

ESPE-DECC

“DECC *Report*, Tendencias en Computación”



ESPE
ESCUELA POLITÉCNICA DEL EJÉRCITO
CAMINO A LA EXCELENCIA

**REVISTA TÉCNICA DEL DEPARTAMENTO DE
CIENCIAS DE LA COMPUTACIÓN.**

ISSN 1390-5236

© 2011, ESPE, Sangolquí-Ecuador

VICERRECTORADO ACADÉMICO.

VOL. 1, No. 3, 2011

RECTOR

GENERAL DE BRIGADA Carlos Rodríguez Arrieta

VICERRECTOR ACADEMICO

CRNL. EMC. Carlos Gutiérrez

VICERRECTOR DE INVESTIGACIÓN Y VINCULACIÓN CON LA COLECTIVIDAD.

CRNL. EMC Mauricio Chávez Ch.

GERENTE ADMINISTRATIVO FINANCIERO

CRNL. CSM. Juan Domínguez

DIRECTOR DEL DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN.

CRNL. EMC.(SP) Fidel Castro De la Cruz

EDITOR

Ing. Walter M. Fuertes D., Ph.D.

Director de la Unidad de Gestión de Postgrado

Escuela Politécnica del Ejército

Sangolquí Ecuador

e_mail: wmfuertes@espe.edu.ec

Revisión del Idioma Inglés

Ing. Ramiro Delgado, MSc.

E_mail: rdelgado@espe.edu.ec

COMITE EDITORIAL

Ing. Ramiro Delgado

Coordinador de Investigación del Departamento de Ciencias de la Computación.

Ing. Mario Ron Egas

Coordinador de la Línea de Investigación de Software Aplicado

Ing. Germán Ñacato

Jefe del Laboratorio de Multimedia

Ing. Mario Almache

Docente Departamento de Ciencias de la Computación.

Ing. Tatiana Gualotuña

Docente Departamento de Ciencias de la Computación.

Portada:	Ing. Germán Ñacato
Diagramación:	Varios autores
Impresión:	Editorial Politécnica de la ESPE
Fecha:	Diciembre de 2011



“LA TECNOLOGÍA ES LA QUE MARCA LA DIFERENCIA EN EL DESARROLLO DE LAS NACIONES. SI NO PODEMOS SER UNA POTENCIA, DEBEMOS SENTIR LA OBLIGACIÓN DE SER UNA NACIÓN DESARROLLADA”

La Historia nos ha ensañado, a través de lo acontecido con los Países denominados Potencias Mundiales, que la automatización, marca el camino hacia un desarrollo sostenible, nos recuerda que las Tecnologías de la Información y Comunicación sirven como un importante instrumento para conseguir el progreso económico. Se ha demostrado que las tecnologías no costosas, en manos de las comunidades locales, pueden ser eficientes motores de cambio, tanto en lo social como en lo material.

Es axiomático mencionar que los campos de acción, en que los sistemas computacionales intervienen, son innumerables e incluso se podría decir, que no existe ámbito en donde las tecnologías que nos ocupan no sean determinantes en la actividad. Hoy, este sometimiento a la tecnología, ha llegado a tal punto que las infraestructuras que soportan el diario vivir de las naciones y sus poblaciones, dependen cada vez más de las infraestructuras tecnológicas.

Esto me hace percibir que en la actualidad estamos siendo afectados por una compleja sistematización de tecnologías, que ha desembocado en la llamada cerebralización de la máquina, es decir, un momento de la historia en el que la actual revolución tecnológica logra que la máquina desempeñe el papel del cerebro y realice funciones cada vez más numerosas e importantes, así como en la era de la revolución industrial los seres humanos cambiamos el músculo por la máquina.

Definitivamente, todo esto conlleva a que exista una continua investigación y una mayor implantación de tecnologías, en las diferentes organizaciones de la sociedad, y es en este punto precisamente, donde las universidades debemos estar presente. De ahí nuestro papel fundamental para que en un esfuerzo mancomunado estudiantes, docentes y autoridades demos ese salto decisivo, hacia la era del conocimiento y nos constituyamos en un país con visión de prospectiva acorde al avance mundial, caso contrario, estaremos condenados a ser un país sumiso, un país maniatado por intereses particulares y con la única capacidad de ofrecer materia prima a las grandes naciones.

Aprovecho esta oportunidad en la que se edita el Vol.1, No.3, 2011 de la Revista DCC- Report “Tendencias en Computación”, para expresar mi fraterno saludo y admiración a los Docentes y Servidores Públicos del Departamento de Ciencias de la Computación, y en especial mi felicitación al Comité Editorial y a todos quienes colaboraron en la difusión de artículos técnico-científicos, quienes han demostrando su esfuerzo y dedicación para coadyuvar en el incremento de la producción científica de nuestra Universidad de Fuerzas Armadas ESPE; así como el sincero anhelo de que día a día sigamos trabajando en busca de mejores objetivos, de alternativas de progreso y transformación, que nos permitan tener un futuro promisorio y de verdadero desarrollo nacional.

**Ing. Fidel Castro De la Cruz, MSc.
Crnl. EMC. (SP)
DIRECTOR DEL DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN**

Presentación



El Departamento de Ciencias de la Computación (DECC) de la Escuela Politécnica del Ejército (ESPE) con el fin de difundir los resultados del quehacer académico-investigativo, pone en consideración de la comunidad el Volumen 1, No. 3, 2011, de la revista técnica “**DECC Report, Tendencias en Computación**”.

En este volumen se recogen los resultados de proyectos de iniciación científica desarrollados en el DECC, las investigaciones y tesis de pregrado y postgrado de la carrera de Ingeniería de Sistemas y de los programas de Maestría en Redes de Información y Conectividad y Maestría en Gerencia de Sistemas respectivamente. Comprende 8 reportes técnicos por egresados, maestrantes y profesores del DECC, los mismos que atravesaron por un proceso de selección, revisión y arbitraje. En números, se presentaron 70 artículos técnicos. En su primera evaluación fueron eliminados 30 por parte del Comité Editorial de la revista. Los restantes fueron enviados a revisión por pares de colegas expertos en las diferentes áreas, de los cuales solamente fueron aceptados 8. Estas publicaciones reportan trabajos técnicos-científicos en Ingeniería de Software, IPTV, Multimedia, Tecnologías de Virtualización, Redes de Computación, Mensajería Electrónica, Desarrollo Web y Web 2.0.

Aprovecho esta ocasión para agradecer todo el apoyo técnico brindado por el Ing. Ramiro Delgado, Coordinador del DECC, en la realización del presente volumen.

Por lo expuesto, “**DECC Report, Tendencias en Computación**” constituye un medio de difusión local y nacional, cuya información esperamos resultará de interés para docentes, investigadores y estudiantes, invitándoles a aprovechar su contenido y a continuar enviando sus contribuciones en las siguientes ediciones semestrales.

En este mismo ámbito, en mi calidad de Coordinador, tengo a bien invitar a docentes, investigadores y estudiantes a que participen activamente en el **Grupo de Investigación de Sistemas Distribuidos del DECC**, grupo que fue formalizado ante el Consejo de Investigación de la ESPE, en abril de 2011.

El Grupo de Investigación en Sistemas Distribuidos tiene como objetivo proponer, investigar, ejecutar y evaluar proyectos de investigación científico-técnico relacionados con el desarrollo de Sistemas Distribuidos, Ingeniería de Software, Redes de Computación, Tecnologías de virtualización, Seguridades Informáticas, Grid y Computación Distribuida en forma general. Además está orientado a integrar al docente y estudiante del Departamento de Ciencias de la Computación a las actividades de generación y aplicación del conocimiento. Sus miembros son:



Ing. Tatiana Gualotuña



Msc. Luis Guerra



MSc. Diego Marcillo



Ing. Fausto Meneses



MSc. Hernán Aules

**Ing. Walter Fuertes D., Ph.D.
Editor.**

Sumario:

Volumen 1, No. 3, 2011

ARTICULO TÉCNICO	PÁGINAS
Sistema Web de Asignación de Aulas de los Laboratorios de Computación de la ESPE, Aplicando la Metodología Agile Unified Process (AUP), utilizando el Framework Junit Tatiana Pozo, Carlos Aucancela, Cecilia Hinojosa, Aly Abdelrahaman	6 - 14
Prototipo de la Tecnología IPTV mediante un Servidor de Contenidos Multimedia en Linux Juan Zarria, Carlos Cevallos, Darwin Aguilar y Diego Marcillo	15 - 23
Desarrollo de una aplicación interactiva aplicando la metodología OpenUP/Basic como parte del Proyecto ESPE-GINGA Ángel Quingaluisa, Jonathan Torres, Danilo Martínez y Santiago Salvador	24 - 31
La importancia de la Gestión de la Configuración del Software, en una Empresa de Desarrollo. Santiago Paredes, Cecilia Hinojosa, Jenny Ruiz	32 - 41
Sistema Cero Papeles para la Revisión de Correspondencia del Banco COFIEC SA. R. Garcés, C. Hinojosa y J. Ruíz	42 - 52
Capacidad de Conmutación de los Equipos Activos para una Red LAN Corporativa Ángel Chinchero Villacís	53 - 65
Marco de Referencia para la Gestión de Proyectos de Desarrollo de Software Sandra Chávez	66 - 73
Análisis, Diseño y Desarrollo del Sistema de Monitorización para el Servicio de Comunicaciones Unificadas en Elastix 2 Antonio Insuasti, Santiago Salvador, Diego Marcillo	74 - 79

Sistema Web de Asignación de Aulas de los Laboratorios de Computación de la ESPE, Aplicando la Metodología Agile Unified Process (AUP), utilizando el Framework Junit

Tatiana Pozo, Carlos Aucancela, Cecilia Hinojosa y Aly Abdelrahman

*Departamento de Ciencias de la Computación, Escuela Politécnica del Ejército, Sangolquí, Ecuador,
pandy1517@hotmail.com, carlos_javier_18@hotmail.com, chinojosa@espe.edu.ec, abdito_8@hotmail.com*

RESUMEN: El presente trabajo está orientado a realizar un estudio del desarrollo guiado por pruebas, o Test Driven Development (TDD), que es una técnica de programación que involucra principalmente dos prácticas: escribir las pruebas primero (Test First Development) y refactorización (Refactoring), aplicado al desarrollo de una herramienta informática que permita minimizar la gestión de reservas de los laboratorios computacionales de la ESPE. El procedimiento de esta técnica empieza por escribir una prueba acorde a un requerimiento específico, en la cual se verifica que dichas pruebas fallen, luego se implementa el mínimo código que haga que la prueba pase satisfactoriamente para proceder a re-factorizar el código escrito si fuere necesario. La aplicación práctica de TDD fue realizada en el desarrollo de un sistema Web de reserva de laboratorios computacionales de la ESPE (SILVERLAB), siguiendo los lineamientos de la metodología de desarrollo de software AUP, la cual se basa en cortas iteraciones, desde el levantamiento de requisitos, análisis, diseño e implementación. Para las pruebas se utilizó la herramienta JUNIT con la finalidad de verificar, manejar y ejecutar conjuntos de pruebas automatizadas. El producto software resultante es un sistema Web distribuido en lenguaje de programación JAVA, basado en el patrón de diseño Modelo Vista Controlador que permite separar en componentes dicho sistema, además posee un motor de base de datos MYSQL. Tras el desarrollo de la aplicación se evidenció las ventajas que provee TDD tales como: Permitir identificar lo que es realmente imprescindible implementar por lo que se ahorrará tiempo desarrollando código que luego no se usará; Ayuda al programador a tener un mayor nivel de confianza en el código desarrollado; Fuerza a un estricto análisis y diseño, ya que el desarrollador no puede crear código de producción sin entender realmente cuales deberían ser los resultados deseados y como probarlos; El conjunto de test unitarios proporciona constante retroalimentación de que cada uno de los componentes sigue funcionando.

Palabras Clave: TDD, Re-factorizar, JUnit, UnitTest.

ABSTRACT: This paper aims to conduct a study of test driven development (TDD), which is a programming technique that mainly involves two practices: writing the first test and code refactoring applied to the development of a software tool that allows to minimize the management computer labs of the ESPE. The procedure of this technique begins with a writing of a test according to a specific requirement, which verifies that these tests fail, and then write the minimum code that successfully pass the test and then proceed to re-factor the code written. The practical application of TDD was made in the development of a ESPE Web booking computational laboratories (SILVERLAB), following the guidelines of the software development methodology AUP, which is based on short iterations, from the requirements lifting, analysis, design and implementation. For testing tool was used JUNIT in order to verify, manage and execute automated test suites. The resulting software product is a distributed Web system in java programming language, based on the standard model view controller design for separating the system components; also it has a database engine MySQL. After the development of the application was demonstrated the advantages of TDD such as: allows identifying what is really essential to implement so it will save time developing code that not be used later; helps the developer to have a higher level of confidence in the code developed; forces to a strict analysis and design, as the developer cannot create production code without really understanding what should be the test result as desired. The test unit set provides constant feedback from each of the components is still running.

Keywords: TDD, Refactoring, JUnit, UnitTest.

1. INTRODUCCIÓN

El proceso de desarrollo de software de calidad se ve opacado cuando no existe el suficiente diálogo o un inadecuado entendimiento de las necesidades de los usuarios, ya que la fase de pruebas es realizada después de la etapa de desarrollo, implicando grandes esfuerzos al momento de realizar cambios en los requisitos.

El desarrollo dirigido por pruebas (TDD) es una técnica que se acopla a las metodologías ágiles, destacando su sencillez y confiabilidad. TDD pretende guiar el desarrollo mediante la creación continua de tests automatizados y en base a los mismos construir la solución que los satisfaga, lo cual reduce los costos de implementación en un equipo de desarrollo, llegando a obtener un eficiente diseño de software con la ayuda de las pruebas unitarias.

La Escuela Politécnica del Ejército se ha visto en la necesidad de crear un sistema Web para la gestión de reservas de los Laboratorios Generales de Computación, con el fin de optimizar dicho proceso, el cual se lo venía haciendo de forma manual. El objetivo del presente proyecto es atender esta necesidad aplicando TDD en dicho sistema, para lograr que el resultado del desarrollo sea “código limpio que funcione”, es decir obtener un código de calidad sin duplicación y de correcta funcionalidad, generados a través de los resultados de los test.

El resto del artículo se ha estructurado de la siguiente manera: La sección 2 describe el marco teórico referencial en donde se analiza el aspecto metodológico empleado. La sección 3 relata el proceso de implementación de la aplicación utilizando TDD. La sección 4 evalúan los resultados obtenidos. En la sección 5 se realiza un análisis de los trabajos relacionados. Finalmente en la sección 5 se puntualizan las conclusiones derivadas del presente estudio.

2. FUNDAMENTO TEORICO

2.1. Metodología

El Desarrollo Dirigido por Test se basa en la ejecución de pruebas antes de la etapa de implementación, por lo que para su comprobación se aplicó dicha técnica en el desarrollo de un sistema para el control de la asignación de laboratorios generales de computación de la Escuela Politécnica del Ejército, el proyecto se encuentra orientado a la Web basado en la metodología ágil de desarrollo de software AUP, la cual se adapta a un modelo iterativo e incremental (entregas pequeñas de software, con ciclos rápidos), cooperativo (cliente y desarrolladores trabajan juntos constantemente con una cercana comunicación), sencillo (el método en sí mismo es fácil de aprender y modificar), y adaptable (permite realizar cambios de último momento)[8], por lo que cada iteración es una versión del sistema mejorado y en donde el usuario forma parte importante durante toda la fase de desarrollo del proyecto con la finalidad de ir puliendo errores de las pruebas generadas.

AUP está basado en disciplinas y entregables incrementales con el tiempo. El ciclo de vida en proyectos grandes es serial mientras que en los pequeños es iterativo.

Las disciplinas se desarrollan de una manera iterativa, definiendo las actividades las cuales los miembros del equipo de desarrollo construirán, validarán y entregaran el software el cual cumple las necesidades de los clientes. Las disciplinas son:

- i) Modelado
- ii) Implementación
- iii) Prueba
- iv) Despliegue
- v) Administración de la configuración
- vi) Administración de Proyecto
- vii) Entorno

AUP aplica técnicas ágiles incluyendo Desarrollo Dirigido por Pruebas (TDD), Modelado Ágil, Gestión de Cambios Ágil, y Refactorización de Base de Datos para mejorar la productividad.

2.2. Herramientas

El presente proyecto utiliza las siguientes herramientas para el desarrollo del sistema Web (ver Tabla 1).

TABLA 1 Herramientas de desarrollo y pruebas

Herramienta	Utilidad
Netbeans	Herramienta para el desarrollo del código fuente de la aplicación.
JEE	Plataforma de programación Web para el desarrollo y ejecutar software de aplicaciones en n capas.
MySQL	Sistema gestor de base de datos relacional, multihilo y multiusuario.
JUnit	JUnit es un conjunto de bibliotecas (framework), creadas por Erich Gamma y Kent Beck que son utilizadas en programación para hacer pruebas unitarias de aplicaciones Java. [1]
Jsp	JSP (Java Server Pages) es una tecnología Web, del lado del servidor, están compuestas de código HTML/XML mezclado con etiquetas esenciales para programar scripts de servidor en sintaxis Java. [2]
Jsf	JavaServer Faces (JSF) es una tecnología y framework para aplicaciones Java basadas en Web que simplifica el desarrollo de interfaces de usuario en aplicaciones Java EE. [3]
Pmd	PMD es una herramienta que comprueba que nuestra aplicación cumpla una serie de reglas que nos ayudan a obtener un código más elegante, sencillo y mantenible. [4]
SourceForge	SourceForge es una central de desarrollos de software que controla y gestiona varios proyectos de software libre y actúa como un repositorio de código fuente. [5]

2.3. Tests Unitarios

Para comprender el funcionamiento de TDD es necesario explicar las características que deben tener los tests unitarios generados durante la aplicación de esta metodología. Los Tests generados durante los ciclos de desarrollo utilizando TDD no son los tests ejecutados comúnmente durante el desarrollo clásico en el cual los desarrolladores prueban los cambios y/o funcionalidades nuevas del producto ejecutando el programa y verificando que las cosas “funcionen” según lo deseado [6], sino que la finalidad de realizar test unitarios es encontrar una solución óptima a un determinado requerimiento antes de implementar código.

TDD no implica solamente realizar pruebas, sino implica tener Tests automatizados, fácilmente repetibles y que permitan ser ejecutados con el mínimo esfuerzo posible. Estos tests automatizados no son tests del tipo de caja negra sobre el producto o sobre módulos del producto, sino tests aplicados directamente sobre las unidades mínimas de comportamiento que conforman el producto como pueden ser por ejemplo los métodos de una clase en un entorno de objetos con clases. [7]

2.4. Test Driven Development

TDD es una técnica para diseñar software que se centra en tres pilares fundamentales:

- i) La implementación de las funciones justas que el cliente necesita y no más.
- ii) La minimización del número de defectos que llegan al software en fase de producción.
- iii) La producción de software modular, altamente reutilizable y preparado para el cambio.

TDD se basa en el siguiente ciclo de desarrollo basado en el libro de *Ensayos de desarrollo impulsado por ejemplo* [9]:

- **Elegir un requerimiento:** Consiste en que el desarrollador escribe un conjunto de requisitos que crea que dará mayor conocimiento del problema y que a la vez sea fácil de implementar.

- **Escribir la prueba.** Para escribir la prueba, el desarrollador debe entender claramente las especificaciones y los requisitos. El diseño del documento deberá cubrir todos los escenarios de prueba y condición de excepciones.
- **Escribir el código haciendo que pase la prueba.** Este paso fuerza al programador a tomar la perspectiva de un cliente considerando el código a través de sus interfaces. Ésta es la parte conducida por el diseño, del TDD. Como parte de la calibración de la prueba, el código debe fallar la prueba significativamente las primeras veces.
- **Ejecutar las pruebas automatizadas.** Si pasan, el programador puede garantizar que el código resuelve los casos de prueba escritos. Si hay fallos, el código no resolvió los casos de prueba.
- **Refactorización y limpieza en el código.** Después se vuelven a efectuar los casos de prueba y se observan los resultados.
- **Repetición.** Después se repetirá el ciclo y se comenzará a agregar las funcionalidades adicionales o a arreglar cualquier error.

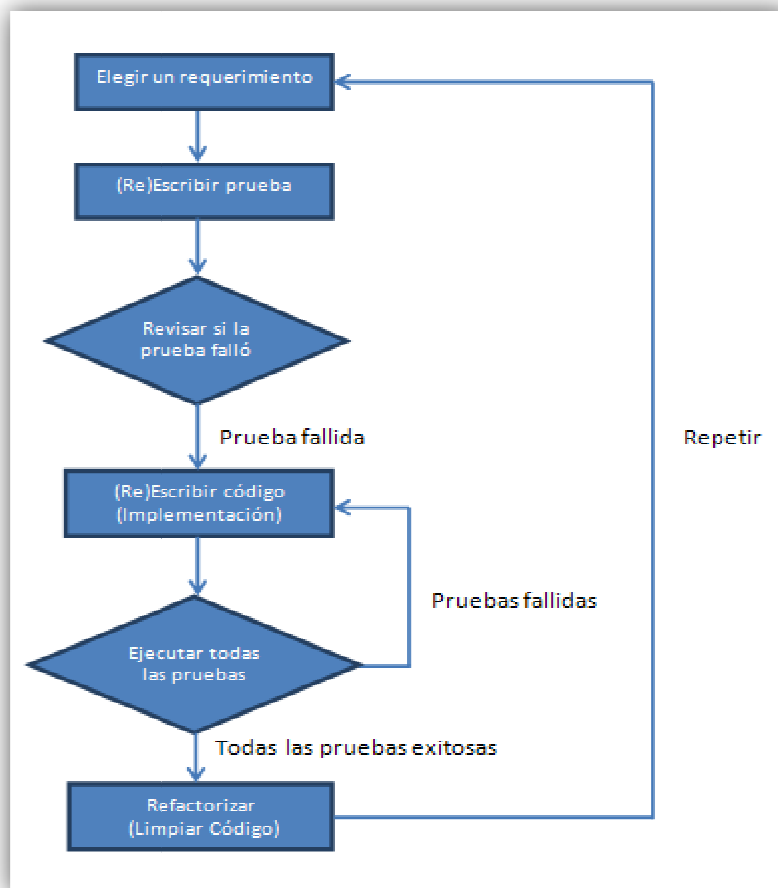


Figura 1. Ciclo de Desarrollo de TDD

2.5. Behavior Driven Development (BDD)

BDD reorienta el enfoque al comportamiento del sistema. BDD usa una plantilla para poder pensar en el comportamiento de las pruebas del código [10]: **Dado** (Given), un contexto inicial; **Cuando** (When) un evento se produce; **Entonces** (Then) asegure algunos resultados.

2.6. Acceptance Test Driven Development (ATDD)

Ayuda a coordinar los proyectos de software de forma de entregar lo que el cliente desea. Las pruebas de aceptación son especificaciones de comportamiento y funcionalidad deseados para un sistema. ATDD determina el cómo se implementará una determinada historia de usuario o caso de uso, el sistema trata determinadas condiciones y entradas. Una buena prueba de aceptación debe ser [11]:

- i) Propiedad de los clientes
- ii) Escrito en conjunto con los clientes, desarrolladores y analistas de prueba
- iii) Sobre el Qué y no sobre el Cómo
- iv) Expresada en lenguaje de dominio del problema
- v) Conciso, preciso y sin ambigüedades

3. IMPLEMENTACION DE LA APLICACIÓN UTILIZANDO TDD

Con la finalidad de aplicar el Desarrollo Dirigido por Test, se desarrolló un sistema orientado a la Web para la gestión de reservas de los laboratorios computacionales de la Escuela Politécnica del Ejército.

La herramienta es un sistema Web distribuido, desarrollado en el lenguaje de programación JAVA, basado en el patrón de diseño Modelo Vista Controlador que permite separar en componentes dicho sistema, además posee un motor de base de datos MYSQL.

Para las pruebas se utilizó la herramienta JUNIT con la finalidad de verificar, manejar y ejecutar conjuntos de pruebas automatizadas. Entre las principales características del sistema desarrollado se puede mencionar:

- i) Administración de usuarios y perfiles de acceso en el sistema.
- ii) Configuración de periodos académicos.
- iii) Disponibilidad de laboratorios en horas detalladas.
- iv) Creación de reservas.
- v) Gestión de entidades (laboratorios, materias, carreras)

ACCESO AL SISTEMA

Consejos de Seguridad
.....
NUNCA entregue sus datos personales, usuario, clave de acceso por cualquier medio. Recuerde que el usuario y las claves son secretas.

Usuario:
chinojosa

Password:
.....

Ingresar Restablecer

[Usuario no registrado](#)

Escuela Politécnica del Ejército 2011. All rights reserved
ESPE

Figura 2. Pantalla inicial del sistema SILVERLAB

A continuación se presenta la pantalla principal del sistema SILVERLAB, donde muestra las principales entidades que abarca el sistema cada una tiene submenús propios de la entidad.



Figura 3. Pantalla inicial del sistema SILVERLAB

4. EVALUACION DE RESULTADOS

Los resultados obtenidos tras el estudio de la técnica TDD, se realizaron mediante el análisis y estudio de técnicas que también se basan en la filosofía de TDD como son ATDD, BDD explicadas anteriormente. La Tabla 2 muestra una comparación entre cada una de ellas clasificadas por criterios de evaluación y prioridades cuyo comportamiento se ilustra en las Figs. 4 y 5.

CRITERIO	
1 = Más Bajo	10 = Más Alto

PESO	
1% = Más Bajo	10% = Más Alto

TABLA 2: Comparación Entre TDD – ATDD – BDD Mediante Criterio De Evaluación

Nro.	Criterio de Evaluación	TÉCNICAS DE DESARROLLO			Peso	General
		TDD	ATDD	BDD		
1	Lenguaje simple y natural	5	5	7	7%	10
2	Multilinguaje	9	7	8	5%	10
3	Consistencia	9	8	7	20%	10
4	Porcentaje de utilidad	10	8	8	10%	10
5	Facilidad al aplicar	8	9	7	7%	10
6	Ayuda y documentación	8	8	8	4%	10
7	Incremento de tiempo de desarrollo	7	6	6	7%	10
8	Porcentaje de disminución de defectos	9	8	7	20%	10
9	Cantidad Documentación en ingles	9	9	7	4%	10
10	Cantidad Documentación en español	8	7	5	4%	10
11	Conocimientos de programación	9	7	6	10%	10
12	Utilización de software libre	10	10	10	2%	10
	TOTAL	101	92	86	100	130
	TOTAL PORCENTAJE	77.69%	70.76%	66.15%	100%	100%
	TOTAL PORCENTAJE PESO	80.55	70.61%	70%	100%	

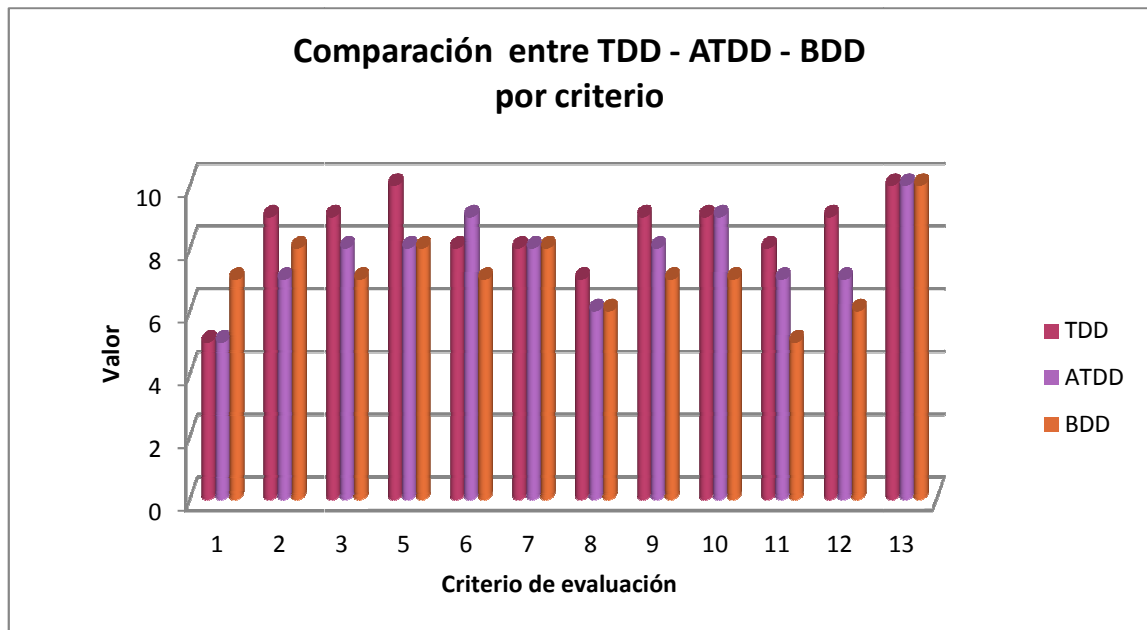


Figura 4. Comparación entre TDD - ATDD – BDD por Criterio de Evaluación

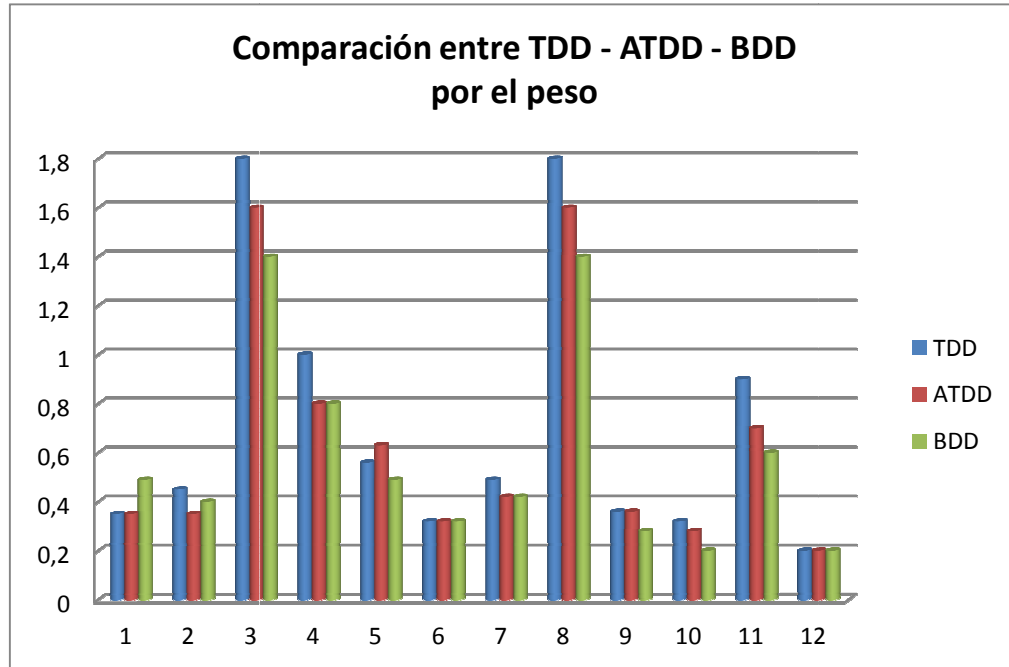


Figura 5. Comparación Entre TDD - ATDD – BDD por Peso

5. TRABAJOS RELACIONADOS

En cuanto al estudio del Desarrollo Dirigido por Test TDD, existen algunas técnicas que también han llevado la filosofía de TDD, estos son Acceptance Test Driven Development (ATDD) y Behaviour Driven Development (BDD). ATDD [11] es una evolución de TDD, orientada a resolver la falta de comunicación con el cliente y abordar adecuadamente los cambios que pueda introducir a lo largo del proceso de desarrollo. En ATDD el proceso de desarrollo está dirigido por las pruebas de aceptación, las cuales deberían ser escritas por el cliente con la ayuda de un desarrollador. Un requisito en ATDD es una Historia de Usuario, especificada con una breve descripción narrativa y a la cual se le asocia un conjunto de pruebas de aceptación, escritas normalmente en lenguaje natural como lo detalla Roger en [12].

Por otra parte, BDD surge como otra propuesta promovida por North [13], con el propósito de conectar las pruebas unitarias con los requisitos. BDD es un enfoque que empieza por identificar la funcionalidad del negocio, y después profundiza en el conjunto de requisitos que esta funcionalidad alcanzará. Ambos, ATDD y BDD, son muy similares en cuanto a que están orientados a la automatización de pruebas y generación de código (al menos parcial). En BDD y ATDD las pruebas se automatizan inmediatamente al ser identificadas, no existe separación temporal en cuanto a la especificación de la prueba y su correspondiente implementación.

En cuanto al desarrollo de sistemas utilizando la técnica TDD existen algunos proyectos relacionados entre ellos:

FITNESSE: FitNesse es un proyecto open Source (<http://fitnesse.org/FitNesse>), que permite a los clientes, testers y programadores diseñar y entender lo que sus programas deben hacer, y comparar automáticamente con lo que realmente hacen. La finalidad es comparar las expectativas de los clientes y si estas se cumplen con los resultados obtenidos. Técnicamente FitNesse tiene alrededor de 50 mil líneas de código java. El mismo se encuentra construido en base al Desarrollo Dirigido por Test (TDD). El sistema es muy robusto y se encuentra abierto para la comunidad donde se puede aportar con el desarrollo de nuevas versiones.

DIMDWARF APPLICATION SERVER: Es un proyecto open Source disponible para la comunidad en <http://dimdwarf.sourceforge.net/>. Está desarrollado por completo con TDD / BDD y tratando de escribir el código más limpio posible (sólido, limpio, de calidad, etc.). Dimdwarf es un servidor de aplicaciones distribuidas para plataforma Java. Donde se escribe un único subproceso, el código de POJO por eventos - el servidor hace que sea multi-hilo, persistente y transaccional. Los usuarios finales de la aplicación son los desarrolladores de juegos multijugador en línea.

Para aportar cambios al proyecto se deberá seguir las siguientes guías:

- i) Aplicar la metodología ágil Extreme Programming (XP): Ejecutar todas las pruebas; No contener duplicación; Desarrollar únicamente el requisito establecido; Reducir al mínimo el número de clases y métodos
- ii) Las pruebas deben estar escritas justo antes del código de producción (es decir, usar TDD)
Las pruebas deben estar organizadas como especificaciones ejecutables de comportamiento del sistema (es decir, el uso de BDD)
- iii) Sigue el buen diseño orientado a objetos, por ejemplo mediante la aplicación de los principios 'SÓLID' [14].

6. CONCLUSIONES Y TRABAJO FUTURO

Tras el estudio de la técnica del Desarrollo Dirigido por Test, se ha podido constatar que la misma no es solamente una técnica que abarca el testing de la aplicación, sino que es una técnica de diseño, ya que intenta mejorar el enfoque de desarrollo obteniendo código de calidad. Se realizó una comparación entre algunas técnicas que también han llevado la filosofía de TDD, como son Acceptance Test Driven Development (ATDD) y Behaviour Driven Development (BDD), ambas son muy similares en cuanto a que están orientados a la automatización de pruebas y generación de código, pero la diferencia radica en

que TDD busca que el programador vaya más allá que implementar código, sino que se convierta en un diseñador de software.

El acoplarse al TDD no es una tarea sencilla, requiere de un largo proceso de adaptabilidad a buenas prácticas de desarrollo ágil, en principio el uso del mismo se torna complejo necesitando de mayor esfuerzo y tiempo para poder lograr resultados eficientes. El Desarrollo Dirigido por Test (TDD) se acopla a diferentes lenguajes de programación como son Ruby, PHP, Java, Pearl, entre otros. Para la aplicación desarrollada se utilizó la plataforma de programación Java Enterprise Edition, facilitando el desarrollo de una aplicación distribuida en niveles de programación, permitiendo ejecutar funcionalidades sencillas que se acoplan al requerimiento del usuario.

La metodología AUP aplica técnicas ágiles como el Desarrollo Dirigido por Pruebas (Test Driven Development - TDD), Modelado Ágil y Gestión de Cambios Ágil, por lo que se acopó de manera significativa en el desarrollo del caso práctico, permitiendo tener una correcta distribución de las actividades de trabajo y a la vez centrarse en actividades de alto valor, logrando una aplicación distribuida y sencilla.

Adicionalmente se utilizó la arquitectura MVC, logrando una clara separación entre interfaz, lógica de negocio y de presentación, facilitando la realización de pruebas unitarias de sus clases, obteniendo un código flexible para cambios.

La aplicación desarrollada fue publicada en el repositorio Web sourceforge.net, el cual se encuentra a disposición de la comunidad con la finalidad de aportar, comentar, y continuar con el desarrollo de funcionalidades sobre la aplicación. Cabe mencionar que cuando existen proyectos realizados sin TDD, se puede realizar test de regresión que consiste en la aplicación de test en el front-end (primera capa), con la finalidad de encontrar divergencias funcionales respecto al comportamiento esperado del software.

Como trabajo futuro se pretende aplicar ATDD al sistema desarrollado con la finalidad de garantizar un óptimo funcionamiento que abarque comportamientos y test de aceptación.

Referencias Bibliográficas

- [1] Anónimo. (s.f.). JUnit.org. Recuperado el 23 de Marzo de 2011, de JUnit.org: <http://www.junit.org/>
- [2] Alvarez, M. A. (08 de Julio de 2002). desarrolloWeb.com Que es JSP. Recuperado el 02 de Abril de 2011, de desarrolloWeb.com Que es JSP: <http://www.desarrolloWeb.com/articulos/831.php>
- [3] Anónimo. (18 de Mayo de 2006). Wikipedia JavaServer Faces . Recuperado el 07 de Abril de 2011, de Wikipedia JavaServer Faces : http://es.wikipedia.org/wiki/JavaServer_Faces
- [4] <http://mundogeek.net/archivos/2009/03/08/mejora-tu-codigo-java-con-pmd/>
- [5] Anónimo. (2005). Wikipedia SourceForge . Recuperado el 20 de Junio de 2011, de Wikipedia SourceForge : <http://es.wikipedia.org/wiki/SourceForge>
- [6] Test-driven development: by example Escrito por Kent Beck Addison Wesley, 2003
- [7] Diseño ágil con TDD (Carlos Blé Jurado)
- [8] Abrahamsson, P., Salo, O., Ronkainen, J., Agile Software Development Methods. Review and Analysis, VTT, 2002.
- [9] Beck, K. (2003). Test Driven Development: by Example.
- [10] Gómez, D. (26 de Marzo de 2009). www.dosideas.com. Obtenido de www.dosideas.com: <http://www.dosideas.com/noticias/metodologias/484-calidad-de-software-con-bdd-y-atdd.html>
- [11] Beck, K. Test Driven Development: By Example. Addison Wesley, 2003
- [12] Roger, S. Acceptance testing vs. unit testing: A developer's perspective. Lecture Notes in Computer Science, 2004 – Springer
- [13] North, D. Introducing BDD, Better Software, March, 2006.
- [14] Esko Luontola , Dimdwarf Application Server <http://dimdwarf.sourceforge.net/>

Prototipo de la Tecnología IPTV mediante un Servidor de Contenidos Multimedia en Linux

Juan Zarria Santillán, Carlos Cevallos López, Darwin Aguilar y Diego Marcillo

Departamento de Ciencias de la computación, Escuela Politécnica del Ejército, Sangolquí, Ecuador, juan.zarria@adc-has.com, charlie_mclaren@msn.com, darwin.aguilar@espe.edu.ec, dmmarcillo@espe.edu.ec

RESUMEN: La comunicación a través de dispositivos IP ha evolucionado notablemente en los últimos años, llevando a la industria a ampliar la gama de productos IP disponibles. El presente trabajo se enfoca en la transmisión de televisión por cable o por aire que actualmente se encuentra en constante evolución para ofrecer más servicios a los televidentes, como la posibilidad de retroceder, avanzar, pausar y grabar lo que están viendo. Esto también se lo puede realizar a través de una red de área local, es decir, a través de la tecnología de IPTV, para lo cual se requiere un servidor de contenidos multimedia que transmita la señal a través del cable de red, y dispositivos llamados set top box que reciban dicha señal, la decodifiquen y la envíen a través de un cable RCA o HDMI hacia un televisor común y corriente. Este proceso permite seleccionar al administrador del servicio o del servidor los contenidos multimedia que serán transmitidos hacia los hosts, y además permite que el usuario final seleccione la película, documental, video musical, o cualquier contenido disponible en el mismo. Finalmente, el administrador del servidor puede transmitir canales de televisión y seleccionar la programación de los mismos, es decir una cola de videos que serán transmitidos de manera continua y podrán ser receptados por cualquier set top box conectado en la red.

Palabras Clave: IPTV, servidor, multimedia, transmisión.

ABSTRACT: Communication through IP devices has evolved significantly in recent years, leading the industry to expand the range of IP products available. This present work focusing in the transmission of cable or air television which is constantly developing to offer more services for TV viewers, for example, the possibility to rewind, fast forward, pause and record what they see. This also can be made through a local area network with IPTV technology, which is required for a media server that transmits the signal through an Ethernet cable and devices called set top boxes that receive, decode, and send the signal through an RCA cable or HDMI to an ordinary television. This process allows the service administrator to select the multimedia content that will be transmitted to the hosts, and allows the end user to select the movie, documentary, music video, or any content available on it. In addition, the server administrator can transmit TV channels and select its programming, it means a queue of videos to be transmitted continuously and can be picked up by any set top box connected to the network. Each type of transmission is performed in different protocols, it means that the transmission of multimedia content that can be selected, moved forward, rewind and paused by the end user are transmitted by RTSP or Real Time Streaming Protocol, while the transmission channels programmed into the server are transmitted over UDP or RTP, the latter being a protocol that ensures data reception, which is not needed in video transmissions because they cause an increase in the consumption of server processor resources but getting a result very similar to using UDP.

Keywords: IPTV, server, multimedia, transmission.

1. INTRODUCCIÓN

IPTV es un sistema de distribución por suscripción de señales de televisión o video usando conexiones de red que trabaja sobre el protocolo IP [1]. Esta nueva tecnología ha ido desarrollándose en base al video-streaming, lo cual quiere decir que es la capacidad de poder transmitir videos o cualquier variedad de contenidos multimedia en tiempo real a través de la red de datos. IPTV consigue aportar ventajas, apoyándose sobre una infraestructura tecnológica dinámica que se caracteriza, entre otros factores, por su grado de automatización, una rápida transmisión de los contenidos multimedia y una elevada capacidad de adaptación para atender a una demanda variable [2].

La tecnología IPTV [3] es un concepto de TI que permite ofrecer y consumir recursos a través de un servidor de contenidos e internet, se tiene la posibilidad de trabajar con ellos desde cualquier lugar donde exista conectividad a redes.

El presente trabajo se enfoca en la transmisión de televisión por cable o por aire que actualmente se encuentra en constante evolución, para ofrecer más servicios a los televidentes, como la posibilidad de retroceder, avanzar, pausar y grabar lo que están viendo. Esto también se lo puede realizar a través de una red de área local, es decir, a través de la tecnología de IPTV, para lo cual se requiere un servidor de contenidos multimedia que transmita la señal a través del cable de red, y dispositivos llamados set top box que reciban dicha señal, la decodifiquen y la envíen a través de un cable RCA o HDMI hacia un televisor común y corriente.

El resto del artículo ha sido organizado de la siguiente manera: La sección 2 describe los materiales y métodos utilizados. La Sección 3 explica la evaluación de resultados con una leve discusión. La sección 4 precisa las conclusiones obtenidas y el trabajo futuro.

2. MATERIALES Y METODOS

Para el desarrollo del presente trabajo se utilizó distintas herramientas que se describen a continuación en la Tabla 1:

TABLA 1 “Herramientas Utilizadas”

Herramienta	Utilidad
Ubuntu Studio	Sistema Operativo basado en Linux que contiene los códec para transmisión de datos
Darwing Streaming Server	Servidor de contenido multimedia desarrollado por Apple.
Live 555	Servidor de contenidos multimedia.
VLC	Reproductor de audio y video.
Quick Time	Reproductor de audio y video.
Traffic Grapher	Monitoreo de red mediante gráficas y estadísticas.

El desarrollo consistió en emitir cuatro videos con los formatos de video MPEG4 y MPEG2 hacia 3 host.

2.1 Marco de Trabajo

La Figura 1 muestra el diseño de la arquitectura IPTV [2]. Para la implementación del prototipo se utilizó un servidor de contenidos basado en Linux [3] y contenidos multimedia almacenados en el mismo que se emiten hacia un televisor.

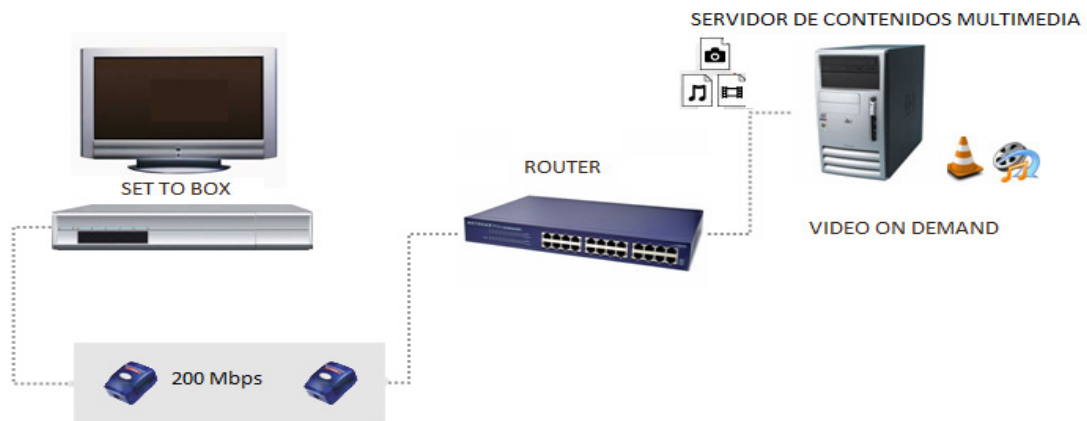


Figura 1. Arquitectura IPTV

Las características de los videos a utilizar se muestran a continuación en la TABLA 2:

TABLA 2 “Características de los videos”

Video	Duración	Bit rate (kbps)	Formato	Tamaño	Frame rate (fps)	Formato Audio
BigBandTheory	0:22:25	1519	Mpeg-4	624x325	25	Mpeg4 aac
BigBandTheory	0:22:25	2692	Mpeg-2	624 x	25	Mp2
Derrapes	0:06:58	2184	Mpeg-4	640 x 480	25	Mpeg4 aac
Derrapes	0:06:58	1311	Mpeg-2	640 x 480	25	Mp2
Carrera carros	0:22:24	2334	Mpeg-4	641 x 480	30	Mpeg4 aac
Carrera carros	0:22:24	2520	Mpeg-2	642 x 480	30	Mp2
Autos	0:05:06	3421	Mpeg-4	643 x 480	30	Mpeg4 aac
Autos	0:05:06	2780	Mpeg-2	644 x 480	30	Mp2

2.2 Identificación del Hardware.

- ✓ **Servidor de contenidos:**
Sistema operativo: Linux Ubuntu Studio 10.4.
Procesador: Intel® Core™ i5 de 3.20 Ghz de 4 hilos y 2 núcleos.
Memoria RAM: 4 GB.
- ✓ **Pc cliente:**
Sistema operativo: Windows xp professional.
Procesador: Intel Core 2 Duo de 2.13 ghz.
Memoria RAM: 2 GB.
- ✓ **Laptop cliente:**
Sistema operativo: Windows 7 professional.
Procesador: Intel Core 2 Duo de 2.0 ghz.
Memoria RAM: 4 GB.
- ✓ **Laptop cliente:**

Sistema operativo: Windows xp profesional.

Procesador: Intel Pentium 4 de 1.8 ghz.

Memoria RAM: 1,5 GB.

2.3 Transmisión Unicast, Broadcast y Multicast

La Figura 2 muestra cómo se debe emitir el video desde el servidor seleccionando el video. Muestra además cómo se envía la emisión trans-codificada a una red a través de UDP. El tipo de trans-codificador, la dirección Unicast, Broadcast o Multicast y el puerto a la que se enviará, se lo realiza con el programa VLC. Para otros protocolos como RTSP, RTP se lo realiza de igual manera.

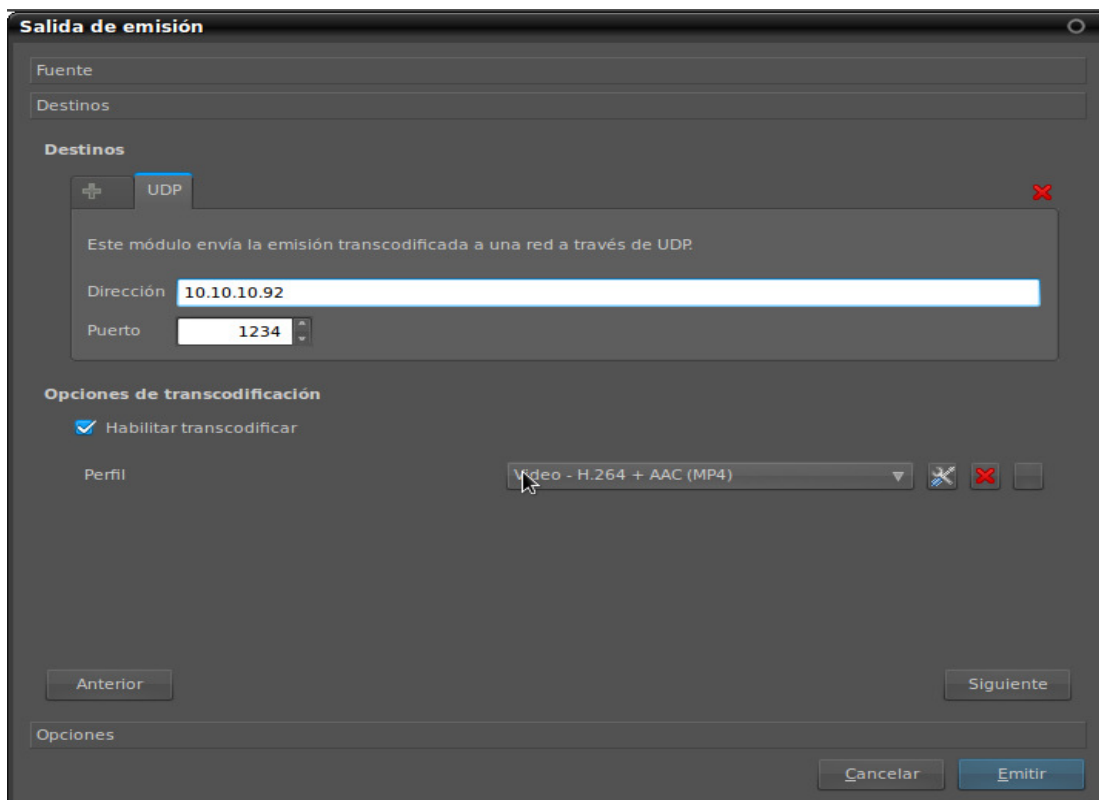


Figura 2. Transmisión Unicast, Broadcast o Multicast

2.4 Configuración en Set Top Box para recibir transmisiones.

En la Figura 3 se puede apreciar las configuraciones de las imágenes correspondientes a cada video, como por ejemplo: "sport.jpg" que es el vínculo a la foto que lo relaciona al contenido multimedia, además se puede ubicar un pequeño título que haga referencia como información previa. La parte más importante es cuando se define el protocolo RTSP "rtsp://192.168.1.3/video2.mp4" con la dirección IP del servidor multimedia y el recurso compartido que es video2.mp4, cabe mencionarse que esto se realiza cambiando el código fuente del set top box.

```
vod.js
var menu,player;

function VodMenu(){
  /*The format is ["Server Name","Clip Image","Clip Description","Clip URL"]*/
  this.rtspPgm = [
    ["", "sport.jpg", "Carrera derrapes Video Sin VOD",
     "rtsp://192.168.1.3/video2.mp4"],
    ["", "bigbang.jpg", "Big Bang Theory Video Sin VOD",
     "rtsp://192.168.1.3/video.mp4"],
  ]
}
```

Figura 3. Configuración del Set top box

3. EVALUACIÓN DE RESULTADOS Y DISCUSIÓN

3.1 Rendimiento del Servidor de contenidos multimedia con transmisión UDP

En la Figura 4 se observan en el lado del servidor la cantidad de 4 transmisiones [5] de video que consumen entre el 90% y 100% de procesador; y la memoria entre el 20 y 30% no viéndose afectada mayormente, mientras más videos se emitan con UDP la calidad de transmisión se verá afectada.



Figura 4. Rendimiento servidor con UDP

3.2 Análisis de paquetes en transmisión UDP.

En la Figura 5 se observan los distintos picos que se van alcanzando en la emisión de paquetes que contiene cada uno de los videos. Como se puede ver, en cada una de las emisiones se alcanza picos máximos; por ejemplo, cuando se empieza a emitir los videos hacia los host sube considerablemente la taza de transferencia a 760/ Kbps. Luego a medida que los videos van terminando bajan los picos; por ejemplo, a 300 Kbps, si se los vuelve a transmitir va sufriendo cambios en los picos de transmisión dependiendo de la cantidad de videos.

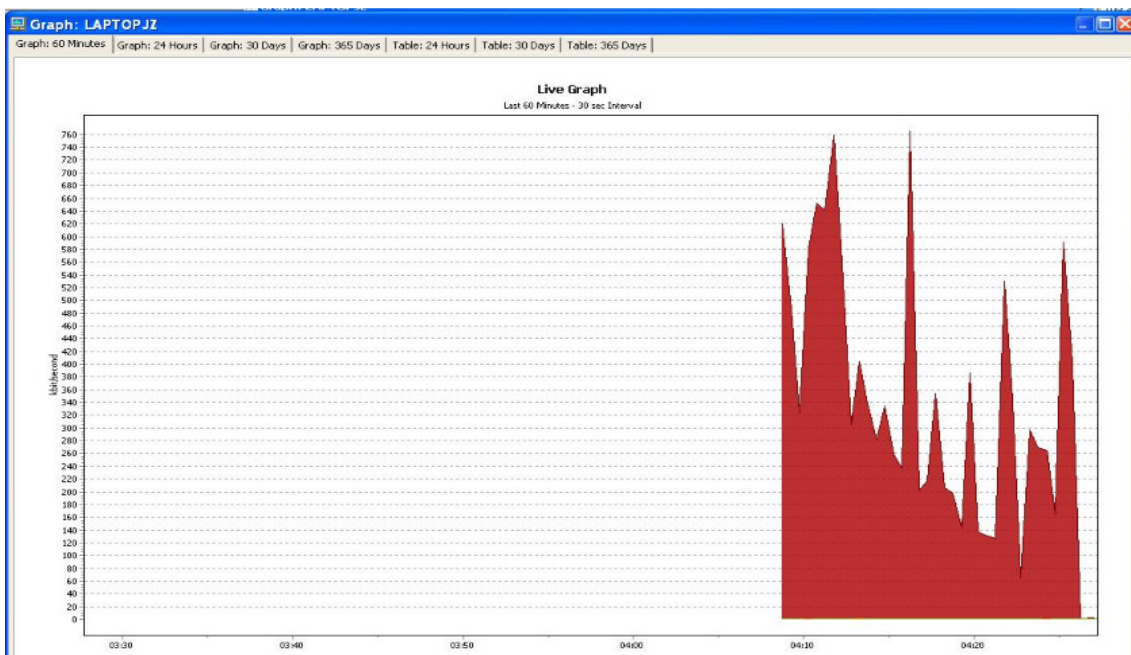


Figura 5. Transmisión en UDP

3.3 Rendimiento del Servidor de contenidos Multimedia con Transmisión RTP

En la Figura 6 se observan en el lado del servidor la cantidad de 4 transmisiones de video que llegan consumir entre el 90% y 100% de procesador; y la memoria entre el 20 y 30% no viéndose afectada mayormente, mientras más videos se emitan con RTP la calidad de transmisión se verá afectada, es muy similar que las transmisiones UDP.



Figura 6. Rendimiento servidor con RTP

3.4 Análisis de paquetes en transmisión RTP.

En la Figura 7 se observan los distintos picos que se van alcanzando en la emisión de paquetes que contiene cada uno de los videos. Se puede ver que en cada una de las emisiones alcanza picos máximos; por ejemplo, cuando se empieza a emitir los videos hacia los host sube considerablemente la taza de transferencia a 760/ Kbps; luego a medida que los videos van terminando bajan los picos; por ejemplo, a 300 Kbps. Si se los vuelve a transmitir va sufriendo cambios en los picos de transmisión dependiendo de la cantidad de videos, se aprecia claramente que se asemeja a UDP.

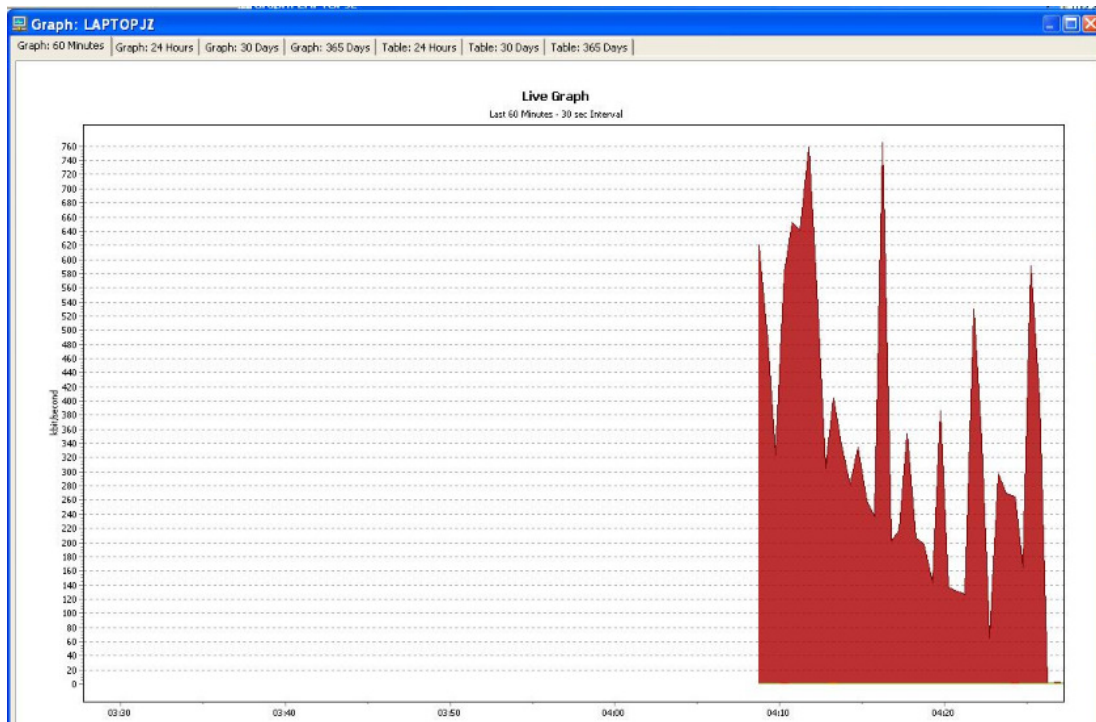


Figura 7. Transmisión en RTP

3.5 Rendimiento de servidor de contenidos multimedia con transmisión RTSP.

En la Figura 8 se observan en el lado del servidor la cantidad de 4 transmisiones de video que llegan a consumir entre el 0% y 25% en utilización del procesador y la memoria constante en 20%. Se pueden emitir varios videos a la vez y la calidad de transmisión es perfecta, quedando evidenciado que la mejor manera de transmisión es RTSP.



Figura 8. Rendimiento servidor con RTSP

3.6 Análisis de paquetes en transmisión RTSP.

En la Figura 9 se observan los distintos picos que se van alcanzando en la emisión de paquetes que contiene cada uno de los videos. Se puede ver que en cada una de las emisiones alcanza picos máximos por ejemplo cuando se empieza a emitir los videos hacia los host sube considerablemente la tasa de transferencia a 7400 Kbps. Luego a medida que los videos van terminando bajan los picos a 2000 Kbps. Si se los vuelve a transmitir va sufriendo cambios en los picos de transmisión dependiendo de la cantidad de videos.

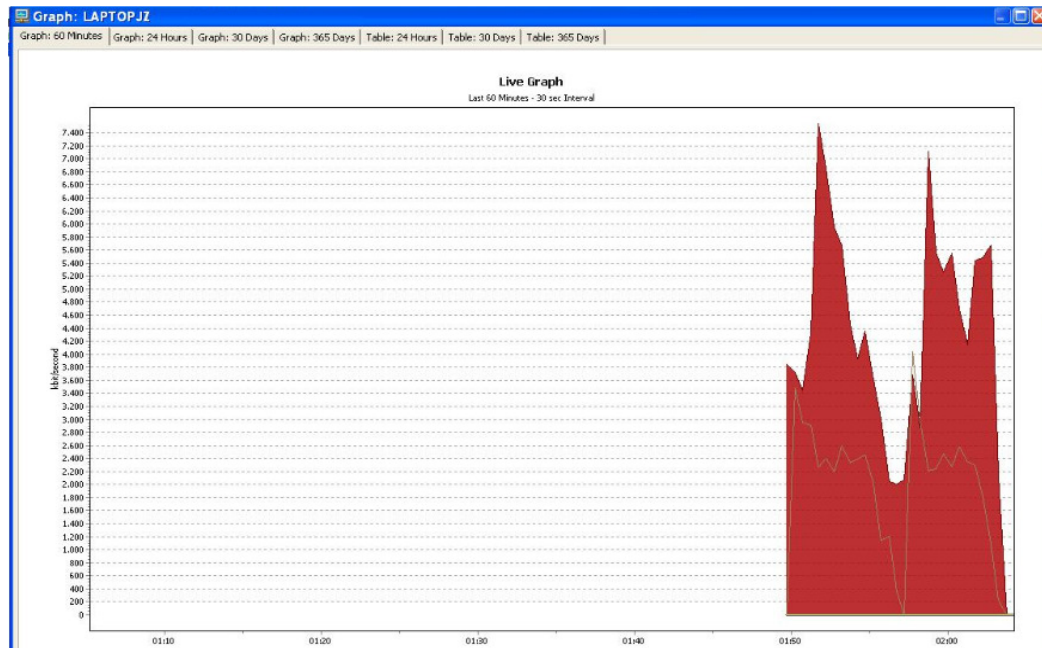


Figura 9. Transmisión en RTSP

4. TRABAJOS RELACIONADOS

A la fecha no existen trabajos relacionados debido a que la tecnología IPTV es realmente nueva, existen libros relacionados pero no existe en sí una implementación completa de servidores de contenidos multimedia que emitan contenidos a través de una red de área local.

5. CONCLUSIONES Y TRABAJO FUTURO

La principal razón por la que llega a colapsar el servidor de IPTV es debido a la saturación del procesador, sin importar la cantidad de usuarios finales, el servidor colapsa debido a la cantidad de canales UDP transmitidos simultáneamente, además que el uso de la memoria RAM del servidor incrementa levemente al ir añadiendo la cantidad de canales transmitidos. Al transmitir canales en alta definición, el servidor soporta como máximo la transmisión de 3 canales en protocolo UDP con buena calidad y 3 canales en protocolo RTP entrecortados debido a que este protocolo garantiza la recepción de los datos, con lo que se consume una mayor cantidad de recursos del procesador del servidor o 2 en RTP con buena calidad como máximo simultáneamente. Al analizar el tráfico de red se puede concluir que transmitir en RTSP es lo ideal ya que se pueden enviar varios videos a la vez y el Jitter siempre se mantiene en cero, al contrario que UDP y RTP que llegan a 0,66 ms. Por otro lado, el número de paquetes perdidos en transmisiones RTSP llegan a cero, mientras tanto que en UDP y RTP llegan entre 90% y 100% de paquetes, lo que evidencia que transmitir en estos formatos no es lo más adecuado. UDP Y RTP consumen un aproximado del 20% más en recursos de red que RTSP y las transmisiones con el protocolo RTSP no afectan el rendimiento del servidor en cuanto a procesador y memoria RAM, al incrementar la cantidad de hosts que reciben 4 contenidos multimedia con excelente calidad simultáneamente.

REFERENCIAS BIBLIOGRÁFICAS

- [1] López Rubio, Gustavo, "Comunicaciones de Datos (Vol.2). Protocolos De Red, Transporte E Internet
- [2] Wright, S. Jones y Sub Lee. "IPTV Systems, Standards and Architectures", IEEE Communications, (2008).
- [3] Joel Barrios Dueñas, "Implementación de Servidores con GNU/Linux", Edición Septiembre 2009.
- [4] Held, Hilbert, "Understanding IPTV". Auerbach Publications, 2007.
- [5] Transmisión de Datos y Redes de Comunicaciones, 2da Edición (Mcgraw-Hill).

Desarrollo de una aplicación interactiva aplicando la metodología OpenUP/Basic como parte del Proyecto ESPE-GINGA

Ángel Quingaluisa, Jonathan Torres, Danilo Martínez y Santiago Salvador

Departamento de Ciencias de la computación, Escuela Politécnica del Ejército, Sangolquí, Ecuador, angel_quingaluisa@hotmail.com, idiiter@hotmail.com, danilo.martinez@espe.edu.ec, mssalvador@espe.edu.ec

RESUMEN: El proyecto ESPE-GINGA busca como objetivo difundir y motivar el desarrollo de contenidos interactivos para la TV digital en Latinoamérica, como parte de la Red Latinoamérica de Cooperación en Investigación, Desarrollo y Formación en el área de Software para TV Digital Interactiva. En este contexto, el Departamento de Ciencias de la Computación de la Escuela Politécnica del Ejército para apoyar al proyecto de implementación del Laboratorio de TV Digital Interactiva realizado por el Departamento de Eléctrica y Electrónica, decide iniciar la investigación del middleware Ginga-J. El presente trabajo propone la investigación y creación de una aplicación interactiva con el estándar Brasileño de TV Digital. Para llevar a cabo el proyecto se realizó la investigación de la arquitectura y funcionamiento del Middleware Ginga-J, además se utilizó el emulador OpenGinga, la arquitectura MVC, metodología OpenUP / Basic, lo que ha permitido obtener como resultado una aplicación interactiva digital cuyo contenido se alimenta de imágenes y archivos XML, demostrando la potencia que tiene el API Ginga-J.

Palabras claves: Middleware, Ginga J, OpenGinga, OpenUP / Basic, API.

ABSTRACT: The project ESPE-GINGA, as part of the Latin America Network of Cooperation on Research and Development of Software for digital TV, tries to introduce and spread the use of new interactive modules for digital TV in Latin America. The Computer Science Department of the Escuela Politécnica del Ejército launch with the research of the Ginga-J middleware to support the project and set up of the Digital TV laboratory managed by the Electric & Electronics' Department, decide to start the research of the middleware Ginga-J. This project creates a digital interactive application using the Brazilian model of digital TV. It was necessary a deep study of the operation of middleware Ginga-J architecture besides, the use of Open Ginga emulator, MVC architecture and OpenUP/Basic. The final product is an interactive digital application that is feed by images and XML files which shows the power of API Ginga-J.

Keywords: Middleware, Ginga J, Open Ginga, OpenUP / Basic, API.

1. INTRODUCCIÓN

En el Ecuador, se resolvió adoptar el estándar Japonés-Brasileño ISDB-T/SBTD como sistema de Televisión Digital Terrestre (TDT). Se oficializó el 24 de marzo del 2010 por parte de la Superintendencia de Telecomunicaciones (SUPTTEL). La interactividad es la parte fundamental del sistema TDT, permitiendo al usuario interactuar con el emisor a través del control remoto, estableciendo la comunicación entre ellos.

El Departamento de Ciencias de la Computación en conjunto con el Departamento de Eléctrica y Electrónica de la Escuela Politécnica del Ejército (ESPE) han decidido impulsar la investigación y desarrollo de aplicaciones interactivas para televisión digital utilizando el middleware GINGA y el API GINGA-J como parte de un proyecto de la Carrera de Ingeniería en Sistemas de la ESPE.

El proyecto ESPE–GINGA, actualmente está orientado a la investigación de aplicaciones interactivas desarrolladas con el middleware Ginga-NCL, pero todavía se desconoce el funcionamiento del recién creado Ginga-J. Por tal motivo se desarrolla una aplicación de tipo direccional bajo el Middleware Ginga-J, las pruebas se realizaron en la máquina virtual OpenGinga.

El presente trabajo propone la investigación y creación de una aplicación interactiva con el estándar Brasileño de TV Digital. Para llevar a cabo el proyecto se realizó la investigación de la arquitectura y funcionamiento del Middleware Ginga-J, además se utilizó el emulador OpenGinga, la arquitectura MVC, metodología OpenUP / Basic, lo que ha permitido obtener como resultado una aplicación interactiva digital cuyo contenido se alimenta de imágenes y archivos XML, demostrando la potencia que tiene el API Ginga-J.

Para el estudio, investigación y desarrollo de una aplicación con el Middleware Ginga-J se utilizó las siguientes metodologías: “Investigación Aplicada” y “OpenUp/Basic”, las cuales permitieron aprovechar los conocimientos ya desarrollados para aplicar en este proyecto de investigación y en la definición de roles, iteraciones y actividades que se dio a cada uno de los integrantes del proyecto para el cumplimiento de las fases de desarrollo.

El resto del artículo ha sido organizado como sigue: La sección 2 describe los fundamentos teóricos que sustentan esta investigación. La sección 3 explica los materiales. La sección 4 detalla el aspecto metodológico. En la sección 5 se evalúan los resultados. Finalmente en la sección 6 se exponen las conclusiones y trabajo futuro.

2. FUNDAMENTOS TEÓRICOS.

Ginga-J es un middleware desarrollado para el estándar Brasileño de TV digital el cual fue acogido por Ecuador. Ginga-J es una de las principales herramientas de desarrollo para aplicaciones interactivas que utilizan programación Java, el mismo que por su arquitectura logra una compatibilidad total sobre las mismas debido a sus APIs. Al momento el desarrollo de aplicaciones en Ginga-J es muy limitado ya que la versión de la máquina virtual de Java es muy básica

2.1 Arquitectura Ginga-J.

Las aplicaciones nativas se pueden ejecutar sin necesidad de contar con las funcionalidades del sistema operativo Ginga o por la aplicación Ginga. Sin embargo también se pueden utilizar los APIS estándares de Ginga-J. Las aplicaciones Xlets deben utilizar los Apis de Ginga-J, como se muestra en la Figura 1. Con el fin de mantener la compatibilidad con el API de GEM [1], Ginga-J se basa en tres grandes grupos de APIS:

- i) API Verde (APIs compatibles con GEM), aquí se encuentran incluyendo las APIs provenientes de los paquetes de Sun JavaTV, DAVIC [DAVIC, 1999] e HAVI [HAVI, 2001].
- ii) API Amarillo (compuesto por el JMF 2.1 API), lo cual es necesario para el desarrollo de aplicaciones, con captura de sonido)

iii) API azul permite al receptor de TV Digital comunicarse con cualquier dispositivo con una interfaz compatible (con conexión con cable, como Ethernet o PLC, de red o inalámbrica, como infrarrojos o Bluetooth), aquí se encuentra el API que permite el desarrollo de las aplicaciones Ginga-J que tengan Ginga-NCL (API puente); como se muestra en la Figura 2.

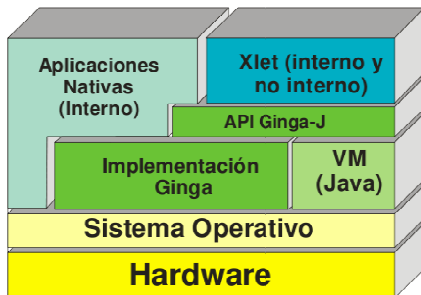


Figura 1. Arquitectura General Ginga-J

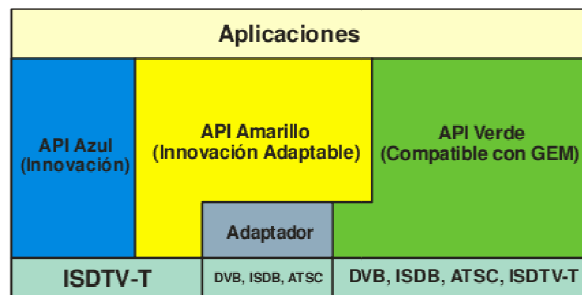


Figura 2. API's GINGA JAVA

2.2 Ginga-Ncl.

Ginga-NCL es el subsistema del Middelware Ginga el cual permite procesar documentos NCL. Un componente clave de Ginga-NCL es el motor decodificador de contenidos declarativos. La especificación de este subsistema se base en las normas ABNT NBR 15606-2 y ABNT NBR 15606-5.

2.3 Ginga-J.

Ginga-J es el subsistema del Middelware Ginga que proporciona una infraestructura para la ejecución de aplicaciones de TV Digital, el ambiente interactivo Ginga-J ofrece soporte a las aplicaciones desarrolladas con el lenguaje de programación Java. La especificación de este subsistema se basa en la norma ABNT NBR 15606-4[2], que mantiene compatibilidad con la mayoría de *middlewares* de TV Digital actuales. [3]

2.4 OpenGinga.

OpenGinga es una máquina virtual creada con Virtual Box y se basa en el Sistema Operativo Ubuntu, con una interfaz gráfica que nos permite ejecutar aplicaciones Ginga-J y GingaNCL en ambiente de PC.

3. MATERIALES.

Para demostrar las diferentes ventajas del Middelware Ginga-J se hizo uso de las siguientes herramientas para el desarrollo de una aplicación interactiva de Televisión Digital Ver Tabla 1.

TABLA 1. Herramientas de desarrollo.

Herramienta	Utilidad
Eclipse	Herramienta para el desarrollo del código fuente de la aplicación.
OpenGinga	Máquina Virtual utilizada para desarrollar y ejecutar aplicaciones de TV Digital.
XML	Archivos de gestión de datos.
GINGA-J	Estas librerías se encuentran divididas de la siguiente manera: <ul style="list-style-type: none"> ✓ El núcleo también llamada API's verde, estas son responsables de mantener el sistema compatible lo máximo posible. ✓ El módulo responsable por el soporte son las llamadas APIs roja y amarilla, las API's amarillas pueden ser exportadas en otros sistemas, soporte a múltiples usuarios, dispositivos y redes, mientras que las API's rojas son el soporte a las necesidades específicas de aplicaciones para Brasil, en especial aplicaciones de inclusión social.

4. IMPLEMENTACION DE LA APLICACIÓN INTERACTIVA UTILIZANDO API GINGA-J.

Se ha desarrollado una aplicación interactiva de TV Digital utilizando un parser XML como canal de retorno, para la visualización de contenidos de los diferentes departamentos de la Escuela Politécnica del Ejército, a través de la navegación de imágenes que se encuentran definidas mediante el archivo de propiedades, basándonos en el paradigma de programación orientada objetos con la estructura Xlet.

Para el desarrollo de esta aplicación se utilizó eventos propios del API Ginga-J, las cuales en la pantalla de interactividad permite realizar una navegación de los diferentes departamentos. Esta aplicación usa como canal de retorno un parser XML mediante el consumo del componente *kxml2-min-2.3.0.jar*; donde el telespectador mediante el teclado del computador simula un control remoto; para ingresar a la pantalla de interactividad se pulsa la tecla M, como se muestra en la Figura 3.



Figura 3. Pantalla inicial de la aplicación interactiva.

A manera de ejemplo, en este aplicativo cuando el telespectador se desplace con las teclas de navegación por los diferentes “Departamentos”, se visualizará los contenidos que tiene almacenado el archivo XML con la utilización del parser. Como resultado se obtendrá la información de cada uno de los diferentes Departamentos, como se muestra en la Figura 4.



Figura 4. Resultado de la búsqueda de la Ingeniería de Software.

5. EVALUACION DE RESULTADOS.

Ginga-J es un middleware que al estar en desarrollo es un Api muy potente el cual nos permitió el desarrollo de nuestra aplicación mediante la construcción de la misma utilizando lenguaje de programación JAVA.

Para el desarrollo en Ginga-J se puede utilizar cualquier IDE de programación compatible con Java, la única regla es que las librerías de Ginga-J y GingaCore deben estar definidas de forma correcta para no tener problemas al momento de la construcción y la compilación.

Para realizar las pruebas de la aplicación se debe tomar la decisión de que emulador se utilizara, para ello se elaboro una matriz de comparación entre los dos principales emuladores de Ginga-J: XletView y Ginga-J.

La Tabla 2 se muestra los puntajes y la comparación que se realizó tomando en cuenta algunos factores importantes como son la portabilidad, la interacción con el usuario y la capacidad de entender y utilizar el mismo.

TABLA 2. Matriz de Comparación XletView vs OpenGinga.

Medidores:				
Se medirá con puntajes de 0 al 10 siendo el ultimo la máxima puntuación				
		Prioridad:	Alta	
			Normal	
			Baja	
No	FACTOR	EMULADORES		
		XletView	OpenGinga	General
1	Fácil de Instalar	4	1	10
2	Multiplataforma(Linux, Windows, MacOS)	3	3	10
3	Software de código abierto	10	10	10
4	Cantidad Documentación en ingles	4	2.5	10
5	Cantidad Documentación en español	2	0.5	10
6	Cantidad Documentación en portugués	4	7	10
7	Estructuración de los path del emulador	1	9	10
8	Facilidad de interacción y comprensión por el usuario	6	8	10
9	Facilidad de instalación de las aplicaciones	8	9	10
10	Facilidad de comprensión para el usuario	8	8	10
11	Facilidad de portabilidad (Código, Herramientas, Emulador).	3	10	10
12	Facilidad de compilación de código.	9	9	10
13	Facilidad de utilización de aplicación de terceros (terminal, eclipse).	10	10	10
14	Compatibilidad con JAVA TV.	10	10	10
15	Utiliza Herramientas libres para su uso, S.O, máquinas virtuales, etc.	5	10	10
16	TOTAL SUMA:	87	107	150
17	TOTAL PORCENTAJE:	58.00%	71.33%	100.00%
18	TOTAL PRIORIDAD ALTA:	31	38	60
19	TOTAL PRIORIDAD NORMAL:	17	20	30
20	TOTAL PRIORIDAD BAJA:	39	49	60
20	TOTAL PRIORIDAD:	87	107	150

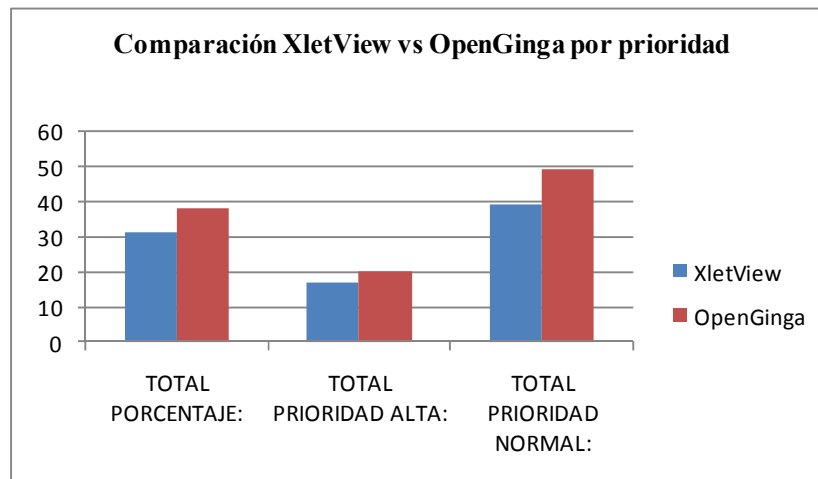


Figura 5. Comparación XletView vs OpenGinga por prioridad

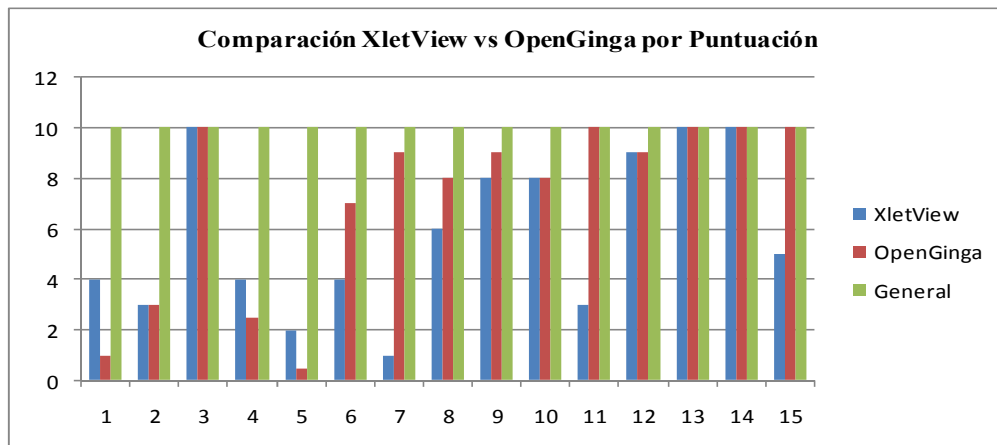


Figura 6. Comparación XletView vs OpenGinga por puntuación

La comparación entre las dos herramientas nos lleva a la conclusión que las dos tienen una funcionalidad semejante, que es la facilidad de interacción con el usuario.

El punto más importante por el cual se elige el emulador OpenGinga es su portabilidad ya que al ser una máquina virtual, posee todas las herramientas necesarias para el desarrollo compilación y ejecución de las aplicaciones interactivas construidas con Ginga-J.

Algo importante sobre la utilización del emulador es que no se necesita comprar licencias para utilizarlo ya que al momento todas las herramientas necesarias para su ejecución se distribuyen de manera libre.

6. CONCLUSIONES Y TRABAJO FUTURO

La metodología OpenUp / Basic permite el desarrollo de aplicaciones de corto alcance con un grupo pequeño de personas ya que las mismas pueden desempeñar varios roles, así mismo con el uso de las iteraciones en cada una de las fases de desarrollo se puede realizar correcciones preventivas de cada componente a desarrollar. La versión actual del emulador OpenGinga tiene limitaciones en la librería API HAVI, por tal razón los objetos gráficos con los que se puede trabajar en el desarrollo de aplicaciones para tv digital aun son limitados. El desarrollo de una aplicación en Ginga-J conlleva entender la estructura principal de un Xlet, el cual almacena los componentes gráficos, los mismos que para ser programados, el desarrollador debe tener un conocimiento amplio sobre programación orientada a objetos empleando el

lenguaje de programación JAVA. Los principales problemas que detectamos dentro del desarrollo de este tipo de aplicaciones son 2 que consideramos son de alta importancia, el primero es la restricción en el uso de memoria por lo cual la ejecución de las aplicaciones es limitada y las aplicaciones pueden colapsar; la segunda es la versión de la máquina virtual de java que utiliza el emulador aun se encuentra en una versión básica de tal manera que para realizar la programación orientada a objetos resulta más compleja. El middleware Ginga-J no se puede instalar fácilmente en una PC con Sistema Operativo Ubuntu, por esta razón se utiliza la máquina virtual con Ginga-J pre-instalado que nos permite levantar un ambiente rápido, siendo esta una opción que está disponible al momento y de esta manera desplegar aplicaciones Ginga J. Para el buen funcionamiento de la arquitectura del Middleware Ginga-J se debe usar las funcionalidades que tienen los API's estandarizadas Ginga-J, ya que los Xlets para su funcionamiento deben utilizar API's estandarizados provistos por Ginga-J.

Para el desarrollo adecuado de aplicaciones de corto alcance se recomienda el uso de la metodología de desarrollo OpenUP / Basic, ya que si se cumple los roles y las iteraciones que se definen en el tiempo estimado tendremos el control que se necesita en el desarrollo de sistemas informáticos. El Proyecto ESPE-GINGA debe realizar una investigación de la incompatibilidad de los componentes gráficos desarrollados con el API HAVI en la versión actual de OPENGINGA, para de esta manera poder realizar el desarrollo de librerías genéricas compatibles que mejoren el rendimiento de dichos componentes gráficos y compartir a la comunidad Ginga j. Para comenzar el desarrollo de aplicaciones de tv digital basadas en Ginga-J se recomienda analizar la estructura de un Xlet aplicando los conocimientos del lenguaje Ginga-J orientado a objetos. El proyecto ESPE-GINGA debe investigar las limitaciones de memoria en el emulador para de esta manera aportar en la investigación del proyecto OpenGinga de Lavid.

Referencias Bibliográficas

- [1] Políticas de asignación de buffers de memoria. URL: <http://dSPACE.ups.edu.ec/bitstream/123456789/144/4/Capitulo%203.pdf>
- [2] GingaEcuador Hpage: URL <http://www.ginga.org.ec/gingaec/index.php?article&id=45&Itemid=54>
- [3] http://www.dtv.org.br/download/es-es/ABNTNBR15606-4_2010Esp_2010.pdf
- [4] GingaPerú Home Page: URL: <http://www.gingaperu.org/tvd-y-ginga/ginga.html>
- [5] <http://wiki.solar.org.ar/publico:ginga>
- [6] http://www.pleiad.cl/_media/research/adi/tvd-desarrollo.pdf
- [7] http://aat.inictel-uni.edu.pe/files/SET_TOP_BOX%28Informe_de_Avance1%29.pdf

La importancia de la Gestión de la Configuración del Software, en una Empresa de Desarrollo.

Santiago Paredes, Cecilia Hinojosa, Jenny Ruiz

*Departamento de Ciencias de la Computación, Escuela Politécnica del Ejército, Ecuador,
ogaitnas6@hotmail.com, cmhinojosa@espe.edu.ec, jaruiz@espe.edu.ec*

RESUMEN: La Gestión de la Configuración del Software (GCS) es un proceso de ingeniería, que indica procedimientos técnicos para el control y mejora de la calidad del software. Los subprocesos o fases que lo conforman son: Identificación de la Configuración, Control de Cambios, Control de Versiones, Informes de Estado, Auditoría de la Configuración. Cada una de las fases tiene parámetros para procesos de control que buscan la integración de las actividades relacionadas con el desarrollo del software desde las primeras fases, asignando roles y responsabilidades al personal de trabajo, para llegar a obtener un producto software de calidad; de esta manera se puede realizar un cambio o mantenimiento de una manera eficiente. Como contribución, el presente trabajo muestra el diseño de una metodología personalizada de GCS para una empresa, la cual fue realizada en base a un análisis comparativo entre el estándar ISO/IEC 12207, la metodología Métrica Versión 3, y el proceso RUP. Como resultado se han obtenido procesos claramente definidos para la identificación de los elementos de configuración, control de cambios y el control de versiones mediante herramientas software.

Palabras Clave: Gestión de la Configuración del Software, Control de Versiones, Identificación de la Configuración, Control de Cambios.

ABSTRACT: Software configuration management (SCM) is a software engineering process, which gives technical procedures to control and increase software quality. The sub process or phases are: Configuration identification, change control, versioning control, status reporting, and configuration audit. Each one of these phases set parameters for procedures and control in order to integrate development activities since the beginning of software life cycle, establish roles and team responsibilities so we can get a software quality product so we could easily do a change or give maintenance in a efficient way. The main the contribution of this paper is to design a personalized approach to SCM for a company, which was carried out based on a comparative analysis of the ISO / IEC 12207, the methodology metric version 3, and RUP process. As a result we were obtained, a clearly defined process for the identification of configuration items, change control and version control using software tools.

KeyWords: Software Configuration Management, Versioning Control, Configuration Identification, Change Control

1. INTRODUCCIÓN

El cambio es un factor que se encuentra presente durante todo el ciclo de vida del software, por lo que es muy importante llevar un control y registro riguroso del mismo. Este control debe ir orientado a reducir errores, aumentar la calidad y productividad; así como, evitar los inconvenientes que puedan causar una mala sincronización de los elementos de configuración del software.

Uno de los problemas en las empresas de desarrollo de software es el no contar con procesos bien definidos o hacer mal uso del proceso de Gestión de Configuración del Software (GCS), tal es el caso de la empresa auspiciante. La GCS es un proceso de ingeniería, que indica procedimientos técnicos para el control y mejora de la calidad del software. Los subprocesos o fases que lo conforman son: Identificación de la Configuración, Control de Cambios, Control de Versiones, Informes de Estado, Auditoría de la Configuración. Cada una de las fases tiene parámetros para procesos de control que buscan la integración de las actividades relacionadas con el desarrollo del software desde las primeras fases, asignando roles y responsabilidades al personal de trabajo, para llegar a obtener un producto software de calidad; de esta manera se espera el poder realizar un cambio o mantenimiento de una manera eficiente [1]. El aporte del presente trabajo es el diseño de una metodología personalizada de GCS para una empresa auspiciante, la cual fue realizada en base a un análisis comparativo entre el estándar ISO/IEC 12207 [2], la metodología Métrica Versión 3 [3], y el proceso RUP [4]. Obteniendo como resultado procesos claramente definidos para la identificación de los elementos de configuración, control de cambios y el control de versiones mediante herramientas software.

Para el desarrollo del proyecto se procedió a realizar una investigación bibliográfica de metodologías y estándares relacionados con la GCS. Tomando como marco de referencia las fases del proceso de GCS, se realizó un análisis comparativo entre las metodologías Métrica v3, RUP y el estándar ISO/IEC 12207, buscando determinar sus fortalezas y debilidades. Este estudio inicial fue la base para diseñar un proceso de GCS que resulte aplicable a la empresa auspiciante. Se definieron roles y responsabilidades para realizar las actividades diseñadas para el control de cambios, identificación de los elementos de configuración y realización de pruebas que involucran un correcto proceso de GCS. Otro punto importante desarrollado fue la búsqueda y definición de las herramientas informáticas que brinden un soporte adecuado al proceso de control de versiones.

Durante la implementación de los procesos diseñados y la implantación de las herramientas para el control de versiones, se debió enfrentar inicialmente la resistencia al cambio, una vez que los usuarios evidenciaron los beneficios del trabajo en base a lineamiento claramente definidos, se consiguió el involucramiento de todos los participantes.

El resto del artículo ha sido organizado de la siguiente manera: La sección 2 describe un resumen del marco teórico que se utilizó de referencia. La sección 3 muestra el análisis comparativo entre las metodologías y estándares, el planteamiento de una metodología propuesta, la implementación de la metodología, las conclusiones y recomendaciones. La sección 4 muestra los resultados alcanzados. Del monitoreo realizado, se pudo evidenciar una mejora del proceso de desarrollo de software, ya que se disminuyó el 36,36% de errores; demostrando la efectividad del proceso diseñado. Finaliza con las conclusiones derivadas del presente trabajo.

2. MARCO TEORICO

Para realizar el análisis comparativo se seleccionaron metodologías y estándares que cumplen o involucran a la Gestión de la Configuración del Software, con esto se obtuvo la adaptación de las actividades según los requerimientos y forma de trabajo de la organización en la cual se implementó la GCS.

2.1 Gestión de la Configuración del Software

“La Gestión de la Configuración del Software (GCS) es una actividad de autoprotección que se aplica durante el proceso del software” [1]. La GCS compromete factores como la identificación de la configuración, control de la configuración, informes de estado, revisión, crear procesos de administración y equipo en trabajo para así mejorar las actividades y software desarrollado [5].

2.2 Beneficios de la Gestión de la Configuración del Software

El proceso de la gestión de la configuración del software tiene varios beneficios en la organización en la que se lo practique. Desarrolladores, “testers”, jefes de proyecto, personal encargado de la calidad del sistema y los clientes pueden obtener beneficios del proceso de la gestión de la configuración del software; entre los siguientes [6]:

- i. Provee la habilidad de dar seguimiento a los cambios durante el desarrollo, sea este secuencial o en paralelo.
- ii. Organiza las tareas y actividades que mantienen la integridad del software.
- iii. Asegura la configuración correcta del software.
- iv. Asegura que los ingenieros implementen cambios correctamente en la línea base o en la versión del software.
- v. Ayuda a reducir el costo de la mantención del ciclo de vida del software, que puede ser fácilmente excedido.
- vi. Provee información para reportes que pueden ser fácilmente generados.
- vii. Permite realizar auditorías rápidas y fáciles.
- viii. Ayuda en la producción de un software de mayor calidad.

2.3 Metodologías y estándares que involucran a la GCS

2.3.1 Métrica Versión 3

Métrica V3 es una metodología útil para la sistematización de actividades que permite definir cuáles son las necesidades de la GC de los diferentes sistemas de información. La Gestión de la Configuración viene dada como una interfaz mostrada en la Fig 1. que consiste en la aplicación de procedimientos tanto administrativos como técnicos durante el desarrollo, y posterior mantenimiento del sistema de información; teniendo como finalidad identificar, definir, proporcionar información y controlar los cambios en la configuración del sistema, así como modificaciones y versiones de los mismos [3].

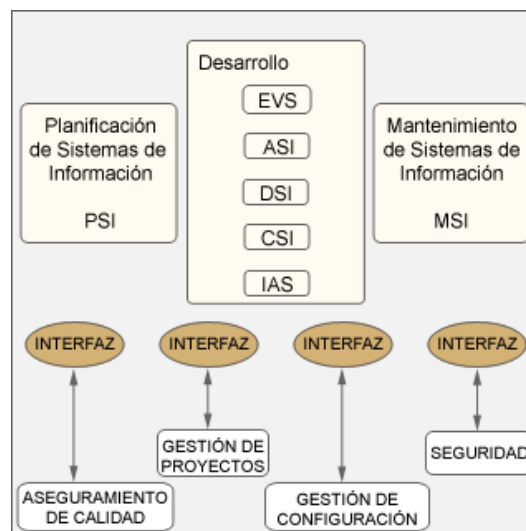


Figura 1. Estructura Métrica V3 [3]

2.3.2 ISO/IEC 12207 (Proceso del Ciclo de vida del Software)

Es un estándar internacional que muestra el proceso del ciclo de vida del software, actividades y tareas que son parte de un gran sistema. Dentro de todo este proceso se encuentra la Gestión de configuración como un proceso de apoyo del ciclo de vida del software como se indica en la Fig 2., que identifica, define y establece la línea base de los elementos software de un sistema, controla modificaciones y liberaciones (releases) de los elementos; informa además el estado de cada uno de los elementos y peticiones de modificación; asegura la completitud, consistencia y corrección de los elementos; y controla su almacenamiento, manipulación y entrega [2].

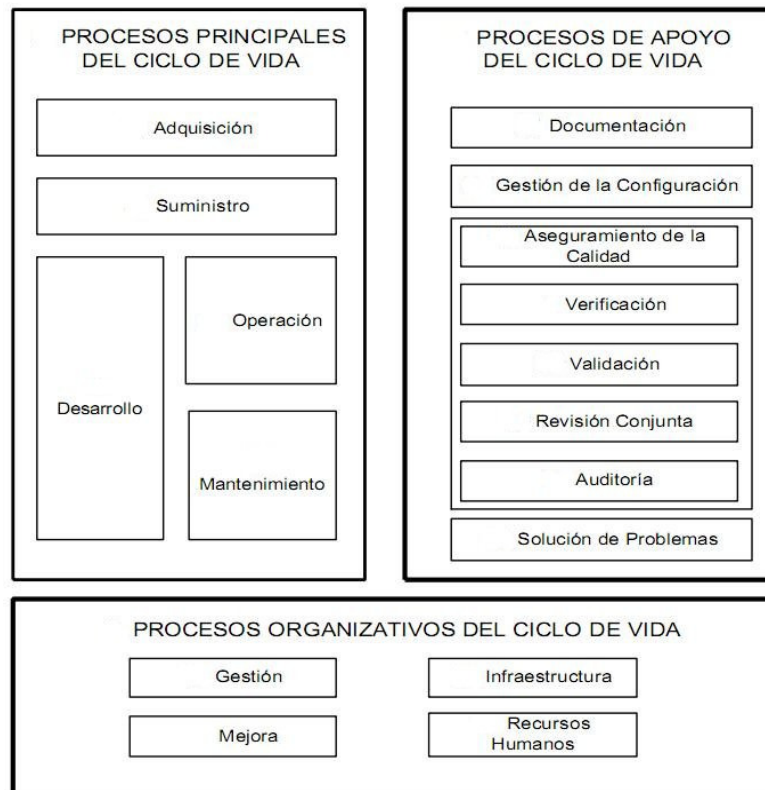


Figura 2. Grupos de procesos del ciclo de vida del software [5].

2.3.3 RUP (Rational Unified Process)

Es un proceso de Ingeniería de Software que abarca la asignación de tareas y responsabilidades dentro de una organización de desarrollo [4]. El objetivo de RUP es producir software de calidad que resuelva las necesidades y requerimientos de los usuarios cumpliendo tiempo y presupuesto establecido.

La Gestión y configuración de cambios ilustrado en la Fig 3., tiene como objetivo controlar y mantener la integridad de los productos que incluyen un proyecto mediante la gestión de solicitudes de cambios, plan de configuración del proyecto y control de cambios, creación de un entorno de gestión de configuración del proyecto, supervisión e informe del estado de la configuración, elementos de la configuración de entrega y cambios, control de líneas base y lanzamientos [4].

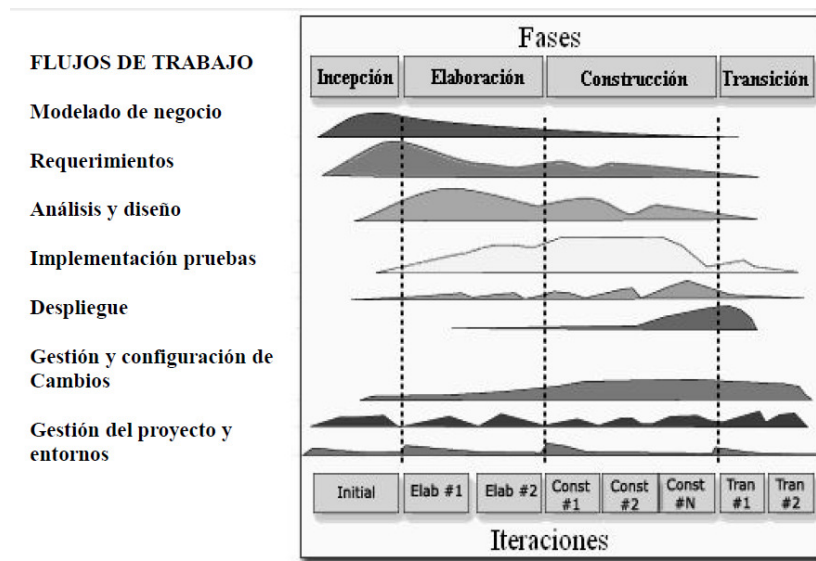


Figura 3. Disciplinas, fases, iteraciones de RUP [7]

3. ANALISIS COMPARATIVO DE LAS METODOLOGIAS

3.1 Identificadores de evaluación

Se tomaron dos aspectos para la evaluación de modelos, metodologías o estándares:

- i. Especificación de actividades: Describe las actividades que se deben realizar.
- ii. Establecimiento de roles: Se indica a las personas involucradas, con sus roles en las diferentes actividades.
- iii. Herramientas: Existen herramientas en el mercado que permitan la automatización de los procesos.

3.2 Pesos para evaluación

La Tabla I muestra los valores y su descripción con los cuales se evaluará a las metodologías y estándares que dentro de sus procesos involucran a la Gestión de la Configuración del Software.

TABLA I. Pesos para Evaluación Gestión de la Configuración del Software

Valor	Título	Descripción
0	Nulo	No existe.
1	Cumple Poco	Provee poca información sobre la forma de ejecutar el proceso.
2	Cumple Medianamente	Provee información en una manera satisfactoria sobre la ejecución del proceso.
3	Cumple Adecuadamente	Provee información en una manera adecuada y clara sobre la forma de ejecutar el proceso.

3.3 Evaluación del Proceso de GCS

Para realizar el análisis comparativo se seleccionaron metodologías y estándares que involucran a la Gestión de la Configuración del Software, con esto se buscará adaptar de la mejor manera a los procesos según los requerimientos y forma de trabajar de la organización en la cual se implantó la GCS. La Tabla II muestra el análisis en cada fase del proceso de GCS con los estándares y metodologías elegido.

TABLA II. Análisis fases procesos Gestión de la Configuración del Software

IMPLEMENTACION DEL PROCESO				
	Especifica actividades	Establece Roles	TOTAL	
ISO/IEC 12207	3	1	4	
METRICA V3	2	1	3	
RUP	3	2	5	
IDENTIFICACIÓN DE LA CONFIGURACIÓN				
	Especifica actividades	Establece Roles	Herramientas	TOTAL
ISO/IEC 12207	3	1	0	4
METRICA V3	1	1	0	2
RUP	1	1	1	3
MANTENIMIENTO Y CONTROL DE LA GESTION DE LA CONFIGURACIÓN				
	Especifica actividades	Establece Roles	Herramientas	TOTAL
ISO/IEC 12207	2	1	2	5
METRICA V3	2	2	2	6
RUP	3	2	2	7
INFORME DE ESTADO DE LA CONFIGURACION				
	Especifica actividades	Establece Roles	Herramientas	TOTAL
ISO/IEC 12207	3	1	0	4
METRICA V3	0	0	0	0
RUP	1	1	1	3
AUDITORIA DE LA CONFIGURACIÓN				
	Especifica actividades	Establece Roles	Herramientas	TOTAL
ISO/IEC 12207	2	1	0	3
METRICA V3	0	0	0	0
RUP	2	1	1	4

Al concluir con el análisis y evaluación de las actividades dentro de los elementos funcionales del proceso de Gestión de la Configuración del Software, se escogió los procesos de las metodologías y estándares que obtuvieron la mayor puntuación; los cuales puedan ayudar a la empresa acoplándose a las necesidades de la organización y así solucionar los problemas que se busca enmendar.

Con el análisis realizado en la Tabla II, se propuso una metodología personalizada para la organización, estableciendo actividades, roles, herramientas y entregables durante la implementación del proceso de GCS que se muestra en la Tabla III.

TABLA III. Resumen metodología propuesta

	ACTIVIDADES	ROLES	HERRAMIENTAS	ENTREGABLE
IMPLEMENTACIÓN DEL PROCESO	<ul style="list-style-type: none"> - Definir prácticas de Identificación de la Configuración - Definir prácticas para las líneas base - Definir prácticas para archivar 	<ul style="list-style-type: none"> - Líder del Proyecto 		Políticas para la Gestión de la Configuración del Software
IDENTIFICACIÓN DE LA GCS	<ul style="list-style-type: none"> - Selección de los Elementos de la Configuración del Software (ECS) - Identificación de líneas base - Definir esquema de Identificación - Definiciones y establecimiento de Bibliotecas Software <ul style="list-style-type: none"> Biblioteca de Trabajo Biblioteca de Soporte al Proyecto Biblioteca Maestra Biblioteca Backup 	<ul style="list-style-type: none"> - Líder del Proyecto - Gestor de la Configuración del Software - Desarrollador 		Plan de Gestión de la Configuración del Software
MANTENIMIENTO Y CONTROL DE LA GCS	<ul style="list-style-type: none"> - Plan de Pruebas - Casos de Pruebas - Control de Versiones - Control de Cambios 	<ul style="list-style-type: none"> - Líder del Proyecto - Gestor de la Configuración del Software - Desarrollador - Comité de Control 	<ul style="list-style-type: none"> - Venture ERP Software - Power Builder 11.5 - Netbeans 6.5.1 - Netbeans 6.7 - Visual SVN Server - CollabNet Subversion Client 	<ul style="list-style-type: none"> - Plan de Pruebas - Casos de Pruebas
INFOME DE ESTADO DE LA GCS	<ul style="list-style-type: none"> - Seleccionar reportes basados en peticiones de cambios - Definir frecuencia de reportes 	<ul style="list-style-type: none"> - Líder del Proyecto - Gestor de la Configuración del Software 	<ul style="list-style-type: none"> - Venture ERP Software 	Políticas para la Gestión de la Configuración del Software
AUDITORÍA DE LA GCS	<ul style="list-style-type: none"> - Realizar Auditoría de la Configuración Física <ul style="list-style-type: none"> - Realizar Auditoría de la Configuración Personal - Reporte de Novedades 	<ul style="list-style-type: none"> - Gestor de la Configuración del Software 		

4. IMPLEMENTACIÓN

4.1 Implementación del Proceso

Como primer paso se elaboró un plan de gestión de la configuración. El plan describe las políticas con las que se trabajó, actividades de gestión de la configuración, procesos, responsables que llevarán a cabo dichas actividades.

4.2 Identificación de la Configuración

La identificación de los ECS es una de las actividades más importantes dentro del proceso de GCS ya que aquí se definieron los elementos que serán controlados como por ejemplo el código, documentos; y cómo estos elementos fueron identificados, marcados, numerados [5].

4.3 Control de Versiones

El control de versiones se llevó en los dos ambientes de trabajo que son los que maneja la empresa, uno Web y otro de ventanas. Se buscó una herramienta de versionamiento para cada uno de estas, indagando

en la Web que sean compatibles con las características de los computadores y se puedan integrar con los IDEs¹ de desarrollo; tomando en cuenta que sean gratuitos para ahorrar gastos a la empresa. Los IDEs con los que se trabaja en la organización son Netbeans para el ambiente Web [8], y Power-Builder para el entorno de ventanas [9]. La Fig 4 muestra el diseño para el versionamiento que se implementó para ambos ambientes de trabajo:

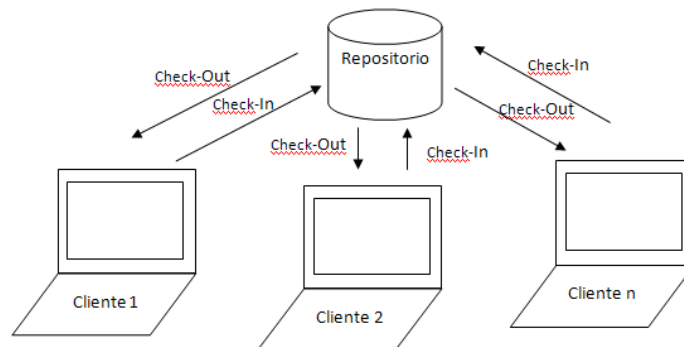


Figura 4. Diseño para el versionamiento implementado Venture Venti

La Tabla IV muestra las herramientas que se utilizaron para el control de versiones en los dos ambientes de desarrollo que posee la empresa Venture Venti, obteniendo así una mejora en la calidad del software, coordinación adecuada entre los grupos de desarrollo, y una mejora en la satisfacción de los clientes

TABLA IV. Herramientas para versionamiento

Software	Descripción	Framework
Visual SVN Server	Herramienta para la creación y administración de repositorios, creación de usuarios, que llevará el control de versiones desarrolladas.	Netbeans 6.5.1 o 6.7.
CollabNet Subversion Client	Plug-in que permite la integración de Visual SVN con Netbeans	Netbeans 6.5.1 o 6.7.
PB Native	Plug-in integrado que permite el control de versiones en objetos creados de una manera individual.	Power Builder 11.5

4.4 Control de Cambios

Los cambios en objetos se dan por la petición de un requerimiento de un cliente o por parte del personal de la empresa para rastrear defectos o agregar modificaciones en el software. Se establecieron procesos para organizar las actividades que se realizarán, como la evaluación, informe de la necesidad del cambio, generación de una orden, ejecución y finalmente la revisión del cambio realizado [6].

¹ Integrated Development Environment (Entorno de Desarrollo Integrado)

4.5 Informes de cambio de la configuración

En esta fase se definieron políticas de entrega de informes de cambios de los elementos de la configuración, estableciendo frecuencia, quién será el encargado de realizar los informes y presentarlos al líder del proyecto [5].

4.6 Auditoría de la configuración

Se establecieron dos procesos de control que se podrían realizar como revisiones técnicas formales y auditorías de configuración del software y así poder a futuro completar el proceso de gestión de la configuración del software mediante un control de los cambios que han sido realizados en los diferentes elementos de la configuración [7].

5. EVALUACION DE RESULTADOS

5.1 Evaluación de objetivos alcanzados

La Tabla V muestra la evaluación por objetivos como resultado del procedimiento de comparación realizado

TABLA V. Evaluación de objetivos

Acción	Objetivo	Resultado
Identificación de la configuración	Identificar los ECS(Elementos de Configuración de Software)	Se definieron los ECS necesarios para llevar un control de los mismos
Control de cambios	Diseñar e implementar el proceso de control de cambios	Controlar las modificaciones de los elementos de configuración realizadas por el equipo de trabajo.
Control de versiones	Definir la mejor herramienta para realizar el control de versiones	Mejorar la calidad del software que se entrega a los clientes

Al realizar el análisis de los resultados obtenidos, hay que tomar en cuenta un aspecto importante sobre la GCS. Sin una identificación adecuada de los elementos de gestión no será posible realizar un proceso de GCS adecuado y óptimo ya que sin una base apropiada el resto de procesos se verá afectado y se encontrarán incongruencias en los procesos y actividades de gestión de configuración.

5.2 Mejoras obtenidas

Al haber finalizado la implementación de procesos de GCS en la empresa auspiciante se puede evidenciar una mejora considerable en los últimos 9 meses con los reportes basados de registro de novedades del sistema ERP de Venture mostrados en la Tabla VI. Indicándonos mejoras al utilizar la gestión de la configuración del software en una forma gradual pudiéndose demostrar que las fallas más graves como el error de programación nivel 3, error de soporte nivel 3 y no realizar control de calidad nivel 3 se disminuyeron en un 100%; mientras que otros problemas se redujeron en porcentajes desde un 11.11% hasta un 75% lo que son unos claros indicadores de que el proceso ha ayudado a mejorar los procesos de GCS y calidad del software en la organización.

TABLA VI. Mejoras obtenidas con el proceso de GCS

Problema	Abr-Dic 2010	Ene-Sep 2011	Diferencia
Error de programación nivel 1	7	5	-2
Error de Programación nivel 2	2	2	0
Error de Programación nivel 3	6	0	-6
Error de soporte nivel 3	3	0	-3
No actualizar código en el servidor	9	8	-1
No cumplir estándares	12	3	-9
No realiza control de calidad 1	1	0	-1
No realiza control de calidad 3	3	0	-3

NOTA: La valoración de los procesos en la Tabla VI está definida con escala de 1 a 3, donde 3 es un tipo de falta de gravedad alta

6. TRABAJOS RELACIONADOS

No se encontró ningún trabajo en el cual en base a propuestas de procesos de GCS se diseñe una metodología personalizada para la implementación en una organización, sin embargo si existen trabajos que de una forma general hablen sobre el proceso de GCS. Así por ejemplo el trabajo propuesto por [10] titulado Software Configuration Management. En este mismo contexto, el trabajo propuesto por [11] titulado Version models for software configuration management

7. CONCLUSIONES

En base al análisis comparativo considerando el proceso de GCS propuesto por varias metodologías y estándares, se diseñó e implementó un proceso personalizado de GCS en la empresa de desarrollo Venture Venti. El cual ayudó a solventar de una forma muy satisfactoria los problemas que se tenía durante el desarrollo de software. Se diseñó un proceso de control de cambios bien definido, el mismo que especifica roles de trabajo y las actividades que cada participante realizará, buscando siempre mejorar la calidad del trabajo individual y colectivo de los involucrados. En cuanto al proceso de pruebas se planteó un plan y definieron casos de pruebas que empleará el personal encargado del desarrollo, como los responsables de su control mediante el uso de plantillas, dando como resultado un mejor control de los cambios efectuados sobre el software. Adicionalmente, se implementaron herramientas de versionamiento con el objetivo de evitar problemas de códigos incompletos o con errores para los IDEs de desarrollo con los que trabaja la empresa, obteniendo una mejor calidad del código desarrollado y reducción de quejas por parte de los clientes. Finalmente, al establecer el rol específico y claro para los empleados en las diferentes etapas del proceso de GCS, se logró reducir la carga de trabajo obteniendo un mejor desempeño laboral del personal. Con la integración de las fases diseñadas para el proceso de GCS en la empresa auspiciante, se logró establecer un mejor entorno de trabajo entre los empleados creando más confianza en sí mismo, seguridad de su trabajo y del resto de personas; ya que los procesos son claros y las herramientas de control útiles.

Como trabajo futuro, con la implementación del proceso de GCS se obtiene mejoras notorias en los procesos de desarrollo de software; el cual podría ser una base adecuada para buscar la aplicación de un estándar de calidad de software como la ISO/IEC 9126 (estándar internacional para la evaluación de la calidad del software), dando así a la organización un fortalecimiento en el mercado de los productos software que desarrollan.

Referencias Bibliográficas

- [1] Pressman, R. (2002). *Ingeniería de Software, Un Enfoque Práctico*. Madrid: Mc Graw Hill.
- [2] Comisión de Reglamentos Técnicos y Comerciales-INDECOPI. (2006). NTP-ISO/IEC 12207. Lima, Perú.
- [3] Ministerio de Administraciones Públicas de España. *Dirección General para el Impulso de la Administración Electrónica*. Obtenido de http://administracionelectronica.gob.es/?_nfpb=true&_pageLabel=P60085901274201580632&langPae=es
- [4] Kruchten, P. (2000). *The Rational Unified Process an Introduction*. Addison Wesley.
- [5] Dart, S. (1992). *The Past, Present, and Future of Configuration Management*. Pittsburgh.
- [6] Keyes, J. (2004). *Software Configuration Management*. Boca Raton: CRC Press.
- [7] Rational the software development company. (1998). Obtenido de http://www.ts.mah.se/RUP/RationalUnifiedProcess/process/workflow/ovu_core.htm
- [8] Visual SVN. (2005). *VisualSVN - Subversion-based source control for Windows*. Obtenido de <http://www.visualsvn.com/server/>
- [9] SYBASE. *Sybooks Online*. Obtenido de http://infocenter.sybase.com/help/index.jsp?topic=/com.sybase.dc00844_1150/html/pbug/CHDBF FEJ.htm
- [10] Berlack, R. (15 de Enero de 2002). *Wiley Online Library: Book Article*. Recuperado el 2011
- [11] Conradi, R., & Bernhard, W. (2 de Junio de 1998). *Version models for software configuration management*. Recuperado el 2011, de <http://dl.acm.org/citation.cfm?id=280280>

Sistema Cero Papeles para la Revisión de Correspondencia del Banco COFIEC SA.

R. Garcés, C. Hinojosa y J. Ruíz

Departamento de Ciencias de la Computación, Escuela Politécnica del Ejército, Sangolquí, Ecuador
raulfernandogarces@gmail.com, chinojosa@espe.edu.ec, jaruiz@espe.edu.ec

RESUMEN: COFIEC S.A es una entidad bancaria que utiliza para sus transacciones internas y externas servicios bancarios, uno de ellos es el servicio de revisión de correspondencia interna, el mismo que no cuenta con la eficiencia y la calidad que el banco necesita. Tomando como referencia los beneficios de la filosofía oficina cero papeles, el presente artículo muestra la solución tecnológica que se dio al servicio de revisión de correspondencia, usando un procedimiento de digitalización de documentos que contiene un sistema cero papeles desarrollado en base a la metodología OOHDM. El proceso de digitalización de documentos permite convertir, a través de un escáner; los documentos analógicos o en papel, en documentos electrónicos o digitalizados con formato pdf. Los documentos digitalizados son enviados a un directorio temporal de donde el sistema cero papeles los toma, los convierte en datos binarios sin formato, y los almacena en una base de datos llamada archivador digital central. Para la revisión de documentos, el sistema cero papeles toma los datos binarios almacenados en el archivador digital central, y se encarga de convertirlos en documentos digitalizados con formato pdf para leerlos desde la computadora sin la necesidad del papel. Gracias a los lineamientos de la metodología OOHDM, el análisis, diseño e implementación del sistema Web cero papeles fue factible y entendible, dando como resultado final una aplicación Web de calidad que permitió el aumento de la eficiencia del servicio de revisión de correspondencia interna, satisfaciendo las necesidades de los usuarios.

Palabras Clave: Digitalización, OOHDM, Web, Cero Papeles.

ABSTRAC: COFIEC SA., is an entity that uses bank services for its internal and external transaction. One of them is the internal correspondence review; which is not enough in efficiency and quality for its entity needs. Taking as a reference the profits of paperless office philosophy, this technical article shows the technology solution that was given to the review mail service, using for these methods of documents digitizing which have a paperless system developed in an OOHDM methodology base. The digital process documents allow us to convert analog or paper documents into electronic or digitized pdf format documents. The digitalized documents are sending to a temporary directory, where they are taken by paperless system. This one turns them into binary data without a format. Then it stores them in a database called central digital archive for the documents review, the paperless system takes them binary data which are stored in the central digital archive, and it is in charge to convert them into pdf format allowing the documents to be able to read from the computer without paper needs. Thanks OOHDM guidelines, analysis, design and paperless Web system implementation was feasible and comprehensible giving us as a final result a quality Web application that allowed increasing efficiency in internal review correspondence, satisfying need of users.

KeyWords: Digitalization, OOHDM, Web, Paperless.

1. INTRODUCCIÓN

En la actualidad, a pesar de los avances tecnológicos existentes; dentro de cualquier empresa, entidad bancaria o pública, aún existen procesos donde el uso del papel es indispensable, provocando pilas de papeles inútiles, desorden y mala organización; obteniendo como resultado final demoras en los procesos, pérdidas y deterioro de documentos importantes. El servicio de revisión de correspondencia interna de Banco COFIEC S.A no es la excepción, por esa razón dicho servicio es deficiente y con pésima calidad.

La solución a este problema es la eliminación de papeles físicos basados en la filosofía oficina cero papeles, que plantea el uso de un proceso de transferencia de archivos de papel en formato digital [1], conocido también como proceso de digitalización de documentos, el cual transforma la información de un papel en datos binarios ceros y unos con un tipo de formato de texto o de imagen para poder visualizar el documento digitalizado en la computadora; éste tipo de conversión se realiza con la entrada de datos al computador desde un escáner [2].

Sin embargo, el proceso de digitalización no solamente consiste en transformar el documento, sino también en proporcionarle gestión y visualización a través del almacenamiento de los documentos digitalizados en una base de datos llamada archivador digital central, de donde cualquier usuario tenga acceso a revisar, consultar, visualizar un documento, e ingresar una observación o comentario de lo revisado cuando lo desee desde su computadora sin utilizar papel. Éste gestión y visualización lo realiza la aplicación Web cero papeles mediante dos tareas; una de almacenamiento, que convierte los documentos digitalizados en tramas binarias de ceros y unos, almacenándolas ordenadamente en un campo binario de una Tabla en la base de datos; y la segunda tarea realiza lo inverso cuando se desea revisar documentos, tomando las tramas binarias de la base de datos y transformándolas en un documento pdf para su respectiva visualización.

Tomando como referencia los beneficios de la filosofía oficina cero papeles, el presente artículo técnico muestra la solución tecnológica que se dio al servicio de revisión de correspondencia, usando un procedimiento de digitalización de documentos que contiene un sistema cero papeles desarrollado en base a la metodología OOHDM.

OOHDM es una metodología de desarrollo orientada a objetos hipermedia, que tiene como objetivo diseñar aplicaciones Web de la manera más factible, más simple y más eficaz; combinando el lenguaje de modelado UML con otros lenguajes propios de la misma metodología durante sus cuatro primeras fases: Obtención de requerimientos, Modelamiento conceptual, Diseño Navegacional, y Diseño de Interfaces Abstractas; dejando a su última fase de lado ya que es la fase de implementación. El análisis, diseño, construcción e implementación de la aplicación Web cero papeles fue realizada bajo la metodología OOHDM.

El resto del artículo ha sido organizado de la siguiente manera: La sección 2 describe los métodos, procesos y esquemas que permitieron la digitalización de documentos y la creación de la aplicación Web cero papeles. La sección 3 detalla las fases del diseño de la aplicación Web cero papeles basado en la metodología OOHDM incluida la fase de implementación. La sección 4 muestra la información de los resultados finales sobre calidad, eficiencia y satisfacción de los usuarios, con respecto al uso de la aplicación Web cero papeles en banco COFIEC. La sección 5 analiza las respectivas relaciones con otros trabajos. Finalmente, la sección 6 muestra las Conclusiones de éste artículo y el trabajo futuro.

2. MÉTODOS, PROCESOS Y ESQUEMAS

2.1 Proceso de Digitalización de Documentos

La digitalización de documentos es la conversión de un documento en papel, a un documento digital bajo un formato de texto o de imagen específico que permitan ser visualizados en la computadora. Para el caso de este estudio, se decidió convertir los documentos en papel a documentos pdf. Antes de visualizar el proceso de digitalización de documentos, es necesario aclarar algunos de sus pasos:

- i. El directorio maestro, donde van a alojarse temporalmente los documentos pdf, debe ser creado en el servidor de aplicaciones.
- ii. La computadora del administrador de correspondencia es la única que debe tener conectado un escáner e instalado su respectivo software de configuración.
- iii. El administrador de correspondencia es la única persona que debe realizar la calibración del escáner y sus respectivas pruebas, reconfigurando en el respectivo software del escáner para que los documentos que se escaneen lleguen al directorio maestro en el servidor de aplicaciones bajo el nombre de digital.pdf.
- iv. Los pasos finales del proceso (gestión de documento digitalizado y visualización de documento digitalizado) lo debe realizar la aplicación Web cero papeles, ya que se refiere al almacenamiento de documentos digitalizados en la base de datos, y a la visualización de documentos digitales durante la revisión de correspondencia, respectivamente.

2.2 Gestión y Visualización de Documentos Digitalizados

El anterior proceso de digitalización de documentos puede transformar un documento en papel a un documento digital con formato pdf, y almacenarlo temporalmente en un directorio maestro. Seguidamente, existen dos pasos del proceso llamados Gestión de documento digitalizado y Visualización del documento digitalizado, que se refiere al almacenamiento del documento pdf en la base de datos y a su respectiva visualización. De estas tareas se encarga la aplicación Web cero papeles, guardando el documento pdf en la base de datos, y descargando lo guardado en la base de datos para la visualización, para esto; la aplicación debe seguir procesos de conversión, de acuerdo a los siguientes esquemas que se muestran en las Figuras 2 y 3.

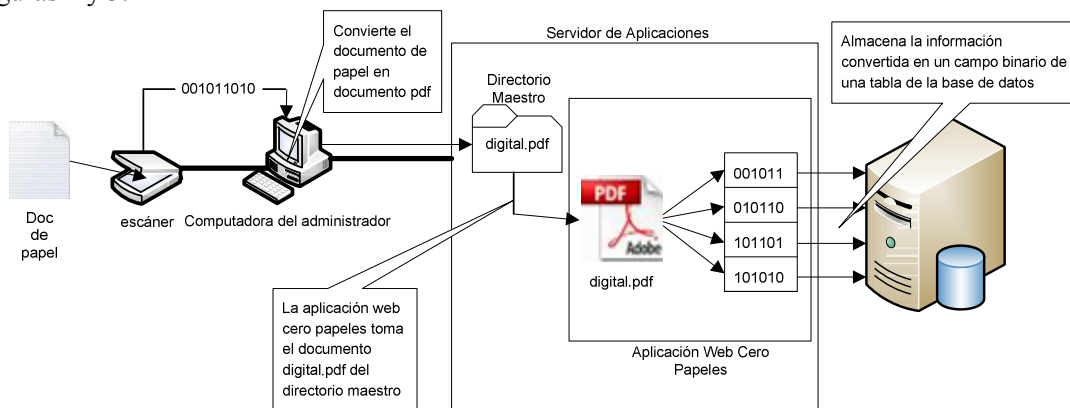


Figura 1. Esquema de Almacenamiento de Documentos Digitalizados

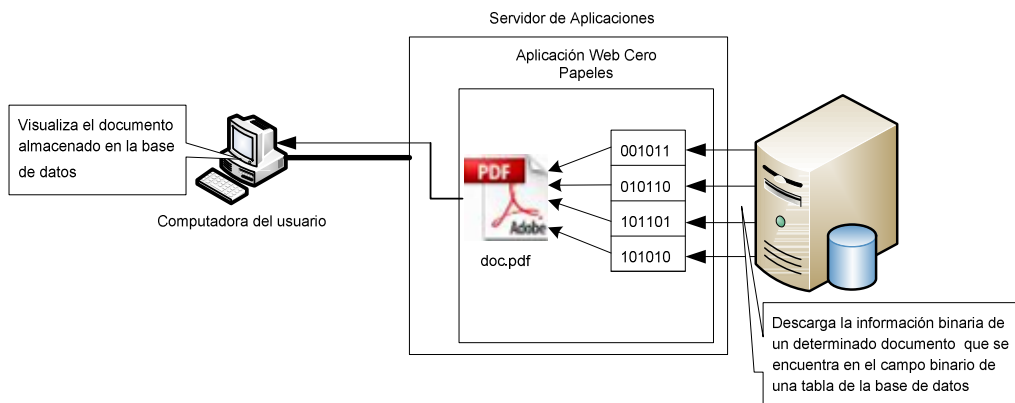


Figura 2. Esquema de Visualización de Documentos Digitales Almacenados

2.3 Metodología OOHDM

Para realizar los pasos de Gestión y Visualización de documentos digitalizados, se necesita una aplicación, por esa razón; se decidió construir una aplicación Web cero papeles que realice el almacenamiento, la visualización, y todos los requerimientos restantes que los usuarios necesitan en el servicio de revisión de correspondencia interna de banco COFIEC.

El diseño de cualquier aplicación debe seguir un lineamiento establecido por una metodología, por esa razón; se decidió utilizar OOHDM, que es una metodología orientada a objetos hipermedia, cuyo objetivo es diseñar aplicaciones Web; combinando el lenguaje de modelado UML con otros lenguajes propios de la misma metodología. La metodología OOHDM tiene las siguientes etapas:

TABLA I: Etapas y Subetapas de la metodología OOHDM

Etapa	Subetapas
Obtención de Requerimientos	Identificación de Roles y Tareas Especificación de Escenarios Especificación de Casos de Uso Especificación de UIDs Validación de Casos de Uso y UIDs
Modelo Conceptual	Diagramas de Clases Diagramas de Secuencia Diagrama de la Estructura de Base de Datos
Modelo Navegacional	Aplicación de Diseño Navegacional Esquema de Clases Navegacionales
Diseño de Interfaz Abstracta	Advs del sistema
Implementación	Desarrollo del Sistema Pruebas del Sistema

La Fig. 3 se muestra la relación que tienen las etapas OOHDM a la hora de diseñar una aplicación Web.

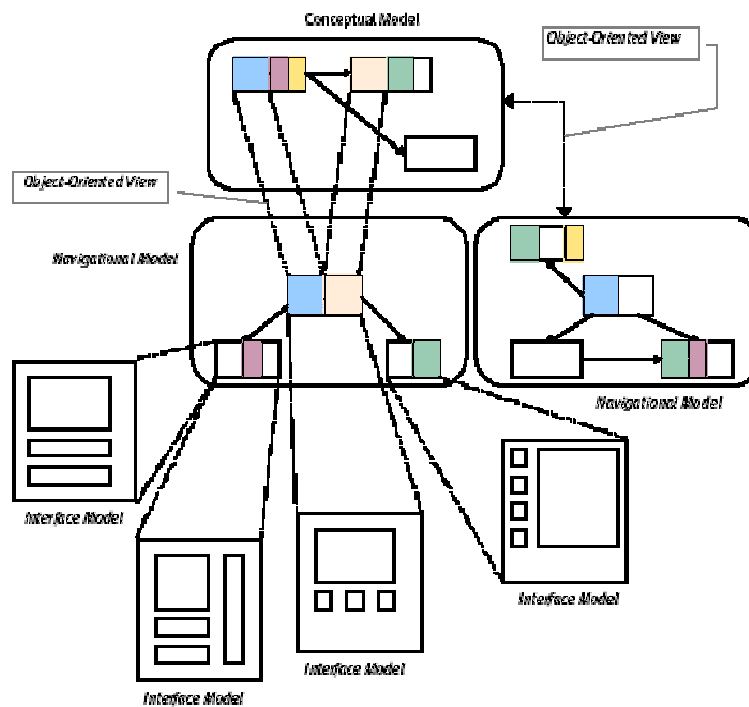


Figura 3. Relación entre Etapas de la Metodología OOHDM [3]

3. DISEÑO E IMPLEMENTACIÓN

El diseño, según la metodología OOADM, inicia con la obtención de requerimientos. En esta fase, se realizó primeramente el documento de Especificación de Requerimientos del sistema basado en el estándar IEEE-830, teniendo identificados como actores principales al administrador de correspondencia y a los usuarios que la revisan. Después, con los requerimientos del sistema, se identificaron los roles y las tareas de los actores; los cuales llevaron a identificar también los principales escenarios que tendrá la aplicación Web cero papeles. Una vez identificados los escenarios, se pudo diseñar el diagrama de casos de uso de general. La figura 4 muestra los casos de uso de la aplicación Web cero papeles.

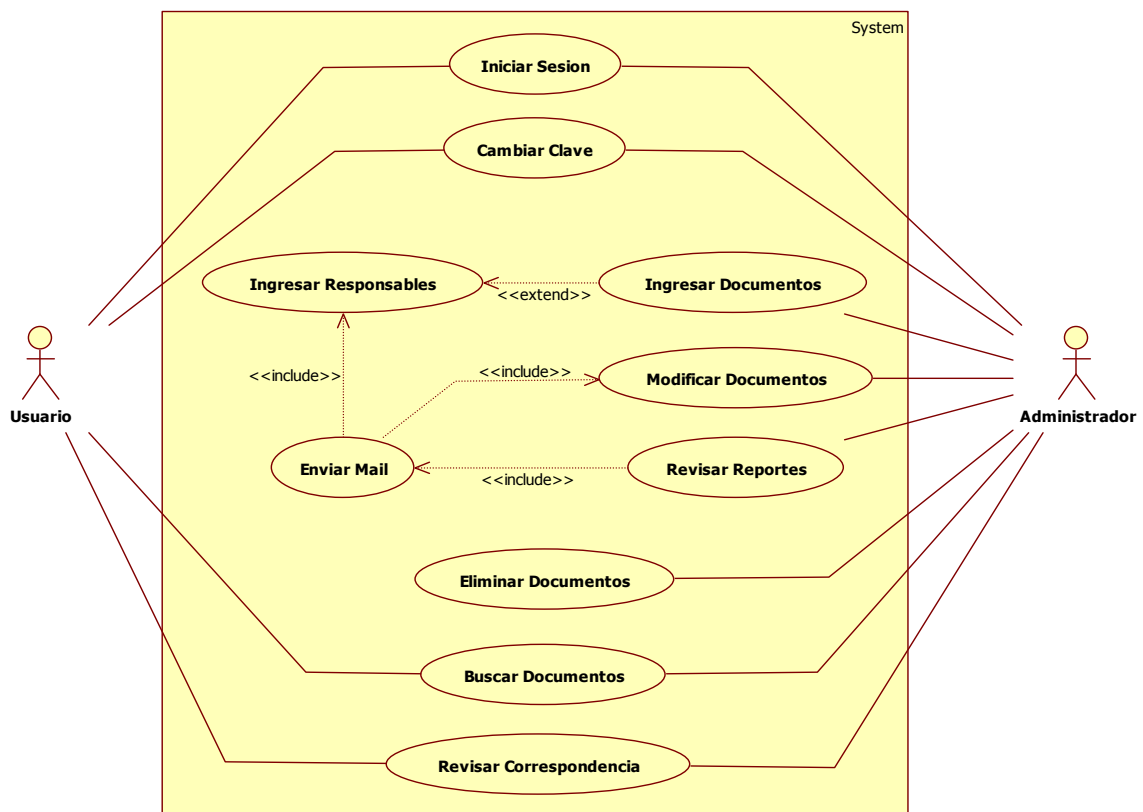


Figura 4. Diagrama General de Casos de Uso de la Aplicación Web Cero Papeles

Diseñados los casos de uso, se prosiguió con el diseño de las clases del negocio, diseño de las secuencias de los procesos de cada caso de uso y el diagrama de la estructura de la base de datos que permita almacenar datos de tipo binario.

La siguiente etapa de OOADM es la del diseño del modelo navegacional, que consiste en unificar todo lo anteriormente diseñado (casos de uso, clases del negocio y secuencias de cada caso de uso) con clases navegacionales o interfaces de usuario; obteniendo con esto una topología navegacional de toda la aplicación.

En este punto, se tiene diseñada toda la parte interna de la aplicación Web, faltando solamente el diseño de lo que el usuario va a observar en el monitor. Con la etapa de la interfaz abstracta, se puede diseñar con detalle cada pantalla de cada interfaz de usuario, estos diseños son llamados advs. La figura 5 muestra el ADV de inicio de sesión de la aplicación Web cero papeles.

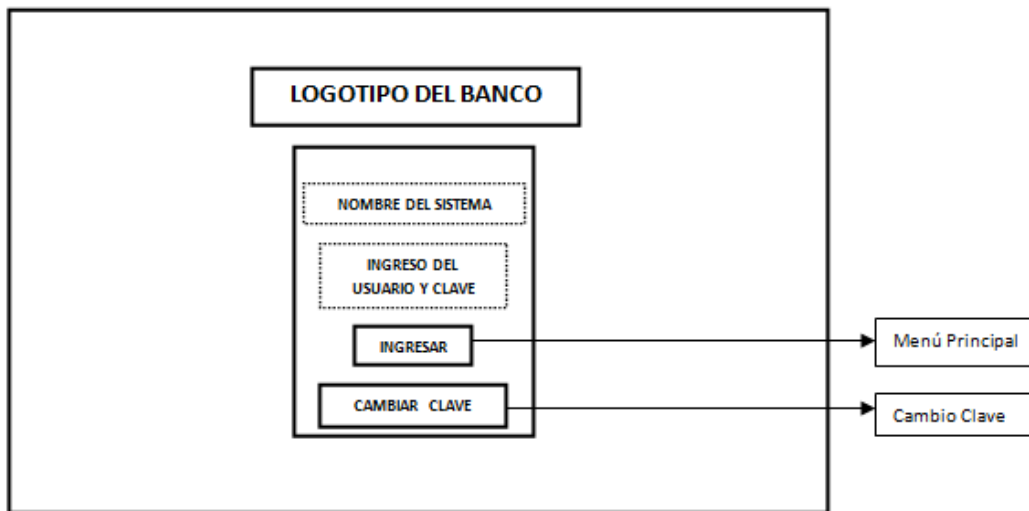


Figura 5. ADV Inicio de Sesión de la Aplicación Web Cero Papeles

Al finalizar los advs, se termina la fase de diseño de la aplicación Web cero papeles, y con esto, se puede avanzar a la última etapa de implementación, que consiste en desarrollar el sistema siguiendo una arquitectura de programación basándose en los diagramas diseñados anteriormente, realizar pruebas y poner en producción el sistema para que los usuarios puedan utilizar sin problemas.

La aplicación cero papeles fue desarrollada siguiendo una arquitectura multicapas, en donde se tiene una capa de presentación, una de negocio y una de base de datos. La capa de presentación está dividida en dos subcapas: una capa front end para entrada, salida y visualización de datos, y otra capa de análisis de datos que interactúa directamente con la capa de negocio y realiza verificaciones, autorizaciones y validaciones de los datos ingresados por los usuarios, ayudando a que la capa de negocio trabaje con la información justa que necesita. Las Figuras 6 y 7 muestran una de las pantallas de la aplicación Web cero papeles después del desarrollo basándose en los diagramas diseñados con la metodología OOADM y la topología de red implantada con la arquitectura de programación 3 capas.

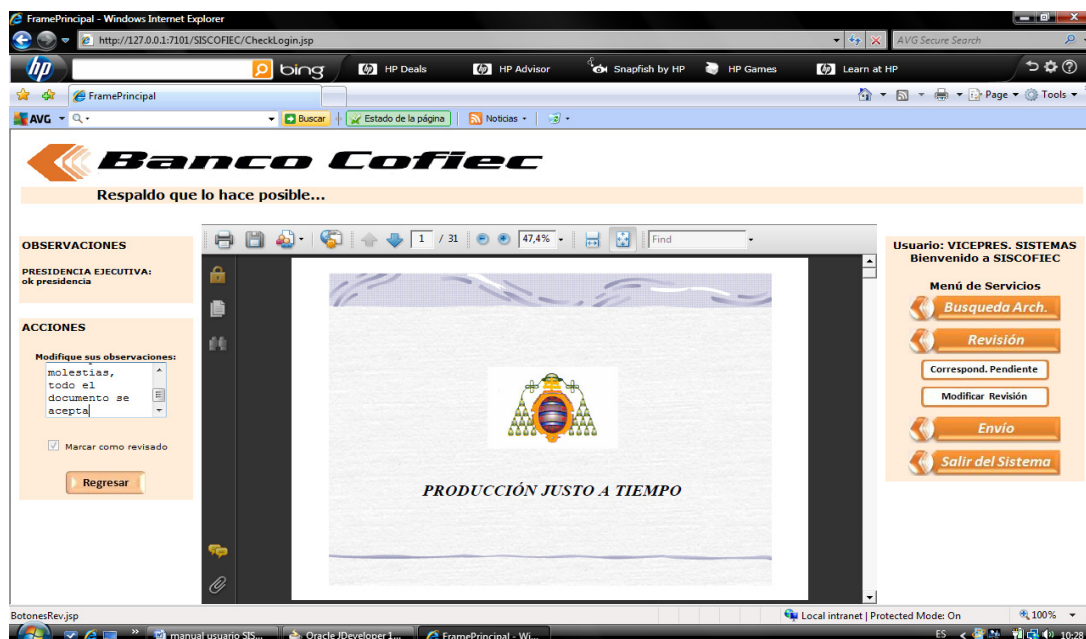


Figura 6. Pantalla de Revisión de la Aplicación Web Cero Papeles

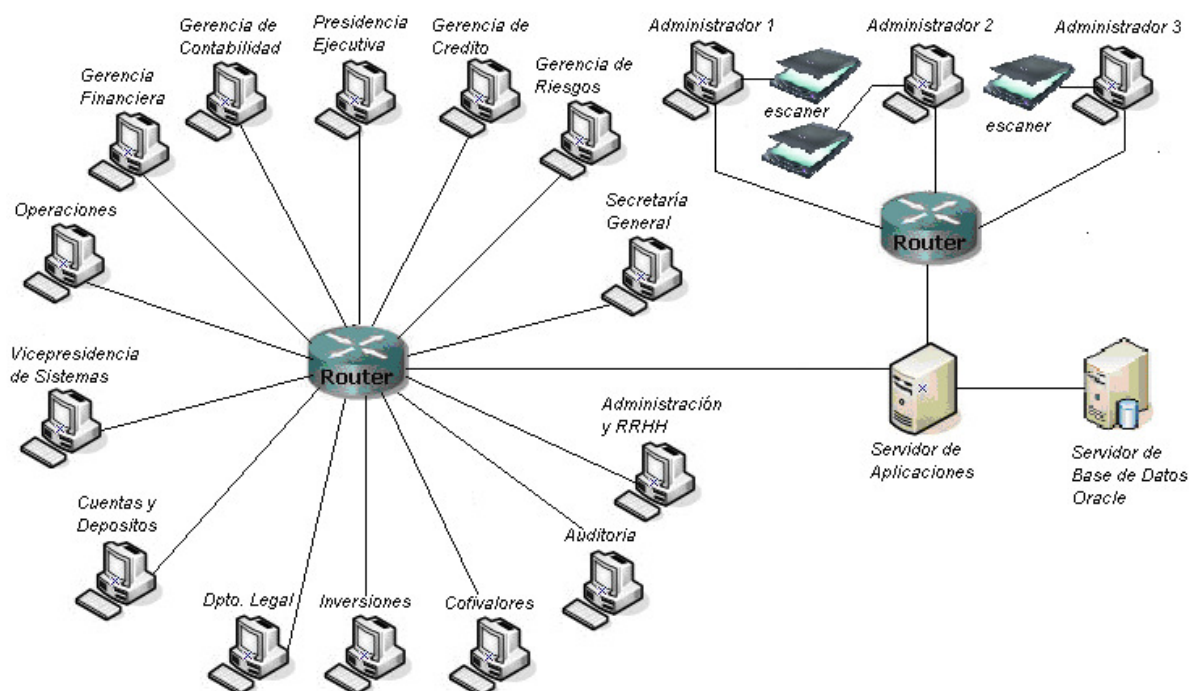


Figura 7. Topología de Red para la Aplicación Web Cero Papeles

4. EVALUACION DE RESULTADOS

Después de la implementación, banco COFIEC ha bautizado a la aplicación Web con el nombre de “Sistema Cero Papeles para la Revisión de Correspondencia Interna”; y se espera que la calidad y eficiencia en el servicio de revisión de correspondencia interna hayan mejorado. Con una muestra de 5 empleados, 4 con perfil de usuario y uno con perfil de administrador, un mes después de la implementación del sistema; se ha realizado una encuesta, la misma que arrojó los siguientes resultados.

4.1. Resultados Sobre Calidad en el Servicio de Revisión de Correspondencia Interna

TABLA II: Resultados de la Encuesta sobre la Calidad del Sistema

Usuario	Tiempo Uso	% Aceptación Nuevo Sistema	Grado de Satisfacción
1 (admin)	De uno a tres meses	100%	9
2	Menos de un mes	75%	9
3	Menos de un mes	75%	9
4	Menos de un mes	75%	8
5	Menos de un mes	75%	9
Promedios		80%	8,8

TABLA III: Resultados de la Encuesta sobre Atributos del Sistema

Atributos del Sistema	Promedio 5 usuarios
Amigable	9,2
Entendible	8,8
Facilidad de Uso y Manejo	7,8
Rapidez entre páginas	8,4
Rapidez al realizar búsquedas	8
Ambiente totalmente gráfico	10
Calidad del Sistema	8

Los resultados de la encuesta sobre la calidad del servicio de revisión de correspondencia interna nos indican que existe un 80% de aceptación del sistema, un grado de satisfacción de 8,8 en la escala del 1 al 10 siendo 10 el mejor grado de satisfacción, y calidad del sistema de 8 en la misma escala de 1 a 10; con lo cual podemos decir que el sistema si se ha satisfecho con lo requerido por los usuarios, es decir; ha mejorado la calidad del servicio de revisión de correspondencia interna al implementar la aplicación Web cero papeles.

4.2. Resultados Sobre Eficiencia en el Servicio de Revisión de Correspondencia Interna



Figura 8. Cantidad de Documentos que Iniciaron y Finalizaron el proceso de Revisión en una semana

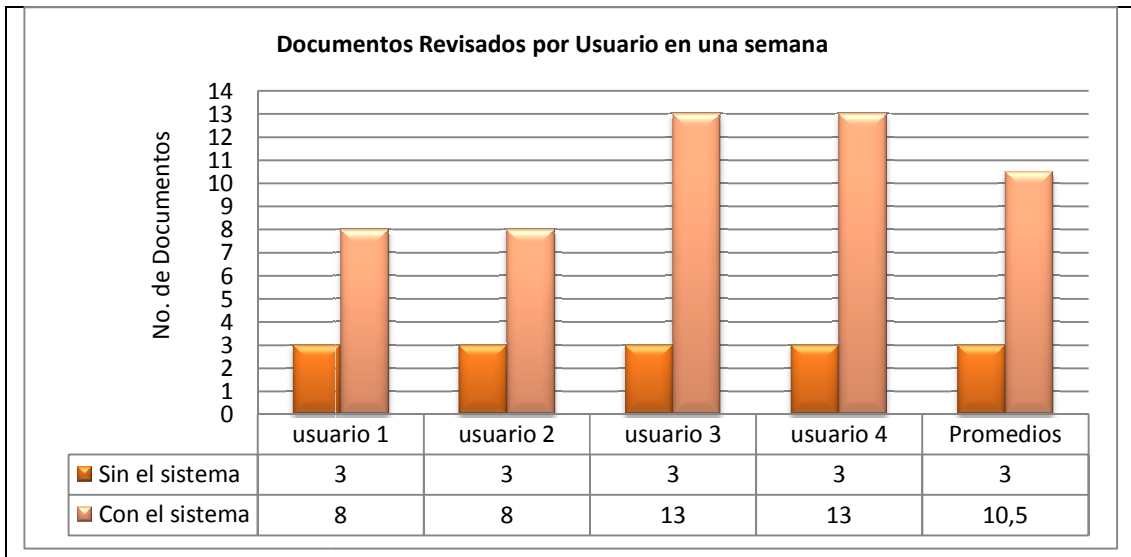


Figura 9. Cantidad de Documentos Revisados por un Usuario en una semana

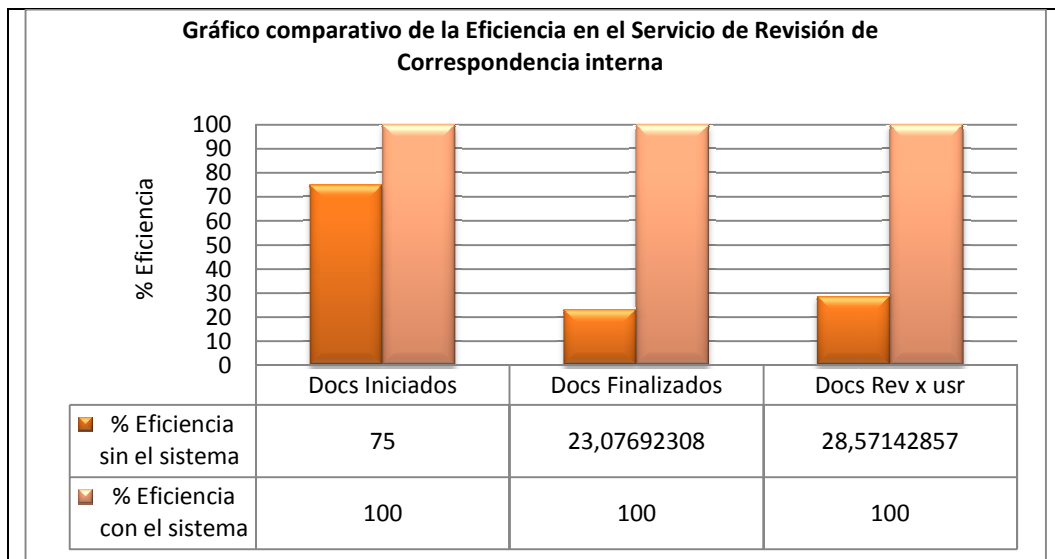


Figura 10. Comparación de la Eficiencia en el Servicio de Revisión de Correspondencia Interna

Los resultados de la encuesta sobre la eficiencia del servicio de revisión de correspondencia interna, indican que hay una mayor cantidad de documentos digitalizados y revisados desde el momento que la aplicación Web cero papeles fue implementada, según muestran las figuras 8 y 9; debido a que, usando el sistema; un documento se puede revisar con mayor velocidad, proporcionando con esto una mayor eficiencia al servicio de revisión de correspondencia interna, tal como se visualiza en el gráfico comparativo (figura 10), en donde se puede apreciar que la eficiencia de documentos digitalizados en una semana aumento un 25%, la eficiencia de documento completamente revisados en una semana aumento en casi un 77%, y la eficiencia de documentos revisados por un usuario en una semana aumento en casi un 71,5%. Se ha realizado el cálculo de la eficiencia usando la siguiente ecuación:

Ecuación (1)

5. TRABAJOS RELACIONADOS

Los análisis y resultados escritos en el presente artículo se relacionan con algunos trabajos de una manera parcial, debido a que existen pocos trabajos que hablen sobre cero papeles. El Proceso de Digitalización de Documentos de la Universidad de CORNELL que se encuentra en [4], y el artículo presentado en el Principado de Asturias llamado “Sistema de Gestión e Información de Archivos SIGIA” [5], se relacionan con el presente trabajo debido a que son procesos de digitalización de documentos que se tomaron como referencia para el proceso de digitalización de documentos de éste artículo. La tesis de la Escuela Politécnica Nacional, titulada “Diseño e Implementación de un Portal Web con Acceso a Datos para el Centro Educativo Paulo IV” de Christian Aguas Tusa, se relaciona con el presente artículo en el diseño de una Aplicación Web siguiendo la metodología OOHDM, debido a que los diagramas de las etapas uno, dos, tres y cinco fueron realizadas en base a lo investigado en ésta tesis. La tesis de la ESPE, titulada “Análisis, Diseño e Implementación de un Sistema Web – Multimedia con Asistentes de Ayuda para la Estimulación de Lenguaje y Agilidad Mental de Niños de 2 a 5 años” de Álvaro Villacis Cajas, se relaciona con el presente artículo en el diseño de ADVs de la aplicación Web cero papeles siguiendo la metodología OOHDM, debido a que los diagramas ADVs realizados en la etapa cuatro de la metodología fueron realizadas basado en el ejemplo investigado en ésta tesis. Existe una tesis de la Universidad Católica, titulada “Evaluación del Sistema Cero Papeles como Agente Dinamizador del Comercio Exterior

en la CAE en el Sector Importador de Equipos de Computación de Quito” de Karla Martínez y Luís Moyano, donde realizan un análisis completo sobre el sistema cero papeles SICE utilizado en la CAE, que debía aumentar el control de ingreso y salida de mercadería como parte del plan de automatización del comercio exterior, y que no lo cumplió; se relaciona en éste trabajo de una manera indirecta ya que se tomó como aprendizaje las razones del fracaso del sistema SICE para evitar repetirlo en el sistema cero papeles de el presente trabajo.

6. CONCLUSIONES Y TRABAJO FUTURO

La filosofía oficina cero papeles plantea la eliminación de la dependencia de documentos de papel mediante un proceso de digitalización de documentos que contiene una aplicación Web cero papeles que ayuda a almacenar los documentos en una base de datos para evitar la pérdida y el deterioro de documentos importantes, y dar más velocidad en el proceso de revisión de correspondencia de Banco COFIEC S.A con el objetivo de mejorar su calidad y eficiencia. Al finalizar el trabajo, se ha podido obtener como conclusiones, que al implementar el proceso de digitalización de documentos, conjuntamente con la respectiva aplicación Web cero papeles, se ha logrado satisfacer las necesidades y los requerimientos de los usuarios del sistema, aumentando la cantidad de correspondencia revisada, la velocidad en la revisión de un documento y la eficiencia del servicio de revisión de correspondencia, eliminando por completo la dependencia de documentos en papel. OOHDM es una metodología factible y eficaz para desarrollar aplicaciones Web, debido a que sigue una secuencia ordenada y entendible de pasos y diagramas de diseño con lenguaje UML, lo que permitió el desarrollo rápido del sistema, y facilitó la localización de los respectivos errores durante las pruebas los mismos que fueron corregidos rápidamente.

Como trabajo futuro se está analizando el rediseño de una segunda versión de la aplicación Web cero papeles, que tenga nuevas funcionalidades a mas de las ya existentes, tales como: el manejo de certificados digitales para la correspondencia digitalizada, un módulo de seguridad que permita al administrador operativo y al DBA el ingreso de nuevos usuarios desde el mismo sistema cero papeles sin tener que ingresar a la base de datos, y la revisión de documentos de correspondencia por medio del celular.

Referencias Bibliográficas

- [1].- Mujeres de Empresa. “El sueño de la oficina sin papeles”;
<http://www.mujeresdeempresa.com/ebusiness/ebusiness060203.shtml>
- [2].- Casanovas, I. *Gestión de Archivos Electrónicos*, Buenos Aires: Alfagrama Editores, Noviembre 2008, pp. 12
- [3].- Lamarca Lapuente, M. J. “Modelo OOHDM”; <http://www.hipertexto.info/documentos/oohdm.htm>
- [4].- Casanovas, I. *Gestión de Archivos Electrónicos*, Buenos Aires: Alfagrama Editores, Noviembre 2008, pp. 86 hasta 100
- [5].- Dirección General de Modernización del Principado de Asturias. (2007). Digitalización de Documentos. *Sistema de Gestión e Información de Archivos SIGIA*. [En línea], pp. 1 – 16. Disponible en: <http://www.digitalag.cl/Documentos/Norma%20Digitalizaci%C3%B3n%20Asturias.pdf>

Capacidad de Conmutación de los Equipos Activos para una Red LAN Corporativa

Ángel Chinchero Villacís

Maestría en Redes de Información y Conectividad, Escuela Politécnica del Ejército, Sangolquí, Ecuador
achinchero@sbs.gob.ecespe.edu.ec

RESUMEN: El documento establece un modelo de guía para calcular la capacidad de conmutación de los equipos activos para una red empresarial de alto rendimiento. Los cálculos se realizan en orden ascendente. Primero se calcula el número de usuarios y dispositivos proyectados en la red, considerando su crecimiento, esto define la cantidad y la capacidad de conmutación de los Switches de Acceso en cada piso, así como también el tipo de puertos de enlaces de backbone. Luego en función de los Switches de Acceso y de la cantidad de Servidores se calculan las capacidades de los Switches de Núcleo.

Palabras clave: Switch, Núcleo, Acceso, Conmutación, Velocidad, NAC.

ABSTRACT: This document provides a guide model to calculate switching capacity for the active equipments to a high performance enterprise networks. Calculations are performed in ascending order. First we calculate the user and devices projected number on the network, considering its growth, this define the quantity and the switching capacity of access switches on each floor, as well as the type of backbone links ports. Then in function of the access switches and the servers number are calculated the capacity of the Core Switches.

Keywords: Switch, core, Access, Conmutation, speed, NAC.

1. INTRODUCCIÓN

Uno de los principales factores que determinan el éxito del diseño de una red, es la habilidad de la red para proporcionar una satisfactoria interacción entre sus dispositivos interconectados, pues los usuarios juzgan la red por la rapidez de su capacidad de transmisión y la confiabilidad del servicio. Uno de los dispositivos de mayor relevancia que proveen esta característica son los switches de acceso, distribución y núcleo. Los switches resuelven los problemas de anchos de banda al segmentar un dominio de colisiones de una LAN, en pequeños dominios de colisiones

La extensa disponibilidad de la tecnología de switch de bajo costo tiene implicaciones para las redes de los edificios y el backbone de campus. Hay una demanda creciente para switches de backbone de alta densidad, con un número grande de puertos de alta velocidad, para enlazar grupos de trabajo individuales.

El presente artículo se enfoca en calcular las capacidades de conmutación que deben tener los Switches de Acceso, los Switches de Núcleo y el Backbone para una red LAN empresarial de alto rendimiento, acorde a las nuevas tecnologías existentes y que permita el funcionamiento de aplicaciones y servicios futuros. Para este cálculo, se ha considerado una empresa con las siguientes características:

- i. Edificio de dos torres. 16 pisos, incluido la planta baja, por cada torre.
- ii. El centro de cómputo de encuentra ubicado en el piso 3.
- iii. 80 usuarios de red actual, con un crecimiento máximo de usuarios del 10% por cada uno de los pisos.
- iv. Crecimiento máximo de dispositivos de red del 10 % por cada piso.

Para llevarlo a cabo se ha seguido el siguiente procedimiento: Primero se calcula la cantidad de usuarios y dispositivos de red proyectados en cada uno de los piso, para calcular la capacidad de los Switch de Acceso. Se calcula la capacidad que de los puertos de up-link para el backbone utilizando los métodos de diseño de las mejores prácticas de Cisco y el método basado en la fórmula de distribución de Poisson. La capacidad y cantidad de puertos de up-link requeridos para los Switches de Acceso y la cantidad y tipo de interfaces de red requeridas para los servidores determinan la capacidad que debe tener los Switches de Núcleo, considerando que la alternativa técnica es instalar dos Switches de Núcleo en modo de operación Activo-Activo.

El resto del artículo se ha organizado de la siguiente manera: La sección 2 describe la metodología utilizada para el cálculo de capacidades de conmutación. La sección 3 muestra la evaluación de resultaos obtenidos. Finalmente, la sección 4 explica las conclusiones obtenidas durante esta investigación y el trabajo futuro.

2. METODOLOGÍA

Para el cálculo las capacidades de conmutación de los equipos activos de la red de datos se utiliza el modelo jerárquico descrito en la Figura 1, que contempla: [4] [6] [8]

- i. Switches de Núcleo;
- ii. Switches de Distribución;
- iii. Switches de Acceso;
- iv. Sistema de gestión de los Switches;
- v. Sistema de Administración y Control de Accesos a la red de datos.

Para el caso de estudio, se tienen dos Switches de Núcleo en topología redundante, en modo de operación activo-activo, Switches de Acceso, Sistema de Administración y Control de Accesos a la red de datos, para contar con una solución altamente disponible, tolerante a fallas y de alto desempeño.

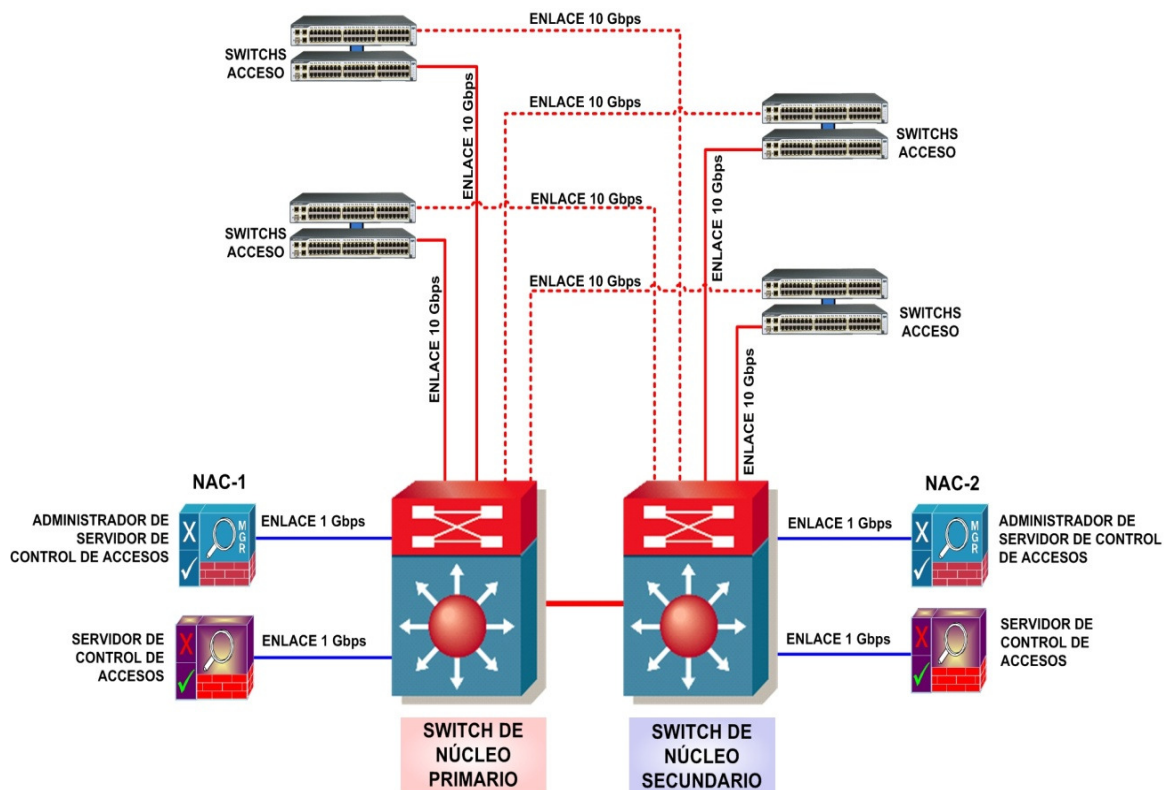


Figura 1. Modelo Jerárquico de diseño

2.1 Cálculo de la capacidad de conmutación de los switches de acceso

2.1.1 Cantidad de usuarios proyectados

Se considera un edificio de dos torres, con 16 pisos incluido la planta baja por torre. Cada piso tiene 80 usuarios de red cableada activos. El crecimiento es del 10 % para estaciones de usuarios de red cableada y del 10 % para otros dispositivos de red.

El número de Switches de Acceso y el número de puertos de red Ethernet requeridos en cada uno de los Switches, se determina en función de la cantidad de usuarios y dispositivos de red proyectados por piso, como indica la ecuación (1).

$$\text{Total de usuarios proyectados por piso} = \left(\text{Total de usuarios actuales} \right) + \left(\frac{10\% \text{ de crecimiento}}{\text{para usuarios cableados}} \right) + \left(\frac{10\% \text{ de crecimiento}}{\text{para dispositivos de red}} \right) \quad \text{ecuación (1)}$$

El número de Switches de Acceso requeridos por piso se calculan utilizando las ecuaciones (2), (3) y (4):

$$\text{Número de Switches de Acceso por piso} = \text{Entero superior} \left[\frac{\text{Total de puertos de red por piso}}{\text{Número de puertos de usuario final por Switch}} \right] \quad \text{ecuación (2)}$$

$$\text{Número de Switches de Acceso por piso} = \text{Entero superior} \left[\frac{\text{Total de puertos de red por piso}}{48 \text{ puertos de usuario final por Switch}} \right] \quad \text{ecuación (3)}$$

$$\text{Número de Switches de Acceso} = \text{Número de Switches de Acceso por piso} + \text{Número de Switches de Acceso de Respaldo} \quad \text{ecuación (4)}$$

Número de Switches de Acceso de Respaldo = 2 switches por motivos de contingencias ecuación (5)

TABLA 1. Cálculo de la cantidad de usuarios proyectados por piso y la cantidad de Switches de Acceso requeridos

cantidad de usuarios proyectados y número de switches de acceso						
Piso	usuarios activos por piso	usuarios activos por rack	ubicación actual de racks	usuarios y dispositivos proyectados por piso	total de puertos de red por rack	número de switches de 48 puertos por rack
Planta Baja	80	80	Planta Baja	96	96	2
Piso 1	80	80	Piso 1	96	96	2
Piso 2	80	80	Piso 2	96	96	2
Piso 3	80	80	Piso 3	96	96	2
Piso 4	80	80	Piso 4	96	96	2
Piso 5	80	80	Piso 5	96	96	2
Piso 6	80	80	Piso 6	96	96	2
Piso 7	80	80	Piso 7	96	96	2
Piso 8	80	80	Piso 8	96	96	2
Piso 9	80	80	Piso 9	96	96	2
Piso 10	80	80	Piso 10	96	96	2
Piso 11	80	80	Piso 11	96	96	2
Piso 12	80	80	Piso 12	96	96	2
Piso 13	80	80	Piso 13	96	96	2
Piso 14	80	80	Piso 14	96	96	2
Piso 15	80	80	Piso 15	96	96	2
Piso 16	80	80	Piso 16	96	96	2
Respaldo						2 equipos de backup
TOTAL	1360	1360		1632	1632	36 Witches

2.1.2 Velocidad y tipo de puertos para los usuarios en los Switches de Acceso

La velocidad y tipo de puertos de usuario final, para los Switches de Acceso, se determina en función del tráfico actual y futuro generado por cada usuario.

2.1.2.1 Estimación de la demanda de tráfico actual y futuro por cada usuario

Para calcular demanda de tráfico actual y futuro por cada usuario de red, se ha realizado un análisis estimado de las diferentes aplicaciones y servicios que puede usar un usuario, como se indica en la Tabla 2.

Del análisis se determina que el ancho de banda instantáneo pico requerido por usuario actualmente es de 38,9 Mbps. En el futuro se requerirá tener mayores anchos de banda para incluir funciones como audio, video, tráfico de administración y control de tráfico multicapa directamente sobre el puerto del usuario, y se estima que el ancho de banda pico llegará alcanzar los 95,9 Mbps por usuario, como indica la Tabla 2.

TABLA 2. Estimación de la demanda de tráfico pico actual y futuro por cada usuario

Requerimiento de ancho de banda pico actual por cada usuario	Capacidad requerida
Navegación	0,20 Mbps
Actualizaciones en línea de sistemas operativos	0,20 Mbps
Actualizaciones en línea de sistemas de seguridad	0,20 Mbps
Acceso a aplicación 1	2,00 Mbps
Acceso a aplicación 2	2,00 Mbps
Acceso a aplicación 3	2,00 Mbps
Acceso al servidor de correo electrónico.	1,00 Mbps
Acceso a los buzones de voz desde el computador a través de cliente de correo electrónico	1,00 Mbps
Transferencia de archivos de datos entre funcionarios	25,00 Mbps
Descarga de videos institucionales,	5,00 Mbps
Telefonía IP	0,06 Mbps
Multi-conferencia IP, para 4 sesiones	0,24 Mbps
Subtotal actual pico por usuario	38,9 Mbps
Requerimiento de ancho de banda futuro por cada usuario	Capacidad requerida
Ambientes de colaboración, presentaciones, video, mensajería, audio	5,00 Mbps
Video llamadas	1,00 Mbps
Video conferencias en el escritorio	1,00 Mbps
Aplicaciones futuras adicionales	
Administración y control de tráfico multicapa, sobre el puerto del usuario, etc	50,00 Mbps
Subtotal adicional pico futuro por usuario	57,00 Mbps
Total de ancho de banda pico futuro requerido por usuario	95,9 Mbps

2.1.2.2 Definición de la velocidad de los puertos para los usuarios en los Switches de Acceso

Para evitar posibles encolamientos y saturación en los puertos de red, estos deberían tener mínimo el doble de la capacidad calculada (**191,8 Mbps \approx 200 Mbps**). Por tal motivo los puertos de red de los Switches de Acceso deben ser de 1 Gbps, al no existir de 200 Mbps, lo cual también permitirá estandarizar con los puertos de red de 1 Gbps de las estaciones de trabajo actual y futura.

2.1.3 Velocidad y tipo de puertos de enlace de Up-link para conexión al Backbone

Para calcular la velocidad del puerto de enlace al backbone (up-link), se utilizó dos métodos:

2.1.3.1 Método de la fórmula de distribución de Poisson

Utilizando la fórmula de distribución de Poisson, se calcula la probabilidad de los arribos al puerto de up-link, en función de la ecuación (6). [5]

$$\text{Probabilidad de arribos}(r) = P(r) = \frac{e^{-\lambda}(\lambda)^r}{r!} \quad \text{ecuación (6)}$$

Donde:

$P(r)$ es la probabilidad de los arribos al puerto de up-link.
 r es el número de arribos al puerto de up-link.
 λ es la velocidad promedio de arribos al puerto de up-link.

El resultado de la ecuación (6), se utiliza para calcular la velocidad del enlace de up-link del Switch de Acceso, mediante la ecuación (7) propuesta:

$$\text{Velocidad de puerto de up-link} \geq (\text{Número de puertos del Switch de Acceso}) * \left(\frac{\text{velocidad de los puertos en half duplex}}{\text{en half duplex}} \right) * P(r) \quad \text{ecuación (7)}$$

Para el caso de un switch de 48 puertos, el número de arribos simultáneos r es 48, la velocidad promedio de arribo es 48 arribos por unidad de tiempo y la probabilidad de arribo al puerto de up-link utilizando la ecuación (6) será:

$$P(48) = \frac{e^{-48}(48)^{48}}{48!} = 0,05748$$

La velocidad del puerto de up-link utilizando la ecuación (7) será:

$$\text{Velocidad de puerto de up-link} \geq (48 \text{ puertos}) * (1 \text{ Gbps}) * (0,05748) = 2,759 \text{ Gbps}$$

De los resultados anteriores, se determina que la velocidad del puerto de up-link, utilizando la fórmula de Poisson debe ser mayor a 2,759 Gbps.

2.1.3.2 Método de las mejores prácticas de diseño de Cisco

Las Mejores Prácticas de Diseño de Cisco para los Switches de Acceso se basan en los niveles de sobreescripción de los equipos, que es la cantidad de puertos de usuario final que funcionando a velocidades wire-rate que pueden transmitir de manera simultánea a través de un enlace de backbone o up-link. [3]. Los niveles de sobreescripción, considerados en los diseños de Cisco son:

- i. De 1:1 a 20:1, para las redes con un nivel de tráfico bajo.
- ii. De 10:1 a 20:1, para redes con un nivel de tráfico medio-bajo, que utilizan la mayor parte del tiempo aplicaciones típicas.
- iii. De 4:1 a 12:1, para las redes empresariales, con un nivel de tráfico medio-alto, que utilizan aplicaciones típicas todo el tiempo y también aplicaciones especiales temporales que requieren de alto ancho de banda.
- iv. De 5:1 a 10:1, para las redes empresariales con tráfico de servidores virtuales.
- v. De 1:1 a 4:1, para las redes de data centers, con nivel de tráfico alto, que utilizan aplicaciones especiales todo el tiempo que requieren de un alto ancho de banda.

La red analizada, es una red con un nivel de tráfico medio-alto, cuya relación de sobreescripción debe estar en el rango de 10:1 a 20:1. Por lo tanto se debe cumplir la relación indicada en la ecuación (8):

$$10 \leq \frac{\text{Número total de puertos de usuario del Switch de Acceso} * \text{Velocidad de los puertos}}{\text{Velocidad del puerto up-link de enlace para conexión al Backbone}} \leq 20 \quad \text{ecuación (8)}$$

Mientras más baja es la relación de sobre suscripción, mejor es el desempeño y la capacidad de crecimiento de los Switches.

Para un Switch de Acceso de 48 puertos Ethernet de 1 Gbps, full dúplex, considerando que todos sus puertos transmiten a su máxima velocidad, en forma simultánea, aplicando la ecuación (8) se requiere transmitir a través del puerto up-link de enlace al backbone un total de 96 Gbps.

$$10 \leq \frac{48 \text{ puertos} * 2 \text{ Gbps}}{\text{Velocidad del puerto up-link de enlace para conexión al Backbone}} \leq 20$$

$$10 \leq \frac{96 \text{ Gbps}}{\text{Velocidad del puerto up-link de enlace para conexión al Backbone}} \leq 20$$

Por lo tanto la velocidad del puerto de up-link debe ser:

$$\frac{96 \text{ Gbps}}{20} \leq \text{Velocidad del puerto up - link de enlace para conexión al backbone} \leq \frac{96 \text{ Gbps}}{10}$$

$$4.8 \text{ Gbps} \leq \text{Velocidad del puerto up - link de enlace para conexión al backbone} \leq 9,6 \text{ Gbps}$$

Para tener un mejor desempeño y capacidad de crecimiento del switch se deben utilizar puertos de enlace Ethernet de 10 Gbps de fibra óptica, para la conexión al backbone, con lo cual la relación de suscripción es:

$$\text{Relación de sobreescripción del Switch de Acceso} = \frac{96 \text{ Gbps}}{10 \text{ Gbps}} = 9,6$$

De los resultados obtenidos utilizando los métodos de Poisson y de Cisco, se concluye que tanto el Switch de Acceso Principal como el Switch de Acceso Secundario deben tener mínimo dos interfaces Ethernet de 10 Gbps, cada uno:

- i. Una interface para la conexión al backbone de 10 Gbps;
- ii. Una interface para la conexión entre switches en el mismo piso, para formar cascada o anillo local de 10 Gbps.

2.2 Capacidad de conmutación de los Switches de Acceso.

La capacidad de conmutación requerida para los Switches de Acceso, se calcula considerando la transmisión simultánea full dúplex de todos sus puertos [1], utilizando la ecuación (9) y cuyos resultados se indican en la Tabla 3.

$$\text{Capacidad de conmutación del Switch de Acces} = \left(\begin{matrix} \text{Número de} \\ \text{puertos de} \\ \text{usuario final} \end{matrix} \right) * 2 * \left(\begin{matrix} \text{Velocidad} \\ \text{del puerto de} \\ \text{usuario final} \end{matrix} \right) + \left(\begin{matrix} \text{Número} \\ \text{de puertos} \\ \text{de up-link} \end{matrix} \right) * 2 * \left(\begin{matrix} \text{Velocidad} \\ \text{del puerto} \\ \text{de up-link} \end{matrix} \right) \text{ ecuación (9)}$$

$$\left(\begin{matrix} \text{Velocidad} \\ \text{del puerto de} \\ \text{usuario final} \end{matrix} \right) = 1 \text{ Gbps, para conexión con las estaciones de trabajo}$$

$$\left(\begin{matrix} \text{Velocidad} \\ \text{del puerto de} \\ \text{usuario final} \end{matrix} \right) = 10 \text{ Gbps, para conexión con el Switch de Núcleo}$$

TABLA 3. Capacidad de conmutación de un Switch de Acceso

Capacidad de conmutación del switch de acceso		
Tipo de puertos	Número de puertos	Capacidad de conmutación en gbps
Puertos de 1 Gbps	48	96
Puertos de 10 Gbps	2	40
	Capacidad de conmutación	136

2.3 Cálculo de la capacidad de conmutación de los switches de núcleo

Para calcular la capacidad de conmutación de los Switches de Núcleo se consideran los siguientes parámetros:

2.3.1 Tipo de topología o esquema de conexión de los Switches de Núcleo

Debido a la cantidad de usuarios existentes y proyectados en la red de datos, es necesario instalar dos Switches de Núcleo conectados en topología redundante en modo operación activo-activo, como se indica en la Figura 1.

2.3.2 Tipo de procesamiento de los Switches de Núcleo

Cada una de las controladoras de los Switches de Núcleo debe tener procesamiento distribuido e independiente, para tener un mayor rendimiento y se integran a través de backplanes y buses de alta velocidad.

2.3.3 Cantidad y tipo de puertos para los Switches de Núcleo

Para calcular la cantidad y el tipo de puertos de red Ethernet requeridos para cada uno de los Switches de Núcleo, se calculan utilizando las ecuaciones (10), (11) y (12), cuyos resultados se indican en la Tabla 4. Se considera los siguientes puntos:

- i. Los enlaces de fibra óptica, multi-modo a 10 Gbps con los Switches de Acceso;
- ii. Las conexiones de fibra óptica, multi-modo a 10 Gbps con los servidores;
- iii. La interconexión de fibra óptica, multi-modo a 10 Gbps entre Switches de Núcleo;
- iv. Las conexiones de fibra óptica y RJ45 a 1 Gbps con los servidores y los equipos de comunicaciones del centro de cómputo;
- v. Respaldo de cada tipo de puertos;
- vi. Los puertos se distribuyen de forma balanceada para los dos Switches de Núcleo.

$$\begin{aligned} \text{Número de puertos de fibra óptica de 10 Gbps requeridos para cada Switch de Núcleo} &= \left(\begin{array}{l} \text{Número de puertos} \\ \text{de fibra de 10 Gbps} \\ \text{para enlaces con los} \\ \text{Switches de Acceso} \end{array} \right) + \left(\begin{array}{l} \text{Número de puertos} \\ \text{de fibra de 10 Gbps} \\ \text{para los servidores} \end{array} \right) + \left(\begin{array}{l} \text{Número de puertos de} \\ \text{fibra de 10 Gbps para} \\ \text{interconexión entre} \\ \text{Switches de Núcleo} \end{array} \right) + \\ &\left(\begin{array}{l} \text{Número de puertos} \\ \text{de fibra de 10 Gbps} \\ \text{de respaldo} \end{array} \right) \end{aligned} \quad \text{ecuación (10)}$$

$$\begin{aligned} \text{Número de puertos de fibra óptica de 1 Gbps requeridos para cada Switch de Núcleo} &= \left(\begin{array}{l} \text{Número de puertos} \\ \text{de fibra de 1 Gbps} \\ \text{para enlaces con los} \\ \text{Switches de Acceso} \end{array} \right) + \left(\begin{array}{l} \text{Número de puertos} \\ \text{de fibra de 1 Gbps} \\ \text{para los servidores} \end{array} \right) + \left(\begin{array}{l} \text{Número de puertos} \\ \text{de fibra de 1 Gbps} \\ \text{de respaldo} \end{array} \right) \end{aligned} \quad \text{ecuación (11)}$$

$$\begin{aligned} \text{Número de puertos RJ45 de 1 Gbps requeridos para cada Switch de Núcleo} &= \left(\begin{array}{l} \text{Número de puertos RJ45 de} \\ \text{de 1 Gbps para los servidores} \end{array} \right) \end{aligned} \quad \text{ecuación (12)}$$

Para este caso, en la ecuación (10) se tiene que:

$$\left(\begin{array}{l} \text{Número de puertos de fibra} \\ \text{de 1 Gbps para enlaces} \\ \text{con los Switches de Acceso} \end{array} \right) = 0$$

2.4 Capacidad de conmutación de los Switches de Núcleo

La capacidad de conmutación del Switch de Núcleo Primario y Secundario, se calcula utilizando las ecuaciones (13), (14), (15), (16) y (17), cuyos resultados se indican en las Tablas 5 y 7. Se debe considerar:

- i. La cantidad de puertos de fibra óptica, multi-modo a 10 Gbps;
- ii. La cantidad de puertos de fibra óptica, multi-modo a 1 Gbps;
- iii. La cantidad de puertos RJ45, de 1 Gbps;
- iv. La cantidad de slots de funciones adicionales;
- v. El porcentaje de slots vacíos para crecimiento;
- vi. La capacidad total de los puertos para transmisión en full dúplex;
- vii. Las controladoras adicionales tienen una capacidad 80 Gbps (según equipos Cisco).

$$\begin{array}{l} \text{Capacidad de conmutación} \\ \text{para los puertos de fibra a} \\ \text{10 Gbps en full duplex} \end{array} = 2 * \left(\begin{array}{l} \text{Número de puertos} \\ \text{de fibra óptica de} \\ \text{10 Gbps requeridos} \end{array} \right) * (10 \text{ Gbps}) \quad \text{ecuación (13)}$$

$$\begin{array}{l} \text{Capacidad de conmutación} \\ \text{para los puertos de fibra a} \\ \text{1 Gbps en full duplex} \end{array} = 2 * \left(\begin{array}{l} \text{Número de puertos} \\ \text{de fibra óptica de} \\ \text{1 Gbps requeridos} \end{array} \right) * (1 \text{ Gbps}) \quad \text{ecuación (14)}$$

$$\begin{array}{l} \text{Capacidad de conmutación} \\ \text{para los puertos RJ45 a} \\ \text{1 Gbps en full duplex} \end{array} = 2 * \left(\begin{array}{l} \text{Número de puertos RJ45} \\ \text{de 1 Gbps requeridos} \end{array} \right) * (1 \text{ Gbps}) \quad \text{ecuación (15)}$$

$$\begin{array}{l} \text{Capacidad total de} \\ \text{conmutación para los} \\ \text{puertos de red para} \\ \text{cada Switch de Núcleo} \end{array} = \left(\begin{array}{l} \text{Capacidad de} \\ \text{conmutación para} \\ \text{los puertos de fibra a} \\ \text{10 Gbps en full duplex} \end{array} \right) + \left(\begin{array}{l} \text{Capacidad de} \\ \text{conmutación para} \\ \text{los puertos de fibra a} \\ \text{1 Gbps en full duplex} \end{array} \right) + \left(\begin{array}{l} \text{Capacidad de} \\ \text{conmutación para} \\ \text{los puertos RJ45 a} \\ \text{1 Gbps en full duplex} \end{array} \right) \quad \text{ecuación (16)}$$

$$\begin{array}{l} \text{Capacidad total} \\ \text{de conmutación} \\ \text{de cada Switch} \\ \text{de Núcleo} \end{array} = \left(\begin{array}{l} \text{Capacidad de} \\ \text{conmutación} \\ \text{para los} \\ \text{puertos de red} \end{array} \right) + \left(\begin{array}{l} \text{Capacidad de} \\ \text{conmutación para} \\ \text{la controladora} \\ \text{de supervisión} \end{array} \right) + \left(\begin{array}{l} \text{Capacidad de} \\ \text{conmutación para} \\ \text{las funciones} \\ \text{especiales} \end{array} \right) + \left(\begin{array}{l} \text{Capacidad de} \\ \text{conmutación para} \\ \text{los slots vacíos} \\ \text{de crecimiento} \end{array} \right) \quad \text{ecuación (17)}$$

$$\left(\begin{array}{l} \text{Capacidad de} \\ \text{conmutación para} \\ \text{la controladora} \\ \text{de supervisión} \end{array} \right) = 80 \text{ Gbps}$$

$$\left(\begin{array}{l} \text{Capacidad de} \\ \text{conmutación para} \\ \text{las funciones} \\ \text{especiales} \end{array} \right) = 80 \text{ Gbps}, \text{ como por ejemplo Firewall, IPS.}$$

$$\left(\begin{array}{l} \text{Capacidad de} \\ \text{conmutación para} \\ \text{los slots vacíos} \\ \text{de crecimiento} \end{array} \right) = 80 \text{ Gbps}$$

TABLA 4. Cálculo de la cantidad y tipo de puertos de los Switches de Núcleo para funcionar con el backbone de 10 Gbps

Pisos	Switches de acceso	Switch de núcleo primario con el backbone a 10 gbps				Switch de núcleo secundario con el backbone a 10 gbps			
		Puertos f.o. a 10 gbps backbone	Puertos f.o. a 10 gbps servidores	Puertos f.o. a 1 gbps	Puertos rj45 1 gbps	Puertos f.o. a 10 gbps backbone	Puertos f.o. a 10 gbps servidores	Puertos f.o. a 1 gbps	Puertos rj45 1 gbps
Piso 16	2	1							
Piso 15	2	1							
Piso 14	2	1							
Piso 13	2	1							
Piso 12	2	1							
piso 11	2	1							
piso 10	2	1							
Piso 9	2	1							
Piso 8	2	1							
Piso 7	2					1			
Piso 6	2					1			
Piso 5	2					1			
Piso 4	2					1			
Piso 3	2		5	5	24	1	5	5	24
Piso 2	2					1			
Piso 1	2					1			
Piso pb	2					1			
Piso sl	2					1			
Inteconexión entre switches		4				4			
Respaldo		3	3	3		3	3	3	
TOTAL		16	8	8	24	16	8	8	24

2.5 Cantidad y tipo de controladoras de puertos de red para cada Switch de Núcleo

La cantidad y tipo de controladoras de puertos de red, para cada uno de los Switches de Núcleo, se calculan utilizando las ecuaciones (18), (19) y (20), cuyos resultados se indica en la Tabla 4.

Número de controladoras de puertos de fibra de 10 Gbps para cada Switch de Núcleo =

$$\text{Entero superior de } \left[\frac{\text{Número de puertos de fibra óptica de 10 Gbps requeridos para cada Switch de Núcleo}}{\text{Número de puertos soportados por cada controladora}} \right] \quad \text{ecuación (18)}$$

$$\text{Número de controladoras de puertos de fibra de 1 Gbps para cada Switch de Núcleo} = \text{Entero superior de } \left[\frac{\text{Número de puertos de fibra óptica de 1 Gbps requeridos para cada Switch de Núcleo}}{\text{Número de puertos soportados por cada controladora}} \right] \quad \text{ecuación (19)}$$

$$\text{Número de controladoras de puertos RJ45 de 1 Gbps para cada Switch de Núcleo} = \text{Entero superior de} \left[\frac{\left(\text{Número de puertos RJ45 de 1 Gbps requeridos para cada Switch de Núcleo} \right)}{\left(\text{Número de puertos soportados por cada controladora} \right)} \right] \quad \text{ecuación (20)}$$

2.6 Número de slots para los chasis de cada Switch de Núcleo

Considerando que cada controladora utiliza un slot, el cálculo del número de slots para cada uno de los chasis requeridos para cada uno de los Switch de Núcleo, se realiza utilizando las ecuaciones (21), (22) y (23), cuyos resultados se indican en la Tabla 4 para el Switch de Núcleo Primario y Secundario.

$$\text{Número de slots ocupados en cada Switch de Núcleo} = \left(\text{Número de slots para la controladora de supervisión} \right) + \left(\text{Número de slots para las controladoras de puertos de fibra óptica de 10 Gbps} \right) + \left(\text{Número de slots para las controladoras de puertos de fibra óptica de 1 Gbps} \right) + \left(\text{Número de slots para las controladoras de puertos RJ45 de 1 Gbps} \right) + \left(\text{Número de slots para las controladoras de funciones especiales} \right) \quad \text{ecuación (21)}$$

$$\text{Número de slots vacíos para crecimiento futuro en cada Switch de Núcleo} = \text{Entero superior de} \left[\left(\frac{\% \text{ de crecimiento futuro}}{1 - \% \text{ de crecimiento futuro}} \right) * \left(\text{Número de slots ocupados en cada Switch de Núcleo} \right) \right] \quad \text{ecuación (22)}$$

$$\% \text{ de crecimiento futuro} = 20\%$$

El porcentaje de crecimiento futuro se considera del 20% en cada Switch de Núcleo.

$$\text{Número total de slots de cada Switch de Núcleo} = \left(\text{Número de slots ocupados en cada Switch de Núcleo} \right) + \left(\text{Número de slots vacíos para crecimiento futuro en cada Switch de Núcleo} \right) \quad \text{ecuación (23)}$$

3. EVALUACION DE RESULTADOS

A los resultados obtenidos en la Tabla 3 y en la Tabla 5, para las capacidades de conmutación de los Switches de Acceso y los Switches de Núcleo respectivamente, se los puede redondear valores matemáticos, estableciendo que las capacidades de conmutación requeridas son de 140 Gbps para los Switches de Acceso y de 950 Gbps para los Switches de Núcleo.

El backbone calculado, es tipo estrella de 10 Gbps por cada uno de los enlaces desde los Switches de Núcleo hacia los Switches de Acceso de los pisos.

Se determinó matemáticamente que la capacidad conmutación de los Switches de Acceso de 48 puertos requeridos, debe ser de mínimo 140 Gbps, lo cual está en función de la cantidad de usuarios y dispositivos terminales que van a ser conectados.

Se dimensionó además en forma matemática la capacidad de los enlaces de backbone entre los Switches de Acceso y los Switches de Núcleo, utilizando los métodos de las mejoras prácticas de Cisco y de la Función de Distribución de Probabilidades Posisson, Los dos métodos presentan resultados coincidentes y coherentes. Los resultados indican que los enlaces de up-link de los equipos activos deben ser de 10 Gbps, lo cual permite tener gran capacidad en cuanto a anchos de banda y velocidades de acceso.

Adicionalmente, Se determinó en forma matemática que la capacidad de conmutación, de cada uno de los Switches de Núcleo debe ser mínimo 950 Gbps, y está en función de la cantidad de enlaces con los Switches de Acceso, la cantidad y tipo de puertos requeridos para los servidores del centro de cómputo, el factor de crecimiento futuro. La capacidad de conmutación total determina el número de Switch Fabric que debe ir instalado en cada uno de los Switches de Núcleo.

A continuación se muestran las Tablas 5,6 y 7 algunos resultados derivados de esta investigación:

TABLA 5. Cálculo de la capacidad del Switch de Núcleo Primario y Secundario, para funcionar con el backbone de 10 Gbps

Capacidad del switch de núcleo primario y secundario con el backbone a 10 gbps					
Requisito	Número de puertos y controladoras		Capacidad de conmutación simétrica gbps	Número de puertos por controladora	Número de slots del chasis
Puertos de 10 gbps fo	24	Puertos de fibra de 10 Gbps	480	16	2
Puertos de 1 gbps fo	8	Puertos de fibra de 1 Gbps	16	24	1
Puertos rj 45 1 gbps	24	Puertos de fibra de 1 Gbps	48	48	1
Supervisora	1	80 Gbps	80		1
Reservados para funciones adicionales	2	80 Gbps	160		2
% de slots vacíos para crecimiento	20 %	2	80 Gbps		2
Capacidad total de conmutación por switch			944 Gbps	Total de slots	9

TABLA 6. Resumen de Capacidades de los Switches de Acceso

Requisito	Switch de acceso
Cantidad de Switches de Acceso	36
Capacidad de conmutación mínima para cada Switch de Acceso	140 Gbps
Número de Puertos de F.O a 10 Gbps, multimodo, de up-link, para cada Switch de Acceso	1
Número de Puertos RJ45, de 1 Gbps	48

TABLA 7. Resumen de Capacidades de los Switches de Núcleo

Requisito	Switch de núcleo primario	Switch de núcleo secundario
Tipo de arquitectura	Modular	Modular
Número de slots, mínimo	9	9
Capacidad de conmutación mínima	950 Gbps	950 Gbps
Supervisora	1	1
Número de Puertos de F.O a 10 Gbps, multi-modo	24	24
Número de Puertos de F.O a 1Gbps, multi-modo	8	8
Número de Puertos RJ45, de 1 Gbps	24	24
Modo de operación	Activo-activo	Activo-activo

4. CONCLUSIONES Y TRABAJO FUTURO

Se determinó matemáticamente que la capacidad conmutación de los Switches de Acceso de 48 puertos requeridos, debe ser de mínimo 140 Gbps, lo cual está en función de la cantidad de usuarios y dispositivos terminales que van a ser conectados. Se dimensionó además en forma matemática la capacidad de los enlaces de backbone entre los Switches de Acceso y los Switches de Núcleo, utilizando los métodos de las mejoras prácticas de Cisco y de la Función de Distribución de Probabilidades Poisson, Los dos métodos presentan resultados coincidentes y coherentes. Los resultados indican que los enlaces de up-link de los equipos activos deben ser de 10 Gbps, lo cual permite tener gran capacidad en cuanto a anchos de banda y velocidades de acceso. Esto permite a la red de datos evitar riesgos por obsolescencia tecnológica en vista de que permite el funcionamiento de servicios que aparezcan en el futuro. Adicionalmente, Se determinó en forma matemática que la capacidad de conmutación, de cada uno de los Switches de Núcleo debe ser mínimo 950 Gbps, y está en función de la cantidad de enlaces con los Switches de Acceso, la cantidad y tipo de puertos requeridos para los servidores del centro de cómputo, el factor de crecimiento futuro. La capacidad de conmutación total determina el número de Switch Fabric que debe ir instalado en cada uno de los Switches de Núcleo.

Como trabajo futuro, se elaborará las especificaciones técnicas de una red auto-defendible de alto rendimiento, que incluya funciones de seguridad, para el análisis, la administración y el control del tráfico multicapa, para controlar a los usuarios y dispositivos de red directamente en los puertos en los cuales se conectan.

Referencias Bibliográficas

- [1] Chao, J. Wiley & Sons (2007). High Performance Switches and Routers. New Jersey. Capítulo 5. Páginas 176-206.
- [2] Chinchero, A. (2011) “Análisis y diseño de una red segura convergente de alto rendimiento para la oficina matriz Quito de la Superintendencia de Bancos y Seguros”. Proyecto de grado para la obtención del título de Maestría en Redes de Información y Conectividad, ESPE. Quito, Ecuador.
- [3] Cisco. Architecture brief. Using Cisco Catalyst 6500 and Cisco Nexus 7000 Series Switching Technology in Data Center Networks. [Fecha de consulta: Diciembre 2010]. Página 4. Disponible en https://www.cisco.com/en/US/prod/collateral/switches/ps9441/ps9402/ps9512/White_Paper_C17-449427.pdf
- [4] Desai, Nimish. Cisco Networkers (2009). Advance enterprise campus high availability. Barcelona.
- [5] Held, G. Wiley & Sons (2001). Data Communications Networking Devices. New York. Páginas 826-828.
- [6] Montañez, Mark. Cisco Networkers (2009). Multilayer campus architecture and design principles. Barcelona.
- [7] Nohre, H. Cisco Networkers (2009). Deploying Cisco NAC Appliance. Barcelona.
- [8] Nortel. (2009). Medium campus technical solution guide. Capítulo 1-2.

Marco de Referencia para la Gestión de Proyectos de Desarrollo de Software

Sandra Chávez

Dirección de Postgrado, Escuela Politécnica del Ejército, Quito, Ecuador
paola_chavezp@yahoo.com

RESUMEN: El presente proyecto tiene como objetivo plantear un marco de referencia para la gestión de proyectos de desarrollo de software para el Grupo Metro, con el fin de eliminar los problemas que actualmente se presentan en el área de sistemas al momento de realizar un proyecto de software. Para alcanzar este objetivo, primero se identifica los problemas que se presentan con mayor frecuencia y tienen un mayor impacto en el desarrollo de un proyecto como: la falta de planificación, tiempos de desarrollo elevados, cambios de especificaciones, entre otros; basados en proyectos anteriormente realizados por el área de sistemas. Segundo, se realiza un estudio comparativo de los diferentes estándares y mejores prácticas como: CMMI, COBIT, ITIL y PMBOK, para determinar el estándar que soluciona o propone un conjunto de procesos para corregir la mayor parte de problemas identificados en la primera etapa. Finalmente, se realiza el planteamiento del marco de referencia para la gestión de proyectos de desarrollo de software, adaptado a las necesidades del área de sistemas del Grupo Metro y basados en el estándar de PMBOK, que se seleccionó en la segunda etapa como el estándar que cubre la mayor parte de problemas y los que inciden sobre todo en el fracaso de los proyectos de software por parte del área de sistemas. El presente planteamiento comprende los diferentes procesos que se deben llevar a cabo por parte del área de sistemas, para culminar un proyecto dentro del tiempo, presupuesto y recursos humanos inicialmente establecidos, independientemente del tamaño del proyecto.

Palabras clave: Gestión de proyectos, desarrollo de software, marco de referencia, estándares, mejores prácticas.

ABSTRACT: This project proposes a framework for managing software development projects for the Grupo Metro, in order to eliminate the problems currently present in the systems area when a software project is developed. To achieve this goal, firstly we identify the problems that occur more frequently and have a greater impact on the development of a project such as: lack of planning, high development time, changes in specifications, among others, based on earlier projects made by the software development area. Secondly, we accomplished a comparative study of different standards and best practices such as CMMI, COBIT, ITIL and PMBOK, to determine the standard that addresses or proposes a set of processes to correct most problems identified in the first stage. Finally, the framework is proposed, tailored to the needs of the Grupo Metro and based on the PMBOK standard, which was selected in the second stage as the standard cover most of problems and affecting mainly the failure of software projects by the software development area. This approach involves different processes to be carried out by the area of systems, to complete a project on time, budget and human resources initially established, regardless of project size.

Key Words: Project management, software development, framework, standards, best practices.

1. INTRODUCCIÓN

El Grupo Metro está dentro del mercado de las Telecomunicaciones alrededor de 35 años, tiempo en el cual se ha posesionado entre las 500 mayores empresas dentro del Ecuador. El crecimiento acelerado del grupo en los últimos años, ha conllevado a que la demanda de nuevos proyectos de desarrollo de software se incremente notablemente en el área de sistemas y surja la necesidad de establecer un marco de referencia para la gestión de proyectos de desarrollo de software.

Los proyectos que se desarrollan dentro del área de sistemas actualmente, tienen serios retrasos en las fechas de entrega, normalmente se supera el presupuesto inicial, no existe planificación, no se llega a alcanzar los objetivos iniciales, no se determina fechas de inicio o de entrega del proyecto, existe una real insatisfacción por parte de los usuarios finales, los recursos son sobrecargados y otros subutilizados, hay duplicación de esfuerzos y frecuentemente los levantamientos de requerimientos son erróneos.

Para tratar de minimizar estos problemas se creó una herramienta para el ingreso de requerimientos, el mismo que permite de manera breve ingresar las solicitudes de los usuarios finales. Esta herramienta permite conocer el usuario y fecha de ingreso del requerimiento, pero no permite realizar una gestión sobre dicho requerimiento.

El presente proyecto plantea un marco de referencia para la gestión de proyectos de desarrollo de software, adaptado a la estructura y necesidades del Grupo Metro y con el fin de eliminar los problemas que se presentan durante la vida de un proyecto que se desarrolla dentro del área de sistemas.

Para cumplir dicho propósito, este proyecto se encuentra dividido en tres etapas. Dentro de la primera etapa, se realiza el análisis del área de sistemas; es decir, sus principales sistemas y los problemas que se presentaron al momento de la realización de dichos proyectos. En la siguiente etapa se procede con el estudio de los estándares y mejores prácticas como: CMMI, COBIT, ITIL y PMBOK; con el fin de realizar un cuadro comparativo y seleccionar el mejor estándar o mejores prácticas que cubre la mayor parte de problemas identificados en la primera etapa. Posteriormente dentro de la tercera etapa se realiza el planteamiento de un marco de trabajo para la gestión de proyectos con el fin de eliminar la mayor parte de problemas identificados en la primera etapa y basado en el estándar o mejores prácticas seleccionado en la segunda etapa. Finalmente se señalan las conclusiones y trabajo futuro.

2. METODOLOGÍA

En la primera etapa, para la identificación de los principales sistemas, se basó en el alcance y duración de cada uno de los proyectos desarrollados por el área de sistemas. Para cada sistema desarrollado, se identificó los problemas que se presentaron en mayor y menor frecuencia, seleccionando los diez problemas que fueron recurrentes en más de un sistema. Para cada problema y en cada sistema se asignó una puntuación, siendo 10 el problema que tuvo mayor impacto y 1 el problema que tuvo menor impacto en la realización de un proyecto. Finalmente, se realizó la sumatoria de la calificación asignada a cada problema en los diferentes sistemas y se determinó de mayor a menor, los problemas que inciden en la culminación exitosa de dicho sistema.

En la segunda etapa, para la identificación del estándar o mejores prácticas se procedió a identificar el área o proceso que cubre uno o más problemas. Cada área o proceso que permita cubrir un problema específico se asignó una calificación, siendo 10 al problema de mayor impacto y 1 al problema de menor impacto. La sumatoria final, permite determinar el estándar o mejores prácticas que se adapta a las necesidades del área de sistemas del Grupo Metro.

Finalmente, se plantea el marco de referencia basado en el estándar seleccionado en la segunda etapa.

3. EVALUACIÓN DE RESULTADOS

3.1 Principales Sistemas y Problemas

A continuación se detalla los principales sistemas que actualmente posee el Grupo Metro, los mismos que han sido implementados en más de una empresa como se muestra en la Tabla 1.

TABLA 1: Principales Sistemas del Grupo Metro

Sistemas		Empresas					
Nombre	Definición	Metro	Tecemovil	Cabice	XYPrepago	Metrocel	Contarcom
Facturación	Permite generar las facturas que van a ser impresas y enviadas a los clientes.	No	Si	Si	Si	Si	No
Consumos	Permite generar los consumos mensuales y semanales que se deben cobrar a los clientes.	No	Si	Si	Si	Si	Si
Acreditaciones	Permite gestionar la acreditación de saldo a los diferentes puntos de venta de los clientes.	No	No	No	Si	Si	No
Conciliaciones	Permite realizar el cruce entre los depósitos del banco y los depósitos que registran las ejecutivas de Call center de los clientes.	No	Si	Si	Si	Si	No
Chips	Permite el ingreso de pedidos y despachos de chips para clientes y mayoristas.	No	Si	Si	No	No	Si
Mensajes	Permite enviar un mensaje de texto a los celulares de los clientes informando el saldo a cancelar o para realizar campañas publicitarias.	Si	Si	Si	Si	Si	No
Cobranza	Es utilizado por Cartera y Call center para gestionar los cobros a los clientes.	No	SI	SI	SI	SI	No

Para cada uno de los sistemas anteriormente detallados, se determinó los problemas que se han presentado en más de un proyecto y que han tenido un impacto significativo en la culminación exitosa de cada sistema. A continuación se muestra en la Tabla 2, la calificación que se asignó a cada problema, siendo 10 de mayor impacto y 1 de menor impacto.

TABLA 2: Resumen de Datos: Problemas vs Sistemas

Problemas	Sistemas De							TOTAL
	Facturación	Consumos	Acreditaciones	Conciliaciones	Venta de Chips	Mensajes	Cobranza	
Tiempos de desarrollo elevados.	8	6	9	8	6	6	10	53
Falta de planificación.	10	7	10	10	9	9	8	63
Recursos subutilizados o sobre utilizados.	6	5	8	7	5	5	7	43
Cambios de especificaciones a mediados o finales del proyecto.	7	8	2	6	7	7	9	46
Elaboración de proyectos simultáneos.	9	1	5	9	4	4	6	38
Los objetivos iniciales no se llegan a alcanzar.	2	2	3	4	3	3	5	22
Duplicación de esfuerzos.	4	4	1	3	2	2	4	20
Ambigüedad en el levantamiento de requerimientos.	5	9	7	5	8	8	3	45
Falta de interés por parte de las personas involucradas o los usuarios finales.	1	10	6	2	10	10	2	41
Producto final diferente a las especificaciones de los usuarios.	3	3	4	1	1	1	1	14

De acuerdo a los datos obtenidos en la Tabla 2, se puede determinar que el problema que mayor incide en la realización de proyectos actualmente es la Falta de Planificación. En la Figura 1, se puede observar el porcentaje de impacto que tiene cada uno de los problemas en la vida de un proyecto de software.

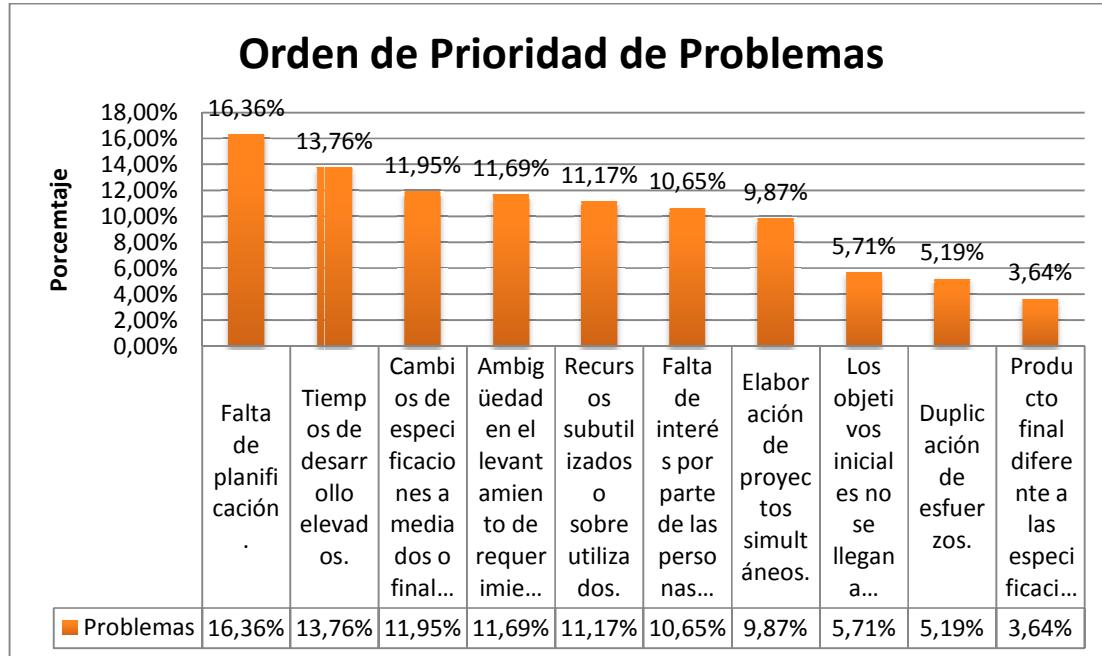


Figura 1: Prioridad de Problemas

3.2 Estándares y Mejores Prácticas

Los estándares y mejores prácticas que se utilizaron dentro del análisis son: CMMI, COBIT, PMBOK e ITIL. Para cada uno de ellos se resume el área o conjunto de procesos que se maneja para la gestión de proyectos:

- i. CMMI está compuesto de cuatro categorías que son: Gestión de Procesos, Gestión de Proyectos, Soporte e Ingeniería.
- ii. COBIT se encuentra dividido en cuatro dominios que son: Planear y Organizar, Entregar y Dar Soporte, Monitorear y Evaluar y Adquirir e Implementar.
- iii. PMBOK es el único estándar para la gestión de proyectos que se basa en 5 Grupos de Procesos y 9 Áreas de Conocimiento. A continuación en la Tabla 5, se muestra el problema y el nombre del área de conocimiento con el que PMBOK podría solucionar o abarcar dicho problema.
- iv. ITIL se compone de cinco procesos referentes a la entrega de servicio, cinco procesos de soporte al servicio y finalmente una función de mesa de servicio, con el fin de facilitar la entrega de servicios

En la Tabla 3, se muestra el resumen de las calificaciones obtenidas por cada estándar o mejores prácticas de acuerdo a la prioridad de cada problema. En este caso, PMBOK es el estándar que cubre la mayoría de problemas identificados en la primera etapa y los que tienen mayor prioridad. De este modo, como resultado de la segunda etapa se determinan que PMBOK es el estándar sobre el cual se planteará el marco de referencia para la gestión de proyecto de desarrollo de software.

TABLA 3: Solución a Problemas con los Diferentes Estándares y Mejores Prácticas

Problema	CMMI	COBIT	PMBOK	ITIL
Falta de planificación.	10	10	10	--
Tiempos de desarrollo elevados.	--	--	9	--
Cambios de especificaciones a mediados o finales del proyecto.	8	8	8	8
Ambigüedad en el levantamiento de requerimientos.	7	--	7	--
Recursos subutilizados o sobre utilizados.	--	6	6	--
Falta de interés por parte de las personas involucradas o los usuarios finales.	5	--	5	--
Elaboración de proyectos simultáneos.	4	4	4	--
Los objetivos iniciales no se llegan a alcanzar.	3	--	--	--
Duplicación de esfuerzos.	--	2	2	--
Producto final diferente a las especificaciones de los usuarios.	1	1	--	1
TOTAL	38	31	51	9

3.3 Planteamiento del Marco de Referencia

Una vez identificado los principales problemas y el estándar que se cubre de mejor manera estos problemas, se procede a plantear el marco de referencia de acuerdo a las cinco áreas de conocimiento de PMBOK que son: Gestión de la Integración de Proyecto, Gestión del Alcance del Proyecto, Gestión del Tiempo del Proyecto, Gestión de los Recursos Humanos y Gestión de las Comunicaciones.

3.3.1 Gestión de la Integración del Proyecto:


El desarrollo de esta área de conocimiento permite eliminar los problemas relacionados con: Falta de planificación, cambios de especificaciones a mediados o finales del proyecto y elaboración de proyectos simultáneos. El principal proceso que se desarrollan en esta área de conocimiento es:

- i. Desarrollar el Acta de Constitución del Proyecto: Este acta tiene como objetivo cubrir los aspectos referentes a: quién solicita, qué se solicita y finalmente, si es viable la ejecución del proyecto. Es una breve descripción del proyecto, pero que precisa las partes involucradas en dicho proyecto. A continuación en la Figura 2, se plantea el Acta de Constitución adaptada a las necesidades del Grupo Metro.

3.3.2 Gestión del Alcance del Proyecto:

El desarrollo de esta área de conocimiento permite eliminar el problema relacionado con: Ambigüedad en el levantamiento de requerimientos. El principal proceso que se desarrollan en esta área de conocimiento es:

- i. Definir el Alcance: El alcance es la definición desde y hasta donde debe llegar un proyecto, gracias a esta definición se puede estimar el periodo de tiempo que tomará el desarrollo del proyecto y los miembros que se necesitarán para el proceso del mismo. A continuación en la Figura 3, se plantea el documento para la Definición del Alcance del proyecto para el área de sistemas del Grupo Metro.

 <p>MACROPROCESO: Gestión de la Integración del Proyecto Acta de Constitución del Proyecto</p>	Código Proyecto: 1A	
	Versión: 1.0	
Página 1 de 1		

1. HISTORIA DEL DOCUMENTO				
Realizado Por:	Revisado Por:	Fecha Realización:	Fecha Revisión:	Versión:


2. DATOS SOLICITANTE		
EMPRESA SOLICITANTE:		
Metro <input type="checkbox"/>	Metrocel <input type="checkbox"/>	Tecemóvil <input type="checkbox"/>
Cabice! <input type="checkbox"/>	XY Prepago <input type="checkbox"/>	Contarcom <input type="checkbox"/>
ÁREA SOLICITANTE:		
Presidencia <input type="checkbox"/>	Finanzas <input type="checkbox"/>	Mercadeo <input type="checkbox"/>
Servicio y Operaciones <input type="checkbox"/>	Administrativa y RRHH <input type="checkbox"/>	Tecnología <input type="checkbox"/>
NOMBRES SOLICITANTES:		
Nombre y Apellido	Email	Teléfono y Ext:

3. DATOS REQUERIMIENTO		
NOMBRE DEL REQUERIMIENTO:		
NECESIDAD DEL REQUERIMIENTO:		
DESCRIPCIÓN BREVE DEL REQUERIMIENTO:		
EMPRESAS A APLICARSE:		
Metro <input type="checkbox"/>	Metrocel <input type="checkbox"/>	Tecemóvil <input type="checkbox"/>
Cabice! <input type="checkbox"/>	XY Prepago <input type="checkbox"/>	Contarcom <input type="checkbox"/>

4. VIABILIDAD DEL PROYECTO		
AUTORIZACIÓN DEL PROYECTO:		
Autorizado: <input type="checkbox"/>	No Autorizado: <input type="checkbox"/>	
Justificación:		
PRIORIDAD:		
Alta: <input type="checkbox"/>	Media: <input type="checkbox"/>	Baja: <input type="checkbox"/>

REALIZADO POR: REVISADO POR: AUTORIZADO POR:

Figura 2: Acta de Constitución del Proyecto

 <p>MACROPROCESO: Gestión del Alcance del Proyecto Alcance del Proyecto</p>	Código Proyecto: 1A	
	Versión: 1.0	
Página 1 de 1		

1. HISTORIA DEL DOCUMENTO				
Realizado Por:	Revisado Por:	Fecha Realización:	Fecha Revisión:	Versión:

2. DATOS SOLICITANTE		
EMPRESA SOLICITANTE:		
Metro <input type="checkbox"/>	Metrocel <input type="checkbox"/>	Tecemóvil <input type="checkbox"/>
Cabice! <input type="checkbox"/>	XY Prepago <input type="checkbox"/>	Contarcom <input type="checkbox"/>
ÁREA SOLICITANTE:		
Presidencia <input type="checkbox"/>	Finanzas <input type="checkbox"/>	Mercadeo <input type="checkbox"/>
Servicio y Operaciones <input type="checkbox"/>	Administrativa y RRHH <input type="checkbox"/>	Tecnología <input type="checkbox"/>
NOMBRES SOLICITANTES:		
Nombre y Apellido	Email	Teléfono y Ext:

3. DATOS DEL PROYECTO	
OBJETIVOS:	
Objetivos Generales:	
Objetivos Específicos:	
DEFINICIÓN DEL PROYECTO:	
ALCANCE DEL PROYECTO:	
ACTIVIDADES A REALIZARSE:	
FECHA DE INICIO:	FECHA DE FINALIZACIÓN:

Figura 3: Definición del Alcance del Proyecto

3.3.3 Gestión del Tiempo del Proyecto:

El desarrollo de esta área de conocimiento permite eliminar el problema relacionado con: Tiempos de desarrollo elevados. El principal proceso que se desarrollan en esta área de conocimiento es:

- i. Desarrollar el Cronograma: El resultado final de la definición de actividades, la secuencia de actividades, la asignación de recursos humanos y la duración de cada actividad da como resultado el cronograma de actividades. En la Figura 4, se muestra un ejemplo de un cronograma de actividades con la herramienta Microsoft Project.

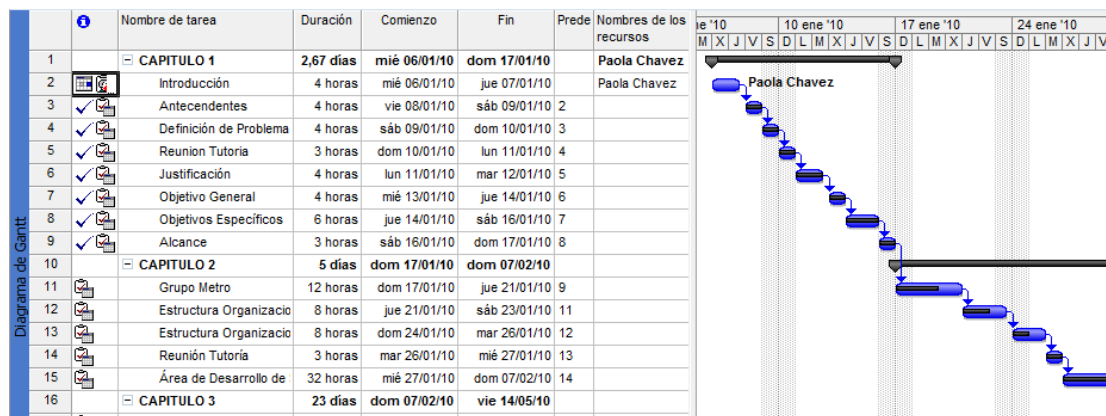


Figura 4: Cronograma de Actividades

3.3.4 Gestión de los Recursos Humanos del Proyecto:

El desarrollo de esta área de conocimiento permite eliminar los problemas relacionados con: recursos subutilizados o sobre utilizados, duplicación de esfuerzos. El principal proceso que se desarrollan en esta área de conocimiento es:

- i. Desarrollar el Plan de Recursos Humanos: El presente plan es específicamente para los miembros del área de desarrollo, el mismo que permite identificar a los recursos humanos en las diferentes etapas del proyecto. En la Figura 5, se muestra el Plan de Recursos Humanos para el área de sistemas del Grupo Metro.



Figura 5: Plan de Recursos Humanos

3.3.5 Gestión de las Comunicaciones:

El desarrollo de esta área de conocimiento permite eliminar el problema relacionado con: falta de interés por parte de las personas involucradas o los usuarios finales. El principal proceso que se desarrollan en esta área de conocimiento es:

- i. Planificar las Comunicaciones: Dentro de este proceso se define la matriz de gestión de las comunicaciones, en donde el principal objetivo es determinar qué se va comunicar, quién va a comunicar, cómo se va a comunicar, cuándo se va a comunicar y a quién se va a comunicar en las diferentes etapas del proyecto. En la Figura 6, se muestra el Plan de Gestión de las comunicaciones.


	MACROPROCESO: Gestión de las Comunicaciones del Proyecto		Código Proyecto: 1A		
	Plan de Gestión de las Comunicaciones		Versión: 1.0		
1. HISTORIA DEL DOCUMENTO					
Realizado Por:	Revisado Por:	Fecha Realización:	Fecha Revisión:	Versión:	
2. MATRIZ DE GESTIÓN DE LAS COMUNICACIONES					
FASE DEL PROYECTO	MENSAJE	EMISOR	MEDIO	FRECUENCIA	RECEPTOR
FASE DE INICIACIÓN					
FASE DE PLANIFICACIÓN					
FASE DE EJECUCIÓN					
FASE DE SEGUIMIENTO Y CONTROL					
FASE DE CIERRE					
REALIZADO POR:		REVISADO POR:		AUTORIZADO POR:	

Figura 6: Plan de Gestión de las Comunicaciones

4. TRABAJOS RELACIONADOS

Existe más de un estándar y mejores prácticas que permiten gestionar de mejor manera los proyectos de desarrollo de software, pero el presente proyecto permite personalizar el estándar PMBOK de acuerdo a las necesidades del área del sistema del Grupo Metro y cubriendo la mayoría de problemas identificados en la primera etapa. Permitiendo de esta manera modificar el planteamiento propuesto de acuerdo al crecimiento o giro del negocio.

5. CONCLUSIONES Y TRABAJO A FUTURO

El presente planteamiento permitió establecer los procesos para la gestión de proyectos de desarrollo de software, con el fin de evitar los diferentes problemas que se presentan actualmente dentro del área de sistemas. Los diferentes esquemas desarrollados en el presente trabajo, han sido adaptados a la realidad del Grupo Metro y basados en el estándar de PMBOK, con el fin de que sean implementados con facilidad y con el objetivo de eliminar los problemas identificados con la gestión de proyectos de desarrollo de software. El desarrollo del área de conocimientos de la Gestión de la Integración del Proyecto, permitió desarrollar los procesos para la formalización de un proyecto y la identificación de los usuarios involucrados, con el fin de cubrir el principal problema del área de sistemas que es la falta de planificación. El área de conocimiento de la Gestión del Alcance permitió establecer los procesos para definir el alcance del proyecto de manera clara y precisa, y además especificar lo requerimientos iniciales

del proyecto, con el objetivo de evitar el problema de la ambigüedad en el levantamiento de requerimientos que se presenta en el 11.69% de los proyectos que realiza el área de sistema. La Gestión del Tiempo permitió identificar a Microsoft Project como la herramienta de soporte, que permitirá al área de sistemas realizar el control estricto de las actividades y de los recursos humanos para cada proyecto, con el propósito de reducir los tiempos de desarrollo que afectan en un 13.76%. Con la Gestión de Comunicaciones, se logró establecer los procesos para mantener una comunicación constante y clara con cada uno de los miembros involucrados en los diferentes proyectos; permitiendo de esta manera, captar el interés de los usuarios en las diferentes etapas del proyecto, lo cual permite evitar el problema de falta de interés que afecta al 10,65% de los proyectos.

Agradecimientos: Al Grupo Metro, por permitir elaborar y aplicar paulatinamente el presente planteamiento.

Referencias Bibliográficas

- [1] Talledo, M. (2009). Guía de los Fundamentos de la Dirección de Proyectos. Estados Unidos, Project Management Institute, Inc.
- [2] CMMI Product Team. (2007). CMMI ForServices. Estado Unidos, Software Engineering Institute.
- [3] IT Governance Institute. (2005). Cobit 4.0. Estado Unidos, ITGI.
- [4] Grupo Metro. (2008). Planificación Estratégica del Grupo Metro. Ecuador.
- [5] División de Tecnología. (2008). Planificación Estratégica de la División de Tecnología del Grupo Metro. Ecuador.
- [6] Grupo Metro.(2008). Grupo Metro.<http://www.grupometro.com.ec/>.
- [7] Chrissis, M. (2009). CMMI Guía para la integración de procesos y la mejora de productos. <http://www.sei.cmu.edu/library/assets/cmmi-dev-v12-spanish.pdf>

Análisis, Diseño y Desarrollo del Sistema de Monitorización para el Servicio de Comunicaciones Unificadas en Elastix 2

Antonio Insuasti, Santiago Salvador, Diego Marcillo

Departamento de Ciencias de la Computación, Escuela Politécnica del Ejército, Sangolquí, Ecuador, antonio@insuasti.ec, smsalvador@espe.edu.ec, dmmarcillo@espe.edu.ec

RESUMEN: Este artículo revela el funcionamiento y la metodología usada para el análisis, diseño y desarrollo de un sistema de monitorización para el servicio de comunicaciones unificadas en Elastix 2. Elastix es una distribución de Gnu/Linux, la cual usa varios servicios como: Asterisk, Dahdi, Hylafax y Postfix, para brindar un entorno completo de Comunicaciones Unificadas basado en Software Libre. Los módulos de monitorización se los desarrolló en Perl los cuales trabajan directamente sobre el Kernel de Linux y envían los estados por medio de NRPE hacia Nagios, el cual notifica los estados de los servicios a los administradores vía correo electrónico, avisos en el portal Web ó por mensajes en la red social Twitter. Nagios es un sistema de monitorización ampliamente usado a nivel mundial de Licencia GPL. El módulo de configuración del sistema se lo desarrolló bajo PHP con el Framework Kohana, el cual permite un desarrollo fácil, seguro y ágil. Al ser éste un proyecto basado completamente en Software Libre se usó como metodología de desarrollo OpenUp/Basic, la cual es parte principal del Eclipse Process Framework. El sistema desarrollado permite a los administradores monitorizar todo el Software y Hardware que interviene en el servicio de Comunicaciones Unificadas como: teléfonos IP, troncales hacia la PSTN, servicio de telefonía, servicio de correo electrónico entre otros. Los resultados obtenidos demuestran que los administradores del sistema solucionan problemas antes de que estos causen inconvenientes a los usuarios.

Palabras Clave: Software Libre, Monitorización, VoIp, Comunicaciones Unificadas, Elastix

ABSTRACT: This document explains the operation and methodology used for analysis, design and development of a monitoring system for unified communications service based on Elastix 2. Elastix is a distribution of GNU/Linux, which uses several services such as Asterisk, Dahdi, Hylafax and Postfix to provide a complete environment of Unified Communications based on Free Software. The monitoring modules were developed in Perl. Perl modules works directly over linux kernel and it establishes a communication with Nagios through NRPE agent. The service's statuses are reported to administrators by mail, Web page or messages on Twitter. Nagios is the most used monitoring system whit GPL License. The configuration module was developed under the PHP Kohana Framework, which allows easy, secure and agile development. This project is based entirely on free software, development methodology used is OpenUP /Basic which is the main part of Eclipse Process Framework. The developed system allows administrators to monitor all software and hardware involved in the Unified Communications service as IP phones, trunks to the PSTN, telephone service, email service and much more. Since the implementation of this system, Administrators are able to solve problems before they affect final users

KeyWords: Free Software, Monitoring, VoIP, Unified Communications, Elastix

1. INTRODUCCION

En los últimos diez años se ha visto como la Voz sobre IP (VoIp) ha permitido que las empresas reduzcan el costo de las llamadas telefónicas y la comunicación entre sucursales. La Voz sobre IP en conjunto con Software Libre representan un gran ahorro a las empresas que lo implementan, pero se necesita el personal capacitado con una gran experiencia en administración para una implementación satisfactoria. Por lo cual este proyecto trata de eliminar de la lista de tareas del administrador la monitorización de servidores de Comunicaciones Unificadas, dándole a una herramienta de notificación de eventos vía correo electrónico o por medio de la red social Twitter, para que esté al tanto del sistema con la libertad de realizar tareas administrativas prioritarias.

El módulo de configuración del sistema se lo desarrolló bajo PHP con el Framework Kohana, el cual permite un desarrollo fácil, seguro y ágil. Al ser éste un proyecto basado completamente en Software Libre se usó como metodología de desarrollo OpenUp/Basic, la cual es parte principal del Eclipse Process Framework. El sistema desarrollado permite a los administradores monitorizar todo el Software y Hardware que interviene en el servicio de Comunicaciones Unificadas como: teléfonos IP, troncales hacia la PSTN, servicio de telefonía, servicio de correo electrónico entre otros. Los resultados obtenidos demuestran que los administradores del sistema solucionan problemas antes de que estos causen inconvenientes a los usuarios.

El resto del artículo ha sido organizado como sigue: La sección 2 describe la arquitectura del sistema de monitorización. La Sección 3 especifica el entorno de desarrollo. La Sección 4 explica el proceso de implementación. La sección 5 detalla la implementación y resultados de este estudio. Finaliza con las Conclusiones y trabajo futuro.

2. ARQUITECTURA DEL SISTEMA DE MONITORIZACIÓN

La mayoría de los sistemas de monitorización utilizan el paradigma Gestor-Agente, que viene a ser una analogía de lo que en Arquitectura de Software se conoce como Cliente-Servidor. El paradigma Gestor-Agente está compuesto por estaciones gestoras que envían solicitudes a los agentes y de agentes que retornan estados por un protocolo de red específico. La Figura 1, muestra sus elementos, mismos que son explicados a continuación:

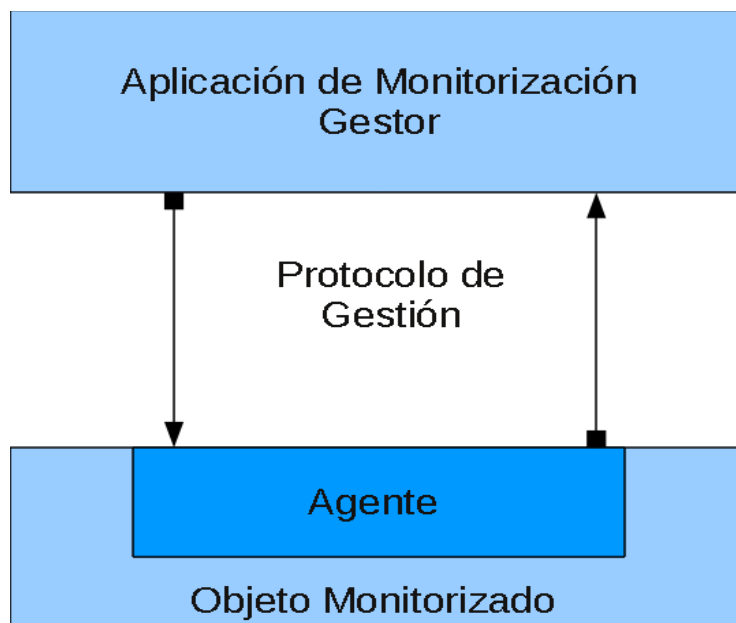


Figura 1. Paradigma Gestor - Agente

Gestor: Sistema donde se ejecuta la aplicación de monitorización de red, es el ente que solicita información de estado a los agentes. El gestor es la interfaz con la que interactúa el administrador de red

Agente: Envía información al sistema Gestor. En éstos residen los agentes, son capacidades por defecto de los dispositivos o sistemas monitorizados.

Protocolo de Gestión: Medio por el cual el Gestor y el Agente pueden intercambiar información.

Por ejemplo, para determinar el estado de un servidor Web, el gestor envía periódicamente una petición HTTP, el agente contesta las peticiones entregando una página Html y por su parte Gestor mide el tiempo de respuesta. En muchos casos se definen métricas que miden el tiempo de respuesta, disponibilidad y tiempo de actividad, entre otras.

Cuando existen fallas de solicitud de estado hacia el agente, por ejemplo, cuando una conexión no puede establecerse, cuando se han superado los tiempos de espera máximos, o el mensaje no se puede recuperar, el Gestor produce una acción en el sistema de monitorización. Estas acciones varían dependiendo de la configuración pero en general una alarma pueden ser enviada a través de SMS, correo electrónico, entre otros, al administrador del sistema para que ejecute las acciones que permitan una continuidad del servicio.

El Módulo de Monitorización se instala como parte del sistema Elastix (véase figura 2). Éste trabaja directamente con el Plugin de monitorización, el cual se utiliza como una interfaz entre los sistema Asterisk, Dahdi, Postfix y Hylafax, de los cuales se toma los datos de estado para poder visualizarlos en Elastix. La arquitectura y el funcionamiento del proyecto se muestran en la Figura 2.

El servicio NRPE - Cliente (Nagios Remote Plugin Executor Client, por sus siglas en ingles), trabaja como agente en conjunto con el Plugin de Monitorización para enviar las acciones solicitadas por el Servidor NRPE el cual usa Nagios para revisar los entornos y servicios Externos.

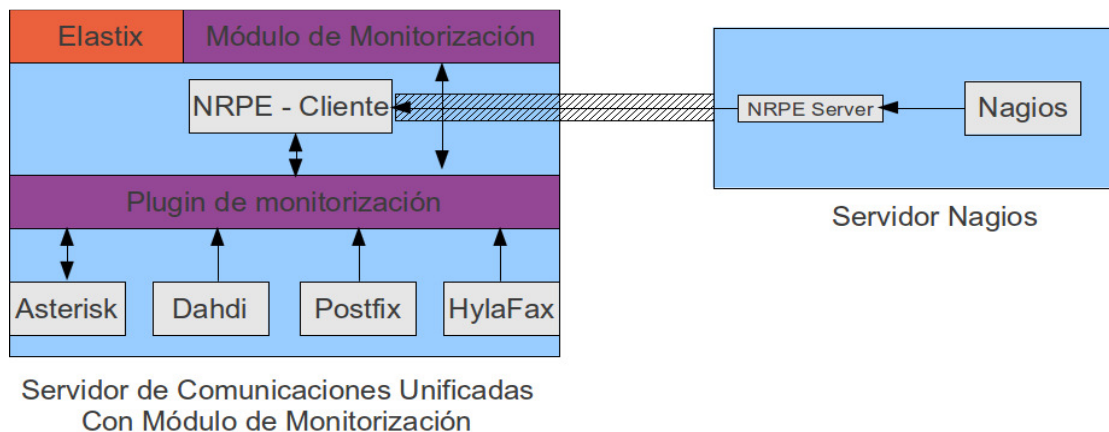


Figura 2. Arquitectura del Sistema de Monitorización.

3. ENTORNO DE DESARROLLO

Para el desarrollo del proyecto se utilizó el entorno de desarrollo integrado Eclipse el cual tiene la particularidad de poder agregarle funcionamientos para el desarrollo en PHP (Eclipse PDT) y para Perl (Eclipse EPIC). Las versiones usadas fueron:

- i) Eclipse PDT versión 2.2.0
- ii) Eclipse EPIC versión 0.5.46

El control de versiones y alojamiento de código fuente, al igual que el manejo y seguimiento de errores, se lo implementó a través de LaunchPad. **LaunchPad** es un conjunto de herramientas Web que apoya el desarrollo de proyectos de Software Libre.

Para el desarrollo del ambiente Web se utilizó Kohana un entorno de trabajo para PHP 5, seguro y ligero que implementa el “Modelo Vista Controlador Jerárquico”.

Se utilizó las siguientes versiones de software:

- iii) PHP 5.2.10

- iv) Apache 2.2.3-45
- v) Kohana PHP Framework, versión 3.2
- vi) MySQL versión 5.0.77-4
- vii) Perl versión 5.8.8 compilado para x86_64-linux-thread-multi

4. IMPLEMENTACIÓN Y RESULTADOS

La implementación, deja el sistema completamente funcionando en un entorno de producción en la Empresa MachangaraSoft. Los datos obtenidos demuestran que un aplicativo de monitorización de servidores de Comunicaciones Unificadas basadas en Elastix versión 2.x permite al administrador adelantarse a la resolución de inconvenientes, aumentando así su productividad, además muestran cuál es el porcentaje real de la disponibilidad de los servicios.

Uno de los servicios más importantes para esta empresa es su troncal hacia Estados Unidos, la cual permite tener una comunicación directa con clientes en este país a costo muy reducido. El funcionamiento de este sistema depende entre otras cosas de la conexión a Internet.

Antes de la implementación del Elastix Monitor no se sabía por qué se perdían llamadas hacia Estados Unidos, el personal técnico de MachangaraSoft adjudicaba el problema a una mala configuración del sistema de Comunicaciones Unificadas. Al instalar el sistema se obtuvo que el problema no es de configuración sino del proveedor de Internet como se demuestra en la Tabla 1.

TABLA 1: Reporte de Errores de la Troncal 3128541265

Service 'ext3128541265' On Host 'localhost'				
09-01-2011 00:00:00 to 10-01-2011 00:00:00				
Duration: 30d 0h 0m 0s				
Service State Breakdowns:				
State	Type / Reason	Time	% Total Time	% Known Time
	Unscheduled	18d 9h 48m 4s	61361%	80036%
	Scheduled	0d 0h 0m 0s	0.000%	0.000%
OK	Total	18d 9h 48m 4s	61361%	80036%
	Unscheduled	0d 0h 0m 0s	0.000%	0.000%
	Scheduled	0d 0h 0m 0s	0.000%	0.000%
WARNING	Total	0d 0h 0m 0s	0.000%	0.000%
	Unscheduled	0d 0h 15m 25s	0.036%	0.047%
	Scheduled	0d 0h 0m 0s	0.000%	0.000%
UNKNOWN	Total	0d 0h 15m 25s	0.036%	0.047%
	Unscheduled	4d 13h 56m 31s	15270%	19917%
	Scheduled	0d 0h 0m 0s	0.000%	0.000%
CRITICAL	Total	4d 13h 56m 31s	15270%	19917%
Undetermined	Nagios Not Running Insufficient Data	0d 0h 0m 0s	0.000%	
	Total	7d 0h 0m 0s	23333%	
All	Total	30d 0h 0m 0s	100000%	100000%

La Tabla 1 demuestra que en un mes, la troncal hacia Estados Unidos ha pasado 4 días sin funcionamiento a causa de desconexión provocada por una falla en el acceso a Internet.

El Histograma entregado por Nagios demuestra que el tiempo máximo de solución del problema fue de 41 minutos y ocurrió el 23 de Septiembre del 2011, como se muestra en la Figura 3.

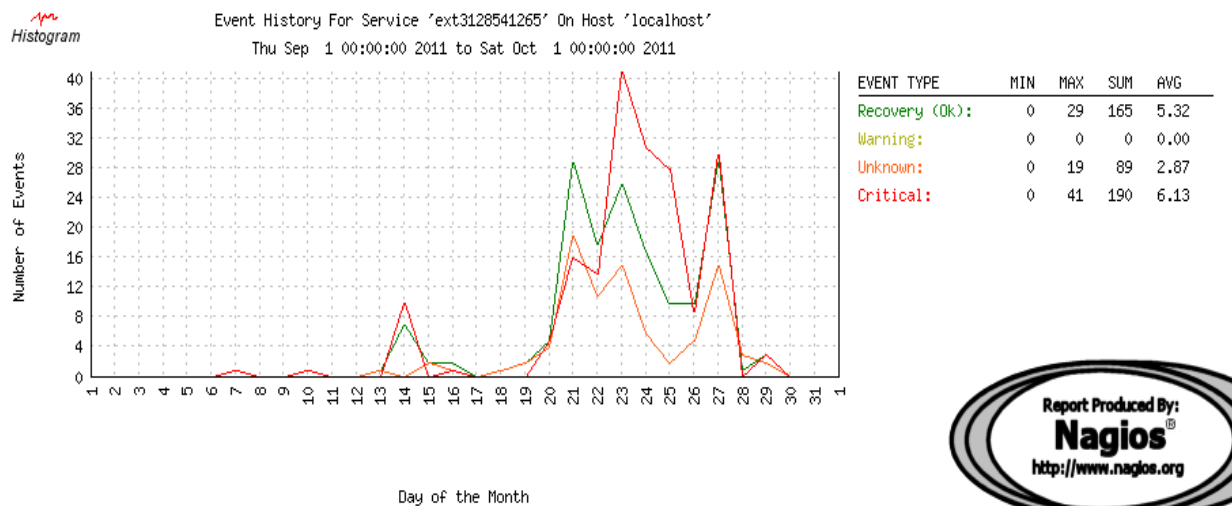


Figura 3. Histograma de Troncal 3128541245

Con estos resultados encontramos que a más de permitir al administrador conocer los problemas y disponibilidad del servicio, se puede saber qué tan eficiente es el administrador en solucionar los problemas del sistema de Comunicaciones Unificadas.

5. CONCLUSIONES

La implementación de Elastix Monitor permite al administrador tener el control total tanto del servidor de Comunicaciones Unificadas, como de los equipos que intervienen en el sistema: teléfonos IP, líneas analógicas, líneas digitales, sistemas de fax, correo electrónico entre otros servicios. Se ha facilitado las tareas de los administradores de un Servidor de Comunicaciones Unificadas basadas en Elastix Versión 2.x ya que no necesitan ingresar al administrador Web o la Consola para revisar que está pasando, sino simplemente reciben las notificaciones en su celular en forma de correo electrónico o "Tweet". El sistema desarrollado permite que el administrador decida qué servicios, extensiones, equipos, líneas o troncales va a monitorizar, direccionando los recursos a los servicios más críticos dependiendo sus necesidades. La instalación del sistema de monitorización en MachangaraSoft permitió a dicha empresa eliminar la tarea de revisión de su sistema de Comunicaciones Unificadas además de conocer el porcentaje de funcionamiento de su troncal hacia Estados Unidos. La integración con Nagios orienta el sistema de monitorización no solo a un servidor de Comunicaciones Unificadas sino también a todos los equipos que se conectan a la red interna, externa, o sistemas que tengan una interfaz conectada a la red de datos. El desarrollo con herramientas de Software Libre permiten el mismo desempeño que herramientas de código cerrado. El uso de LaunchPad como portal de control de versiones, hosting de código y sistema de manejo y seguimiento de errores demuestra que el desarrollo de Software Libre si mantiene un proceso ordenado, organizado y eficaz. El uso de la metodología OpenUp facilitó el desarrollo ágil del sistema ocupando menor tiempo en documentación, y aumentando la productividad del desarrollador con micro avances. El uso de Kohana permitió un desarrollo de interfaces Web de una forma rápida y segura ya que tiene un sistema de protección interno en contra de ataques SQL injection, "Cross Site Scripting" XSS, entre otros.

REFERENCIAS BIBLIOGRÁFICAS

- [1] Nagios Core Version 3.x Documentation Copyright © 2009 Nagios Core Development Team and Community Contributors. [<http://library.nagios.com/library/products/nagioscore/manuals/>].
- [2] Ethan Galstad, NRPE Documentation Copyright (c) 1999-2007 [<http://nagios.sourceforge.net/docs/nrpe/>].
- [3] Bruno Macias V., Manual de Desarrollo de Módulos para Elastix en Español Version 1.0-4 Beta
- [4] [http://sourceforge.net/projects/elastix/files/Tutorials_Docs_Manuals/Elastix_Developer_Module]
- [5] Antonio Martín Montes., Sistema Expertos y Gestión de Redes, Universidad de Sevilla, España. Octubre del 2002.
- [6] Wojciech Kocjan , Learning Nagios 3.0 , Packt Publishing Ltd, ISBN 978-1-84719-518-0, Octubre 2008
- [7] Shane Warden , Modern Perl , Copyright ©2010 chromatic , ISBN-10: 0-9779201-5-1 ISBN-13: 978-0-9779201-5-0, By Chromatic, Octubre 2011
- [8] Roberto Feltre Oreja, Software Libre y la construcción ética de la sociedad del conocimiento, Sociedad del Conocimiento , Icara Editorial, ISBN: 978-87-7426-959-8, La propiedad intelectual Página 103 – 105
- [9] Ricardo Goldberger ET ALL , Linux en las Pymes , Editorial Norma, ISBN:987-545-221-1, Capitulo 8 - Páginas 65 – 70