejecutadas para tener toda la evidencia posible sobre la ejecución de dicha prueba. Esto con la finalidad de garantizar el funcionamiento del sistema.

5.5 Pruebas realizadas

Para validar las funcionalidades básicas del repositorio se aplicaron a usuarios finales las siguientes pruebas:

- Prueba de búsqueda de recurso
- Prueba de registro de recurso educativo

Ambas pruebas se ejecutaron en dos iteraciones, para esto la selección de usuarios es un elemento fundamental en el proceso de evaluación, para esto se definió el perfil del usuario en el capítulo 5.3.

En el registro de los recursos en la primera iteración de pruebas, no se observó ninguna anomalía. Los usuarios lograron completar el llenado sin complicaciones, como se puede observar en la Figura 31. En el registro de los recursos que se realizó en la segunda iteración de pruebas se obtuvo un 90 % de pruebas positivas y el otro 10% no contaban con los requerimientos del repositorio Figura 32.

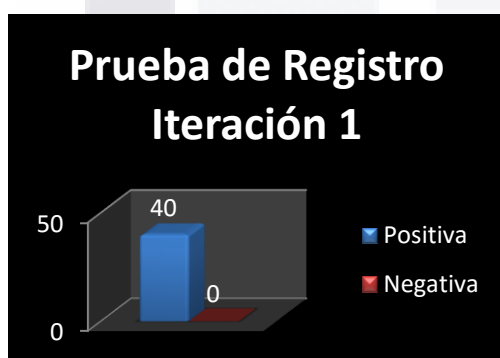


Figura 31. Gráfica de resultados de la prueba de registro en la primera iteración.

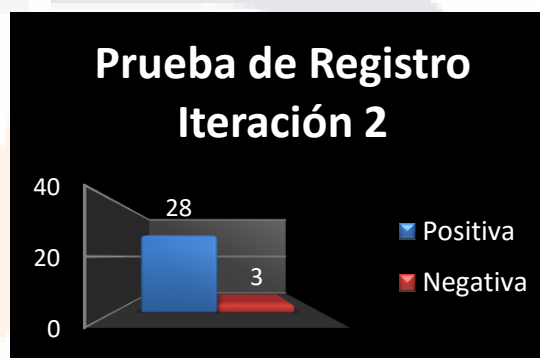


Figura 32. Gráfica de resultados de la prueba de registro en la segunda iteración.

En la prueba de búsqueda de ambas iteraciones se observó que el 100 % de los usuarios pudo realizar la prueba con éxito, el único detalle que se presentó fue que la mayoría de los usuarios realizaban la búsqueda solo por autor, ya que era el dato que más recordaban, como se puede apreciar en las Figura 33 y Figura 34.

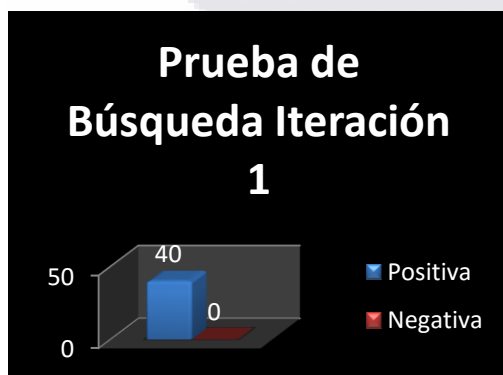


Figura 33. Gráfica de resultados de la prueba de búsqueda en la primera iteración.

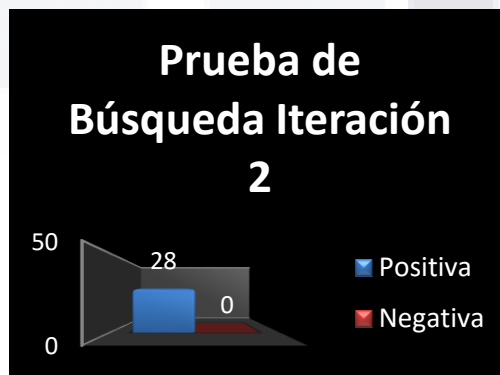


Figura 34. Gráfica de resultados de la prueba de búsqueda en la segunda iteración.

5.4 Conclusiones

En este capítulo se muestran diversos elementos que conciernen a la implementación y validación de la arquitectura desarrollada a través de este documento. Para lograr esto, se siguen algunos pasos claves como; Definición de objetivos, desarrollo de la implementación y por último la definición de las pruebas que validaran que los servicios que se pretenden ofrecer por parte de la arquitectura funcionan.

Los objetivos pretendidos por la arquitectura del repositorio mostrado en este documento, se logran validar en base a las pruebas realizadas al repositorio. Tal como se ha mencionado en apartados anteriores, los servicios más importantes que presenta el repositorio son la búsqueda y la recuperación de recursos educativos, y son los principales objetivos a cumplir, por lo que la validación de estos servicios es de gran importancia.

La implementación de la arquitectura se logra en base a los requerimientos solicitados y al seguimiento e implementación de los servicios y elementos mostrados en el capítulo 4.2.

Por último la validación de los objetivos con las pruebas presentadas en este capítulo. Las pruebas descritas anteriormente, logran validar los servicios que la arquitectura ofrece. Para esto también se considera importante los formatos, ya que con ellos se puede tener evidencia sobre los resultados de dichas pruebas.

6 Conclusiones y trabajos futuros

En este capítulo se recogen las conclusiones de la investigación realizada, algunas de ellas ya explicadas en los capítulos anteriores. Se plantea como se cumplieron los objetivos planteados en este documento, así como también se indica cómo se resolvieron las preguntas de investigación. Además de esto, también se plantean los trabajos futuros.

6.1 Conclusiones, objetivos y aportaciones.

Conviene recordar el objetivo inicial que nos propusimos lograr al iniciar este trabajo de investigación, y que enunciábamos de la manera siguiente:

- **Desarrollar una arquitectura en capas orientada a servicios a un repositorio de recursos educativos que permita el almacenaje/recuperación de recursos educativos en múltiples formatos.**

Para alcanzar este objetivo se plantearon otros objetivos intermedios. Estos objetivos serán mencionados a continuación y como estos fueron cumplidos a través de este documento de tesis:

- **Analizar las tecnologías de software libre existentes que permitan la implementación de una arquitectura en capas orientada a servicios en el desarrollo de un repositorio de recursos educativos y seleccionar una que se adecue a los propósitos de esta tesis.**
- **Definir los elementos arquitectónicos que deberá contener el diseño que funcionara como solución al problema planteado.**
- **Definir una estructura de metadatos para el almacenaje de recursos educativos en un repositorio.**

Comenzando con el primero objetivo, se realizó un análisis de las tecnologías existentes que permitieran una implementación de una arquitectura en capas orientada a servicios para el desarrollo de un repositorio de recursos educativos. Este análisis se basó en los lenguajes de desarrollo actuales y tecnologías de bases de datos que sean tecnologías libres. En la siguiente figura se muestra una síntesis de dicho análisis.

Características	Oracle	MySQL	SQL Server
Interfaz	GUI, SQL	SQL	GUI, SQL, Various
Lenguaje Soportado	Many, including C, C#, C++, Java, Ruby, and Objective C	Many, including C, C#, C++, D, Java, Ruby, and Objective C, PHP	Java, Ruby, Python, VB, <u>.Net</u> , and PHP
Sistema Operativo	Windows, Linux, Solaris, HP-UX, OS X, z/OS, AIX	Windows, Linux, OS X, FreeBSD, Solaris	Windows
Licencia	Propietario	Código Libre	Propietario

Figura 35. Síntesis de análisis de tecnologías para el desarrollo de un repositorio de recursos educativos

En la figura anterior, se muestran algunas de las tecnologías más importantes encontradas para trabajar con bases de datos. Por conclusión respecto a este objetivo específico, se define de la necesidad de tener interoperabilidad, es decir que pueda ser accedido desde cualquier plataforma al repositorio. Además de esto se requiere que la tecnología implementada sea de código libre y no requiera una licencia.

El segundo objetivo; definir los elementos arquitectónicos necesarios para desarrollar la arquitectura. Para cumplir este objetivo, se realizó un análisis de diferentes arquitecturas que cumplen de manera parcial la problemática planteada en este documento de tesis. Este análisis se lleva a cabo en el capítulo 4. En base a los elementos contenidos en dichas arquitecturas, se contempló cuáles de ellos resolvían la problemática en cuestión. Se tomaron dichos elementos y se realizó una tabla de los elementos necesarios para el desarrollo arquitectónico presentada también en el capítulo 4.

Por ultimo cabe mencionar que en base a este análisis se realizó una publicación en el congreso Ccita 2015 en Miami.

- Luna, Mario., Álvarez, Francisco., Muñoz, Jaime. *Una Arquitectura en Capas Orientada a Servicios para el Desarrollo de un Repositorio de Recursos de Aprendizaje ArCaOrSe* (2015)

Se realizó una segunda publicación en base a los resultados obtenidos del repositorio. Dicho trabajo fue aceptado, pero aun no publicado por el XI conferencia latinoamericana de objetos y tecnologías de aprendizaje (LACLO 20016) en Costa rica.

- Luna, Mario., Álvarez, Francisco., Muñoz, Jaime, Vargas, Miguel. *Implementación de Arquitectura ArCaOrSe de un Repositorio de Recursos Educativos y Obtención de Resultados Preliminares de Pruebas Realizadas a la Implementación*

Por último el tercer objetivo: Definir una estructura de metadatos para el almacenaje de los recursos en el repositorio. Este objetivo se cumplió en base a un análisis realizado sobre las estructuras y estándares de metadatos actuales, entre estos se encuentran:

- DDCMI
- LORM
- SCORM

Se definió que dichos estándares son muy complejos debido a su estructura y dadas las necesidades para almacenar recursos educativos, se propuso una estructura para almacenar estos recursos en el repositorio. Esta estructura se presenta en el capítulo 3.2.

Para finalizar las conclusiones de esta tesis, cumpliendo los objetivos mencionados anteriormente, se concluye que el objetivo principal mencionado anteriormente, fue demostrado en el capítulo 4.2. Para lograr esto, se utilizó la tecnología seleccionada, se implementaron los elementos arquitectónicos necesarios y se implementó la estructura de metadatos para definir la arquitectura que esta tesis propone.

Por ultimo cabe mencionar que esta arquitectura fue implementada en la Universidad Autónoma de Aguascalientes. En este prototipo de repositorio se implementaron algunas de las pruebas mencionadas en el capítulo 5. Las pruebas implementadas a dicho prototipo fueron las siguientes:

- Prueba de identificador de plataforma
- Prueba de búsqueda de recursos
- Prueba de registro de recursos

Estas pruebas se llevaron a cabo en el prototipo y los resultados se muestran en la figura 36. Los resultados obtenidos de dichas prueba, indican que los servicios presentados funcionan. Es importante señalar que al momento de realizar las pruebas, no existía un contenido de recursos educativos de gran número. Pero en base al contenido existente se realizaron las pruebas

Lista de dispositivos y resultados de pruebas								
Nombre	Identificador de plataforma		Búsqueda de recursos			Registro de recursos		
	Iteraciones	Identificado como	Palabras clave utilizadas	Cantidad de resultados devueltos	Búsqueda por	Nombre del recurso registrado	Formato del recurso	Metadatos almacenados correctamente
Galaxy Tab S2	10	Dispositivo móvil	Matemáticas	8	Descripción	Que es la IA.docx	Documento	Si
Galaxy Tab 4	10	Dispositivo móvil	Jorge Bucay	5	Autor	Cubix.apk	Aplicación móvil	Si
Nokia Lumia 920	10	Dispositivo móvil	Aplicaciones	3	Palabra clave	Explicación matemática sobre ecuaciones diferenciales	Vídeo	Si
Laptop Lenovo Y510P	10	Computadora de escritorio	Arquitectura	7	Descripción	Explicación sobre protocolo bluetooth	Audio	Si
Pc escritorio	10	Computadora de escritorio	Inteligencia artificial	7	Descripción	Protocolo de almacenaje de recursos	Imagen	Si

Figura 36. Resultados obtenidos de las pruebas realizadas al prototipo.

6.2 Trabajos futuros

Como continuación del trabajo de investigación objeto a esta tesis, se plantea realizar pruebas a gran escala sobre el repositorio, para validar el funcionamiento del mismo. Así como también realizar publicaciones referenciando el trabajo realizado. Actualmente se tiene un artículo en proceso de evaluación para el congreso; XI Conferencia Latinoamericana de Objetos y Tecnologías de Aprendizaje.

Uno de los elementos más importantes actualmente para los repositorios de recursos educativos abiertos y repositorios de objetos de aprendizaje, tiene que ver con la capacidad de búsquedas federadas. Esta característica no fue agregada a este repositorio, debido a que la finalidad de la arquitectura presentada es la recuperación y almacenamiento de recursos educativos. Siendo esta área una posibilidad de trabajos futuros, la creación de un repositorio de recursos educativos de la mismo índole que la mostrada en este documento de tesis, pero con la capacidad de realizar búsquedas federadas.



7 Referencias

Adamczyk, P., Smith, P. H., Johnson, R. E., & Hafiz, M. (2011). REST and Web services: In theory and in practice. In REST: from research to practice (pp. 35-57). Springer New York.

Alvarez F. Arteaga M. (2011). Avances en objetos de aprendizaje. Experiencia de redes de colaboración en México.

Álvarez F., Muñoz J. (2011) Avances en objetos de aprendizaje, Experiencias de redes de colaboración en México.

Aretio, G. (2005). Objetos de aprendizaje. Características y repositorios.

Brand A., Daly F., Meyers B. (2003). Metadata Demystified. The shediran Press and NISO Press

Barrueco, J. M., & García, C. (2009). Repositorios institucionales universitarios: evolución y perspectivas. Zaragoza: Fesabid, XI Jornadas Españolas de Documentación.

Brown, N., Nord, R., & Ozkaya, I. (2010). Enabling agility through architecture. CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST.

Cáceres, P., Marcos, E., & Kybele, G. (2001). Procesos ágiles para el desarrollo de aplicaciones Web. Taller de Web Engineering de las Jornadas de Ingeniería del Software y Bases de Datos de, 2001.

Cáceres J. (2007) Estudio exploratorio de defectos en registros de metadatos IEEE LOM de objetos de aprendizaje

CanCore (2003) CanCore Guidelines Version 2.0: Meta-Metadata category.

CanCore (2004) CanCore Guidelines for the implementation of Learning Object Metadata

Cano, P., Koppenberger, M., Ferradans, S., Martinez, A., Gouyon, F., Sandvold, V., & Wack, N. (2004, September). Mtg-db: A repository for music audio processing. In Web Delivering of Music, 2004. WEDELMUSIC 2004. Proceedings of the Fourth International Conference on (pp. 2-9). IEEE.

Chazarra Bernabé, J., Requena López, V. M., & Valverde Jerónimo, S. (2010). Desarrollo de un repositorio de objetos de aprendizaje usando DSpace.

Cavanaugh, E. (2006). Web services: Benefits, challenges, and a unique, visual development solution. White paper, Feb, 10.

- Cuomo, G. J. (2005). IBM SOA on the Edge. In Proceedings of the 2005 ACM SIGMOD international Conference on Management of Data (pp. 840-843). ACM.
- Downes, S. (2003). Design and reusability of learning objects in an academic context: A new economy of education. *USDLA Journal*, 17(1), 3-22.
- Enríquez, L. (2004). LCMS y Objetos de Aprendizaje. *Revista Digital Universitaria*, 5 (10), 2-9.
- EDUTEKA. Recursos Educativos Abiertos (REA). [Consultado 23 Oct 2015]. Disponible en: <http://www.eduteka.org/OER.php>
- EDUTEKA. Recursos educativos abiertos y su importancia para la educación. [Consultado 24 Oct 2015]. Disponible en: <http://www.eduteka.org/EntrevistaCyranek.php>
- González, L. Á., & González, M. G. (2003). Repositorio de Objetos de Apoyo al aprendizaje colaborativo. Santiago, Chile: Taller Internacional de Software Educativo.
- Guzmán, C. L., Peñalvo, F. G., & Peco, P. P. (2005). Desarrollo de repositorios de objetos de aprendizaje a través de la reutilización de los metadatos de una colección digital: de Dublin Core a IMS. RED. *Revista de Educación a Distancia*, febrero, año, 4.
- Fernández, N. (2006). Características de Desarrollo Psicológico del Adulto. México. Universidad Autónoma de México
- Hashimi, S. (2003). Service-oriented architecture explained [Consultado 12 Sep 2015] Disponible en: http://www.Matcom.uh.cu/Weboo/_Rainbow/Documents/ONDotNet.com_%20Service-Oriented%20Architecture%20Explained.Pdf.
- Heery, R. & Anderson, S. (2005). Digital Repositories Review. Extraído el 01 de diciembre, 2015, de <http://www.ukoln.ac.uk/repositories/publications/review-200502/>.
- Hernández M. (25 de junio 2011) Definamos Contenido Educativo Digital [<http://ced-unerg.blogspot.mx/2011/06/definamos-contenido-educativo-digital.html>]
- Hodgins, H. W. (2002). The future of learning objects.
- Ioannis G. (2005) Introduction to Web Services
- Islas, M. D. L. Á. S., & Escolar, R. Objetos de Aprendizaje. (2010)
- John C. (2006) Identifying and Planning Services within Policy Framework
- JORUM Team. (2006). E-learning repository systems research watch. Project Document.
- Juan T. (2011), Arquitectura de software para aplicaciones Web
- Lagoze, C., & Payette, S. (1998). An infrastructure for open-architecture digital libraries. Cornell University.

- López-Morteo, G., Briseño, J. L., & Aguilera, M. G. (2007). Sistema INteractivo Distribuido de Repositorios de Objetos de Aprendizaje MatEmáticos (SINDROME). Encuentro Internacional de Computación, 2007.
- López, M. G., Hernández Bieliukas, Y., Beleño, C., Pernalet, D., Miguel, V., & Montaña, N (2008). Un Repositorio basado en Servicios Web para el Sistema Generador de Ambientes de Aprendizaje AMBAR. In V Simposio Pluridisciplinar sobre Diseño y Evaluación de Contenidos Educativos Reutilizables. Universidad Pontificia de Salamanca, Escuela Universitaria de Informática, Salamanca, España (Vol. 20).
- Lorna M., Blinco K., Mason J. (2004). Repository Management and Implementation
- Mabrouk, M. (2008). SOA fundamentals in a nutshell. IBM Corp.
- Machuca, C. A. M. (2009) Estado del Arte: Servicios Web.
- McGreal, R. (2004). Learning objects: A practical definition. International Journal of Instructional Technology and Distance Learning (IJITDL), 9(1).
- Marcano, B. (2008). Juegos serios y entrenamiento en la sociedad digital. Revista Electrónica Teoría de la Educación. Educación y Cultura en la Sociedad de la Información, 9(3), 93-105.
- Martínez, F., Prendes, M. (2007). Matemática en red. Los objetos de aprendizaje en sistemas presenciales de enseñanza secundaria. Recuperado de http://digitum.um.es/xmlui/bitstream/10201/12636/1/Proyecto_OA.pdf
- Monge, S., Ovelar, R., & Azpeitia, I. (2008). Repository 2.0: Social dynamics to support community building in learning object repositories. Interdisciplinary Journal of E-Learning and Learning Objects, 4(1), 191-204.
- Montilva, J., Orjuela, A., & Rojas, M. (2010). Diseño de un repositorio de objetos de aprendizaje implementado con servicios web. Publicado en: Revista Avances en Sistemas e Informática. Vol. 7, 2.
- Mortimer, L. (2002). Objects of desire: Promise and practicality. Learning circuits, 3(4).
- Muñoz Arteaga, J., Álvarez Rodríguez, F. J., Osorio Urrutia, B., & Cardona Salas, J. P. (2006). Objetos de aprendizaje integrados a un sistema de gestión de aprendizaje. Apertura, 6(3), 109-117.
- NISO (2004) Understanding metadata. national information standards organization.
- Rojas, M., & Montilva, J (2013). Diseño de repositorios de objetos de aprendizaje como estrategia de reutilización e integración de contenidos en modelos de educación virtual.
- Russel S., Norving P. (2010) Artificial Intelligence A Modern Approach Third edition.
- Sprott, D., & Wilkes, L. (2004). Understanding service-oriented architecture. The Architecture Journal, 1(1), 10-17.

- OASIS (2006). Reference Model for Service Oriented Architecture 1.0
- OPOSICIONES TIC (2012). Arquitectura SOA orientada a servicios. Fecha de consulta 12 de febrero 2015. <http://oposicionestic.blogspot.mx/2012/08/arquitectura-soa-orientada-servicios.html>
- Orphanoudakis, S. C., Chronaki, C. E., & Vamvaka, D. (1996). I2Cnet: content-based similarity search in geographically distributed repositories of medical images. *Computerized Medical Imaging and Graphics*, 20(4), 193-207.
- Pautasso, C., Zimmermann, O., & Leymann, F. (2008). Restful web services vs. big'web services: making the right architectural decision. In *Proceedings of the 17th international conference on World Wide Web* (pp. 805-814). ACM.
- PHP manual. Inyección SQL. Fecha de consulta 20 de abril 2016. <http://php.net/manual/es/security.database.sql-injection.php>
- Polsani, P. R. (2006). Use and abuse of reusable learning objects. *Journal of Digital Information*, 3(4).
- Quinn, C., & Hobbs, S. (2000). Learning objects and instruction components. *Educational Technology & Society*, 3(2), 13-20.
- UNESCO. Open educational resources (OER). [Consultado 25 Oct 2015]. Disponible en: <http://www.unesco.org/new/es/communication-and-information/access-to-knowledge/open-educational-resources/>
- Valero, C. C., Redondo, M. R., & Palacín, A. S. (2012). Tendencias actuales en el uso de dispositivos móviles en educación. *La Educación Digital Magazine*, 147, 1-21.
- Whitepaper (2006) Web services: Benefits, challenges, and unique, visual development situation.
- Wiley, D. A. (2003). Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy.
- Zapata, M. (2005). Secuenciación de contenidos y objetos de aprendizaje. Recuperado de <http://redalyc.uaemex.mx/redalyc/src/inicio/ArtPdfRed.jsp?iCve=54709510>
- UNESCO (2012). [Consultado 25 Oct 2015] Recuperado de http://portal.unesco.org/geography/es/ev.php-URL_ID=15763&URL_DO=DO_TOPIC&URL_SECTION=201.html
- Welling, L., & Thomson, L. (2005). *Desarrollo web con PHP y MySQL*. C (11) Acceder a la base de datos MySQL desde la web con PHP. (P) 313-319

Anexos

Anexo 1 Código de funcionalidad de la capa 1

```

<?php
$tablet_browser = 0;
$mobile_browser = 0;

if (preg_match('/(tablet|ipad|playbook)|(android(?!.*(mobi|opera
mini)))/i', strtolower($_SERVER['HTTP_USER_AGENT']))) {
    $tablet_browser++;
}

if
(preg_match('/(up.browser|up.link|mmp|symbian|smartphone|midp|wap|phone|a
ndroid|iemobile)/i', strtolower($_SERVER['HTTP_USER_AGENT']))) {
    $mobile_browser++;
}

if
((strpos(strtolower($_SERVER['HTTP_ACCEPT']), 'application/vnd.wap.xhtml+xml') > 0) or ((isset($_SERVER['HTTP_X_WAP_PROFILE']) or
isset($_SERVER['HTTP_PROFILE'])))) {
    $mobile_browser++;
}

$mobile_ua = strtolower(substr($_SERVER['HTTP_USER_AGENT'], 0, 4));
$mobile_agents = array(
    'w3c ', 'acs-
', 'alav', 'alca', 'amoi', 'audi', 'avan', 'benq', 'bird', 'blac',
    'blaz', 'brew', 'cell', 'cldc', 'cmd-
', 'dang', 'doco', 'eric', 'hipt', 'inno',
    'ipaq', 'java', 'jigs', 'kddi', 'keji', 'leno', 'lg-c', 'lg-d', 'lg-g', 'lge-
',
    'maui', 'maxo', 'midp', 'mits', 'mmef', 'mobi', 'mot-', 'moto', 'mwbp', 'nec-
',
    'newt', 'noki', 'palm', 'pana', 'pant', 'phil', 'play', 'port', 'prox',
    'qwap', 'sage', 'sams', 'sany', 'sch-', 'sec-', 'send', 'seri', 'sgh-
', 'shar',
    'sie-', 'siem', 'smal', 'smar', 'sony', 'sph-', 'symb', 't-mo', 'teli', 'tim-
',
    'tosh', 'tsm-', 'upg1', 'upsi', 'vk-v', 'voda', 'wap-
', 'wapa', 'wapi', 'wapp',
    'wapr', 'webc', 'winw', 'winw', 'xda ', 'xda-', 'zte', 'zte');

if (in_array($mobile_ua, $mobile_agents)) {
    $mobile_browser++;
}

if (strpos(strtolower($_SERVER['HTTP_USER_AGENT']), 'opera mini') > 0) {
    $mobile_browser++;
    //Verificación para tablets y headers alternativos
    $stock_ua =
    strtolower(isset($_SERVER['HTTP_X_OPERAMINI_PHONE_UA'])?$_SERVER['HTTP_X_

```

```

OPERAMINI_PHONE_UA']:(isset($_SERVER['HTTP_DEVICE_STOCK_UA'])?$_SERVER['H
TTP_DEVICE_STOCK_UA']:''));
    if (preg_match('/(tablet|ipad|playbook)|(android(?!.mobile))/i',
$stock_ua)) {
        $tablet_browser++;
    }
}

if ($tablet_browser > 0) {
    header ("Location: sistemamovile.php");
    //print "es movil";
}
else if ($mobile_browser > 0) {
    header ("Location: sistemamovile.php");
    //print "es movil";
}
else {
    header ("Location: sistemadestop.php");
    //print "es PC";
}
?>

```

Anexo 2 Código de para interfaz de búsqueda de recursos

```

<table align="center" style="">
  <tr>
    <td>Bienvenidos al sistema de busqueda del repositorio de recursos
educativos</td>
  </tr>
  <tr>
    <td> <form method="post" action="buscardata.php">
Realizar busqueda por:
<select name="tipo">
<option value="autor">Autor</option>
<option value="descripcion">Descripción del recurso</option>
<option value="nombre">Nombre del recurso</option>
<option value="palabra">Palabra clave</option>
</select><Br><Br>
Buscar:
<input type="text" name="palabra" maxlength="100"
placeholder="Palabras clave"><Br><Br>
<input type="submit" value="Realizar busqueda">
</form> </td>
  </tr>
</table>

```

Anexo 3 Código de para interfaz de listado de recursos

```

<?php echo "
  <table border='1' cellpadding='0' cellspacing='0' width='600'
align='center'>
  <tr>
    <td width='150'>".$registro['nombre']."</td>
    <td width='150'>".$registro['descripcion']."</td>

```

```

        <td width='150'>".$registro['autor']. "</td>
        <td width='150'>".$registro['palabra1'].",
".$registro['palabra2']. """, " ".$registro['palabra3']. "</td>
        <td width='150'>".$registro['tema']. "</td>
        <td width='150'>".$tipofile. "</td>
        <td width='150'>".$visualizar. "</td>
        <td width='150'>".$descargar. "</td>
    </tr>
</table>
";
?>

```

Anexo 4 Código implementado para controles de archivos de audio

```

<?php
$audio = $_GET['audio'];
echo '<video id="video" width="600" height="600" preload="auto"
controls>';
    echo "<source src=" . "" . $audio . "" . "" type="video/mp4">';
    echo "Your browser does not support HTML5 video.";
echo '</video>';
?>

```

Anexo 5 Código implementado para lector de documentos

```

<?php $visualizar = '<iframe src=http://docs.google.com/viewer?url=' .
Dirección de tu sitio" . $filedir . '&embedded=true" width="100"
height="100" style="border: none" scrolling="no"></iframe> ' ;
?>

```

Anexo 6 Código implementado para ver imágenes en el repositorio

```

<a target="" . "_blank href="" . $filedir . ">Click aqui para
descargar</a>";

```

Anexo 7 Código implementado para la reproducción de video

```

<?php $video = $_GET['vid'];
echo '<video id="video" width="600" height="600" preload="auto"
controls>';
    echo "<source src=" . "" . $video . "" . "" type="video/mp4">';
    echo "Your browser does not support HTML5 video.";
echo '</video>';
?>

```

Anexo 8 Código para realizar la búsqueda de recursos

```

<?php

switch ($tipo){
    case "autor":
        $sql = "(SELECT * FROM recursos_videos WHERE autor LIKE
'%"$palabra%') UNION (SELECT * FROM recursos_aplicaciones WHERE autor LIKE

```



```
'%$palabra%') UNION (SELECT * FROM recursos_documento WHERE autor LIKE
'%$palabra%') UNION (SELECT * FROM recursos_imagenes WHERE autor LIKE
'%$palabra%') UNION (SELECT * FROM recursos_audio WHERE autor LIKE
'%$palabra%')";
break;
case "descripcion":
    $sql = "(SELECT * FROM recursos_videos WHERE descripcion LIKE
'%$palabra%') UNION (SELECT * FROM recursos_aplicaciones WHERE
descripcion LIKE '%$palabra%') UNION (SELECT * FROM recursos_documento
WHERE descripcion LIKE '%$palabra%') UNION (SELECT * FROM
recursos_imagenes WHERE descripcion LIKE '%$palabra%') UNION (SELECT *
FROM recursos_audio WHERE descripcion LIKE '%$palabra%')";

break;
case "nombre":
    $sql = "(SELECT * FROM recursos_videos WHERE nombre LIKE
'%$palabra%') UNION (SELECT * FROM recursos_aplicaciones WHERE nombre
LIKE '%$palabra%') UNION (SELECT * FROM recursos_documento WHERE nombre
LIKE '%$palabra%') UNION (SELECT * FROM recursos_imagenes WHERE nombre
LIKE '%$palabra%') UNION (SELECT * FROM recursos_audio WHERE nombre LIKE
'%$palabra%')";

break;
case "palabra":
    $sql = "(SELECT * FROM recursos_videos WHERE palabra LIKE
'%$palabra%') UNION (SELECT * FROM recursos_aplicaciones WHERE palabra
LIKE '%$palabra%') UNION (SELECT * FROM recursos_documento WHERE palabra
LIKE '%$palabra%') UNION (SELECT * FROM recursos_imagenes WHERE palabra
LIKE '%$palabra%') UNION (SELECT * FROM recursos_audio WHERE palabra LIKE
'%$palabra%')";

break;
}
?>
```

Anexo 9 Código para realizar el registro de recursos

```
<?php switch ($tipo){
case "aplicaciones":
    $dir = "recursos/aplicaciones/";
    $filedir = $dir . str_replace(' ', '_', $_FILES["archivo"]["name"]);
    $sql = "INSERT INTO recursos_aplicaciones (Campos en la BD)
VALUES (Valores a registrar en la BD)";
break;
case "audio":
    $dir = "recursos/audios/";
    $filedir = $dir . str_replace(' ', '_', $_FILES["archivo"]["name"]);
    $sql = "INSERT INTO recursos_audio (Campos en la BD)
VALUES (Valores a registrar en la BD)";
break;
case "documento":
    $dir = "recursos/documentos/";
    $filedir = $dir . str_replace(' ', '_', $_FILES["archivo"]["name"]);
    $sql = "INSERT INTO recursos_documento (Campos en BD)
VALUES (Valores a registrar en BD)";
```

```

break;
case "imagen":
$dir = "recursos/imagenes/";
$filedir = $dir . str_replace(' ', '_', $_FILES["archivo"]["name"]);
$sql = "INSERT INTO recursos_imagenes (Campos en la BD)
VALUES (Valores a registrar en la BD)";
break;
case "video":
$dir = "recursos/videos/";
$filedir = $dir . str_replace(' ', '_', $_FILES["archivo"]["name"]);
$sql = "INSERT INTO recursos_videos (Campos de la BD)
VALUES (Valores ingresados a la BD)";
break;
}

// Creamos conexión
$conn = mysql_connect($servername, $username, $password, $dbname);
// Validamos conexión
if (!$conn) {
    die("Error de conexión: " . mysql_error());
}

// Configuramos la entrada de datos a la BD
mysql_set_charset('utf8');

// Seleccionamos DB
mysql_select_db('dataset', $conn);
?>

```

Anexo 10 Código implementado para la descarga de recursos

```

<?php if($verificar == ".exe" || $verificar == ".apk"){
    $descargar = "<a target=" . "_blank href=" . $filedir . ">Click
aquí para descargar</a>";
} elseif ($verificar == ".mp3"){
    $descargar = '<a href="' . $filedir . '" > Click para descargar
archivo</a>';
} elseif ($verificar == ".jpg" || $verificar == ".png") {
    $descargar = '<a href="' . $filedir . '" > Click para descargar
archivo</a>';
} elseif ($verificar == ".mp4"){
    $descargar = '<a href="' . $filedir . '" > Click para descargar
archivo</a>';
} elseif($verificar == ".docx" || $verificar == ".pdf") {
    $descargar = "<a target=" . "_blank href=" . $filedir . ">Click
aquí para descargar</a>";
}
?>

```

Anexo 11 Código implementado para el registro de metadatos

```

<?php // Creamos conexión
$conn = mysql_connect($servername, $username, $password, $dbname);
// Validamos conexión
if (!$conn) {
    die("Connection failed: " . mysql_error());
}

```

```
}  
  
// Configuramos los datos de entrada a la BD  
mysql_set_charset('utf8');  
// Selección de BD  
mysql_select_db('dataset', $conn);  
?>
```

Anexo 12 Código implementado para evitar SQL

```
<?php  
// Código para filtrar todos los caracteres especiales durante el  
formulario.  
  
$_MensajeError="ERROR: caracteres no admitidos";  
if (ereg("[^A-Za-z0-9]+", $_POST['consulta'])) {  
echo $_MensajeError;  
}  
else{  
echo '<div class="result">Tu nombre es: '.$consulta.'</div>';  
}  
?>
```

Anexo 13 Formato de prueba de identificador de plataforma

Prueba de identificador de cliente		Formato: IdP1
		No:
<p>Descripción de la prueba: La prueba consiste en utilizar un navegador web, entrar a la aplicación web del repositorio de recursos de aprendizaje y que este identifique correctamente el tipo de dispositivo.</p>		
Fecha de realización de prueba:		
Nombre del dispositivo:		
Tipo de dispositivo:		
Sistema operativo del dispositivo:		
Nombre del navegador utilizado:		
<p>Iteración de pruebas de identificación de dispositivo</p> <p>Resultados de la prueba</p>		
	Positiva	Negativa
1		
2		
3		
4		
5		
6		
7		
8		
9		
10		
		<p>Nota: Marcar con una paloma si el repositorio identifico el dispositivo correctamente</p>

Anexo 14 Formato de prueba búsqueda de recursos

Búsqueda de recursos educativos	Formato: IdP5			
	No:			
<p>Descripción de la prueba: La prueba consisten en validar la capacidad del repositorio en permitir al usuario la búsqueda de recursos educativos en el repositorio</p>				
Fecha de realización de prueba:				
Dispositivo de donde se visualiza el recurso:				
Nombre del navegador utilizado:				
<p>Iteración de pruebas de búsqueda de recursos educativos Resultados de la prueba</p>				
	Nombre del recurso	Tipo	Positiva	Negativa
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				

Nota: Marcar con una paloma si el repositorio permitió el registro del recurso

Anexo 15 Formato de prueba de registro de recursos

Registro de recursos educativos		Formato: IdP4		
		No:		
<p>Descripción de la prueba: La prueba consisten en validar la capacidad del repositorio en permitir al usuario el registro de recursos educativos en el repositorio</p>				
Fecha de realización de prueba:				
Nombre del recurso:				
Dispositivo de donde se visualiza el recurso:				
Nombre del navegador utilizado:				
<p>Iteración de pruebas de registro de recursos educativos Resultados de la prueba</p>				
	Nombre del recurso	Tipo	Positiva	Negativa
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				

Nota: Marcar con una paloma si el repositorio permitió el registro del recurso

Anexo 16 Formato de prueba de descarga de recursos

Descarga de recursos		Formato: IdP3
		No:
<p>Descripción de la prueba: La prueba consiste en validar la capacidad del repositorio de descargar el recurso educativo del repositorio</p>		
Fecha de realización de prueba:		
Nombre del recurso:		
Tipo del recurso:		
Dispositivo de donde se visualiza el recurso:		
Nombre del navegador utilizado:		
<p>Iteración de pruebas de descarga de recurso Resultados de la prueba</p>		
	Positiva	Negativa
1		
2		
3		
4		
5		
6		
7		
8		
9		
10		
<p>Nota: Marcar con una paloma si el repositorio permitió la descarga del recurso</p>		

Anexo 17 Formato de prueba de visualización de recurso

Visualización de recursos		Formato: IdP2		
		No:		
<p>Descripción de la prueba: La prueba consiste en validar la capacidad de visualizar un recurso educativo que previamente fue buscado en el repositorio. Para esto se procede primero a realizar la búsqueda del recurso en el repositorio y luego se procede a seleccionar la opción de visualizar recurso</p>				
Fecha de realización de prueba:				
Nombre del recurso:				
Dispositivo de donde se visualiza el recurso:				
Nombre del navegador utilizado:				
<p>Iteración de pruebas de visualización de recurso educativo Resultados de la prueba</p>				
	Nombre del recurso	Tipo	Positiva	Negativa
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				

Nota:
Marcar con una paloma si el repositorio permitió el registro del recurso

Anexo 18 Carta aceptación a conferencia Ccita 2015

30/7/2016

Gmail - CcITA-2015 notification for paper 51



Mario Alberto Luna Preciado <marioal.luna@gmail.com>

CcITA-2015 notification for paper 51

1 message

CcITA-2015 <ccita2015@easychair.org>

Fri, Jun 5, 2015 at 3:47 PM

To: Mario Alberto Luna Preciado <marioal.luna@gmail.com>

Trabajo titulado: Una Arquitectura en Capas Orientada a Servicios para el Desarrollo de un Repositorio de Recursos de Aprendizaje ArCaOrSe
 Autores: Mario Alberto Luna Preciado
 Número de Referencia en EasyChair: 51

Estimados autores:

Ha concluido el proceso de evaluación de las contribuciones presentadas a la VII edición de la Conferencia CcITA-2015. Como Ud. sabe, más de 70 Doctores de 12 países dedicaron su esfuerzo desinteresado en analizar críticamente las propuestas y de ese modo mantener el prestigio del evento y del libro que se publica anualmente.

Tengo la satisfacción de informarle que su trabajo arriba mencionado, ha sido ACEPTADO como Comunicación. Tenga presente que el trabajo recibió varias críticas y sugerencias importantes que deben subsanarse en la Versión Definitiva.
 Este documento enviado desde EasyChair, es la carta de aceptación formal de su trabajo.

POR FAVOR, LEA DETENIDAMENTE Y CUMPLIMENTE LAS INDICACIONES QUE APARECEN A CONTINUACIÓN:

Tenga en cuenta que, para poder incluir su trabajo en el programa de la Conferencia y el libro, es imprescindible revisarlo y hacer algunos cambios para editar y enviar su VERSIÓN DEFINITIVA. Para ello, siga verifique cuidadosamente la siguiente LISTA DE CHEQUEO que ha sido elaborada a partir de los errores y problemas más comunes observados hasta ahora:

1. REVISIONES: Modifique su trabajo considerando las observaciones, críticas y sugerencias de los revisores
2. ESTILO: Ajuste estrictamente el estilo de su trabajo al estándar SPRINGER. El suyo, será un capítulo del libro. Es responsabilidad de cada autor, velar por cada detalle del Estilo. Revise bien los detalles de la primera página (título, autores, filiación, resumen, palabras clave, márgenes y sangrías). Revise la sección final no numerada de Referencias.
3. FORMATO: Prepare y envíe su trabajo en formato WORD. Las páginas, NO numeradas.
4. TÍTULO: El título no debe ser modificado. No obstante se han detectado casos de títulos inconvenientes por ser muy largos (más de 20 palabras). Tampoco es usual hacer mención de instituciones concretas en el título o mención de siglas que no sean de conocimiento general (P Ejemplo: La U de X). Solo en alguno de estos casos, ajuste su título.
5. REDACCIÓN: Revise a fondo la redacción y la ortografía. Trate de no utilizar párrafos largos. Los párrafos largos aumentan el número de errores y son menos comprensibles. No permita que su trabajo sea publicado con problemas de este tipo.
6. NÚMERO DE PÁGINAS: Aunque hay flexibilidad, trate de ajustarse lo más

<https://mail.google.com/mail/u/0/?ui=2&ik=f5ae17318&view=pt&q=ccita&qs=true&search=query&th=14dc57a6809fdcc8&siml=14dc57a6809fdcc8>

1/3

Anexo 19 Artículo Ccita 2015: Una Arquitectura en Capas Orientada a Servicios para el Desarrollo de un Repositorio de Recursos de Aprendizaje ArCaOrSe

Una Arquitectura en Capas Orientada a Servicios para el Desarrollo de un Repositorio de Recursos de Aprendizaje ArCaOrSe

Mario Alberto Luna Preciado, Francisco Javier Álvarez Rodríguez, Jaime Muñoz Arteaga
Universidad Autónoma de Aguascalientes.
Av. Universidad #904, Ciudad de Aguascalientes CP. 20131, Aguascalientes, Ags. México
{marivul.luna, {jfabar.ana, jmuua}@gmail.com

Resumen. Los repositorios de recursos de aprendizaje hoy en día ofrecen una limitada variedad de servicios. Estos servicios son usualmente búsqueda de recursos, registro de recursos, búsquedas federadas, etc. Sin embargo es posible ofrecer más servicios que le permitan al usuario diversas opciones como visualizar los recursos en línea, soportar diferentes tipos de recursos (extensiones de archivo) en el repositorio, ofrecer seguridad para inyecciones SQL, entre otros. Para ello este documento propone una arquitectura en capas orientada a servicios (ArCaOrSe) para el desarrollo de un repositorio de recursos de aprendizaje. Esta arquitectura pretende incrementar los alcances de los repositorios de recursos de aprendizaje ofreciendo los servicios anteriormente mencionados.

Palabras Clave: Diseño, Repositorio, Recurso de aprendizaje, Arquitectura basada en capas, Arquitectura orientada a servicios, Arquitecturas de software

1 Introducción

Hoy en día existe una gran cantidad de repositorios de recursos de aprendizaje en el internet. Estos repositorios ofrecen funciones para los usuarios como búsqueda/almacenaje/recuperación de recursos, publicar recursos, permitir búsquedas federadas. Siendo estos los servicios básicos de los repositorios de recursos de aprendizaje [1] [2].

En la actualidad existen diversos tipos de repositorios y estos almacenan diversos tipos de archivos. Siendo estos últimos archivos de audio, imágenes, video, aplicaciones ejecutables o aplicaciones móviles y documentos de texto.

Los recursos de aprendizaje no solo se limitan a archivos de documentos de texto, también existen los recursos de aprendizaje en formatos tales como; Audio, Video, Imagen y juegos serios [3] [4].

La idea es generar una arquitectura en capas orientada a servicios para un repositorio de recursos de aprendizaje que conserve las características básicas de los repositorios además de permitir agregar servicios como la visualización de los recursos en línea y soportar múltiples tipos de recursos en el mismo repositorio además de otras funciones que se abordaran más adelante.

Un repositorio digital es un sistema que "permite el almacenaje, descubrimiento y recuperación de metadatos y/o de objetos electrónicos/digitales almacenados a un nivel distribuido o local" [5]. Sin embargo los límites de lo que puede ser llamado LOR permanece borroso dado que no existe un acuerdo común acerca de la naturaleza de los recursos de aprendizaje [6].

2 Arquitectura Orientada a servicios y orientada en capas

La Arquitectura Orientada a Servicios (SOA, siglas del inglés Service Oriented Architecture) es un paradigma para organizar y utilizar capacidades distribuidas que pueden estar bajo el control de dominios [7] [8].

La arquitectura basada en capas se enfoca en la distribución de roles y responsabilidades de forma jerárquica proveyendo una forma muy efectiva de separación de responsabilidades. Garlan y Shaw definen el estilo en capas como una organización jerárquica tal que cada capa proporciona servicios a la capa inmediatamente superior y se sirve de las prestaciones que le brinda la inmediatamente inferior [9].

Existen varias arquitecturas que soportan otro tipo de recursos tanto de aprendizaje como recursos para otro tipo de uso. A continuación se habla brevemente sobre las arquitecturas que ofrecen soluciones parciales de la problemática que aborda el documento.

MTG-DB: A Repository for Music Audio Processing: El MTG-DB proporciona un marco para buscar, navegar y visualizar resultados. En este marco, un usuario registra una descripción compatible extractor, el extractor se ejecuta en una selección de archivos de audio para que el usuario pueda visualizar una lista de los archivos encontrados. Este repositorio cumple con las características básicas de un repositorio pero debido a los avances tecnológicos en la actualidad, los repositorios de este tipo deben contar con la capacidad de acceder a ellos a través de un dispositivo móvil, y para ello es necesario que el repositorio cuente con un módulo que identifique la plataforma desde la cual se accede al repositorio para así mostrar una interfaz adecuada. Además de esto la arquitectura no presenta una interfaz de usuario que permita la identificación del usuario.

An Infrastructure for Open-Architecture Digital Libraries: La arquitectura del repositorio biblioteca digital usa una abstracción, un objeto digital, como unidad atómica en colecciones en la infraestructura de la biblioteca digital. Utilizando una estructura de tres capas logran ofrecer almacenamiento y recuperación de archivos de video. Cabe mencionar que el repositorio cuenta con algunas deficiencias utilizando las tecnologías móviles actuales, ya que no presenta una interfaz adecuada para el dispositivo. Tampoco presenta algún módulo de servicios web el cual habilita la capacidad de interoperabilidad, lo que significa que no es posible el uso de este repositorio desde diferentes sistemas operativos. Además de no contar con algún módulo de seguridad el cual evite algún tipo de ataque al repositorio. Aunado a esto el repositorio no tiene la capacidad de visualización de los archivos de video en línea y únicamente permite la descarga de los archivos de video.

12Cnet: content-based similarity search in geographically distributed repositories of medical images: 12C es un sistema de administración de imágenes el cual ha sido desarrollado como parte de un proyecto en marcha para el diseño, implementación, y evaluación de representación de contenido de imágenes y contenido basado en estrategias similares para la recuperación. La arquitectura presenta diversas deficiencias entre ellas encontramos la imposibilidad de ser usado en un dispositivo móvil, tampoco cuenta con algún servicio web para habilitar la interoperabilidad o alguna interfaz de usuario que permita al mismo identificarse en el repositorio así como tampoco cuenta con un catalogador de los archivos que facilite el almacenaje y recuperación de los archivos. Estas son algunas de las características que este documento plantea presentar en la arquitectura propuesta.

Enabling agility through architecture: La arquitectura implementada en este repositorio habilita la posibilidad de almacenar apelaciones móviles. Para ello implementa diversos módulos en la arquitectura como: interfaz de servicio, auto publicación de aplicaciones, administración de tienda, administración de catálogos, agente de servicio y lógica de acceso de datos.

Las deficiencias encontradas en esta arquitectura son la falta de un identificador de plataforma de acceso, una interfaz de usuario para para identificación del mismo en el repositorio. Cabe mencionar que debido a la naturaleza de los archivos almacenados en este tipo de repositorio la posibilidad de visualizar dichos archivos no es posible ya que son aplicaciones, pero si es posible mostrar imágenes o videos de dichas aplicaciones por lo que permitir el almacenaje de archivos de imágenes y video se convierte en una necesidad la cual no satisface esta arquitectura.

Sistema Interactivo Distribuido de Repositorios de Objetos de Aprendizaje Matemáticos: El sistema SINDROME tiene una arquitectura base organizada en capas y basado en componentes, fundamentada en el modelo J2EE aunque incorporando WebServices como el mecanismo de acceso a componentes distribuidos. Esta es una arquitectura de un repositorio de recursos de aprendizaje, por lo que es una arquitectura importante para este documento. Es necesario notar que presenta algunas deficiencias tales como la falta de un identificador de plataforma de acceso para mostrar a una interfaz adecuada a los usuarios del repositorio desde dispositivos móviles, tampoco cuenta con la capacidad de visualizar los recursos en línea. En la arquitectura de este repositorio no se muestra algún módulo de seguridad para evitar ataques al repositorio. Y por último este repositorio de recursos de aprendizaje no permite el almacenaje de distintos tipos de archivo.

La problemática identificada en la literatura que se analizó fue que las arquitecturas implementadas en los repositorios de recursos de aprendizaje tienen la característica que solo permiten el almacenaje de documentos de texto. A esto se suma el hecho de que no permiten la visualización de los recursos en línea [10]. Dado los avances tecnológicos con los dispositivos móviles, es necesario mencionar que acceder a un repositorio de recursos de aprendizaje desde un dispositivo móvil presenta ciertas limitaciones debido al tamaño de la pantalla de dicho dispositivo [11] además de la posibilidad de algunas incompatibilidades debido a la estructura de las páginas web. Por lo que identificar la plataforma de acceso se vuelve importante para los usuarios de los repositorios de recursos de aprendizaje desde un dispositivo móvil.

3 Aportación de arquitectura ArCaOrSe

La tendencia actual en los servicios ofrecidos por los repositorios de recursos de aprendizaje va en incremento por ello la arquitectura propuesta pretende satisfacer la siguiente lista de características

- **Identificador de plataforma de acceso:** Debido a las limitaciones de tamaño de pantalla de algunos dispositivos móviles, identificar desde donde accede un usuario a los servicios del repositorio es necesario para ofrecer una interfaz correcta. También es importante señalar que se pueden generar ciertas incompatibilidades de la interfaz diseñada para usuarios de
- **Interfaz de usuario:** Para el manejo de los servicios que ofrece el repositorio de recursos de aprendizaje es necesario una interfaz que ofrezca una retroalimentación clara sobre el uso de la interfaz [12]
- **Servicios web e Interoperabilidad:** Permite al repositorio la capacidad de operar sin la necesidad de estar atado a una plataforma en específico implementando servicios web [13]
- **Servicio de búsqueda y registro de recursos:** Los repositorios de recursos de aprendizaje deben proveer servicios básicos de búsqueda y registro de recursos
- **Catalogador de archivos:** Dado que se pretende almacenar una gran cantidad de recursos, es necesario un componente de software que almacene correctamente dichos recursos. Esta tecnología se implementa comúnmente en repositorios de aplicaciones móviles [14]
- **Visualización de recursos en línea:** La capacidad de visualizar los objetos almacenados, se ha resuelto parcialmente de manera individual en los diferentes tipos de repositorios de recursos. Por ello la arquitectura pretende permitir al usuario poder visualizar los recursos en línea.
- **Soporte a múltiples tipos de archivos:** Los recursos de aprendizaje pueden ser de diferentes tipos de archivos. Estos archivos pueden ser docs, pdf, jpeg, txt, mp3, video, aplicaciones, etc.
- **Metadatos:** Al registrar o buscar archivos se utilizan claves únicas para poder identificarlos, para ello se almacenan también los metadatos que identifican los recursos

En la Figura 3 se muestra una lista de las capacidades mencionadas anteriormente como requerimientos de la arquitectura además de hacer una comparación de estos requisitos con las capacidades que ofrecen dichas arquitecturas.

En la Figura 1 se muestra la arquitectura en capas orientada a servicios diseñada para implementar las características mencionadas.

Las capas y su descripción presentadas en la figura 1 son:

- **Cliente:** Esta capa se encarga de identificar la plataforma de acceso al repositorio.
- **Interfaz:** Esta capa se encarga de mostrar la interfaz adecuada al usuario ya sea para un dispositivo móvil o para un usuario de una computadora y facilitar la administración de sesión del usuario. Además de mostrarle al usuario los resultados obtenidos de las búsquedas y habilitar la descarga de los recursos.

- **Seguridad:** Esta capa se encarga de la seguridad del repositorio para evitar ataques de inyección SQL.
- **Servicios:** Esta capa se encarga de administrar los servicios que se le ofrecen al usuario. La búsqueda de recursos de aprendizaje y el registro de nuevos recursos al repositorio en conjunto con el catalogador de archivos que se encarga de almacenar correctamente los recursos nuevos. Además de que en esta capa se encuentra el servicio web que habilita la interoperabilidad.
- **Almacenamiento:** En esta capa se encuentra la administración de los recursos almacenados. Aquí es donde se guardan todos los recursos que son registrados en el repositorio además de también almacenar los metadatos que identifican a cada uno de los recursos.
- La arquitectura propuesta es la solución para las características anteriormente mencionadas. Así de esta manera logramos integrar las capas y el conjunto de funcionalidades para que los servicios disponibles en el repositorio sean incrementados.

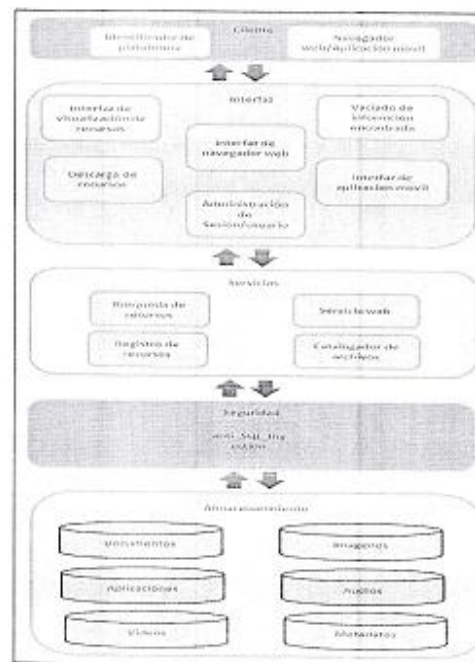


Figura 1. Arquitectura ArCaOrSe propuesta que implementa las características mencionadas

4 Implementación parcial de la arquitectura

La implementación completa de la arquitectura sigue pendiente. Pero se implementó un módulo de la capa de servicios específicamente el registro de archivos. Este módulo habilita la capacidad del repositorio de soportar múltiples tipos de archivos para que estos sean almacenados. Para lograr esto se realizó la prueba de guardar archivos en el repositorio utilizando el siguiente código en el lenguaje PHP mostrado en algoritmo 1.

ID	Nombre	Fecha de creación	Extensión	Categoría	Subcategoría	URL
1	Las matemáticas	May 2015-11-01	Matemáticas	Matemáticas	Matemáticas	archivo/DocumentoPDF1.pdf
11	Fracturas	May 2015-11-01	Fracturas	Fracturas	Fracturas	archivo/DocumentoPDF1.pdf
12	Tablas	May 2015-11-01	Tablas	Tablas	Tablas	archivo/DocumentoPDF1.pdf
12	Imágenes	May 2015-11-01	Imagenes	Imágenes	Imágenes	archivo/imagen.jpg
12	Análisis de tablas	May 2015-11-01	Matemáticas	Matemáticas	Matemáticas	archivo/DocumentoPDF1.pdf
12	Clases de la lengua	May 2015-11-01	Asignatura de Lengua	Lengua	Lengua	archivo/DocumentoPDF1.pdf
12	Clases de la lengua	May 2015-11-01	Lengua	Lengua	Lengua	archivo/DocumentoPDF1.pdf
12	Clases de la lengua	May 2015-11-01	Psicología	Psicología	Psicología	archivo/DocumentoPDF1.pdf
12	Clases de la lengua	May 2015-11-01	Imágenes	Imágenes	Imágenes	archivo/imagen.jpg

Figura 2. Resultado de la prueba de almacenar archivos en el repositorio con diferentes extensiones

Algoritmo 1. La función del código es tomar el objeto que el usuario desea guardar en el repositorio y verificar que no exista un archivo con el mismo nombre. En caso de que no exista un archivo con el mismo nombre este se guarda en una ruta determinada del servidor. Caso contrario se le indica que existe un archivo con el mismo nombre y es necesario cambiar el nombre del archivo.

```

if (!file_exists($ruta2 . $archivo))
{
    //móvenos el archivo al servidor.
    $resultado =
    @move_uploaded_file($_FILES["archivo"]["tmp_name"],
    $ruta2 . $archivo);
    echo $_FILES["archivo"]["name"] . " fue subido con
    éxito.";
} else {
    //mensaje de error de que ya existe el archivo
    echo $_FILES["archivo"]["name"] . " ", este archivo existe,
    cambie el nombre del archivo";
}

```

En la figura 2 se puede observar los resultados obtenidos al guardar los archivos en el repositorio. De esta manera de se comprueba la posibilidad de almacenar en el repositorio diferentes tipos de archivos.

Requerimientos de Arquitectura ArCaOrSe	A Repository for Music Audio Processing	An Infrastructure for Open-Architecture Digital Libraries	IL4.net	Enabling agility through architecture	SINDROME
Identificador de plataforma					
Interfaz de usuario	x	x	x	x	x
Login de usuario					x
Interoperabilidad	x			x	x
Servicio de búsqueda	x	x	x	x	x
Servicio de registro de recursos	x	x	x	x	x
Servicios web	x			x	x
Seguridad				x	
Catalogador de archivos		x		x	x
Visualización de recursos en línea	x				
Almacenaje de archivos de audio	x				
Almacenaje de archivos de vídeo		x			
Almacenaje de archivos de imágenes			x		
Almacenaje de archivos de documentos					x
Almacenaje de archivos de aplicaciones				x	
Almacenaje de archivos de metadatos	x	x	x	x	x

Figura 3. Comparación de requerimientos propuestos de la arquitectura ArCaOrSe con las arquitecturas analizadas

5 Conclusiones

La idea de la arquitectura mostrada como solución es la de ampliar los servicios ofrecidos por los repositorios de recursos de aprendizaje. De esta manera logramos que los repositorios de recursos de aprendizaje logren más interacción con el usuario y no solo sirvan como medio de almacenamiento y recuperación.

El diseño de ArCaOrSe habilita tener una serie de servicios que corresponden a la problemática abordada en el documento. Estos servicios trabajan principalmente en las funciones que requiere la arquitectura que permiten a esta la visualización de los recursos en línea, mostrar una interfaz adecuada para una PC o un dispositivo móvil, lograr la interoperabilidad del repositorio y permitir las funciones básicas como búsqueda/recuperación/registro de los recursos de aprendizaje, además la implementación parcial mostrada en el documento demuestra que es posible almacenar distintos tipos de archivos en el repositorio, logrando así que los recursos de aprendizaje almacenados no sean únicamente del tipo documento.

Por último es necesario mencionar que la arquitectura presentada en este documento no ha sido implementada completamente, pero varios de los módulos se han probado de manera individual mostrando resultados positivos. En trabajos futuros se planea mostrar los resultados generados de la implementación completa de la arquitectura.

Referencias

- [1]. ADL (2002). ADL SCORM Version 1.3 Application <http://www.imerypages.org/SCORM1.3-Seq.html?profile.pdf> accedido el 10 de enero 2015 (27 de noviembre 2002)
- [2]. Rosarigo, Z. B., Bramati, P., & Bramati, S. (2009). Objetos de Aprendizaje para la cédrua de Proyecto I. *In II Congreso de Tecnología en Educación y Educación en Tecnología.*
- [3]. Hernandez M. Definamos Contenido Educativo Digital [<http://eed-umerg.blogspot.mx/2011/06/definamos-contenido-educativo-digital.html>] accedido el 10 de febrero (25 de junio 2011)
- [4]. Marcano, B. Juegos serios y entrenamiento en la sociedad digital. *Revista Electrónica Teoría de la Educación. Educación y Cultura en la Sociedad de la Información*, 9(3), pp. 93-105. (2008).
- [5]. JORUM Team. E-learning repository systems research watch. *Project Document*. (2006).
- [6]. McGreal, R. Learning objects: A practical definition. *International Journal of Instructional Technology and Distance Learning (IJITDL)*. (2004).
- [7]. MacKenzie, C. Matthew, Ken Laskey, Francis McCabe, Peter F. Brown, Rebekah Metz, and Booz Allen Hamilton. "Reference model for service oriented architecture 1.0." *OASIS Standard 12* (2006).
- [8]. Brown, P. F., Metz, R., & Hamilton, B. A Reference model for service oriented architecture 1.0. Tech. rep. <http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.pdf> accedido el 15 de febrero 2015 (2005).
- [9]. Reynoso, C. B. Introducción a la Arquitectura de Software. *Universidad de Buenos Aires*, (2004).
- [10]. Alvarez F, Arteaga M. Avances en objetos de aprendizaje. Experiencia de redes de colaboración en México. pp. 161 (2011)
- [11]. Hernández-García, A., Iglesias-Produs, S., Chaparro-Peláez, J., & Páscual-Miguel, F. J. La Web en el móvil: tecnologías y problemática. *El profesional de la información*, 18(2), pp. 137-144 (2009).
- [12]. Muñoz Arteaga, J., Álvarez Rodríguez, F. J., Osorio Urrutia, B., & Carlona Salas, J. P. Objetos de aprendizaje integrados a un sistema de gestión de aprendizaje. *Apertura*, 6(3), pp. 109-117, (2006).
- [13]. Montilva, J., Ojuela, A., & Rojas, M. Diseño de un repositorio de objetos de aprendizaje implementado con servicios web. Publicado en: *Revista Avances en Sistemas e Informática*, Vol 7, 2. (2010)
- [14]. Brown, N., Nord, R., & Ozkaya, I. Enabling agility through architecture. CARNEGIE-MELLON UNIV PITTSBH RGHT PA SOFTWARE ENGINEERING INST. (2010).

Anexo 20 Carta aceptación a XI Conferencia latinoamericana de objetos y tecnologías de aprendizaje LACLO 2016

LACLO 2016 | XI Conferencia Latinoamericana de
COSTA RICA | Objetos y Tecnologías de Aprendizaje

July 27, 2016

Dear

Mario Alberto Luna Preciado, Francisco Javier Álvarez Rodríguez, Jaime Muñoz Arteaga, Miguel Vargas Martin

On behalf of the Eleventh Latin American Conference on Learning Objects and Technologies Organizing Committee, we are pleased to inform you that your paper "Implementación de Arquitectura ArCaOrSe de un Repositorio de Recursos Educativos y Obtención de Resultados Preliminares de Pruebas Realizadas a la Implementación (140)" has been ACCEPTED as SHORT PAPER to be presented at LACLO 2016.

LACLO this year is in its 11th edition and will be held on October 3-7, 2016, in the beautiful city of San Carlos, Costa Rica.

Please note that at least one of the authors of accepted papers should register online to the conference by 31 August, 2016. In order to do that, please visit: <http://tecdigital.tec.ac.cr/servicios/laclo2016/registro.html>.

Accepted and presented full and short papers will be indexed and published by IEEE Xplore Library. Please note that, according to the IEEE Conferences No Show Policy, we should notify you that "IEEE reserves the right to exclude a paper from distribution after the conference (e.g., removal from IEEE Xplore) if the paper is not presented at the conference". Therefore, please take into serious consideration this notification when planning your participation to the LACLO 2016 Conference.

The camera-ready version of your paper should be uploaded to the LACLO submission web site no later than 1 September, 2016. Please consider the reviewers' comments that were made to your paper when preparing the final version of your paper. It should be prepared using the templates available at http://www.ieee.org/conferences_events/conferences/publishing/templates.html. Please remember that full papers have a page limit of 12 pages, and short papers of 6 pages. The title and abstract must be in English.

Soon we will be in touch with authors to give them information regarding how to submit the IEEE Copyright Form.

We hope to see you at LACLO 2016! If any question, please contact us at laclo2016@tec.ac.cr.

Best,



Dr. Mario Chacón Rivas
General Chair
LACLO 2016

Anexo 21 Artículo LACLO 2016: Implementación de Arquitectura ArCaOrSe de un Repositorio de Recursos Educativos y Obtención de Resultados Preliminares de Pruebas Realizadas a la Implementación

Implementación de Arquitectura ArCaOrSe de un Repositorio de Recursos Educativos y Obtención de Resultados Preliminares de Pruebas Realizadas a la Implementación

Mario Alberto Luna Preciado, Francisco Javier Álvarez Rodríguez, Jaime Muñoz Arteaga.
Universidad Autónoma de Aguascalientes
 Av. Universidad #904, Ciudad de Aguascalientes CP.
 20131, Aguascalientes, Ags. México
 {marioal.luna, fjalvar.uaa, jmuuaa}@gmail.com

Miguel Vargas Martín
University of Ontario Institute of Technology
 2000 Simcoe Street North Oshawa, Ontario L1H 7K4
 Canada.
 Miguel.VargasMartin@uoit.ca

Abstract— Repositories of educational resources currently offer various services, including searching and storing learning resources, these services are considered basic in the repositories of educational resources. Universities are producing educational resources, but the reality is that usually these resources generated by students and teachers of the university are often lost. This is mainly because these resources are not be stored or retrieved in a database. These resources can be useful to new generations of students. This document aims to show the results of architecture implementation shown in previous work as well as the implementation and results of tests that validate the services offered by the architecture.

Keywords— *Layer-based architecture, learning resources, Repositories, service oriented architecture, software architectures, search of resources, storage of resources, testing repositories, test results*

I. INTRODUCCIÓN

La gestión de contenidos en los entornos de aprendizaje, está generando cambios en sentidos diversos. Entre estos cambios, los contenidos educativos se están construyendo como recursos educativos y se están recopilando en contenedores, conocidos como repositorios de recursos educativos, que los organizan y los hacen accesibles para diferentes aplicaciones y perfiles de usuarios [1].

Los repositorios en la actualidad ofrecen servicios básicos como: búsquedas y registros de recursos educativos. En la actualidad nos encontramos con repositorios que almacenan objetos de aprendizaje (OA) además de los recursos de aprendizaje educativos abiertos (REA), estos tipos de recursos educativos se almacenan en sus propios repositorios de recursos. Es decir contamos con repositorios de recursos de aprendizaje y repositorios de recursos educativos abiertos [2] [3].

En la actualidad existen diversos tipos de repositorios y estos almacenan diversos tipos de archivos. Siendo estos últimos archivos de audio, imágenes, video, aplicaciones

ejecutables o aplicaciones móviles y documentos de texto, además de ofrecer la capacidad de visualizar los recursos educativos en línea, es decir reproducción de videos y audios, así como visualizar documentos e imágenes en el mismo repositorio.

Los recursos de aprendizaje no solo se limitan a archivos de documentos de texto, también existen los recursos de aprendizaje en formatos tales como: Audio, Video, Imagen y juegos serios [4].

El trabajo detrás de este documento es generar una arquitectura en capas orientada a servicios para un repositorio de recursos de aprendizaje, que permita el almacenaje de los recursos generados en las universidades por los estudiantes y profesores para que estos puedan servir para generaciones futuras de estudiantes. Además de esto, la arquitectura aquí planteada, pretende ofrecer la capacidad de almacenaje de los recursos educativos en diferentes tipos de formatos como: archivos de aplicaciones educativas, audio, documentos, imágenes y video.

El repositorio de recursos educativo con las capacidades anteriormente mencionadas, se encuentra en la actualidad en desarrollo en la universidad Autónoma de Aguascalientes. En este documento se pretende mostrar la problemática que se ha trabajado, así como presentar las pruebas y los resultados parciales obtenidos al realizar dichas pruebas en el repositorio de recursos educativos de la universidad Autónoma de Aguascalientes.

II. ARQUITECTURA ORIENTADA A SERVICIOS Y ORIENTADA EN CAPAS

La Arquitectura Orientada a Servicios (SOA, siglas del inglés Service Oriented Architecture) es un paradigma para organizar y utilizar capacidades distribuidas que pueden estar bajo el control de dominios [5] [6].

La arquitectura basada en capas se enfoca en la distribución de roles y responsabilidades de forma jerárquica proveyendo

una forma muy efectiva de separación de responsabilidades. Garland y Shaw definen el estilo en capas como una organización jerárquica tal que cada capa proporciona servicios a la capa inmediatamente superior y se sirve de las prestaciones que le brinda la inmediatamente inferior [7].

La arquitectura planteada en este documento se desarrolló en bases a los requerimientos y lista de capacidades mostradas en la figura 1.

Dicha arquitectura permite ofrecer las capacidades mencionadas en el apartado anterior atendiendo las adecuaciones de los estilos de arquitectura orientado a servicio y basada en capas.

Requerimientos de Arquitectura ArCaXSe	A Repository for Music Audio Processing	An Infrastructure for Open-Access to Digital Libraries	PLand	Enabling ability through architecture	SINDROME
Identificador de plataforma					
Interfaz de usuario	x	x	x	x	x
Login de usuario					x
Interoperabilidad	x			x	x
Servicio de búsqueda	x	x	x	x	x
Servicio de registro de recursos	x	x	x	x	x
Servicio web	x			x	x
Seguridad				x	
Catálogo de archivos		x		x	x
Visualización de recursos en línea	x				
Almacenaje de archivos de audio	x				
Almacenaje de archivos de video		x			
Almacenaje de archivos de imágenes			x		
Almacenaje de archivos de documentos					x
Almacenaje de archivos de aplicaciones				x	
Almacenaje de archivos de metadatos	x	x	x	x	x

Figura 1. Comparación de requerimientos propuestos de la arquitectura ArCaXSe con las arquitecturas analizadas [10].

La arquitectura propuesta para el repositorio de recursos educativos se presenta en la figura 2.

La problemática identificada en la literatura que se analizó fue que las arquitecturas implementadas en los repositorios de recursos de aprendizaje tienen la característica que solo permiten el almacenaje de documentos de texto. A esto se suma el hecho de que no permiten la visualización de los recursos en línea [8]. Dado los avances tecnológicos con los dispositivos móviles, es necesario mencionar que acceder a un repositorio de recursos de aprendizaje desde un dispositivo móvil presenta ciertas limitaciones debido al tamaño de la pantalla de dicho dispositivo [9] además de la posibilidad de algunas incompatibilidades debido a la estructura de las páginas web. Por lo que identificar la plataforma de acceso se vuelve importante para los usuarios de los repositorios de recursos de aprendizaje desde un dispositivo móvil.

III. DEFINICIÓN DE PRUEBAS PARA LA ARQUITECTURA

La variedad de servicios ofrecidos por el repositorio de recursos educativos necesitan ser validados, es decir para demostrar que dichos servicios pueden ser puestos en la práctica. Para ello se implementa una serie de pruebas para cada uno de los servicios ofrecidos por el repositorio de recursos educativos. Cada prueba implementada tendrá algún objetivo en específico que ayudara a validar la funcionalidad del servicio.

Las pruebas que se implementaran, abarcan los siguientes servicios:

- Identificador de plataforma
- Búsqueda de recursos educativos en el repositorio
- Registro de recursos educativos en el repositorio
- Prueba de descarga de recursos educativos del repositorio
- Prueba de visualización para cada uno de los tipos de recursos almacenados en el repositorio (imágenes, audio, video y documentos)

Como se mencionó, el objetivo de cada una de las pruebas mencionadas es para validar cada uno de los servicios que ofrece la arquitectura. A continuación se definirá a mayor detalle cada una de las pruebas. Para la aplicación de las pruebas primero es necesario definir las condiciones y los escenarios en los que aplicaran dichas pruebas.

Una de las condiciones para ejecutar las pruebas es definir los dispositivos y computadoras que se usaran, estos dispositivos están definidos en la tabla 1.

Otro elemento importante a definir para la ejecución de las pruebas son los escenarios de pruebas. Para lograr esto se define que los escenarios de pruebas se conformaran de tres etapas:

Tabla 1. Lista de dispositivos para la ejecución de las pruebas

Lista de dispositivos y sus características			
Nombre	Sistema Operativo	Navegador web	Cantidad de memoria RAM
Galaxy Tab S2	Android 5.02	Google Chrome Build 49.0.2623.105	2 GB
Galaxy Tab 4	Android 5	Google Chrome Build 49.0.2623.105	2 GB
Nokia Lumia 920	Windows phone 8.1	Internet explorer 11	1 GB
Laptop Lenovo Y510P	Windows 10	Google Chrome Build 49.0.2623.110 m	8 GB
Pc escritorio	Windows 10	Google Chrome Build 49.0.2623.110 m	16 B

- **Antes:** En esta parte del escenario de prueba, se definen las condiciones y elementos necesarios para la ejecución de la prueba (Servicios web, interfaces, dispositivos, etc.).
- **Durante:** en esta parte del escenario de prueba, se definen los pasos necesarios para ejecutar la prueba.
- **Después:** En esta parte del escenario de prueba, se define el formato para registrar los resultados de la prueba ejecutada.

A. Prueba identificador de plataforma

Uno de los servicios de la arquitectura consiste en que la aplicación web del repositorio pueda identificar correctamente el dispositivo desde el que se está accediendo a la aplicación. Logrando identificar el dispositivo, podemos mostrar una interfaz correcta para dicho dispositivo.

Esta prueba permite validar la funcionalidad del servicio de identificador de cliente, es decir identificar si el usuario del repositorio, accede a la aplicación desde un dispositivo móvil, una Tablet, o una computadora de escritorio. Esto con la finalidad de mostrar una interfaz de tamaño adecuada para el usuario.

Los elementos del escenario de prueba incluyen:

- Lista de dispositivos a implementar
- Servicios web
- Diseño y desarrollo de interfaces web
- Formato para registro de resultado de pruebas

Los resultados de la prueba fueron satisfactorios con los dispositivos probados, esto quiere decir que los dispositivos mencionados anteriormente, mostraron una interfaz adecuada para la plataforma que accedió a la aplicación web del repositorio. Cabe mencionar que el identificar el tipo de plataforma que accede a la aplicación web del repositorio depende completamente del código implementado para la misma detección.

B. Prueba búsqueda de recursos

El servicio de búsqueda de recursos es uno de los elementos clave de cualquier repositorio [11] [12], por lo que es una de las pruebas que se pretenden presentar en este artículo.

Esta prueba permite validar la funcionalidad del servicio de búsqueda de recursos, la prueba no está diseñada para medir la precisión ni la exactitud de la búsqueda de recursos. La finalidad de la prueba es verificar que se encuentre algún recurso relacionado con los términos definidos para realizar la búsqueda.

Los elementos del escenario de prueba incluyen:

- Lista de dispositivos a implementar
- Servicios web
- Diseño y desarrollo de interfaces web
- Formato para registro de resultado de pruebas

El servicio de búsqueda implementado en el repositorio, permite al usuario definir el método de búsqueda por tres tipos (Ver figura 3):

- Autor del recurso
- Descripción del recurso
- Palabras clave del recurso

Los resultados obtenidos de esta prueba, validan que la búsqueda recursos educativos recupera por lo menos un recurso relacionado con los términos de búsqueda del usuario. Esto nos asegura que cualquier búsqueda realizada devuelve resultados positivos al usuario.

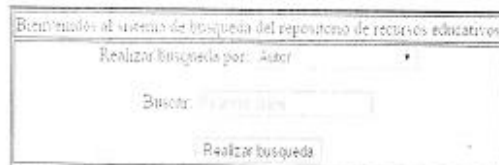


Figura 3. Interfaz de búsqueda del repositorio de recursos educativos.

C. Prueba registro de recursos educativos

El servicio de registro de recursos educativos, es otro de los servicios básicos que debe ofrecer todo repositorio de recursos educativos [11] [12], por lo que también se presentan el objetivo de la prueba y los resultados de la misma.

Los elementos del escenario de prueba incluyen:

- Lista de dispositivos a implementar
- Servicios web
- Diseño y desarrollo de interfaces web
- Listado de metadato a almacenar
- Formato para registro de resultado de pruebas

La interfaz para el registro de recursos, se mantiene simple ver figura 4. Dado que los recursos educativos que se almacenan no requieren una estructura de metadatos compleja, se optó en almacenar los siguientes metadatos:

The screenshot shows a web form titled 'Formulario de registro de recursos educativos. Por favor completar por favor llenar los campos obligatorios'. The form contains the following fields: 'Título del recurso', 'Descripción del recurso', 'Tipo de recurso' (with a dropdown menu showing 'Aplicaciones'), 'Autor', 'Tema de cobertura', 'Formato del recurso', and 'Palabras clave'. There is a 'Enviar' button at the bottom right.

Figura 4. Interfaz de registro de recursos en el repositorio de recursos educativos.

- Autor del recurso
- Título del recurso
- Descripción del recurso
- Tema de cobertura
- Formato del recurso
- Palabras clave

Se realizó una prueba con cada tipo de recurso, almacenando el mismo recurso en las 10 iteraciones y cambiando de recurso en cada dispositivo, para verificar la fidelidad del almacenaje. (Aplicaciones móviles, Audio, documentos, imágenes y video) Los resultados obtenidos de la prueba, validan la capacidad de almacenamiento del recurso en el repositorio, así como también almacenar los metadatos del mismo.

IV. RESULTADOS OBTENIDOS DE LAS PRUEBAS

Las pruebas realizadas a la arquitectura del repositorio de recursos educativos implementada en el servidor de la Universidad Autónoma de Aguascalientes, tiene como finalidad validar el cumplimiento de las características de dicha arquitectura.

La prueba de identificador de plataforma como se mencionó, tiene como finalidad identificar la plataforma de acceso a la aplicación web del repositorio de recursos educativos. Esto pretende que después de identificar la plataforma, se muestre una interfaz correcta para la plataforma de acceso. Con esto se logra que la aplicación del repositorio pueda ser usada desde cualquier plataforma sin los inconvenientes derivados del tamaño de pantalla del dispositivo. Los resultados obtenidos de la prueba de identificador de plataforma se muestran en la tabla 2.

Cabe señalar que los dispositivos con los que se trabajó, si bien es reducido el número de dispositivos, es posible incrementar dicho número para validar que diferentes marcas también puedan ser identificadas.

Los resultados obtenidos de la prueba de búsqueda de recursos (Ver tabla 2), demuestra la fidelidad obtenida de las búsquedas al utilizar los mismos criterios de búsqueda. Actualmente el repositorio no cuenta con una gran cantidad de recursos almacenados, lo que nos da un número reducido de recursos devueltos. Esto valida que el servicio de búsqueda implementado como característica de la arquitectura funciona.

Por último los resultados obtenidos de la prueba registro de recursos. Como se mencionó, por cada dispositivo se realizó diez iteraciones de registro de un mismo recurso, cambiando de recurso cuando se cambia de dispositivo. La finalidad de esto, es validar la finalidad del registro del recurso y de los metadatos del mismo. Los resultados obtenidos muestran que la característica implementada en la arquitectura, permiten que los usuarios (alumnos y profesores) puedan subir los recursos al repositorio.

La idea de la arquitectura mostrada, es facilitar a los alumnos y profesores los servicios de un repositorio de recursos educativos. Esto quiere decir que los recursos educativos generados por el mismo alumnado y profesores, pueden ser almacenados en el repositorio. Esto sin la necesidad de pasar por todo un proceso para convertir dichos recursos en recursos educativos abiertos u objetos de aprendizaje. Esta arquitectura, se implementó en el servidor de universidad autónoma de Aguascalientes.

Tabla 2. Resultados de las pruebas realizadas a la arquitectura implementada.

Nombre	Lista de dispositivos e resultados de pruebas						
	Identificador de plataforma	Búsqueda de recursos			Registro de recursos		
Dispositivo	Identificado correctamente	Palabras clave utilizadas	Cantidad de recursos devueltos	Búsqueda por	Contenido del recurso registrado	Formato del recurso	Metadatos almacenados correctamente
Galaxy Tab 2	SI	Dispositivo móvil	1 (Aplicaciones)	3	Descripción	Video en HTML5	Si
Galaxy S3	SI	Dispositivo móvil	1 (Audio)	4	Audio	Aplicación móvil	SI
Smart TV	SI	Dispositivo móvil	Aplicaciones	3	Palabras clave	Aplicaciones multimediales con dominios multimedia	SI
Laptop Linux	SI	Computadora de escritorio	Aplicaciones	4	Descripción	Aplicaciones y otros formatos multimedia	SI
PC Windows	SI	Computadora de escritorio	Aplicaciones multimedia	3	Descripción	Aplicaciones multimediales con dominios multimedia	SI

Por conclusión general, se define que los resultados obtenidos de las pruebas demuestran que las características implementadas funcionan en este repositorio. Es importante

señalar que estos resultados son preliminares, por lo que aún falta realizar las pruebas finales a una escala mayor, pero que de cierta manera, se prueba que dichos servicios funcionan. En trabajos futuros, se pretende mostrar los resultados obtenidos con los usuarios del repositorio.

REFERENCIAS

- [1]. Vidal Lado, M. J., Alfonso Sánchez, I., Zaca González, G., & Martínez Hernández, G. (2013). Recursos educativos abiertos. *Educación Médica Superior*, 27(3), 307-320.
- [2]. Aretio, G. (2005). Objetos de aprendizaje. Características y repositorios.
- [3]. Hernández M. Definimos Contenido Educativo Digital [<http://ced-userg.blogspot.mx/2011/06/definimos-contenido-educativo-digital.html>] accedido el 10 de febrero (25 de junio 2011).
- [4]. Mareano, B. Juegos serios y entrenamiento en la sociedad digital. *Revista Electrónica Teoría de la Educación: Educación y Cultura en la Sociedad de la Información*, 9(3), pp. 93-105. (2008).
- [5]. MacKenzie, C. Matthew, Ken Laskey, Francis McCabe, Peter F. Brown, Rebekah Metz, and Booz Allen Hamilton. "Reference model for service oriented architecture 1.0." OASIS Standard 12 (2006).
- [6]. Brown, P. F., Metz, R., & Hamilton, B. A Reference model for service oriented architecture 1.0. Tech. rep. <http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.pdf> accedido el 15 de febrero 2015 (2005).
- [7]. Reynoso, C. B. *Introducción a la Arquitectura de Software*, Universidad de Buenos Aires, (2004).
- [8]. Álvarez F, Arteaga M. Avances en objetos de aprendizaje. Experiencia de redes de colaboración en México, pp. 161 (2011).
- [9]. Hernández-García, Á., Iglesias-Pradas, S., Chaparro-Peláez, J., & Pascual-Miguel, F. J. La Web en el móvil: tecnologías y problemática. *El profesional de la información*, 18(2), pp. 137-144 (2009).
- [10]. Luna M., Álvarez F., Muñoz J. *Una Arquitectura en Capas Orientada a Servicios para el Desarrollo de un Repositorio de Recursos de Aprendizaje ArCaOrSe* (2015)
- [11]. ADL (2002). ADL SCORM Version 1.3: Application <http://adl.sourceforge.org/SCORM/1.3-ScgAppProfile.pdf> accedido el 10 de enero 2015 (27 de noviembre 2002).
- [12]. Rosánigo, Z. B., Bramati, P., & Bramati, S. (2009). Objetos de Aprendizaje para la cátedra de Proyecto I. *In II Congreso de Tecnología en Educación y Educación en Tecnología*.