



Facultad 6

**Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas.**

Título: Servicio Web para el intercambio de metadatos geográficos entre aplicaciones SIG del centro GEYSED y las bases de datos de SyGMe.

Autor: Bárbaro Orestes Ramos Díaz

Tutor: Ing. Carlos Enrique Ramírez Martín

Ciudad de La Habana, julio del 2013

“Año 54 de la Revolución”



“O nosotros somos capaces de destruir con argumentos las ideas contrarias, o debemos dejar que se expresen. No es posible destruir ideas por la fuerza, porque esto bloquea cualquier desarrollo libre de la inteligencia.”

Dedicatoria

Quiero dedicar este sueño hecho realidad principalmente a mi bisabuela Ester que no se encuentra viva pero sé que de una forma u otra está aquí pendiente de mí, y sé que está muy orgullosa y contenta por este logro donde quiera que esté.

A mi abuelita que esto se lo debo principalmente a ella, por ser mi principal fuente de lucha en la vida, estar siempre disponible para todos mis caprichos y ser esa persona que me entrega todo su cuerpo y alma en cada suspiro, y que si hoy soy ingeniero te lo debo a ti.

A todos los que me quieren y de una forma u otra se preocuparon por mí e hicieron posible que sea un profesional.

Declaración de Autoría

Agradecimientos

A Dios, Elegua y todos los santos por darme la fe que tanto me ayudó para salir siempre hacia adelante.

A la universidad por permitirme ser un profesional y superarme cada día.

A mi tutor que muy quisquilloso pero lo hacía con toda la intención era lograr una documentación de gran calidad.

A mi tribunal que nunca perdió el contacto y la preocupación porque yo cumpliera mi objetivo especialmente al presidente Yoenis.

A mi oponente Alain que con sus indicaciones y sugerencias pude culminar con éxito la aplicación y la documentación correspondiente.

A mi mamá por ser mi mejor amiga y ser esa persona que desde pequeño me dio todo el amor, comprensión y cariño del mundo, por siempre apoyarme en los momentos más difíciles de mi vida y por permitirme hoy ser la persona que soy.

A mi papá por estar siempre pendiente de mí a pesar de no vivir conmigo, por mostrarme el camino muchas veces cuando estaba perdido y darme todo el apoyo de padre, hermano y amigo.

A mis hermanos por estar siempre conmigo y compartir los mejores momentos de mi vida (Yuni, Yenny y Yon).

A mis primos por compartir los mejores momentos de mi infancia y ser mis mayores confidentes, especialmente Deyvis, Jaime, Diosvaldo y Caruso.

A mis tíos por siempre darme todo el amor y cariño de un hijo (Figurin, Vivian, Lazara, Russo, Yamile y Odalis).

A mi familia de Guanabacoa por estar siempre pendiente de mi (Yanier, Deysi,)

A toda mi familia en general por siempre apoyarme sacando lo mejor de mí, y mantener siempre ese espíritu de unidad y confianza.

A mi segunda gran familia que me apoya, me quiere y me hizo miembro de ella, que cualquier problema es capaz de resolver, a todos los quiero igual (Alfredo, Cosme, Dariel, Regla, Juana, Damian y Paulito).

A todas las chicas del 93 especialmente a Aida y Darcy a las cuales mortifique y pedí consejos durante estos 5 años.

Declaración de autoría

Declaro por este medio que yo, Bárbaro Orestes Ramos Díaz con carné de identidad 89102027305, soy el autor de este trabajo y que autorizo a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio, así como los derechos patrimoniales con carácter exclusivo.

Para que así conste, firma la presente declaración jurada de autoría en La Habana a los ____ días del mes ____ del año _____.

Bárbaro Orestes Ramos Díaz

Autor

Carlos Enrique Ramírez Martín

Tutor

Resumen

Con la evolución de la ciencia y la tecnología, el desarrollo de la Informática se ha incrementado considerablemente. Como resultado de este avance, los proyectos de desarrollo de software se han acrecentado a gran escala. Como parte de la realización de estos proyectos en la Universidad de las Ciencias Informáticas (UCI) surge la Línea de Productos de Software Aplicativos SIG, encargada de desarrollar Sistemas de Información Geográfica para la Web, debido a la necesidad que tiene la LPS de utilizar metadatos geográficos para organizar sus base de datos espaciales y las dificultades que presenta la herramienta SyGMe para el intercambio de información con otros sistemas surge la necesidad de desarrollar un mecanismo de comunicación entre dicha herramienta y los SIG desarrollados en el centro GEYSED.

La presente investigación tiene como objetivo principal desarrollar un Servicio Web que permita la organización e intercambio de metadatos geográficos entre aplicaciones SIG pertenecientes al centro GEYSED y las bases de datos de SyGMe. Se analizan los conceptos teóricos relacionados a la investigación, las principales técnicas de asociación de metadatos, así como los servicios que se emplean para el intercambio de metadatos geográficos y las herramientas utilizadas para el desarrollo del Servicio Web. Se realizan las actividades correspondientes a la metodología XP para el desarrollo del software, generando los artefactos necesarios. Se valida el componente de software obtenido a partir de la realización de pruebas que certifican la calidad del mismo.

Palabras clave:

Servicios Web, intercambio, metadatos geográficos.

ÍNDICE DE CONTENIDOS

INTRODUCCIÓN ----- 1

CAPÍTULO 1: FUNDAMENTOS TEÓRICOS DE LA INVESTIGACIÓN ----- 6

1.1 Introducción ----- 6

1.2 Conceptos asociados al dominio del problema ----- 6

1.3.3 Servicios Web----- 10

1.3.1 Modelo de Arquitectura de un Servicio Web----- 11

1.3.2 Ventajas de los Servicios Web----- 11

1.3.3 ¿Por qué un Servicio Web? ----- 13

1.4 Servicio CSW ----- 13

1.5 Situación actual del problema. ----- 15

1.6 Análisis de soluciones existentes ----- 17

1.7 Conclusiones parciales ----- 20

CAPÍTULO 2: Tendencias y tecnologías actuales a desarrollar ----- 22

2.1 Introducción ----- 22

2.2 Propuesta de herramientas----- 22

2.2.1 XML----- 22

2.2.2 Simple Object Access Protocol (SOAP) ----- 23

2.2.3	Web Services Description Language (WSDL) -----	23
2.3	Metodología de Desarrollo -----	24
2.3.1	Programación extrema (XP) -----	25
2.4	Arquitectura de software -----	28
2.4.1	Arquitectura cliente-servidor -----	28
2.5	Servidores Web -----	29
2.5.1	Servidor Apache 2.2 -----	30
2.6	Lenguaje de programación Web -----	31
PHP 5.2.12	-----	31
2.7	Entorno Integrado de Desarrollo (IDE) -----	32
2.7.1	NetBeans 7.3 -----	32
2.9	SOA Client -----	32
2.10	Conclusiones parciales -----	33
CAPÍTULO 3: Presentación de la solución propuesta -----		34
3.1	Introducción -----	34
3.2	Fase de Exploración -----	34
3.2.1	Historias de Usuarios -----	34
3.2.2	Relación de las Historias de Usuarios -----	35

3.3	Fase de Planificación -----	35
3.3.1	Plan de Iteraciones -----	36
3.3.2	Estimación de esfuerzos.-----	37
3.3.3	Plan de duración de las iteraciones.-----	37
3.3.4	Plan de entrega-----	38
3.4	Conclusiones parciales -----	38
CAPÍTULO 4: Construcción y Validación de la solución propuesta-----		40
4.1	Introducción -----	40
4.1	Diseño de la solución propuesta.-----	40
4.2.1	Tarjetas CRC. -----	40
4.3	Patrones Arquitectónicos-----	42
4.4.1	Patrón arquitectónico N-Capas -----	42
4.4	Patrones de Diseño-----	44
4.4.1	Patrones GRASP-----	44
4.5	Desarrollo de las iteraciones. -----	45
4.5.1	Iteración 1:-----	46
4.5.2	Iteración 2:-----	47
4.6	Requisitos no funcionales.-----	48

4.7	Pruebas.....	49
4.7.1	Pruebas de aceptación.....	50
4.8	Conclusiones Parciales.....	52
	CONCLUSIONES GENERALES.....	53
	RECOMENDACIONES.....	54
	REFERENCIAS BIBLIOGRÁFICAS.....	55
	BIBLIOGRAFÍA CONSULTADA.....	61
	GLOSARIO DE TÉRMINOS.....	66

ÍNDICE DE TABLAS

Tabla 1: HU Recuperar metadatos asociados a una capa.....35

Tabla 2: Estimación de Esfuerzos37

Tabla 3: Plan de duración de Iteraciones38

Tabla 4: Plan de entrega.....38

Tabla 5: Plantilla para Tarjetas CRC41

Tabla 6: CRC_ Servicio.....42

Tabla 7: Abordadas en la primera iteración46

Tabla 8: Tarea #1 de la HU #1.47

Tabla 9: Tarea #2 de la HU #1.47

Tabla 10: Abordadas en la segunda iteración.47

Tabla 11: Tarea #1 de la HU #448

Tabla 12: HU1_PA152

Tabla 13: HU1_PA252

ÍNDICE DE ILUSTRACIONES

Ilustración 1: Modelo de arquitectura de un Servicio Web (Pereira, 2003)..... 11

Ilustración 2: Interfaz GeoNetwork (Yurtseven, 2007) 18

Ilustración 3: Interfaz Portal del Instituto Geográfico de España. (Ibero) 19

Ilustración 4: Interfaz Geosur. (Geosur) 20

Ilustración 5: Arquitectura Cliente-Servidor 29

Ilustración 6: Funcionamiento de un Servicio Web (Machuca, 2012)..... 43

Ilustración 7: Prueba de aceptación 51

INTRODUCCIÓN

La creación de Internet supuso una revolución sin precedentes en el mundo de la informática y de las comunicaciones. La necesidad de lograr la interoperabilidad y permitir compartir información entre distintas plataformas de software y hardware, trajo consigo la utilización de un lenguaje común de intercambio de información que aprovechara los estándares existentes. Es entonces que surgen los Servicios Web basados en XML, cuya principal característica es que ayudan al desarrollo de sistemas heterogéneos, pudiendo compartir información y código; es decir, logran la interoperabilidad y proporcionan herramientas que ayudan al desarrollo y mantenimiento de aplicaciones. (W3C Consortium, 2006)

La evolución de la Web conllevó al aumento y desarrollo de soluciones informáticas, debido al elevado volumen de información que estas generaban y la necesidad de mantener una organización en los datos, surgen los metadatos. Éstos alcanzan nuevas áreas de aplicación donde juegan un papel primordial, principalmente para localizar y gestionar recursos, permitiendo a los usuarios, conocer los datos sobre los recursos de su interés. Los metadatos poseen gran importancia cuando se maneja documentación, a la vez que permiten verificar la autenticidad, fiabilidad e integridad de los documentos a lo largo del tiempo. (García, 2008)

Uno de los principales servicios que se brindan en la actualidad son los relacionados con la gestión y análisis de la información geográfica, donde los metadatos geográficos tienen un papel fundamental. Estos surgen debido a la necesidad de agrupar los elementos más relevantes en la gestión de la información geográfica, evitando así, el almacenamiento de la información referente a los datos geoespaciales en variados formatos, por la dificultad que representa para su manipulación.

Las metas principales de la Industria Cubana del Software radican en: informatizar la sociedad, elevar los niveles de desarrollo de software y exportar productos de software. La Universidad de las Ciencias Informáticas (UCI), como uno de los principales centros de capacitación y de desarrollo de software en el país, contribuye al desarrollo de la economía y de la sociedad, en general, dado el avance que han alcanzado los proyectos que se desarrollan en ella. Como parte del aumento de aplicaciones desarrolladas en la UCI, los proyectos que se llevan a cabo exigen crear aplicaciones más complejas, con menos tiempo y presupuesto que antes. Crear estas aplicaciones requiere, en muchos casos, funcionalidades antes implementadas como

parte de otros sistemas. Una de las estrategias es facilitar su integración garantizando un óptimo aprovechamiento de recursos y tecnologías.

En el centro de desarrollo y producción de software Geoinformática y señales digitales (GEYSED), perteneciente a la UCI, específicamente en el departamento Geoinformática, se encuentra la Línea de Productos de Software (LPS) Aplicativos SIG, el cual es el encargado del desarrollo de Sistemas de Información Geográfica para la Web. Debido al aumento del desarrollo de SIG en la LPS Aplicativos SIG, la organización y distribución de las capas geográficas se ve afectada, dado el crecimiento considerable de las mismas. Obtener la información de las capas geográficas resulta una tarea ardua, pues no se dispone de la utilización de metadatos geográficos, los cuales ayudan a organizar, entender y agilizar las funcionalidades relacionadas con la información geográfica. LPS Aplicativo SIG no se encuentran en las mejores condiciones para satisfacer las necesidades correspondientes al trabajo y búsqueda de información a través de metadatos geográficos, puesto que las bases de datos que ella dispone son solamente espaciales y no poseen una estructura definida para trabajar con este tipo de información.

En el centro GEYSED se desarrolló la herramienta SyGMe, la cual es la encargada de gestionar los metadatos geográficos asociados a las bases de datos espaciales, haciendo uso de la norma ISO 19115. El principal problema que presenta esta herramienta es que la información no puede ser consultada por otros sistemas, lo cual dificulta la interoperabilidad de la herramienta SyGMe y las demás aplicaciones que necesiten realizar algún tipo de trabajo con los metadatos geográficos. Para obtener la información de los metadatos geográficos de la herramienta SyGMe, se debe realizar una integración a nivel de bases de datos, configurando una conexión y realizando las consultas a través del propio código, del sistema a desarrollar. Esto trae como problema principal que los mismos desarrolladores deban conocer la estructura en que está constituida la base de datos de SyGMe, el formato en que se envían los datos y la manera en la que se reciben e interpretan los mismo. Estas cuestiones afectan la seguridad de los datos al existir dentro de los sistemas a desarrollar la configuración de la conexión a la base de datos donde se encuentran dichos metadatos. Desarrollar la opción que permita en un SIG el intercambio de la información de los metadatos geográficos, resulta engorroso, teniendo en cuenta las condiciones antes mencionadas, el código tiende a crecer, surgen irregularidades en el mismo y el producto final puede no quedar de la manera deseada por el cliente.

Debido a la situación problemática antes descrita, se define **como problema a resolver**: ¿Cómo garantizar el intercambio de metadatos geográficos entre las aplicaciones SIG desarrolladas en el centro GEYSED y las bases de datos de SyGMe de forma estandarizada?

De acuerdo al problema planteado se definen como **objeto de estudio** los procesos para el intercambio de metadatos geográficos entre aplicaciones.

El **campo de acción** en el que se verá enmarcado el estudio estará dado por los procesos para el intercambio de metadatos geográficos entre aplicaciones en el centro GEYSED.

Como **objetivo general** desarrollar un Servicio Web que permita la comunicación y el intercambio de metadatos geográficos entre las aplicaciones SIG desarrolladas en el centro GEYSED y las bases de datos de SyGMe.

Se plantea como **idea a defender** que con el desarrollo de un Servicio Web que permita el intercambio de metadatos geográficos, se garantizará la interoperabilidad y la organización de información geográfica entre las aplicaciones SIG del centro GEYSED de forma estandarizada.

Para dar cumplimiento al objetivo general de la presente investigación se definieron las siguientes **tareas**:

1. Identificar los conceptos asociados al entorno donde está enmarcado el problema.
2. Seleccionar y fundamentar las tendencias y tecnologías actuales a utilizar.
3. Especificación de requisitos no funcionales de software e historias de usuario.
4. Definir o identificar las principales funcionalidades a implementar en el Servicio Web.
5. Elaboración de los artefactos de la etapa de diseño correspondientes.
6. Implementar las funcionalidades del Servicio Web propuesto.
7. Aplicación de pruebas de aceptación, pruebas unitarias y análisis de sus resultados para validar la implementación del componente.

Métodos Teóricos:

Analítico-Sintético: Este método se utilizó para realizar el estudio detallado de los temas relacionados con la búsqueda de metadatos geográficos y hacer una recopilación de la bibliografía a utilizar.

Análisis histórico – lógico: Este método permitió el análisis histórico del trayecto y lógica interna que han seguido en su desarrollo los procesos de intercambio y búsqueda de los metadatos geográficos.

Modelación: Este método permitió realizar todos los diagramas asociados al análisis del Servicio Web para el intercambio de metadatos geográficos y agilizar el desarrollo de la investigación.

Métodos Empíricos:

Observación: Este método se utilizó durante todo el proceso de investigación para percibir los hechos y características que se manifiestan alrededor de la problemática planteada, los procesos de gestión analizados así como también arribar a conclusiones que permitan modelar y aplicar la solución que se propone.

Sistémico: Este método permitió a partir de la manera en la que se relacionan los componentes de los procesos de intercambio de los metadatos geográficos ya elaborados, llegar a conclusiones y generalizaciones sobre el funcionamiento del Servicio Web en cuestión.

Se espera como **posible resultado** de la presente investigación, contar con un Servicio Web que permita la comunicación y el intercambio de metadatos geográficos entre las aplicaciones SIG desarrolladas en el Centro GEYSED y las bases de datos de SyGMe.

El documento se estructura de la siguiente forma:

Capítulo 1. Fundamentación teórica. Se referencian los conceptos principales, al estado del arte del tema que se aborda, así como se analizarán otras posibles soluciones para el problema científico que se plantea en este trabajo. Se mostrará además la descripción de lo que será el entorno del negocio y se encontrará descrita la situación problemática.

Capítulo 2. Tendencias y tecnologías actuales a desarrollar. En este capítulo se explican las principales tecnologías, lenguajes de programación y herramientas que se utilizarán para la construcción de la solución

propuesta, así como las ventajas de utilizarlas. Se analiza la metodología de desarrollo de software que se usó durante el análisis y que además se empleará para la implementación del software.

Capítulo 3: Presentación de la solución propuesta. En este capítulo se abordará las fases iniciales de la Programación Extrema (en lo adelante XP). En la primera fase, exploración, el cliente definirá las historias de usuario, para la familiarización con las herramientas y el entendimiento del software y en la segunda fase, planificación, se estima el esfuerzo que se requiere para cada historia de usuario y se realiza una programación de acuerdo a esa estimación.

Capítulo 4: Construcción y Validación de la solución propuesta. En este capítulo se plantea la construcción propuesta en el capítulo anterior abordando las fases restantes que contiene la XP, la fase de construcción y la de prueba. Serán descritas las tarjetas CRC, se describen las iteraciones hechas en la etapa de construcción de la aplicación, las actividades generadas por las historias de usuarios y estándares del diseño, así como se especifican los requerimientos no funcionales del Servicio Web. Se ejecuta un análisis exhaustivo del Servicio Web creado, a partir de la construcción y aplicación de un conjunto de pruebas que garanticen dada la corrección de errores, el correcto funcionamiento del software creado.

CAPÍTULO 1: FUNDAMENTOS TEÓRICOS DE LA INVESTIGACIÓN

1.1 Introducción

En el presente capítulo, se tratan aspectos relacionados con el diseño teórico de la investigación, el objeto de estudio y temas referidos al marco conceptual que se aborda en el trabajo, los conceptos y a la tecnología que permite el desarrollo de Servicios Web. Se hace además, un análisis de las normas internacionales que presiden el proceso de intercambio de metadatos geográficos. A su vez, se hace un estudio del estado del arte, profundizándose en la situación existente en cuanto a normas, estándares y soluciones que enmarcan el intercambio de los metadatos geográficos actualmente. Se enfatiza también en la situación problemática tratada.

1.2 Conceptos asociados al dominio del problema

Los términos tratados en este epígrafe fueron seleccionados de forma tal que se puedan comprender mejor los principales aspectos que conforman esta investigación para ello es importante conocer algunos conceptos asociados al dominio del problema.

La Web

Para un correcto entendimiento de los Servicios Web es importante conocer el medio donde estos se despliegan: la Web. El desarrollo de esta no se debe a una sola persona, pero si se busca un padre para ella, sería Tim Bernes-Lee, pues a él se le deben los tres elementos fundamentales que fueron claves en su nacimiento.

HTML¹ para crear los contenidos de la Web.

¹ HyperText Markup Language (en español lenguaje de marcado de hipertexto)

HTTP² como protocolo de comunicación entre los ordenadores de la Web, encargado de la transferencia de las páginas Web y demás recursos.

URL³ como medio de localización (direccionamiento) de los distintos recursos de la Web.

Con el paso del tiempo y unido al incremento exponencial del uso de la Web, otros elementos se fueron sumando y estos a su vez fueron mutando la forma en la que desarrolladores y usuarios finales compartían sus datos en la red.

Información geográfica

“La información geográfica es un conjunto de datos acerca de algún evento, hecho o fenómeno, que organizados en un contexto determinado, tienen su significado, cuyo propósito puede reducir la incertidumbre o incrementar el conocimiento acerca de algo.” (Thompson, 2008)

“La Información Geográfica se especifica a su vez como: aquella información que describe fenómenos vinculados de forma directa o indirecta a una ubicación en la superficie terrestre o sus proximidades. En muchas ocasiones esta información cuenta también con datos de tiempo.” (Manso Callejo, 2003)

Partiendo de las anteriores definiciones se determina la información geográfica, como al conjunto de datos espaciales que brinda conocimientos sobre la posición explícita e implícita del espacio descrito y ofrece información sobre sucesos o hechos de carácter geográfico.

Sistema de Información Geográfica

“Un Sistema de Información Geográfica (SIG), es la integración de hardware, software y usuarios que mediante operaciones diseñadas para soportar la captura, el manejo, la manipulación, el análisis, modelado y despliegue de datos espacialmente referenciados, permiten la solución de problemas complejos del manejo y proyección

² Protocolo de transferencia de Hipertextos para la transmisión de documentos en formato HTML entre el servidor y el cliente.

³ Uniform resource locator (en español localizador de recursos uniforme)

territorial. Los Sistemas de Información Geográfica como herramienta principal para la gestión de datos geográficos se convierten en la opción fundamental para la manipulación y análisis de datos espaciales con el fin de automatizar procesos y consultar estados geográficos de forma rápida y precisa.” (Cisneros Díaz, 1994)

Por tanto, se define un Sistema de Información Geográfica a la combinación del hombre, técnicas informáticas y cómputos para conformar un sistema que soporte el manejo, gestión y análisis de información geoespacial para permitir la toma de decisiones ante los problemas que se presentan a la sociedad.

Infraestructura de Datos Espaciales

“Una Infraestructura de Datos Espaciales (IDE) se puede definir como el conjunto de datos, servicios, metodologías, normas, estándares y acuerdos, que permiten visualizar, superponer, consultar y analizar la información geográfica publicada en Internet, según estándares bien definidos.” (Rodríguez, 2006).

“La Infraestructura de Datos Espaciales está formada por un grupo de entidades que definen e implementan una serie de estrategias o políticas a seguir. Supone la forma en la que se utilizarán los datos espaciales para un mayor entendimiento entre plataformas de desarrollo con el fin de que estos datos sigan una misma política de desarrollo a la hora de ser utilizados.” (Bohorquez, 2005)

Se concluye, que una IDE, es un grupo de servicios y normas que mediante la utilización de las tecnologías de la información facilita el acceso, registro y gestión de los datos georreferenciados.

Open GIS Consortium

Según (Nebert, 2004) plantea *“La especificación del Open GIS Catalog Service de OGC⁴, que está en camino de convertirse en un estándar, establece cómo deben estructurarse e implementarse los servicios de*

⁴ Organización para la gestión y el procesamiento abierto y operable de la información geográfica. Fundada en 1947 por entidades académicas, gubernamentales e industriales.

catalogación y de búsqueda de metadatos geoespaciales, estableciendo el subconjunto mínimo de metadatos que deben ser interrogables.”

OGC es una organización internacional que establece especificaciones, sobre la estructura de los datos y servicios geográficos para conseguir un geoprocesamiento abierto e interoperable. Además, esta organización participa en el desarrollo de los estándares de datos geográficos de ISO. (Nebert, 2004)

Mapa

Un mapa es una representación geográfica de una parte de la superficie terrestre, en la que se presenta información relativa a una ciencia determinada. (Real Academia Española)

Metadatos geográficos

Según la Revista Internacional de Ciencias de la Tierra; *“los metadatos de información geográfica describen el contenido, la calidad, el formato y otras características que lleva asociadas un recurso, constituyendo un mecanismo para caracterizar datos y servicios de forma que usuarios (y aplicaciones) puedan localizarlos y acceder a ellos”*. (Caballeros, 2008) Dan respuestas a preguntas del tipo:

- El qué: nombre y descripción del recurso.
- El cuándo: fecha de creación de los datos, periodos de actualización, etc.
- El quién: creador de los datos.
- El dónde: extensión geográfica.
- El cómo: modo de obtención de la información, formato, etc. (Sánchez Maganto, 2008)

“Los metadatos son “datos sobre los datos” Describen el contenido, la calidad, la condición y otras características de los datos. Ayudan a una persona o sistema inteligente a localizar y entender los datos espaciales disponibles.” (Manso Callejo, 2003)

Se puede concluir a partir de las anteriores citas, que un metadato geográfico trata de caracterizar los datos espaciales respondiendo al quién, qué, cuándo, dónde, por qué y cómo de estos datos. Contribuyen a

entender y describir la información geográfica, proporcionando el acceso a los datos de forma más rápida y organizada.

1.3.3 Servicios Web

Un Servicio Web (en inglés, Web services) es una tecnología que utiliza un conjunto de protocolos y estándares tales como SOAP, WSDL y UDDI que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes y ejecutadas sobre cualquier plataforma, pueden utilizar los Servicios Web para intercambiar datos en redes de ordenadores como Internet. (W3C Consortium, 2006)

Al conjunto de estándares 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:

- Servicio de transporte: Es el encargado del transporte de los mensajes entre aplicaciones sobre la red. Incluye varios protocolos del nivel de aplicación (HTTP, FTP⁵, SMTP⁶). Para el Servicio Web en cuestión el transporte será a través del protocolo HTTP.
- Mensajería XML: Es el conjunto encargado de la codificación de los mensajes en XML estándar y pueda así ser interpretado en cualquiera de los nodos de la red.
- Descripción del servicio: El Servicio Web debe contar con una interfaz pública la cual es descrita por un formato llamado WSDL (Web Services Description Language).

⁵ Protocolo de red para la transferencia de archivos entre sistemas conectados a una red TCP.

⁶ Protocolo de la capa de aplicación. Protocolo de red basado en texto, utilizado para el intercambio de mensajes de correo electrónico entre computadoras u otros dispositivos.

- Descubrimiento de Servicios: Especificación y ubicación de los servicios para su uso, alojados en un servidor público llamado UDDI (Universal Description Discovery and Integration).

1.3.1 Modelo de Arquitectura de un Servicio Web

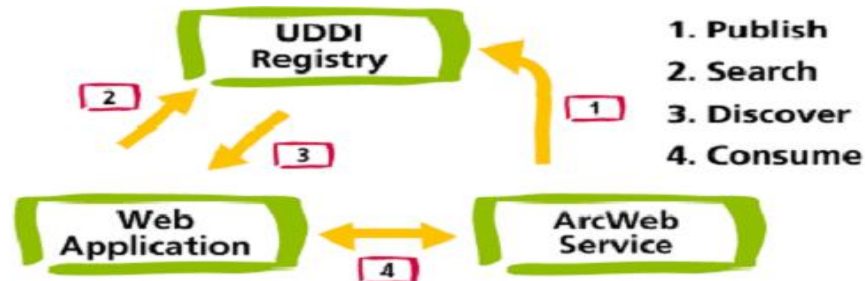


Ilustración 1: Modelo de arquitectura de un Servicio Web (Pereira, 2003)

En este modelo se identifican las acciones que afectan el funcionamiento de un Servicio Web, así:

- 1 Una vez desarrollado el Servicio Web, se registra. Usando **UDDI**.
- 2 Un cliente, en este caso una aplicación Web, lanza una solicitud para conseguir una lista de los Servicios Web que satisfagan las condiciones de búsqueda.
- 3 El cliente recibe información a partir de la cual puede determinar la petición que más se ajuste a su necesidad.
- 4 El cliente lanza la petición al Servicio Web y recibe la respuesta. Los Servicios Web como sistemas abiertos, logran su funcionamiento adoptando tecnologías basadas en XML como:

1.3.2 Ventajas de los Servicios Web

- No está asociado con ningún lenguaje:

Si bien tiene como parámetro XML, los desarrolladores involucrados en nuevos proyectos pueden elegir desarrollar con el último y mejor lenguaje de programación que exista, pero los desarrolladores responsables de mantener antiguas aflicciones heredadas, podrían no hacer esta elección sobre el lenguaje de

programación que utilizan. SOAP no especifica una API⁷, por lo que la implementación de la misma se deja al lenguaje de programación.

- No se encuentra fuertemente asociado a ningún protocolo de transporte:

Su especificación no describe cómo se deberían asociar los mensajes de SOAP con HTTP. Un mensaje de SOAP no es más que un documento XML, por lo que puede transportarse utilizando cualquier protocolo capaz de transmitir texto.

- Aprovecha los estándares existentes en la industria:

Los principales desarrolladores de este protocolo evitaron intencionadamente reinventar las cosas, optaron por extender los estándares existentes para que coincidieran con sus necesidades, por ejemplo, aprovecha XML para la codificación de los mensajes, en lugar de utilizar su propio sistema, que ya están definidas en la especificación de esquemas de XML.

- Permite la interoperabilidad entre múltiples entornos:

Se desarrolló sobre los estándares existentes de la industria, por lo que las aplicaciones que se ejecuten en plataformas con dichos estándares pueden comunicarse mediante mensaje SOAP con aplicaciones que se ejecuten en otras plataformas. Por ejemplo, una aplicación de escritorio que se ejecute en una computadora, puede comunicarse con una aplicación, ejecutándose en un mainframe capaz de enviar y recibir XML sobre HTTP.

Las ventajas de desarrollar Servicios Web se miden en los ahorros significativos en tiempo de desarrollo, en el costo y en recursos de ordenador, pues el tratamiento de la información se realiza en los ordenadores donde el

⁷API: Interfaz de programación de aplicaciones (del inglés Application Programming Interface) es el conjunto de funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

Servicio Web es localizado y devuelve los resultados al usuario local. Para el caso de los sistemas de Información Geográfica un desarrollador no necesita mantener herramientas de aplicación SIG o los datos geográficos asociados sobre su sistema local.

Los servicios relacionados con la organización y búsqueda de metadatos tienen un gran impacto para las empresas desarrolladoras de SIG, porque los metadatos permiten: recuperar recursos, administrar documentos, gestionar derechos, valorar contenidos, conocer el estado de los archivos, conocer productos y servicios, esquemas de bases de datos, control o descripción de procesos. Ayudan a organizar y mantener la inversión en datos de las organizaciones, brindándonos la información en forma de catálogos estandarizados. Gracias a ellos es posible conocer la existencia de los diferentes conjuntos de datos, y pueden localizarse aquellos que están disponibles, evaluando cuáles son relevantes para nuestras actividades. La redacción de metadatos descriptivos, fomenta la disponibilidad y uso de los datos más allá del ámbito o disciplina en que fueron creados. También son una buena forma de promoción de los datos creados por una institución.

1.3.3 ¿Por qué un Servicio Web?

Los Servicios Web exponen funcionalidad útil a los usuarios Web mediante un protocolo Web estándar. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferente y ejecutada sobre cualquier plataforma pueden utilizar estos servicios para intercambiar datos en redes de ordenadores.

Se decidió crear un Servicio Web porque aporta interoperabilidad entre aplicaciones de software independientemente de sus propiedades o de las plataformas sobre las que se instalen, permite que servicios y software de diferentes compañías geográficamente distantes puedan ser combinados fácilmente para proveer servicios integrados, además de que fomentan los estándares y protocolos basados en texto, que hacen más fácil acceder a su contenido y entender su funcionamiento. Otra razón para usar estos servicios es que son muy prácticos ya que pueden aportar gran independencia entre la aplicación que usa el Servicio Web y el propio servicio. De esta forma, los cambios a lo largo del tiempo en uno no deben afectar al otro.

1.4 Servicio CSW

El Catálogo de metadatos forma parte de uno de los tres servicios fundamentales que debe existir en una Infraestructura de Datos Espaciales (IDE). Este servicio consiste en un buscador de datos geográficos

existentes en territorios y medio ambientes, definiendo una interfaz común para la recuperación, captura y consulta de metadatos referentes a datos, servicios y recursos geográficos. El protocolo estándar que utiliza el Catálogo o Buscador, es el llamado **Servicio CSW**, estándar definido por la organización Open Geospatial Consortium. (Valenciana, 2010)

Permite la publicación y búsqueda de información (metadatos) que describen datos, servicios, aplicaciones y en general todo tipo de recursos. Los servicios de catálogo son necesarios para proporcionar capacidades de búsqueda e invocación sobre los recursos registrados dentro de una IDE. (Fernández, 2008)

En los últimos años han proliferado los clientes interoperables, básicamente visores, Web o de escritorio, que gestionaban múltiples conexiones a diferentes servicios OGC, tales como Web Map Service (WMS) y en menor medida Web Feature Service (WFS) sin embargo, este fenómeno, no se ha producido en el campo de los catálogos de metadatos, pieza fundamental dentro de las Infraestructuras de Datos Espaciales (IDEs).

El contenido principal del estándar es:

- Modelo de información abstracto: en el que se realiza una definición del lenguaje de consulta y del esquema básico de metadatos, proporcionando un vocabulario común para búsqueda, recuperación, visualización y asociación
 - Lenguaje de consulta: define el mínimo conjunto de tipos de datos y operaciones de consulta que debe soportar un CSW, la sintaxis está basada en SQL y es extensible
 - Esquema básico de metadatos: establece las propiedades de consulta y respuesta mínimas y recomienda Dublin Core Metadata (ISO15836), aunque permite otras normas (ISO 19115/ISO 19119)
- Modelo general: con definición de clases e implementación del modelo general. Define la clase CatalogService y cinco clases más asociadas, así como los interfaces que soportan la funcionalidad de búsqueda, acceso, mantenimiento y organización.
- Protocolos de conexión: Z39.50, CORBA/IOP y HTTP. Modelos que añaden guías para el diseño de la implementación del modelo general. Incluyen un mapeo entre las interfaces, operaciones y parámetros generales disponibles en el protocolo. (GEOGRÁFICO, 2012)

Las operaciones que se definen en este estándar OGC son 7, siendo 3 obligatorias y 4 opcionales. A continuación se muestran ejemplos de algunas de las operaciones utilizando el servicio de catálogo CSW del Instituto Geográfico de España:

- GetCapabilities (obligatoria): Para obtener las características del servicio CSW.
- GetRecords (obligatoria): Las consultas que se realizan al Servicio, para que te devuelvan los metadatos que coinciden con unas determinadas condiciones.
 - ✓ Consulta por Título: Metadatos que contengan la palabra 'Cuevas' en el Título.
 - ✓ Consulta por Resumen: Metadatos que contengan la palabra 'Cuevas' en el Resumen.
 - ✓ Consulta en cualquier lugar: Metadatos que contengan la palabra 'Cuevas'.
- GetRecordById (obligatoria): Para recuperar un metadato a partir de su Identificador.
- DescribeRecords (opcional): Informa acerca de la estructura de los registros de metadatos.
- GetDomain (opcional): Permite a los usuarios consultar los valores permitidos de un parámetro o propiedad determinados.
- HarvestRecord (opcional): Mediante la ubicación de un recurso de metadatos el servicio analiza el recurso y crea o modifica los registros de metadatos de su catálogo. El objetivo es registrar y modificar registros de metadatos automáticamente.
- Transaction (opcional): Permite insertar, actualizar y borrar registros del catálogo de metadatos. (GEOGRÁFICO, 2012)

1.5 Situación actual del problema.

La LPS “Aplicativos SIG” perteneciente al departamento GeoInformática del centro GEYSED, es el encargado de desarrollar las aplicaciones relacionadas con mapas en la Web, utiliza como base de apoyo la Plataforma soberana para el desarrollo de Sistemas de Información Geográfica (GeneSIG). La misma surge por la necesidad de contar con un producto que sirva como soporte de aplicaciones SIG en entornos Web con tecnologías libres.

Capítulo 1: Fundamentos teóricos de la investigación

Sus objetivos principales son permitir la representación geoespacial de la información asociada a cualquier negocio que lo requiera, proporcionar servicios de acceso a la información geográfica, para su consulta, análisis y visualización, mediante una interfaz de usuario sencilla y de fácil manejo que pueda ser utilizada por usuarios no especializados en tecnología SIG. Integrar la información socioeconómica existente (recursos humanos, activos fijos, entidades de servicios y lugares de interés.) con la información geográfica asociada. Los diferentes productos desarrollados por el equipo de trabajo de la LPS Aplicativos SIG en general realizan disímiles funciones para el trabajo con mapas, que van desde gestión de información, análisis espacial, tematizaciones, entre otras, todas con el objetivo final de satisfacer las necesidades de los clientes y lograr un producto de elevada calidad.

Actualmente se encuentran en desarrollo varios productos que responden a necesidades de diversas ramas de la sociedad, dentro de los que se encuentra SIG- MINAL, SIG-Rutas, SIG- APICOLA, SIG-CENAI, SIG- ONEI, SIG- PPD y SIG-UCI. Debido al aumento considerable de las bases de datos que estos productos gestionan y el elevado número de capas espaciales que existe, es necesaria la incorporación de metadatos geográficos para ayudar a la organización y búsqueda de información geográfica.

En el centro GEYSED se desarrolló la herramienta SyGMe, la cual es la encargada de gestionar los metadatos geográficos asociados a las bases de datos espaciales que se trabajan en dicho departamento. Es una aplicación desarrollada por los mismos desarrolladores del centro y que entre sus principales características se encuentra la gestión de metadatos haciendo uso de la norma ISO 19115. El principal problema que presenta es que la información no puede ser consultada por otros sistemas, pues para ello necesitan integrarse a través de la base de datos. Para poder obtener la información de los metadatos asociados a una base de datos, éstos deben de realizar la integración a nivel de base de datos, configurando una conexión y realizando las consultas directamente a través del propio código del SIG a desarrollar.

Para los desarrolladores de la LPS Aplicativos SIG los problemas antes mencionados se resuelven con la creación de un módulo para la plataforma GeneSIG que permita esta integración a nivel de base de datos con la herramienta SyGMe. Esto trae como problema principal que los mismos desarrolladores deban conocer la estructura en que está constituida las bases de datos de SyGMe, el formato en que se envían los datos y la manera en la que se reciben e interpretan. Las bases de datos del SyGMe no se encuentra en el mismo lugar

que las bases de datos espaciales trabajada en Aplicativos SIG, estas cuestiones afectan la seguridad de los datos al existir dentro del módulo a desarrollar la configuración de la conexión a la base de datos donde se encuentran dichos metadatos.

Aun cuando la creación de dicho módulo soluciona los problemas de integración entre la plataforma GeneSIG y los metadatos gestionados en la herramienta SyGMe, no constituye una solución a todo el problema, pues, aunque la mayoría de los SIG se realicen a partir de dicha plataforma, existen otras personalizaciones que tienen como base otras tecnologías, además se requiere una solución para el centro GEYSED y no para un producto en específico.

Para la LPS Aplicativo SIG se hace necesario contar con un sistema que elimine los inconvenientes antes mencionados, que permita a los desarrolladores de Aplicativos SIG poder hacer la búsqueda de la información de los metadatos sin tener en cuenta la estructura de las bases de datos de SyGMe, conexión y ubicación con el objetivo de poder brindar la opción en los productos desarrollados a los clientes, logrando una mayor aceptación de los mismos. Con el desarrollo del Servicio Web se pretende ofrecer la búsqueda y organización de información geográfica a cualquier SIG del centro de GEYSED, así como abogaría a la interoperabilidad con el uso de formatos estándar como es el caso de XML para la comunicación con el Servicio Web.

1.6 Análisis de soluciones existentes

Al día de hoy existen diversas aplicaciones relacionadas estrechamente al tema de intercambio de metadatos geográficos a través de Servicios Web. El objetivo de indagar y profundizar en las soluciones existentes es establecer criterios acerca de cuáles de estas pudieran favorecer y aportar en cuanto a tecnología se refiere para el desarrollo de la solución de esta investigación.

GeoNetwork, es una de las aplicaciones más utilizadas para la edición de metadatos de información geográfica. Esta aplicación de escritorio y entorno Web, basa la gestión de metadatos de acuerdo a las normas 19115, Dublin Core y FGDC y a varios perfiles de estos estándares, garantizando la visualización de los

metadatos en variados esquemas. Permite las operaciones programables de recolección (harvesting⁸) y sincronización de metadatos en catálogos distribuidos, así como el intercambio y búsqueda de metadatos a través del Servicio CSW. Para la utilización del CSW la herramienta GeoNetwork configura el puerto de conexión del servidor Web para comunicarse con el CSW y de esta forma obtiene los metadatos geográficos. Internacionalmente, figura como una de las plataformas más empleadas para el manejo y la búsqueda de metadatos geográficos. (Yurtseven, 2007)

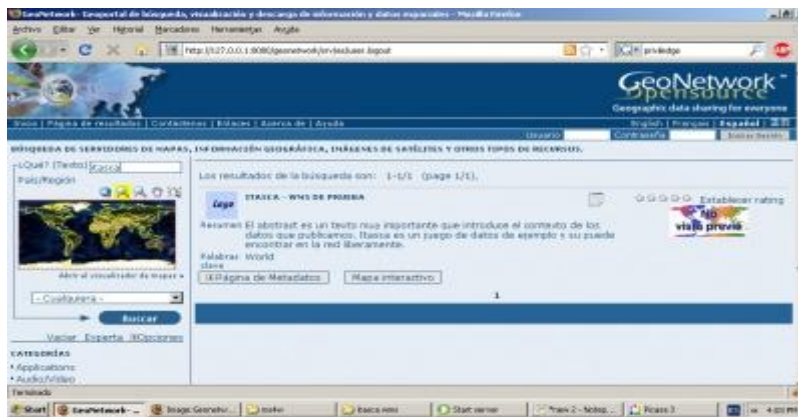


Ilustración 2: Interfaz GeoNetwork (Yurtseven, 2007)

ArcGIS for Server es un software que pone su información geográfica a disposición de otras personas de su organización y, de manera opcional, de cualquiera con una conexión a Internet. Esto se logra a través de los servicios web, que permiten que un equipo servidor potente reciba y procese las solicitudes de información enviadas por otros dispositivos. Uno de los módulos que contiene **ArcGIS** es **ArcMap** el cual tiene un cliente CSW que permite buscar servicios de catálogo OGC para la web. Una vez que los registros de metadatos se devuelven, se pueden ver o descargar. Además, si los datos de referencia de metadatos geográficos identificados como "en vivo de datos y mapas", los datos se pueden agregar al documento de ArcMap o ArcGIS Explorer Globe. En ArcMap, hay funcionalidades adicionales que permiten a los usuarios ver la extensión espacial de los datos.

⁸ Harvesting: es un proceso normal automatizado de recogida de descripciones de metadatos de distintas fuentes para crear agregaciones útiles de metadatos y servicios relacionados

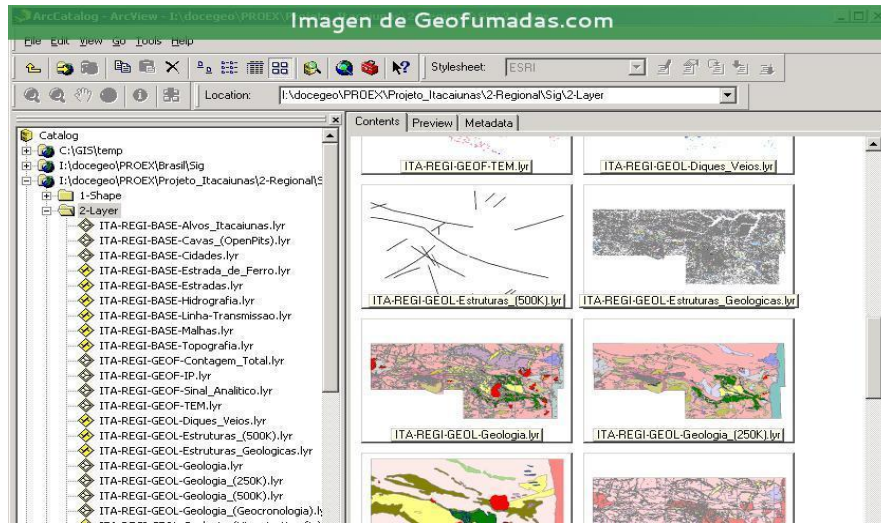


Ilustración 3: Interfaz ArcMap (ESRI, España, 2010)

El Instituto Geográfico de España es un portal Web encargado de la modelación y análisis geoespacial, triangulaciones geodésicas de diversos órdenes, nivelaciones de precisión, topografía de mapas, así como las cuestiones relativas a pesos y medidas geográficas. Opcionalmente se pueden consultar la información asociada a la capa, la leyenda si existe y los atributos disponibles en un punto. Contiene un catálogo de metadatos que proporciona información descriptiva de la información geográfica que gestiona, mediante la utilización del servicio CSW, el cual permite la búsqueda e intercambio de metadatos geográficos, servicios, aplicaciones y, si se desea, todo tipo de recursos. También visualiza la información geográfica para un área determinada a través de internet haciendo uso del servicio WMS. (Ibero, 2009).



Ilustración 4: Interfaz Portal del Instituto Geográfico de España. (Ibero, 2009)

El portal **Geosur** es una de las aplicaciones más utilizadas para la edición y búsqueda de metadatos geográficos, así como también un servicio de mapas establecido y vinculado en los metadatos. Es una aplicación Web que hace uso de estándares geoespaciales que implementan el Sistema de Gestión de Metadatos Geográficos del Distrito Capital. Dichos estándares son la especificación del OGC relacionado CSW y el esquema de implementación de la ISO 19139:2007. Posee un registro de catálogos de metadatos geográficos, el cual mediante el uso del CSW realiza búsquedas y garantiza la organización de los mismos. (Geosur, 2007)



Ilustración 5: Interfaz Geosur. (Geosur, 2007)

Después de haber realizado un análisis de las principales características de las herramientas investigadas se descartan el uso de las mismas, pues el interés no es como las soluciones gestionan los metadatos, sino la forma en que los metadatos geográficos que ellas contienen son publicados en la Web. La utilización de dichas herramientas se ve incapacitada porque las mismas no se ajustan a las condiciones de los metadatos geográficos gestionados en SyGMe, no hay forma de acceder desde las herramientas investigadas a la base de datos de SyGMe para acceder a los metadatos geográficos. Sin embargo sirve de ejemplo para la implementación del Servicio Web en cuestión, pues la implementación del mismo se basará en el servicio CSW el cual es utilizado por los sistemas investigados, ya que los mismos son uno de los más potentes que tratan el tema de metadatos geográficos.

1.7 Conclusiones parciales

Desarrollar un Servicio Web es la solución que mejor se adapta a las necesidades de los desarrolladores del proyecto Aplicativos SIG a la hora de obtener los metadatos de las BD del SyGMe. El desarrollo de la herramienta propuesta enriquecerá los conocimientos existentes en la Universidad respecto a este tema, lo que influirá positivamente en el desarrollo hacia la soberanía tecnológica del País. El análisis del estado del arte realizado en el presente capítulo permitió obtener un conocimiento profundo acerca de la temática abordada estableciendo las bases para una mejor comprensión acerca del negocio a desarrollar. El estudio de las diferentes aplicaciones existentes y el análisis de sus características aportan los elementos suficientes para justificar el desarrollo de la propuesta de la investigación, ya que ninguna de las herramientas estudiadas da solución al problema planteado al inicio de la investigación. Los aspectos abordados proveen la base de conocimientos necesaria para llevar a cabo la implementación del Servicio Web. Teniendo en cuenta el conocimiento adquirido hasta este punto se puede proceder a la presentación de la solución propuesta, la especificación de los requisitos deseados por el cliente y la selección de las herramientas y tecnologías a emplear.

CAPÍTULO 2: Tendencias y tecnologías actuales a desarrollar

2.1 Introducción

En el desarrollo del presente capítulo se caracterizará la metodología a utilizar, tecnologías, lenguajes de programación y herramientas para el desarrollo del software. Se realizará una selección de las mismas para la construcción de la solución propuesta. Para ello se tuvo en cuenta como principal elemento a considerar que fueran libres o de código abierto, pues en Cuba actualmente se aboga por la soberanía e independencia tecnológica y la UCI no se encuentra exenta de esto. Por lo tanto en la decisión a tomar se debe ser consecuente con la programación de Servicios Web, metodología, entre otras tecnologías.

2.2 Propuesta de herramientas

Con la siguiente propuesta de herramientas se pretende alcanzar un Servicio Web de alta calidad, libre, escalable, adaptable, que cumpla con las expectativas planteadas y que sea de gran utilidad para la universidad en especial para el proyecto Aplicativos SIG.

2.2.1 XML

Es un lenguaje desarrollado por el W3 Consortium para permitir la descripción de información contenida en la red de redes a través de estándares y formatos comunes, de manera que tanto los usuarios de Internet como programas específicos (agentes) pueden buscar, comparar y compartir información en la red. El formato de XML es muy parecido al de HTML aunque no es una extensión ni un componente de éste (W3C, 2006). Sirve para que muchos programas interpreten bien cualquier tipo de dato y se comuniquen entre ellos sin intervención humana. Constituye el lenguaje base de los Servicios Web. Este es un lenguaje extensible de etiquetas estandarizado para describir datos.

Inicialmente la interoperabilidad se logró con el uso de formatos de datos propietarios, por lo tanto, el desarrollo estaba sujeto al modo de trabajar del fabricante. Por esta razón, se buscó avanzar en el desarrollo de software, convirtiendo a los Servicios Web en estándares a partir de la portabilidad de datos que permite XML y la portabilidad de código que permite Java. XML, el lenguaje de marca extensible ha permitido la evolución de otras tecnologías. Las herramientas que brinda XML no son muy útiles sin lenguajes de

programación como Java, que fijan el estándar de cómo utilizar XML en las aplicaciones. “Primero se analizan los datos y después se manipulan”. Una vez el documento XML es analizado, la información está disponible para ser utilizada por un cliente. Dado que XML es usado para la presentación, la comunicación y la configuración, un cliente de un Servicio Web puede ser una aplicación, otro Servicio Web o un humano.

2.2.2 Simple Object Access Protocol (SOAP)

Es un protocolo de la capa de aplicación para el intercambio de mensajes basados en XML sobre redes de computadores. Básicamente es una vía de transmisión entre un SOAP Sender y un SOAP Receiver, pero los mensajes SOAP deben interactuar con un conjunto de aplicaciones para que se pueda generar un “diálogo” a través de mensajes SOAP. Un mensaje SOAP es la unidad fundamental de una comunicación entre nodos SOAP. SOAP es básicamente un paradigma de una sola vía pero con la ayuda de las aplicaciones se puede llegar a crear patrones más complejos. SOAP básicamente está constituido por:

- Un marco que describe el contenido del mensaje e instrucciones de proceso.
- Un conjunto de reglas para representar los tipos de datos definidos.
- Convenciones para representar llamadas a procedimientos remotos y respuestas. (W3C, 2006)

2.2.3 Web Services Description Language (WSDL)

WSDL es un tipo de documento XML que describe lo que hace un Servicio Web, donde se encuentra y la forma de ser invocado. Al describir un Servicio Web, el cliente sabe lo que el servicio hace, define que métodos están disponibles, los tipos de parámetros y que estos estén. Un programador debe saber cómo está constituido un Servicio Web para poder codificar un cliente capaz de invocarlo, debe conocer su API⁹, que Servicios ofrece y qué tipo de protocolo de codificación se usará. Si un Servicio Web se encuentra disponible al público, su WSDL también estará a la vista de cualquiera. Estos documentos XML pueden ser encontrados en

⁹API: Interfaz de programación de aplicaciones (del inglés Application Programming Interface) es el conjunto de funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

registros UDDI o en Web especializadas que funcionan como un buscador normal de Internet. Siempre un documento XML WSDL presenta los siguientes elementos:

- Tipos: Tipos de datos usados por los mensajes.
- Mensaje: Que datos son enviados desde un nodo a otro.
- Tipo de puerto: Define las operaciones que pueden ser llamadas.
 - Operación: Define la configuración de mensajes de entrada, salida y falta.
 - Entrada: Mensaje que es enviado hacia el servidor.
 - Salida: Mensaje enviado hacia el cliente.
 - Falta: Error en el envío de un mensaje.
- Límite: Es la descripción del protocolo que se está utilizando para transportar el mensaje que puede ser HTTP POST, HTTP GET, SOAP y MIME.
- Servicio: Define una colección de puertos (nodos); el puerto especifica una dirección para el límite definiendo así la comunicación para un nodo específico. (Tidwell, 2002)

EL WSDL no es obligatorio, sin embargo su uso permite ventajas como:

- Reducir el costo de mantenimiento en clientes.
- Facilitar la estructuración y construcción de objetos.
- Automatizar la construcción de clientes volviendo atractivo el uso de Servicios Web.
- La generación del WSDL también puede automatizarse, con herramientas tales como: NuSOAP y Zend Framework.

2.3 Metodología de Desarrollo

“Las metodologías y estándares utilizados en el desarrollo de software proporcionan las guías para poder conocer todo el camino a recorrer desde antes de empezar la implementación, con lo cual se asegura la calidad del producto final, así como también el cumplimiento en la entrega del mismo en un tiempo estipulado.”
(CHACÓN, 2006)

“Se dice que una metodología de desarrollo de software la constituye un conjunto de procedimientos, herramientas, tecnología y un soporte teórico-documental para el desarrollo guiado, estructurado y organizado del software.” (Jacobson, 2000)

Se puede llegar a la conclusión de que una metodología de software es la que posibilita paso a paso desarrollar todos los procesos envueltos en la elaboración del producto, convirtiéndose en la guía para obtener un resultado de gran calidad, además de procurar cuales son los actores que deben participar en la construcción del producto así como el papel que deben tener en el mismo, con esto se podría garantizar la calidad del software y un mayor control sobre el proceso.

2.3.1 Programación extrema (XP)

Es una metodología ágil basada en una serie de valores y de prácticas de buenas maneras que persigue el objetivo de aumentar la productividad a la hora de desarrollar programas, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores y propiciando un buen clima de trabajo. XP se basa en la realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y facilidades para enfrentar los cambios. Además se define especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes y donde existe un alto riesgo técnico. Los principios y prácticas son de sentido común pero llevadas al extremo, de ahí proviene su nombre. (Penadés, 2013)

Una de las principales ventajas que ofrece la programación extrema es que se adapta al desarrollo de sistemas pequeños y grandes; optimiza el tiempo de desarrollo; el código es sencillo y entendible, además de la poca documentación a elaborar para el desarrollo del sistema. Por otra parte no se tiene la definición del costo y el tiempo de desarrollo; el sistema va creciendo después de cada entrega al cliente y nadie puede decir que el cliente no querrá una función más; se necesita de la presencia constante del cliente, lo cual en la realidad es muy difícil de lograr.

Comunicación

Es muy importante que haya una comunicación constante con el cliente y dentro de todo el equipo de trabajo,

de esto dependerá que el desarrollo se lleve a cabo de una manera sencilla, entendible y que se entregue al cliente lo que necesita.

Simplicidad

Se basa en la filosofía de que el mayor valor de negocio es entregado por el programa más sencillo que cumpla los requerimientos. Se enfoca en proporcionar un sistema que cubra las necesidades inmediatas del cliente, ni más ni menos. Este proceso permite eliminar redundancias y rejuvenecer los diseños obsoletos de forma tal que el diseño sea sencillo y amigable al usuario, el código debe ser simple y entendible.

Retroalimentación

Es la comunicación constante entre el desarrollador y el usuario. El usuario o cliente pasa a ser parte implicada en el equipo de desarrollo, tiene un papel importante de interacción con el equipo de programadores, sobre todo después de cada cambio y de cada posible problema localizado, mostrando las prioridades y expresando sus sensaciones.

Valentía

Se refiere al coraje que se debe tener al modificar o eliminar el código que se realizó con tanto esfuerzo; el desarrollador debe saber cuándo el código que desarrolló no es útil en el sistema y, por lo mismo, debe ser eliminado. También se refiere a tener la persistencia para resolver los errores en la programación.

Herramientas de XP:

Historias de usuarios: Son tarjetas físicas en las cuales se anota una descripción de una funcionalidad del sistema, en una oración, se le da un número y un título para ser identificada.

Casos de prueba de aceptación: Son tarjetas que se elaboran para realizar las pruebas de cada historia de usuario.

Tarea de ingeniería: Son tarjetas que se elaboran para ayudar y simplificar la programación de una historia de usuario.

Tarjetas (Clase, Responsabilidad y Colaboración) CRC: Describen las clases utilizadas en la programación de una historia.

El ciclo de vida ideal consta de 4 fases: (J.J. Gutierrez, 2012)

Exploración: En esta fase, los clientes plantean a grandes rasgos las historias de usuario que son de interés para la primera entrega del producto. Al mismo tiempo el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto. Se prueba la tecnología y se exploran las posibilidades de la arquitectura del sistema construyendo un prototipo.

Planificación: En esta fase el cliente establece la prioridad de cada historia de usuario y correspondientemente, los programadores realizan una estimación del esfuerzo necesario de cada una de ellas. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente. Esta fase incluye varias iteraciones sobre el sistema antes de ser entregado. El plan de entrega está compuesto por iteraciones de no más de tres semanas.

Construcción: La fase de construcción requiere de las tarjetas CRC, estas permiten desprenderse del método de trabajo basado en procedimientos y trabajar con una metodología basada en objetos. En esta fase se realiza el desarrollo de las iteraciones, donde las historias de usuario se descomponen en tareas de programación o en tareas de ingeniería.

Prueba: Esta fase garantiza el correcto funcionamiento de las funcionalidades que el sistema posee. Las pruebas se dividen en dos: en pruebas unitarias y pruebas de aceptación. Las de aceptación verifican que todas las funcionalidades devuelvan el valor correspondiente a los requisitos funcionales del sistema a implementar.

Se selecciona XP para desarrollar el Servicio Web previsto porque se adapta y ofrece características que se acoplan al proyecto en función. Se adapta a los cambios necesarios y continuos que puedan surgir a lo largo de la implementación debido a la fluida comunicación entre el cliente y el equipo de desarrollo facilitando el intercambio de ideas, además que se basa en entregar al cliente lo que este necesite y de forma sencilla, debido a que se centra solo en la implementación. Por ello resulta ideal para proyectos pequeños, porque genera muy poca documentación.

2.4 Arquitectura de software

“La Arquitectura de Software es, a grandes rasgos, una vista del sistema que incluye los componentes principales del mismo, la conducta de esos componentes según se la percibe desde el resto del sistema y la forma en la que los componentes interactúan y se coordinan para alcanzar la misión del sistema”. (Clements, 1996). La definición oficial que brinda el documento del Instituto de Ingenieros Eléctricos y Electrónicos (IEEE Std 1471-2000) plantea que: *“La Arquitectura de Software es la organización fundamental de un sistema encarnada en sus componentes, las relaciones entre ellos y el ambiente y los principios que orientan su diseño y evolución”.*

A partir de las anteriores definiciones se puede decir que la arquitectura de software es aquella rama de la ingeniería de software dedicada a organizar, relacionar y estructurar los componentes del sistema con el que se interactúa durante el desarrollo.

2.4.1 Arquitectura cliente-servidor

En el Servicio Web a implementar se utilizará el tipo o modelo de arquitectura de entorno Web: Cliente-Servidor. Este modelo arquitectónico se caracteriza por el envío de mensajes a través del cliente para realizar una determinada petición al servidor, a partir de lo cual, este último envía uno o varios mensajes con la respuesta (servicio). Se basa en la existencia de dos tipos de aplicaciones que se ejecutan de manera independiente. (Cesar de la Torre Llorente, 2010)

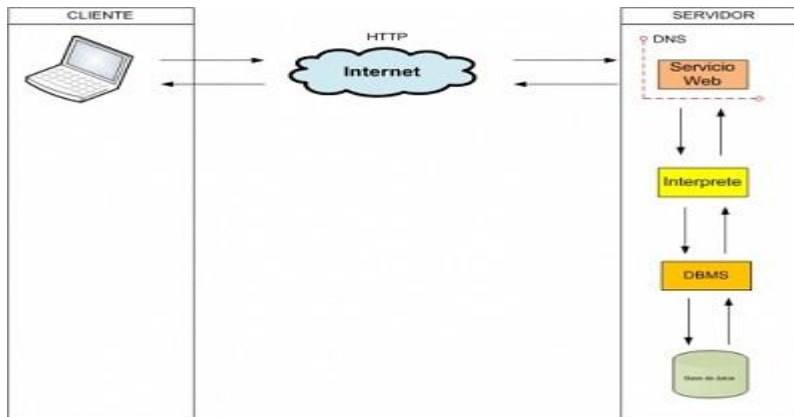


Ilustración 6: Arquitectura Cliente-Servidor

Es una arquitectura que se emplea para sistemas distribuidos pues de esta forma el Servicio Web es publicada por el servidor y consumida por distintas aplicaciones clientes mediante protocolos de conexión de red, los clientes pueden hacer una o más peticiones y a su vez el servidor le responde con el resultado de la misma. Con esto se garantiza el acceso centralizado a los datos que éste gestiona y que son almacenados en la base de datos.

EL Servicio Web fue desarrollado bajo la arquitectura Cliente-Servidor definida por el envío de mensajes desde el cliente al servidor de Servicios Web y las respuestas que este le da a las solicitudes. El cliente que desee consumir el servicio primeramente debe conocer la dirección URL donde se encuentra alojado el servidor para así de esta forma acceder al documento WSDL que es descrito y presentado en el servidor. El cliente al acceder al servicio realiza una o varias peticiones y el servidor realiza los procedimientos para darle respuesta.

2.5 Servidores Web

Un servidor Web es un programa que sirve datos en forma de páginas Web, hipertextos o páginas HTML (Hyper Text Markup Language). La comunicación de estos datos entre cliente y el servidor se hace por medio de un protocolo, concretamente el protocolo HTTP. El servidor es el responsable de proporcionar el acceso a los recursos solicitados que están bajo el control del sistema operativo. Estos recursos pueden ser estáticos, como páginas HTML o textos dinámicos, como por ejemplo CGI's. (Common Gateway Interface o Pasarela de Interface Común). (Fernández, 2008)

2.5.1 Servidor Apache 2.2

Apache es un servidor Web flexible, rápido y eficiente, consecutivamente actualizado y adaptado a los nuevos protocolos (HTTP). Apache HTTP es de tecnología Open Source sólido y para uso comercial creado por la Apache Software Foundation. Se propone el uso de este servidor Web porque es personalizable. (Aries, 2012)

Los archivos en cuanto a la administración de configuración de Apache están en ASCII¹⁰, por lo que posee un formato simple y logran ser editados tan solo con un editor de texto. Estos son transferibles, lo que admite la clonación positiva de un servidor. Puede ser el servidor administrado vía línea de instrucciones, lo que hace la administración remota muy útil. (Aries, 2012)

Apache Web Server tiene un gran conjunto de funcionalidades de gran alcance. Estas características principales, junto con las extensiones creadas por programadores de todo el mundo, ayudan a que la plataforma Apache sea competitiva incluso frente a rivales de alto precio. Apache ha incorporado en su soporte a una amplia gama de lenguajes de programación Web, como Perl, PHP y Python. Estos lenguajes son fáciles de aprender y se pueden utilizar para crear potentes aplicaciones en línea. (Aries, 2012)

El servidor Web elegido es Apache, pues brinda varias ventajas dentro de las que se encuentra fundamentalmente que es de código abierto, otros aspectos muy importantes que lo caracterizan son su estructura modular, además de ser multiplataforma convirtiéndolo en el servidor más utilizado en la actualidad. Apache es una tecnología gratuita, presenta una cómoda interacción con PostgreSQL y es el encargado de gestionar la totalidad de las funciones de lógica de negocio y acceso a datos de la aplicación con la que esté trabajando, es fácil de configurar y de adaptar a las necesidades del proyecto.

¹⁰ ASCII Sistema de codificación de siete bits que asigna un número del 0 al 127 a cada letra, número, carácter especial y carácter de control que recoge: los ficheros de texto contienen información escrita en código ASCII, lo que les permite ser entendidos por la mayoría de los programas del mercado.

2.6 Lenguaje de programación Web

Un lenguaje de programación es una construcción mental del ser humano para expresar programas. Está constituido por un grupo de reglas gramaticales, de símbolos utilizables, de términos mono sémicos (es decir, con sentido único) y una regla principal que resume las demás, todas estas son utilizadas por el programador a través de las cuales crea un programa o subprograma. (Español.)

En este epígrafe se abordan algunas características de lenguajes de programación del lado del servidor, para poder argumentar con estos elementos los lenguajes seleccionados.

PHP 5.2.12

PHP es un lenguaje interpretado de propósito general ampliamente usado, diseñado especialmente para desarrollo Web. Generalmente se ejecuta en un servidor Web, tomando el código en PHP como su entrada y creando páginas Web como salida. Puede ser desplegado en la mayoría de los servidores Web y en casi todos los sistemas operativos y plataformas sin costo alguno.

PHP es un lenguaje script procesado en el lado del servidor. El funcionamiento de PHP se evidencia cuando el navegador realiza una petición al servidor (se escribe la URL), después el servidor ejecuta el código PHP solicitado y retorna el código HTML generado al navegador y el mismo muestra la respuesta del servidor. En caso de los Servicios Web PHP, cuando el cliente hace una petición al servidor para que le envíe la respuesta de un servicio, el servidor ejecuta el intérprete de PHP. Éste procesa el Script solicitado que generará el contenido de manera dinámica (por ejemplo obteniendo información de una base de datos). El resultado es enviado por el intérprete al servidor, quien a su vez se lo envía al cliente. Puede ser desplegado en la mayoría de los servidores Web y en casi todos los sistemas operativos y plataformas sin costo alguno lo cual lo convierte en la tecnología ideal para la creación de Servicios Web. (Rodas, 2007)

A partir de lo planteado se seleccionó como lenguaje de programación: PHP 5.2.12, por su rapidez, su facilidad de aprendizaje, su soporte multiplataforma, así como servidores HTTP y de bases de datos. Brinda soporte para una gran cantidad de bases de datos como: MySQL, PostgreSQL, Oracle, MS SQL Server, Sybase mSQL, Informix, entre otras. Aporta funcionalidades de soporte integrado para SOAP y XML-RPC en la creación de Servicios Web, lo cual es de vital importancia porque no se necesita importar ninguna librería para

la creación del Servicio Web, además se distribuye de forma gratuita bajo una licencia de código abierto, principal propósito de selección de herramientas y tecnologías a utilizar.

2.7 Entorno Integrado de Desarrollo (IDE)

Un entorno de desarrollo integrado o en inglés Integrated Development Environment (IDE) es un programa compuesto por un conjunto de herramientas para un programador. Puede dedicarse a un sólo lenguaje de programación o a varios. Los IDEs proveen un marco de trabajo amigable para la mayoría de los lenguajes, incluso en algunos de ellos pueden funcionar como un sistema en tiempo de ejecución donde se puede emplear el lenguaje de programación en forma interactiva. A continuación se exponen las características fundamentales del IDE NetBeans, el cual se emplea en la construcción del sistema propuesto.

2.7.1 NetBeans 7.3

El IDE NetBeans es una herramienta para programadores pensada para escribir, compilar, depurar y ejecutar programas. Es un IDE de código abierto escrito en Java pero puede servir para cualquier otro lenguaje de programación. Existe además un número importante de módulos para extender el IDE NetBeans. Es un producto libre y gratuito sin restricciones de uso. Entre sus características se encuentra un sistema de proyectos basado en control de versiones y refactorización. Adicionalmente, el NetBeans Enterprise Pack soporta el desarrollo de Aplicaciones empresariales con Java EE 5, incluyendo herramientas de desarrollo visuales de SOA, herramientas de esquemas XML, orientación a Servicios Web y modelado UML. (Wheeler, 2008)

Se ha seleccionado además la herramienta NetBeans como editor de PHP, pues ofrece un eficiente auto-completado de código, marcado de error de PHP, plantillas, macros, control de versiones, también es muy importante decir que es multiplataforma y de código abierto lo que se corresponde con el objetivo principal de este proyecto.

2.9 SOA Client

Numerosos servicios en la Internet, así como dentro de las empresas implementan Servicios Web. Este cliente se ha utilizado durante el estudio para mostrar el correcto funcionamiento del Servicio Web desarrollado,

porque invoca fácilmente Servicios Web basados en objetos SOAP y presenta la capacidad para actualizar dinámicamente las variables en tiempo de ejecución.

2.10 Conclusiones parciales

La selección de las herramientas y tecnologías libres y de un elevado nivel científico-técnico permitió que el Servicio Web desarrollado fuese una solución igualmente libre, y de elevada calidad y eficiencia, libre del pago a entidades por concepto de licencias. El desarrollo de la investigación se apoya en las políticas nacionales de uso de herramientas libres logrando así incrementar el desarrollo tecnológico del País a favor de la soberanía tecnológica. La documentación y desarrollo del ciclo de trabajo se realizó de manera organizada siguiendo los estándares de la metodología escogida permitiendo la obtención de un sistema de elevada calidad y eficiencia. La arquitectura es un uno de los elementos claves en el desarrollo de una solución informática y mediante esta se pudo obtener una visión inicial del producto final. El uso de las tecnologías de la Web, y los protocolos de comunicación de esta, aportan importantes elementos a la seguridad, calidad y fiabilidad de los datos intercambiados entre dos o más sistemas. Al hacer uso del lenguaje PHP y una arquitectura dividida en capas el sistema podrá ser fácilmente adaptable a las condiciones de los diferentes proyectos, escalable y fácil de actualizar por futuros desarrolladores. A partir del conocimiento adquirido hasta este momento se puede acceder a la fase de implementación luego que sean definidas las funcionalidades del sistema.

CAPÍTULO 3: Presentación de la solución propuesta

3.1 Introducción

El objetivo que se persigue en el presente capítulo es, abordar temáticas referentes al análisis y al diseño, así como la descripción textual de cada uno de estos conceptos para arribar a conclusiones que fundamenten y sean el punto de partida para la implementación. También se especifican los requisitos funcionales (RF) y no funcionales que el sistema debe cumplir y se representa la solución propuesta mediante el diagrama de casos de uso del sistema, la descripción de los actores y los casos de uso detallados.

3.2 Fase de Exploración

En esta fase, los clientes plantean a grandes rasgos las historias de usuario que son de interés para la primera entrega del producto. Al mismo tiempo el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto. Se prueba la tecnología y se exploran las posibilidades de la arquitectura del sistema construyendo un prototipo. La fase de exploración toma de pocas semanas a pocos meses, dependiendo del tamaño y familiaridad que tengan los programadores con la tecnología. (Penadés, 2013)

3.2.1 Historias de Usuarios

Las historias de usuario (HU) son la técnica utilizada en XP para especificar los requisitos del software. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de las historias de usuario es muy dinámico y flexible, en cualquier momento las historias de usuario pueden romperse, reemplazarse por otras más específicas o generales, añadirse nuevas o ser modificadas. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas.

Respecto a la información contenida en la historia de usuario, existen varias plantillas sugeridas pero no existe un consenso al respecto. En muchos casos sólo se propone utilizar un nombre y una descripción o sólo una descripción, o quizás una estimación de esfuerzo en días. Beck en su libro presenta un ejemplo de ficha

(*customer story and task card*) en la cual pueden reconocerse los siguientes contenidos: fecha, tipo de actividad (nueva, corrección, mejora), prueba funcional, número de historia, prioridad técnica y del cliente, referencia a otra historia previa, riesgo, estimación técnica, descripción, notas y una lista de seguimiento con la fecha, estado, elementos por terminar y comentarios. (Penadés, 2013)

3.2.2 Relación de las Historias de Usuarios

Como parte del proceso de trabajo dentro de la fase de exploración se identificaron un total de 6 Historias de Usuarios de las cuales se muestra una, las demás se encuentran en los Anexos.

Historia de Usuario	
Número: 1.	Nombre de historia: Recuperar metadatos asociados a una capa
Usuario: Cliente.	
Prioridad en el negocio: Alta.	Riesgo en Desarrollo: Alto.
Puntos estimados: 1.	Iteración asignada: 1.
Programador responsable: Bárbaro Orestes Ramos Díaz	
Descripción: Permite obtener los metadatos geográficos asociados a una capa geoespacial almacenados en la base de datos.	
Observaciones: A este requisito se le pasa el nombre de una capa geoespacial y devuelve los metadatos asociados a la capa.	

Tabla 1: HU Recuperar metadatos asociados a una capa.

3.3 Fase de Planificación

En esta fase el cliente establece la prioridad de cada historia de usuario y los programadores realizan una estimación del esfuerzo necesario de cada una de ellas. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente. Una entrega debería obtenerse en no más de tres meses. Esta fase dura unos pocos días.

Las estimaciones de esfuerzo asociado a la implementación de las historias la establecen los programadores utilizando como medida el punto. Un punto, equivale a una semana ideal de programación. Las historias

generalmente valen de 1 a 3 puntos. Por otra parte, el equipo de desarrollo mantiene un registro de la “velocidad” de desarrollo establecida en puntos por iteración, basándose principalmente en la suma de puntos correspondientes a las historias de usuario que fueron terminadas en la última iteración.

La planificación se puede realizar basándose en el tiempo o el alcance. La velocidad del proyecto es utilizada para establecer cuántas historias se pueden implementar antes de una fecha determinada o cuánto tiempo tomará implementar un conjunto de historias. Al planificar por tiempo, se multiplica el número de iteraciones por la velocidad del proyecto, determinándose cuántos puntos se pueden completar. Al planificar según el alcance del sistema, se divide la suma de puntos de las historias de usuario seleccionadas entre la velocidad del proyecto, obteniendo el número de iteraciones necesarias para su implementación. (Penadés, 2013)

3.3.1 Plan de Iteraciones

Esta fase incluye varias iteraciones sobre el sistema antes de ser entregado. El plan de entrega está compuesto por iteraciones de no más de tres semanas. En la primera iteración se puede intentar establecer una arquitectura del sistema que pueda ser utilizada durante el resto del proyecto. Esto se logra escogiendo las historias que fueren la creación de esta arquitectura, sin embargo, esto no siempre es posible pues es el cliente quien decide qué historias se implementarán en cada iteración (para maximizar el valor de negocio). Al final de la última iteración el sistema estará listo para entrar en producción.

Los elementos que deben tomarse en cuenta durante la elaboración del plan de iteración son: historias de usuario no abordadas, velocidad del proyecto, pruebas de aceptación no superadas en la iteración anterior y tareas no terminadas en la iteración anterior. Todo el trabajo de la iteración es expresado en tareas de programación, cada una de ellas es asignada a un programador como responsable, pero llevadas a cabo por parejas de programadores. (Penadés, 2013)

Después de haber definido las HU y estimado el esfuerzo propuesto para la realización de cada una de ellas, se tomó la decisión de realizar el sistema en iteraciones, las cuales se detallan a continuación:

Iteración # 1: La primera iteración tiene como objetivo la implementación de las HU #1, #2 y #3 con prioridad de altas para el cliente. En esta fase la HU Seleccionar Proyecto es la base para implementar las demás iteraciones. Se dispone de 3 semanas para implementar todas las tareas. Se obtiene como resultado una

primera versión del sistema propuesto la cual será mostrada al cliente para desarrollar la retroalimentación con el equipo de desarrollo y luego pasar a la siguiente iteración.

Iteración # 2: En esta iteración se darán cumplimiento a las HU #4, #5 y #6 las cuales hacen alusión a la configuración y cambio de los componentes. Se cuenta con 3 semanas para llevar a cabo la implementación de esta iteración. Al finalizar se obtendrá una segunda versión del sistema propuesto y se le hará llegar al cliente la iteración anterior junto con la presente para la aprobación o cambios pertinentes con el cliente.

3.3.2 Estimación de esfuerzos.

Para lograr un desarrollo eficiente y satisfactorio, se realizó una estimación de esfuerzos para cada una de las Historias de Usuarios identificadas en el proceso de planificación, llegando a los resultados que se muestran a continuación.

Historias de Usuarios	Puntos de estimación
Recuperar metadatos asociados a una capa	1 semana
Recuperar capas asociadas a un metadato	1 semana
Recuperar metadatos dado un atributo identificador	1 semana
Asociar capas geoespaciales a metadatos	1 semana
Recuperar metadatos existentes	1 semana
Recuperar capas existentes	1 semana

Tabla 2: Estimación de Esfuerzos

3.3.3 Plan de duración de las iteraciones.

Para lograr una mayor organización del trabajo se crea un plan de duración de las iteraciones; el mismo tiene como objetivo mostrar la duración de cada iteración así como el orden en que serán implementadas las historias de usuarios en cada una de ellas. Este plan tiene como finalidad mostrar la duración de cada iteración, así como el orden en que serán implementadas las HU en cada una de ellas.

No Iteración	Historias de Usuario	Duración total de iteraciones
Iteración # 1	Recuperar metadatos asociados a una capa	3 semanas.
	Recuperar capas asociadas a un metadato	
	Recuperar metadatos dado un atributo identificador de metadato	
Iteración # 2	Asociar capas geoespaciales a metadatos	3 semanas.
	Recuperar metadatos existentes	
	Recuperar capas existentes	

Tabla 3: Plan de duración de Iteraciones

3.3.4 Plan de entrega

A continuación se muestra el plan de entrega desarrollado para dar solución al problema planteado. Para desarrollar el mismo se tuvo en cuenta los puntos de estimación para obtener un resultado final.

Nro. Iteración	Duración (Iter.)	Fecha inicio.	Fecha final.
Iteración # 1	3 semanas.	1-5-2013.	22-5-2013.
Iteración # 2	3 semanas.	22-5-2013.	11-6-2013.

Tabla 4: Plan de entrega

3.4 Conclusiones parciales

El Servicio Web a desarrollar es una importante herramienta cuyas funcionalidades van a dar respuesta a una necesidad existente actualmente en el Centro y que fueron recogidas en las correspondientes HU, con lo que quedaron documentados los requisitos del sistema así como sus funcionalidades. Esta documentación permite que la solución pueda ser entendida por los desarrolladores con vistas a futuras mejoras y adaptaciones a diversas condiciones de diversos proyectos. El entendimiento del negocio de la solución y especificación de los requisitos, vistos en la fase de exploración, permiten definir al Servicio Web como una solución de nivel medio y de una elevada importancia. Una correcta y fluida comunicación entre el cliente y el equipo de desarrollo es

un factor importante en la planificación del Servicio Web, puesto que permite adaptar el sistema respecto a posibles cambios, modificar las funcionalidades teniendo en cuentas las necesidades del cliente logrando así un entorno de trabajo agradable y eficiente. La planificación del trabajo y el desarrollo por iteraciones permiten que la implementación del sistema se haga de manera ordenada y limpia, ahorrando en recursos y tiempo, obteniéndose al final de la investigación un software completamente funcional que responda a las necesidades del cliente planteadas al inicio de la investigación.

CAPÍTULO 4: Construcción y Validación de la solución propuesta

4.1 Introducción

En el presente capítulo se realiza una valoración de las fases de construcción y prueba de la metodología XP abordadas anteriormente, incidiendo fundamentalmente en las restantes fases de la metodología, se describen las tarjetas CRC (Contenido, Responsabilidad y Colaboración) para un mejor entendimiento del Servicio Web. Además se exponen las tareas de programación o ingeniería generadas por cada historia de usuario y las pruebas de aceptación efectuadas sobre el Servicio Web.

4.1 Diseño de la solución propuesta.

La metodología XP plantea que el desarrollo de una aplicación o software debe realizarse de forma iterativa y obtener al término de cada iteración un producto funcional que debe ser probado; además debe ser mostrado al cliente y al emitir una opinión se pueda obtener una mejor visión de lo que desea. Esta metodología está dirigida a potenciar las relaciones interpersonales como premisa para lograr el mejor resultado posible en el desarrollo del software, proponiendo un trabajo en equipo y buscando la preparación de los desarrolladores para lograr un buen ambiente de trabajo.

En las aplicaciones desarrolladas bajo las pautas de XP no se requiere la representación del sistema mediante diagramas de clases utilizando notación UML, al utilizarse otras técnicas como las tarjetas CRC. Sin embargo se puede hacer uso de los diagramas cuando estos aporten mejoras a la hora de la comunicación entre los desarrolladores y su enfoque en la información importante.

4.2.1 Tarjetas CRC.

XP estimula el uso de tarjetas CRC como un mecanismo eficaz para pensar en el software en un contexto orientado a objetos. Las tarjetas CRC identifican y organizan las clases orientadas a objetos que son relevantes para el incremento actual de software. Las tarjetas CRC son el único producto de trabajo de diseño que se generan como parte del este proceso. (Prezi, 2013)

Capítulo 4: Construcción y validación de la solución propuesta

Cada tarjeta contiene el nombre de la clase que representa, además existe una descripción de las responsabilidades (métodos) asociados con la clase, así como una lista de otras clases relacionados mediante el envío de mensajes. El sistema CRC se usa principalmente como herramienta didáctica y como metodología para estudiar la conducta de los diseñadores orientados a objetos. Las tarjetas CRC son también un recordatorio y ayudan a los programadores experimentados y principiantes a comunicarse entre sí acerca de la modelación del entorno con objetos. (Prezi, 2013)

El factor decisivo para utilizar esta técnica para diseñar la aplicación que se desea desarrollar fueron las características que esta brinda en cuanto a facilidad de su uso y entendimiento.

Tarjeta CRC	
Clase: Nombre de la clase que se está modelando	
Responsabilidades: Es una descripción de alto nivel del propósito de la clase.	Colaboraciones: Indica con cuales otras clases se requiere relación para cumplir la responsabilidad

Tabla 5: Plantilla para Tarjetas CRC

Tarjeta CRC	
Clase: Servicio	
Responsabilidades:	Colaboraciones:
listarCapas	GestionCapa
listarMetadatos	GestionMetadato
listarCapasMetadato	GestionDatos
listarMetadatosCapa	Capa
obtenerMetadatosIndicador	Metadato

asociarCapaMetadato	SoporteWS
limpiarEntrada	

Tabla 6: CRC_ Servicio

4.3 Patrones Arquitectónicos

Los patrones arquitectónicos o patrones de arquitectura, son patrones de diseño de software que ofrecen soluciones a problemas de arquitectura de software. Dan una descripción de los elementos y el tipo de relación que tienen junto con un conjunto de restricciones sobre cómo pueden ser usados. Un patrón arquitectónico expresa un esquema de organización estructural esencial para un sistema de software, que consta de subsistemas, sus responsabilidades e interrelaciones. En comparación con los patrones de diseño, los patrones arquitectónicos tienen un nivel de abstracción mayor. (Pressman, 2011)

Por lo que se puede concluir que los patrones arquitectónicos no son más que patrones de diseño de software implementados por los desarrolladores con el objetivo de describir soluciones más factibles a problemas comunes. Describen los elementos más importantes y el tipo de relación que tienen junto con un conjunto de restricciones sobre cómo pueden ser usados.

4.4.1 Patrón arquitectónico N-Capas

“Se basa en una distribución jerárquica de los roles y las responsabilidades para proporcionar una división efectiva de los problemas a resolver. Los roles indican el tipo y la forma de la interacción con otras capas y las responsabilidades la funcionalidad que implementan.” (Cesar de la Torre Llorente, 2010)

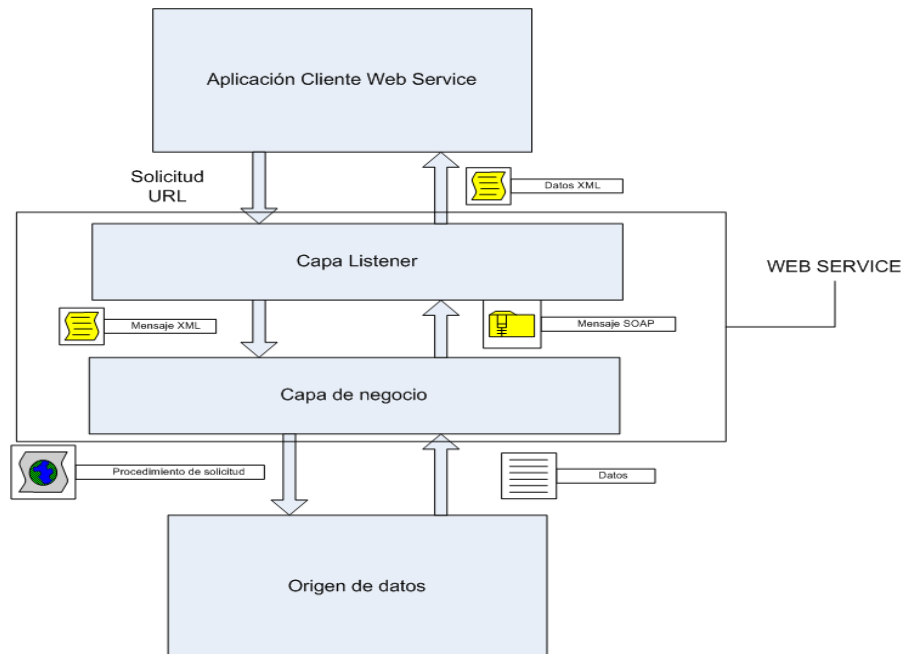


Ilustración 7: Funcionamiento de un Servicio Web (Machuca, 2012)

Beneficios

- Abstracción en los cambios que se realizan a alto nivel y se puede incrementar o reducir el nivel de abstracción que se usa en cada capa del modelo.
- Aislamiento pues se pueden realizar actualizaciones en el interior de las capas sin que esto afecte al resto del sistema.
- Rendimiento porque distribuyendo las capas en distintos niveles físicos se puede mejorar la escalabilidad, la tolerancia a fallos y el rendimiento.
- Testeabilidad porque cada capa tiene una interfaz bien definida sobre la que realizar las pruebas y la habilidad de cambiar entre diferentes implementaciones de una capa.
- Independencia pues elimina la necesidad de considerar el hardware y el despliegue así como las dependencias con interfaces externas.

Uso

- La aplicación debe soportar distintos tipos de clientes y distintos dispositivos.
- Quieres implementar reglas y procesos de negocio complejos o configurables.

El funcionamiento interno del Servicio Web es apoyado en el patrón arquitectónico N-Capas, específicamente sistema en 3 capas. El mismo separa la lógica del negocio, la vista y el modelo de datos en capas independientes, siendo la capa de presentación la que contiene el documento WSDL utilizado para describir lo que hace el Servicio Web. La misma se comunica a través de mensajes XML y de forma escalada, interactuando directamente con las peticiones hechas por un cliente, la capa de negocio procesa dichas solicitudes y finalmente la capa del modelo de datos obtiene la información que será presentada al cliente.

4.4 Patrones de Diseño

Según Craig Larman, los patrones de diseño son una descripción de un problema/solución, con un nombre y que puede emplearse en otros contextos, con una sugerencia sobre la manera de usarlo en situaciones nuevas. Por lo que se puede concluir que los patrones de diseño no son más que pautas definidas que aplican ciertos estilos con el objetivo de encaminar la construcción de un software. (Larman, et al., 2003)

4.4.1 Patrones GRASP

Los patrones GRASP describen la asignación de responsabilidades a objetos, expresados en forma de patrones. Para la construcción de la solución propuesta se seleccionaron los patrones siguientes:

- **Experto:** Encargado de asignar una responsabilidad al experto en información, la clase que cuenta con la información necesaria para cumplir la responsabilidad. Es la asignación de responsabilidades a cada clase, facilitando que los sistemas sean más fáciles de entender, mantener y ampliar. Presentando la oportunidad de reutilizar los componentes en futuras aplicaciones. Está presente en las clases *GestionCapa* y *GestionMetadatos* de manera que cada responsabilidad está asignada a la clase que cuenta con la información necesaria para realizarla.
- **Creador:** Se refiere a asignar a las clases la responsabilidad de crear una instancia de otra. Soportando mayor claridad y encapsulamiento. Este patrón se evidencia en la clase *Servicio*, pues la misma es la encargada de crear instancias.

- **Controlador:** Se encarga de asignar la responsabilidad del manejo de los eventos de un sistema a una clase. Es un objeto de interfaz no destinado al usuario que se encarga de manejar un evento del sistema. Permite, un mayor potencial de los componentes reutilizables. Presente en la clase *Servidor*, que es la encargada de atender los eventos del sistema que son generados por un actor externo y se asocian a operaciones del sistema.
- **Bajo acoplamiento:** Es asignar una responsabilidad para mantener el bajo acoplamiento. Mantener las clases lo menos ligadas posibles, es la medida de la fuerza con que una clase está conectada a otras. Soporta el diseño de clases más independientes, que reducen el impacto de los cambios, y también más reutilizables, acrecentando la oportunidad de una mayor productividad. Está presente en todas las clases del proyecto.
- **Alta cohesión:** Se refiere a asignar una responsabilidad de modo que la cohesión siga siendo alta. Es la medida de cuán relacionadas y enfocadas están las responsabilidades de una clase. Una clase con alta cohesión es fácil darle mantenimiento, entenderla y reutilizarla. Está presente en todas las clases del proyecto.

4.5 Desarrollo de las iteraciones.

En la fase de planificación de la metodología XP se detallaron cada una de las HU correspondientes a cada iteración según la selección del cliente. Durante el desarrollo de las iteraciones se va realizando una revisión del plan de iteraciones producto de que el mismo puede o no ser modificado. Como parte de este plan se crean tareas para ayudar a organizar la implementación exitosa de la HU.

Estas tareas en la que se descomponen las HU son las llamadas tareas de ingeniería, que son asignadas a una persona perteneciente al equipo de desarrollo, las mismas son escritas por los programadores y no por el cliente, dado que son para uso estricto de los programadores.

La planificación que se llevó a cabo para el desarrollo del Servicio Web, está compuesta por dos iteraciones, permitiendo que al final de la última iteración se logre un producto con todas las restricciones y características deseadas por el cliente. A continuación se detallan cada una de las tareas de la ingeniería por iteraciones.

4.5.1 Iteración 1:

Esta iteración tiene como objetivo darle cumplimiento a las HU que se consideraron de mayor importancia para el desarrollo del software. Al concluir dicha iteración se contará con todas las funcionalidades descritas en las HU #1, #2 y #3.

Historias de Usuario	Tiempo de implementación (semanas)	
	Estimación	Real
Recuperar metadatos asociados a una capa	1	1
Recuperar capas asociadas a un metadato	1	1
Recuperar metadatos dado un identificador de metadato	1	1

Tabla 7: Abordadas en la primera iteración

A continuación mediante tablas se evidencian las tareas de programación o ingeniería en las que las Historias de Usuario mencionadas anteriormente fueron desglosadas, para un mejor funcionamiento del Servicio Web, de las cuales se presentan tres, las demás se encuentran en los Anexos

Tareas de la Ingeniería	
No. de la tarea: 1.	No. de la HU: 1.
Nombre de la tarea: Consultar a la base de datos.	
Tipo de tarea: Desarrollo.	Puntos estimados: 1/2
Fecha inicio: 1-5-2013.	Fecha fin: 4-5-2013.
Programador responsable: Bárbaro Orestes Ramos Díaz	
Descripción: La tarea permite recibir una capa para buscar en la base datos los metadatos asociados a dicha capa.	

Tabla 8: Tarea #1 de la HU #1.

Tareas de la Ingeniería	
No. de la tarea: 2.	No. de la HU: 1.
Nombre de la tarea: Devolver los metadatos asociados a una capa.	
Tipo de tarea: Desarrollo.	Puntos estimados: 1/2
Fecha inicio: 4-5-2013.	Fecha fin: 8-5-2013.
Programador responsable: Bárbaro Orestes Ramos Díaz	
Descripción: La tarea devuelve los metadatos que tienen asociados la capa geográfica en la base de datos.	

Tabla 9: Tarea #2 de la HU #1.

4.5.2 Iteración 2:

Esta iteración tiene como finalidad desarrollar las HU #4, #5 y #6.

Historias de Usuario	Tiempo de implementación (semanas)	
	Estimación	Real
Asociar capas geoespaciales a metadatos	1	1
Recuperar metadatos existentes	1	1
Recuperar capas existentes	1	1

Tabla 10: Abordadas en la segunda iteración.

Tareas de la Ingeniería	
No. de la tarea: 1	No. de la HU: 4
Nombre de la tarea: Asociar capas geoespaciales a metadatos.	
Tipo de tarea: Desarrollo.	Puntos estimados: 1
Fecha inicio: 22-5-2013	Fecha fin: 29-5-2013

Programador responsable: Bárbaro Orestes Ramos Díaz.
Descripción: La tarea permite asociar la capa y el metadato ingresado por el usuario.

Tabla 11: Tarea #1 de la HU #4

4.6 Requisitos no funcionales.

Se entiende por requisito no funcional a la característica requerida del sistema, dígase del proceso de desarrollo, del servicio prestado o de cualquier otro aspecto del desarrollo, que señala una restricción del mismo. El buen funcionamiento y la calidad de la solución propuesta en el diseño de la presente investigación, no sólo depende de las capacidades o condiciones que debe cumplir el sistema, sino que también procura ser más atractivo al cliente, contando con una aplicación de fácil manipulación, segura y rápida.

Requisitos de usabilidad.

- El Servicio Web debe mostrar la información de forma coherente y correctamente estructurada.
- El Servicio Web debe contar con una documentación que permita el completo entendimiento del mismo y sus funcionalidades.

Requisitos de disponibilidad

El Servicio Web estará disponible en todo momento siempre y cuando el servidor donde se encuentre instalado se encuentre en funcionamiento.

Requisitos de eficiencia

El tiempo de respuesta estará dado por la cantidad de información a procesar, entre mayor cantidad de información almacenada en la base de datos, mayor será el tiempo de procesamiento y respuesta.

Requisitos de software

El servidor donde estará instalado el Servicio Web deberá contar con un Sistema Operativo y sus actualizaciones de seguridad. Deberá estar presente un servidor Web para atender las conexiones al mismo

debido a que la misma funcionará bajo los conceptos de arquitectura cliente/servidor. Por tal motivo se propone que el servidor tenga instalado los siguientes elementos.

1. Ubuntu server o Windows server.
2. Apache 2.2 o superior configurado con las extensiones pgsq.
3. PHP 5 o superior debido que tiene embebido el soporte para Servicios Web.

Requisitos de hardware

Se propone como requisitos mínimos de hardware poseer una tarjeta de red, un micro de tipo “Procesador x86 a 700 MHz.”, 1 GB de RAM y al menos 10 GB de espacio de almacenamiento y en sistemas con volúmenes de peticiones grandes se recomienda dedicar 20 GB. Estos datos están respaldados por (Ortiz, 2008) ya que con los mismos se pueden gestionar 256 procesos hijos httpd ¹¹que pueden ser procesados simultáneamente, cada proceso puede utilizar unos 15 a 20 MB de memoria física.

Requisitos de seguridad

Garantizar la validación de los datos que son entrados por los usuarios al Servicio Web. El mismo contará con un método que limpiará los datos de entradas para impedir las inyecciones SQL, dicho método se invocará antes de hacer una consulta a las bases de datos del SyGMe pasando los datos enviados por el usuario que utiliza el Servicio Web.

4.7 Pruebas.

Uno de los pilares de la Programación Extrema es el proceso de pruebas. XP anima a probar constantemente tanto como sea posible. Esto permite aumentar la calidad de los sistemas reduciendo el número de errores no detectados y disminuyendo el tiempo transcurrido entre la aparición de un error y su detección. También permite aumentar la seguridad de evitar efectos colaterales no deseados a la hora de realizar modificaciones y refactorizaciones.

¹¹ httpd: Almacena información acerca de diversas funciones del servidor, que pueden añadirse o eliminarse.

XP divide las pruebas del sistema en dos grupos: pruebas unitarias, encargadas de verificar el código y diseñada por los programadores así como pruebas de aceptación o pruebas funcionales destinadas a evaluar si al final de una iteración se consiguió la funcionalidad requerida diseñadas por el cliente final. (J.J. Gutierrez, 2012)

4.7.1 Pruebas de aceptación.

Las pruebas de aceptación son más importantes que las pruebas unitarias dado que significan la satisfacción del cliente con el producto desarrollado y el final de una iteración y el comienzo de la siguiente. Se elaboran a lo largo de la iteración, en paralelo con el desarrollo del sistema y adaptándose a los cambios que el sistema sufra. Los clientes son responsables de verificar que los resultados de estas pruebas sean correctos. Así mismo, en caso de que fallen varias pruebas, deben indicar el orden de prioridad de resolución. Una historia de usuario no se puede considerar terminada hasta tanto pase correctamente todas las pruebas de aceptación. (J.J. Gutierrez, 2012)

Para garantizar la correcta implementación de las pruebas se utilizó el plugin SOA Client de Firefox, el cual mediante la interfaz para el uso de Servicios Web que brinda, permitió validar las respuestas a las distintas funcionalidades que contiene el Servicio Web, logrando validar su correcto funcionamiento. Se realizó una prueba piloto a un total de 5 profesionales pertenecientes a la Línea de Productos de Software Aplicativos SIG para comprobar el nivel de aceptación de las funcionalidades, Se realizaron dos iteraciones arrojando los siguientes datos:

Como se puede observar en la gráfica, los resultados obtenidos no fueron del todo satisfactorios, teniendo un nivel de aceptación de un 80% en la primera iteración, donde luego en una segunda iteración se logró el 100% de aceptación por parte de los clientes.

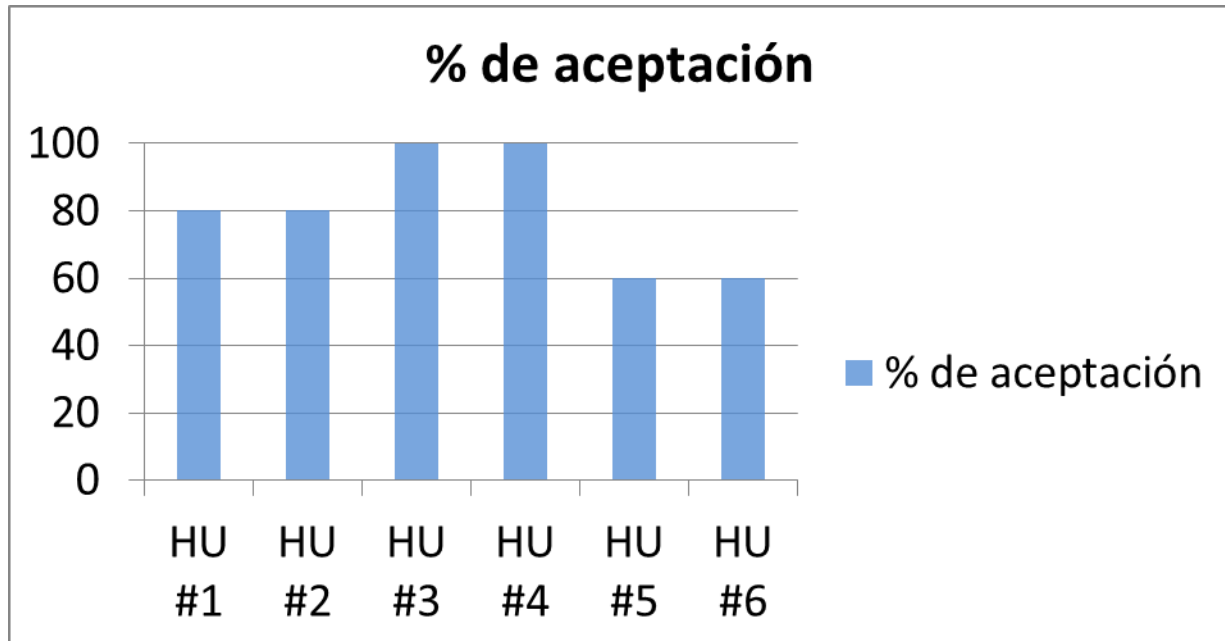


Ilustración 8: Prueba de aceptación

A continuación se presenta las tablas de caso de pruebas realizadas al Servicio Web correspondiente a la HU #1, los demás casos de prueba se encuentran en los anexos.

Caso de prueba de aceptación	
Código: HU1_PA1	No. De la HU: 1
Nombre: Recuperar metadatos asociados a una capa	
Descripción: Permite al usuario obtener todos los metadatos asociados a una capa determinada	
Condiciones de Ejecución: El cliente accede a la funcionalidad.	
Entrada/ Pasos de ejecución: El cliente ingresa el nombre de la capa geoespacial almacenada en la base de datos.	
Resultado Esperado: El Servicio Web debe devolver los metadatos asociados a partir de una capa determinada.	
Evaluación de la Prueba: Prueba desfavorable.	

Tabla 12: HU1_PA1

Caso de prueba de aceptación	
Código: HU1_PA2	No. De la HU: 1
Nombre: Recuperar metadatos asociados a una capa	
Descripción: Permite al usuario obtener todos los metadatos asociados a una capa determinada	
Condiciones de Ejecución: El cliente accede a la funcionalidad.	
Entrada/ Pasos de ejecución: El cliente ingresa el nombre de la capa geoespacial almacenada en la base de datos.	
Resultado Esperado: El Servicio Web debe devolver los metadatos asociados a partir de una capa determinada.	
Evaluación de la Prueba: Prueba satisfactoria.	

Tabla 13: HU1_PA2

4.8 Conclusiones Parciales.

Mediante la realización de construcciones y validaciones en una versión de la solución propuesta que corresponde con la aceptación final por parte del cliente luego de haber concluido las 2 iteraciones, se obtuvieron resultados satisfactorios. Se desarrollaron las fases de construcción y prueba de la metodología XP, describiéndose las tarjetas CRC para lograr el mayor entendimiento posible del Servicio Web. Para la comprobación del funcionamiento del software se realizó un estudio de las pruebas de aceptación arrojando un alto grado de aceptación del cliente respecto a al producto.

CONCLUSIONES GENERALES

Luego de culminar todo el proceso de construcción del Servicio Web que permite la interoperabilidad entre Sistemas de Información Geográfica y la base de datos de SyGMe, se arribaron a las siguientes conclusiones:

- El análisis de la situación en el Centro respecto al trabajo de metadatos con la herramienta SyGMe y el desarrollo de los SIG por parte del proyecto Aplicativos SIG permite concluir que efectivamente existe un problema respecto al intercambio de datos entre los sistemas desarrollados en el Proyecto y la BD del SyGMe.
- Las tecnologías y herramientas utilizadas para la construcción del sistema contribuyeron a lograr el éxito de la aplicación, garantizando que funcione según las necesidades actuales del centro GEYSED correspondientes a las tareas relacionadas con la información geográfica. Asimismo la utilización de herramientas libres y de elevada calidad permiten que el Servicio Web desarrollado sea una herramienta libre, de calidad y libre del pago de licencias a entidades privadas.
- Los artefactos generados como las Historias de Usuario permitieron un mayor entendimiento entre el cliente y el desarrollador para elaborar con éxito las funcionalidades que el sistema debe cumplir. Además las tarjetas CRC y las tarjetas de ingeniería dieron paso al desarrollador para la implementación de las HU. Conjuntamente la documentación generada a lo largo de la investigación permite que el Servicio Web pueda ser entendido por los demás desarrolladores del centro GEYSED.
- Con la implementación del Servicio Web, el departamento GeoInformática puede contar con un Servicio Web propio, que responda a los intereses de su centro correspondiente a la organización y búsqueda de metadatos geográficos.
- La realización de pruebas al sistema permitieron obtener un sistema de elevada calidad y eficiencia, libre de errores, por lo que puede ser puesto en explotación inmediatamente. Asimismo el Servicio Web propuesto cumple con las expectativas del cliente y enriquece los conocimientos existentes en la Universidad y el País respecto a esta rama de la informática, encaminando al País a cada vez estar más cerca de la soberanía tecnológica.

RECOMENDACIONES

Partiendo de la experiencia acumulada durante todo el período de desarrollo de la investigación y de los resultados obtenidos, se proponen las siguientes recomendaciones:

- Desarrollar una versión superior del producto donde se brinden nuevas funcionalidades de búsqueda e intercambio de metadatos geográficos.
- Permitir al Servicio Web la exportación de metadatos geográficos basado en la norma ISO 19139 para un mayor grado de satisfacción de los clientes.
- Incluir el Servicio Web desarrollado mediante un módulo a la herramienta SyGMe.

REFERENCIAS BIBLIOGRÁFICAS

Bibliografía

Departamento de prensa. 2012. Universo. [En línea] Universidad de Veracruz- Mexico, 11 de junio de 2012. [Citado el: 24 de 3 de 2013.] http://www.uv.mx/universo/486/infgral/infgral_15.html.

Aries, Benjamin. 2012. eHow en Español. [En línea] 2012. [Citado el: 16 de mayo de 2013.] http://www.ehowenespanol.com/ventajas-apache-web-server-lista_109947/.

Associates, O'Reilly &. 2003. *Extreme Programming Pocket Guide*. s.l. : edición, 2003. pág. 80 .

Bohorquez, Hanller Cueto y Valenzuela, Johan Ferney. 2005. Telecentro regional de Tecnologías Espaciales. *Fundamentos de Infraestructura de Datos Espaciales*. [En línea] 2005. [Citado el: 17 de noviembre de 2012.] <http://geoservice.igac.gov.co/moodle/course/info.php?id=23..>

Brea, Orlando Fabián. 2005. [En línea] marzo de 2005. [Citado el: 2 de abril de 2013.] <http://www.desarrolloweb.com/manuales/61/>.

Caballeros, Jerez de los. 2008. Revista Internacional Digital de las Ciencias de la Tierra. [En línea] 2008. [Citado el: diciembre de 16 de 2012.] Mappinginteractivo.com.

Callejo, Miguel Angel Manso. 2009. *El uso de los metadatos para el desarrollo de un modelo de interoperabilidad para las infraestructura de datos*. Madrid : s.n., 2009.

Cesar de la Torre Llorente, Unai Zorrilla Castro, Miguel Angel Ramos Barroso, Miguel Calvarro Nelson. 2010 . *Guía de Arquitectura N-Capas orientada al Dominio*. . 2010 .

CHACÓN, JULIO CÉSAR RUEDA. 2006. *APLICACIÓN DE LA METODOLOGÍA RUP PARA EL DESARROLLO RÁPIDO DE APLICACIONES*. Guatemala : Universidad de San Carlos de Guatemala. Facultad de Ingeniería, 2006.

Cisneros Díaz, Luís R. y Duffatt Candeaux, Rafael. 1994. *LOS SISTEMAS DE INFORMACIÓN GEOGRÁFICA*. 1994.

Committee, US Federal Geographic Data. Metadata Home Pag. [En línea] [Citado el: 21 de marzo de 2013.] <http://www.fgdc.gov/metadata/metadata.html>..

Committee, US Federal Geographic Data. Metadata Home Pag. [En línea] [Citado el: 12 de 2 de 2013.] <http://www.fgdc.gov/metadata/metadata.html>..

Español., Enciclopedia Libre Universal en. Enciclopedia Libre Universal en Español. Lenguaje de programación. [En línea] [Citado el: 18 de febrero de 2013.] http://enciclopedia.us.es/index.php/Lenguaje_de_programaci%C3%B3n..

ESRI, España. 2010. Geoportal Extensión. [En línea] 2010. [Citado el: 23 de 4 de 2013.] http://webhelp.esri.com/geoportal_extension/9.3.1/index.htm#ext_csw_clnts.htm.

Fernández, Sagrario Peralta. 2008. El servidor Web. Arquitectura y funcionamiento. [En línea] 2008. [Citado el: 20 de marzo de 2013.] <http://observatorio.cnice.mec.es/modules.php?op=modload&name=News&file=article&sid=366>..

García, Nelida Elba. 2008. *Metadatos : necesidad e importancia de integrar estándares*. Misiones, Argentina : s.n., 2008. pág. 15.

GEOGRÁFICO, CONSEJO SUPERIOR. 2012. *Catalog Services (CSW)*. Madrid : s.n., 2012.

Geosur. Arquitectura de los sistemas asociados al programa Geosur. [En línea] [Citado el: 18 de 5 de 2013.] (http://www.geosur.info/geosur/contents/Dise%F1o_de_arquitectura_del_Programa_GeoSUR.pdf).

Ibero, Ibáñez de. 2009. Instituto geográfico de España. [En línea] 2009. [Citado el: 18 de febrero de 2013.] <http://www.ign.es/ign/main/index.do>.

J.J. Gutierrez, M.J. Escalona, M.Mejias, J.Torres. 2012. Sistema de Programación Extrema. [En línea] 2012. [Citado el: 4 de mayo de 2013.] http://www.lsi.us.es/~javierj/investigacion_ficheros/PSISEXTREMA.pdf.

Jacobson, Ivar, Booch, Grady y Rumbaugh, James. 2000. *El proceso unificado de desarrollo de software*. Madrid : s.n., 2000.

- Javier J. Gutierrez, M. J. Escalona, M. Mejías, J. Torres.** SISTEMA DE PROGRAMACION EXTREMA. [En línea] [Citado el: 12 de 3 de 2013.] http://www.lsi.us.es/~javierj/investigacion_ficheros/PSISEXTREMA.pdf.
- Jerez de los Caballeros. 2009.** Revista Internacional Digital de las Ciencias de la Tierra. [En línea] 2009. [Citado el: 12 de 4 de 2013.] Mappinginteractivo.com.
- Larman, Craig y Hall, Prentice. 2003.** *UML y Patrones*. 2da Edición. 2003.
- Machuca, Carlos Andrés Morales. 2012.** *Estado del Arte: Servicios Web*. Colombia : s.n., 2012. pág. 18.
- Manso Callejo, Miguel Ángel. 2003.** *Metadatos en los sistemas de información geográfica (ISO-19115)*. Madrid : s.n., 2003.
- Métodologías ágiles para el desarrollo de software: eXtreme Programming (XP)*. **Penadés, Patricio Letelier y M^a Carmen.** Valencia : Universidad Politécnica de Valencia.
- Nebert, D., Whiteside. 2004.** Open Geospatial Consortium Inc. [En línea] 2004. [Citado el: 16 de marzo de 2013.] <http://www.opengeospatial.org>.
- O'Reilly. 2002.** *Web Services Essentials*. [ed.] Ethan Cerami. 1era edicion. 2002. pág. 304. ISBN: 0-596-00224-6.
- Ortiz, Rubén. 2008.** Apache. MaxClients y más.... [En línea] 13 de 5 de 2008. [Citado el: 10 de 6 de 2013.] <http://www.rubenortiz.es/2008/05/13/apache-maxclients-y-mas/>.
- Penadés, Patricio Letelier y M^a Carmen. 2013.** *Métodologías ágiles para el desarrollo de software: Extreme Programming (XP)*. Valencia : Universidad Politécnica de Valencia, 2013.
- Pereira, Nuvia Esperanza Villamizar. 2003.** *IMPLEMENTACIÓN DE UN SERVICIO WEB*. Barcelona : s.n., 2003. pág. 12.
- Pressman, Roger S. 2011.** *Software Engineering, a practitioner's approach*. 7th edición. s.l. : , McGraw-Hill, pp., 2011. pág. 895. ISBN 9780071267823.

Prezi. 2013. [En línea] 2013. [http://prezi.com/9hc9opeav71b/tarjetas-crc/..](http://prezi.com/9hc9opeav71b/tarjetas-crc/)

Real Academia Española. DICCIONARIO DE LA LENGUA ESPAÑOLA. [En línea] [Citado el: 24 de noviembre de 2012.] [http://buscon.rae.es/drae/..](http://buscon.rae.es/drae/)

Rodas, Raul Hinostroza. 2007. Usuarios y Grupos en Linux. . [En línea] 2007. [Citado el: 21 de diciembre de 2012.] . www.linuxcentro.net..

Sánchez Maganto, Alejandra, Noguerras, Javier y Ballari, Daniela. 2008. *Normas sobre metadatos (ISO19115, ISO19115-2, ISO19139, ISO 15836)*. Madrid : s.n., 2008.

The Arte of Service Orientation, SOA WebServices Journal Montvale. **Anantha Rangachar, Raghu. 2006.** s.l. : US: SYS-CON MEDIA, 2006, Vols. Vol 6 1535-6906.

Thompson, Ivan. 2008. Portal de Mercadotecnia. [En línea] 2008. [Citado el: 23 de Noviembre de 2012.] <http://www.promonegocios.net/mercadotecnia/definicion-informacion.html...>

Tidwell, Doug, snell, james y kulcheko, pavel. 2002. *Programing Web Services whit SOAP.* 2002.

Vadim Paz Madrid Gorelov, Juan Francisco De Paz Santana. *Servicios Web.* Salamanca, : s.n. pág. 16.

Valenciana, Generalitat. 2010 . CATÁLOGO DE METADATOS . [En línea] 2010 . <http://geocatalogo.cma.gva.es/geonetwork/srv/es/csw>.

W3C Consortium. 2006. Web Services Architecture. [En línea] Febrero de 2006. [Citado el: 22 de Abril de 2013.] <http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/#whatis..>

Wheeler, Veronika. 2008. Linux en Español. Entorno de desarrollo integrado. [En línea] 26 de marzo de 2008. [Citado el: 20 de diciembre de 2012.] <http://www.linuxespanol.com/viewtopic.php?p=96480&sid=c1e2f61c2c9f25c5ca0e492a6c87c62e..>

Yurtseven, Lale. 2007. GeoNetwork. [En línea] 2007. [Citado el: 18 de febrero de 2013.] <http://geonetwork-opensource.org>.

BIBLIOGRAFÍA CONSULTADA

Departamento de prensa. 2012. Universo. [En línea] Universidad de Veracruz- Mexico, 11 de junio de 2012. [Citado el: 24 de 3 de 2013.] http://www.uv.mx/universo/486/infgral/infgral_15.html.

Aries, Benjamin. 2012. eHow en Español. [En línea] 2012. [Citado el: 16 de mayo de 2013.] http://www.ehowenespanol.com/ventajas-apache-web-server-lista_109947/.

Associates, O'Reilly &. 2003. *Extreme Programming Pocket Guide*. s.l. : edición, 2003. pág. 80 .

Bohorquez, Hanller Cueto y Valenzuela, Johan Ferney. 2005. Telecentro regional de Tecnologías Espaciales. *Fundamentos de Infraestructura de Datos Espaciales*. [En línea] 2005. [Citado el: 17 de noviembre de 2012.] <http://geoservice.igac.gov.co/moodle/course/info.php?id=23..>

Brea, Orlando Fabián. 2005. [En línea] marzo de 2005. [Citado el: 2 de abril de 2013.] <http://www.desarrolloweb.com/manuales/61/>.

Caballeros, Jerez de los. 2008. Revista Internacional Digital de las Ciencias de la Tierra. [En línea] 2008. [Citado el: diciembre de 16 de 2012.] Mappinginteractivo.com.

Callejo, Miguel Angel Manso. 2009. *El uso de los metadatos para el desarrollo de un modelo de interoperabilidad para las infraestructura de datos*. Madrid : s.n., 2009.

Cesar de la Torre Llorente, Unai Zorrilla Castro, Miguel Angel Ramos Barroso, Miguel Calvarro Nelson. 2010 . *Guía de Arquitectura N-Capas orientada al Dominio*. . 2010 .

CHACÓN, JULIO CÉSAR RUEDA. 2006. *APLICACIÓN DE LA METODOLOGÍA RUP PARA EL DESARROLLO RÁPIDO DE APLICACIONES*. Guatemala : Universidad de San Carlos de Guatemala. Facultad de Ingeniería, 2006.

Cisneros Díaz, Luís R. y Duffatt Candeaux, Rafael. 1994. *LOS SISTEMAS DE INFORMACIÓN GEOGRÁFICA*. 1994.

Committee, US Federal Geographic Data. Metadata Home Pag. [En línea] [Citado el: 21 de marzo de 2013.] <http://www.fgdc.gov/metadata/metadata.html>..

Committee, US Federal Geographic Data. Metadata Home Pag. [En línea] [Citado el: 12 de 2 de 2013.] <http://www.fgdc.gov/metadata/metadata.html>..

Español., Enciclopedia Libre Universal en. Enciclopedia Libre Universal en Español. Lenguaje de programación. [En línea] [Citado el: 18 de febrero de 2013.] http://enciclopedia.us.es/index.php/Lenguaje_de_programaci%C3%B3n..

ESRI, España. 2010. Geoportal Extensión. [En línea] 2010. [Citado el: 23 de 4 de 2013.] http://webhelp.esri.com/geoportal_extension/9.3.1/index.htm#ext_csw_clnts.htm.

Fernández, Sagrario Peralta. 2008. El servidor Web. Arquitectura y funcionamiento. [En línea] 2008. [Citado el: 20 de marzo de 2013.] <http://observatorio.cnice.mec.es/modules.php?op=modload&name=News&file=article&sid=366>..

García, Nelida Elba. 2008. *Metadatos : necesidad e importancia de integrar estándares*. Misiones, Argentina : s.n., 2008. pág. 15.

GEOGRÁFICO, CONSEJO SUPERIOR. 2012. *Catalog Services (CSW)*. Madrid : s.n., 2012.

Geosur. Arquitectura de los sistemas asociados al programa Geosur. [En línea] [Citado el: 18 de 5 de 2013.] (http://www.geosur.info/geosur/contents/Dise%F1o_de_arquitectura_del_Programa_GeoSUR.pdf).

Google. 2013. Sites Google XP_Metodologías. [En línea] 2013. [Citado el: 2 de 3 de 2013.] <https://sites.google.com/site/xpmetodologia/marco-teorico/funcionamiento>.

Ibero, Ibáñez de. 2009. Instituto geográfico de España. [En línea] 2009. [Citado el: 18 de febrero de 2013.] <http://www.ign.es/ign/main/index.do>.

J.J. Gutierrez, M.J. Escalona, M.Mejias, J.Torres. 2012. Sistema de Programación Extrema. [En línea] 2012. [Citado el: 4 de mayo de 2013.] http://www.lsi.us.es/~javierj/investigacion_ficheros/PSISEXTREMA.pdf.

Jacobson, Ivar, Booch, Grady y Rumbaugh, James. 2000. *El proceso unificado de desarrollo de software*. Madrid : s.n., 2000.

Javier J. Gutierrez, M. J. Escalona, M. Mejías, J. Torres. SISTEMA DE PROGRAMACION EXTREMA. [En línea] [Citado el: 12 de 3 de 2013.] http://www.lsi.us.es/~javierj/investigacion_ficheros/PSISEXTREMA.pdf.

Jerez de los Caballeros. 2009. Revista Internacional Digital de las Ciencias de la Tierra. [En línea] 2009. [Citado el: 12 de 4 de 2013.] Mappinginteractivo.com.

Larman, Craig y Hall, Prentice. 2003. *UML y Patrones*. 2da Edición. 2003.

Machuca, Carlos Andrés Morales. 2012. *Estado del Arte: Servicios Web*. Colombia : s.n., 2012. pág. 18.

Manso Callejo, Miguel Ángel. 2003. *Metadatos en los sistemas de información geográfica (ISO-19115)*. Madrid : s.n., 2003.

Métodologías ágiles para el desarrollo de software: eXtreme Programming (XP). **Penadés, Patricio Letelier y M^a Carmen.** Valencia : Universidad Politécnica de Valencia.

Nebert, D., Whiteside. 2004. Open Geospatial Consortium Inc. [En línea] 2004. [Citado el: 16 de marzo de 2013.] <http://www.opengeospatial.org>.

NetBeans IDE. 2012. Pagina Oficial. [En línea] 2012. [Citado el: 15 de Diciembre de 2012.] <http://www.netbeans.org/about/index.html>.

O'Reilly. 2002. *Web Services Essentials*. [ed.] Ethan Cerami. 1era edicion. 2002. pág. 304. ISBN: 0-596-00224-6.

Ortiz, Rubén. 2008. Apache. MaxClients y más.... [En línea] 13 de 5 de 2008. [Citado el: 10 de 6 de 2013.] <http://www.rubenortiz.es/2008/05/13/apache-maxclients-y-mas/>.

Penadés, Patricio Letelier y M^a Carmen. 2013. *Métodologías ágiles para el desarrollo de software: Extreme Programming (XP)*. Valencia : Universidad Politécnica de Valencia, 2013.

- Pereira, Nuvia Esperanza Villamizar. 2003.** *IMPLEMENTACIÓN DE UN SERVICIO WEB.* Barcelona : s.n., 2003. pág. 12.
- Pressman, Roger S. 2011.** *Software Engineering, a practitioner's approach.* 7th edición. s.l. : , McGraw-Hill, pp., 2011. pág. 895. ISBN 9780071267823.
- Real Academia Española.** DICCIONARIO DE LA LENGUA ESPAÑOLA. [En línea] [Citado el: 24 de noviembre de 2012.] [http://buscon.rae.es/drae/..](http://buscon.rae.es/drae/)
- Rodas, Raul Hinostroza. 2007.** Usuarios y Grupos en Linux. . [En línea] 2007. [Citado el: 21 de diciembre de 2012.] . www.linuxcentro.net..
- Sánchez Maganto, Alejandra, Nogueras, Javier y Ballari, Daniela. 2008.** *Normas sobre metadatos (ISO19115, ISO19115-2,ISO19139, ISO 15836).* Madrid : s.n., 2008.
- The Arte of Service Orientation, SOA WebServices Journal Montvale.* **Anantha Rangachar, Raghu. 2006.** s.l. : US: SYS-CON MEDIA, 2006, Vols. Vol 6 1535-6906.
- Richardson, Leonard. Ruby, Sam.** RESTful Web Services.O'Reilly Media, May 2007,ISBN-10: 0-596-52926-0.
- Sandoval,Jose.***RESTfulJava Web Services.* BIRMINGHAM -MUMBAI,Packt Publishing,Nov2009. ISBN 978-1-847196-46-0.
- Smiley, David. Pugh, Eric.** Apache Solr 3 Enterprise Search Server. BIRMINGHAM - MUMBAI Packt Publishing, Aug 2009, ISBN 978-1-84951-606-8
- SOAP y WebServices.** 13 de Enero de 2003.
<http://www.microsoft.com/spanish/msdn/comunidad/mtj.net/voices/art51.asp>.
- Thompson, Ivan. 2008.** Portal de Mercadotecnia. [En línea] 2008. [Citado el: 23 de Noviembre de 2012.] <http://www.promonegocios.net/mercadotecnia/definicion-informacion.html...>
- Tidwell, Doug, snell, james y kulcheko, pavel. 2002.** *Programing Web Services whit SOAP.* 2002.

Vadim Paz Madrid Gorelov, Juan Francisco De Paz Santana. *Servicios Web*. Salamanca, : s.n. pág. 16.

Valenciana, Generalitat. 2010 . CATÁLOGO DE METADATOS . [En línea] 2010 .
<http://geocatalogo.cma.gva.es/geonetwork/srv/es/csw>.

W3C Consortium. 2006. Web Services Architecture. [En línea] Febrero de 2006. [Citado el: 22 de Abril de 2013.]
<http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/#whatis..>

W3C, Consortium. 2006. Web Services Architecture. [En línea] Febrero de 2006. [Citado el: 22 de Abril de 2013.]
<http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/#whatis..>

Wheeler, Veronika. 2008. Linux en Español. Entorno de desarrollo integrado. [En línea] 26 de marzo de 2008. [Citado el:
20 de diciembre de 2012.]
<http://www.linuxespanol.com/viewtopic.php?p=96480&sid=c1e2f61c2c9f25c5ca0e492a6c87c62e..>

Yurtseven, Lale. 2007. GeoNetwork. [En línea] 2007. [Citado el: 18 de febrero de 2013.] <http://geonetwork-opensource.org>.

GLOSARIO DE TÉRMINOS

Web: Su traducción es telaraña o red, y se refiere a la red de comunicación por todo el mundo.

Mapa: es una representación reducida de todo o parte del planeta Tierra. Este, constituye un esbozo gráfico, de medidas estrictas, acerca de la información que es extraída en un territorio determinado.

Dato Espacial: Un dato espacial, es una variable asociada a una determinada localización geográfica, que posee características específicas del lugar referido, como: color, forma, ubicación en el espacio tridimensional, entre otras.

SIG: Sistema de información geográfica.

XML: Lenguaje extensible de Marcado, o en sus siglas en inglés: Extensible Markup Language.

SOAP: Es un protocolo de la capa de aplicación para el intercambio de mensajes basados en XML.

WSDL: Es un tipo de documento XML que describe lo que hace un Servicio Web.

UDDI: Servidor donde están alojados Servicios Web.

CSW: Servicio utilizado para la catalogación de metadatos en IDE.

CRC: Clase + Responsabilidad + Colaboración.

PHP: Es un lenguaje de programación de uso general del lado del servidor, acrónimo que significa Hypertext Pre-processor

LPS: Línea de Productos de Software.