



SEP

SECRETARÍA DE
EDUCACIÓN PÚBLICA



TECNOLÓGICO
NACIONAL DE MÉXICO

Tecnológico Nacional de México

Centro Nacional de Investigación
y Desarrollo Tecnológico

Tesis de Maestría

Sistema integrador de componentes de gestión de servicios web de
aprendizaje

presentada por

Ing. Gibrán Gutiérrez Juárez

como requisito para la obtención del grado de
Maestro en Ciencias de la Computación

Director de tesis

Dra. Olivia Graciela Fragoso Díaz

Cuernavaca, Morelos, México. Junio de 2018.



TECNOLÓGICO NACIONAL DE MÉXICO

Centro Nacional de Investigación
y Desarrollo Tecnológico

Cuernavaca, Morelos a 14 de junio del 2018
OFICIO No. DCC/186/2018

Asunto: Aceptación de documento de tesis

DR. GERARDO V. GUERRERO RAMÍREZ
SUBDIRECTOR ACADÉMICO
PRESENTE

Por este conducto, los integrantes de Comité Tutorial del **Ing. Gibrán Gutiérrez Juárez**, con número de control M16CE006, de la Maestría en Ciencias de la Computación, le informamos que hemos revisado el trabajo de tesis profesional titulado "**Sistema integrador de componentes de gestión de servicios web de aprendizaje**" y hemos encontrado que se han realizado todas las correcciones y observaciones que se le indicaron, por lo que hemos acordado aceptar el documento de tesis y le solicitamos la autorización de impresión definitiva.

DIRECTOR DE TESIS

Dra. Olivia Graciela Fragoso Díaz
Doctora en Ciencias de la Computación
7420199

REVISOR 1

M.C. Humberto Hernández García
Maestro en Ciencias con
Especialidad
en Sistemas Computacionales
7573641

REVISOR 2

Dr. René Santaolaya Salgado
Doctor en Ciencias de la
Computación
4454821

C.p. M.T.I. María Elena Gómez Torres - Jefa del Departamento de Servicios Escolares.
Estudiante
Expediente

NACS/Imz

cenidet
Centro Nacional de Investigación
y Desarrollo Tecnológico

Interior Internado Palmira S/N, Col. Palmira, C. P. 62490, Cuernavaca, Morelos.
Tels. (01) 777 3 62 77 70, ext. 4106, e-mail: dir_cenidet@tecnm.mx
www.cenidet.edu.mx



Cuernavaca, Mor., 19 de junio de 2018
OFICIO No. SAC/273/2018

Asunto: Autorización de impresión de tesis

ING. GIBRÁN GUTIÉRREZ JUÁREZ
CANDIDATO AL GRADO DE MAESTRO EN CIENCIAS
DE LA COMPUTACIÓN
PRESENTE

Por este conducto, tengo el agrado de comunicarle que el Comité Tutorial asignado a su trabajo de tesis titulado "**Sistema integrador de componentes de gestión de servicios web de aprendizaje**", ha informado a esta Subdirección Académica, que están de acuerdo con el trabajo presentado. Por lo anterior, se le autoriza a que proceda con la impresión definitiva de su trabajo de tesis.

Esperando que el logro del mismo sea acorde con sus aspiraciones profesionales, reciba un cordial saludo.

ATENTAMENTE
EXCELENCIA EN EDUCACIÓN TECNOLÓGICA®
"CONOCIMIENTO Y TECNOLOGÍA AL SERVICIO DE MÉXICO"



DR. GERARDO VICENTE GUERRERO RAMÍREZ
SUBDIRECTOR ACADÉMICO



SEP TecNM
CENTRO NACIONAL
DE INVESTIGACIÓN
Y DESARROLLO
TECNOLÓGICO
SUBDIRECCIÓN
ACADÉMICA

C.p. M.T.I. María Elena Gómez Torres.- Jefa del Departamento de Servicios Escolares.
Expediente

GVGR/mcr

DEDICATORIA

El presente trabajo es dedicado con mi amor y cariño a:

A **DIOS**, no hay razón en mi vida más grande que El, **SEÑOR** sé que lee esto, gracias por cumplir este deseo de mi corazón, siempre estaré en deuda contigo.

A mi padre **Moisés**, papá, gracias por su amor, por creer en mí y por decirme que llegaré lejos, eres un ejemplo para mí.

A mi madre **Maribel**, tengo que ser conciso, pero sabes que te amo con todo mi corazón, un día te prometí que haría una maestría, tu lo has hecho posible, te debo tanto que siento que en mi vida te podré pagar todo.

A mi hermano **Adolfo**, gracias por ser mi mejor amigo, por darme tu apoyo y amor, creo que siempre necesitaré de ti.

A mi hermana **Zurisaday**, hija, gracias por estar conmigo y apoyarme y ser mi pequeña pero gran amiga, te amo mucho.

A mi novia **Migdal**, mi amor, llegaste a mi vida en el momento indicado, tengo tanto que agradecerte que resumo todo este sentir en una expresión, te amo.

A mis amigos **Mena, Roberto, Mariela, Lorelí**, gracias por hacer de este tiempo algo agradable al estar con personas que valen mucho, que me hacen ser mejor, siempre aprendo de ustedes.

AGRADECIMIENTOS

El presente agradecimiento a:

A la Dra. **Olivia Graciela Fragoso Díaz**, más que una investigadora la considero una amiga, va más allá de su profesión, pocas personas marcan mi vida, usted es una de ellas.

Al **Consejo Nacional de Ciencia y Tecnología** (CONACYT), gracias por facilitarme los medios para cumplir una meta personal.

Al **Centro Nacional de Investigación y Desarrollo Tecnológico** (CENIDET), gracias por abrirme las puertas y capacitarme con dedicación y profesionalismo.

Al Dr. **René Santaolaya Salgado**, gracias por darme un poco de su conocimiento y experiencia, por su franca amistad y su atención.

Al MC. **Humberto Hernández García**, gracias por ayudarme en mi formación al hacer su labor con dedicación y profesionalismo.

A las áreas de **gestión de servicios escolares y ciencias computacionales**, gracias por facilitar la gestión de mis trámites para poder titularme.

Resumen

El trabajo que se describe en este documento propone un sistema de integración de componentes de gestión de Servicios Web de aprendizaje. Los componentes originales se encuentran de manera independiente. Sin embargo, se requiere que se puedan utilizar de tal manera que sea transparente para el usuario la navegación de una hacia la otra y viceversa. Asimismo, que permita agregar nuevos componentes de manera fácil para poder ampliar su utilización en conjunto y su despliegue independiente.

La solución de este trabajo comprende la definición del sistema integrador de componentes con una arquitectura basada en servicios web. La característica principal del sistema es la capacidad de integración de los componentes. La forma de construir el sistema integrador es a través de una arquitectura basada en servicios web y a través de una metodología que extrae los pasos más importantes de las formas de integración en sistemas basados en servicios web según los trabajos relacionados.

El sistema integrador tiene dos componentes principales. El primero consiste en un modelo de calidad para medir la reusabilidad de los servicios web de aprendizaje, dicho componente cuenta con una arquitectura basada en servicios web, por lo que sólo se procede a integrar. El segundo componente es un sistema de evaluaciones de diferentes dominios para calificar el aprendizaje en estudiantes. Tal componente fue reestructurado para que pueda atender la arquitectura que se plantea en el presente trabajo.

Para probar la integración y el correcto funcionamiento de los componentes integrados, se llevaron a cabo pruebas de carácter funcional donde se probó cada componente evaluado de acuerdo a datos de entrada y salida obteniendo un porcentaje de acuerdo a los criterios establecidos como exitoso y no exitoso. Al obtener el resultado de las pruebas del sistema completo se demuestra que el sistema obtenido en el trabajo presente, cumple con los objetivos y alcances planteados y presenta una solución de problema antes descrito.

La principal contribución de este trabajo es un sistema capaz de integrar componentes dedicados a la gestión de Servicios Web de Aprendizaje con una arquitectura extensible para integrar tantos componentes como sean requeridos.

Abstract

The work described in this document proposes a system for integrating learning Web Services management components. The original components are found independently. However, it is required that they can be used in such a way that it is transparent for the user to navigate from one to the other and vice versa. Also, that allows to add new components in an easy way to expand its use as a whole and its independent deployment.

The solution of this work includes the definition of the component integrator system with an architecture based on web services. The main characteristic of the system is the integration capacity of the components. The way the integrating system was built is through an architecture based on web services and through a methodology that extracts the most important steps of the forms of integration in systems based on web services according to the related works.

The integrating system has two main components. The first one consists of a quality model to measure the reusability of learning web services, this component has an architecture based on web services, so it is only necessary to integrate it. The second component is a system of evaluations of different domains to qualify student learning. This component was restructured so that it can address the architecture proposed in this work.

To test the integration and proper functioning of the integrated components, functional tests were carried out where each evaluated component was tested according to input and output data, obtaining a success percentage according to the criteria established as successful and not successful. When obtaining the results of the tests of the complete system it is shown that the system obtained in the present work, fulfills the objectives and scope proposed and presents a problem solution described above.

The main contribution of this work is a system capable of integrating components dedicated to the management of Learning Web Services with an extensible architecture to integrate as many components as required

Contenido

Capítulo 1. Introducción	2
1.1 Introducción	2
1.2 Descripción del problema	2
1.3 Objetivos	2
1.3.1 Objetivo general	2
1.3.2 Objetivos específicos	2
1.4 Justificación	3
1.5 Alcances y limitaciones	3
1.5.1 Alcances	3
1.5.2 Limitaciones	3
1.6 Beneficios	3
1.7 Organización del documento	3
Capítulo 2. Marco teórico y conceptual	6
2.1 E-Learning (aprendizaje electrónico), por Roberto Baelo Álvarez	6
2.2 Arquitecturas orientadas a servicios	6
2.3 Servicios web, por Carlos Andrés Morales	6
2.4 Servicio web de aprendizaje	7
2.5 Componente, según Pressman	8
2.6 Recurso, definición del 2016	8
2.7 Integración, según Hamdan (2015)	8
2.8 Cliente, según Snell (2002)	9
Capítulo 3. Antecedentes y trabajos relacionados	11
3.1 Antecedentes	11
3.1.1 composición de servicios WEB [7]	11
3.1.2 Extensión al sistema de composición de servicios web utilizando diagramas de actividad [8]	11
3.2 Trabajos relacionados	12
3.2.1 Trabajos relacionados	12
3.2.1.1 Integration of web services with e-learning for knowledge [9]	12
3.2.1.2 Software architecture for integration of institutional and social learning environments [10]	12
3.2.1.3 Analysis of knowledge management and e-learning integration models [11]	13
3.2.1.4 A Web Services-enabled B2B Integration Approach for SMEs [12]	14

3.2.1.5 Improving the dependability of Web Services Integration [13]	15
3.2.1.6 A Cloud Framework for Enterprise Application Integration on SOA [14]	16
3.2.1.7 Web-Service Interface-based IMS-SPSC Integration. [15]	17
3.2.1.8 Web Service Integration within Next Generation CEMIS [16]	18
3.2.2 Plataformas relacionadas.....	20
3.2.2.1 Edu next [17]	20
3.2.2.2 Spot LMS [18]	21
3.2.2.3 CAE innovative learning solutions [19]	21
3.2.2.4 Paradiso [20]	22
Capítulo 4. Metodología de solución	24
4. 1 Analizar requerimientos actuales del componente.....	24
4.1.1 Requerimientos funcionales del sistema de evaluación (Martínez, J. y Gutiérrez, G.,2015)	25
4.2 Reestructurar requerimientos para una arquitectura nueva.....	25
4. 3. Analizar arquitectura actual	34
4.3.1 Diseñar arquitectura general a partir de la arquitectura Actual	35
4.3.2 Arquitectura planteada con servicios web	35
4.3.3 Arquitectura orientada a servicios implementada con java	36
4.5 Implementacion de requerimientos y arquitectura actualizados.....	37
4.5.1 Identificar entidades en los requerimientos	38
4.5.2 Establecer funcionalidades por entidades	39
4.5.3 Implementar funcionalidades por entidades.....	39
4.5.4 Agregar capa facade de web service por entidad	39
Capítulo 5 Pruebas	47
5.1.1 Introducción.....	47
5.1.2 Objetivo de las pruebas	47
5.1.3 Procedimiento de pruebas.....	47
5.1.4 Descripción de los elementos de prueba.	48
5.2. Descripción de funcionalidad del componente.....	49
5.2.1 Localización física de componentes del sistema integrador.....	49
5.3. Descripción del cliente, interfaces.	51
5.3.1 Portal.....	51
5.3.2 Clientes del sistema de evaluación.	52

5.3.2.1 Clientes del componente del modelo de calidad para servicios web de aprendizaje	52
5.3.2.2 Cliente LWS	52
5.3.2.3 Cliente LWS2	53
5.3.3 Clientes de Sistema de evaluaciones.....	53
5.3.3.1 Conexión a base de datos	54
5.3.3.2 Login	54
5.3.3.3 Manejador de recursos.....	55
5.3.3.4 Panel de administrador	55
5.3.3.5 Gestión de dominios.....	56
5.3.3.6 Gestión de subdominios.....	57
5.3.3.7 Gestión de recursos.....	57
5.3.3.8 Gestión de evaluaciones.....	58
5.3.3.9 Modal para crear dominio.	59
5.3.3.10 Modal para editar dominio.....	59
5.3.3.11 Modal para crear sub dominio.	59
5.3.3.12 Modal para crear sub dominio.	60
5.3.3.13 Modal para dar de baja dominio.....	60
5.3.3.14 Modal para editar sub dominio.	60
5.3.3.15 Modal para dar de baja un sub dominio.	61
5.3.3.16 Modal para crear recurso.	61
5.3.3.17 Modal para editar recurso.	61
5.3.3.18 Modal para dar de baja recurso.....	62
5.4. Ejecución de las pruebas	62
5.4.1 pruebas al componente de calidad de servicios web de aprendizaje	62
5.4.1.1 Pruebas realizadas	63
5.4.2 pruebas al componente de sistema de evaluaciones.	68
5.5 Análisis de pruebas	78
5.5.1 Análisis de pruebas del Componente del modelo de calidad de SWA.....	78
5.5.2 Análisis de pruebas del Componente del sistema de evaluaciones.....	80
5.6 Conclusiones.....	86
5.6.1 Conclusión de análisis de componente de modelo de calidad de servicios web de aprendizaje.....	86
5.6.2 Conclusión del componente del sistema de evaluaciones.	87
Capítulo 6 Conclusiones	90

6.1 Conclusiones	90
6.2 Trabajos futuros	92
Anexo A WSDL de los componentes del Sistema de evaluación Gregorio, J. and Gutiérrez, G. (2015)	93
Referencias	94

Índice de figuras

Figura 2. 1 Infraestructura de servicio web	7
Figura 2. 2 Modelo general de un servicio Web de Aprendizaje	8
Figura 2. 3 Representación de un cliente web	9
Figura 3. 1 Arquitectura planteada para la integración de componentes en el artículo Software architecture for integration of institutional and social learning environments	13
Figura 3. 2 Representación de Web Services-enabled B2B	15
Figura 3. 3 Marco Java WS-mediator.....	16
Figura 3. 4 Marco funcional de la arquitectura en la nube	17
Figura 3. 5 Diseño de arquitectura para IMS-SPSC basada en servicios web	17
Figura 3. 6 Diagrama de clases del sistema IMS-SPSC	18
Figura 3. 7 Arquitectura Next Generation CEMIS	19
Figura 4. 1 Metodología general de solución	24
Figura 4. 2 RF1, primera versión.....	26
Figura 4. 3 RF1, versión actual.....	27
Figura 4. 4 RF2, primera versión.....	27
Figura 4. 5 RF2, versión final.....	27
Figura 4. 6 RF3, primera versión.....	28
Figura 4. 7 RF3, Versión final	28
Figura 4. 8 RF4, primera versión.....	29
Figura 4. 9 RF4, versión final	29
Figura 4. 10 RF5, primera versión.....	30
Figura 4. 11 RF5, versión Final	30
Figura 4. 12 RF6, versión considerada	31
Figura 4. 13 RF7, primera versión.....	31
Figura 4. 14 RF7, versión final.....	31
Figura 4. 15 RF8, versión considerada	32
Figura 4. 16 RF9, primera versión.....	32
Figura 4. 17 RF9, versión final	33
Figura 4. 18 RF10, primera versión.....	33
Figura 4. 19 RF10, versión final	33
Figura 4. 20 Funcionamiento básico de un servicio web	35
Figura 4. 21 Diagrama de elementos de un servicio web hecho en java	37
Figura 4. 22 Subproceso del proceso 5.....	38

Figura 4. 23 Diagrama de componentes de acuerdo a funcionalidades por entidades	38
Figura 4. 24 Arquitectura general del componente: Sistema de evaluación	40
Figura 4. 25 Arquitectura en diagrama de clases del sub componente usuario	41
Figura 4. 26 Arquitectura en diagrama de clases del sub componente dominio	42
Figura 4. 27 Arquitectura en diagrama de clases del sub componente sub dominio.....	43
Figura 4. 28 Arquitectura en diagrama de clases del sub componente recurso	44
Figura 4. 29 Arquitectura en diagrama de clases del sub componente Evaluación.....	45
Figura 5. 1 Pasos para el desarrollo y análisis de las pruebas del sistema integrador	47
Figura 5. 2 Diagrama despliegue del sistema integrador de componentes.....	50
Figura 5. 3 Vista del portal del sistema integrador de componentes.....	52
Figura 5. 4 Interfaz del cliente LWS	53
Figura 5. 5 Interfaz del cliente LWS2	53
Figura 5. 6 Interfaz de cliente de conexión a base de datos	54
Figura 5. 7 Interfaz de cliente de ingreso de usuario.....	55
Figura 5. 8 Interfaz de gestión de dominios, recursos y evaluaciones	55
Figura 5. 9 Interfaz de cliente de administrador de usuarios.....	56
Figura 5. 10 Interfaz de cliente de gestión de dominios.....	56
Figura 5. 11 Interfaz de cliente de gestión de sub dominios	57
Figura 5. 12 Interfaz de cliente de gestión de recursos	58
Figura 5. 13 Interfaz de cliente de gestión de evaluaciones	58
Figura 5. 14 Ventana modal de creación de dominios	59
Figura 5. 15 Ventana modal de actualización de dominios.....	59
Figura 5. 16 Ventana modal de creación de sub dominios.....	60
Figura 5. 17 Ventana modal para dar de baja un dominio.....	60
Figura 5. 18 Ventana modal de actualización de sub dominio	61
Figura 5. 19 Ventana modal de sub dominio.....	61
Figura 5. 20 Ventana modal de creación de recurso multiple choice.....	61
Figura 5. 21 Ventana modal de actualización de recurso	62
Figura 5. 22 Ventana modal para dar de baja un recurso	62
Figura 5. 23 Interfaz del cliente para consumir el componente webServiceSoap2.....	65
Figura 5. 24 Resultados de datos de salida del caso de prueba Comp2Sub2Docx2.....	66
Figura 5. 25 Gráfico de resultados de pruebas del componente webServiceSoap	79
Figura 5. 26 Gráfico de resultados de pruebas del componente webServiceSoap2	80
Figura 5. 27 Gráfico de los porcentajes obtenidos de wsUser.....	81
Figura 5. 28 Gráfico de los porcentajes obtenidos de wsDomain	82
Figura 5. 29 Gráfico de los porcentajes obtenidos de wsDomain	83
Figura 5. 30 Gráfico de los porcentajes obtenidos de wsResources.....	84
Figura 5. 31 Gráfico de los porcentajes obtenidos de wsEvaluation.....	85
Figura 5. 32 Gráfico de los porcentajes obtenidos del sistema de evaluaciones	86

Índice de tablas

Tabla 3. 1 Trabajos relacionados según autores	19
Tabla 3. 2 Comparativa de plataformas LMS.....	22
Tabla 4. 1 Relación de nuevos requerimientos funcionales con los procesos de negocio....	34

Tabla 4. 2	Funcionalidad por entidad de usuario.....	39
Tabla 4. 3	Funcionalidades implementadas de usuario	39
Tabla 5. 1	Servicios web de aprendizaje para el componente del modelo de calidad para servicios web de aprendizaje	48
Tabla 5. 2	Servicios web del sistema de evaluaciones e-learning	49
Tabla 5. 3	Caso de prueba de cliente para consumir el servicio webServiceSoap	63
Tabla 5. 4	Caso de prueba del cliente para consumir el servicio webServiceSoap2	64
Tabla 5. 5	Resultados de casos de prueba del componente 2, webServiceSoap2	66
Tabla 5. 6	Ejemplo de un caso de prueba al sistema de evaluaciones.....	68
Tabla 5. 7	Concentración de pruebas del sub componente WsUser.....	69
Tabla 5. 8	Concentración de pruebas del sub componente swDomain	71
Tabla 5. 9	Concentración de pruebas del sub componente wsSubDomain	73
Tabla 5. 10	Concentración de pruebas del sub componente wsoResources.....	74
Tabla 5. 11	Concentración de pruebas del sub componente wsEvaluation	77
Tabla 5. 12	Criterios de calificación de pruebas	78
Tabla 5. 13	Registro de resultados de pruebas hechas al componente webServiceSoap	78
Tabla 5. 14	Registro de resultados de pruebas hechas al componente webServiceSoap2 ...	80
Tabla 5. 15	Porcentajes obtenidos para el sub componente wsUser	81
Tabla 5. 16	Porcentajes obtenidos para el sub componente wsDomain	82
Tabla 5. 17	Porcentajes obtenidos para el sub componente wsSubDomain.....	82
Tabla 5. 18	Porcentajes obtenidos para el sub componente	83
Tabla 5. 19	Tabla porcentajes obtenidos para el sub componente wsEvaluation.....	84
Tabla 5. 20	Porcentajes obtenidos para el sistema de evaluaciones	85



**TECNOLÓGICO
NACIONAL DE MÉXICO**

cenidet[®]
Centro Nacional de Investigación
y Desarrollo Tecnológico

1

La discriminación, el entendimiento incompleto y fragmentario, siempre se encuentran en el punto de partida del conocimiento humano.

-Masanobu Fukuoka

INTRODUCCIÓN

En este capítulo se describe brevemente el propósito de la tesis, brindando un panorama claro y conciso.

Los temas a tratar son: Introducción, descripción del problema, justificación, objetivos, alcances y limitaciones, producto resultado y beneficios

Capítulo 1. Introducción

1.1 Introducción

En la investigación y desarrollo de la generación y gestión de servicios web de aprendizaje¹, se han encontrado componentes desarrollados, los cuales tienen problemas de integración para su utilización en conjunto con los componentes dedicados a la gestión de Servicios Web de Aprendizaje. El propósito del presente trabajo de tesis es la labor de integración de componentes dirigidos a la gestión de servicios web de aprendizaje a través de la construcción de un sistema capaz de integrar los componentes dedicados a la gestión de SWA, el alcance del presente trabajo es integrar dos componentes. Uno de ellos (Componente 1) es un sistema que realiza evaluaciones a partir de un conjunto de preguntas registradas en una base de datos (Martínez, J. and Gutiérrez, G.,2015)². El otro componente (Componente 2) es el producto de software que implementa un modelo de calidad para el contenido de los servicios web de aprendizaje (Salinas Román, H.,2017)³. El presente trabajo fue realizado para solventar la necesidad de utilización e integración de componentes dedicados a la gestión de SWA.

El resultado obtenido del presente trabajo es un sistema integrador de componentes y como producto de la investigación, la definición de una metodología para la integración de componentes dirigidos a la gestión de SWA,

1.2 Descripción del problema

Las arquitecturas orientadas a servicios son modelos que prometen resolver problemas de reusabilidad de componentes de software, en particular lo que se conoce como objetos de aprendizaje para e-learning. Se han propuesto ya modelos de solución basados en arquitecturas SOA. Sin embargo, es necesario utilizar estos modelos ampliamente y es aquí en donde se presenta un problema, a pesar de que pueda haber la infraestructura, no se pueden generar beneficios porque no se cuentan con los componentes reusables en forma de servicios web de aprendizaje debido a una construcción en la arquitectura con un alto nivel de acoplamiento, para potenciar su utilización.

1.3 Objetivos

1.3.1 Objetivo general

El objetivo de la tesis que se propone es hacer que diferentes componentes de software dedicados a la gestión de servicios web de aprendizaje, puedan ser empleados de manera integrada sin tener que iniciar cada componente aparte cuando se requiera la generación de servicios web de aprendizaje.

1.3.2 Objetivos específicos

¹ Véase la referencia en el capítulo 2

² Véase la referencia 21

³ Véase la referencia 22

- Analizar dos componentes a integrar para ver su estructura y de ser necesario, realizar re ingeniería para reestructurarlos.
- Establecer el mecanismo de comunicación entre los componentes.
- Desarrollar un sistema capaz de integrar los componentes de forma conjunta para su funcionamiento.

1.4 Justificación

El propósito de esta tesis se justifica bajo la necesidad de integración de componentes que están aislados pero que están contruidos para el mismo propósito, que es la gestión de servicios web de aprendizaje. El componente de modelo de calidad de Servicios Web de aprendizaje (Salinas Román, H.,2017). Y el sistema de evaluaciones (Martínez, J. y Gutiérrez, G.,2015)., son el resultado de dos proyectos concluidos.

1.5 Alcances y limitaciones

1.5.1 Alcances

- Se integraron dos componentes.
- Se desarrolló un sistema integrador de componentes usando servicios web.

1.5.2 Limitaciones

- Se generaron servicios web para un sólo modelo, ya sea en el protocolo SOAP o RESTFul.
- Los servicios web que se desarrollaron son con propósitos de prueba del sistema, por lo tanto, no se tomó en cuenta su contenido para propósitos de evaluación

1.6 Beneficios

El principal beneficio del trabajo presente es contar con un sistema integrador de componentes de gestión de SWA, esto con el fin de solventar los problemas de integración en los componentes actuales y en componentes futuros.

1.7 Organización del documento

A continuación, se muestra la distribución de los capítulos, así como una breve descripción de los mismos.

Capítulo 2 Marco Teórico y conceptual: El capítulo 2 es usado para tomar un contexto del tema a tratar durante toda la tesis, para ello, se identifican los conceptos principales que el lector debe reconocer

Capítulo 3 Antecedentes y trabajos relacionados: Todo trabajo de tesis debe ser respaldado por trabajos que se han hecho antes del propuesto, así como trabajos que lo han precedido. El capítulo 3 da un contexto en cuanto al avance que se tiene del tema de tesis.

Capítulo 4 Metodología de solución: En este capítulo se describe la metodología de integración de componentes dedicados a la gestión de servicios web de aprendizaje, el análisis de requerimientos actualizados, así como la nueva arquitectura a partir de la metodología planteada.

Capítulo 5 Pruebas y resultados: una vez desarrollada la metodología de solución y teniendo un producto terminado se prosigue a realizar las pruebas correspondientes a los componentes integrados, también se realiza un análisis de los resultados obtenidos.

Capítulo 6 Conclusiones y trabajos futuros: con la metodología implementada y los resultados obtenidos el capítulo 6 presenta una conclusión y se proponen trabajos futuros.

2

MARCO TEÓRICO Y CONCEPTUAL

Aquella teoría que no
encuentre aplicación
práctica en la vida, es
una acrobacia del
pensamiento.

-Swami Vivekananda

En este capítulo se presentan los conceptos principales para comprender el contenido del trabajo, como: E-learning, Servicios Web de Aprendizaje, integración, entre otros.

Capítulo 2. Marco teórico y conceptual

2.1 E-Learning (aprendizaje electrónico), por Roberto Baelo Álvarez⁴

La traducción literal al español nos llevaría a la concepción de “aprendizaje electrónico”, no obstante Martín Hernández (2006) señala la concepción compleja del e-learning que engloba aquellas aplicaciones y servicios que, tomando como base las TIC, se orientan a facilitar el proceso de enseñanza aprendizaje. Esta complejidad conceptual, así como la diversidad de términos, conceptos y vocablos que se utilizan de forma indiscriminada para referirse a un sistema de formación singular, llevan a matizar lo que se entiende por e-learning como un tipo de enseñanza a distancia con un carácter abierto, interactivo y flexible que se desarrolla a través de las nuevas tecnologías de la información y la comunicación, aprovechando sobre todo las bondades que ofrece la red Internet. Por otra parte, Bartolomé (2004) señalaba como el e-learning hace referencia al desarrollo de la formación a distancia reforzada por las posibilidades educativas que tienen las TIC, así como el uso para el desarrollo de la labor formativa. Como se puede observar el e-learning se ha relacionado de forma general con el desarrollo de experiencias formativas no presenciales de carácter interactivo, abierto y flexible, que tienen en las TIC, y principalmente en la red Internet, el soporte fundamental para su desarrollo. En este sentido se toma el e-learning como el uso de las tecnologías de comunicación con fines de aprendizaje educativos, ya sea de manera presencial o a distancia.

2.2 Arquitecturas orientadas a servicios

Las arquitecturas orientadas a servicios son una forma de desarrollar sistemas distribuidos en la que los componentes del sistema son servicios independientes y se ejecutan en computadoras distribuidas geográficamente. Los protocolos estándar basados en XML, tales como SOAP y WSDL, se diseñaron para dar soporte al servicio de comunicación e intercambio de información.

Por consiguiente, los servicios son independientes de la plataforma y del lenguaje de implementación. Los sistemas de software pueden construirse al componer servicios locales y servicios externos de diferentes proveedores, con interacción uniforme entre los servicios del sistema.

2.3 Servicios web, por Carlos Andrés Morales⁵

La World Wide Web Consortium define a un servicio web como un sistema de software diseñado para soportar interacción interoperable máquina a máquina sobre una red.

Este tiene una interface descrita en un formato procesable por una máquina (específicamente WSDL). Otros sistemas interactúan con el servicio web en una manera prescrita por su descripción usando mensajes SOAP, típicamente enviados usando HTTP con una serialización XML en relación con otros estándares relacionados con la web”. Se puede

⁴ Para mayor detalle véase la referencia [1]

⁵ Para mayor detalle véase la referencia [2]

definir de manera más sencilla como un conjunto de tecnologías estándares de software para el intercambio de datos entre aplicaciones tales como SOAP, WSDL y UDDI. Estos pueden ser desarrollados en una gran variedad de lenguajes para ser implementados sobre muchos tipos de redes de computadores. El éxito de la interoperabilidad se consigue gracias a la adopción de protocolos y estándares abiertos.

The Organization for the Advancement of Structured Information Standards y el World Wide Web Consortium son los responsables de la estandarización y arquitectura de los servicios web. La industria en su interés por el desarrollo de los servicios web ha creado la WS-I (Web Services Interoperability Organization) cuya intención es la integración de los estándares que garanticen y mejoren la interoperabilidad de los servicios web. Al conjunto de servicios y protocolos para los servicios web es conocido comúnmente como “Web Services Protocol Stack” y básicamente son utilizados para definir, localizar, implementar y hacer que un servicio web interactúe con otro. Este conjunto está conformado esencialmente de cuatro subconjuntos:

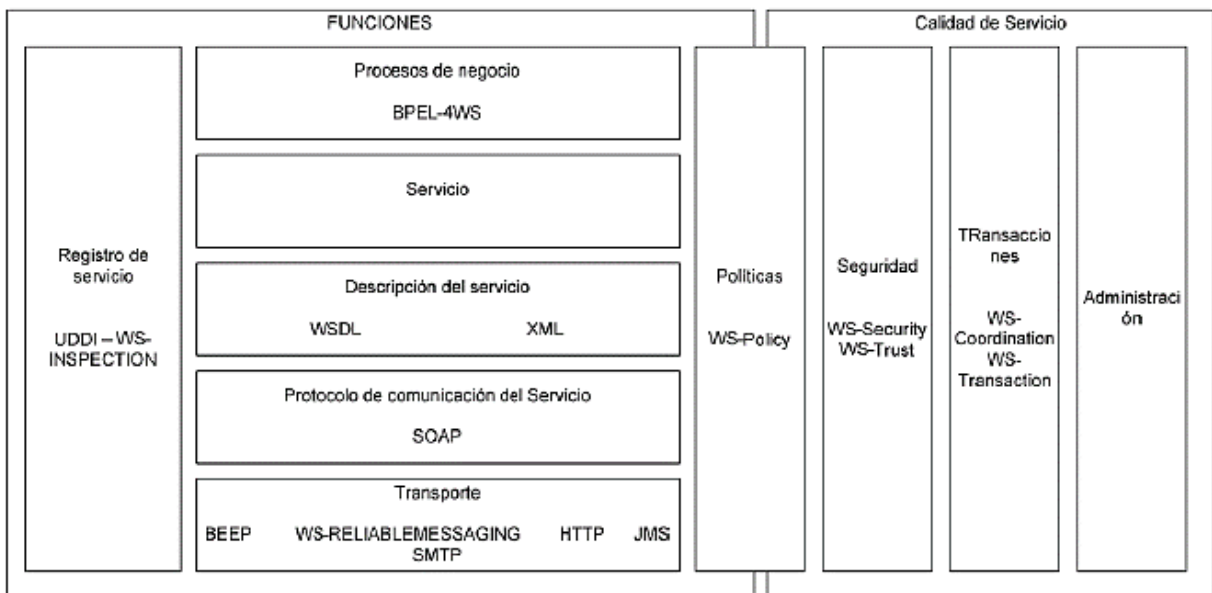


Figura 2. 1 Infraestructura de servicio web

2.4 Servicio web de aprendizaje

Un servicio web de aprendizaje (SWA) es un servicio que accede vía internet, que contiene un recurso de aprendizaje que se entrega a un cliente. También se puede definir como un servicio que funciona como un intermediario entre cliente y recurso de aprendizaje, se conoce como Servicio Web de Aprendizaje Referenciado.

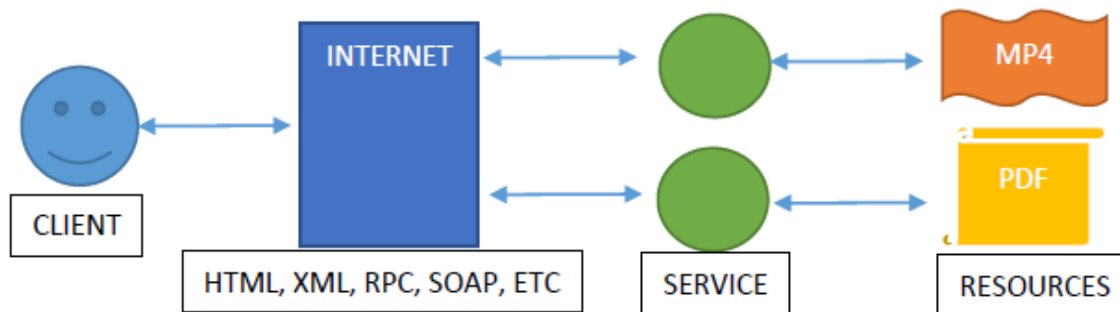


Figura 2. 2 Modelo general de un servicio Web de Aprendizaje

2.5 Componente, según Pressman⁶

Un componente es un bloque de construcción de software de cómputo. Con más formalidad, la Especificación OMG del Lenguaje de Modelado Unificado define un componente como “una parte modular, desplegable y sustituible de un sistema, que incluye la implantación y expone un conjunto de interfaces”.

Los componentes forman la arquitectura del software y, en consecuencia, juegan un papel en el logro de los objetivos y de los requerimientos del sistema que se va a construir. Como los componentes se encuentran en la arquitectura del software, deben comunicarse y colaborar con otros componentes y con entidades (otros sistemas, dispositivos, personas, etc.) que existen fuera de las fronteras del software.

2.6 Recurso, definición del 2016⁷

Los materiales digitales se denominan Recursos Educativos Digitales cuando su diseño tiene una intencionalidad educativa, cuando apuntan al logro de un objetivo de aprendizaje y cuando su diseño responde a unas características didácticas apropiadas para el aprendizaje. Están hechos para: informar sobre un tema, ayudar en la adquisición de un conocimiento, reforzar un aprendizaje, remediar una situación desfavorable, favorecer el desarrollo de una determinada competencia y evaluar conocimientos.

2.7 Integración, según Hamdan (2015)⁸

La integración de software es la práctica de vincular subsistemas o componentes de software para producir una sola unificado sistema. Es parte de cualquier ciclo de vida de desarrollo de software, ya que el software suele desarrollarse a través de diferentes fases y por un equipo de ingenieros. La integración de software se realiza en el desarrollo de software tradicional del ciclo de vida como un paso independiente en una fase posterior, después de la aplicación de software se ha completado.

⁶ Para mayor detalle, consulte la referencia [3].

⁷ Concepto obtenido del sitio: Aprendeonline.udea.edu.co, para mayor detalle, consulte la referencia [4].

⁸ Para mayor detalle, consulte la referencia [5].

2.8 Cliente, según Snell (2002)⁹

Un servicio web es una Interfaz de red accesible a la funcionalidad de la aplicación, construida utilizando Internet estándar de tecnologías (3), se sabe que los servicios web son independientes, pero necesitan ser llamados para ser consumidos, los clientes son los encargados de realizar dicha tarea, como se muestra en la siguiente imagen.

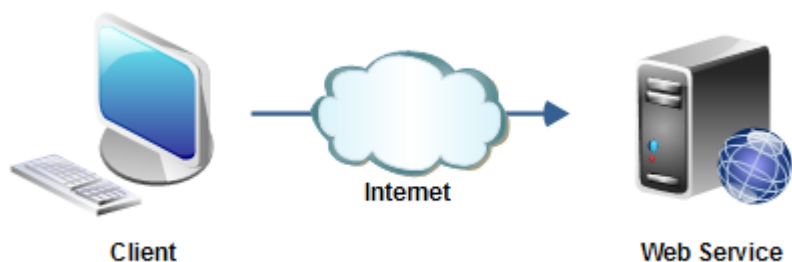


Figura 2. 3 Representación de un cliente web

⁹ Para mayor detalle, consulte la referencia [6].

3

El trabajo más productivo es el que sale de las manos de un hombre contento.
-Víctor Pauchet

ANTECEDENTES Y TRABAJOS RELACIONADOS

En este capítulo se hace un breve resumen de los antecedentes del proyecto, es decir, trabajos realizados en el CENIDET que conforman el contexto del trabajo actual. Se muestra un conjunto de trabajos relacionados al proyecto de tesis, artículos y plataformas web de e-learning.

Capítulo 3. Antecedentes y trabajos relacionados

3.1 Antecedentes

Como antecedente de este trabajo se analizaron dos tesis elaboradas en el CENIDET de acuerdo con la respuesta de problemas de integración componentes. La primera es una tesis enfocada a la composición de servicios web y la segunda de igual forma, una tesis, enfocada a una extensión de un sistema generador de servicios web a partir de diagramas de actividad.

3.1.1 composición de servicios WEB [7]

Guzmán Ruiz, M. (2006). Realizó un trabajo de tesis para la creación de clientes para consumir servicios web compuestos.

La herramienta está definida para trabajar con la composición de dos servicios web y fue realizada con éxito.

El problema que atendió esta tesis es que un usuario puede requerir varios servicios web se comunicaran para resolver una aplicación.

La forma de resolver dicho problema es a través de la construcción de tres componentes.

1. Un analizador de documentos WSDL, el cual accede al archivo WSDL de un servicio web para realizar el analizar el análisis de los métodos y sus respectivas entradas y salidas para la correcta composición.
2. Módulo de composición, se encarga de generar el archivo xml que invoca los métodos extraídos del WSDL de los servicios web. También se genera código de los stubs2, el cual es otro archivo que contiene código para la conexión entre los servicios web y el cliente. Este módulo genera el código que indica la secuencia de la invocación de los servicios web
3. Módulo de interfaz, es la interfaz gráfica para que el usuario pueda componer el servicio web, además facilita el encontrar la dirección del servicio, mostrar los métodos y seleccionar la secuencia de los mismos.

3.1.2 Extensión al sistema de composición de servicios web utilizando diagramas de actividad [8]

Otro de los trabajos que usan de integración de componentes fue la tesis de Maribell Orozco. El objetivo de esta tesis fue generar de forma automática código a partir de diagramas de actividad BPEL usando los elementos fork, join, mensajes de flujo, expresiones de resguardo y note.

El problema se presentaba a la limitación que los usuarios tenían al usar los elementos de fork, join, mensajes de flujo, expresiones de resguardo y note, en WS-SIDDOO, ya que no se podía generar el código en BPELW4S.

La forma de solución consistió en crear una extensión de gramática para poder utilizar la notación faltante y a la vez generarla en el código BPEL, de esta forma la limitación a la hora de crear diagramas de actividad.

Los resultados obtenidos fueron la obtención del código BPEL de acuerdo a la representación visual del diagrama de actividad, incluyendo el uso de los elementos fork, join, mensajes de flujo, expresiones de resguardo y note.

3.2 Trabajos relacionados

3.2.1 Trabajos relacionados

El objetivo de la tesis es la integración de componentes para la generación de servicios web de aprendizaje, actualmente existen muchas plataformas dirigidas al e-learning, pero en especial el dominio en el que se centra es el aprendizaje electrónico, los trabajos relacionados se dividen en dos secciones, son trabajos relacionados con autores en artículos y algunas plataformas educativas y sus principales características.

3.2.1.1 Integration of web services with e-learning for knowledge [9]

El propósito del autor en este artículo es crear una Sociedad del conocimiento a través de del E-learning, esto con apoyo de los servicios web.

La idea es crear un sistema e-learning en base a los distintos componentes del e-learning para crear una base de conocimientos. cada componente es tratado como si fuera un servicio web, los componentes mencionados son: personas (instructores, profesores, estudiantes), infraestructura técnica y entorno adecuado, herramientas de comunicación e intercambio de recursos, contenido de conocimiento y recursos digitales.

Lo que se obtiene es un marco, plasmado en un diagrama donde se muestran los elementos que componen el e-learning y el uso de la web 2.0 y 3.0, como el resultado del sistema se obtiene una base de conocimiento generada por la web, incluso con la ayuda de algoritmos de IA para explotar la información.

Como se había mencionado antes, dicho autor propone el marco y el esquema general, mas no se tiene un producto, como conclusión se llega a que tal integración de servicios web con cada uno de los componentes del sistema e-learning da una nueva dirección al e-learning permitiendo al estudiante como el potencial productor de contenido.

3.2.1.2 Software architecture for integration of institutional and social learning environments [10]

La evolución de la tecnología hace que la metodologías efectivas de enseñanza crezcan y cambien, Los sistemas de gestión de aprendizaje (LMS) muchas veces son adaptados a sistemas externos, sin embargo la mayoría de estos sistemas están limitados a su integración a sistemas externos por la arquitectura que maneja, en este artículo se centra en el diseño y los requerimientos tecnológicos de la arquitectura de software para integrar un sistema e-learning institucional, un sistema de gestión educativa y un entorno de aprendizaje social.

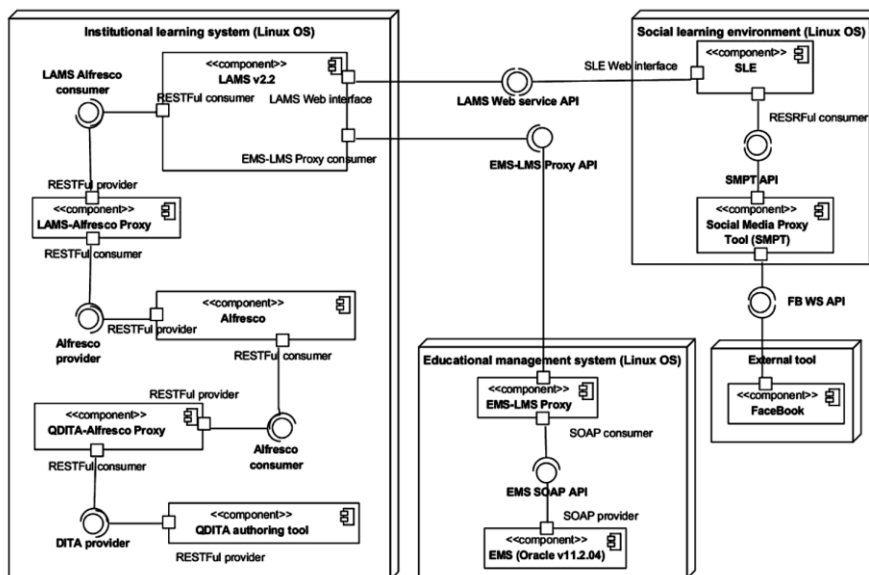


Figura 3. 1 Arquitectura planteada para la integración de componentes en el artículo Software architecture for integration of institutional and social learning environments

La forma de crear dicha arquitectura se basa en primera instancia que los componentes ya existían. El siguiente paso consiste en definir los requisitos que tienen que cumplir, el sistema e-learning y cada subsistema tienen que estar creados con base en servicios web para que la integración sea más sencilla y, por último, la información generada por los servicios web es compartida entre los subsistemas.

El resultado del trabajo es un prototipo con el sistema y los subsistemas desarrollados, en el artículo hay varias capturas de lecciones del curso, una vista de comentarios hechos por Facebook en un tema, y una vista de un curso de redes. La integración hecha por servicios web fue satisfactoria usando SOA y los protocolos SOAP y RESTful.

Como documentación tienen la prueba del sistema hecha por estudiantes, los cuales están satisfechos según el artículo. También cuentan con la documentación del sistema, ya que se centraron en realizar la arquitectura en base a los requerimientos.

3.2.1.3 Analysis of knowledge management and e-learning integration models [11]

El propósito del artículo es el análisis de los modelos existentes de integración para el e-learning y la gestión del conocimiento con un enfoque de integración a la empresa.

El artículo se centra sólo en el estudio de los modelos para la integración del e-learning y la gestión del conocimiento, algunos modelos son el de integración de tecnología, Integración de la gestión del conocimiento y del e-Learning mediante el aprendizaje corporativo contextual, un sistema integrado por un marco de diseño en el dominio de e-Learning para la difusión del conocimiento entre otros. Se obtuvo un modelo donde se identifican los elementos que integran una base de conocimientos en el e-learning, como entidades.

El resultado es la identificación de siete modelos para la integración de KM y e-learning, solamente el modelo de Schmidt se implementó como un prototipo relativamente aceptado, mientras que los otros modelos necesitan ser más elaborados para su posterior implementación, en conclusión, se llega a que la diversidad de modelos de integración

demuestra claramente que no hay una manera generalmente aceptada en función de las necesidades de la organización.

3.2.1.4 A Web Services-enabled B2B Integration Approach for SMEs [12]

En la última década, las empresas orientadas a la gestión de negocios han buscado y usado las tecnologías para apoyar la colaboración entre las empresas, a esta colaboración se le llama B2B. Dentro de la problemática se busca un intercambio eficaz de información entre estos negocios, con base en el enunciado se requiere una arquitectura flexible y estandarizada para apoyar mejor la conexión entre varias aplicaciones y el intercambio de datos. La solución encontrada es el uso de arquitecturas orientadas a servicios, para una comunicación entre múltiples plataformas, de esta forma se pueden simular los mecanismos que se usan en las colaboraciones B2B basadas ahora en servicios web. Con la consideración de las características de los servicios web y las funcionalidades de TI de las Pymes, se construyeron las funcionalidades por bloque y el mecanismo del enfoque de integración B2B orientado a servicios web, a continuación, se mencionan dichas funcionalidades.

- Servicios web de puerta de enlace. Envía y reciben mensajes por una puerta de enlace B2Bi.
- Cliente de puerta de enlace. Cliente que envía y recibe mensajes proporcionados por la url del wsdl.
- Daemon. Examina periódicamente la ubicación específica para los mensajes nuevos.
- Administrador de mensajes. Analiza y genera los mensajes entrantes y salientes según el esquema del xml.
- Controlador de proceso. Funcionalidad encargada la configuración y gestión de procesos empresariales.
- Carpetas específicas. Gestión de tipos de carpetas para almacenar mensajes.

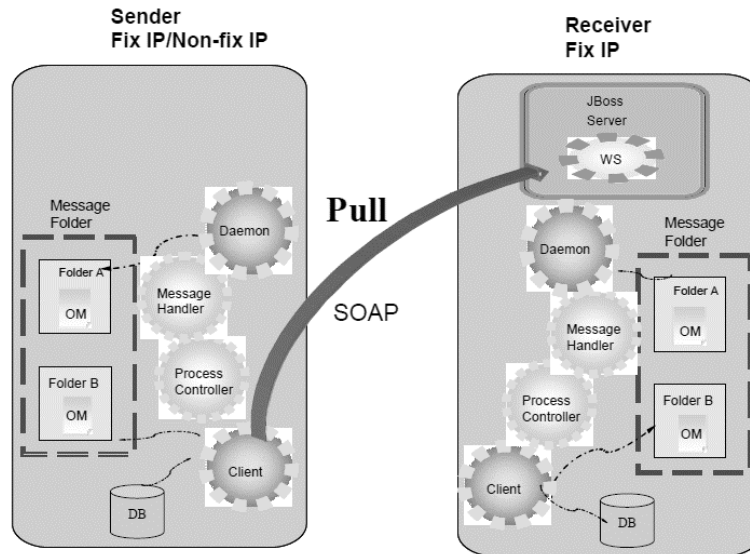


Figura 3. 2 Representación de Web Services-enabled B2B

3.2.1.5 Improving the dependability of Web Services Integration [13]

Los servicios web y las arquitecturas orientadas a servicios (SOA) presentan un nuevo paradigma en las aplicaciones informáticas distribuidas, los servicios web ofrecen muchas ventajas sobre las plataformas y arquitecturas convencionales de middleware de computación distribuida, debido a sus beneficios atraen a más simpatizantes, aunque los sw traen oportunidades junto con los desafíos, uno de ellos es garantizar la confiabilidad. Una aplicación de servicios web se puede integrar dinámicamente por la UDDI, este tipo de integración puede socavar la confiabilidad, dado a que no se puede asegurar personalmente la disponibilidad y otros factores que afectan la confiabilidad del servicio.

En el artículo se propone un enfoque para mejorar la confiabilidad de la integración de los servicios mediante una arquitectura mediadora estándar para respaldar la integración dinámica de los servicios web. El mediador es implementado para aumentarla resiliencia dentro de la arquitectura adaptando las técnicas de tolerancia a fallos y la redundancia de servicios disponibles. Con un mecanismo de monitoreo el servicio web genera datos de confiabilidad como tiempo de respuesta, tasa de fallos, tipos de fallos etc.

Para prueba se ha implementado un framework java independiente de la plataforma que se puede integrar fácilmente en cualquier aplicación de servicios web java.

Los resultados obtenidos fueron satisfactorios, el enfoque fue probado dentro de la arquitectura planteada, actualmente se traba en su mejora e integración de componentes eficiencia en el mundo real, en la figura se puede apreciar la integración del mediador.

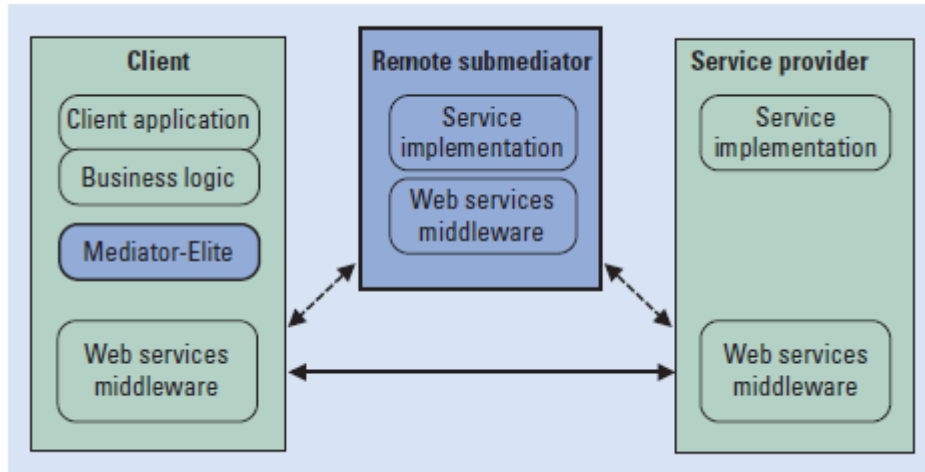


Figura 3. 3 Marco Java WS-mediator

3.2.1.6 A Cloud Framework for Enterprise Application Integration on SOA [14]

La agilidad de negocio y la capacidad de adaptarse rápidamente al cambio es uno de los requisitos competitivos para la mayoría de las empresas. El crecimiento de los servicios web, la arquitectura orientada a servicios y el cloud computing son factibles para hacer la integración de aplicaciones empresariales (EAI) e interacciones automáticas entre otras empresas.

La mayoría de las empresas ven la integración de datos como el paso más crítico para la integración de sistemas de información dentro de la organización, la respuesta en tiempo real al cambio de negocio requiere una infraestructura de aplicaciones.

En el artículo se propone un marco de nube que integra los datos de la empresa de diversas fuentes a través de un mecanismo de intermediarios de mensajes, proporcionar acceso remoto a las bases de datos y tener consistencia de datos entre las aplicaciones empresariales. A través del marco propuesto, la información empresarial se puede integrar de una más eficiente y efectiva.

El objetivo de un mecanismo de integración de datos es mover los datos entre almacenes de datos con el fin de compartir información comercial entre diversas aplicaciones empresariales. El resultado obtenido fue un sistema que integra las aplicaciones empresariales y como resultado la integración de datos.

Se simuló las bases de datos con la información correspondiente y el flujo de datos a un alta, mediana y baja intensidad, los resultados fueron satisfactorios al integrar la arquitectura y los datos de forma correcta.

En la figura se muestra el diseño de la arquitectura implementada.

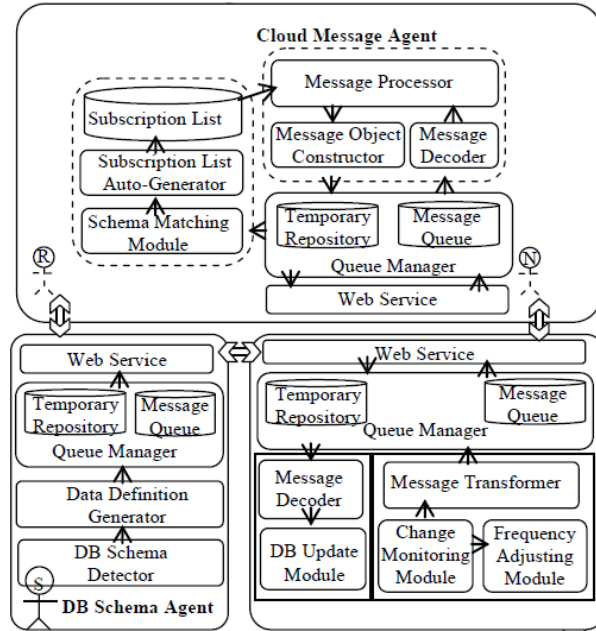


Figura 3. 4 Marco funcional de la arquitectura en la nube

3.2.1.7 Web-Service Interface-based IMS-SPSC Integration. [15]

El trabajo describe la integración de sistemas inteligentes de mantenimiento mediante servicios web, el artículo hace una investigación de la interfaz que un servicio web puede proporcionar para la captura de conocimiento distribuido entre diferentes sistemas y humanos.

El problema principal dentro de los sistemas es la falta integración de datos, ya que los sistemas de mantenimiento inteligente se encuentran distribuidos, el objetivo es proporcionar una planificación colaborativa y coordinación descentralizada. Realizando la investigación correspondiente se encontró que los servicios web satisfacen las necesidades de reutilización, interoperabilidad e integración. En el artículo se puede apreciar la implementación de un modelo con una arquitectura orientada a servicios

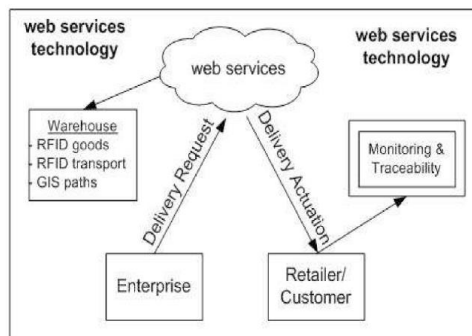


Figura 3. 5 Diseño de arquitectura para IMS-SPSC basada en servicios web

Se realizaron módulos de proveedor, fabricante, distribuidor, centro de servicio y cliente, cualquier módulo de la cadena puede usar la herramienta que está establecida, los servicios se modelaron como eventos para cada actor involucrado en la integración

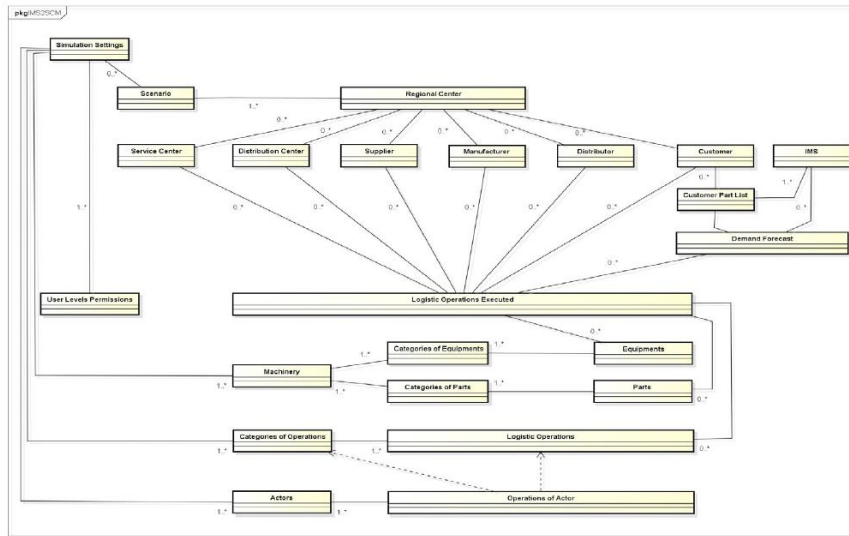


Figura 3. 6 Diagrama de clases del sistema IMS-SPSC

3.2.1.8 Web Service Integration within Next Generation CEMIS [16]

El trabajo se centra en una implementación de la plataforma Next Generation, proveniente de los sistemas de información de gestión ambiental corporativa (CEMIS), la cual se dedica a que las empresas y sus procesos sean más respetuosos del medio ambiente por medio del procesamiento de información, el trabajo realiza la integración de dos componentes principales: el sistema de administración de flujo de trabajo implementado y el centro de servicios ecológicos, hechos como servicios web estandarizados.

La forma en que lograron integrar los dos componentes fue mediante servicios de acoplamiento que encapsulan las funciones CEMIS, logrando que se llamen desde una descripción de la aplicación con una orquestación web, interfaz de usuario y eventos emitidos vinculados a diferentes elementos de interacción. Se usó un formato de descripción SCXML como un lenguaje de definición que permite la interpretación en tiempo de ejecución, dentro del sistema usaron un formato independiente, DSL, el cual se integró perfectamente debido a que el formato usado en SCXML fue el mismo. Algunas funcionalidades dentro de los módulos son procedimientos complejos, por ellos se realizó una interfaz para cada uno.

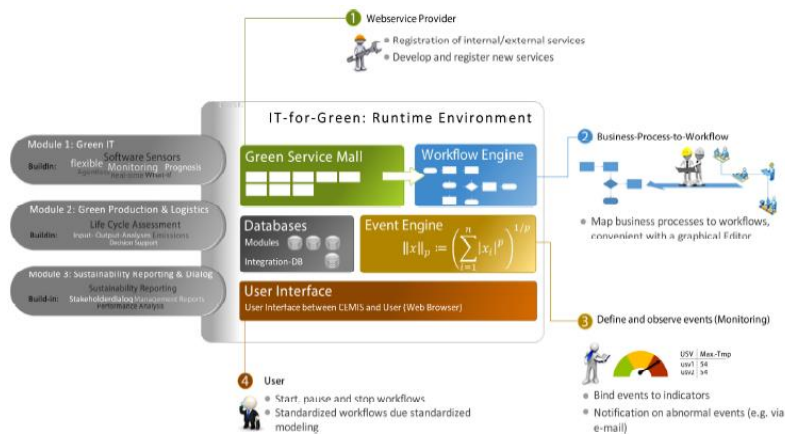


Figura 3. 7 Arquitectura Next Generation CEMIS

Tabla 3. 1 Trabajos relacionados según autores

Titulo	Uso de servicios web	Forma de integración	Producto obtenido	Dominio del e-learning	Tipo de diseñado usado
Integration of web services with e-learning for knowledge society	Si	Marco mediante servicios web	Marco orientado a servicios	Si	Marco y modelo
Software architecture for integration of institutional and social learning environments	SOAP y RESTFul	Implementación de servicios web SOAP y RESTFul	Arquitectura y prototipo inicial	Si	Arquitectura bajo componentes
Analysis of knowledge management and e-learning integration models	No	Modelo conceptual Servicio por entidad	Análisis de modelos	Si	Modelos obtenidos
A Web Services-enabled B2B Integration Approach for SMEs	SOAP	Funcionalidades por bloque	Diseño de arquitectura para las B2B basada en servicios web	No	Arquitectura SOA
Dependability of Web Services Integration	SOAP	Componente middleware	Nuevo enfoque de integración	No	Marco con mediador

dinámica de sw					
A Cloud Framework for Enterprise Application Integration on SOA	SOAP	Arquitectura en la nube	Sistema integardor de aplicaciones empresariales	No	Marco
Web-Service Interface-based IMS-SPSC Integration.	SOAP	Integración de servicios de eventos por autor	Integración de sistemas reutilizables	No	Arquitectura de clases
Web Service Integration within Next Generation CEMIS	SOAP	Uso de SCXML para comunicación universal	Componentes integrados	No	Arquitectura de componentes

3.2.2 Plataformas relacionadas

3.2.2.1 Edu next [17]

Es una plataforma en línea para diseñar cursos en líneas con fines meramente educativos, sus creadores son los fundadores de MIT y la universidad de Harvard, algunas de las características que lo hacen resaltar es:

- Plataforma abierta. Edu next tiene la extensión de open edX el cual permite a los desarrolladores, educadores y público en general poder crear sus propios cursos y crear herramientas con fines específicos, es una plataforma sin fines lucrativos.
- edX Studio. Se pueden crear y manipular cursos para las necesidades específicas del usuario, se puede también usar estructuras para añadir videos y documentos. No se necesita de ningún plugin para esto.
- Open edX LMS. Es una herramienta de esta plataforma para que los usuarios estudiantes puedan acceder y comprobar sus avances y calificaciones. El instructor posee un dashboard para poder administrar a sus alumnos. También ofrece una plataforma para un foro de discusión.
- Xblock. Es la arquitectura en la que está basado open Edx, bajo componentes. A partir de cursos y secciones de aprendizaje más grandes, se pueden realizar componentes que se pueden desplegar fácilmente.

3.2.2.2 Spot LMS [18]

Proveniente de Marseille Francia, es una herramienta o aplicación para formación online, para alumnos y profesores, el alumno puede usar los cursos online, el maestro o formador puede seguir el progreso de los estudiantes, el curso e incluso puede hacer video llamadas. El administrador puede realizar seguimiento en tiempo real, y puede usar plantillas de los recursos creados, algunas de sus características son:

- Herramienta de formación online para la creación de cursos, difusión, seguimiento de los alumnos, la administración y edición de informes, gratuita
- Se distribuyen sus cursos de e-learning (incluyendo videos) en una red local sin necesidad de conexión a Internet, esta funcionalidad está en progreso aun
- Dashboard de administrador: la funcionalidad de monitorear cursos y ver el progreso de los estudiantes

3.2.2.3 CAE innovative learning solutions [19]

Proveniente de Estados Unidos, CAE ofrece soluciones en formación e-learning que utilizan una metodología que integra las últimas tendencias en tecnología aplicada a la formación presencial, blended³. Cuenta con más de 35 años desarrollando tecnología y contenido multimedia especializado en educación, ofreciendo las soluciones y los servicios de la formación.

La empresa cuenta con un enfoque más serio y con contenidos de calidad, avalados por los certificados de calidad que cumplen. Entre sus principales características está:

- Portal educativo de idiomas. CAE pone a disposición una plataforma educativa personalizable y escalable, se puede acceder a la formación blended mediante un portal educativo especializado en idiomas, cuenta con todos los recursos en un mismo sitio.
- LMS/LCMS con aula virtual y comunidad. Cuentan con dicha tecnología para gestionar proyectos de formación e-learning y aprendizaje en laboratorios online/offline.
- Contenidos SCORM. Contenidos interactivos 100% en formato scorm, son estructurados de forma pedagógica y cuenta con un amplio catálogo.
- Contenido online. Los cursos y contenidos de la plataforma están creados por una herramienta propia, hecha por los expertos de la misma, los cuales se actualizan con las nuevas metodologías.
- DexWay. Es una extensión de CAE que se dedica especialmente a la formación e-learning en el dominio de idiomas el cual ofrece cursos online, blended y aulas en red.
- Metodología power, human, social and mobile. Son los cuatro pilares por los que se rige la enseñanza de la plataforma.

- Como comentario adicional la empresa cuenta con 35 años de trayectoria, más de 700 contenidos educativos, más de 100 horas de formación y más de 2.5 millones de alumnos formados.

3.2.2.4 Paradiso [20]

Paradiso es de Estados Unidos, es una plataforma de aprendizaje virtual basada en los inicios de Moodle, aunque totalmente modificada para brindar una mejor experiencia. Las ventajas sobre su LMS son muchas, principalmente la posibilidad de no depender de expertos en programación y brindar todas las herramientas necesarias para una correcta administración de esta, tanto de implementación de imágenes corporativas hasta construcción de reportes.

Entre las características principales de la plataforma se encuentra.

- Interfaz personalizable.
- Compatible con equipos móviles
- Extensión empresarial
- Compatible con SCORM y Tin Can API

A continuación, se hace un análisis comparativo de las plataformas encontradas hasta el momento, el significado de cada característica se describe en la siguiente lista.

- Arquitectura: Se define cual es la estructura general de cada sistema.
- Modalidad de pago: en algunas plataformas el uso de los cursos y recursos son bajo un pago.
- Creación de servicios: la plataforma que cumple esta característica es capaz de crear recursos como servicios web.
- Abierto/cerrado: En algunos casos la plataforma tiene sus componentes abiertos a modificaciones por parte de los usuarios.
- Valor agregado: la característica que hace diferente a la plataforma de las demás.

Tabla 3. 2 Comparativa de plataformas LMS

Nombre Proyecto	País de origen	Arquitectura	Modalidad pago	Gestión de servicios	Abierto/cerrado	Valor agregado
Edu Next	Estados Unidos, Harvard	Componentes	Si	Si	Abierto	Foro de discusión
Spot LMS	Marseille, Francia	Web, componentes MOOC y SPOC	Si	No	Cerrado	Servidor Streaming
CAE innovative	Estados Unidos	SOA	Si	No	Cerrado	Metodología propia
Paradiso	Estados Unidos	SOA	si	no	Cerrado	Compatibilidad con móviles

4

Divide las dificultades que examines en tantas partes como sea posible, para su mejor solución.

-René de Descartes

METODOLOGÍA DE SOLUCIÓN

En este capítulo se describe la metodología de solución aplicada para el presente trabajo de investigación. De acuerdo con los trabajos relacionados y la investigación descrita en el capítulo 3, se propone una metodología capaz de integrar componentes de gestión de servicios web de aprendizaje.

Capítulo 4. Metodología de solución

Como se ha visto en el capítulo 3, según los trabajos relacionados por el autor, existen múltiples formas de integrar componentes de los cuales, la mayoría son de dominios distintos a la gestión de servicios web de aprendizaje o del dominio del e-learning.

Se han tomado las formas de integración que usan en los trabajos relacionados destacados para poder formar una metodología para integrar componentes dedicados a la gestión de servicios web de aprendizaje, a esto se le ha añadido el uso del patrón de diseño facade, el cual es usado ampliamente como un patrón de integración de sistemas y subsistemas.

El diagrama de la figura 4.1 describe las 5 actividades generales de la metodología para poder realizar la integración de componentes de gestión de SWA. El proceso 5 (Figura 4.22) es un proceso compuesto el cual se describirá más adelante.

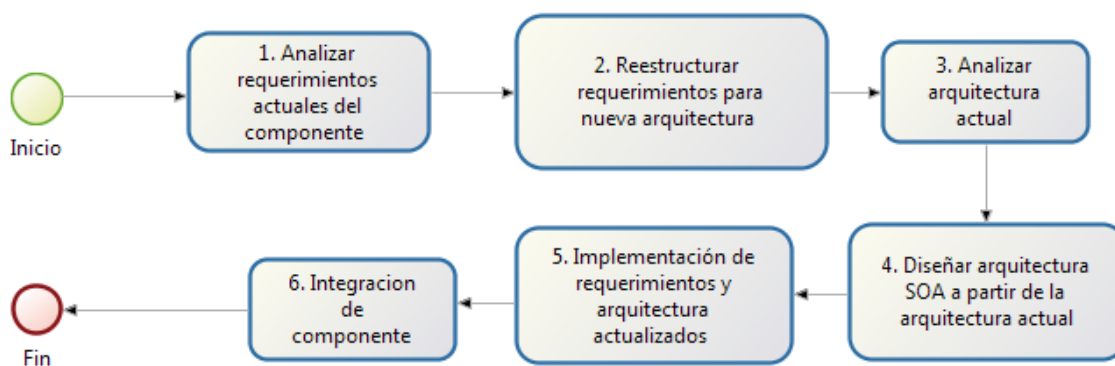


Figura 4. 1 Metodología general de solución

A continuación, se describen las tareas a realizar para la solucionar el problema a tratar, el primer componente a tratar es: un sistema de evaluaciones.

4. 1 Analizar requerimientos actuales del componente

Como se definió en los alcances del presente trabajo, el propósito es integrar dos componentes, el componente de modelo de calidad de Servicios Web de aprendizaje (Salinas Román, H.,2017), consiste en un componente que mide la reusabilidad de los servicios web de aprendizaje. Actualmente cuenta con una arquitectura implementada según la metodología definida en la figura 4.1, y está lista para integrarse como componente dentro del sistema.

El componente de sistema de evaluaciones (Martínez, J. y Gutiérrez, G.,2015) consiste en un sistema que permite realizar evaluaciones de un dominio usando varios tipos de evaluaciones, específicamente cuatro: verdadero o falso, opción múltiple, relación de columnas y colocar en orden. dicho componente cuenta con problemas de integración y alto acoplamiento entre los componentes internos del mismo.

El sistema cuenta con la opción de poder editar los recursos disponibles, así como llevar un seguimiento básico de las calificaciones, también se permite la administración de los usuarios registrados, A continuación, se nombran los requerimientos funcionales.

4.1.1 Requerimientos funcionales del sistema de evaluación (Martínez, J. y Gutiérrez, G.,2015)

RF1 El sistema permite el inicio de sesión con un usuario y contraseña previamente registrados.

RF2 El sistema permite crear usuarios.

RF3 El sistema permite la conexión a base de datos

RF4 El usuario puede consultar los recursos disponibles en la base de datos previamente seleccionada.

RF5 el usuario puede elegir los recursos disponibles a través de una lista desplegable.

RF6 El usuario puede visualizar el recurso seleccionado

RF7 El sistema permite al usuario editar el recurso seleccionado

RF8 El sistema permite evaluar los test seleccionados por el usuario.

RF9 El sistema permite editar los datos del usuario previamente registrado e iniciado sesión.

RF10 El sistema permite eliminar la cuenta del usuario que previamente ha iniciado sesión.

RF11 El sistema permite visualizar el historial de los test contestados por los alumnos.

4.2 Reestructurar requerimientos para una arquitectura nueva

Con base en los requerimientos anteriores se establecen las nuevas funcionalidades que permitan dar flexibilidad al propósito original del sistema, recordando que el objetivo es generar una arquitectura orientada a servicios, donde los requerimientos deben quedar ajustados a funcionalidades por entidad.

Funcionalidades por entidad

Una de los pasos expuestos en la metodología para integrar componentes de gestión de SWA, es ajustar los requerimientos y agruparlos por entidad, de acuerdo al trabajo relacionado: Analysis of knowledge management and e-learning integration models, descrito en el capítulo 3, antecedentes y trabajos relacionados.

Una entidad¹⁰ es la representación de un objeto o un concepto en el del mundo real que se describe en un sistema o una base de datos. El propósito de la separación de funcionalidades por entidad, es brindar sub componentes totalmente independientes que sean capaces de desplegarse de forma independiente, pero a la vez poder funcionar en conjunto, ya que siguen formando parte del mismo sistema.

- RF1: El Sistema ahora entrara desde un sistema principal
- RF2: El sistema permite crear usuarios con roles.
- RF3: el sistema permite conexión a cualquier base de datos, con el esquema seleccionado.
- RF4: el alumno puede consultar las evaluaciones mientras que el docente puede consultar los dominios, recursos y evaluaciones.
- RF5: el usuario alumno puede elegir una evaluación a partir de un dominio previamente seleccionado.
- RF7: El sistema permite al docente modificar evaluaciones, dominios y recursos.
- RF8: El sistema permite evaluar los test seleccionados por el usuario.
- RF9: El sistema permite al administrador modificar y eliminar las cuentas de usuario.
- RF10: el usuario alumno solo puede ver su historial mientras que el profesor puede consultar todos los historiales.

Con base en los requerimientos anteriores se plantea en un diagrama de procesos de negocios los cambios requeridos.

RF1: El usuario podrá ingresar al sistema con una cuenta previamente registrada, accediendo desde el sistema principal. La figura 4.2 muestra el estado actual del componente y la figura 4.3 muestra cómo debe cumplir con el requerimiento.

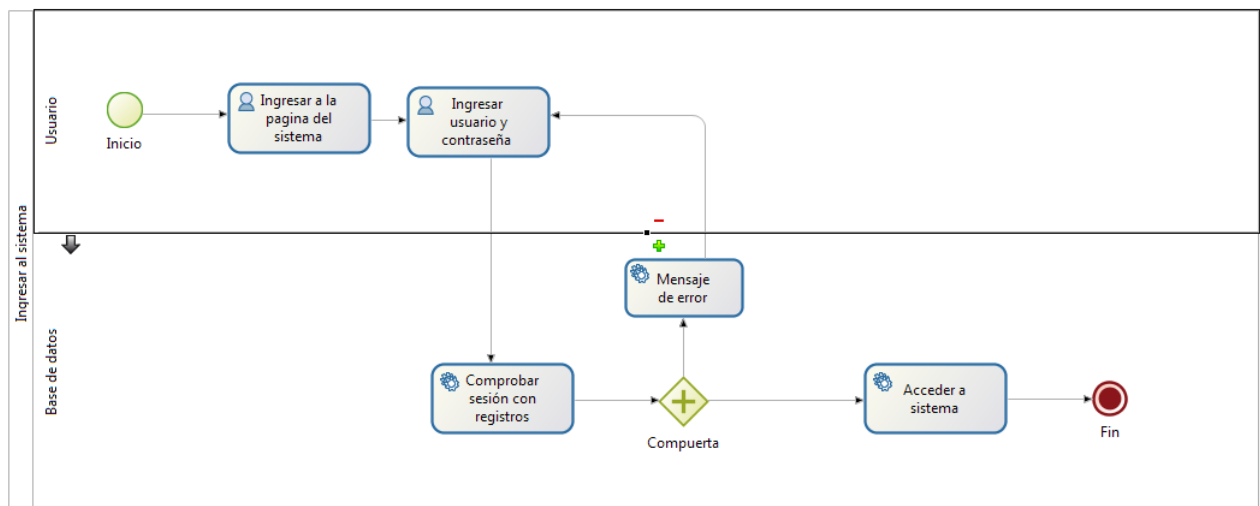


Figura 4. 2RF1, primera versión

¹⁰ Véase referencia 23

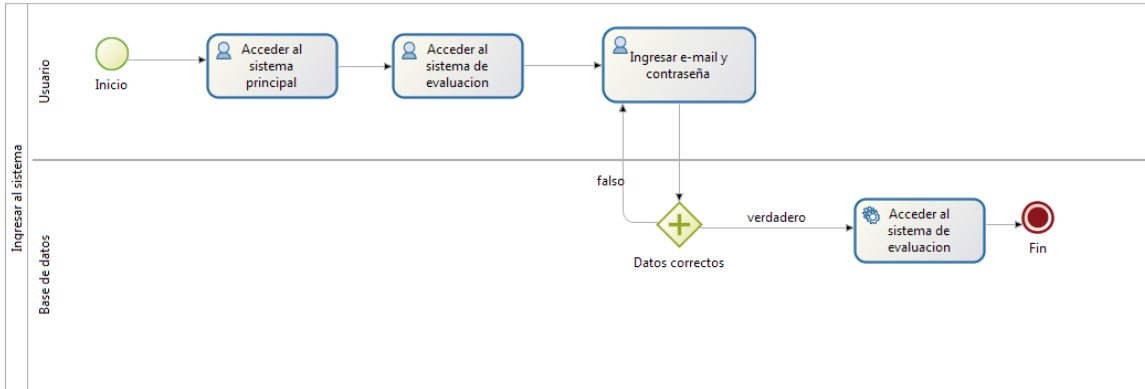


Figura 4. 3 RF1, versión actual

El cambio en el nuevo requerimiento está en acceder a través de un sistema principal y que ahora el método de inicio de sesión será con los campos de correo electrónico y contraseña como se aprecia en la figura 4.3.

RF2: El sistema permite la creación de usuarios, la primera versión del componente se visualiza en la figura 4.4, mientras que la versión actual se ve en la figura 4.5.

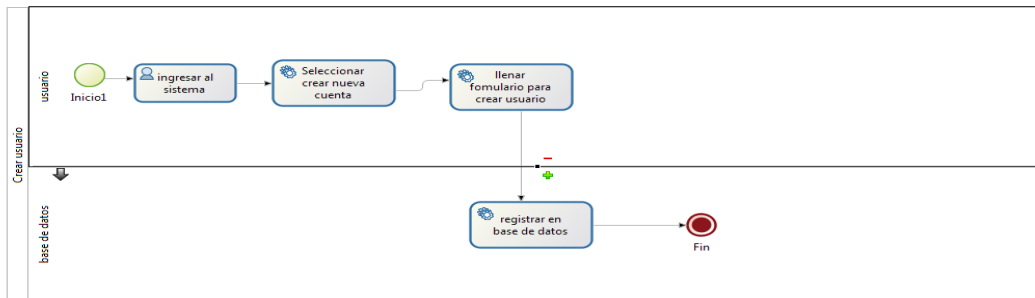


Figura 4. 4 RF2, primera versión

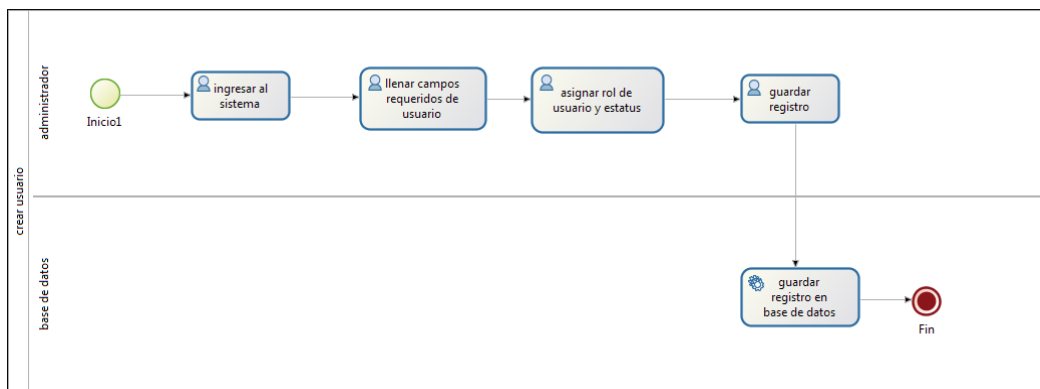


Figura 4. 5RF2, versión final

El cambio del requerimiento 2 es asignar un rol y estatus, ya que el docente tendrá actividades propias, es necesario editar el requerimiento funcional como se aprecia en la figura 4.5.

RF3: el sistema permitirá la conexión con distintas bases de datos, con el mismo esquema, la primera versión del requerimiento se aprecia en la figura 4.6 y la nueva versión con los cambios se aprecia en la figura 4.7.

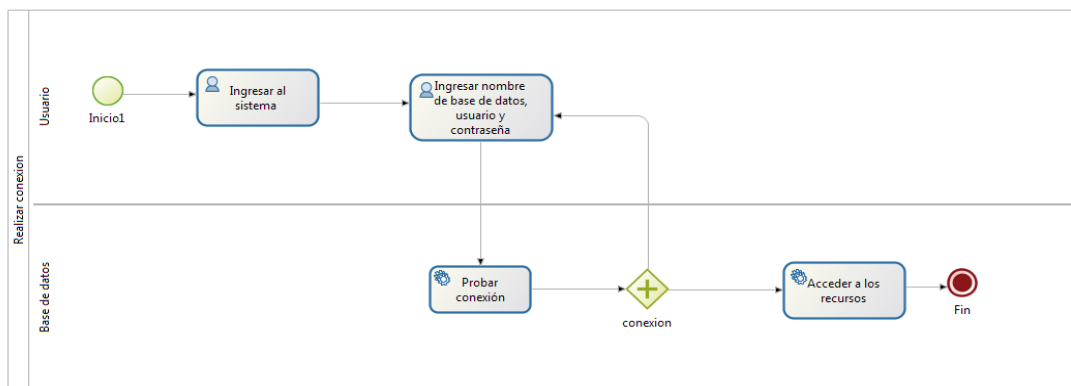


Figura 4. 6 RF3, primera versión

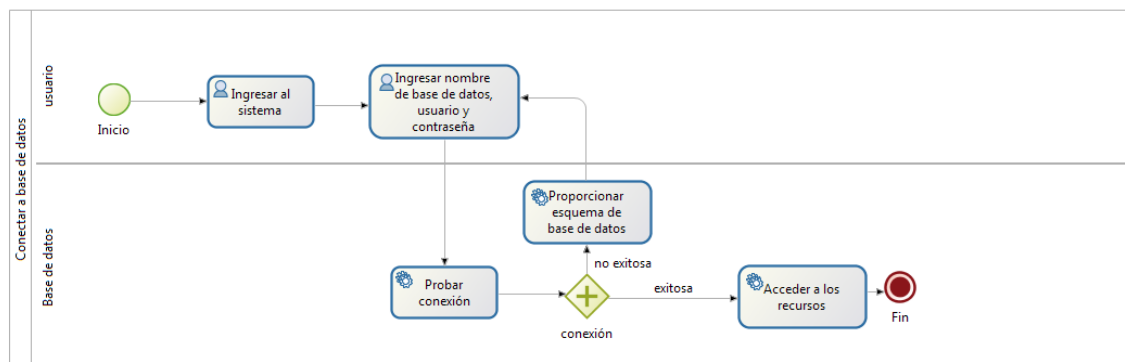


Figura 4. 7 RF3, Versión final

Los cambios hechos en el nuevo requerimiento se hicieron para facilitar la conexión con más de una base de datos, se añadió proporcionar el esquema del mismo, como se aprecia en la figura 4.7.

RF4: el alumno puede consultar las evaluaciones mientras que el docente puede consultar los dominios, recursos y evaluaciones, la primera versión del requerimiento se ve en la figura 4.8 y la actual versión se ve en la figura 4.9.

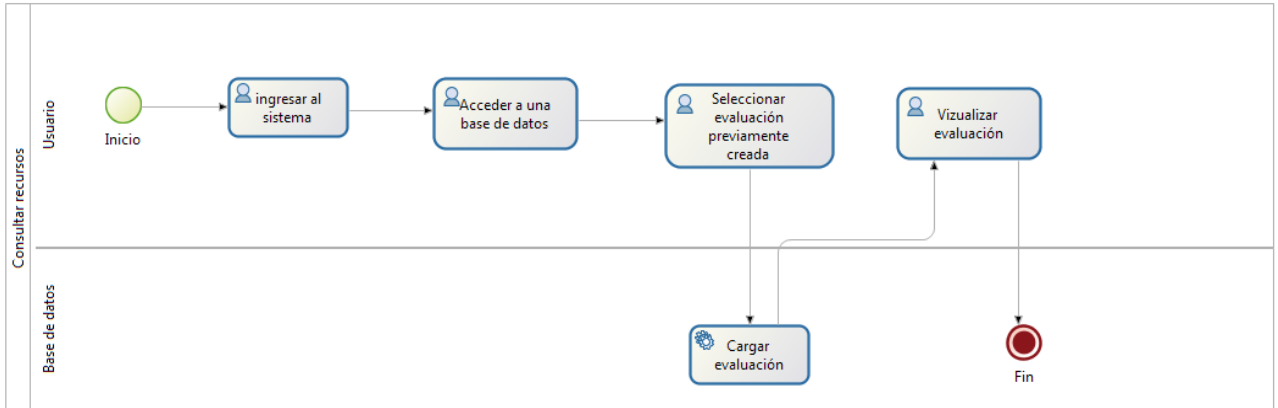


Figura 4. 8 RF4, primera versión

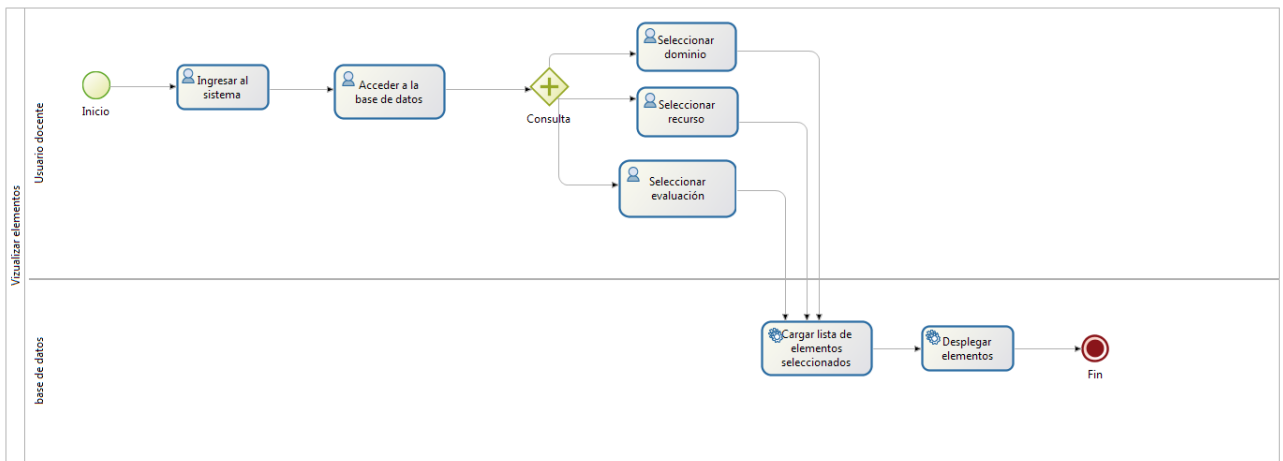


Figura 4. 9 RF4, versión final

Con base la modificación del requerimiento 4, se restringe la visualización de dominios, recursos y evaluaciones, el alumno solo puede ver las evaluaciones que necesita ver, como se aprecia en la figura 4.9.

RF5: el usuario alumno puede elegir una evaluación a partir de un dominio previamente seleccionado, la primera versión del requerimiento funcional 5 se aprecia en la figura 4.10 y la actualización del mismo se aprecia en la figura 4.11.

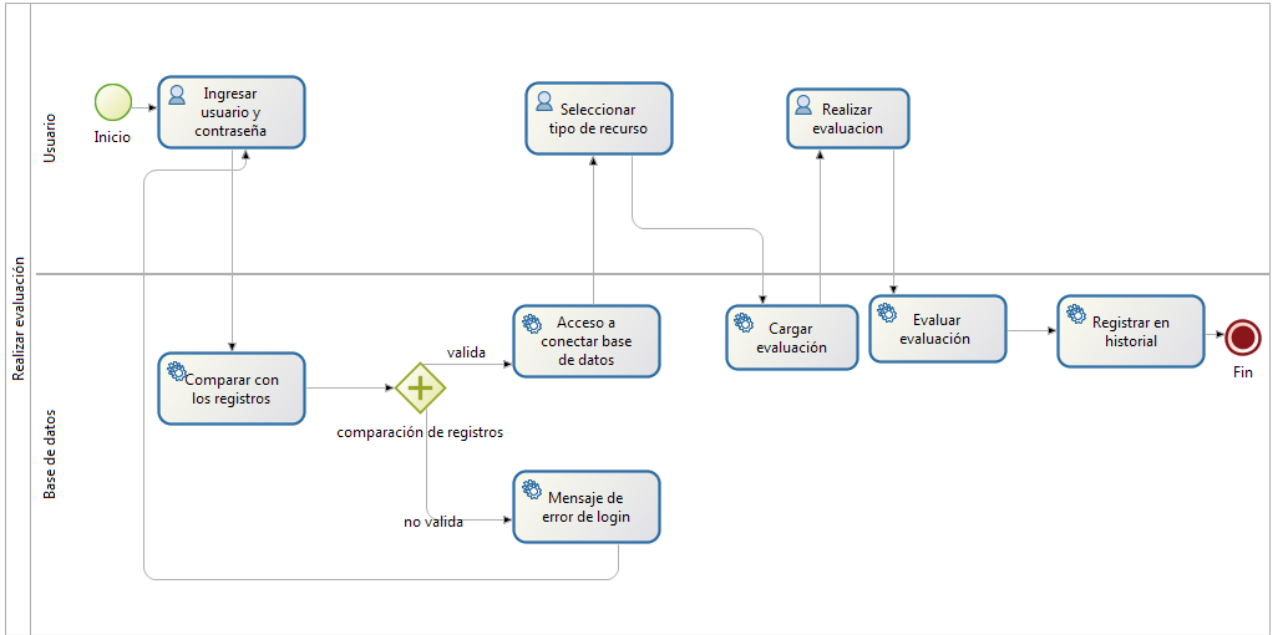


Figura 4. 10 RF5, primera versión

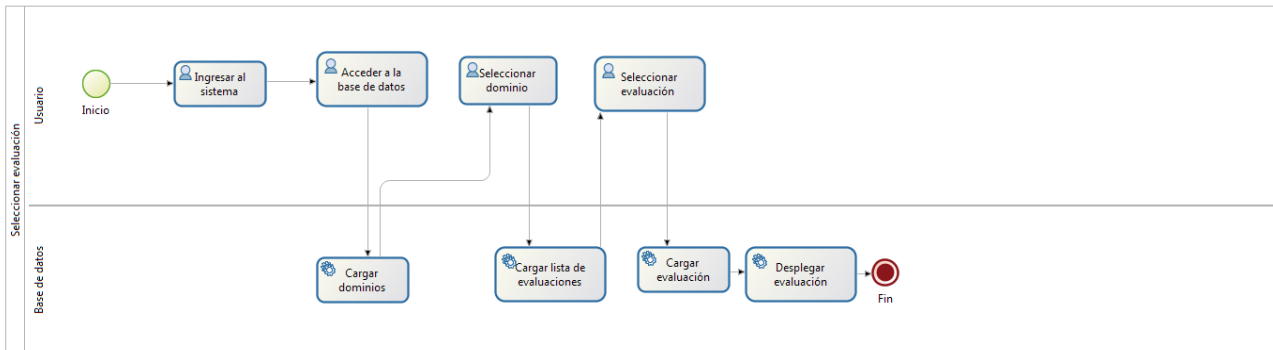


Figura 4. 11 RF5, versión Final

A diferencia de la primera versión del requerimiento cinco, con base en la actualización del requerimiento, permite elegir múltiples evaluaciones a partir de un dominio seleccionado, como se aprecia en la figura 4.11

RF6: El usuario puede visualizar el recurso seleccionado, la primera versión del requerimiento se ve en la figura 4.12.

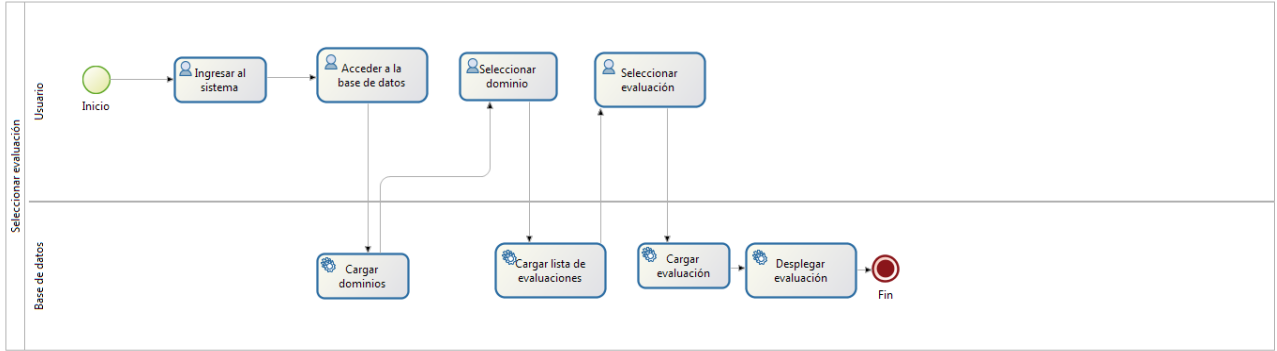


Figura 4. 12 RF6, versión considerada

El requerimiento seis se mantiene, ya que un docente y alumno puede visualizar el elemento que le es permitido.

RF7: El sistema permite al docente modificar evaluaciones, dominios y recursos, la primera versión del requerimiento se aprecia en la figura 4.13, la actualización requerida del mismo se encuentra en la figura 4.14

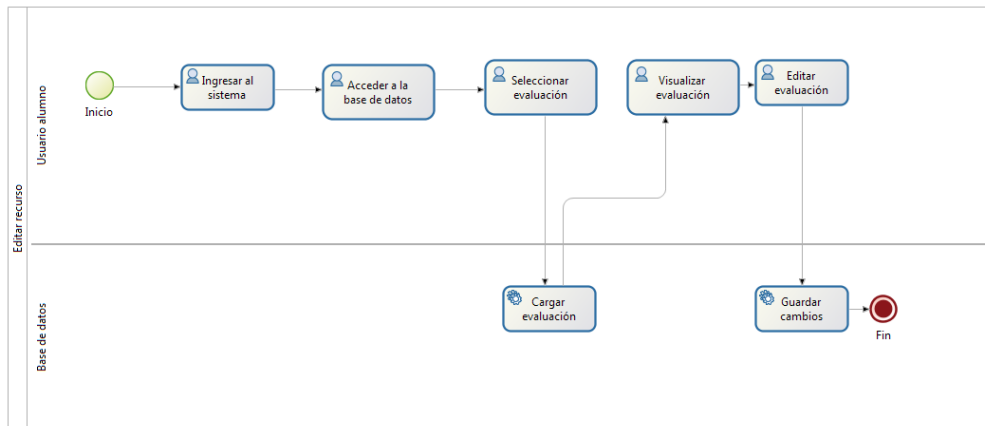


Figura 4. 13RF7, primera versión

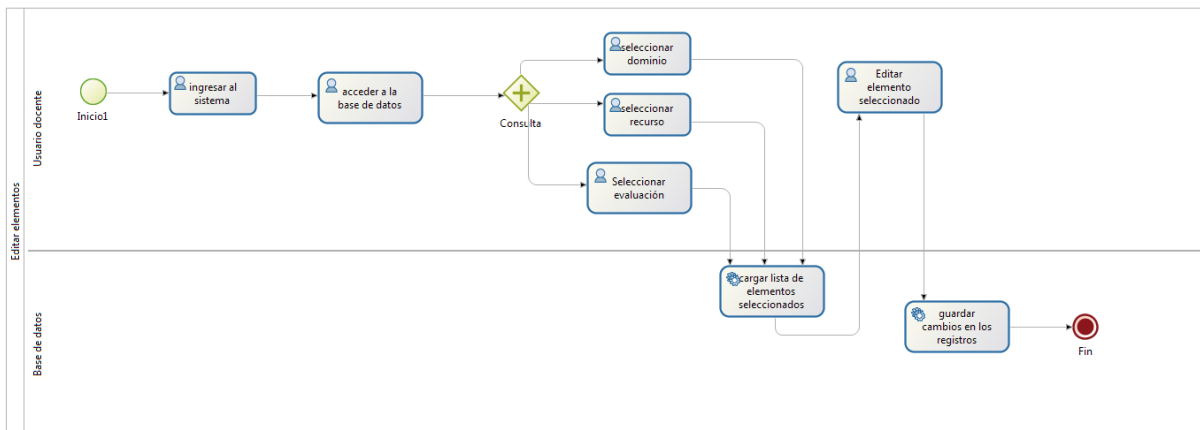


Figura 4. 14 RF7, versión final

Con base en la primera versión del requerimiento el único tipo de usuario, podía hacer la evaluación y modificarla a la vez, en la nueva versión el permiso de editar dominios, evaluaciones y recursos es propio de los docentes como se aprecia en la figura 4.14.

RF8: El sistema permite evaluar los test seleccionados por el usuario, el requerimiento es actualizado como se muestra en la figura 4.15.

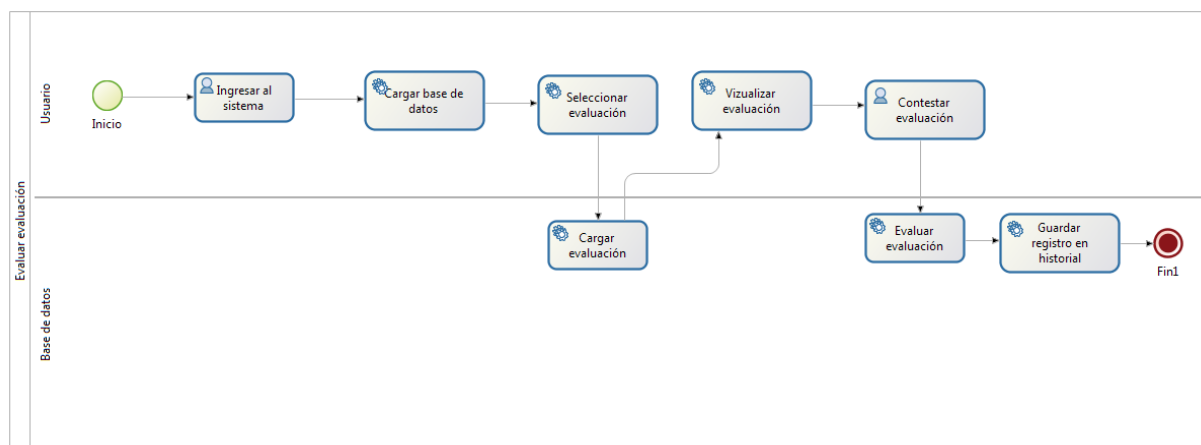


Figura 4. 15 RF8, versión considerada

El requerimiento es actualizado a partir del requerimiento seis, con la selección del dominio, este paso es sobre entendido en la actividad seleccionar evaluación como se aprecia en la figura 1.14.

RF9: El sistema permite al administrador modificar y eliminar las cuentas de usuario, la primera versión del requerimiento se visualiza en la figura 4.16, la nueva versión del mismo se aprecia en la figura 4.17.

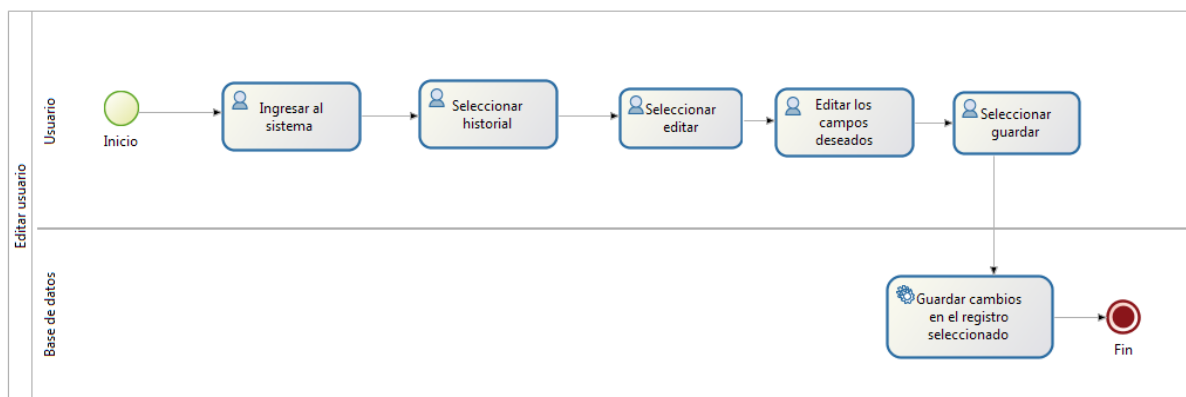


Figura 4. 16 RF9, primera versión

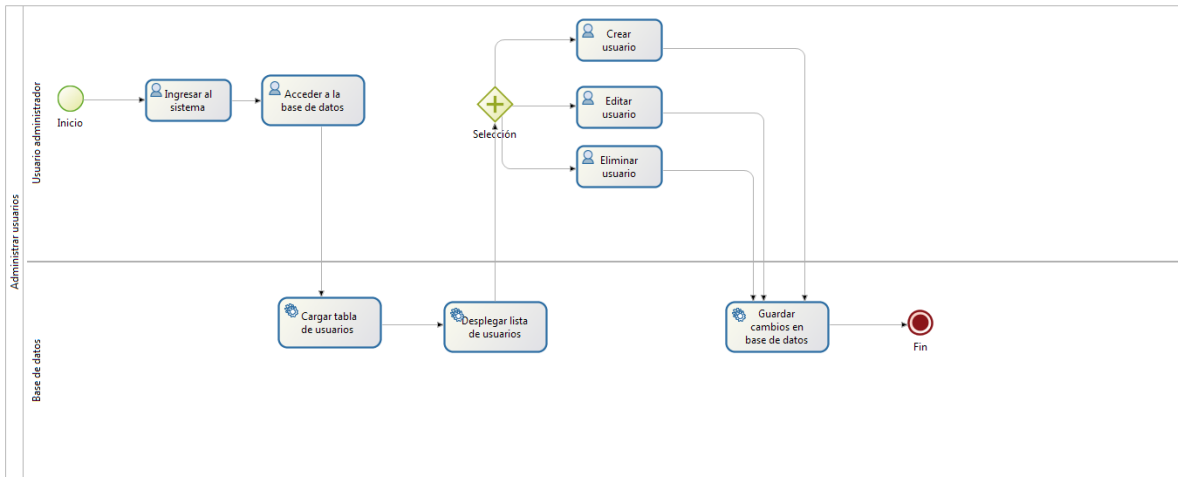


Figura 4. 17 RF9, versión final

Con base en la primera versión del requerimiento, la nueva versión permite solo al administrador del sistema crear, modificar y eliminar cuentas de usuario. Una cuenta también puede ser creada fuera del sistema, como se aprecia en la figura 4.17.

RF10: el usuario alumno solo puede ver su historial como lo muestra la figura 4.18 mientras que el profesor puede consultar todos los historiales, la versión actualizada del requerimiento se visualiza en la figura 4.19.

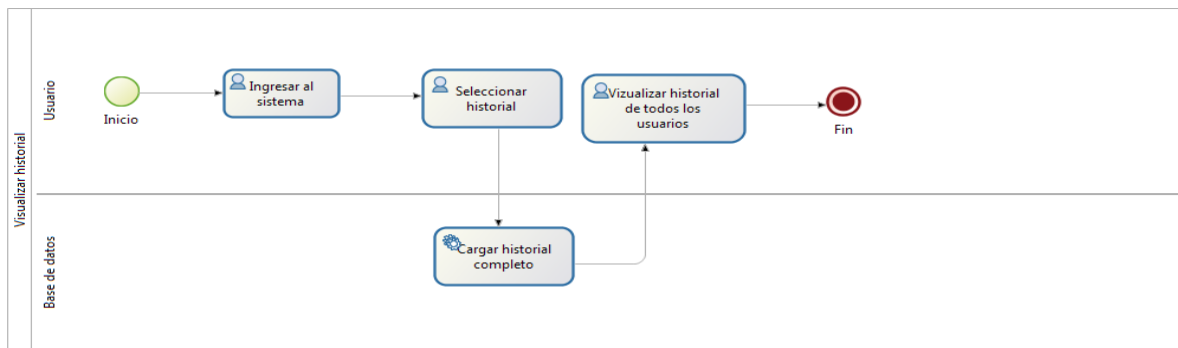


Figura 4. 18 RF10, primera versión

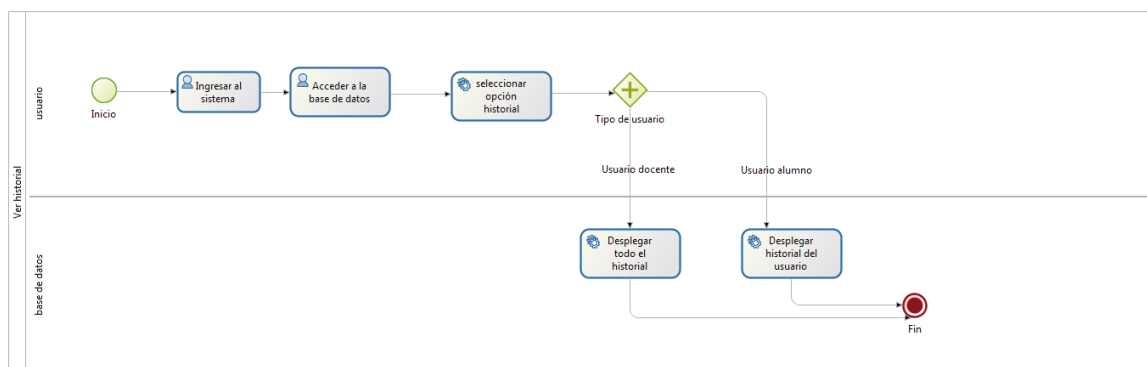


Figura 4. 19 RF10, versión final

Con base en el requerimiento de ver el historial de evaluaciones realizadas es diferente dependiendo del tipo de usuario, el docente puede ver la lista completa, mientras que el alumno solo puede ver su historial, como se aprecia en la figura 4.19.

A continuación, se muestra en la tabla 4.1 con los requerimientos funcionales actuales mapeados con su proceso de negocios, mostrados en las figuras anteriores.

Tabla 4. 1 Relación de nuevos requerimientos funcionales con los procesos de negocio

Requerimiento funcional	Figura
RF1. El usuario podrá ingresar al sistema con una cuenta previamente registrada, accediendo desde el sistema principal	4.3
RF2.El sistema permite la creación de usuarios	4.5
RF3. el sistema permitirá la conexión con distintas bases de datos	4.7
RF4. el alumno puede consultar las evaluaciones mientras que el docente puede consultar los dominios, recursos y evaluaciones	4.9
RF5. el usuario alumno puede elegir una evaluación a partir de un dominio previamente seleccionado	4.11
RF6. El usuario puede visualizar el recurso seleccionado	4.12
RF7. El sistema permite al docente modificar evaluaciones, dominios y recursos,	4.14
RF8. El sistema permite evaluar los test seleccionados por el usuario	4.15
RF9. El sistema permite al administrador modificar y eliminar las cuentas de usuario	4.17
RF10. el usuario alumno solo puede ver su historial mientras que el profesor puede consultar todos los historiales	4.19

4. 3. Analizar arquitectura actual

I

Después del análisis de los requerimientos funcionales a partir de varios modelos de procesos de negocios, se procedió a analizar la arquitectura, dentro de la documentación no se encontró ninguna arquitectura documentada, realizando inspección de código se encontró que se usó la arquitectura Modelo, Vista, Controlador usado con la Tecnología Java EE.

Se encontró en un trabajo de residencias hecho por el estudiante Gibrán *Gutiérrez*, un diagrama de estados para mostrar el flujo del sistema, del siguiente análisis de arquitectura se puede concluir:

Como se puede apreciar, los requerimientos, considerados como componentes se encuentran fuertemente acoplados, ya que, al modificar una clase, se ve estrictamente necesario modificar las clases que hacen uso de la misma.

Así mismo se encontró que no se hace uso de servicios web, aunque la naturaleza del sistema va enfocada al e-learning, se vio la necesidad de migrar la arquitectura a una que haga uso de servicios web.

La base de datos utilizada se ve obligada a reestructurar su modelo por los nuevos requerimientos funcionales.

4.3.1 Diseñar arquitectura general a partir de la arquitectura Actual

Con los nuevos requerimientos planteados, con los objetivos planteados en la propuesta y visualizando la arquitectura del primer componente se plantea una arquitectura que contenga los siguientes elementos:

- Uso de la arquitectura MVC: se plantea usar la misma arquitectura manteniendo el uso de Java Server Pages(JSP), Java Servlets y como capa de persistencia de datos una nueva base de datos hecha en MySQL.
- Uso de servicios Web: como una prioridad en el trabajo de tesis es migrar la arquitectura anterior a una que haga uso de servicios web, para el primer componente se hará uso del protocolo SOAP, para probar la capacidad de integración con dicha tecnología. Cabe mencionar que al usar servicios web se podrá separar el cliente de la funcionalidad.
- Uso de patrón de diseño facade: como el componente forma parte de un subsistema y éste a su vez de un sistema principal, se plantea el patrón de diseño facade para poder acceder a los componentes de una forma sencilla y sin alterarlos.

4.3.2 Arquitectura planteada con servicios web

Antes de plantear la nueva arquitectura basada en servicios web, se plantea el concepto para sustentar su uso. Un servicio web es un componente de la aplicación (programa) puesto a disposición de otros sistemas en la web. [2], una representación básica se muestra en la figura 4.20.



Figura 4. 20 Funcionamiento básico de un servicio web

El uso de una arquitectura orientada a servicios trae muchos beneficios entre ellos se destacan los *siguientes*.

- **Interoperabilidad:** El uso de servicios web permite intercambiar información entre ellos mismos sin importar el lenguaje en el que estén desarrollados e incluso la plataforma en la que se encuentren desplegados, para ellos el formato de comunicación es xml y para restFul xml y Json.

- **Débil acoplamiento:** la baja dependencia entre los componentes desarrollados con servicios web es alta, debido a que su despliegue es independiente no requiere de algún otro componente desplegado.
- **Descomposición modular:** El proceso de romper un sistema en componentes para facilitar el diseño y el desarrollo (Glosario IEEE), es una característica natural en una arquitectura orientada a servicios ya que al tener débil acoplamiento e interoperabilidad es fácil modular y a si mismo integrar.

4.3.3 Arquitectura orientada a servicios implementada con java

En la figura 4.21 se propone un servicio web desplegado en un servidor apache tomcat, el cual es consumido por un cliente, a continuación, se describen los compontes.

1. **JSP y archivos JavaScript y CSS.** Junto con el servlet(2) son la parte que consume el servicio web, el archivo JSP es una página HTML con código java embebido y es la parte visual para el cliente, aquí se despliega el resultado de la petición a un servicio web, los archivos de JavaScript sirven para dar dinamismo a la página y la hoja de estilos(CSS) sirve para organizar y dar presentación a la página web.
2. **Servlet.** Como parte de cliente, funciona como controlador cuando requiera enviar parámetros a un servicio web, a su vez sirve para realizar una llamada al mismo (call web service) y recibir una respuesta a la petición.
3. **Llamada de servicio web (call webs service operation).** Es la llamada específica a una operación de un servicio web el cual se encuentra desplegado en un servidor, el servidor puede ser independiente del cliente o el mismo, las SOA tienen este beneficio.
4. **Cloud.** La nube es el punto intermedio que divide el cliente del servicio, es la red en donde conecta con el servidor el cual puede estar ubicado en cualquier parte.
5. **WSDL (Web Services Description Language).** Es el descriptor del servicio web en esencia, contiene las operaciones y mensajes del servicio web, así como los parámetros y la información necesaria del servicio web, funciona para presentar los tipos de parámetros y las operaciones que pueden ser invocadas.
6. **Web service.** Contiene las operaciones en este caso en la clase especificada, para responder a las peticiones del cliente, los métodos a su vez pueden llamar a otras clases para modular aún más la funcionalidad, en la arquitectura general del sistema se puede ver que el servicio web actúa como patrón de diseño facade ya que la funcionalidad se pone en paquetes separados. Por parte del componente el servicio web se toma también como parte del controlador.
7. **Parte lógica.** La clase del servicio web accede a un paquete específico de acuerdo a la necesidad atendida, dentro de ella se encuentra las operaciones y peticiones a la base de datos, así como la relación con getters y setters de las entidades requeridas.

8. **Paquete de conexión a base de datos.** La funcionalidad contenida en el paquete de conexión contiene la clase que hace la conexión a la base de datos, todos los componentes lógicos hacen uso del componente de conexión.
9. **Capa de persistencia de datos y API JDBC.** La capa más baja de la arquitectura es la base de datos la cual está hecha en MySQL y conecta con la API de Java Database Connectivity.

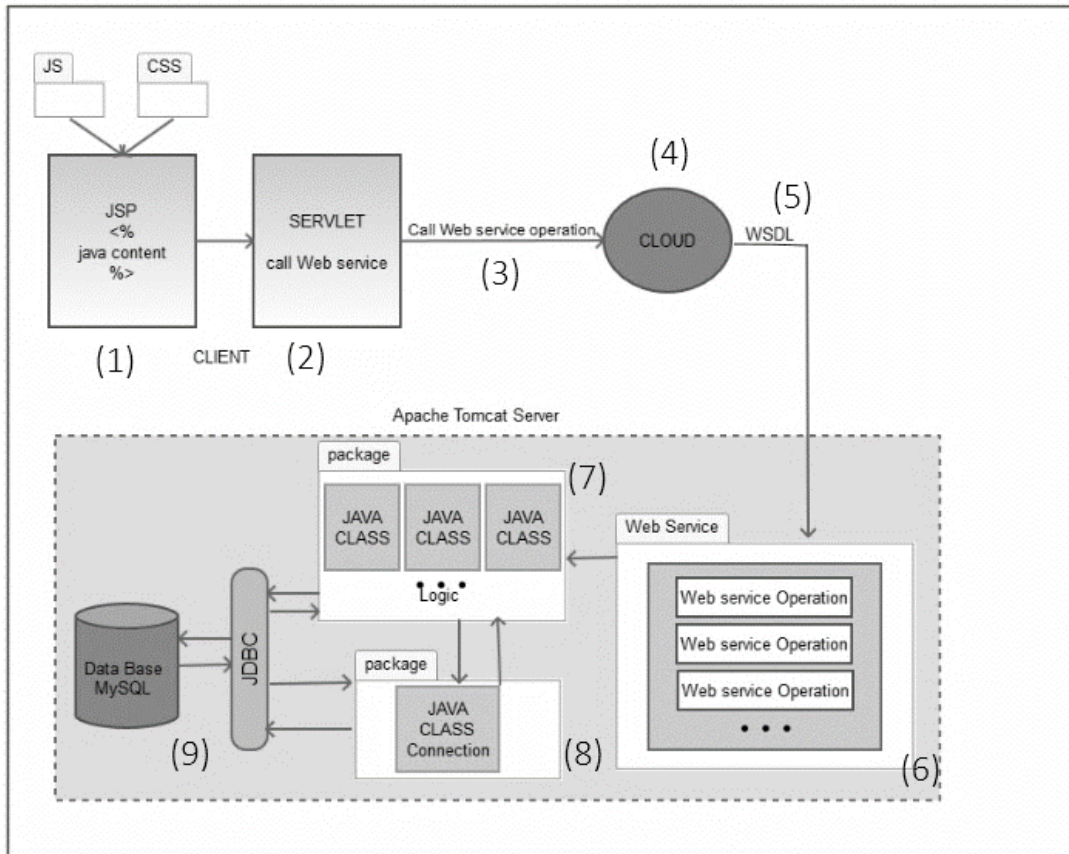


Figura 4. 21 Diagrama de elementos de un servicio web hecho en java

4.5 Implementacion de requerimientos y arquitectura actualizados

La implementación de la tarea 5 comprende un conjunto de actividades que se describen en la figura 4.22

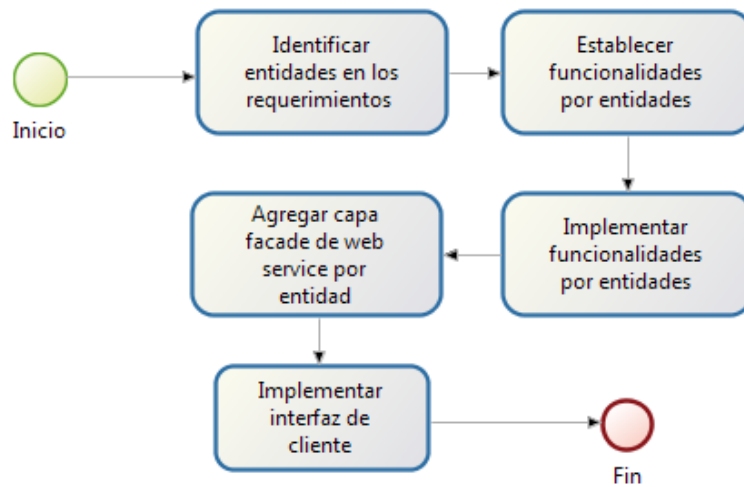


Figura 4. 22 Subproceso del proceso 5

4.5.1 Identificar entidades en los requerimientos

Dentro de la implementación el primer paso es mapear las funcionalidades de acuerdo a entidades, la razón de esto es porque en los trabajos relacionados se encontró que el crear módulos de acuerdo a las entidades se lograban integrar de manera exitosa.

En la figura 4.23 se puede apreciar el módulo de usuarios compuesto por sub componentes que logran su funcionamiento, todos los componentes son autónomos.

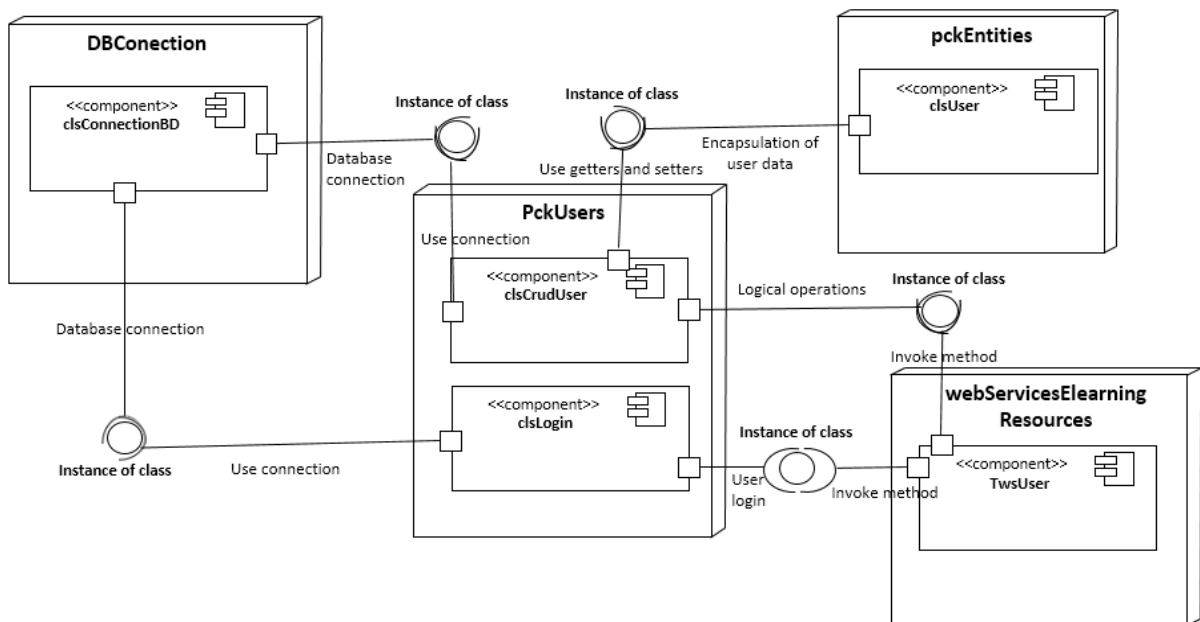


Figura 4. 23 Diagrama de componentes de acuerdo a funcionalidades por entidades

4.5.2 Establecer funcionalidades por entidades

De acuerdo con los requerimientos y siguiendo con el ejemplo del usuario encontramos que las funcionalidades que debe cumplir son las que se muestran en la tabla 4.2:

Tabla 4. 2Funcionalidad por entidad de usuario

Funcionalidades de usuario	
Tipo de usuario	funcionalidad
Administrador	gestionar usuarios con roles

4.5.3 Implementar funcionalidades por entidades

Por lo tanto, los métodos a implementar serían los siguientes, mostradas en la tabla 4.3

Tabla 4. 3 Funcionalidades implementadas de usuario

funcionalidad	Método
Crear usuario	public boolean metCreateUser(String pUser, String pLastName, String pPass, int pType, int pStatus, String pEmail)
Leer usuario	<ul style="list-style-type: none">• public java.sql.ResultSet metReadTableUserComplete()• public String metSetUserValues(int pUserId)
Actualizar usuario	public boolean metUpdateUser(int pId,String pUser, String pLastName, String pPass, int pType, int pStatus, String pEmail)
Dar de baja usuario	public boolean metDeleteUser(int pIdUser)

4.5.4 Agregar capa facade de web service por entidad

Para diseñar los componentes que constituyen el componente general de el sistema de evaluación, es necesario contemplar la arquitectura general de dicho componente, la cual se aprecia en la figura 4.24, como se puede apreciar cada componente es desarrollado de forma independiente y es llamado a través de las llamadas de servicio web y consumidas por un cliente, las cuales son las páginas del sistema, que a su vez es accedido a través del portal, el cual es el sistema integrador.

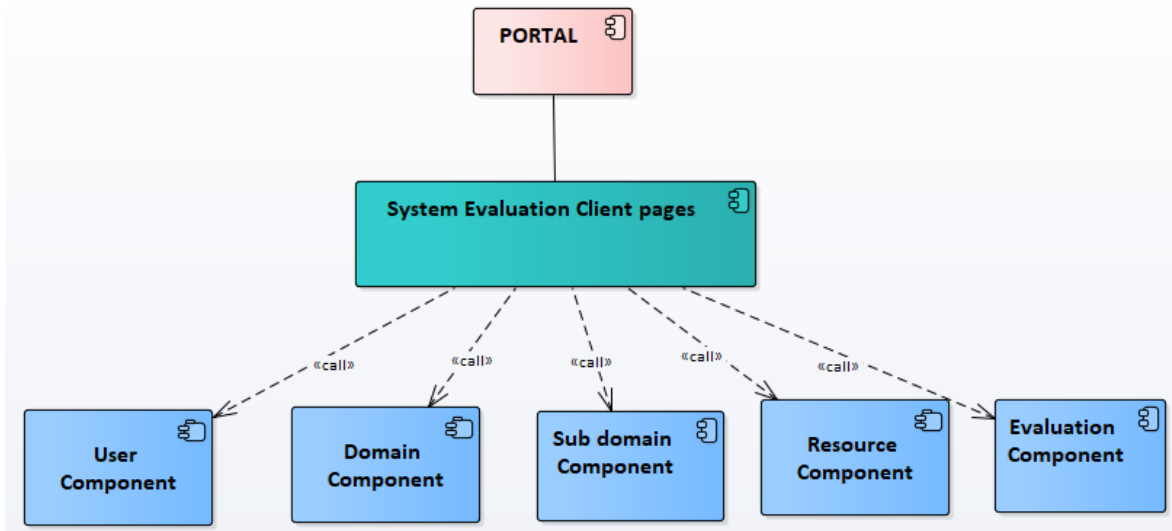


Figura 4. 24 Arquitectura general del componente: Sistema de evaluación

Una vez obtenidas e implementadas las funcionalidades se procede a crear el servicio web pero como fachada, es decir, solo llamando a las funcionalidades, para diseñar la estructura de la arquitectura a un nivel de componente, se debe usar diagramas de clases, a continuación, se muestran cada entidad con su respectiva arquitectura en diagrama de clases.

4.5.4.1 Arquitectura del sub componente Usuario

Una de las entidades identificadas en el componente 1, sistema de evaluaciones, es el usuario. Como se puede apreciar en la figura 4.25, las clases utilizadas en el componente tienen la palabra “user” para identificarse dentro del sistema, cabe mencionar que todas las funcionalidades se encuentran separadas por paquetes dentro del proyecto.

Haciendo un análisis de la arquitectura desde la parte más baja, se encuentra el componente de la conexión de la base de datos, el cual se comporta a su vez como un sub componente independiente el cual usan los demás componentes separados por entidad.

La clase `clsCrudUser` hace uso del sub componente de conexión a la base de datos y de la clase `clsUser`, con el fin de gestionar las funcionalidades relacionadas con el usuario dentro del sistema de evaluaciones, mientras la clase `clsLogin` encontrada en el mismo paquete de la entidad usuario, realiza la función de validar la entrada de un usuario al sistema, hace uso de la conexión a la base de datos. Por último, se encuentra el servicio web como una fachada el cual invoca todas las funcionalidades encontradas en las clases `clsCrudUser` y `clsLogin`, y actúa como una capa para estar disponible para cualquier cliente a través del WSDL del servicio.

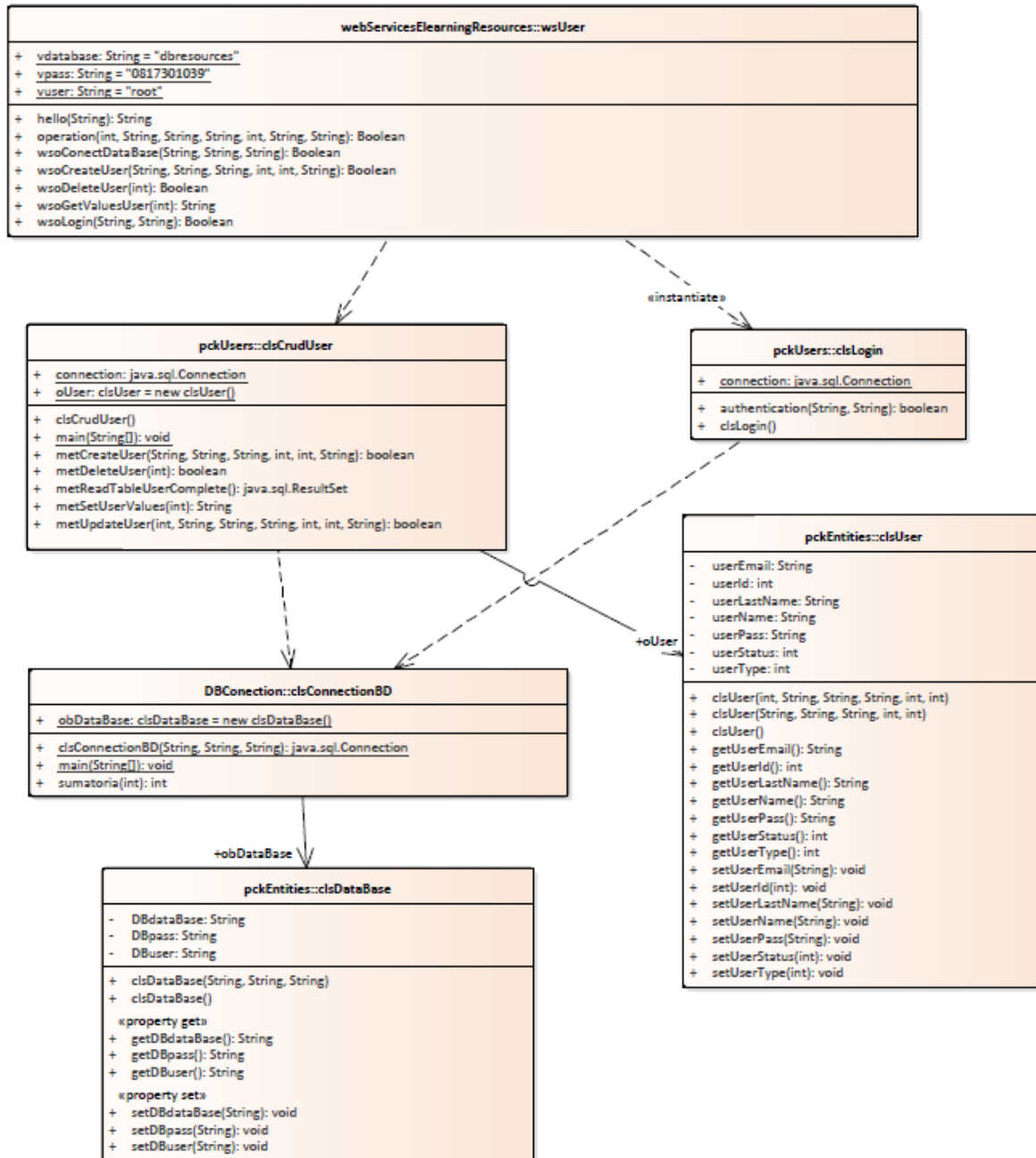


Figura 4. 25 Arquitectura en diagrama de clases del sub componente usuario

4.5.4.2 Arquitectura del sub componente Dominio

Otra de las entidades identificadas en el componente 1, sistema de evaluaciones, es el dominio. Dicho sub componente que se muestra en la figura 4.26, mantiene la misma relación de funcionalidad de la figura 4.25, la clase clsCrudDomain hace uso de las clases de clsConnectionBD y clsDomain para gestionar las actividades acordes a los dominios. Y por último se agrega la capa de fachada del servicio web denominado wsDomain.

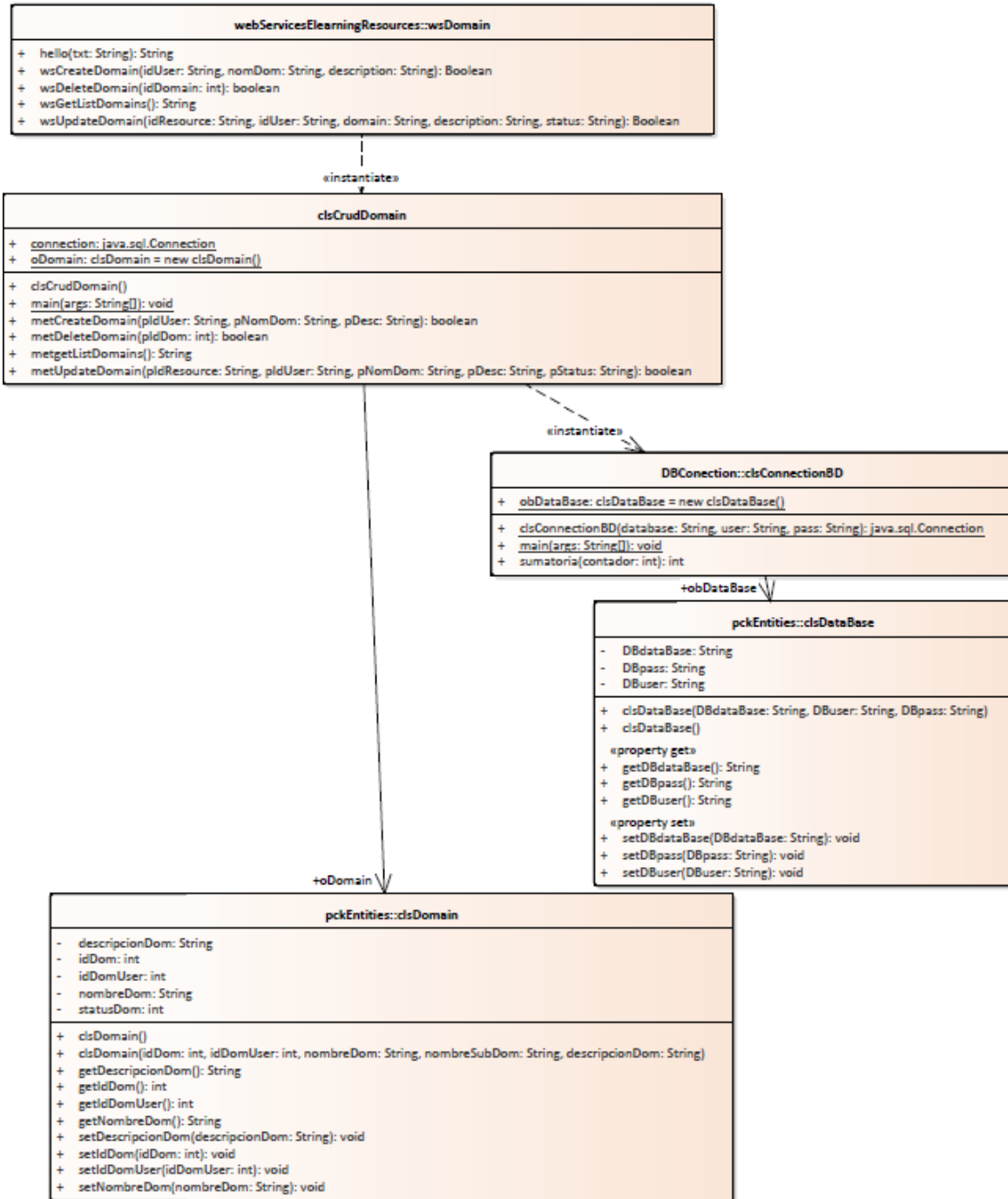


Figura 4. 26 Arquitectura en diagrama de clases del sub componente dominio

4.5.4.3 Arquitectura del sub componente Sub Dominio

Otra entidad identificada en el componente 1, sistema de evaluaciones, es el sub dominio. Dicho sub componente que se muestra en la figura 4.27, mantiene la misma relación de funcionalidad de la figura 4.25, las funcionalidades son casi idénticas, debido a que la

estructura de un dominio y un sub dominio son iguales. Por último, se agrega la capa de fachada, denominada wsSubDomain.

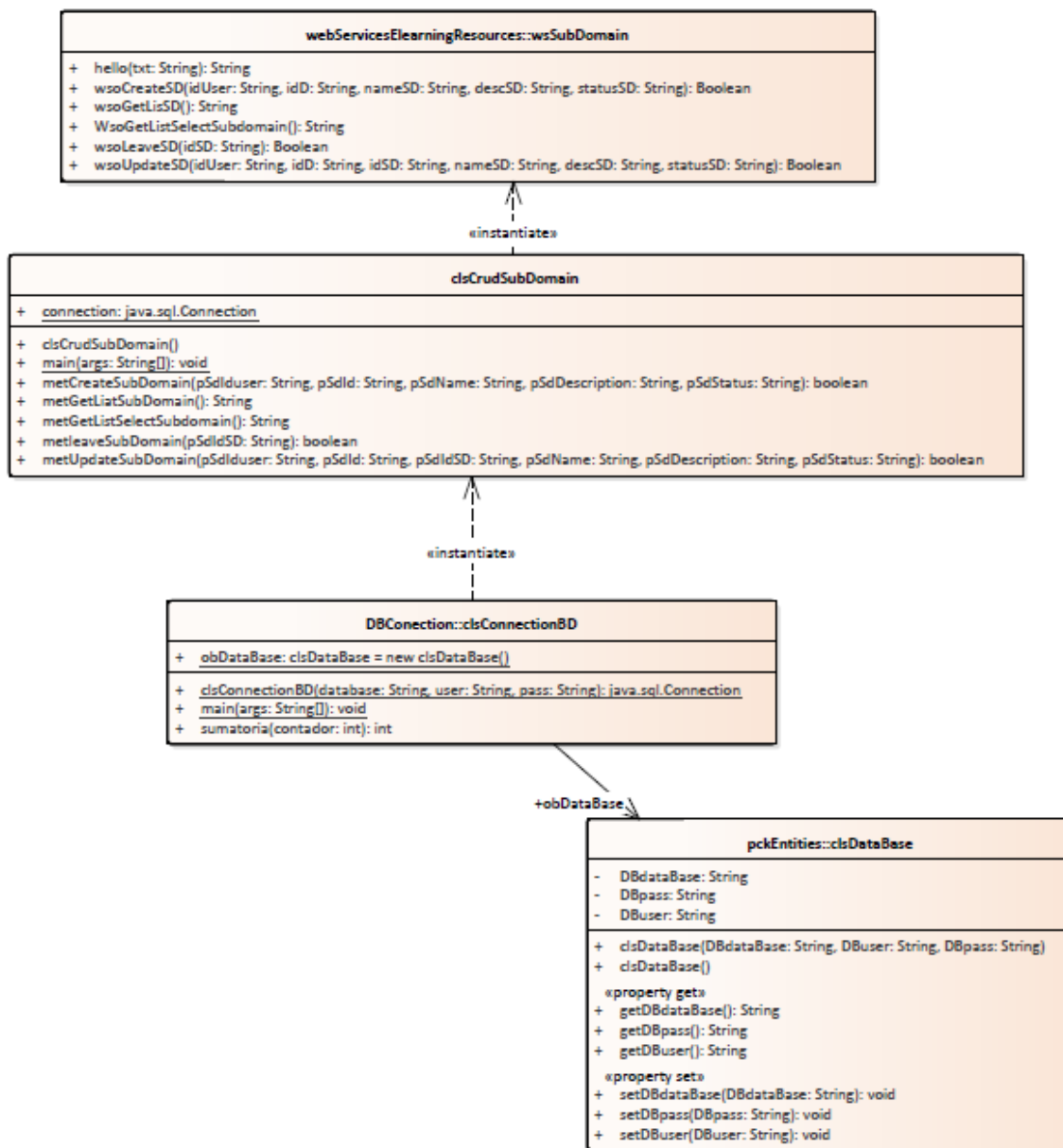


Figura 4. 27 Arquitectura en diagrama de clases del sub componente sub dominio

4.5.4.4 Arquitectura del sub componente recurso

Una de las entidades identificadas en el componente 1, sistema de evaluaciones, es el recurso. Dicho sub componente que se muestra en la figura 4.28, mantiene la misma relación de funcionalidad de los sub componentes anteriormente mencionados. Un recurso pertenece a un sub dominio, la clase clsCrudResources hace uso de las clases clsConnectionBD y

clsResource, para gestionar las operaciones de un recurso, por último, se agrega la fachada de servicio web, denominado wsResources.

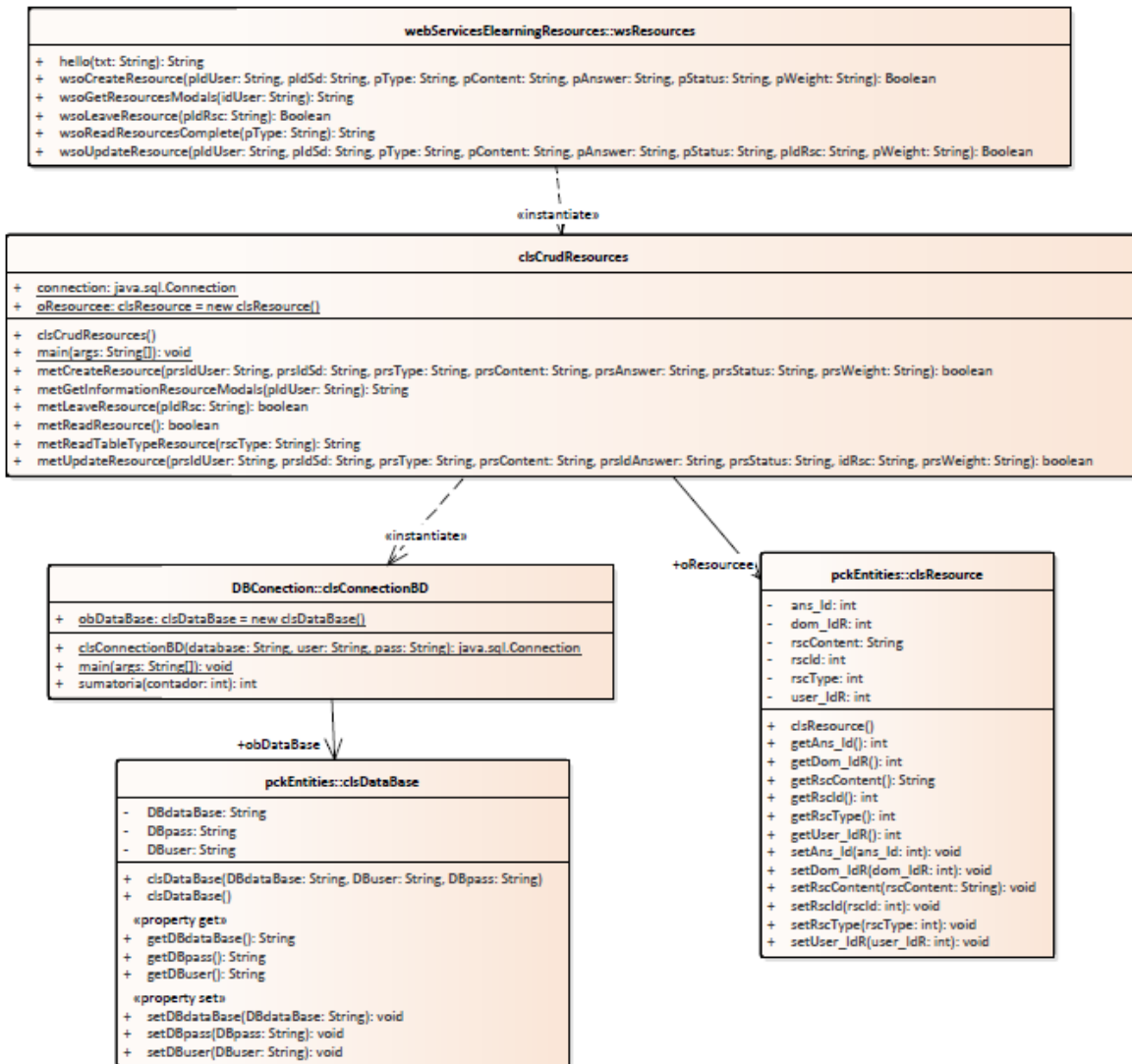


Figura 4. 28 Arquitectura en diagrama de clases del sub componente recurso

4.5.4.5 Arquitectura del sub componente Evaluación

La última entidad identificada en el componente 1, es la evaluación. Dicho sub componente que se muestra en la figura 4.29, mantiene la misma relación de funcionalidad de los sub componentes anteriormente mencionados. Una evaluación está compuesta de recursos que son de un sub dominio, el cual pertenece a un dominio existente.

La clase clsCrudEvaluation accede a las clases de clsConnectionBD para gestionar las operaciones de evaluación. Este sub componente no posee la clase de clsEvaluation, ya que

no se consideró necesaria para el funcionamiento del mismo. Por último, se integra la capa de servicio web wsEvaluation a modo de fachada para generar el sub componente solicitado.

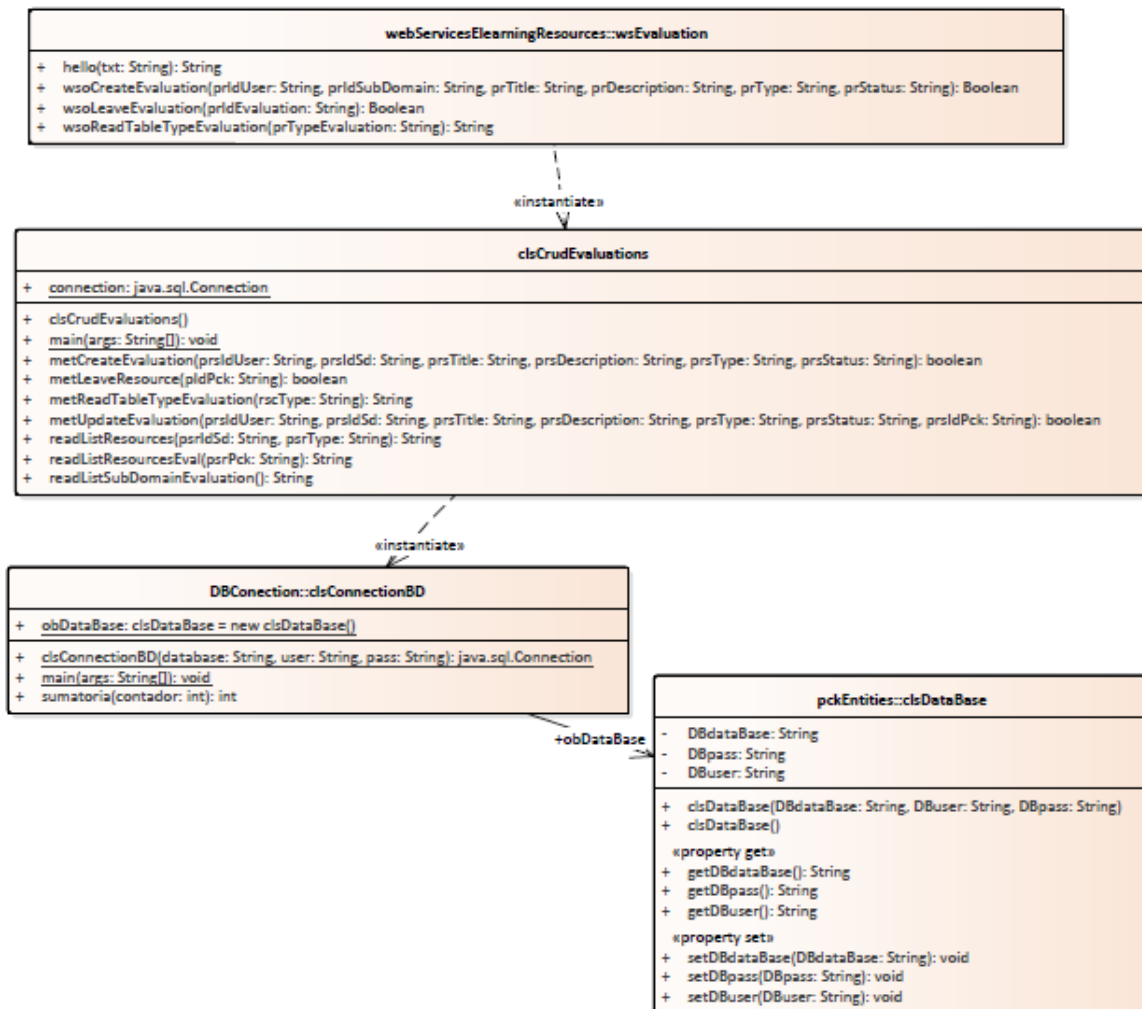


Figura 4. 29Arquitectura en diagrama de clases del sub componente Evaluación

5

PRUEBAS REALIZADAS

La prueba de toda
verdad reside,
sencillamente, en su
eficacia.

- William James

En este capítulo se describen las pruebas realizadas a los componentes a integrar, así como su análisis y conclusiones.

Capítulo 5 Pruebas

5.1.1 Introducción

En el siguiente capítulo se detallan y explican las pruebas hechas a la solución implementada para el sistema integrador de componentes. En el primer componente, del componente del modelo de calidad para servicios web de aprendizaje, se realizan las pruebas sobre una arquitectura con la arquitectura propuesta ya implementada. En el segundo componente, el sistema evaluador de recursos, se realizaron las pruebas sobre un sistema con necesidad de reestructuración, para una arquitectura basadas en servicios web.

5.1.2 Objetivo de las pruebas

El objetivo de las pruebas descritas en este capítulo es comprobar como a partir de una metodología de solución planteada e implementada se pueden solucionar los problemas de integración de los componentes de gestión de servicios web de aprendizaje para su despliegue independiente y en conjunto dentro del sistema integrador.

5.1.3 Procedimiento de pruebas

A continuación, en la figura 5.1 Se describen los pasos generales para ejecutar y describir las pruebas a los dos componentes.

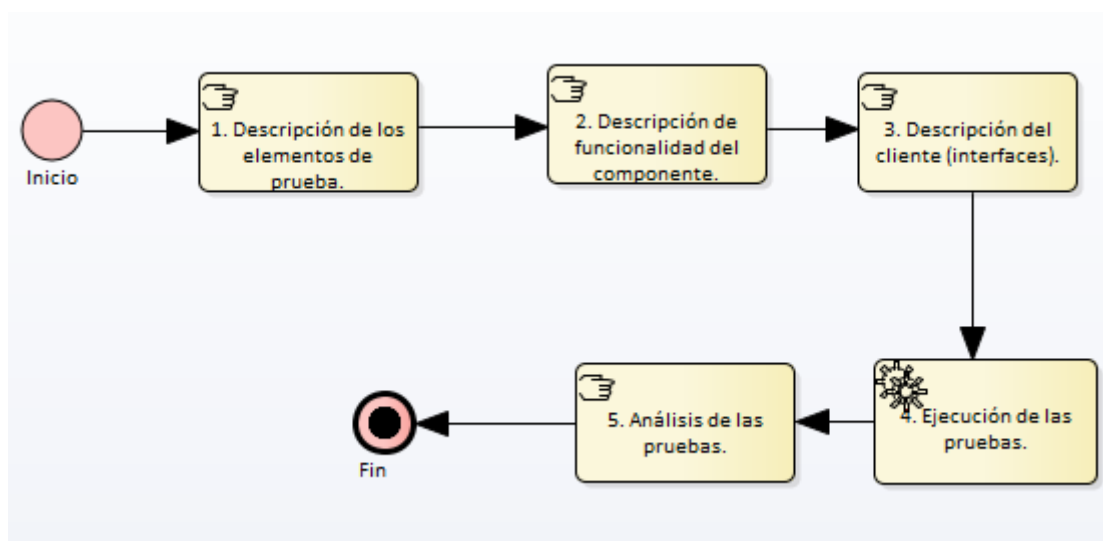


Figura 5. 1 Pasos para el desarrollo y análisis de las pruebas del sistema integrador

5.1.4 Descripción de los elementos de prueba.

Los elementos de prueba se dividen en dos conjuntos de servicios web, el primer conjunto (véase tabla 5.1) son servicios web de aprendizaje que se usaron para probar el componente del modelo de calidad de servicios web de aprendizaje. El segundo conjunto (véase tabla 5.2) se basa en los servicios web integrados para el sistema de evaluaciones.

Tabla 5. 1 Servicios web de aprendizaje para el componente del modelo de calidad para servicios web de aprendizaje

URL del servicio web de aprendizaje	Descripción
http://192.168.8.216/LWSEvaluate/lwsdoc.php?wsdl	Servicio web de aprendizaje con contenido en formato .doc
http://192.168.8.216/LWSEvaluate/lwsdocx.php?wsdl	Servicio web de aprendizaje con contenido en formato .docx
http://192.168.8.216/LWSEvaluate/lwspdf.php?wsdl	Servicio web de aprendizaje con contenido en formato .pdf
http://192.168.8.216/LWSEvaluate/lwsppt.php?wsdl	Servicio web de aprendizaje con contenido en formato .ppt
http://192.168.8.216/LWSEvaluate/lwspptx.php?wsdl	Servicio web de aprendizaje con contenido en formato .pptx
http://192.168.8.216/LWSEvaluate/lwshtml.php?wsdl	Servicio web de aprendizaje con contenido en HTML
http://192.168.8.216/LWSEvaluate/undefinide	Elemento que no corresponde a ningún servicio web de aprendizaje

La tabla 5.1 muestra la información de los elementos de prueba para el componente del modelo de calidad para servicios web de aprendizaje, los cuales son servicios web de aprendizaje, que serán medidos. Debido a que el propósito de las pruebas es comprobar la integración con débil acoplamiento, el cliente y el componente se encuentran físicamente separados, en el diagrama de despliegue 5.2 se muestra a detalle dicha información.

- Los servicios web de aprendizajes están desarrollados sobre el protocolo SOAP en el lenguaje de PHP en su versión 7.09.
- El software de despliegue de servicios es XAMMP en su versión 3.2.2

Los elementos de la tabla 5.2 muestran los servicios web que conforman los componentes del sistema de evaluaciones, dichos servicios serán probados en distintos casos.

Tabla 5. 2 Servicios web del sistema de evaluaciones e-learning

URL del servicio web de aprendizaje	Descripción
http://localhost:8084/Elearning_backEnd/wsUser?wsdl	Servicio web para obtener las operaciones de gestión de usuarios.
http://localhost:8084/Elearning_backEnd/wsDomain?wsdl	Servicio web para obtener las operaciones de gestión de dominios.
http://localhost:8084/Elearning_backEnd/wsSubDomain?wsdl	Servicio web para obtener las operaciones de gestión de sub dominios.
http://localhost:8084/Elearning_backEnd/wsResources?wsdl	Servicio web para obtener las operaciones de gestión de recursos.
http://localhost:8084/Elearning_backEnd/wsEvaluation?wsdl	Servicio web para obtener las operaciones de gestión de evaluaciones.

Se consideran elementos de prueba a los servicios web de cada componente ya que actúan como fachada (patrón de diseño facade) y acceden a la funcionalidad de cada componente, almacenada en paquetes.

Al igual que el componente del modelo de calidad de servicios web de aprendizaje, el objetivo es probar la integración de dichos servicios a través de un único sistema, incluso es posible desplegar un servicio de forma independiente para el uso que fue destinado, por citar un ejemplo puede ser una conexión independiente a una base de datos o el registro de un usuario, fuera del sistema.

5.2. Descripción de funcionalidad del componente

5.2.1 Localización física de componentes del sistema integrador.

En la figura 5.2 se puede apreciar la distribución física de los componentes que componen el sistema integrador de componentes de gestión de servicios web de aprendizaje, como se recuerda, el propósito de las pruebas es comprobar la integración de los componentes de gestión de SWA.

- El equipo de HP Elite book, con la dirección ip 192.168.8.76 contiene dentro del servidor Apache Tomcat los componentes del portal y el cliente, el cual contiene todas las vistas que consumen los servicios web ubicados en el nodo de Web services back end, dicho nodo contiene toda la funcionalidad separa por entidades, encargada de realizar las operaciones lógicas con la base de datos, el nodo cuenta con la capa de

servicios web, usada como una fachada para comunicarse con los consumidores (clientes). Como último nodo del equipo, se encuentra la base de datos, la cual actúa como la capa de persistencia de datos, y en el caso del sistema de evaluaciones, guarda los datos de los usuarios, dominios, subdominios, evaluaciones, entre otros.

- El equipo HP Pavilion, con la dirección ip 192.168.8.216 contiene un servidor cliente Apache para desplegar clientes, sobre el lenguaje PHP, dicho cliente ha sido cambiado al de la versión con las que se elaboraron las pruebas descritas en el trabajo de tesis de Honorio Salinas, para propósitos de prueba de interoperabilidad e integración con el sistema actual. El nodo del servidor contiene dos componentes, un cliente llamado LWS client que consume el servicio web denominado WebServiceSoap y otro cliente llamado LWS2 client el cual consume el servicio web denominado WebServiceSoap2.
- El equipo HP All in One, con la dirección ip 192.168.8.231 el cual contiene un servidor en Apache, para desplegar servicios hechos en lenguaje PHP, el cual contiene dos componentes. El componente denominado WebServiceSoap tiene la función de medir la calidad del contenido de un servicio web de aprendizaje, a partir de la dirección del WSDL, mientras que el componente denominado WebServiceSoap2 tiene la función de medir la funcionalidad del contenido de un servicio web de aprendizaje a partir de parámetros establecidos por el usuario.

Como se puede apreciar en la figura 5.2, la distribución física de componentes muestra la integración establecida en el alcance del trabajo.

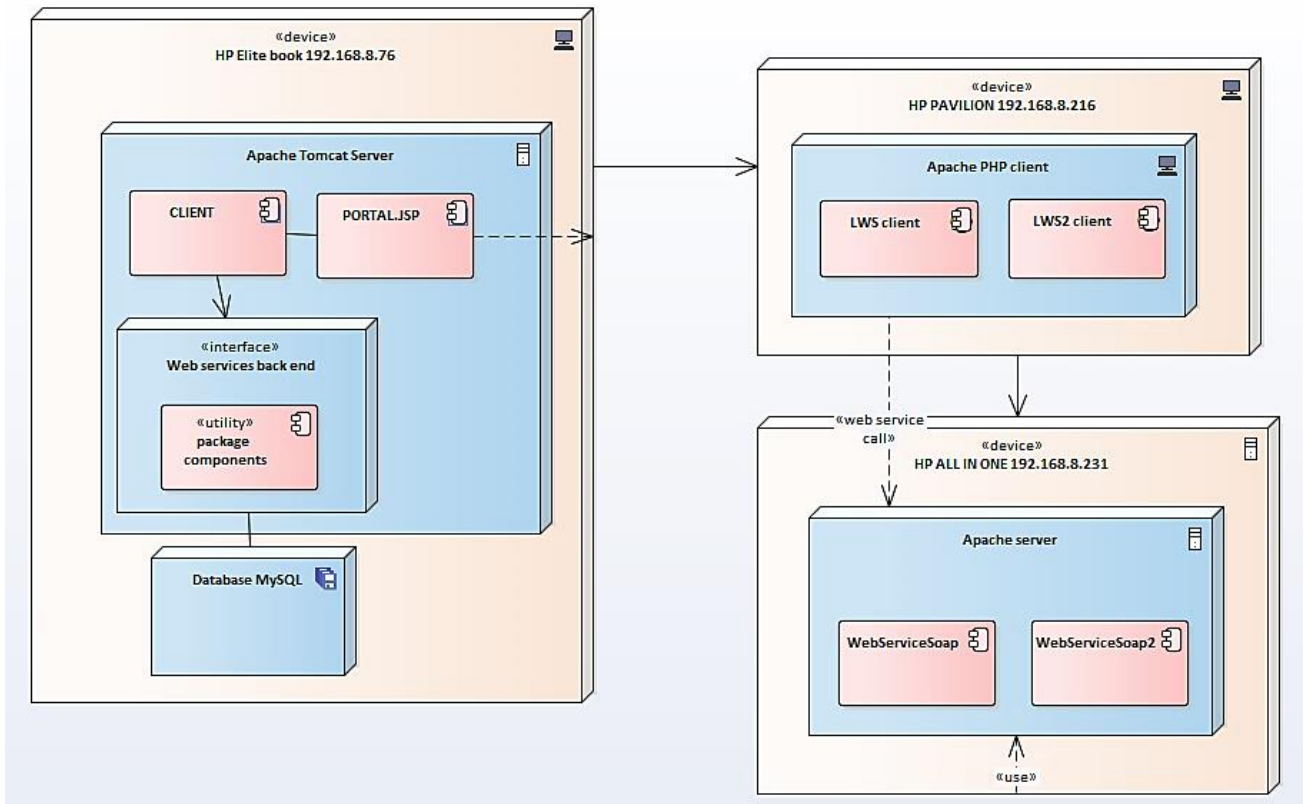


Figura 5. 2 Diagrama despliegue del sistema integrador de componentes

5.3. Descripción del cliente, interfaces.

El sistema integrador de componentes está hecho bajo una arquitectura de servicios web, por lo tanto, se necesitan clientes que consuman los servicios dentro del sistema, los clientes web que a continuación se describen, fueron desarrollados con las siguientes tecnologías:

- Framework css materialize. Es un framework creado por colaboradores de Google para dar atributos responsivos y con diseño de rejilla a los clientes. También cuenta con el tema de material design para proporcionar una vista agradable al usuario.
- Paquete de Google icons. Iconos creados por Google para brindar una mejor experiencia en la navegación del usuario.
- Jquery-2.1.1 y Jquery-2.1.4. para proporcionar una experiencia de contenido dinámico, se incorporó el framework de Jquery que proporciona las animaciones de contenido y el uso de modals y contenido web.

5.3.1 Portal

La vista principal del sistema será el cliente llamado PORTAL.jsp, dicho cliente es encargado de agregar los componentes que se irán agregando poco a poco a través de un enlace, sin necesidad de hacer cambios a los dos componentes ya integrados.

En el primer componente, el sistema de evaluaciones se encuentra un botón llamado deploy componente, el cual dirige al sistema seleccionado, el segundo componente, contiene dos botones como lo muestra la imagen 5.3, el cual despliega los componentes de LWS y LWS2, ubicados físicamente en otro equipo de cómputo, como lo describe la figura 5.3.

Con propósito de probar la integración de algunos componentes del sistema de evaluación, se crearon clientes independientes para consumir dichos servicios.

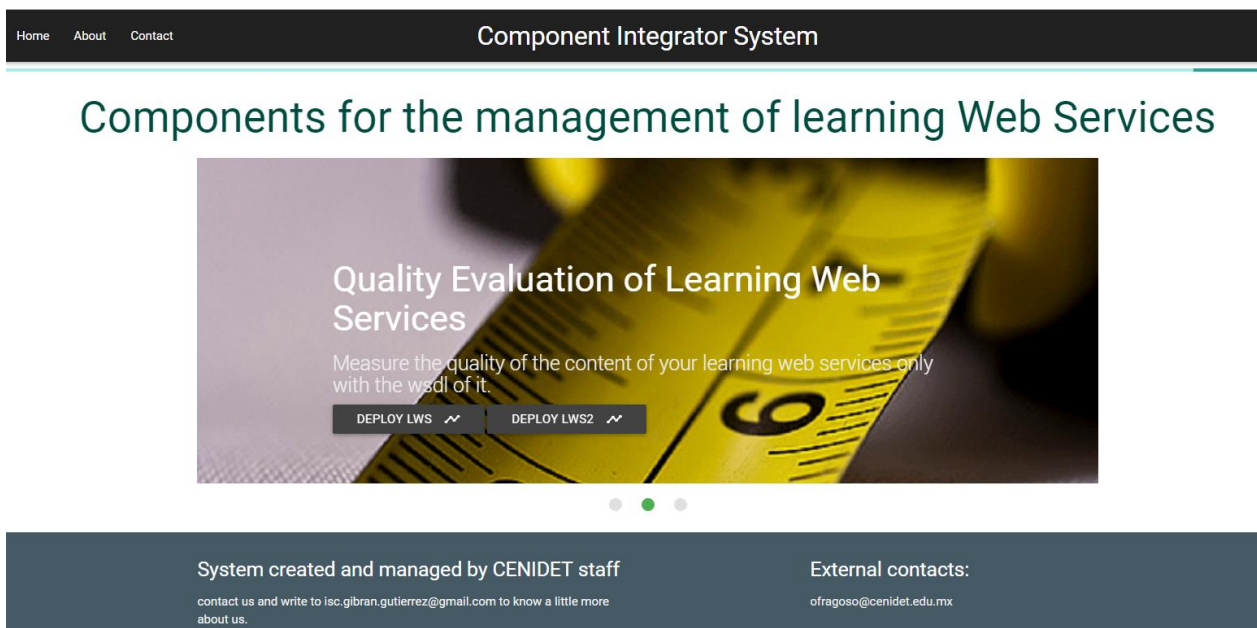


Figura 5. 3 Vista del portal del sistema integrador de componentes

5.3.2 Clientes del sistema de evaluación.

5.3.2.1 Clientes del componente del modelo de calidad para servicios web de aprendizaje

Como se aprecia en el diagrama de la figura 5.3, la ubicación física de los clientes del componente del modelo de calidad se encuentra alojados en el equipo HP Pavilion, con el fin de probar la integración a través de nodos físicos entre los componentes, ya que dicho equipo cuenta con una versión distinta de sistema operativo e incluso el cliente está construido en el lenguaje PHP sobre un servidor distinto al del portal principal, a continuación, se describen los dos clientes.

5.3.2.2 Cliente LWS

El primer cliente, se denomina LWS, es una página web construida en PHP la cual consume el servicio web denominado webServiceSoap, dicho servicio web tiene la funcionalidad de brindar la reusabilidad de un SWA a través de la dirección WSDL. El cliente LWS es distinto al que utilizaron para las pruebas del trabajo del modelo de calidad de SWA, dicho cliente está hecho para funcionar de forma independiente y con una nueva apariencia, el cual envía el WSDL del SWA a procesar, y como resultado, obtiene un porcentaje de calidad del servicio web de aprendizaje, para más detalle de las pruebas, véase el análisis de resultados.



Figura 5. 4 Interfaz del cliente LWS

5.3.2.3 Cliente LWS2

El cliente denominado LWS2, es una página web construida en lenguaje PHP la cual consume el servicio web denominado webServiceSoap2, el cual se ilustra en la figura 5.5 cuenta con las mismas especificaciones que el LWS, solo que en el LWS2 se pueden fijar una serie de parámetros para realizar medidas de acuerdo con la solicitud especificada por el usuario. El resultado de retorno es un porcentaje que indica la reusabilidad del contenido del SWA. De igual forma, el cliente fue reconstruido y distribuido en un nodo físico distinto para probar la integración a través de varios equipos de cómputo.

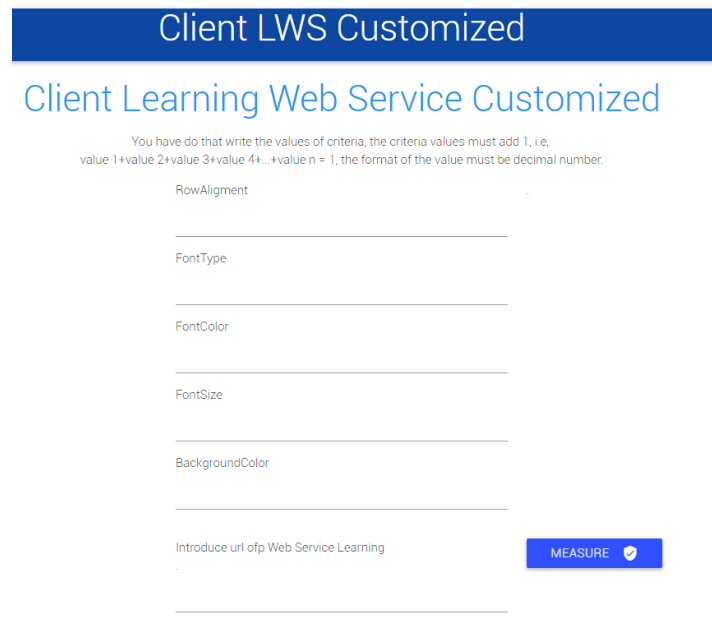


Figura 5. 5 Interfaz del cliente LWS2

5.3.3 Clientes de Sistema de evaluaciones.

Los clientes hechos para el sistema de evaluaciones, fueron creados para consumir los servicios web que se encuentran en el mismo servidor, las funcionalidades son producto de

peticiones a las operaciones que están agrupadas de acuerdo a la entidad de la que trata la petición.

5.3.3.1 Conexión a base de datos

El cliente hace una petición enviando el nombre de la base de datos, el usuario y la contraseña, como resultado obtiene una variable booleana que confirma la conexión y guarda los datos para la conexión, dicha conexión sirve para conectar cualquier base de datos.

The image shows a web interface for connecting to a database. At the top, there is a blue header with the word "Connection". Below this, the main heading is "Connect a database". The central part of the interface is a light gray box containing the text "Select a database to start" in green. Below this text are three input fields: "Data Base" (with a hamburger menu icon), "User" (with a person icon), and "Password" (with a key icon). Below the input fields, there is a line of text: "You can use any database to get the resources you need". At the bottom of the interface, there is a blue footer with the text "Centro Nacional de Investigación y Desarrollo Tecnológico" on the left and "About us" on the right.

Figura 5. 6 Interfaz de cliente de conexión a base de datos

5.3.3.2 Login

El ingreso del usuario, que se presenta en la figura 5.7, consiste en enviar el correo electrónico con el que fue registrado y su respectiva contraseña. El valor de retorno es una variable booleana y de ser correcto el ingreso, los datos del usuario son guardados en un bean¹¹ propio del usuario.

¹¹ Bean es una clase destinada a almacenar una cantidad de datos.

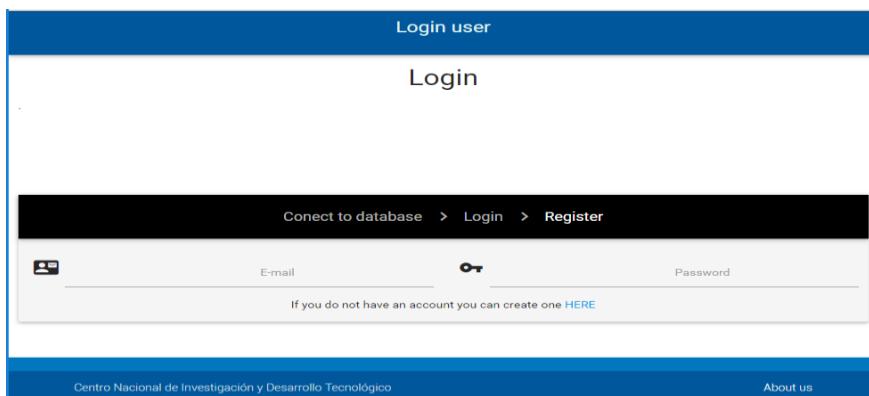


Figura 5. 7 Interfaz de cliente de ingreso de usuario

5.3.3.3 Manejador de recursos

Una interfaz que proporciona al usuario autorizado, el poder crear recursos, crear dominios y sub dominios, y generar evaluaciones a partir de estos.



Figura 5. 8 Interfaz de gestión de dominios, recursos y evaluaciones

5.3.3.4 Panel de administrador

El panel de administrador, que se muestra en la figura 5.9, permite sólo al administrador realizar las operaciones de visualizar, crear, editar y eliminar a los usuarios que desee, cabe mencionar que todas las operaciones mencionadas son ejecutadas bajo el servicio web de usuario. En otro cliente se usa sólo la función de crear usuario para comprobar integración del componente en otras funcionalidades.

Id	Name	Last Name	Password	Type	Status	E-mail
456	Alvin	Eclair	0912	1	2	unemail@dsd
1	Gibran	Gutierrez	0817301039	3	1	isc.gibran.gutierrez@gmail.com
2	Olivia	Graciela	1232	2	1	ofragoso@gmail.com
3	Rene	Santaolaya	9843	2	1	santaolaya@gmail.com

Create User

Insert Name

Insert Last Name

Insert Password

Select Rol: Student

Select Status: Active

Insert E-mail

CREATE **CLEAN**

Update user

Select ID: 1 Gibran Gutierrez

Insert Name

Insert Last Name

Insert Password

Select Rol: Student

Select Status: Active

Insert E-mail

UPDATE **CLEAN**

Delete user

Delete User: 1

DELETE USER

Figura 5. 9 Interfaz de cliente de administrador de usuarios

5.3.3.5 Gestión de dominios

La gestión de dominios permite visualizar y listar los dominios disponibles por los usuarios autorizados, la creación y actualización de los mismos se describe en la figura 5.10

The screenshot shows a web browser window with the URL 'localhost:8084/Elearning_frontEnd/domain.jsp#'. The page title is 'Domains and Subdomains' and it features 'CREATE DOMAIN' and 'CREATE SUBDOMAIN' buttons. The main content is titled 'Management of Domains and Subdomains' and displays a list of four domains, each with an 'EDIT' and 'REMOVE' button.

Id domain	Name	Description	Status	Actions
Id domain: 1	Software Engineering update	Software engineering is the practical application of scientific knowledge to the design and construction of computer programs and associated documentation required to develop, operate and maintain them	active	EDIT, REMOVE
Id domain: 2	Web programming	Web development is a term that defines the creation of websites for the Internet or an intranet.	active	EDIT, REMOVE
Id domain: 3	Programming logic	Fundamentals of programming and pseudocode	active	EDIT, REMOVE
Id domain: 4	computer networks	Logical and physical configurations of computer networks	active	EDIT, REMOVE

Figura 5. 10 Interfaz de cliente de gestión de dominios

5.3.3.6 Gestión de subdominios

Un dominio, el cual se presenta en la 5.11, puede tener varios subdominios, los cuales son creados y poseen las mismas funciones de gestión que un dominio, la interfaz muestra su listado y la creación y actualización de los mismos.

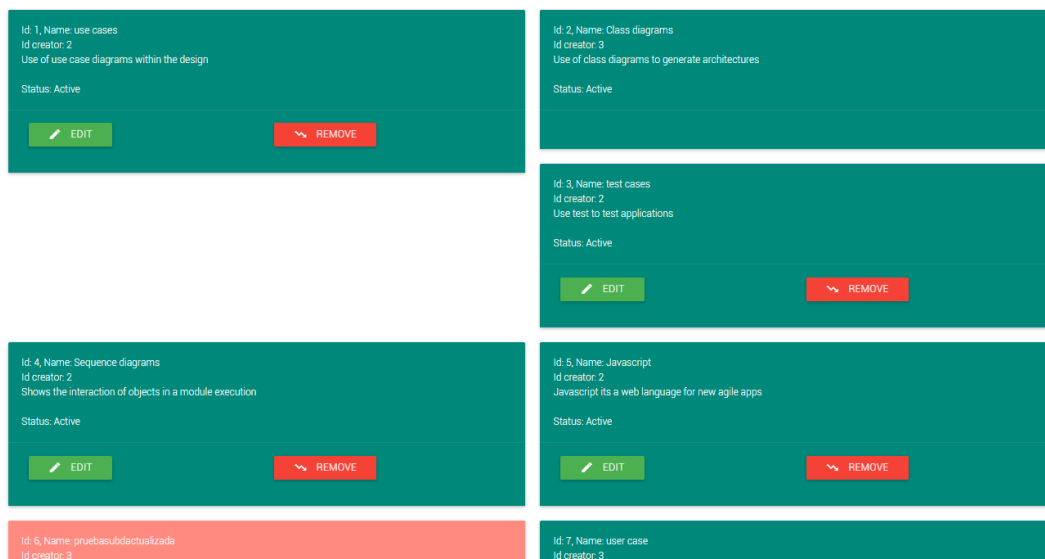


Figura 5. 11 Interfaz de cliente de gestión de sub dominios

5.3.3.7 Gestión de recursos

La interfaz de gestión de recursos, como se aprecia en la figura 5.12, contiene los elementos de la evaluación generados a partir de un subdominio específico. El cliente descrito es encargado de crear, visualizar, editar y dar de baja un recurso a través de una operación de servicio web.

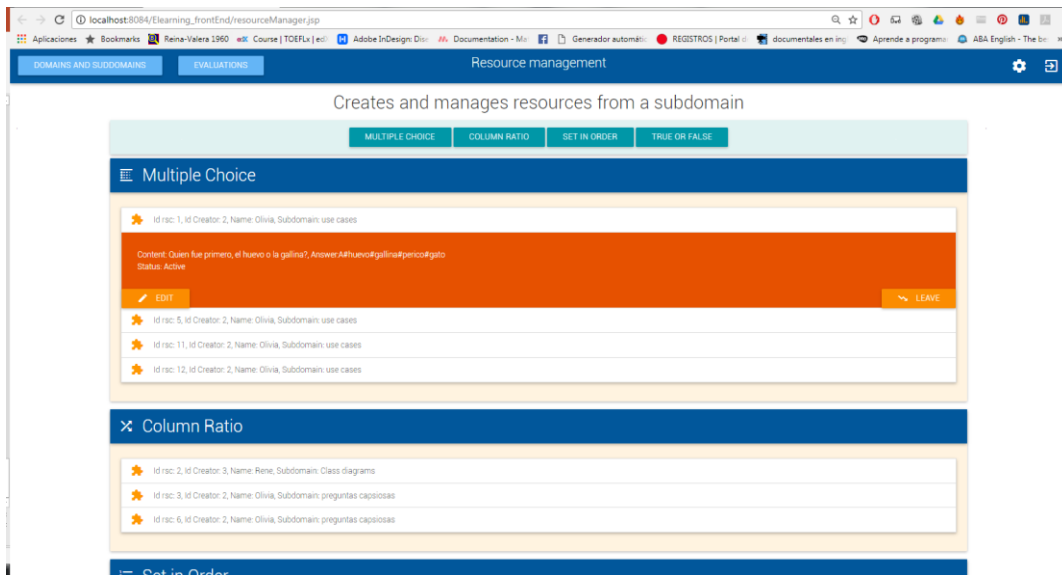


Figura 5. 12 Interfaz de cliente de gestión de recursos

5.3.3.8 Gestión de evaluaciones.

El cliente de gestión de evaluaciones, véase figura 5.13, se encarga de crear las evaluaciones que contendrán un conjunto de recursos de un tipo específico. A través de servicios web, se pueden visualizar, crear, modificar y dar de bajas dichas evaluaciones.

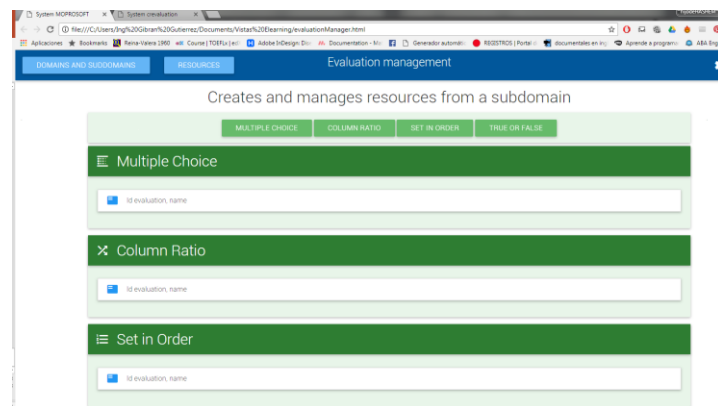


Figura 5. 13 Interfaz de cliente de gestión de evaluaciones

5.3.3.9 Modal para crear dominio.

El cliente de modal¹², que se presenta en la figura 5.14, facilita un cuadro emergente para crear un dominio a partir de un servicio web.

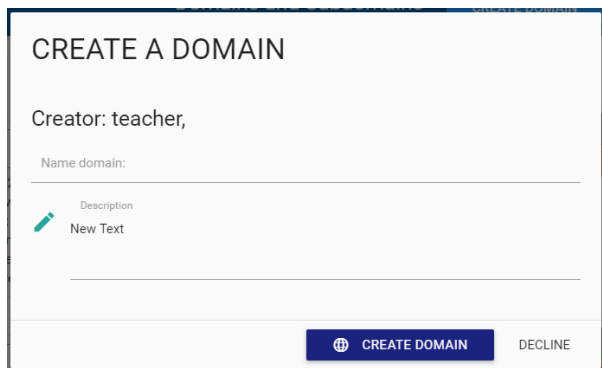


Figura 5. 14 Ventana modal de creación de dominios

5.3.3.10 Modal para editar dominio.

El cliente de modal, de la figura 5.15, facilita un cuadro emergente para editar un dominio a partir de un servicio web.

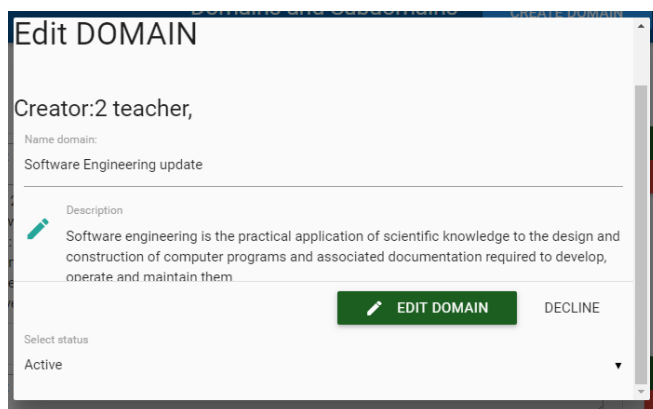


Figura 5. 15 Ventana modal de actualización de dominios

5.3.3.11 Modal para crear sub dominio.

El cliente de modal, de la figura 5.16, facilita un cuadro emergente para crear un sub dominio a partir de un servicio web.

¹² Un modal es un cuadro de diálogo o ventana emergente que se muestra en la parte superior de la página actual.

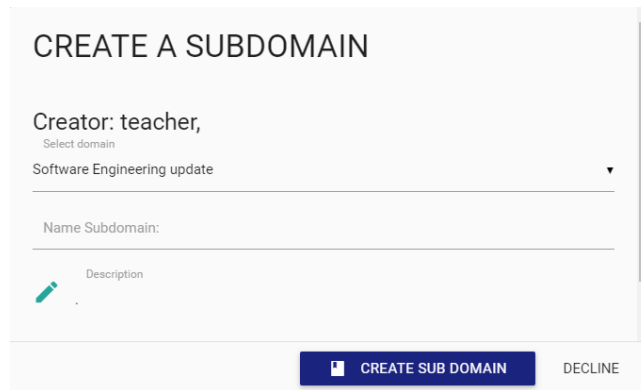


Figura 5. 16 Ventana modal de creación de sub dominios

5.3.3.12 Modal para crear sub dominio.

El cliente de modal, de la figura 5.17, facilita un cuadro emergente para crear un sub dominio a partir de un servicio web.

5.3.3.13 Modal para dar de baja dominio.

El cliente de modal, de la figura 5.17, facilita un cuadro emergente para dar de baja un dominio a partir de un servicio web.

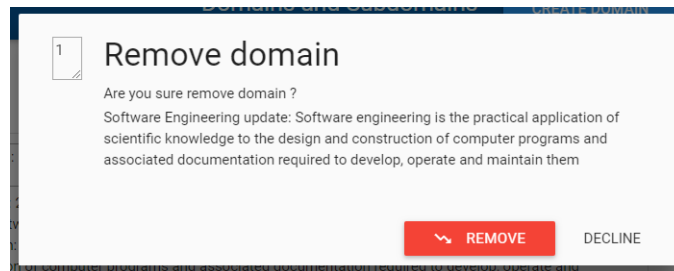


Figura 5. 17 Ventana modal para dar de baja un dominio

5.3.3.14 Modal para editar sub dominio.

El cliente de moda, de la figura 5.18, facilita un cuadro emergente para editar un sub dominio a partir de un servicio web.

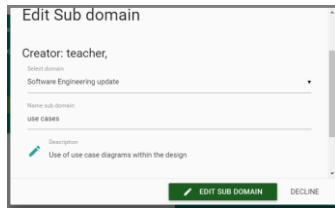


Figura 5. 18 Ventana modal de actualización de sub dominio

5.3.3.15 Modal para dar de baja un sub dominio.

El cliente de modal, de la figura 5.19, facilita un cuadro emergente para dar de baja un sub dominio a partir de un servicio web.

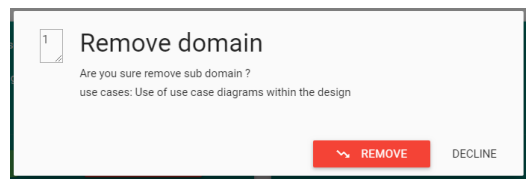


Figura 5. 19 Ventana modal de sub dominio

5.3.3.16 Modal para crear recurso.

El cliente de modal, de la figura 5.20, facilita un cuadro emergente para crear un recurso a partir de un servicio web.

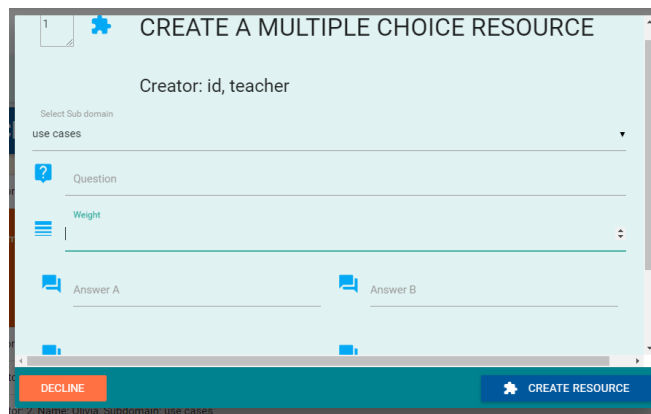


Figura 5. 20 Ventana modal de creación de recurso multiple choice

5.3.3.17 Modal para editar recurso.

El cliente de modal, de la figura 5.21, facilita un cuadro emergente para editar un recurso a partir de un servicio web.



Figura 5. 21 Ventana modal de actualización de recurso

5.3.3.18 Modal para dar de baja recurso.

El cliente de modal, de la figura 5.22, facilita un cuadro emergente para dar de baja un recurso a partir de un servicio web.

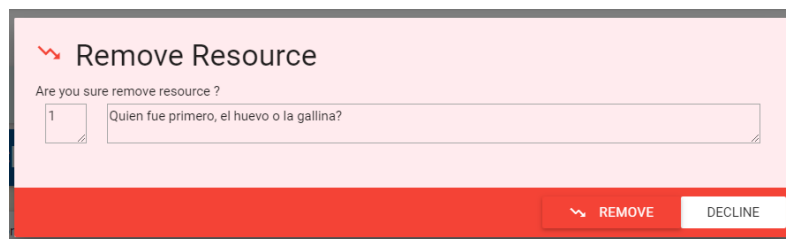


Figura 5. 22 Ventana modal para dar de baja un recurso

5.4. Ejecución de las pruebas

5.4.1 pruebas al componente de calidad de servicios web de aprendizaje

Para hacer la ejecución de las pruebas al componente de calidad de servicios web de aprendizaje se tomó en cuenta el objetivo del presente trabajo, la integración de componentes. Los aspectos que se consideraron, fueron los siguientes:


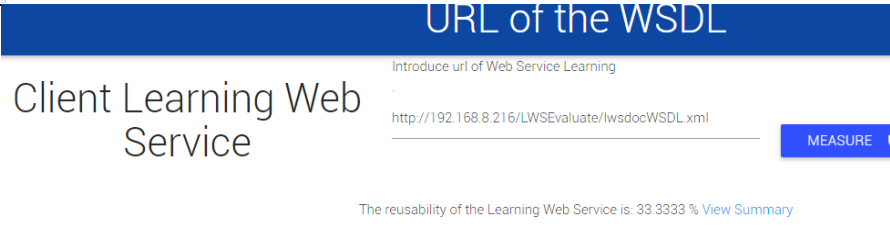
- Integración de los componentes. Los componentes deben de funcionar en conjunto y proporcionar la salida deseada de acuerdo a los requerimientos establecidos.
- Funcionalidad. Los servicios web invocados deben funcionar correctamente y desplegarse de forma independiente.
- Funcionamiento en la red. Los servicios web deben responder de forma correcta en un ambiente donde el cliente y el servicio a invocar se encuentran separados físicamente.

5.4.1.1 Pruebas realizadas

A continuación, se muestran los casos de pruebas realizadas al componente del modelo de calidad de los servicios web de aprendizaje. Para fines prácticos se muestra un caso de prueba por cada funcionalidad y la tabla 5.3 muestra un resumen de los porcentajes de reusabilidad del SWA obtenidos.

El siguiente caso de prueba describe los resultados obtenidos al probar el cliente que invoca al webserviceSoap, el cual provee la funcionalidad de obtener la reusabilidad del servicio web de aprendizaje a partir de una dirección WSDL dado.

Tabla 5. 3 Caso de prueba de cliente para consumir el servicio webServiceSoap

ID de la prueba	Comp2Sub1Doc1	
Ejecutor	Gibrán Gutiérrez Juárez	Fecha: 25 de enero de 2018
Objetivo	Probar la funcionalidad del sub componente evaluador de calidad, en un ambiente de integración.	
Precondiciones	1.- El servidor Glashfish ubicado en el primer nodo se encuentra iniciado 2.- El servidor cliente apache ubicado en el primer nodo se encuentra iniciado y configurado correctamente. 3.- El portal tiene la dirección correcta del segundo nodo.	
Resultado Esperado	El componente debe retornar el porcentaje de reusabilidad del SWA dado a través de la dirección WSDL.	
Entrada:		
Salida:		
Resultado Obtenido	El resultado fue el porcentaje de reusabilidad del SWA proporcionado.	

Se pudo apreciar que el resultado obtenido fue satisfactorio al retornar el porcentaje requerido por el usuario. Las pruebas realizadas resultaron exitosas, como se muestra en el caso de prueba de la tabla 5.3, para el análisis de resultados, véase la parte de análisis de resultados.

En el caso de prueba de la tabla 5.4, se muestra la prueba realizada al componente invocado en el servicio web denominado webServiceSoap2 el cual obtiene la reusabilidad de un servicio web de aprendizaje a partir de una serie de pesos dados por el usuario como se muestra en la figura 5.23. Para fines prácticos sólo se muestra un caso de prueba realizado a dicho componente.

Tabla 5. 4 Caso de prueba del cliente para consumir el servicio webServiceSoap2

ID de la prueba	Comp2Sub2Docx2	
Ejecutor	Gibrán Gutiérrez Juárez	Fecha: 25 de enero de 2018
Objetivo	Probar la funcionalidad del sub componente evaluador de calidad, en un ambiente de integración.	
Precondiciones	1.- El servidor Glashfish ubicado en el primer nodo se encuentra iniciado 2.- El servidor cliente apache ubicado en el primer nodo se encuentra iniciado y configurado correctamente. 3.- El portal tiene la dirección correcta del segundo nodo.	
Resultado Esperado	El componente debe retornar el porcentaje de reusabilidad del SWA dado a través de la dirección WSDL y parámetros de peso, especificados por el usuario.	
Entrada	Vease Fig. 5.23	
Salida	Vease Fig 5.24	
Resultado Obtenido	El resultado fue el porcentaje de reusabilidad del SWA proporcionado.	

Se pudo observar que la prueba hecha para el componente webServiceSoap2, fue el esparado de acuerdo al caso de prueba descrito en la tabla 5.4. al cumplir con las demandas de integracion y atributos propios de un arquitectura basada en servicios web.

Client LWS Customized

Client Learning Web Service Customized

You have do that write the values of criteria, the criteria values must add 1, i.e, value 1+value 2+value 3+value 4+...+value n = 1, the format of the value must be decimal number.

RowAligment

FontType

FontColor

FontSize

BackgroundColor

Introduce url ofp Web Service Learning

MEASURE 

Figura 5. 23 Interfaz del cliente para consumir el componente webServiceSoap2

Client LWS Customized

Client Learning Web Service Customized

You have do that write the values of criteria, the criteria values must add 1, i.e, value 1+value 2+value 3+value 4+...+value n = 1, the format of the value must be decimal number.

RowAligment
0.1

FontType
0.2

FontColor
0.3

FontSize
0.3

BackgroundColor
0.1

Introduce url ofp Web Service Learning

<http://192.168.8.216/LWSEvaluate/lwsdocx.php?wsdl>

MEASURE ✓

: reusability of the Learning Web Service is: 94.4444 % [View Summary](#)

Figura 5. 24 Resultados de datos de salida del caso de prueba Comp2Sub2Docx2

En la tabla 5.5, se muestra los resultados de los casos de pruebas del componente dos, concentrados en una sola tabla para fines prácticos, los servicios web de aprendizaje probados contienen contenido de archivos de documento de documento de texto (.doc y .docx), presentaciones en powerPoint (.ppt y .pptx), documentos en PDF y documento HTML.

Tabla 5. 5 Resultados de casos de prueba del componente 2, webServiceSoap2

Id de la prueba	Tipo de contenido de SWA	Datos de entrada	Datos de salida
Comp2Sub2Doc8	doc	<ol style="list-style-type: none"> http://192.168.8.216/LWSEvaluate/lwsdocx.php?wsdl RowAligment: 0.3 FontType: 0.2 FontColor: 0.3 	92.0748 %

		<ol style="list-style-type: none"> 5. FontSize: 0.1 6. BackgroundColor: 0.1 	
Comp2Sub2Docx9	docx	<ol style="list-style-type: none"> 1. http://192.168.8.216/LWSEvaluate/lwsdocx.php?wsdl 2. RowAligment: 0.1 3. FontType: 0.2 4. FontColor: 0.3 5. FontSize: 0.3 6. BackgroundColor: 0.1 	94.4444 %
Comp2Sub2Pdf10	pdf	<ol style="list-style-type: none"> 1. http://192.168.8.216/LWSEvaluate/lwspdf.php?wsdl 2. RowAligment: 0.15 3. FontType: 0.15 4. FontColor: 0.3 5. FontSize: 0.3 6. BackgroundColor: 0.1 	89.3685 %
Comp2Sub2Ppt11	ppt	<ol style="list-style-type: none"> 1. http://192.168.8.216/LWSEvaluate/lwsppt.php?wsdl 2. RowAligment: 0.05 3. FontType: 0.2 4. FontColor: 0.3 5. FontSize: 0.3 6. BackgroundColor: 0.15 	81.7718 %
Comp2Sub2Pptx12	pptx	<ol style="list-style-type: none"> 1. http://192.168.8.216/LWSEvaluate/lwspptx.php?wsdl 2. RowAligment: 0.1 3. FontType: 0.2 4. FontColor: 0.2 5. FontSize: 0.3 6. BackgroundColor: 0.2 	63.5268 %
Comp2Sub2Htm12	html	<ol style="list-style-type: none"> 1. http://192.168.8.216/LWSEvaluate/lwshtml.php?wsdl 2. RowAligment: 0.1 3. FontType: 0.1 4. FontColor: 0.35 5. FontSize: 0.35 6. BackgroundColor: 0.1 	74.3497 %

Comp2Sub2Dic l12		<ol style="list-style-type: none"> 1. http://192.168.8.216/LWSEvaluate/undefinide 2. RowAligment: 0.1 3. FontType: 0.2 4. FontColor: 0.2 5. FontSize: 0.3 6. BackgroundColor: 0.2 	Please review the data provided for a correct evaluation of the service
---------------------	--	---	---

Los resultados obtenidos en la tabla 5.5, muestran que las pruebas realizadas con diferentes tipos de servicios web de aprendizaje y con diferentes pesos establecidos por el usuario, fueron satisfactorias.

5.4.2 pruebas al componente de sistema de evaluaciones.

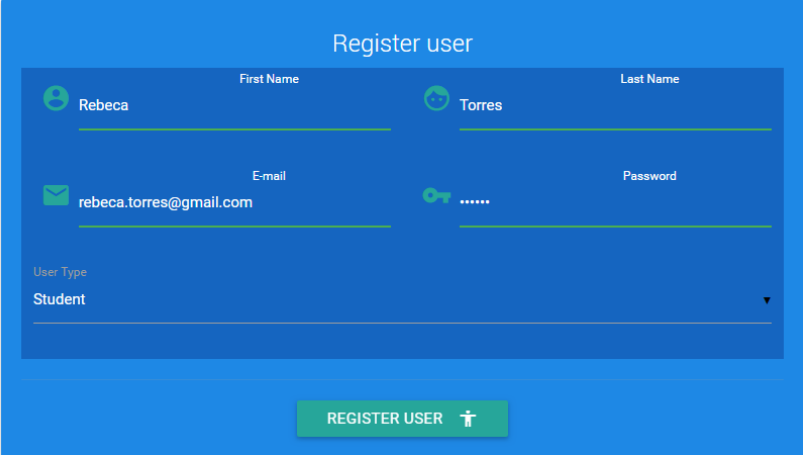
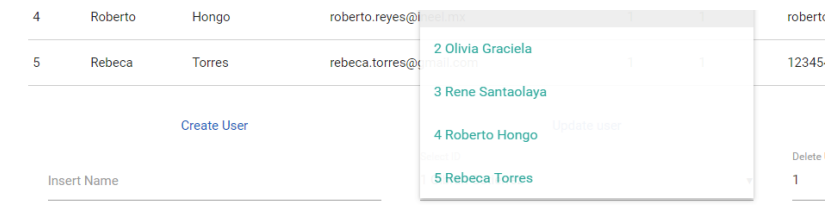
Para entender la forma en que fueron hechas las pruebas en el componente del sistema de evaluaciones, es necesario recordar que cada sub componente del sistema está hecho según la entidad que corresponde al sistema, y el servicio web actúa como patrón facade, de esta forma, cuando se integra por completo, sigue un flujo el cual es transparente para el usuario.

Para fines prácticos de la documentación de las pruebas se muestra solo un caso de uso, con una operación por cada servicio web creado.

A continuación, en la tabla 5.6, se muestra un caso de prueba para el componente1, sistema de evaluación. En el caso de la operación de crear un usuario, el objetivo es registrar un usuario en la base de datos y obtener un resultado de retorno el cual puede ser booleano, las precondiciones dadas son que los servidores tanto para el servicio como para el cliente se encuentren disponibles y desplegados, así como que el portal tenga la dirección correcta del sub componente del sistema evaluador.

Tabla 5. 6 Ejemplo de un caso de prueba al sistema de evaluaciones.

ID de la prueba:	Comp2Sub2WsUserCrUs1	
Ejecutor:	Gibrán Gutiérrez Juárez	Fecha: 20 de Abril de 2018
Objetivo:	Probar la funcionalidad de crear usuario del sub componente de gestión de usuarios.	
Precondiciones:	1.- El servidor Tomcat ubicado en el nodo principal se encuentra iniciado	

	<p>2.- El cliente Tomcat ubicado en el nodo principal se encuentra iniciado y configurado correctamente.</p> <p>3.- El portal tiene la dirección correcta del componente llamado.</p>
Resultado Esperado	El servicio web invocado debe registrar el usuario solicitado en la base de datos y retornar un resultado booleano de acuerdo a la operación.
Entrada:	
Salida:	
Resultado Obtenido	El usuario fué registrado de forma exitosa en la base de datos retornando el valor true.

Se puede observar que en la entrada se introduce desde un cliente independiente los datos para el usuario, y como salida. el registro se puede ver registrado exitosamente en el cliente del administrador. Con esto se puede comprobar la integración del componente de usuario, ya que un cliente puede utilizar dicha funcionalidad sin afectar la integridad de los mismos.

En la siguiente sección se han concentrado los resultados de las pruebas realizadas por cada sub componente, es decir por servicio web, que, en conjunto, proporcionan la funcionalidad del sistema. Se realizaron tres pruebas por cada caso, en las cuales dos pruebas fueron hechas en casos ordinarios y una es un caso de envío de datos incorrectos, la concentración de los datos se puede apreciar en el análisis de los mismos.

las operaciones contenidas dentro del caso de prueba del sub componente wsUser son: crear un usuario, obtener los valores de un usuario de acuerdo a un id, actualizar los valores de un usuario, eliminar un usuario, conectar a una base de datos y realizar el ingreso con un correo y una contraseña como se aprecia en la tabla 5.7.

Tabla 5. 7 Concentración de pruebas del sub componente WsUser

Nombre de operación	Nombre de prueba	Datos de entrada	Datos de salida, comentario opc.
wsoCreateUser	Comp2Sub2WsUserCrUs1	Str: Gibran Str: Gutierrez Str:1234 Str: 1 Str: 1 Str: gibran@gmail	True Registrado exitosamente
	Comp2Sub2WsUserCrUs2	Str: Roberto Str: Martinez Str:232 Str: 1 Str: 1 Str: rober@gmail	True Registrado exitosamente
	Comp2Sub2WsUserCrUs3 Nota: prueba con datos incompletos	Str: Str: Str:1234 Str: 1 Str: 1 Str: @gmail	False resultado exitoso genera excepción a nivel componente
wsoGetValuesUser	Comp2Sub2WsUserGtVa1	Int:1	Cadena con valores de usuario
	Comp2Sub2WsUserGtVa2	Int:2	Cadena con valores de usuario
	Comp2Sub2WsUserGtVa3 Nota: prueba de error	Int: "null"	Cadena vacía
wsoUpdateUser	Comp2Sub2WsUserUpUs1	Str: 1 Str: Roberto Str: Perez Str:2rf Str: 1 Str: 0 Str: rober@gmail	True, Actualización exitosa.
	Comp2Sub2WsUserUpUs2	Str: 1 Str: Rober Str: Mancera Str:975j Str: 1 Str: 1 Str: roberto@gmail	True, Actualización exitosa.

	Comp2Sub2WsUserUpUs3 Nota: prueba de error	Str: 9 Str: "" Str: "" Str: "" Str: 1 Str: 0 Str: rober@gmail	False, Resultado exitoso
wsoDeleteUser	Comp2Sub2WsUserDIUs1	Int: 1	True
	Comp2Sub2WsUserDIUs2	Int: 2	True
	Comp2Sub2WsUserDIUs3 Nota: prueba de error	Int: 1	False
wsoConectdatabase	Comp2Sub2WsUserCnDb1	Str: dbresources Str: root Str: 0813	True
	Comp2Sub2WsUserCnDb2	Str: db Str: root Str: 083	False
	Comp2Sub2WsUserCnDb3 Nota: prueba de error	Str: "" Str: "" Str: ""	Exitoso, Excepción
wsoLogin	Comp2Sub2WsUserLogi1	Str: roberto@gmail Str: 975j	True
	Comp2Sub2WsUserLogi2	Str: roberto@gmail Str: 9	False
	Comp2Sub2WsUserLogi3 Nota: prueba de error	Str: "" Str: ""	Exitoso, Excepción generada

Se puede apreciar que los resultados obtenidos son satisfactorios, en la mayoría de las operaciones, las excepciones programadas para los posibles errores respondieron de forma exitosa. En el caso de la conexión de la base de datos genera una excepción a nivel de servidor programada, por lo que es considerado como un resultado exitoso.

Las operaciones contenidas dentro del caso de prueba del sub componente swDomain son: crear un dominio, obtener la lista de los dominios disponibles, actualizar un dominio y dar de baja dicho dominio como se aprecia en la tabla 5.8.

Tabla 5. 8 Concentración de pruebas del sub componente swDomain

Nombre de operación	Nombre de prueba	Datos de entrada	Datos de salida, comentario opc.
---------------------	------------------	------------------	----------------------------------

wsoCreateDomain	Comp2Sub2WsDomnCrDm1	Str: 1 Str: Ingles Str: dominio Str: ingles	True Registrado exitosamente
	Comp2Sub2WsDomnCrDm2	Str: 1 Str: Espaniol Str: materia	True Registrado exitosamente
	Comp2Sub2WsDomnCrDm3 Nota: prueba con datos incompletos	Str: 8 Str: "" Str: ""	False resultado exitoso, genera excepción
wsoGetListDomain	Comp2Sub2WsDomnGtLs1		Cadena con Dominios
	Comp2Sub2WsDomnGtLs2		Cadena con Dominios
	Comp2Sub2WsDomnGtLs3 Nota: prueba de error		Cadena con Dominios
wsoUpdateDomain	Comp2Sub2WsDomnUpDm1	Str: 1 Str: 1 Str: English Str: description Str: 1	True, Actualización exitosa.
	Comp2Sub2WsDomnUpDm2	Str: 1 Str: 1 Str: English Str: desc. update Str: 0	True, Actualización exitosa.
	Comp2Sub2WsDomnUpDm3 Nota: prueba de error	Str: 9 Str: 9 Str: null Str: null Str: 5	False, Excepción generada, resultado exitoso
wsoDeleteDomain	Comp2Sub2WsDomnDlDm1	Int: 1	True
	Comp2Sub2WsDomnDlDm2	Int: 2	True
	Comp2Sub2WsDomnDlDm3 Nota: prueba de error	Int: 1	False

Los resultados obtenidos fueron altamente satisfactorios, ya que solo el caso de la actualización de dominio tuvo una excepción controlada, mientras que las demás pruebas respondieron de forma correcta, en el caso de la creación de un dominio la excepción respondió con la bandera booleana falsa y continuó con el flujo del sistema.

Las operaciones contenidas dentro del caso de prueba, como se aprecia en la tabla 5.9, del sub componente swSubDomain son: crear un sub dominio a partir de un dominio creado, obtener el listado de los sub domonios disponibles, actualizar un sub dominio existente, dar de baja un sub dominio y obtener una cadena de sub dominios con una estructura de <SELECT> para su fácil uso de un cliente.

Tabla 5. 9 Concentración de pruebas del sub componente wsSubDomain

Nombre de operación	Nombre de prueba	Datos de entrada	Datos de salida, comentario opc.
wsoCreateSD	Comp2Sub2WsSbdmCrSd1 (idUser,idDom,nameSD, description, status)	Str: 1 Str: 2 Str: Grammar Str: desc. gram Str: 1	True Registrado exitosamente
	Comp2Sub2WsSbdmCrSd2	Str: 1 Str: 2 Str: Comprehension Str: desc. Crom. Str: 1	True Registrado exitosamente
	Comp2Sub2WsSbdmCrSd3 Nota: prueba con datos incompletos	Str: 11 Str: 2 Str: Grammar2 Str: desc. Gram2 Str: 0	False resultado esperado genera excepción, resultado exitoso
wsoGetListSD	Comp2Sub2WsSbdmGtLs1		Cadena con Sub Dominios
	Comp2Sub2WsSbdmGtLs2		Cadena con Sub Dominios
	Comp2Sub2WsSbdmGtLs3 Nota: prueba de error		Cadena con Sub Dominios
wsoUpdateSD	Comp2Sub2WsSbdmUpSd1 (pSdlduser, pSdld, pSdldSD, pSdName, pSdDescription, pSdStatus)	Str: 2 Str: 2 Str: 1 Str: Gramar upd Str: desc upd	True, Actualización exitosa.

		Str: 0	
	Comp2Sub2WsSbdmUpSd2	Str: 1 Str: 2 Str: 1 Str: Grammar upd2 Str: desc upd Str: 1	True, Actualización exitosa.
	Comp2Sub2WsSbdmUpSd3 Nota: prueba de error	Str: 28 Str: 2 Str: 1 Str: Grammar upd Str: desc upd7 Str: 1	False, Excepción generada
wsoLeaveSD	Comp2Sub2WsSbdmLvSd1	Int: 2	True
	Comp2Sub2WsSbdmLvSd2	Int: 3	True
	Comp2Sub2WsSbdmLvSd3 Nota: prueba de error	Int: 2	False, excepción generada. Resultado exitoso
wsoGetListSelectSubDomain	Comp2Sub2WsSbdmGsSd1		Cadena con estructura de select.
	Comp2Sub2WsSbdmGsSd2		Cadena con estructura de select.
	Comp2Sub2WsSbdmGsSd3 Nota: prueba de error		Cadena con estructura de select.

Los resultados obtenidos son satisfactorios ya que se obtuvieron el manejo de dos excepciones, las cuales respondieron de forma esperada, los demás resultados fueron satisfactorios de acuerdo a los parámetros dados.

Las operaciones contenidas dentro del caso de prueba del sub componente wsoResources son: crear un recurso perteneciente a un sub dominio, obtener listado completo de un tipo de recurso especificado, actualizar un recurso, dar de baja un recurso, obtener información listada de los recursos disponibles para su uso en modals como se aprecia en la tabla 5.10.

Tabla 5. 10 Concentración de pruebas del sub componente wsoResources

Nombre de operación	Nombre de prueba	Datos de entrada	Datos de salida, comentario opc.
wsoCreateResource	Comp2Sub2WsRescCrRs1 (prsIdUser, prsIdSd, prsType, prsContent, prsAnswer, prsStatus, prsWeight)	Str: 1 Str: 3 Str: 1 Str: un numero mayor a 5? Str: 1#3#2#8 Str: 1 Str: 0.23	True Registrado exitosamente
	Comp2Sub2WsRescCrRs2	Str: 1 Str: 3 Str: 1 Str: un numero mayor a 9? Str: 1#5#2#12 Str: 1 Str: 0.50	True Registrado exitosamente
	Comp2Sub2WsRescCrRs3 Nota: prueba con datos incompletos	Str: 92 Str: 3 Str: 1 Str: un numero mayor a 5? Str: 1#3#2#8 Str: 1 Str: 0.43	False resultado exitoso, genera excepción
wsoReadResourceComplete	Comp2Sub2WsRescRsCo1 String rscType	Str: 1	Cadena con recursos según tipo.
	Comp2Sub2WsRescRsCo2	Str: 1	Cadena con recursos según tipo.
	Comp2Sub2WsRescRsCo3 Nota: prueba de error	Str: 8	Resultado exitoso, genera excepción
wsoUpdateResource	Comp2Sub2WsRescUpRs1 prsIdUser, prsIdSd, prsType, prsContent, prsIdAnswer, prsStatus, idRsc, prsWeight	Str: 2 Str: 3 Str: 1 Str: un numero mayor a 5? Str: 1#3#2#8 Str: 1 Str: 3	True, Actualización exitosa.

		Str: 0.43	
	Comp2Sub2WsRescUpRs2	Str: 2 Str: 3 Str: 1 Str: un numero mayor a 9? Str: 1#3#2#8 Str: 1 Str: 3 Str: 0.98	True, Actualización exitosa.
	Comp2Sub2WsRescUpRs3 Nota: prueba de error	Str: 2 Str: 3 Str: 1 Str: prueba de excepción? Str: 1#3#2#8 Str: 1 Str: 354 Str: 0.80	False, Excepción generada. Resultado exitoso
wsoLeaveResource	Comp2Sub2WsRescLvRs1	str: 2	True
	Comp2Sub2WsRescLvRs2	str: 3	True
	Comp2Sub2WsRescLvRs3 Nota: prueba de error	Str: null	False
wsogetResourceModals	Comp2Sub2WsRescGtMd1 String pidUser	Str: 2	Cadena con recursos para modals
	Comp2Sub2WsRescGtMd2	Str: 54	Cadena vacía
	Comp2Sub2WsRescGtMd3 Nota: prueba de error	Str: null	Cadena vacía Excepción generada, resultado exitoso

Los resultados obtenidos fueron satisfactorios, se obtuvieron dos excepciones que fueron probadas y con resultados exitosos. Los demás resultados fueron calificados como excelente, para ver la ponderación de los resultados, véase la sección de análisis de pruebas.

Las operaciones contenidas dentro del caso de prueba del sub componente wsEvaluation son: crear una evaluación a partir de un sub dominio especificado y un usuario, obtener un listado de evaluaciones específicas de un tipo y dar de baja una evaluación como se aprecia en la tabla 5.11.

Tabla 5. 11 Concentración de pruebas del sub componente wsEvaluation

Nombre de operación	Nombre de prueba	Datos de entrada	Datos de salida, comentario opc.
wsoCreateEvaluation	Comp2Sub2WsEvalCrEv1 (prslidUser, prslidSd, prsTitle, prsDescription, prsType, prsStatus)	Str: 1 Str: 3 Str: Evaluation spanish Str: test Str: 1 Str: 1	True Registrado exitosamente
	Comp2Sub2WsEvalCrEv2	Str: 1 Str: 3 Str: Diagnostic vocabulary Str: description test. Str: 1 Str: 1	True Registrado exitosamente
	Comp2Sub2WsEvalCrEv3 Nota: prueba con datos incompletos	Str: 1 Str: 3 Str: Eval test Str: test Str: 54 Str: 64	False resultado esperado genera excepción, resultado exitoso
wsoLeaveEvaluation	Comp2Sub2WsEvalLvEv1	str: 1	True
	Comp2Sub2WsEvalLvEv2	str: 2	True
	Comp2Sub2WsEvalLvEv3 Nota: prueba de error	Str: null	False, excepción, resultado exitoso
wsoReadTableTypeEvaluation	Comp2Sub2WsEvalRtTe1 String rscType	Str: 1	Cadena con Evaluaciones según tipo.
	Comp2Sub2WsEvalRtTe2	Str: 3	Cadena con Evaluaciones según tipo.
	Comp2Sub2WsEvalRtTe3 Nota: prueba de error	Str: 432	genera excepción,

			retorna cadena vacía. Resultado exitoso.
--	--	--	---

Los resultados obtenidos en las pruebas al sub componente wsEvaluation fueron satisfactorias, se obtuvieron dos excepciones que fueron probadas y se obtuvo un resultado exitoso, las demás pruebas hechas con datos ordinarios, se obtuvieron resultados, altamente satisfactorios.

5.5 Análisis de pruebas

5.5.1 Análisis de pruebas del Componente del modelo de calidad de SWA.

Tras haber realizado las pruebas en el primer componente del sistema integrador, se prosiguió a concentrar los resultados obtenidos en gráficos para su análisis detallado.

Para calificar el resultado obtenido y documentarlo de forma cuantificable, se establecieron indicadores para el resultado de las pruebas, los tabuladores y su descripción a manejar en dicha prueba y en el componente dos (Sistema de evaluaciones) se muestran en la tabla 5.6

Tabla 5. 12 Criterios de calificación de pruebas

Criterios de calificación de pruebas		
Estatus	Porcentaje	Descripción
Exitoso	100%	Prueba con el resultado esperado según el caso de prueba
No exitoso	50%	Prueba con error de respuesta, no esperado.

En la tabla 5.7 se puede apreciar los resultados para la prueba del sub componente webServiceSoap, en el cual se puede apreciar que en base al número de pruebas las cuales fueron 7, se obtuvo un porcentaje de 100% de éxito promedio en total. Se puede decir que el resultado de las pruebas para el componente del modelo de calidad de los servicios web de aprendizaje, fue satisfactorio, según el enfoque del trabajo propuesto.

Tabla 5. 13 Registro de resultados de pruebas hechas al componente webServiceSoap

Sub Componente 1		
clave de prueba	estatus de prueba	Porcentaje
Comp2Sub1doc1	Exitoso	100%
Comp2Sub1docx2	Exitoso	100%

Comp2Sub1Pdf3	Exitoso	100%
Comp2Sub1Ppt4	Exitoso	100%
Comp2Sub1Pptx5	Exitoso	100%
Comp2Sub1Html6	Exitoso	100%
Comp2Sub1DicI7	Exitoso	100%
Total		100%

En el gráfico 5.25, se muestra la ponderación a nivel general de cada prueba y la sumatoria de los mismos para llegar al resultado documentado en la tabla 5.7 como se puede apreciar, la suma del resultado de cada prueba da el peso aproximado de 15% por cada prueba, ya que un resultado esperado puede afectar a otros casos de prueba hechos, lo cual indica, que, para obtener un resultado óptimo, todas las pruebas deben responder de forma exitosa.

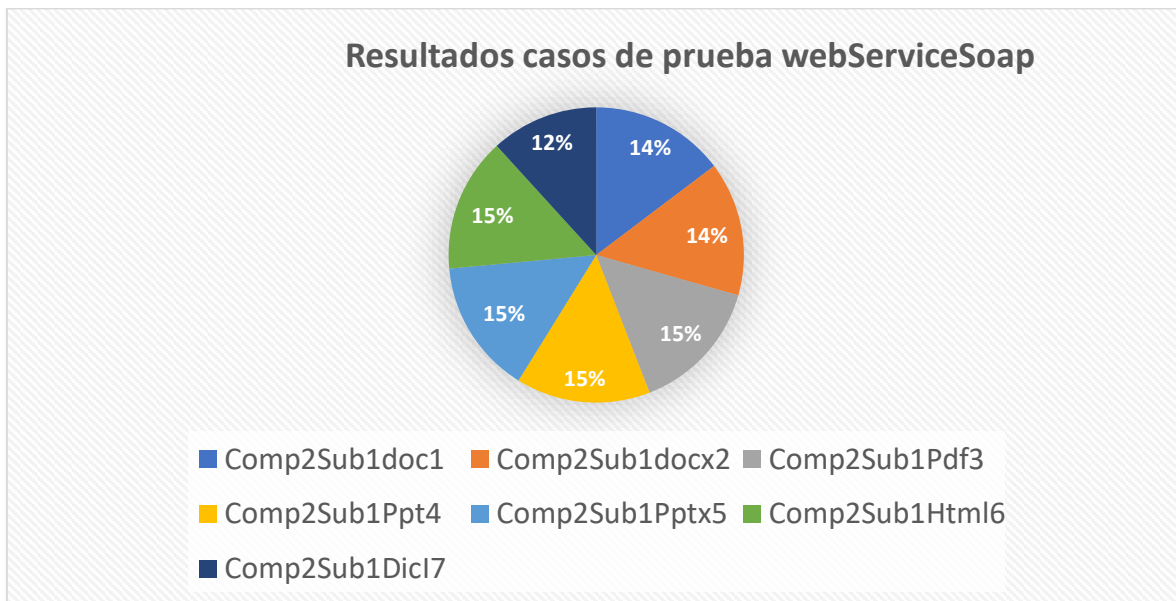


Figura 5. 25 Gráfico de resultados de pruebas del componente webServiceSoap

Para el resultado del sub componente 2 los resultados obtenidos fueron los mismos. Se comprobó que, al asignar pesos para obtener la reusabilidad de un servicio web de aprendizaje, los resultados se mantienen, como se puede apreciar en la tabla 5.8.

Tabla 5. 14 Registro de resultados de pruebas hechas al componente webServiceSoap2

Sub Componente 2		
clave de prueba	estatus de prueba	Porcentaje
Comp2Sub2doc8	Exitoso	100%
Comp2Sub2docx9	Exitoso	100%
Comp2Sub2Pdf10	Exitoso	100%
Comp2Sub2Ppt11	Exitoso	100%
Comp2Sub2Pptx12	Exitoso	100%
Comp2Sub2Html13	Exitoso	100%
Comp2Sub2Dic14	Exitoso	100%
Total		100%

De igual forma, en el gráfico, se puede apreciar una ponderación idéntica, ya que como se mencionó anteriormente, el asignar pesos a la prueba no altero el valor de los resultados.

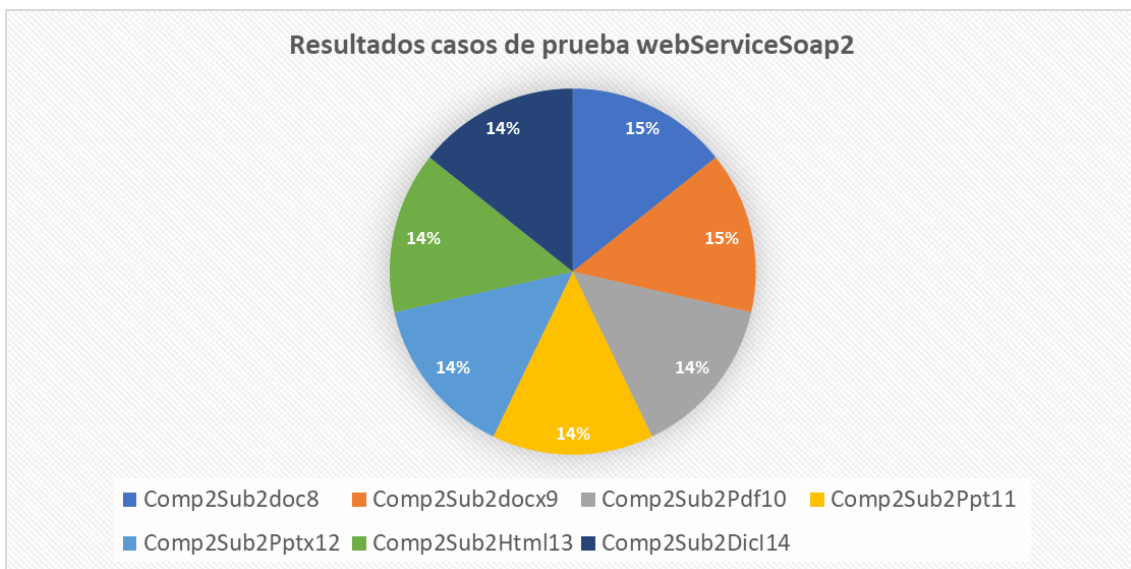


Figura 5. 26 Gráfico de resultados de pruebas del componente webServiceSoap2

Para mayor información de los resultados obtenidos en los casos de prueba, véase en los anexos del trabajo.

5.5.2 Análisis de pruebas del Componente del sistema de evaluaciones.

Para una mayor comprensión más amplia de los resultados obtenidos de las pruebas realizadas y documentadas en las tablas anteriores, se procedió a concentrar los resultados de las tres pruebas en un solo ponderador, de manera que cada operación de cada sub componente tiene el promedio general de las tres pruebas realizadas. Posteriormente se

calcula el promedio general a nivel de sub componente y se muestra el grafico mostrando el comportamiento según la operación.

Por último, se muestra la ponderación final de cada sub componente para así obtener el grafico de resultados y el porcentaje final del componente del sistema de evaluaciones.

En la Tabla 5.15 se puede apreciar que los resultados obtenidos fueron calificados como excelentes a excepción del sub componente de la conexión a la base de datos, por una falla menor en el manejo de parámetros. El resultado general es del 100%, un porcentaje excelente.

Tabla 5. 15 Porcentajes obtenidos para el sub componente wsUser

wsUser		
Operación	Estatus	Promedio general
wsoCreateUser	Excelente	100%
wsoGetValuesUser	Excelente	100%
wsoUpdateUser	Excelente	100%
wsoDeleteUser	Excelente	100%
wsoConectdatabase	Excelente	100%
Total		100%

En la figura 5.27 se puede apreciar como el resultado de las pruebas se mantiene arriba del 950%, un resultado aceptable para todas las operaciones probadas.

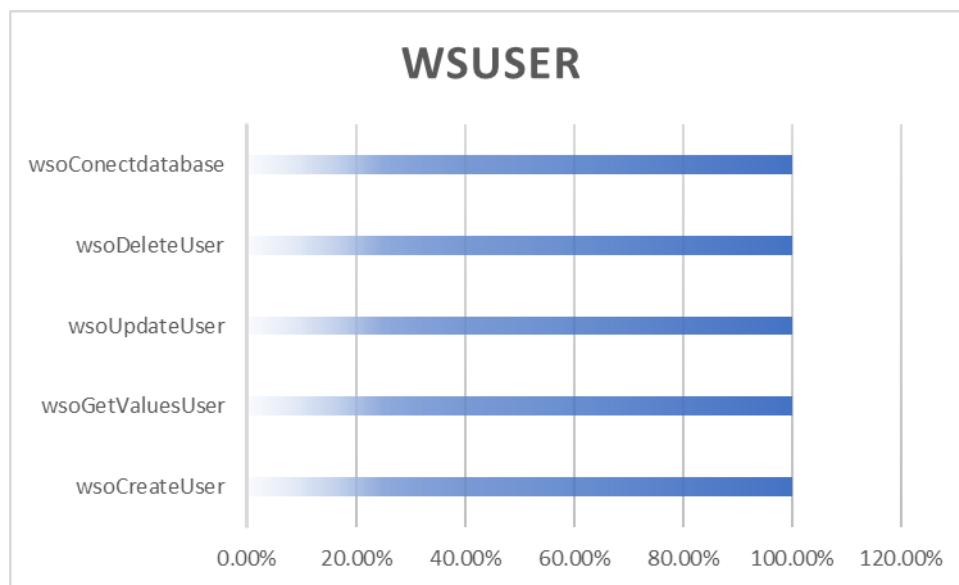


Figura 5. 27 Gráfico de los porcentajes obtenidos de wsUser

En la Tabla 5.16 se puede apreciar que los resultados obtenidos fueron calificados como excelentes. Esto se debe a que el manejo de pruebas con datos incorrectos arrojó resultados manejados por excepciones, los cuales ciertamente son resultados que están planeados

esperar, y como tal exitosos. El porcentaje general obtenido es de 100%, un resultado excelente.

Tabla 5. 16 Porcentajes obtenidos para el sub componente wsDomain

wsDomain		
Operación	Estatus	Promedio general
wsoCreateDomain	Excelente	93.30%
wsoGetListDomain	Excelente	100.00%
wsoUpdateDomain	Excelente	100.00%
wsoDeleteDomain	Excelente	100.00%
	Total	100%

En la figura 5.28 se puede apreciar que los porcentajes de las operaciones se mantienen aún arriba del 95%, lo cual lo mantiene como un resultado aceptable.

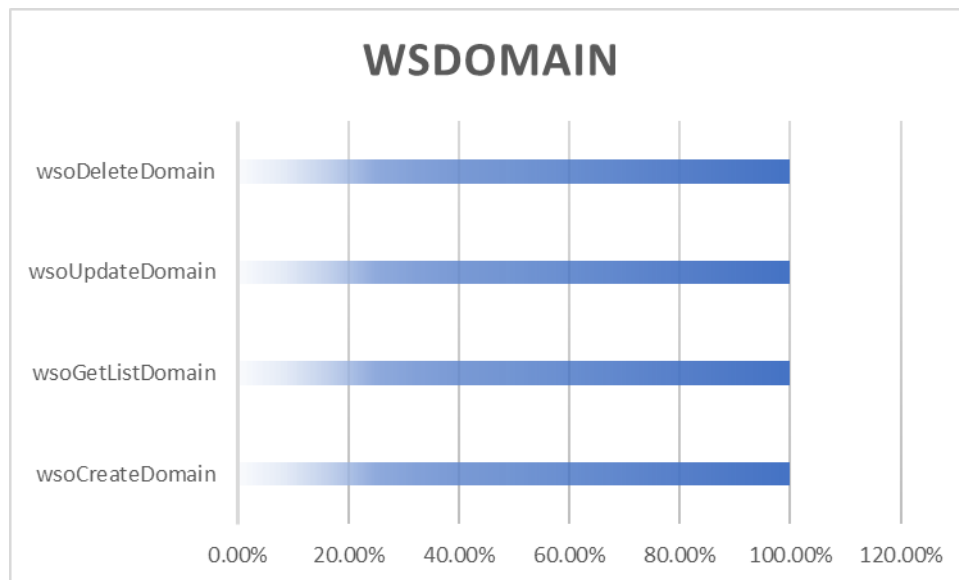


Figura 5. 28 Gráfico de los porcentajes obtenidos de wsDomain

En la Tabla 5.17 se puede apreciar que todos los resultados obtenidos fueron calificados como excelente. En el sub componente de sub dominios se manejaron de forma satisfactorias las entradas con datos incompletos y los resultados obtuvieron un porcentaje general de 100%

Tabla 5. 17 Porcentajes obtenidos para el sub componente wsSubDomain

wsSubDomain		
Operación	Estatus	Promedio general

wsoCreateSD	Excelente	100.00%
wsoGetListSD	Excelente	100.00%
wsoUpdateSD	Excelente	100.00%
wsoLeaveSD	Excelente	100.00%
wsoGetListSelectSubDomain	Excelente	100.00%
Total		100.00%

En la figura 5.29, se puede apreciar que todas las operaciones alcanzaron un porcentaje arriba del 95%, lo cual es un resultado altamente aceptable

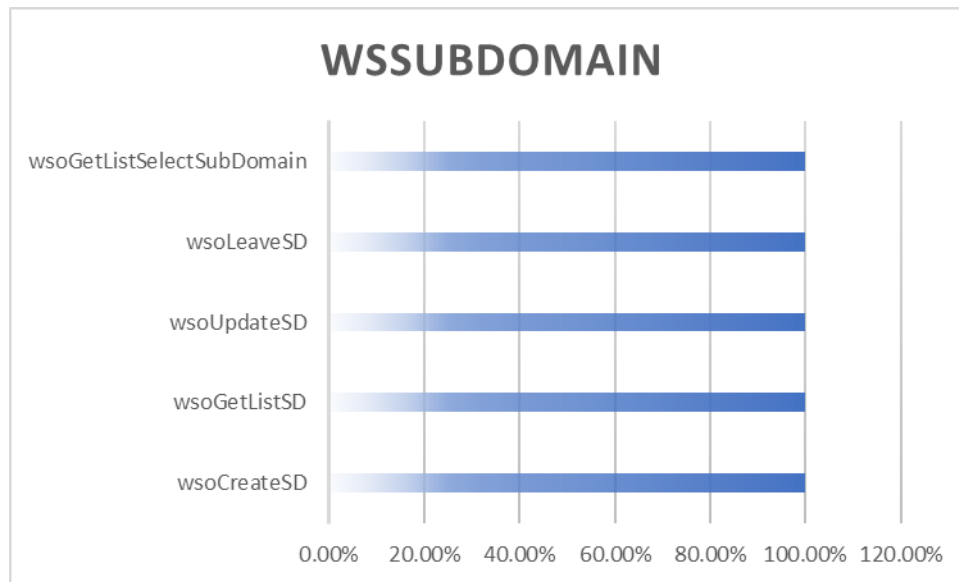


Figura 5. 29 Gráfico de los porcentajes obtenidos de wsDomain

En la Tabla 5.18 se puede apreciar que cuatro de los cinco resultados obtenidos fueron calificados como excelente. El porcentaje general obtenido es de 100%

Tabla 5. 18 Porcentajes obtenidos para el sub componente

wsResources		
Operación	Estatus	Promedio general
wsoCreateResource	Excelente	100%
wsoReadResourceComplete	Excelente	100%
wsoUpdateResource	Excelente	100%
wsoLeaveResource	Excelente	100%
wsoGetResourceModals	Excelente	100%
Total		100%

En la figura 5.30 se puede apreciar que cuatro de los cinco resultados se mantuvieron arriba del 90% un resultado aceptable en términos generales del sub componente.

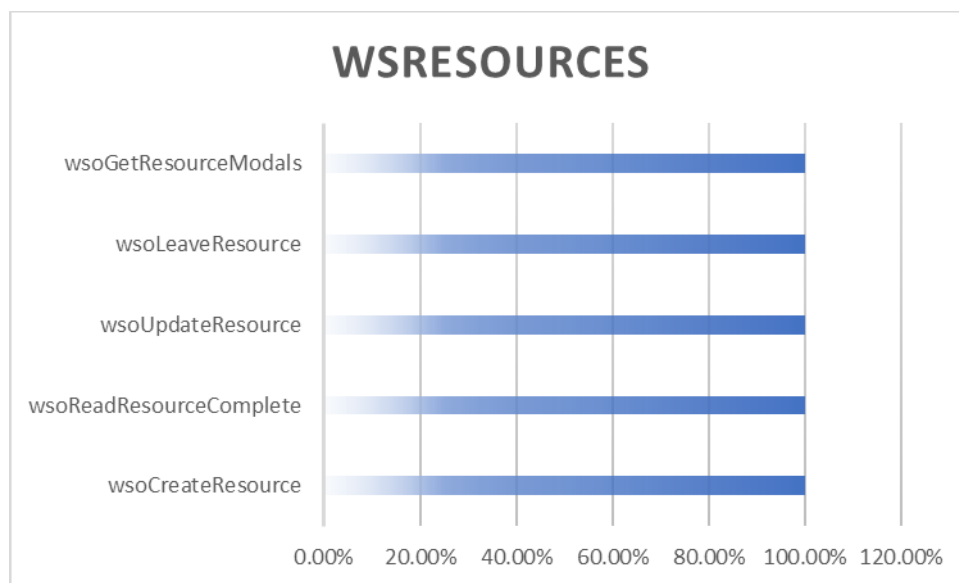


Figura 5. 30 Gráfico de los porcentajes obtenidos de wsResources

En la Tabla 5.19 se puede apreciar que dos de los tres resultados obtenidos fueron calificados como excelente. El porcentaje general obtenido es de 100%, un porcentaje excelente.

Tabla 5. 19 Tabla porcentajes obtenidos para el sub componente wsEvaluation

wsEvaluation		
Operación	Estatus	Promedio general
wsoCreateEvaluation	Excelente	100%
wsoReadTableTypeEvaluation	Excelente	100%
wsoLeaveEvaluation	Excelente	100%
Total		100%

En la figura 5.31 se puede apreciar que las operaciones se mantienen arriba del 95%.

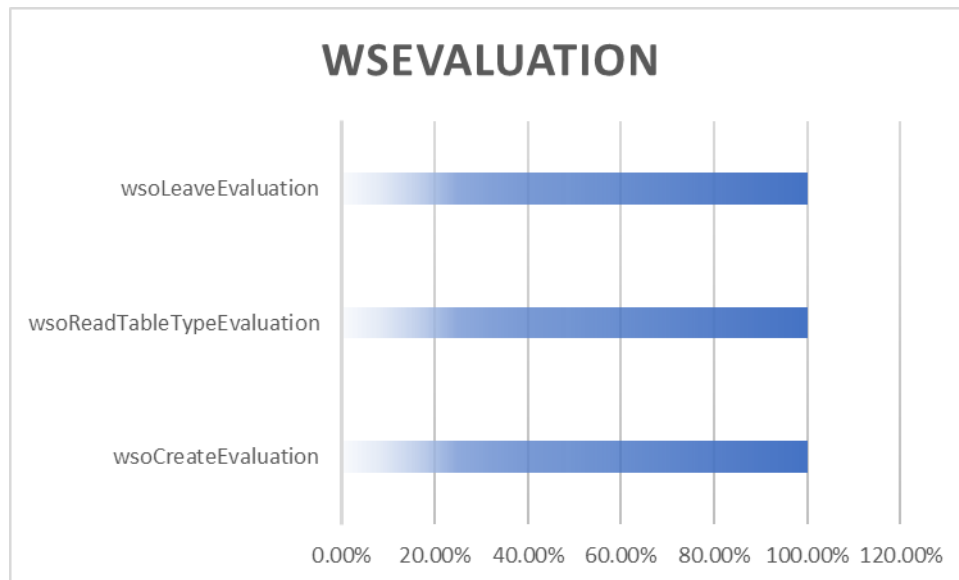


Figura 5. 31 Gráfico de los porcentajes obtenidos de wsEvaluation

Por último, se concentraron todos los sub componentes en la Tabla 5.20, evaluados por servicio y el porcentaje general obtenido. Se puede apreciar que todos han sido calificados como excelente y que el porcentaje general del sistema de evaluaciones es de 92.5%

Tabla 5. 20 Porcentajes obtenidos para el sistema de evaluaciones

Servicios web		
Nombre Servicio	Estatus	Promedio general
wsUser	Excelente	100%
wsDomain	Excelente	100%
wsSubDomain	Excelente	100%
wsResources	Excelente	100%
wsEvaluation	Excelente	100%
	Total	100%

En la figura 5.32 se puede apreciar que todos los sub componentes se mantienen en un rango arriba del 95% a nivel general, lo cual se entiende como un resultado altamente aceptable.

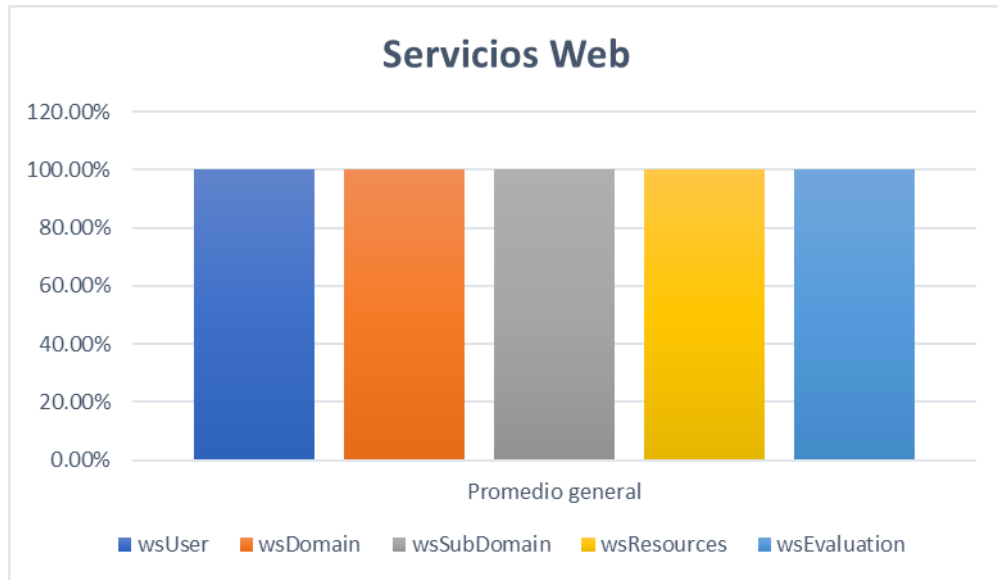


Figura 5. 32 Gráfico de los porcentajes obtenidos del sistema de evaluaciones

5.6 Conclusiones

5.6.1 Conclusión de análisis de componente de modelo de calidad de servicios web de aprendizaje.

Los resultados de las pruebas ejecutadas del componente de calidad del contenido de servicios web de aprendizaje son el porcentaje de la calidad del contenido ya se por medio de pesos dados por el usuario o solo enviando el WSDL del servicio web.

Con propósitos de prueba práctica, se emplearon los mismos casos de prueba que en el documento de Salinas Roman, H. (2017), pero con otro cliente y en una ubicación física diferente del servicio web a llamar, ya que como se ha mencionado a lo largo del capítulo, el fin de las pruebas es mostrar la integración que implica el uso de una arquitectura basada en servicios web, específicamente usando el protocolo SOAP.

Algunas apreciaciones relevantes en los que se concluyen las pruebas realizadas son las siguientes:

- El tiempo de respuesta de petición del cliente al servicio a solicitar es un poco más, debido a la petición a través de tres nodos físicos (equipos de cómputo), el tiempo aproximado es de 2 a 3 segundos más, cabe mencionar que los resultados obtenidos son satisfactorios.
- La reacción de respuesta con un cliente distinto, construido para las pruebas del trabajo presente, es satisfactoria ya que las peticiones y el uso de la interfaz responden de forma satisfactoria al usuario.

- Se encontró que al usar algunos pesos distintos para invocar el servicio llamado "webServiceSoap2" reaccionaba de forma no esperada, es decir, arrojaba resultados no esperados como en el caso de prueba donde se insertan valores equivocados, es recomendable hacer pruebas con pesos distintos para comprobar por completo la funcionalidad del servicio web solicitado.
- El servicio web solicitado puede ser accedido, ya que puede ser llamado desde cualquier equipo que se encuentre en la red, sin depender del cliente y con un despliegue independiente. Los resultados obtenidos respecto a la disponibilidad de los servicios "webServiceSoap" y "webServiceSoap2" son satisfactorios

Con las pruebas realizadas de forma exitosa, y obteniendo conclusiones satisfactorias, se puede decir que la arquitectura de integración, propuesta en el trabajo presente es adecuada para la integración de componentes de gestión de servicios web de aprendizaje(SWA).

5.6.2 Conclusión del componente del sistema de evaluaciones.

Los resultados obtenidos del componente del sistema de evaluaciones son la integración correcta de los sub componentes que conforman el componente, y el correcto funcionamiento del mismo, ya que, en conjunto de llamadas de servicios web, los clientes invocan las operaciones de servicios web de acuerdo a la petición solicitada. Al alimentar el sistema con dominios específicos, sub dominios a partir de ellos y posteriormente recursos para convertirse en evaluaciones, se obtiene un sistema capaz de generar evaluaciones dinámicas de cuatro tipos. El resultado obtenido del trabajo a nivel general es una arquitectura basada en servicios web con el patrón de diseño facade implementado, el cual funciona a manera de componentes integrados por sub componentes, los cuales pueden ser fácilmente acoplados y desacoplados para fácil integración.

Objetivamente, algunas de las apreciaciones que se lograron obtener respecto al componente de sistema de evaluaciones son:

- Integración a nivel de información. Se logró integrar los sub componentes a nivel de información ya que varios sub componentes hicieron uso de la misma información sin generar inconsistencias, incluso, se generaron dos clientes para la creación de usuarios, los resultados obtenidos fueron satisfactorios, ya que no se presentó ninguna inconsistencia y su uso en el sistema integrado se acopló con facilidad.
- Despliegue independiente de componente. Los sub componentes desarrollados en el sistema de evaluaciones fueron desarrollados bajo la metodología planteada en el trabajo presente, al realizar las pruebas de cada operación, se probaron por separado, desplegando cada servicio de forma independiente, los resultados obtenidos en las condiciones descritas resultaron satisfactorios.
- Fácil integración de sub componentes. Uno de los elementos de la metodología con la que se desarrolló el sistema es el desarrollo de sub componentes por entidad. Con

ello se logró que los componentes brinden sus funcionalidades en un ambiente de integración sencillo, a partir de llamadas de servicios web y en colaboración con los otros servicios web disponibles.

6

CONCLUSIONES Y TRABAJOS FUTUROS

Los argumentos más fuertes no prueban nada, siempre y cuando las conclusiones no son verificadas por la experiencia. La ciencia experimental es la reina de las ciencias y la meta de toda especulación.
-Roger Bacon

En este capítulo se describe la conclusión general de acuerdo a los resultados obtenidos en el presente trabajo, así como observaciones y un análisis de trabajos futuros.

Capítulo 6 Conclusiones

6.1 Conclusiones

El problema tratado en el presente trabajo fue resolver la integración de componentes de gestión de servicios web de aprendizaje basado en la metodología que se definió en el capítulo 3. Con el estudio de los trabajos relacionados, se encontraron modelos propuestos con una arquitectura basada en componentes, un sistema basado en servicios web, dividido en entidades y algunas plataformas de LMS. Se encontró que algunos modelos sólo proponían el diseño y algunos más obtenían un pequeño prototipo, pero no se describió con suficiente detalle las pruebas necesarias. Dentro de los componentes a integrar se eligieron un sistema que implementa un modelo de calidad de servicios web de aprendizaje y un sistema para evaluar el aprendizaje de estudiantes en diferentes dominios.

A partir de la metodología basada en servicios web para la integración de sistemas y tomando los factores esenciales para obtener componentes independientes, se definió una arquitectura integral que contempla la forma de integración de un componente dedicado a la gestión de servicios web de aprendizaje a un sistema general, el cual puede desplegar cada componente de forma independiente.

El resultado obtenido es un sistema capaz de facilitar la integración de componentes dedicados a la gestión de SWA. El sistema a nivel general ha logrado integrar los dos componentes (Salinas Roman, H. (2017) y Gregorio, J. and Gutiérrez, G. (2015).), en forma de servicios web, que, con fines de prueba, fueron desplegados en nodos físicos distintos, de los cuales, los resultados obtenidos y documentados fueron altamente aceptables. Los componentes desarrollados fueron creados bajo el protocolo de SOAP, ya que dicha tecnología ha sido probada ampliamente de acuerdo a los trabajos relacionados.

El objetivo del presente trabajo se cumplió y se observaron las siguientes conclusiones.

1. Sistema integrador de componentes de gestión de servicios web de aprendizaje

- El sistema se desarrolló bajo una arquitectura basada en componentes de acuerdo con la metodología definida. Se observó que la integración bajo dicho enfoque fue el esperado de acuerdo a las pruebas funcionales realizadas.
- Se integró el componente 1 (Salinas Roman, H. (2017)), modelo de calidad para servicios web de aprendizaje, con el fin de integrar un componente que ya tuviera una arquitectura basada en servicios web y con el patrón de diseño strategy, sin necesidad de usar el patrón de diseño facade que es el indicado para la integración. La integración del componente fue fácil debido a que se contaba con una arquitectura basada en servicios web y con un patrón de diseño implementado. Se pudo comprobar que, a pesar de la complejidad de la funcionalidad del componente, la integración fue sencilla. Debido a lo anterior, se propone integrar más componentes bajo el mismo enfoque para tener un sistema más amplio para la gestión de servicios web de aprendizaje.

- La integración del componente 2 (Gregorio, J. and Gutiérrez, G. (2015)), sistema de evaluaciones, se hizo a partir de un sistema que no tenía una arquitectura basada en servicios web y con elementos altamente acoplados, con el fin de probar la metodología planteada en el presente trabajo. Se pudo observar que la implementación de la metodología para obtener componentes con el propósito de integración con otros componentes, fue satisfactoria.
- Un reto importante en el trabajo logrado, fue integrar componentes en nodos físicos distintos. Para demostrar la integración se usaron los sistemas (Salinas Roman, H. (2017) y Gregorio, J. and Gutiérrez, G. (2015), los cuales se desplegaron en un servidor alojado en una maquina distinta a la de los clientes que usaron dichos componentes, los clientes a su vez fueron llamados por la página del portal alojado en otra máquina para demostrar que a pesar que las versiones de sistema operativo y de servidor fueran distintas, el resultado obtenido de la integración fue exitoso de acuerdo a las pruebas. Una sugerencia con propósitos de prueba, es desplegar futuros componentes dentro de sistemas operativos Linux.
- Una observación a tomar en cuenta es que un sistema integrador como el que se describió en este documento implemente un componente genérico para cambiar automáticamente las direcciones ip para enlazar los componentes externos. Una sugerencia es crear una aplicación web que pueda obtener la dirección ip de los equipos que contienen los componentes en forma de servicios web.
- El componente de sistema de evaluaciones, posee la capacidad de integrar futuras funcionalidades ya que se pueden incorporar más funcionalidades sin necesidad de alterar el código existente, aplicando el principio de diseño abierto-cerrado.

2. Metodología para integrar componentes de dedicados a la gestión de servicios web de aprendizaje.

La metodología definida a partir del estudio de trabajos relacionados, no es un objetivo descrito en el presente trabajo. Sin embargo, se puede considerar como aportación adicional, esto es resultado del proceso de investigación sobre los trabajos relacionados.

- Para obtener y definir la metodología de integración se hizo un estudio de trabajos relacionados donde se extrajeron las características más importantes para la integración de componentes mediante servicios web. Dentro del estudio de los trabajos relacionados, aproximadamente la mitad de ellos no tenían que ver con el dominio de servicios web de aprendizaje o de e-learning, por lo que se procedió a tomar sólo las fases utilizadas para integrar componentes. Se sugiere ampliar el estudio de trabajos relacionados.
- La metodología utilizada en el modo más amplio fue en el sistema de evaluaciones. El resultado sobre dicho componente fue satisfactorio de acuerdo a las pruebas llevadas a cabo y documentadas en el capítulo 5.
- Al definir la metodología de integración, se observó que algunos pasos pueden definirse en una manera más amplia para su mayor beneficio en los resultados obtenidos.

- Para lograr obtener una metodología integradora de componentes de gestión de servicios web de aprendizaje, con las bondades que ofrece un LMS, es necesario contemplar los componentes que actualmente la mayoría de los LMS ofrecen, como pueden ser monitoreo de calificaciones, plataforma para móviles, contenido calificado, entre otros.

6.2 Trabajos futuros

A continuación, se proponen algunos aspectos importantes para extender el trabajo presente.

- Generar componentes móviles para la gestión de servicios web de aprendizaje. Para obtener un mayor beneficio del sistema integrador de componentes, el usar componentes móviles dará la oportunidad de gestionar contenidos de SWA de una forma más práctica y accesible.
- Estudio de granulación de los componentes para determinar su nivel de reusabilidad.
- Integración del sistema integrador a las funcionalidades de LMS.

Anexo A WSDL de los componentes del Sistema de evaluación Gregorio, J. and Gutiérrez, G. (2015)

Endpoint	Information
<p>Service Name: {http://webServicesElearningResources/}wsUser</p> <p>Port Name: {http://webServicesElearningResources/}wsUserPort</p>	<p>Address: http://localhost:8084/Elearning_backEnd/wsUser</p> <p>WSDL: http://localhost:8084/Elearning_backEnd/wsUser?wsdl</p> <p>Implementation class: webServicesElearningResources.wsUser</p>
<p>Service Name: {http://webServicesElearningResources/}wsDomain</p> <p>Port Name: {http://webServicesElearningResources/}wsDomainPort</p>	<p>Address: http://localhost:8084/Elearning_backEnd/wsDomain</p> <p>WSDL: http://localhost:8084/Elearning_backEnd/wsDomain?wsdl</p> <p>Implementation class: webServicesElearningResources.wsDomain</p>
<p>Service Name: {http://webServicesElearningResources/}wsSubDomain</p> <p>Port Name: {http://webServicesElearningResources/}wsSubDomainPort</p>	<p>Address: http://localhost:8084/Elearning_backEnd/wsSubDomain</p> <p>WSDL: http://localhost:8084/Elearning_backEnd/wsSubDomain?wsdl</p> <p>Implementation class: webServicesElearningResources.wsSubDomain</p>
<p>Service Name: {http://webServicesElearningResources/}wsResources</p> <p>Port Name: {http://webServicesElearningResources/}wsResourcesPort</p>	<p>Address: http://localhost:8084/Elearning_backEnd/wsResources</p> <p>WSDL: http://localhost:8084/Elearning_backEnd/wsResources?wsdl</p> <p>Implementation class: webServicesElearningResources.wsResources</p>
<p>Service Name: {http://webServicesElearningResources/}wsEvaluation</p> <p>Port Name: {http://webServicesElearningResources/}wsEvaluationPort</p>	<p>Address: http://localhost:8084/Elearning_backEnd/wsEvaluation</p> <p>WSDL: http://localhost:8084/Elearning_backEnd/wsEvaluation?wsdl</p> <p>Implementation class: webServicesElearningResources.wsEvaluation</p>

Referencias

- [1] Roberto Baelo Álvarez. (Julio 2009). EL E-LEARNING, UNA RESPUESTA EDUCATIVA A LAS DEMANDAS DE LAS SOCIEDADES DEL SIGLO XXI. Pixel-Bit, 35, 87-96.
- [2] Carlos Andrés Morales Machuca. (no especificada). Estado del Arte: Servicios Web. None, 1, 1-2.
- [3] Pressman, R., Campos Olguín, V. and Enríquez Brito, J. (2010). Ingeniería del software. 1st ed. México: McGraw-Hill.
- [4] Aprendeonline.udea.edu.co. (2016). Recursos educativos digitales: Conceptos básicos. [online] Available at:
http://aprendeonline.udea.edu.co/boa/contenidos.php/d211b52ee1441a30b59ae008e2d31386/845/estilo/aHR0cDovL2FwcmVuZGVlbmxpbmVhLnVkZWVhZWR1LmNvL2VzdGlzb3MvYXp1bF9jb3Jwb3JhdGl2by5jc3M=/1/contenido/#referencia_3a [Accessed 20 Nov. 2016].
- [5] Hamdan, S. and Alramouni, S. (2015). A Quality Framework for Software Continuous Integration. Procedia Manufacturing, 3, pp.2019-2025.
- [6] Snell, J., Tidwell, D. and Kulchenko, P. (2002). Programming Web services with SOAP. Sebastopol, CA: O'Reilly & Associates.
- [7] Guzmán Ruiz, M. (2006). Composición de servicios web. Maestría. Centro Nacional de Investigación y Desarrollo Tecnológico.
- [8] Orozco Ayometzi, M. (2009). Extensión al sistema de composición de servicios web utilizando diagramas de actividad. Maestría. Centro Nacional de Investigación y Desarrollo Tecnológico.
- [9] Pattnayak, J. and Pattnaik, S. (2016). Integration of Web Services with E-Learning for Knowledge Society. Procedia Computer Science, 92, pp.155-160.
- [10] Raspopović, M., Cvetanović, S., Stanojević, D. and Opačić, M. (2016). Software architecture for integration of institutional and social learning environments. Science of Computer Programming, 129, pp.92-102.
- [11] Judrups, J. (2015). Analysis of Knowledge Management and E-Learning Integration Models. Procedia Computer Science, 43, pp.154-162.
- [12] Yan, W., Tan, P. and Lee, E. (2008). A Web Services-enabled B2B integration approach for SMEs. 2008 6th IEEE International Conference on Industrial Informatics.
- [13] Chen, Y. and Romanovsky, A. (2008). Improving the Dependability of Web Services Integration. IT Professional, 10(3), pp.29-35.
- [14] Su-Hua Wang and Dengjie Chen (2014). A Cloud Framework for Enterprise Application Integration on SOA. International Conference on Software Intelligence Technologies and Applications & International Conference on Frontiers of Internet of Things 2014.

- [15] Garcia, N., Espíndola, D., Botelho, S., Albrecht, A., Israel, E., Frazzon, E., Hellingrath, B., Schroeder, G. and Pereira, C. (2016). Web-Service Interface-based IMS-SPSC Integration. IFAC-PapersOnLine, 49(30), pp.192-197.
- [16] Mahmoud, T., Rapp, B. and van Vliet, S. (2013). Web Service Integration within Next Generation CEMIS. Procedia Technology, 9, pp.282-290.
- [17] eduNEXT | Open edX as a Service. (2016). eduNEXT - Your ideal cloud LMS provider for Open edX. [online] Available at: <https://www.edunext.co/> [Accessed 7 Oct. 2016].
- [18] Spotlms.info. (2016). SPOT LMS, Portal digital de aprendizaje en la nube, gratuito y fácil de usar. [online] Available at: <https://www.spotlms.info> [Accessed 7 Oct. 2016].
- [19] CAE Computer Aided E-learning. (2016). CAE - Expertos en formación e-learning. [online] Available at: <http://www.cae.net/> [Accessed 7 Oct. 2016].
- [20] Paradisosolutions.com. (2016). Best LMS - Learning Management System | Paradiso LMS. [online] Available at: <https://www.paradisosolutions.com/> [Accessed 7 Oct. 2016].
- [21] Gregorio, J. and Gutiérrez, G. (2015). Informe de residencias: Análisis de Sistema de evaluaciones. Cuautla, Morelos.
- [22] Salinas Roman, H. (2017). Marco Orientado a Objetos para Medir la Calidad en Servicios Web de Aprendizaje. Maestría. Centro Nacional de Investigación y Desarrollo Tecnológico.
- [23] tecnología, D. and datos, B. (2018). Definición de Entidad. [online] Alegs.com.ar. Available at: <http://www.alegsa.com.ar/Dic/entidad> [Accessed 19 Jun. 2018].