ESPE-DECC

"DECC *Report,* Tendencias en Computación"

REVISTA TÉCNICA DEL DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN.

ISSN 1390-5236

VICERRECTORADO ACADÉMICO.

VOL. 1, No. 2, 2010

Tal como lo expresa Alberto Einstein, "La mayoría de las ideas fundamentales de la ciencia son esencialmente sencillas y, por regla general pueden ser expresadas en un lenguaje comprensible para todos.". Basado en esta reflexión, en mi calidad de Vicerrector Académico, cúmpleme expresar mi profundo agradecimiento por este nuevo esfuerzo que realiza el Comité Editorial y el editor de la revista "**DECC-*Report*, Tendencias en computación**", del Departamento de Ciencias de la Computación, y mi sincera felicitación a todos quienes están participando con sus artículos, para la materialización del segundo volumen de esta revista.

En esta etapa de la historia de la humanidad, donde la investigación es la base del desarrollo tecnológico, la innovación, y el progreso de los pueblos, conviene impulsar el trabajo creativo llevado a cabo de forma sistemática y rigurosa, que permita contribuir a la ciencia. Esta es la mística de este nuevo volumen, derivado de la aplicación de las diferentes áreas del conocimiento de las Ciencias de la Computación, que a pesar de la omisión, la falta de participación y compromiso de algunos colegas, a quienes no les interesa publicar sus trabajos, ha sido inspirada en la necesidad de generar la cultura de escribir y socializar lo que en términos de ciencia, se realiza al interior de nuestra Escuela.

Hago mis sinceros votos por que florezca la entrañable capacidad que tienen docentes, estudiantes de pregrado y postgrado de difundir sus trabajos e investigaciones, lo cual coadyuvará en incrementar la producción científica de nuestra Escuela Politécnica del Ejército y de de nuestro querido país.

CRNL. EMC. Ing. Wilson Sánchez Valverde.
Vicerrector Académico de la ESPE.

Presentación

El Departamento de Ciencias de la Computación (DECC) de la Escuela Politécnica del Ejército (ESPE) consciente de la importancia de contar con un instrumento de difusión del quehacer académico e investigativo, pone en consideración de la comunidad el Volumen 1, No. 2, 2010, de la revista técnica "**DECC *Report*, Tendencias en Computación**".

Es una lamentable realidad por todos conocida, que la investigación científica en el Ecuador es una actividad de muy pocos. Según un estudio realizado por César Albornoz et al., en marzo de 2009, de todo lo que se puede considerar como investigación en el Ecuador, la mayor parte la realizan las universidades, pero no todas. De las 73 existentes, menos de diez abarcan casi la totalidad de esas investigaciones, y dentro de las mismas universidades, son pocos los docentes que consideran o ejercen la investigación como algo prioritario. En este estudio, se ha determinado que las causas para ello son entre otros los insignificantes recursos que se dedican a la investigación, la excesiva carga horaria para dictar clases a la que se somete a los profesores, la insuficiente remuneración que les obliga a trabajar en varias instituciones, y un ausente espíritu científico en un gran porcentaje de ellos.

A pesar de lo anteriormente expuesto, al interior de la ESPE, se siguen desarrollando esfuerzos por fortalecer su sistema de investigación, generando por ejemplo, la cultura de difundir los resultados de las investigaciones. En este contexto, en este volumen se recogen los resultados de proyectos de iniciación científica desarrollados en el DECC y comprende seis reportes técnicos de tesis de pregrado que han sido elaborados por egresados y profesores del DECC, los mismos que atravesaron por un riguroso proceso de selección, revisión y arbitraje. En números, se presentaron 52 artículos técnicos. En su primera evaluación por parte del Comité Editorial de la revista, fueron eliminados 26. Los restantes fueron enviados a revisión por pares de colegas expertos en las diferentes áreas, de los cuales solamente fueron aceptados seis. Estas publicaciones reportan trabajos técnicos-científicos en Ingeniería de Software, Tecnologías de Virtualización, Seguridades en Redes, Mensajería Electrónica, Desarrollo Web y Web 2.0.

Dado el vertiginoso avance de la ciencia y la tecnología en el campo de la Computación, así como el crecimiento exponencial del Internet, de sus aplicaciones o servicios, y de la dinamia de la tecnología, considero que "**DECC *Report*, Tendencias en Computación**" es una gran oportunidad para presentar los avances y logros en el campo de las Ciencias de la Computación y ramas afines en el contexto de la investigación, innovación, desarrollo tecnológico y emprendimiento.

Por tanto "**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.

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

Sumario:

Volumen 1, No. 2, 2010

Propuesta Uso de Metodologías Formales Combinadas con Metodologías Agiles para el Desarrollo de Software

[1]F. J. Lomas, [2]G. Raura y [2]M. Campaña
[1]*Kruger Corporation* S.A.
[2]*Departamento de Ciencias de la Computación, Escuela Politécnica del Ejército, Sangolquí, Ecuador*
xlomas@kruger.com.ec, georaura@gmail.com, mcampana@espe.edu.ec

RESUMEN: Una ventaja competitiva dentro de la industria del software es la velocidad con la cual se puede innovar y crear productos para distribuirlos al mercado. El presente artículo propone la combinación de metodologías formales, como el Proceso Unificado de Desarrollo, y Metodologías Agiles de Desarrollo de Software, como la Programación Extrema y sus técnicas, como métodos para mejorar la velocidad de construcción del software sin dejar de lado la calidad del mismo. Esto se consigue usando los artefactos de las metodologías formales en la etapa de toma de análisis del problema y toma de requerimientos, para en el resto de fases del desarrollo proceder a usar los artefactos y técnicas de las metodologías agiles. Para llevarlo a cabo, se modificó el documento de caso de uso incluyendo información grafica detallada en forma de un flujo de actividades para describir y delimitar el proceso contenido en el caso de uso. Así mismo, se generan casos de evaluación para ser verificados con pruebas unitarias extendidas para cubrir el alcance de cada caso. Los resultados obtenidos muestran una mejora en la velocidad de comprensión de los requerimientos por parte del equipo técnico. Como consecuencia de esto, el tiempo que toma la verificación y corrección de la funcionalidad es reducido entre el 15% y 35%, además de que los defectos del producto descubiertos cuando ya es liberado al mercado se reducen entre un 10% y un 20%.

Palabras clave: Proceso unificado de Desarrollo, Metodologías Agiles, Programación Extrema

ABSTRACT: Inside the software industry a competitive advantage is the speed on the innovation and creation process oriented to put products on the market. The present article propose the combination of formal software development methodologies, like the Unified Process, and agile methodologies, like Extreme Programming and it's techniques, as a method to speed up the software develop time without leaving the quality behind, this can be done using the formal methodologies artifacts in the requirements gathering and analysis phase, for the next phases we proceed to use the artifacts and techniques of the agile methodologies. To accomplish the objective the use case document was modified to include graphic information like a flow chart that represents the different activities of the process, the flow chart helps on the description and delimitation of the process. Also the test cases are generated to verify the use cases with extended unit tests to cover the use case scope. The results show a improve in the way and speed on understanding of the requirements inside the technical team, an secondary result is the reduced time to verify and fix the functional errors, the improvement is measured in the range of 15% and 35%, also the released to market product defects where reduced in the range of 10% and 20%.

Keywords: Agile Methodologies, Extreme Programming, The Rational Unified Process.

1. INTRODUCCIÓN

En la creciente industria del desarrollo de software, cuyas ventas se han incrementado en un 22.8% [1], que se enfrenta a un mundo completamente conectado y que brinda la suficiente información para que cualquier persona pueda plasmar sus ideas en programas de computadora simples o complejos sistemas de software, la innovación y velocidad al crear nuevos productos es la mejor ventaja competitiva que se puede tener [2]. La construcción de software a pesar de ser un tópico reciente ha evolucionado con gran velocidad en los últimos años, hoy se conocen ya metodologías que tienen varios años siendo usadas como el Proceso Unificado de Desarrollo que propone estructurar dicho proceso en varias fases, así como disciplinas que ejecutan las diferentes tareas basados en artefactos que pueden ser documentos o productos de software propios o de terceros. Sin embargo este tipo de metodologías que son más estructuradas y maduras requieren también de un mayor esfuerzo para ser aplicadas correctamente, este esfuerzo se ve traducido en un incremento los recursos que se usan en el proyecto como tiempo y dinero.

La calidad del software producido es uno de los factores que ha empujado el surgimiento de las distintas metodologías de desarrollo de software, dado que el software es incluido cada vez más en industrias que requieren de alta fiabilidad como la medicina y el automovilismo. Sin embargo el proceso de verificación de calidad debe ser riguroso, ya que el costo de cada falla encontrada esta dado por el costo de el trabajo que toma repararlo [3], por lo tanto menos fallas se reflejan en un menor costo. Al tener la necesidad de producir software de calidad en un tiempo corto se nos presenta el reto de lograr este objetivo con las herramientas maduras que son las metodologías formales de desarrollo de software, y además se tiene la posibilidad de usar nuevas herramientas como las técnicas de la programación extrema [4], que han probado ser una alternativa válida y sencilla de usar [5], se propone combinar a conveniencia el uso de los artefactos de las metodologías formales con el uso de las técnicas de las metodologías agiles para acortar el tiempo empleado para sacar el producto al mercado. En general el propósito es generar un ahorro en tiempo y recursos al ejecutar un proyecto de desarrollo de software combinando los artefactos y las técnicas de las metodologías de desarrollo de software formales con las metodologías de desarrollo agiles.

El resto del artículo ha sido organizado como sigue: La sección 2 muestra los artefactos del Proceso Unificado a ser modificados y las técnicas de programación extrema a ser usados. La sección 3 detalla las modificaciones y forma de empleo de las técnicas de la sección 2. En la sección 4 se muestran los resultados obtenidos. En la sección 5, se analizan algunos trabajos relacionados. Finalmente, en la sección 6, se presentan las conclusiones y líneas de trabajo futuro sobre la base de los resultados obtenidos.

2. MATERIALES, ARTEFACTOS Y TÉCNICAS A SER USADOS Y/O MODIFICADOS

Caso de Uso: El documento de caso de uso tiene por objetivo organizar la información de uno o más requerimientos para que quien construye el sistema pueda guiarse estructuradamente, además sirve como un documento que debe ser presentado a los interesados con fines de validación de los requerimientos. Al ser este artefacto de comunicación entre el equipo de desarrollo y los interesados en el desarrollo se entiende la importancia del mismo.

Caso de Prueba: Los casos de prueba están derivados de los casos de uso, en resumen un caso de prueba es un plan estructurado de cómo se debe verificar la funcionalidad construida. En un caso de prueba se debe tomar los subprocesos del proceso principal automatizado para ser verificados.

Pruebas Unitarias: Las pruebas unitarias son una técnica de la Programación Extrema, que propone generar código que prueba la funcionalidad de ciertas piezas del sistema en manera aislada, los resultados se basan en la comparación del resultado obtenido con el resultado esperado.

3. DISEÑO E IMPLEMENTACIÓN

Este es un ejemplo del caso de uso modificado:
Especificación CU01: Evaluación y selección de Aspirantes

Descripción: La presente especificación tiene por objetivo mostrar la propuesta de automatización del proceso de la Evaluación y selección de aspirantes de la Carrera de Ingeniería en Informática en Sistemas en un nivel macro.
Precondiciones: Que existan aspirantes a la Carrera de Ingeniería en Informática y Sistemas. Que exista un modelo de competencias definido para la Carrera, incluyendo las evaluaciones correspondientes.
Post condiciones: Existirán evaluaciones realizadas por los Aspirantes de la Carrera y calificadas en forma automática. Existirán informes sobre los Aspirantes que cubran los requerimientos mínimos definidos en el modelo de competencias de la Carrera.

I. **Actores**

i. Actores Principales

TABLA 1: Listado de actores que participan en el caso de uso, se detalla cómo se lo conoce dentro del caso de uso y su rol dentro del proceso que se está describiendo en el documento de caso de uso.

Actor	*Descripción*
Aspirante	*Persona nacional o extranjera que ha culminado sus estudios secundarios y que cumple con los requisitos legales de la República del Ecuador establecidos por los organismos competentes para acceder a la Educación de Tercer Nivel.*
Administrador de la Solución	*Persona encargada de ingresar los datos básicos necesarios y vigilar el funcionamiento de la Solución de Evaluación y Selección de Aspirantes para que pueda cumplir su objetivo.*
Solución de Evaluación y Selección de Aspirantes (E-Recruit)	*Sistema informático que ayuda con el procesamiento de los datos de los Aspirantes a la Carrera en el proceso de selección y evaluación, este automatiza el proceso de recolección de datos, calificación de evaluaciones basado en el modelo de competencias de la Carrera y generación de informes del resultado del proceso.*

ii. Actores Secundarios: *N/A*

II. Flujo de Eventos

i. Flujo Base

a. *El Administrador del Sistema ingresa los datos del modelo de competencias de la Carrera y define los niveles mínimos.*
b. *El Administrador del Sistema ingresa las posiciones o cupos disponibles para el siguiente periodo.*
c. *El Administrador define las evaluaciones y su correspondencia con el modelo de competencias ingresado para la Carrera.*

d. El Aspirante accede al sistema e ingresa sus datos básicos.
e. El aspirante procede a rendir las evaluaciones.
f. El sistema procede a calificar las evaluaciones terminadas.
g. El Administrador invoca la generación de informes y notificación de resultados.

ii. Flujo Alternativo: *N/A.*

III. Diagramas

i. Diagramas de Caso de Uso

Este diagrama muestra qué actividades se realizan dentro del caso de uso y su relación entre los actores, también muestra dependencia entre las actividades.

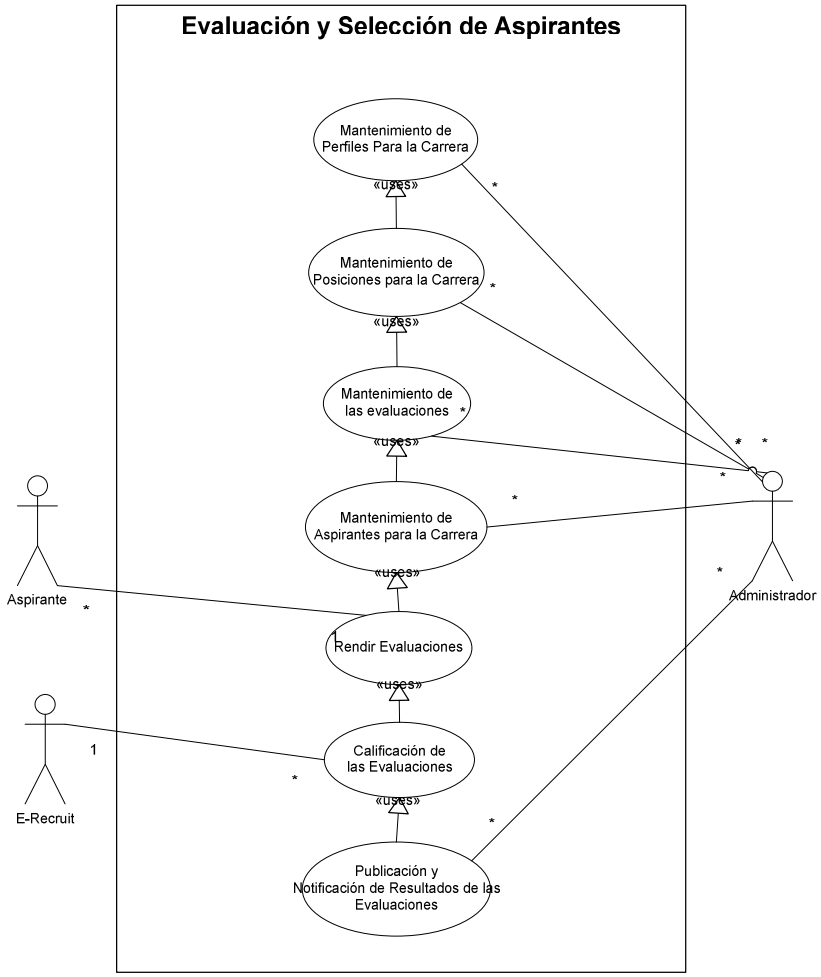

Figura 1: diagrama de caso de uso de evaluación y selección de aspirantes

ii. Diagramas de Flujo

El diagrama de flujo muestra cómo se dan las actividades dentro del proceso, delimitando qué actor está involucrado con qué actividad y a su vez clarifica como debe fluir la información.

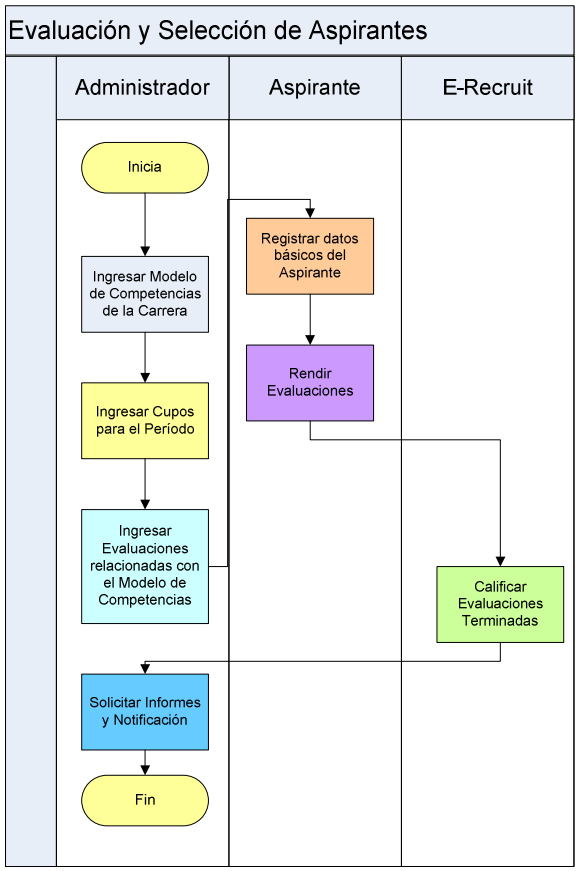

Figura 2: Diagrama de Flujo del Caso de Uso de Evaluación y Selección de Aspirantes

IV. Interfaz Gráfica de Usuario

Este proceso tendrá una Interfaz Gráfica de Usuario basada en un explorador de Internet, se definirán pantallas de acuerdo a la necesidad para satisfacer los requerimientos del usuario.

V. Relaciones

i. Interfaces con otros sistemas: *La presente solución no plantea ninguna interfaz con otro sistema.*

VI. Inclusiones: *N/A.*

VII. Exclusiones: *N/A.*

VIII. Suposiciones y Dependencias

i. Suposiciones: *Se cuenta con toda la información del modelo de competencias para la Carrera.*

IX. Dudas: *N/A.*

X. Observaciones

La solución procesará el grupo de evaluaciones basada en los parámetros ingresados en el modelo de competencias y la definición de perfiles, si estos datos no están verificados pueden ocasionar contratiempos dentro del proceso.

XI. Requerimientos Especiales: *N/A.*

Nótese en la sección de *Diagramas* en la Fig. 2 se encuentra un gráfico que normalmente no es incluido en un caso de uso, esta figura refleja el flujo de eventos y de información que se describe en la sección *Flujo de Eventos*, este diagrama puede ser generado en cualquier herramienta, tomando en cuenta que mientras

más descriptivo es el mismo mejor serán los resultados obtenidos a posterior. A continuación un caso de prueba de ejemplo:

CP03: Revisión y publicación de resultados

Descripción: *En este caso de prueba se pretende verificar las notificaciones a los aspirantes que han sido considerados idóneos para cursar la carrera.*

Precondiciones: *Evaluaciones ejecutadas*

Escenario (ver Tabla 2)

Como se puede notar en el cuadro que muestra el escenario, el caso de prueba se compone de varias verificaciones en pasos individuales, análogamente las pruebas unitarias verifican funcionalidad de un componente a la vez comparando los resultados obtenidos con resultados esperados teóricamente.

En este caso la propuesta requiere que se generen una prueba de unidad por cada paso del caso de prueba, para posteriormente en una prueba unitaria más grande juntar todas las pruebas unitarias que representan al caso de prueba. Para ejecutar las pruebas se pueden usar otras técnicas de la programación extrema como es la integración continua, o a su vez cuando el software pase a control de calidad estas pruebas pueden ser ejecutadas por las herramientas que se disponga en el ambiente de trabajo. Un ejemplo de las pruebas que se deben generar se lo puede encontrar en el blog *"Algo de .NET"* [6].

También es posible usar la técnica de integración continua, la cual a más de combinar y construir el código del sistema en cuestión puede ejecutar las pruebas unitarias de forma automática con los parámetros que sean necesarios, con esto se consigue que se pruebe la calidad del software incluso antes de que llegue al equipo de control de calidad.

TABLA 2: Aquí se describe el escenario que debe ser probado, que tiene que ser concordante con el caso de uso que pretende cubrir, describiendo claramente los pasos de cómo se debe ejecutar la prueba.

CP02: Revisión y publicación de resultados (CU08)

Paso del Caso de Uso	***Descripción del Paso***	***Resultado Esperado***	***Resultado Obtenido***	***Completo/ Fallo***	***Ambiente***	***Número de Log***
1. El administrador solicita el informe de aspirantes idóneos	*El administrador solicita el reporte de aspirantes idóneos*	*Presencia en el menú de opciones el reporte de aspirantes idóneos*				
2. La solución genera el reporte	*La solución genera el reporte de aspirantes idóneos*	*Reporte generado listo para revisión*				
3. El administrador pública la lista	*Si se considera completa la lista, se puede publicar y generar notificaciones*	*Pantalla que permite publicar la lista final de aspirantes aceptados*				
4. Se generan notificaciones para los aspirantes aceptados	*Se notifica a los aspirantes que han sido aceptados*	*Envío de mails a los aspirantes aceptados, se publica el reporte*				

4. EVALUACION DE RESULTADOS

Para poder medir la efectividad de los artefactos y de los métodos aplicados, se consideró ejecutar dos comparaciones, la primera se ejecuta dentro de un mismo proyecto, y la segunda se compara con otro proyecto de similares proporciones. En el primer proyecto se decidió dividir en 2 equipos, los que usan los artefactos modificados y las técnicas de pruebas, y quienes usan los artefactos y proceso normal, el trabajo fue repartido equitativamente a los dos equipos en función del esfuerzo estimado requerido para que la medición sea más certera. El siguiente cuadro muestra los datos más significativos obtenidos:

TABLA 3: Comparación de los resultados obtenidos entre los equipos A y B que participan en el experimento.

	Equipo A (Artefactos Modificados)	**Equipo B (Desarrollo Tradicional)**	**Variación Porcentual**
Número de Casos de Uso	10	10	0%
Tiempo estimado (Horas Hombre)	640	640	0%
Tiempo real (Horas Hombre)	660	743	11%
Número de Errores Reportados por QA	305	437	30%
Número de Errores Reportados por Usuario Final	65	83	22%

En la Fig. 3, se puede apreciar una comparación de las horas hombre empleadas para ejecutar los trabajos por parte de cada uno de los equipos dentro del proyecto, se puede ver un menor tiempo empleado por el equipo A.

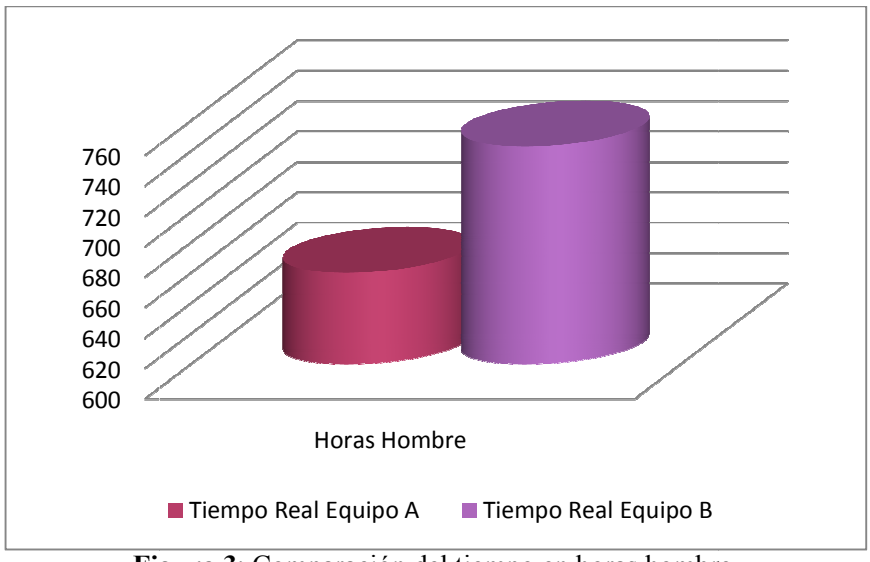

Figura 3: Comparación del tiempo en horas hombre.

En la Fig. 4, se comparan el número de errores reportados por QA y por el usuario final, se nota nuevamente que quienes usan las técnicas propuestas tienen una mejora en sus resultados.

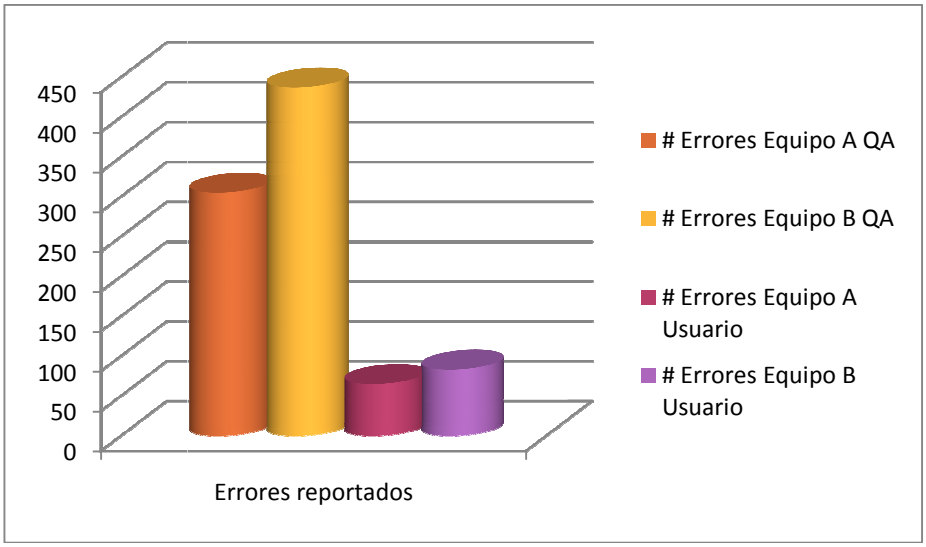

Figura 4: Número de errores reportados para los equipos A y B.

Como se puede observar el tiempo de desarrollo mejoró en un 11%, el número de errores reportados por Control de Calidad fue un 30% menor, y los errores reportados por el usuario final son un 22% menores. En la segunda comparación se tomaron dos proyectos, uno previamente realizado, y uno que se realizo con los artefactos realizados y las técnicas propuestas. Se usaron parámetros como puntos de función para comparar su tamaño así como recursos empleados en el proyecto, basados en esos factores se compararon proyectos de similares tamaños, los datos recogidos son los siguientes:

TABLA 4: Comparación de los resultados obtenidos entre los proyectos A y B, siendo el proyecto A el que usa artefactos modificados y técnicas agiles y el proyecto B desarrollo tradicional.

	Proyecto A (Artefactos Modificados)	**Proyecto B (Desarrollo Tradicional)**	**Variación Porcentual**
Número de Puntos de Función	481	489	2%
Tiempo estimado (Horas Hombre)	2886	2934	2%
Tiempo real (Horas Hombre)	3095	3589	14%
Número de Errores Reportados por QA	367	475	23%
Número de Errores Reportados por Usuario Final	175	231	24%

En la Fig. 5 se muestra una comparación del número de horas hombre que se emplearon en los proyectos A y B, el proyecto A que usa las técnicas propuestas usa menor cantidad de horas hombre.

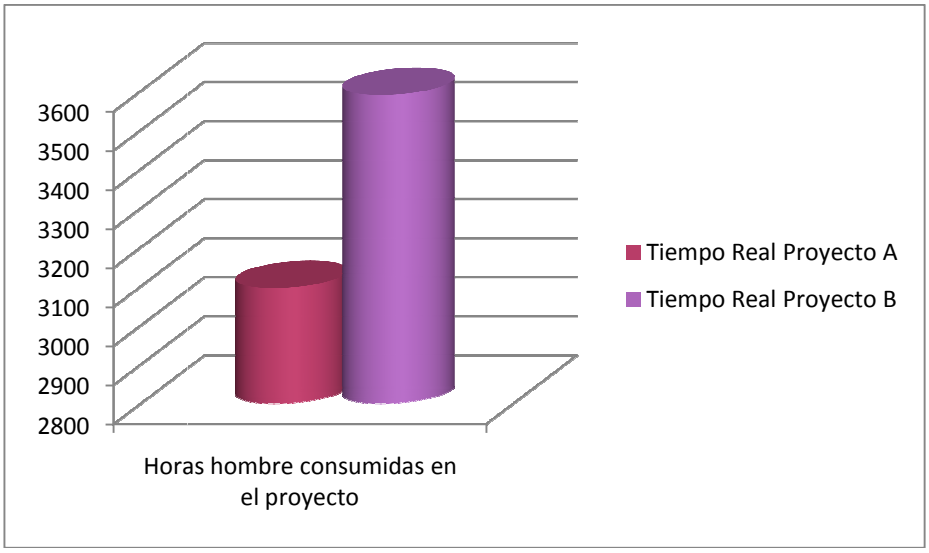

Figura 5: Horas hombre por cada proyecto.

La Fig. 6 muestra la comparación de número de errores reportados para los proyectos A y B, nuevamente se nota una mejora en el proyecto A que usa las técnicas propuestas incluso si se totalizan los errores.

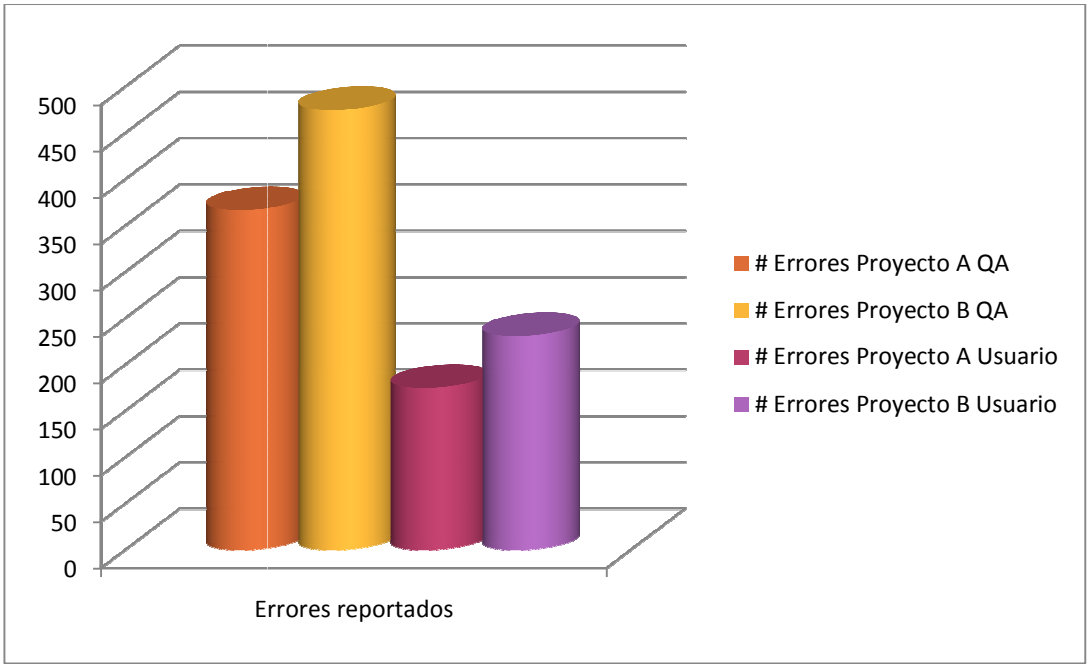

Figura 6: Comparación de errores reportados entre los proyectos A y B

Tomando en cuenta que hay una diferencia del tamaño del proyecto de aproximadamente 2%, los resultados pueden ser interpretados como que necesitan ser ajustados a la misma variación porcentual, el tiempo de desarrollo mejoró un 12%, el número de errores reportados por control de calidad fue menor en un 21%, y el número de errores reportados por el usuario final fue menor en un 23%.

5. DISCUSIÓN

Como se muestran en los cuadros y gráficos anteriores, la modificación de los artefactos y la combinación de técnicas agiles con metodologías formales puede traer beneficios inmediatos en la reducción de uso de recursos durante la vida del proyecto, el medio es mejorar los artefactos de forma que sean más claros y entendibles tanto para el equipo técnico como para los interesados, esto nos asegura que lo que se va a desarrollar es más aproximado a lo que desean los interesados. Un problema que puede presentarse es el alcance y extensión de los casos de uso, según quien analice y diseñe pueden existir un numero alto de casos de uso, esto puede ser contraproducente porque requerimos más recursos para generar las pruebas, así como para ejecutarlas, y claro esto también suma en número de errores, tiempo de manejo de los errores, tiempo de corrección y tiempo de verificación, por lo que es recomendable tener en cuenta si se desea usar las modificaciones y técnicas sugeridas en este articulo se debe también evaluar la granularidad de los casos de uso.

Las limitaciones que existirían por las herramientas de pruebas a ser usadas se pueden resolver con uso de herramientas de código abierto como NUnit y JUnit, siempre es recomendable usar herramientas integradas con el entorno de desarrollo, la falta integración de estas herramientas en el entorno de desarrollo puede suponer un problema de bajo impacto. Otro posible problema es la necesidad de entrenamiento para el equipo técnico con el fin de aprovechar al máximo las técnicas propuestas, por esto es recomendable escoger herramientas de pruebas que son comunes en el mercado además de que en lo posible estas deberían estar orientadas a cualquier tipo de usuario técnico, no solo al personal de control de calidad.

6. TRABAJOS RELACIONADOS

No existe suficiente documentación sobre trabajos relacionados, dado que la modificación de artefactos de metodologías y la personalización de una metodología, como la del proceso unificado de desarrollo, generalmente son consideradas como un trabajo comercial, sin embargo existen publicaciones de libros como *"Building J2EE applications with the rational unified process"* de Peter Eeles, Kelli Houston y Wojtek Kozaczynski, en las cuales se trata el tema de la personalización de la metodología de desarrollo de software para adaptarlo a las necesidades del equipo de desarrollo. Un libro que puede servir también de guía sobre cómo aplicar correctamente el proceso unificado de desarrollo es *"The rational unified process made easy: a practitioner's guide to the RUP"* de Per Kroll y Philippe Kruchten.

Con respecto a la aplicación de las pruebas unitarias con casos de pruebas se tiene más difusión sobre su uso y beneficio. La aplicación puede ser revisada en la referencia [7], donde se muestra una técnica muy parecida a la propuesta en el presente documento, además en la referencia [8] se puede revisar como implementar las pruebas unitarias con un alto detalle. También se habla de la efectividad de estas técnicas, en algunos casos se ha hecho referencia a que los resultados son concluyentes y además un poco exagerados [9].

7. CONCLUSIONES Y TRABAJO FUTURO

Con los datos recopilados se puede concluir que existe una mejora en el tiempo de desarrollo de un proyecto de software modificando los artefactos que se usan y aplicando técnicas no convencionales al desarrollo de software tradicional. Se consiguió una mejora de entendimiento entre los interesados y quienes desarrollan el software volviendo el documento de caso de uso más visual, agregándole un diagrama de flujo que resuma los requerimientos del proceso a ser automatizado. El menor número de errores reportado por el área de control de calidad es una consecuencia directa de ejecutar las pruebas unitarias que reflejan los casos de pruebas como política para que las secciones del proyecto sean pasadas a revisión, desembocando en un menor tiempo de revisión del software así como menor tiempo de reparación de errores de software y otros beneficios indirectos como menor probabilidad de regresión de errores. El menor número de errores reportados también es influenciado por la modificación de los casos de uso, al comprender mejor la funcionalidad quien construye puede ser más efectivo el desarrollador al construir la solución. Si bien los resultados oscilan entre el 10% y 25% de mejora estos números pueden mejorar en una medida moderada confirme a que quienes usan estos artefactos y estas técnicas adquieren mayor experiencia en el uso de los mismos, sin embargo estos resultados también pueden ser afectados por un sinnúmero de otras variables de ambiente que afectan comúnmente a los proyectos de desarrollo de software, es por esto que se deben aplicar estas técnicas recomendadas paso a paso en los ambientes de desarrollo finales, evaluando y ajustando a las necesidades propias los artefactos y las técnicas.

Como trabajo futuro se investigará la generación de código automática, incluyendo pruebas de unidad, basada en los artefactos de documentación creada. También se puede proponer un estudio sobre la aplicación de las técnicas y modificación de los artefactos propuestos sobre proyectos de largo aliento, es decir de duración de más de 6 meses.

Referencias Bibliográficas

[1.] The Entretainment Software Association. The Entretainment Software Association. [En línea] The Entretainment Software Association, 01 de 01 de 2009. [Citado el: 07 de 12 de 2009.] http://www.theesa.com/facts/index.asp.

[2.] The Free Library. Software company a roaring success; INDUSTRY: Innovative product helps H&S firm increase sales by ten-fold. *The Free Library.* [En línea] 13 de 11 de 2009. [Citado el: 7 de 12 de 2009.]

[3.] W. Ward *Calculating the real cost of software defects.*. 1, s.l. : Hewlett-Packard Journal, 1991, Vol. 1.

[4.] Wells, Don. Extreme Programming Rules. *Extreme Programming* . [En línea] 1 de 1 de 1999. [Citado el: 7 de 12 de 2009.] http://www.extremeprogramming.org/rules.html.

[5.] Extreme Programming Research. Extreme Programming Research. [En línea] 22 de 06 de 2002. [Citado el: 7 de 12 de 2009.] http://c2.com/xp/ExtremeProgrammingResearch.html.

[6.] Lomas, Francisco. Ejemplo de Pruebas Unitarias. *Algo de .NET.* [En línea] 01 de 01 de 2009. [Citado el: 8 de 1 de 2010.] http://fcolomas.blogspot.com.

[7.] Exforsys. Unit Testing: Why? What? & How? *Free Training.* [En línea] Exforsys, 01 de 01 de 2009. [Citado el: 14 de 12 de 2009.] http://www.exforsys.com/tutorials/testing/unit-testing.html.

[8.] Laurie Williams, Dright Ho, Ben Smith and Sarah Heckman. Unit Testing in Jazz Using JUnit. *North Carolina State University.* [En línea] North Carolina State University, 28 de 08 de 2008. [Online:] http://agile.csc.ncsu.edu/SEMaterials/tutorials/junit/junit_tutorial_jazz.html.

[9.] Proffitt, Jacob. TDD Proven Effective! Or is it? *TheRuntime.com.* TheRuntime.com, [Online:] http://theruntime.com/blogs/jacob/archive/2008/01/22/tdd-proven-effective-or-is-it.aspx.

Formulación de un modelo para evaluar herramientas de análisis de requerimientos, basado en la norma ISO 25000.

S. Moya, C. Hinojosa y R. Reyes
Departamento de Ciencias de la Computación, Escuela Politécnica del Ejército, Sangolquí, Ecuador
linita1985@gmail.com, chinojosa@espe.edu.ec, rprc2@hotmail.com

RESUMEN: Estudios realizados muestran que más del 53% de los proyectos de software fracasan por no realizar una adecuada administración de los requerimientos. Ante este problema, el presente artículo propone un modelo para evaluar herramientas de esta índole. El objetivo principal de las herramientas CASE que apoyan esta fase del desarrollo de software es facilitar la ejecución del proceso de ingeniería de requisitos, mediante un sistema computacional, que propicie una mejor comunicación entre los equipos de trabajo, defina los roles de mejor manera y además reduzca el riesgo de los proyectos, permitiendo tener una perspectiva global de los proyectos en todo momento. Para la formulación del modelo de evaluación se consideró el modelo de calidad presentado por la Norma ISO 25000, cuyas características son: funcionalidad, fiabilidad, mantenibilidad, eficiencia, usabilidad y portabilidad. En base a un caso de estudio se efectuó el análisis de requisitos con tres herramienta y se pudo evaluar el desempeño y cumplimiento de la información entregada por los fabricantes, y así de una manera objetiva desarrollar la evaluación. Una vez aplicado el modelo, los resultados obtenidos mostraron las debilidades y fortalezas de las herramientas, así como su desempeño en cada parámetro evaluado. El aporte de este trabajo al área de la Ingeniería de Software es la propuesta de un modelo que puede ser utilizado para evaluar herramientas CASE que brindan soporte al análisis de requisitos y que puede ser tomado como base para otros estudios similares; así también los resultados obtenidos en este trabajo pueden ser un referente para los desarrolladores al momento de decidir qué herramienta utilizar.

Palabras clave: Herramientas CASE, Normas ISO, modelo de evaluación

ABSTRACT: Studies made show that more than 53% of the software projects failed because there is an inadequate administration of the requirements. Therefore this article proposes an evaluation model that will allow evaluate tools of this kind. The basic idea of the CASE tools is that they support the development software phase providing an easier execution process of the engineering requirements through computational systems. This will consent a better communication between work teams, define the roles more clearly and also lessen the risk of projects enabling a global perspective of the projects at all time. This model was made taking into consideration quality model presented by the Norma ISO 25000, which characteristics are: functionality, reliability, maintainability, efficiency, utility, and portability. The analysis of the requirements with three tools was made based on a case study that allowed the evaluation on the development and compliance of the delivered information by the manufacturers and through this objective way expands de evaluation. Once the model was applied the results showed the strengths and weakness of tools, like there enrollment in each evaluated task. The contribution of this study to the area of Software Engineering is the proposal of a model that can be use to evaluate CASE tools that give support to the requirement analysis and can be taken like a baseline for similar studies. This way too the developers of the study can take it as a baseline for the manufacturers to decide which tool to use.

Keywords: CASE Tools, ISO standards, evaluation model

1. INTRODUCCIÓN

A través de los años se ha podido constatar que los requerimientos o requisitos son la pieza fundamental en un proyecto de desarrollo de software, ya que marcan el punto de partida para actividades como la planeación, estimaciones de tiempos y costos, así como la definición de recursos necesarios y la elaboración de cronogramas que se constituyen en los principales mecanismos de control con los que se contará durante la etapa de desarrollo.

En el mercado se pueden encontrar herramientas para administración de requerimientos como lo son: *Rational RequisitePro*, *Web Requisite* o *CaliberRM*; herramientas CASE que permiten especificar requerimientos, pero como saber ¿cuál utilizar?, esta investigación presenta una propuesta de un modelo de evaluación que permitirá elegir la herramienta que más se ajuste a las necesidades.

Frente a este escenario, como contribución, el presente artículo se basa en la definición de un modelo de evaluación basado en la norma ISO 25000. Para la realización del modelo, se propone la utilización de algunos parámetros propios de este tipo de herramientas, además de las características especificadas por la norma, las cuales han recibido una ponderación según el impacto en el proceso de requisitos. Se escogieron tres herramientas representativas del mercado como son *Rational RequisitePro, CaliberRM y Doors*, a las cuales se aplicó el modelo propuesto, para determinar las características más relevantes de cada una.

El resto del artículo ha sido organizado como sigue: La sección 2 describe los fundamentos teóricos del modelo de evaluación. La sección 3 detalla el diseño del modelo e implementación en un caso de estudio. En la sección 4 se muestran la matriz resultante del modelo de evaluación. En la sección 5, se analizan algunos trabajos relacionados. Finalmente, en la sección 6, se presentan las conclusiones y líneas de trabajo futuro sobre la base de los resultados obtenidos.

2. MODELO DE EVALUACIÓN DE HERRAMIENTAS DE ANÁLISIS DE REQUERIMIENTOS.

2.1 Definición de Requerimientos

Un requerimiento es algo que el producto debe hacer o una cualidad que el producto debe tener. Un requerimiento existe ya sea porque el tipo de producto demanda ciertas necesidades o cualidades, o porque el cliente desea que ese requerimiento sea parte del producto entregado.

2.2 Actividades de la Ingeniería de Requerimientos.

Según Mead [1] dentro de la IR existen cuatro actividades básicas que se tienen que llevar a cabo para completar el proceso (ver Fig. 1). Estas actividades ayudan a reconocer la importancia que tiene para el desarrollo de un proyecto de software realizar una especificación y administración adecuada de los requerimientos de los clientes o usuarios..Las actividades se describen a continuación:

Recolección: Representa el comienzo de cada ciclo. Involucra el descubrimiento de los requerimientos del sistema. Se trabaja junto al cliente.

Análisis: Se enfoca en descubrir problemas con los requerimientos del sistema identificados hasta el momento.

Especificación: Se documentan los requerimientos acordados con el cliente, en un nivel apropiado de detalle.

Validación: Su objetivo es, ratificar los requerimientos, es decir, verificar todos los requerimientos que aparecen en el documento especificado para asegurarse que representan una descripción, por lo menos, aceptable del sistema que se debe implementar. Esto implica verificar que los requerimientos sean consistentes y que estén completos.

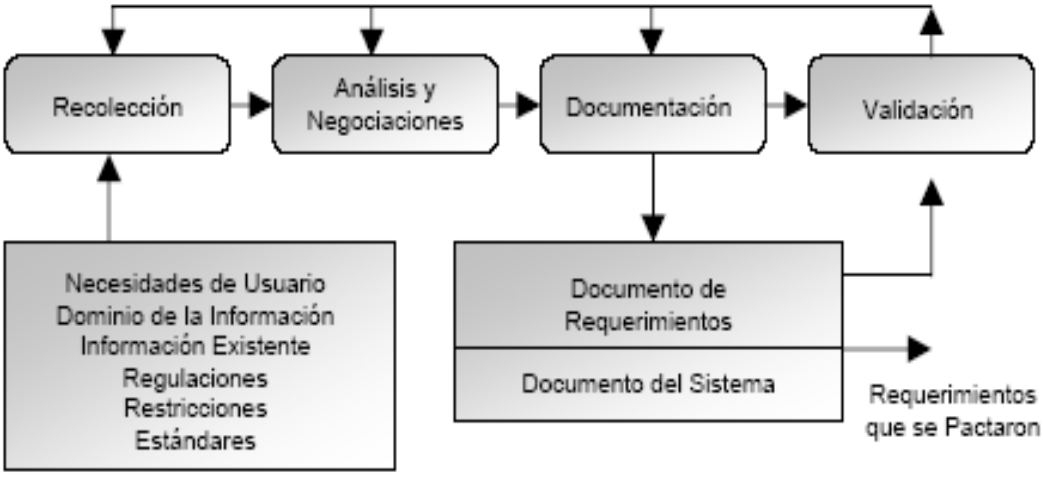

Figura 1 "Actividades de la IR" según Mead [1].

El papel del usuario es crucial en todo este proceso, tanto para transmitir conocimiento como para certificar que el analista comprende el problema, en el marco de un dominio de problema específico (ver Fig. 2).

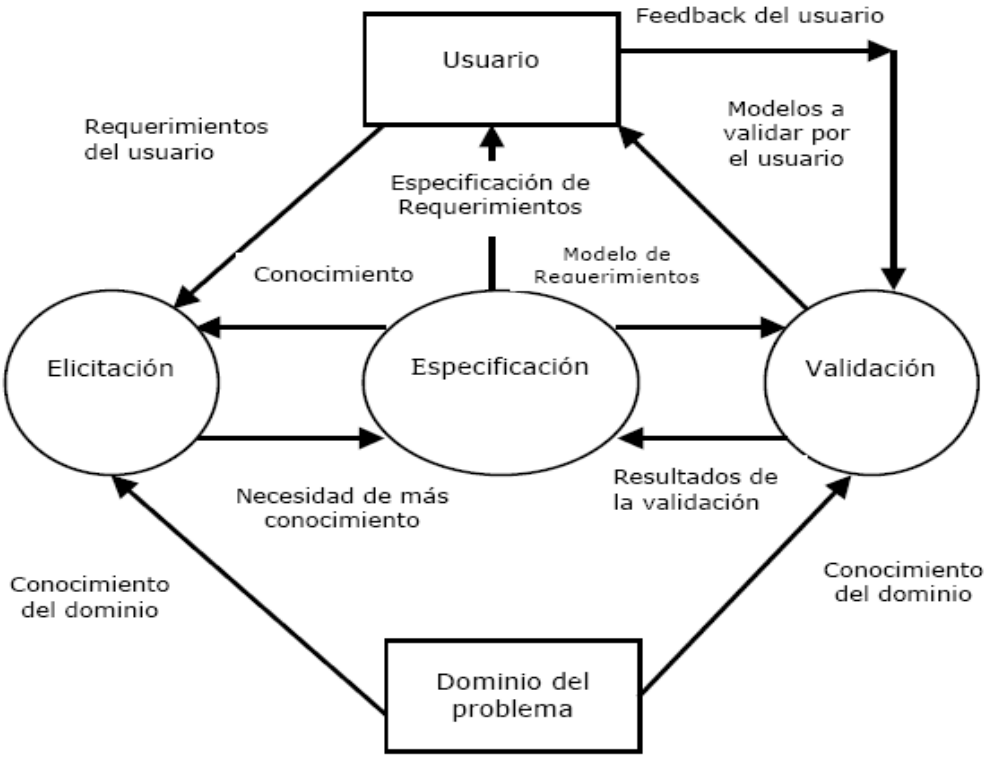

Figura 2. "Etapas del ciclo de vida de la IR" según Loucopoulos [2]

2.3 Herramientas Case para análisis de requerimientos.

Como herramienta CASE (*Ingeniería del Software Asistida por Computadora*) se le conoce a todo aquel software que es usado para ayudar a las actividades del proceso de desarrollo del software, en donde se ubica la ingeniería de requerimientos. Estas herramientas se concentran en capturar requerimientos, administrarlos y producir una especificación de requisitos. Estas herramientas permiten no solo almacenar e indexar los requisitos y casos de uso de un sistema, si no también poder apreciar sus relaciones y lograr denotar el impacto que sufriría un proyecto en general a raíz de los cambios que se planteen a lo largo del proceso. Esto posibilitaría entonces a los líderes del proyecto el poder observar todas las etapas del análisis y determinar cuellos de botella o problemas que se puedan dar a lo largo de dicho proceso, permitiendo mejorar la productividad de los equipos de trabajo.

2.4 ISO 25000 Evaluación de Software de Calidad.

Proporciona orientación para el uso de la nueva serie de Normas Internacionales de productos de software denominada Requisitos de calidad y Evaluación. El propósito de esta guía es proporcionar una visión general de modelos de referencia común y definiciones, así como la relación entre los documentos.

3. DISEÑO DEL MODELO DE EVALUACIÓN

En esta sección se describe el proceso utilizado para obtener un modelo de evaluación de herramientas de análisis de requerimientos.

Tomando como referencia la norma ISO 25000 se eligió algunos parámetros importantes para la evaluación, dentro de cada parámetro se resaltaron algunas características que se vieron necesarias para evaluar las herramientas de análisis de requerimientos. A continuación se detallan los parámetros y características considerados para realizar el modelo.

TABLA 1. "Características y Sub-características de Calidad según ISO 25000[3]"

Características	-	Atributo
Funcionalidad	-	**Trazabilidad**
	-	**Áreas de la IR**
	-	**Interfaz Web**
	-	**Priorización de Requisitos**
	-	**Líneas Base**
	-	**Atributos de los requisitos**
	-	**Retroalimentación visual**
	-	**Validación de Requisitos**
	-	**Sistema de Cambios**
	-	**Integración con otras herramientas**
	-	**Generación de archivos de texto**
	-	**Inclusión de casos de uso y Reglas de negocio**
Fiabilidad	-	Tolerancia a fallos
	-	Recuperabilidad
Usabilidad	-	Comprensibilidad
	-	Capacidad de aprendizaje
	-	Inclusión de elementos de aprendizaje
Eficiencia	-	Tiempo de Respuesta
	-	Utilización de Recursos
Mantenibilidad	-	Capacidad de cambio
Portabilidad	-	Adaptabilidad
	-	Capacidad de instalación.

Una vez elegidas las características fue necesario asignarles un valor del 0 al 5 que nos indicaría la importancia que tiene cada característica en la evaluación, esto se realizó con una matriz de ponderación que nos permitió asignar a cada número lo que significa, como por ejemplo.

TABLA 2. "Modelo de cómo se midió la importancia"

Áreas para la definición de requerimientos que soporta.	
0	No son importantes las áreas de captura, análisis, especificación, validación.
1	Es importante el área de captura de requisitos, ya que se recopila la información facilitada por los usuarios, no importa el área de análisis, especificación y validación.
2	Es importante el área de análisis de requisitos, ya que es la comprensión del problema planteado por el usuario, no es importante el área de captura, análisis, especificación y validación.
3	Es importante el área de especificación de requisitos, ya que es la descripción de las funcionalidades del sistema y su comportamiento en relación con el exterior, no importa el área de captura, análisis, especificación.
4	Es importante el área de validación de requisitos, ya que ahí es la comprobación de la adecuación de la solución especificada a los requisitos que debe resolver, no importa el área de captura, análisis, y especificación.
5	Cumple todas las áreas de captura, análisis, especificación y validación.

Dependiendo del evaluador, si uno desea tener software de calidad en el que cada característica es igualmente importante se pondría una importancia de 5 a todas las características.

Después se elabora la columna de cumplimiento en la que de acuerdo a los resultados obtenidos luego de aplicar el caso práctico se evalúa con una escala del 0 al 4, en la que 0 no cumple, 1 cumplimiento bajo, 2 cumple parcialmente, 3 cumple, 4 cumple totalmente.

De esta manera se generó un una matriz de las herramientas con las características, en la cual obtendríamos la valoración en cada parámetro multiplicando la importancia por el cumplimiento.

4. EVALUACION DE RESULTADOS

Al evaluar cada una de las herramientas, sus características, los valores de importancia que tiene cada característica en el proceso de ingeniería de requisitos y en base a la aplicación de caso práctico se evidenció su nivel de cumplimiento obteniendo como resultado una matriz que se muestra en la Tabla 3.

Aplicando el modelo establecido a las tres herramientas generó los siguientes resultados (ver Tabla 4):

Evaluar la calidad de un producto software es una tarea compleja, ya que no se han logrado establecer parámetros que sean fácilmente cuantificables, haber fundamentado este estudio en una norma internacional le ha aportado madurez al mismo.

En cuanto a las herramientas evaluadas se pudo obtener sus fortalezas, debilidades, ventajas y desventajas así es más fácil escoger una de ellas al momento de administrar los requisitos., permitiendo tener un mayor control en proyectos complejos, reducir costos y retrasos en los proyectos, se debe elegir una herramienta de acuerdo al presupuesto y tamaño del proyecto:

TABLA 3. "Resultados de la matriz"

Parámetros	Herramienta x		Herramienta y	
	Importancia	Cumplimiento	Importancia	Cumplimiento
Funcionalidad				
Trazabilidad	0…5	1…4	0…5	1…4
Áreas de la IR	0…5	1…4	0…5	1…4
Interfaz Web	0…5	1…4	0…5	1…4
Priorización de Requisitos	0…5	1…4	0…5	1…4
Líneas Base	0…5	1…4	0…5	1…4
Atributo de los requisitos	0…5	1…4	0…5	1…4
Retroalimentación visual	0…5	1…4	0…5	1…4
Validación de requisitos	0…5	1…4	0…5	1…4
Sistema de Cambios	0…5	1…4	0…5	1…4
Integración con otras herramientas	0…5	1…4	0…5	1…4
Generación de archivos texto	0…5	1…4	0…5	1…4
Inclusión de casos de uso y reglas de negocio	0…5	1…4	0…5	1…4
Definición usuarios y grupos de usuarios	0…5	1…4	0…5	1…4
Fiabilidad				
Tolerancia a fallos	0…5	1…4	0…5	1…4
Recuperabilidad	0…5	1…4	0…5	1…4
Usabilidad				
Capacidad de aprendizaje	0…5	1…4	0…5	1…4
Inclusión de elementos de aprendizaje	0…5	1…4	0…5	1…4
Comprensibilidad	0…5	1…4	0…5	1…4
Eficiencia				
Tiempo de respuesta	0…5	1…4	0…5	1…4
Utilización de recursos	0…5	1…4	0…5	1…4
Mantenibilidad				
Capacidad de cambios	0…5	1…4	0…5	1…4
Portabilidad				
Adaptabilidad	0…5	1…4	0…5	1…4
Capacidad de instalación	0…5	1…4	0…5	1…4

TABLA 4. "Resultados de la matriz aplicado a las tres herramientas"

	Importancia	Calificación	RequisitePro	Importancia	Calificación	CaliberRM	Importancia	Calificación	Importancia	Calificación	Doors
Funcionalidad											
Define relaciones de trazabilidad.	5	3	15	5	4	20	5	4	20	4	20
Integración con herramientas de pruebas, diseño y administración de proyectos	2	4	8	2	4	8	2	4	8	4	8
Áreas para la definición de requerimientos que soporta	5	3	5	5	4	20	5	3	15	4	20
Permite priorizar los requisitos	5	4	20	5	4	20	5	4	20	4	20
Permite la generación de archivos de texto	1	4	4	1	4	4	1	4	4	4	4
Permite Validar los requisitos	5	2	10	5	4	20	5	4	20	4	20
Incluye un sistema de propuestas de cambio embebido	5	4	20	5	4	20	5	4	20	4	20
Definición de Líneas Base de requerimientos.	5	4	20	5	4	20	5	3	15	4	20
Permite la generación de interfaces	2	0	0	2	4	8	2	1	2	4	8
Fiabilidad											
Tolerancia a fallos	3	4	12	3	4	12	3	4	12	4	12
Recuperabilidad	2	2	4	2	1	2	2	4	8	4	8
Usabilidad											
Capacidad de aprendizaje	1	3	3	1	3	3	1	3	3	4	4
Comprensibilidad	5	4	20	5	4	20	5	4	20	4	20
Incluye elementos para el aprendizaje como tutoriales y proyectos de ejemplo	1	4	4	1	2	2	1	3	3	4	4
Eficiencia											
Tiempo de respuesta	2	4	8	2	4	8	2	4	8	4	8
Utilización de recursos	2	2	4	2	3	6	2	3	6	4	8

5. TRABAJOS RELACIONADOS

Aunque exista una diversidad de trabajos relacionados, en esta sección se han incluido los más relevantes, que se han encontrado durante la investigación:

En lo que se refiere a ingeniera de requerimientos, el trabajo presentado por Hadad [4], presenta una estrategia en la Ingeniería de Requisitos, denominada SDRES, que intenta abordar temas poco tratados en la práctica real, tales como los cambios constantes en los requisitos, defectos del software originados en los

requisitos, el contexto organizacional que rodea al sistema de software y la consideración de requisitos de calidad.

En lo que se refiere a las herramientas de gestión de requisitos, existe el trabajo propuesto por McDonald [5], presentan una definición de los perfiles de las herramientas de gestión de requisitos, que abarca la definición de algunas herramientas y características que se deberían evaluar.

En relación a la ingeniería de requerimientos, Ayala Ramírez [6], presentan la IE aplicada al desarrollo de sistema de información, que especifican los buenos métodos y técnicas que se debe utilizar, así como lo que deben reflejar los requerimientos, resaltando las técnicas y herramientas que nos pueden ayudar en este proceso.

6. CONCLUSIONES Y TRABAJO FUTURO.

Tomando como referencia la norma ISO 25000 se formuló un modelo de evaluación que permitió realizar un análisis comparativo entre las herramientas *Rational RequisitePro, CaliberRM y Doors*, determinando así sus fortalezas, ventajas y desventajas en el momento de administrar los requisitos. Es significativo resaltar la importancia que tiene la ingeniería de requerimientos en generar una adecuada especificación que contemple claramente y sin ambigüedades los requerimientos del sistema a desarrollar, con el fin primordial de evitar que los proyectos fracasen debido a una mala elaboración de la definición, y especificación de requerimientos. El uso de una herramienta de gestión de requisitos proporciona a la organización un ahorro en costes de especificación y de desarrollo, minimizando el impacto de errores, mejora la calidad mediante un adecuado análisis y gestión de los requisitos, mejora la productividad facilitando la reutilización real desde la especificación, permite especificar sistemas de una forma estructurada y gráfica. Es necesario que las empresas mejoren su proceso de ingeniería de requerimientos, si desean ser competitivas con el desarrollo de software a nivel nacional e internacional: las empresas del mercado exterior exigen altos estándares de calidad que la mayoría de las pequeñas empresas establecidas en el país no puede satisfacer en la actualidad.

Como trabajo futuro, la universidad puede proponer como proyecto de desarrollo una herramienta CASE para la administración de requerimientos y acorde a las necesidades del mercado nacional.

Referencias Bibliográficas

[1.] Mead, Nancy R. Requirement's management and requirements engineering: You can't have one without the other. En: Cutter IT Journal. Vol. 13. No. 5, New York, 2000.

[2.] Loucopoulos, Pericles. "System Requirements Engineering", McGraw-Hill, Estados Unidos, 1995.

[3.] ISO/IEC, ISO/IEC 25000-2005

[4.] Graciela Dora Susana Hadad, Uso de Escenarios en la Derivación de Software, Argentina, Noviembre 2007.

[5.] Bárbara A. McDonald Landázuri, Definición de Perfiles en Herramientas de Gestión de Requisitos, Madrid, Septiembre 2005.

[6.] Beatriz Ayala, Claudia Marcela Ramírez, Lina María Ocampo, La ingeniería de requerimientos aplicada al desarrollo de sistema de información, Sevilla, 2002.

Estudio de la Metodología Midas y la Plataforma Rails para el Desarrollo de un Sistema Web de Control de Proyectos

A. García, C. Hinojosa y R. Reyes
Departamento de Ciencias de la Computación, Escuela Politécnica del Ejército, Sangolquí, Ecuador
andres.garcia@abcos.com.ec, {chinojosa, rreyes} @espe.edu.ec

RESUMEN: El estudio fue desarrollado utilizando la metodología Midas y la plataforma Rails para la creación de un sistema Web de control de proyectos. El análisis de compatibilidad entre la metodología y la plataforma se realizó comparando las iteraciones de Midas con el Modelo Vista Controlador que propone Rails. La elección de estas dos herramientas fue gracias a que Rails permite crear sistemas robustos, confiables y seguros, que pueden ser mantenibles y administrables en corto tiempo. Por otro lado, gracias a la Metodología Midas se pudo realizar de una manera eficiente el análisis, diseño y construcción del sistema de Control de Proyectos, los lineamientos dados por la metodología facilitaron el desarrollo del trabajo y se pudo lograr un proceso de desarrollo adecuado y el resultado fue un sistema de calidad, cumpliendo con las necesidades establecidas por los usuarios. La elección de las tecnologías a utilizar en el desarrollo del presente sistema Web contribuyó al éxito del mismo, se utilizaron herramientas de código abierto: plataforma Rails, lenguaje de programación Ruby, servidor Web Apache y base de datos MySql. No solo se creó un sistema, sino que se mostró las ventajas de utilizar Software Libre en la creación de Sistemas de Información, obteniendo así productos de calidad con mayores funcionalidades, servicios y un ágil desarrollo.

Palabras clave: Modelo vista controlador, desarrollo Web, metodología Midas, Rails

ABSTRACT: The study was conducted using the methodology Midas and the Rails platform to create a Web project control. The analysis of compatibility between the methodology and the platform was made by comparing the iterations of Midas with Model View Controller proposed Rails. The choice of these two tools was thanks to the Rails to create robust, reliable and secure, which can be maintained and managed in a short time. On the other hand, thanks to the Methodology Midas could efficiently perform the analysis, design and construction of the Project Control system, the limits given by the methodology facilitated the development of work and to reach an adequate development process and the result was a quality system complying with the requirements set by users. The choice of technologies used in the development of this Web system contributed to its success, we used open source tools: platform Rails, Ruby programming language, Apache Web server and MySQL database. Not only created a system, but showed the benefits of using Free Software in the creation of Information Systems, thus obtaining quality products with more features, services, and agile development.

Keywords: Model View Controller, Web development, Midas Methodology, Rails Platform

1. INTRODUCCIÓN

El acelerado crecimiento de la tecnología y el surgimiento de nuevos lenguajes de programación, ya sean gratuitos o propietarios, han generado una competitividad considerable entre las plataformas desarrolladas, Por esta razón, *"Ruby on Rails"* presenta como su principal ventaja la sencillez y productividad al momento de desarrollar la programación orientada a objetos [2], permitiendo programar de una manera más intuitiva, a diferencia de sus principales competidores como lo son PHP y Java.

Midas es una metodología para aplicaciones orientadas a la Web que propone la utilización de modelos mediante un proceso iterativo e incremental, dichos procesos permiten un desarrollo de sistemas ahorrando tiempo, ya que utiliza prácticas extraídas de metodologías ágiles como XP (*Xtreme Programming*) [6]. La principal ventaja de la metodología Midas, está en las diferentes iteraciones tanto en las Especificaciones de Requerimientos de Software como en el desarrollo de sistemas, las cuales se han definido para satisfacer las necesidades de los clientes como de los desarrolladores.

La combinación de estas dos tecnologías como lo son Midas y Ruby on Rails representan un reto al momento de desarrollar un sistema Web, ya que se deben aprovechar todas las ventajas que proveen cada una [1].

Frente a este escenario, como contribución, el presente artículo se basa en la presentación de la experiencia obtenida en el desarrollo de una aplicación Web, utilizando la metodología Midas y la plataforma Rails. Para cumplir con los objetivos propuestos, inicialmente se realizó una investigación documental bibliográfica, de donde se obtuvieron los lineamientos, conceptos y procedimientos que permitieron realizar el estudio comparativo y el desarrollo del producto software.

El resto del artículo ha sido organizado como sigue: La sección 2 describe los métodos de la investigación documental y la aplicación de la metodología Midas en la plataforma Rails. La sección 3 detalla la implementación del sistema de control de proyectos con la metodología y la plataforma propuesta. En la sección 4 se muestra la evaluación de los resultados de utilizar Midas y la plataforma Rails para el desarrollo de un sistema de control de proyectos. En la sección 5, se analizan los trabajos relacionados. Finalmente, en la sección 6, se presentan las conclusiones y líneas de trabajo futuro sobre la base de los resultados obtenidos.

2. MÉTODOS

2.1 Método de Investigación Documental

La recopilación de la información se la inició mediante Internet, por ser temas relativamente nuevos había poca documentación. En los foros de discusión y al compartir conocimientos de desarrolladores experimentados, se pudo obtener información relevante.

Fue necesaria la aplicación de una investigación documental bibliográfica de fuentes primarias y secundarias internacionales [4], en donde la metodología Midas y la plataforma Rails han tenido éxito en organizaciones mundialmente reconocidas.

2.2 Aplicación de la metodología Midas en Ruby on Rails

Posteriormente, se definieron las iteraciones de Midas de la siguiente manera:

Dentro de la Iteración 1 (ver Fig. 1) se plasmaron todos los modelos independientes de computación los cuales se definen con los requisitos del sistema y se especifican los casos de uso, esta iteración no interactúa con Ruby on Rails ya que se realizan los diagramas del análisis de requerimientos del sistema [3].

Posteriormente, en la Iteración 2 (ver Fig. 1) se desarrolló el modelo conceptual de datos y se realizó el primer prototipo del sistema. La utilización de Rails en esta iteración es importante porque se diseña las

interfaces de los usuarios y facilita el realizar un prototipo real del sistema gracias a su generación automática de código.

En la Iteración 3 (ver Fig. 1) se definió el modelo conceptual de datos, se implementaron e integraron los diseños lógicos de los datos. Rails facilita la conexión e implementación de la base de datos, ya que se definen las tablas mediante el ingreso de código, y el usuario puede migrar las tablas dentro de la base de datos.

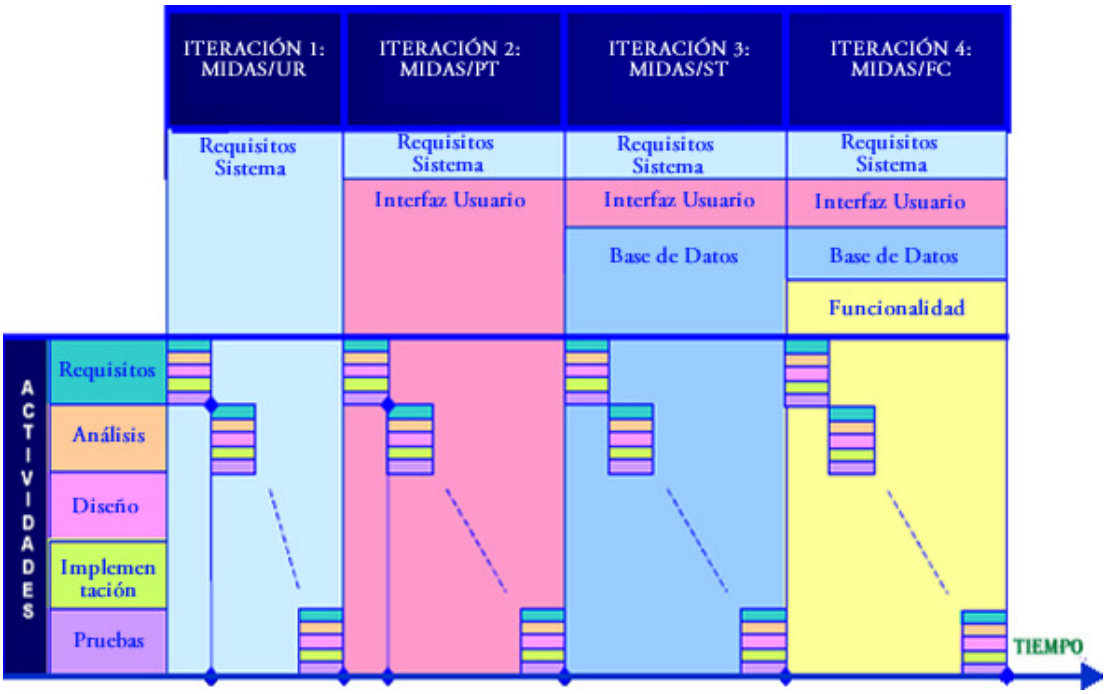

Figura 1: Ciclo de vida del desarrollo del SIW con Midas [6] [8]

Finalmente, en la Iteración 4 (ver Fig. 1) se realizó el modelo de descripción del lenguaje del servicio Web *Web Services Description Language* (WSDL) que define la funcionalidad del sistema basado en el modelo de casos de uso y en el de composición de servicios.

La plataforma Rails utiliza el patrón de desarrollo Modelo Vista Controlador el cual permitió definir y desarrollar cada una de las iteraciones de Midas.

Modelo.- Representa los datos y las relaciones entre ellos (llamado también lógica de negocios). Consiste en las clases que representan a las tablas de la base de datos. Son gestionadas por ActiveRecord. Por lo general, lo único que tiene que hacer el programador es heredar de la clase ActiveRecord::Base, y el programa averiguará automáticamente qué tabla usar y qué columnas tiene. Aquí se aplicó la Iteración 3 de Midas [10].

Vista.- La interfaz de usuario. Representa cómo se muestran los datos de las clases del Controlador. Con frecuencia en las aplicaciones Web la vista consiste en una cantidad mínima de código incluido en HTML. Aquí se aplicó la Iteración 2 de Midas [10].

Controlador.- Quien recibe los eventos solicitados a través de la vista, los empuja al modelo, y genera/refresca a la vista. Describe a la interacción del usuario e invocan a la lógica de la aplicación, que a su vez manipula los datos de las clases del Modelo y muestra los resultados usando las Vistas. En las aplicaciones Web basadas en MVC, los métodos del controlador son invocados por el usuario usando el navegador Web. Aquí se aplicó la Iteración 4 de Midas [10].

3. IMPLEMENTACION

En esta sección se describe la metodología Midas con la plataforma Rails en el desarrollo de un sistema Web de control de proyectos, para lo cual se dividió según las iteraciones de Midas de manera secuencial:

3.1 Iteración 1: Requisitos del sistema

Para proceder a la implementación, en primer lugar se tomaron los requisitos del sistema mediante entrevistas a los diferentes usuarios y actores del sistema [5]. Las técnicas aplicadas fueron principalmente grabar las conversaciones, documentarlas y revisarlas paulatinamente con los clientes. Esta es la única iteración en la que la plataforma Rails no tiene ninguna participación.

3.2 Iteración 2: Interfaz de usuario

Una vez documentados y aprobados los requisitos del sistema, se procede a realizar las interfaces de los usuarios, para lo cual Rails permitió mediante la Vista representar con código HTML y Ruby cada una de las pantallas del sistema [11]. La mezcla de código entre HTML y Ruby fue de gran ayuda, ya que permitió posicionar los elementos de una forma fácil e intuitiva.

Para un mayor control de los archivos, Rails creó la carpeta Views, en la cual se almacenaron todas las interfaces de usuario, en archivos de extensiones .html.erb.

3.3 Iteración 3: Base de datos

Se definió, realizó, implementó e integró los modelos conceptual y lógico de la base de datos, con ayuda de los requisitos del sistema [9]. Rails mediante el Modelo y la migración de la base de datos permitió la implementación del modelo lógico de los datos, dentro de los cuales se validaron los datos para asegurar la integridad y veracidad de los mismos.

Para un mayor control de los archivos, Rails creó la carpeta Models y Database Migrations [15], en las cuales se almacenaron todos los modelos y creaciones de la base de datos, en archivos de extensiones .rb.

3.4 Iteración 4: Funcionalidad

Por último, se desarrolló el sistema tomando en cuenta los casos de uso. En esta iteración se desarrollaron los Controladores de Rails, tomando en cuenta las interfaces de los usuarios y la base de datos. Se integró a todo el Modelo Vista Controlador que ofrece Rails [7] mediante las iteraciones de Midas.

Para un mayor control de los archivos, Rails creó la carpeta Controllers, en la cual se almacenaron todos los controladores del sistema, en archivos de extensiones .rb.

El sistema inicia con una pantalla de ingreso en la que se solicita el nombre de usuario y password asignados por el gerente de sistemas; se da la opción de ingresar a través de dos usuarios:

i. Administrador de proyectos.- Con la opción de ingresar a administrar proyectos y módulos para crear nuevos proyectos, editar uno ya existente, eliminarlo y obtener reportes, consultar estimaciones y avances, además se puede ingresar los valores de conductores de coste. Se pueden crear nuevos módulos, editarlos, eliminarlos, establecer avances, los valores de factores de complejidad y parámetros de medición.

A través del cálculo de los factores de cada proyecto se tiene el factor de complejidad, parámetros de medición y conductor de coste que permiten realizar el cálculo del esfuerzo, tiempo, productividad y personal necesario para el desarrollo.

ii. Gerente de sistemas.- Con la opción de ingresar a administración de proyectos y módulos para hacer consultas acerca del avance y estimaciones de los proyectos ingresados al sistema. Y, administrador de usuarios, para crear, editar o eliminar usuarios.

4. EVALUACIÓN DE RESULTADOS

Como resultado se obtuvo que la metodología Midas y la plataforma Rails son completamente compatibles en su desarrollo, ya que las iteraciones de Midas y el patrón MVC que utiliza Rails se complementan entre sí como lo muestra la Fig. 2.

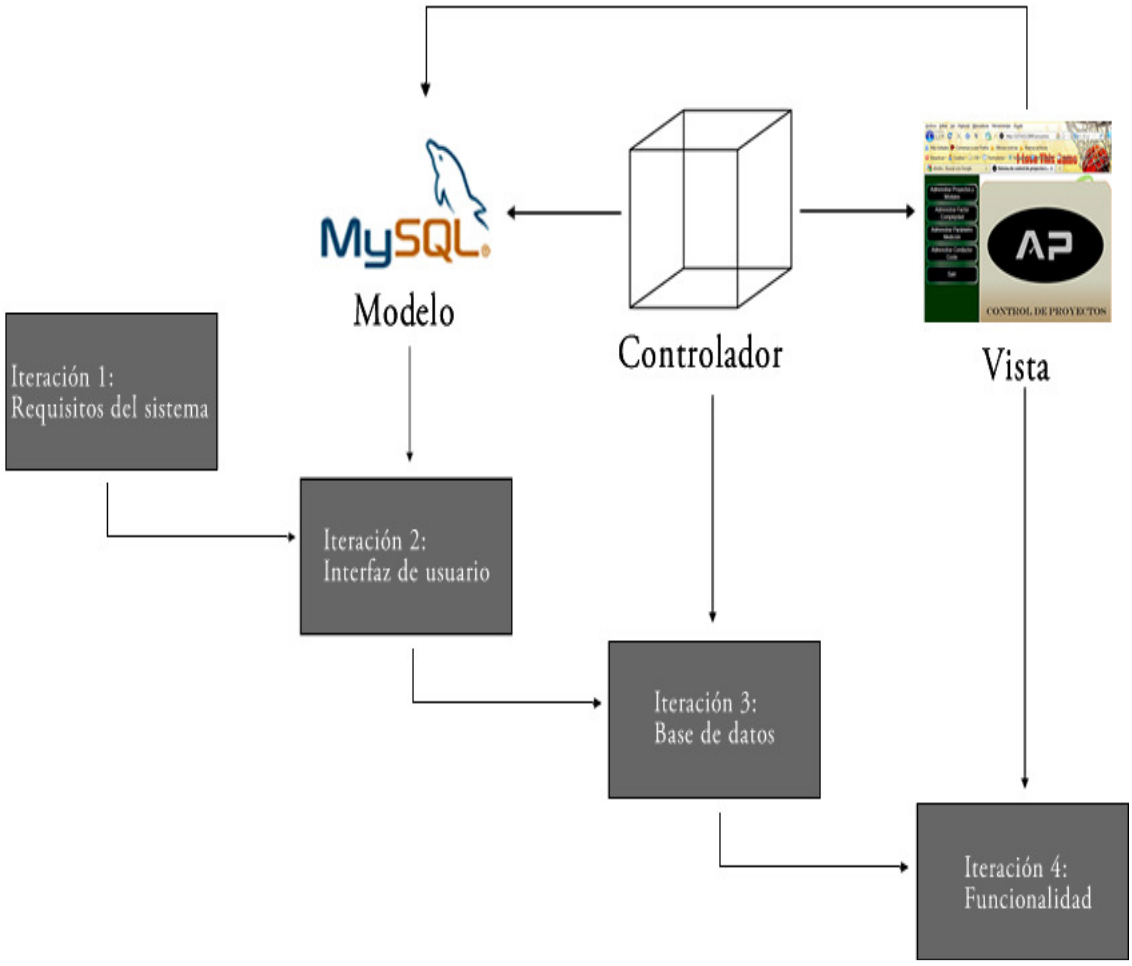

Figura 2: Comparación de la metodología Midas con la plataforma Rails

Además, la plataforma Rails con su lenguaje de programación Ruby frente a sus principales competidores como lo son PHP y Java tiene facilidad de mantenibilidad y mayor velocidad de ejecución, pero su debilidad lo demuestra en la escases de herramientas y escalabilidad como lo demuestra la Fig. 3.

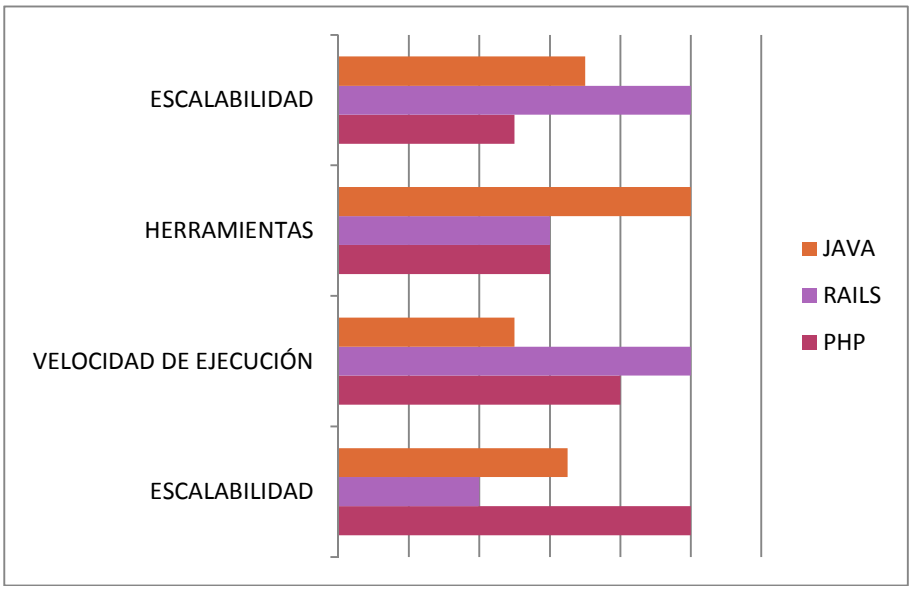

Figura 3: Comparación entre Java vs Php vs Rails [14]

El sistema de control de proyectos permitió obtener resultados reales del esfuerzo, tiempo, productividad y el personal que necesitan para realizar proyectos específicos (ver Tabla 1). Permitió ser eficientes y eficaces en los recursos que destinan a los proyectos que se están ejecutando. Este software estableció parámetros a los sistemas, que por su naturaleza intangible es difícil valorar, permitiendo determinar costos y personal [12].

TABLA 1: Estimaciones de proyectos Web de la Empresa Adgesproyect Cía. Ltda.

Proyecto	Punto De Función	Esfuerzo (Personas / Mes)	Tiempo (Meses)	Productividad (Líneas de Código (LDC) / Esfuerzo)	Personal (Personas)
Floricolas	0.0	0.00	0.00	0.00	0.00
Mapa1	140.97	2.00	3.25	465.11	0.61
secob	130.02	0.08	0.95	429.06	0.08

El sistema de control de proyectos permitió el control del avance de los proyectos (ver Tabla 2) para tomar decisiones sobre planes de contingencia dentro de la empresa y cumplir con los plazos acordados con el cliente evitando pérdidas económicas y problemas legales por el incumplimiento de contratos suscritos [13].

Estableció un proceso de seguimiento de las fases de desarrollo del sistema, verificando el tiempo requerido en cada una.

TABLA 2: Control de avance de un proyecto Web de la Empresa Adgesproyect Cía. Ltda.

cajeros					
Hito	Fecha Inicio	Fecha Entraga	Fecha Finalización	Estado	Observación
asasx	2009-10-06	2009-10-15	2009-10-30	Análisis	asd
asasx	2009-10-06	2009-10-15	2009-10-30	Análisis	asd
111	2009-09-29	2009-10-08	2009-10-23	Análisis	11

5. TRABAJOS RELACIONADOS

La plataforma Rails es relativamente nueva, han pasado 4 años desde su publicación, está en su etapa de despliegue por lo que son escasos los sistemas desarrollados con esta plataforma. En la investigación no se encontraron sistemas de control de proyecto desarrollados utilizando la metodología Midas en la plataforma Rails.

6. CONCLUSIONES Y TRABAJO FUTURO

La aplicación fue desarrollada en la plataforma Rails, lenguaje de programación Ruby y demás herramientas de código abierto para la implementación del sistema a nivel de programación, servidor Web y base de datos. En el lenguaje de programación Ruby aplica el concepto de "todo es objeto", debido a que, incluso los tipos de datos los maneja como objetos independientes lo cual permite la reutilización del código, facilita su mantenimiento y agilita el desarrollo del software. Midas se ha definido mediante un proceso iterativo e incremental, basado en las propuestas de Investigación en Acción, utilizando casos de estudio, los mismos que han servido para determinar los problemas y necesidades, desde un punto de vista metodológico e ingenieril, en el desarrollo de SIW. Pero, además de los casos de estudio, una vez especificada la metodología, Midas se ha aplicado a distintos casos de prueba, lo que ha permitido refinarla. Las iteraciones de Midas son compatibles con el patrón de desarrollo Modelo Vista Controlador que aplica Ruby on Rails, lo cual permite el trabajo óptimo entre estas dos tecnologías haciéndolas complementarias. Se recomienda el uso de NetBeans para el desarrollo del sistema, porque es un Entorno de Desarrollo Integrado de código abierto para diferentes lenguajes de programación entre ellos Ruby utilizando la plataforma Rails, además posee una interfaz gráfica que permite a los desarrolladores el ordenamiento del código y facilita la utilización de la herramienta. Se recomienda grabar las conversaciones con los usuarios al momento de obtener los requerimientos del sistema, de esa manera se facilita la comprensión y documentación de los mismos. Para la seguridad e integridad de los datos se recomienda a los usuarios poner contraseñas seguras y no dejarlas en blanco. Se recomienda realizar pruebas informales periódicamente con los usuarios para garantizar la calidad del producto. Se recomienda la aplicación de Midas utilizando la plataforma Rails ya que sus procesos tanto en las iteraciones como en el patrón MVC son compatibles en su desarrollo.

Como trabajo a futuro se creará un sistema de Inteligencia de Negocios (Business Intelligence BI) para el manejo de estadísticas de los proyectos, saber sus pérdidas y ganancias, además de la ayuda en la toma de decisiones para proyectos posteriores.

AGRADECIMIENTOS

Agradecemos a la Escuela Politécnica del Ejército (ESPE), que a través de la facilidad de recursos económicos, técnicos y humanos, de parte del Departamento de Ciencias de la Computación han hecho realidad la consecución de este proyecto. Un cordial agradecimiento a la Ingeniera Cecilia Hinojosa, y al Ingeniero Rolando Reyes por haber compartido sus conocimiento desinteresadamente, siendo una guía y respaldo a lo largo de la elaboración del proyecto. A mis padres por su amor, comprensión y apoyo incondicional, que han fomentando valores y enseñanzas que nos han ayudado durante toda la carrera. A mis profesores, por sus valiosos consejos y por compartir sus amplios conocimientos y experiencia. Y a todas aquellas personas que de una u otra forma, colaboraron o participaron en la realización de este proyecto, hago extensivo mi más sincero agradecimiento.

Referencias Bibliográficas

[1]. D. Thomas, C. Fowler, A. Hunt; "Programming Ruby - The Pragmatic Programmer's Guide", 2da Edición, Octubre 2004.
[2]. D. Thomas, D. Hansson; "Agile Web Development with Rails", 2005.
[3]. C. Hibbs, B. Tale; "Ruby on Rails: Up and Running", 2006.
[4]. P. Roger, "Ingeniería De Software Un Enfoque Práctico", Mc. Graw Hill, Madrid – España, 2002.
[5]. S. Perdita, P. Rob, W. Addison; "Utilización de UML en ingeniería de software con objetos y componentes", Octubre 2007.
[6]. P. Cáceres; "An approach for Navigation Model Construction from the Use Cases Model".
[7]. O. Bini, "Practical JRuby on Rails Web 2.0 Projects: Bringing Ruby on Rails to the Java™ Platform", 2007.
[8]. V. De Castro, J. Cavero, B. Vela, P. Cáceres; "Seminario de Lenguajes y Sistemas Informáticos", 22 de enero de 2003.
[9]. OMG. Object Management Group, en: http://www.omg.org/.
[10]. Desarrollo Orientado a Objetos con UML, en: http://www.clikear.com/manuales/uml/.
[11]. Linux para todos, en: http://www.linuxparatodos.net/portal/staticpages/index.php?page=servidor-Web/.
[12]. Somerville, Ian (2005). Ingeniería del software. Séptima edición. Editorial Pearson Educación.
[13]. I., Booch, G. y Rumbaugh J. Addison Wesley; "El Proceso Unificado de Desarrollo de Software", 2001.
[14]. E. Marcos Dykinson; "Investigación en Ingeniería del Software vs. Desarrollo Software", 2003.
[15]. W3C Web Services Description Language (WSDL); "W3C Working Draft", marzo 2003, en http://www.w3.org.

Evaluación y Mitigación de Ataques Reales a Redes IP utilizando Tecnologías de Virtualización de Libre Distribución

W. Fuertes, P. Zapata, L. Ayala y M. Mejía
Dirección de Postgrado, Escuela Politécnica del Ejército, Sangolquí - Ecuador
wfuertesd@espe.edu.ec, lzapata@ups.edu.ec, flacoleas@hormail.com, mike_m78@hotmail.com

RESUMEN: Las redes teleinformáticas están expuestas a ataques e intrusiones que pueden dejar inoperativos los recursos y causar pérdidas de la imagen, productividad, credibilidad y competitividad, provocando perjuicios económicos que podrían comprometer la continuidad del negocio. La presente investigación se enfoca en la evaluación de diversos ataques reales de redes IP utilizando plataformas de virtualización *VMware Player 3.01 y Virtual Box 3.2,* con el fin de establecer mecanismos de seguridad para mitigarlos. Para llevarlo a cabo, se diseñó e implementó varias topologías de experimentación utilizando entornos de red virtuales, dentro de las cuales se probaron el escaneo de puertos, fuerza bruta, y suplantación de identidad, tanto en una red de área local como en una extendida. Para cada topología, se utilizó diferente software libre tanto para producir el ataque como para obtener el flujo de tráfico, evaluándose el tiempo que se tarda y las consecuencias en el rendimiento de la red causadas por ataque. Para contrarrestar dichos ataques, se desarrolló un programa en Shell script que sea capaz de detectar, controlar y mitigar los ataques mencionados de manera programable y constante. Los resultados muestran la funcionalidad de esta investigación que reduce las amenazas y vulnerabilidades de las seguridades de las redes de información.

Palabras clave: Ataques de seguridad, evaluación, mitigación, tecnologías de virtualización

ABSTRACT: IP networks Attacks can collapse the continuity of business services affecting its image and causing important economic losses. This research focuses on the evaluation of several IP networking real attacks using virtualization platforms *VMware Player 3.01 y Virtual Box 3.2,* to provide security mechanisms to mitigate them. To carry out this work, we designed and implemented several experimentation topologies using virtual network environments, within which were tested port scans, brute force and spoofing, both on a local area network as wide area network. For each topology, different free open source software was used both to produce the attack and to obtain the traffic flow, evaluating the consequences of these attacks. To deal with such attacks, we developed a demon program that is able to prevent, detect and mitigate these attacks mentioned. The results show the functionality of this research that reduces threats and vulnerabilities in networks security.

Keywords: security attacks, virtualization technology

1. INTRODUCCION

Las redes teleinformáticas están expuestas a ataques e intrusiones que pueden dejar inoperativos los recursos y causar pérdidas de la imagen, productividad, credibilidad y competitividad, provocando perjuicios económicos que podrían comprometer la continuidad del negocio [1.]. Esta incertidumbre sigue agravándose, pues continúan apareciendo diversas amenazas, vulnerabilidades y tipos de ataques que implican hurto, modificación, espionaje, interrupción, falsificación, denegación de servicios etc., perjudicando directamente a los negocios que son altamente dependientes de sus sistemas y redes de información [2.].

Para prevenir y contrarrestar una amplia gama de amenazas a las seguridades de las redes, es necesario conocer las vulnerabilidades de las empresas e identificar diversos tipos de ataques. Para manejar esta situación se propone crear un ambiente de red controlado con los componentes necesarios que detecten ataques maliciosos, para analizarlos y contrarrestarlos. Una primera alternativa sería mediante equipos reales, sin embargo esto encarecería la solución y pondría en riesgo la red en producción. Otra alternativa sería utilizar máquinas virtuales, con las cuales es posible reducir costos de inversión de hardware, costos de mantenimiento, costo y tiempo de experimentación y sobre todo reduciría el riesgo del colapso de la red en producción [3.].

En este contexto, la comunidad científica ha mostrado un creciente interés en investigar e implementar soluciones para disminuir los ataques de seguridad a la redes aprovechado las tecnologías de virtualización. El trabajo propuesto por Keller y Naues [4.], formula la implementación de un laboratorio colaborativo de seguridad utilizando máquinas virtuales. Li y Mohammed [5.], proponen la integración de las tecnologías de virtualización para la instrucción de seguridad en redes implementando un laboratorio remoto de detección de intrusiones. Otros investigadores [6.][7.], han utilizado el concepto de *Honeynet* basada en máquinas virtuales, como una herramienta de seguridad cuyo propósito es el estudio de las técnicas y motivaciones de los atacantes al romper los sistemas de seguridad. En este mismo ámbito [8.][9.][10.], han utilizado las plataformas de virtualización para recuperación de desastres y mitigación de ataques reales a redes IP.

El presente trabajo tiene como objetivo diseñar e implementar una plataforma de experimentación para evaluar ataques reales de redes IP utilizando plataformas de virtualización de libre distribución, e implementar mecanismos de control y mitigación para contrarrestarlos. Para llevarlo a cabo, se diseñó e implementó diferentes escenarios de experimentación utilizado VMware Player y VirtualBox. Luego se aplicó diversos tipos los ataques a cada escenario creado. Posteriormente se evaluó el impacto que provocan los diversos ataques analizando la información de las trazas. Finalmente se proponen mecanismos de mitigación de cada uno de estos ataques. Todo esto utilizando diversas herramientas de código abierto y de libre distribución.

Como principal contribución de esta investigación se han evaluado diversos ataques utilizando como plataforma de experimentación las tecnologías de virtualización, contrarrestando dichos ataques con programas en Shell script que corren como procesos en segundo plano.

El resto del artículo ha sido organizado de la siguiente manera: La sección 2 presenta el marco teórico que fundamenta esta investigación. En la sección 3 se describe el entorno en el que se desarrollaron los ataques, la configuración de la topología de pruebas y los diversos tipos de ataques evaluados. La sección 4 analiza, evalúa y discute los resultados. En la sección 5, se resume los trabajos relacionados. Finalmente en la sección 6 se establecen las conclusiones sobre la base de los resultados obtenidos y se delimita el trabajo futuro.

2. FUNDAMENTACION

2.1 Virtualización y Escenarios Virtuales de Red como plataforma de experimentación

La *Virtualización* es la forma de particionamiento lógico de un equipo físico en diversas máquinas virtuales, para compartir recursos de hardware, como CPU, memoria, disco duro y dispositivos de entrada y salida [11.]. Esta tecnología permite la ejecución de múltiples máquinas virtuales y sus aplicaciones simultáneamente, siendo una gran alternativa para la implementación de escenarios virtuales de red que permiten la reproducción de la funcionalidad de redes reales, facilitando la evaluación de múltiples ambientes de experimentación y validación de software [12.].

Un *escenario virtual de red* puede ser definido como un conjunto de equipos virtuales (tanto sistemas finales como elementos de red -enrutadores y conmutadores) conectados entre sí en una determina topología desplegada sobre uno o múltiples equipos físicos, que emula un sistema equivalente y cuyo entorno deberá ser percibido como si fuera real [13.].

Para implementar los escenarios virtuales para esta investigación, se ha elegido *VMware Player y VirtualBox* que son plataformas de libre distribución basadas en tecnología de virtualización completa que permiten la creación de máquinas virtuales X86 de 32 y 64 bits y que son muy utilizadas en la industria [14.]. En el caso de *VMware Player*, porque es capaz de crear y desplegar máquinas virtuales, de tal forma que múltiples sistemas operativos pueden ejecutarse sin modificación y al mismo tiempo. VMware Player funciona bajo Microsoft Windows, Linux, NetWare y Solaris [15.]. En el caso de *VirtualBox*, porque es un software que dispone de una interfaz gráfica denominada Virtual Box Manage, la misma que permite crear máquinas virtuales, definiendo sus características virtuales de memoria, disco, teclado, mouse y CDROM, así como la respectiva configuración de red [16.].

2.2 Tipos de ataques a redes IP evaluados

A través de los años se han desarrollado formas cada vez más sofisticadas de ataque a redes IP. Entre los más comunes y que han sido evaluados a lo largo de esta investigación se pueden describir los siguientes:

Escaneo de Puertos, que consiste en el envío de una serie de señales (paquetes), que llegan a la máquina atacada, y ésta responde reenviando otra determinada cantidad de paquetes, que el escaneador decodificará y traducirá. Dicha información consta esencialmente del número IP de la máquina atacada y datos sobre el o los puertos que se encuentran en ese momento abiertos. La aplicación por excelencia para realizar exploración de puertos es *Nmap (Network Mapper)*[17.]

Fuerza Bruta, que es la forma de recuperar una clave probando todas las combinaciones posibles hasta encontrar aquella que permite el acceso. Consiste en generar el diccionario (hash) de todas las posibles combinaciones y compararlas con el patrón (hash) que permita el acceso [18.]. Técnicamente, el término Hash se refiere a una función o método para generar claves que representen de manera casi unívoca a un documento, registro, archivo, etc., [19.]. Una manera eficiente de realizar ataque de fuerza bruta es mediante el uso de diccionarios de contraseñas. Los ataques tradicionales más conocidos de fuerza bruta son *Jhon the Ripper [20.] e Hydra*.

Suplantación de Identidad (Spoofing), que consiste en aplicar técnicas de suplantación de identidad generalmente con usos maliciosos o de investigación [21.]. Existen diferentes tipos como el IP spoofing, ARP spoofing, DNS spoofing, Web spoofing o e-mail spoofing. Para efecto del presente estudio nos hemos enfocado al ARP spoofing. ARP Spoofing hace referencia a la construcción de tramas de solicitud y respuesta ARP falseadas, de forma que en una red local se puede forzar a una determinada máquina a que envíe los paquetes a un host atacante en lugar de hacerlo a su destino legítimo.

Los métodos de ataque descritos serán evaluados en los escenarios virtuales de red cuya topología será descrita en la siguiente sección.

3. CONFIGURACION DEL EXPERIMENTO

3.1. Diseño y configuración del escenario

Se ha diseñado una topología de prueba tanto con VMware Player como con VirtualBox, aplicando las mismas condiciones y parámetros de configuración para ambas plataformas, tomado como modelos aquellos escenarios de uso más común en pequeña y mediana organización. A continuación, se ha implementado dicha topología donde los equipos involucrados, ya sean virtuales o físicos, comparten un mismo espacio de direcciones IP. La Fig. 1 representa el caso real en el cual una red LAN/WAN es sometida a ataques IP y los atacantes son usuarios de la Intranet o del Internet (véase Fig.1).

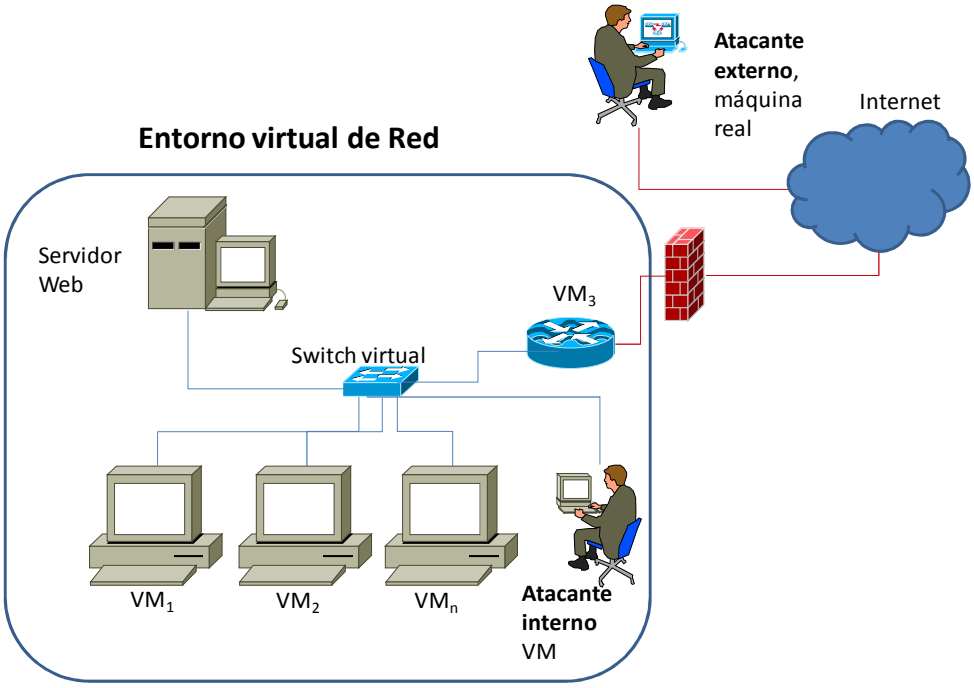

Figura 1. Diseño de la topología de prueba

3.2. Implementación del escenario

Todas las pruebas se desarrollaron sobre Linux Ubuntu Server -i386, en un computador Pentium Intel core duo, RAM de 4GB y una partición Ext3 de 120 GB. En todas las VMs se instaló el mismo sistema de ficheros y el mismo kernel.

El procedimiento utilizado para implementar el experimento consistió en los siguientes pasos: instalación de VMware Player y VirtualBox, creación de máquinas virtuales en cada herramienta de virtualización, direccionamiento IP, configuración de servicios Web y Ssh, creación y aplicación de algoritmos para arranque automático en shell script del escenario, sincronización de reloj con NTP (Network Time Protocol), configuración del ataque y aplicación del algoritmo o aplicación de software para análisis de tráfico.

Para la captura de tráfico de los dos experimentos que se exponen a continuación, se utilizó Tcpdump [22.]. Tcpdump es una herramienta en línea de comandos cuya utilidad principal es analizar el tráfico que circula por la red.

3.3. Implementación de los ataques

Para la implementación de cada uno de los ataques, fue necesario instalar algunas herramientas de libre distribución que han permitido generar los diversos ataques y que también han facilitado la captura de tráfico, tanto para Linux como para Windows. La Tabla 1 describe el tipo de ataque y las diversas herramientas utilizadas en esta plataforma de experimentación.

TABLA 1. RESUMEN DE HERRAMIENTAS UTILIZADAS PARA LA EJECUCION DE LOS ATAQUES.

Nro. Ataque	*Descripción*	*Sistema Operativo*	*Software para el ataque*	*Software para obtener el Flujo de tráfico*
1	Escaneo de Puertos	Ubuntu	Nmap	Tcpdump
		Windows	Zenmap	Ettercap
2	Fuerza Bruta	Ubuntu	Medusa Linux_hash_password.py	
		Windows	John the Ripper	
3	Suplantación de Identidad	Ubuntu	Nemesis	Ettercap
		Windows	Cain & Abel	Cain&Abel

3.3.1 Escaneo de Puertos

Para la generación de este ataque se utilizó *Nmap [23.]*, el mismo que con sus opciones específicas, fue capaz de detectar equipos conectados, puertos abiertos, servicios y aplicaciones en ejecución, el tipo de sistema operativo, el firewall, entre los principales.

Tanto para el caso del *atacante interno*, como externo se probaron algunas opciones de *nmap*, obteniendo una diferencia poco significativa referente a la velocidad de escaneo. Los tipos de scaneo seleccionados fueron: el *ACK scan* que permite identificar de forma precisa cuándo un puerto se encuentra en estado silencioso; y *TCP connect()* que intenta establecer una conexión con cada uno de los puertos del host a escanear, es muy rápido y no se necesita privilegios de root para poder efectuar el escaneo.

3.3.2 Ataque de fuerza bruta

Para la generación de este ataque se aplicó el tradicional *Jhon the Ripper*, el mismo que tiene como objetivo medir el nivel de seguridad de las contraseñas asignadas, utilizando la topología de experimentación basada en plataformas de virtualización descritas en el apartado 3.1. Para aplicarlo, se requiere de un fichero de contraseñas cifradas que se genera para el caso de Ubuntu con el comando *unshadow* y para Windows a través del programa *pwdump*. Una vez generado el fichero, el programa *Jhon the Ripper* empieza a trabajar automáticamente y va mostrando por pantallas las contraseñas que haya ido descifrando sobre dicho archivo.

3.3.3 Ataque de Suplantación de Identidad

Para la generación de este ataque se utilizó *Némesis* [24.], que es una herramienta mediante consola de comandos, con la cual se puede crear diferentes tipos de ataques de suplantación de identidad denominados ARP-spoofing. Este ataque consiste en la inyección de diferentes tipos de paquetes (ARP, TCP, UDP, ICMP) y enviarlos por la red. Se trata de la construcción de tramas de solicitud y respuesta ARP modificadas con el objetivo de falsear la tabla ARP de una máquina víctima y forzarla a que envíe los paquetes a un host atacante en lugar de hacerlo a su destino legítimo. ARP es el protocolo encargado de traducir direcciones IP a direcciones MAC para que la comunicación pueda establecerse.

4 EVALUACION DE RESULTADOS

4.1 Escaneo de puertos

En relación al ataque de ***escaneo de puertos***, las Tabla 2 y 3, muestran el tiempo que se tarda un atacante interno y externo en hacer un escaneo de puertos con *Nmap* a un equipo víctima en los escenarios descritos en el apartado 3.1, tanto con VMware, como con VirtualBox con Ubuntu o Windows como sistemas operativos hospedados. .

TABLA 2. Comparación de Resultados al ejecutar el escaneo de un atacante interno.

INTERNO	VIRTUALBOX			VMWARE		
MAQ1	53,39	62,30	45,16	56,91	45,22	52,92
VM1(ub)	67,13	66,11	69,94	69,68	85,64	52,44
VM2(win)	86,14	46,73	78,13	68,63	68,78	56,19

TABLA 3. Comparación de Resultados al ejecutar el escaneo de un atacante externo.

EXTERNO	VIRTUALBOX			VMWARE		
MAQ1	201,42	201,64	201,51	201,59	201,64	201,86
VM1(ub)	202,53	202,78	202,42	202,20	202,39	202,15
VM2(win)	402,44	402,46	402,28	59,58	66,48	56,94

Con el fin de determinar el tiempo máximo que se demora un equipo atacante en hacer un escaneo de puertos a sus equipos víctimas, se realizaron varios ataques a equipos tanto reales como virtuales. La Fig. 2 muestra el histograma y la frecuencia acumulada, en donde se puede apreciar que la mayoría de quipos toma un promedio de 72 segundos en realizar un escaneo *TCP connect()* que es el tipo de escaneo más eficiente.

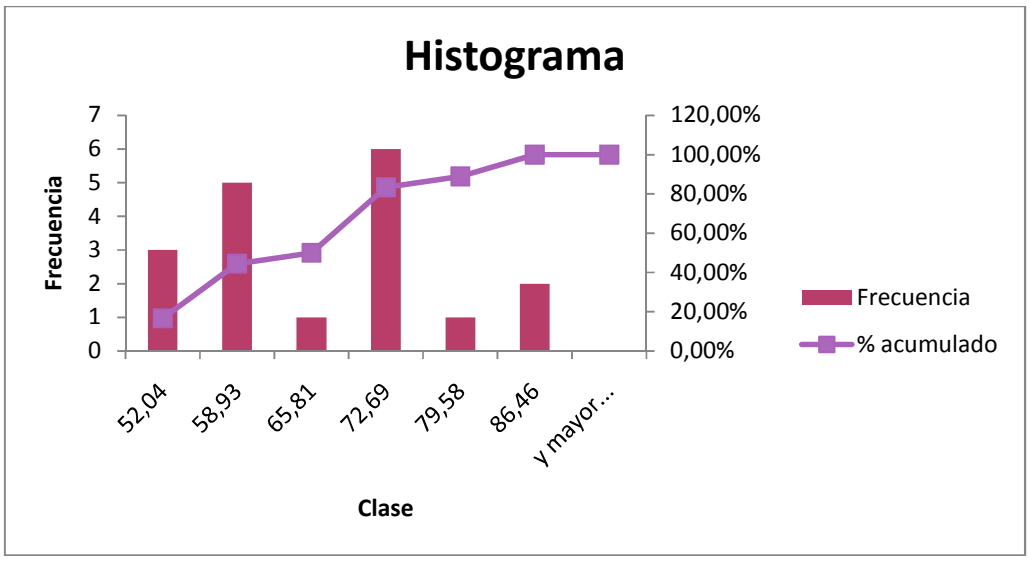

Figura 2. Histograma y % acumulado del escaneo de un atacante externo TCP Connect.

4.2 Fuerza Bruta

En relación al ataque de fuerza bruta, la Fig. 3, muestra el tiempo en segundos que se toma un equipo en descifrar su contraseña utilizando John the Ripper, con sistema operativo Windows. La medición se efectuó sobre claves de cuatro, seis y ocho caracteres de longitud y con combinaciones entre minúsculas, mayúsculas y números, obteniéndose como resultado que mientras más caracteres y combinaciones de letras y números tiene una clave, más tiempo tomará el programa en descifrarla. Cabe mencionar que las pruebas se efectuaron en 5 equipos sobre los cuales se aplicaron las mis mismas claves no habiendo diferencia en el tiempo que tomaron en descifrar cada clave. Adicionalmente se puede concluir que el tiempo que se demora en descifrar una clave también depende de las letras con que se inicie la misma, es decir si una clave está formada por las primeras letras del alfabeto le tomará menos tiempo que una que comience con las últimas letras del alfabeto.

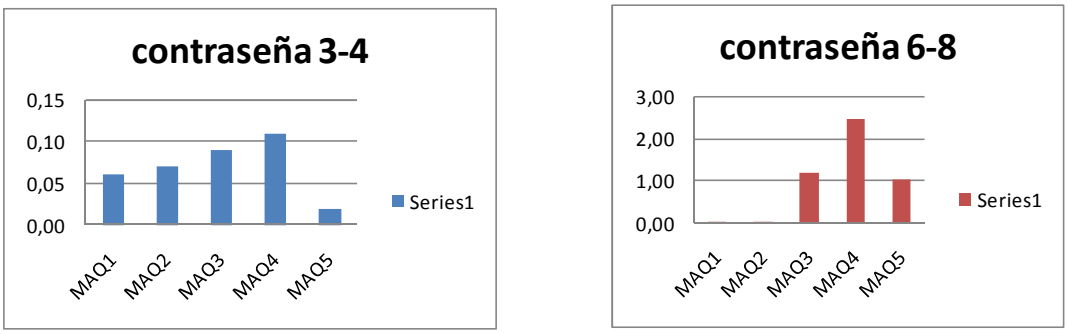

Figura 3. Tiempo que utiliza un equipo en descifrar una clave con John the Ripper. **3.a** contraseña de 3 a cuatro caracteres, **3.b** contraseña de 6 a 8 caracteres.

4.3 Ataque de suplantación de Identidad

En la Fig. 4 se observa los resultados frente a un ataque ARP-Spoofing utilizando Némesis. Con el fin de determinar el consumo de ancho de banda ante este tipo de ataque, se realizaron varias pruebas donde los equipos víctimas fueron máquinas virtuales tanto de Virtual Box como VMWare. Se tomaron 35 muestras en 60 segundos. Como se puede apreciar los equipos víctimas ocupan un ancho de bando de 963 kbps a los 60 segundos.

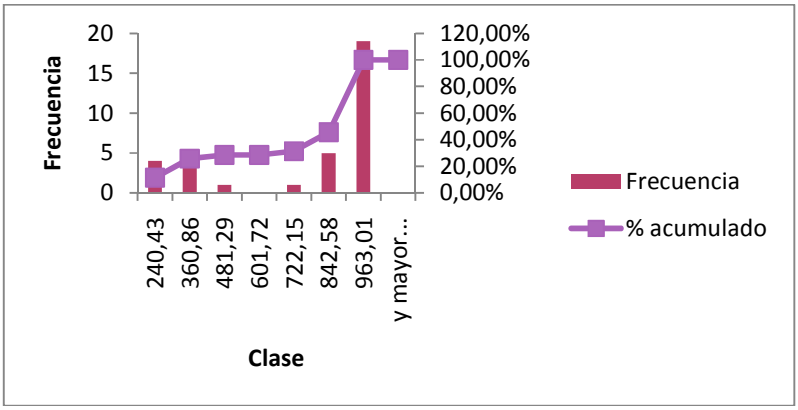

Figura 4. Función de distribución acumulada del consumo de Ancho de Banda frente a un ataque de suplantación de identidad.

4.4 Evaluación del Algoritmo de detección, control y mitigación

Cómo alternativa de solución para detectar, controlar y mitigar los ataques descritos en este artículo, se desarrolló un demonio o administrador regular de procesos en segundo plano que ejecuta comandos programados en Shell scripts a intervalos regulares, basado en la configuración del archivo ***crontab***. Este demonio automatiza los mecanismos de mitigación para controlar los cuatro tipos de ataques evaluados. Luego se procede a registrar la ejecución del monitoreo de manera constante (parametrizable) en el cron ejecutando crontab –e, registrando y definiendo cronogramas (cada minuto por ejemplo).

Para el caso del ataque de escaneo de puertos, suplantación de identidad y denegación de servicios citados en la sección 4, ha sido implementado un firewall de Linux. En concreto es un algoritmo que ante la evidencia de un paquete ICMP, ARP-, *Nmap*, ejecuta un Shell script que activa por consola comandos *Iptables*, que modifican la configuración del firewall de Linux. El algoritmo se basó tanto en *reglas* para filtrar paquetes y decidir si dejarlo pasar o no; así como *cadenas*, que se ejecutan de arriba a abajo hasta que se cumpla una de ellas, que son políticas para decidir qué hacer con los paquetes que no coincidieron con ninguna de las reglas. Una vez que se cumple las condiciones y se activa, se cierra la conexión a dicha IP y por lo tanto se detiene el ataque.

Para el caso del ataque de fuerza bruta, ha sido implementado un mecanismo de autenticación mediante un Shell script que realiza una revisión sobre el archivo de logs de validación de autentificaciones (auth.log). Este busca autentificaciones inválidas y cuando supera el límite definido de fallas (parametrizable) accede al archivo de denegación de hosts para registrar la IP de la máquina que está intentando realizar los accesos fallidos. Una vez que se cumple las condiciones y se activa, se cierra la conexión a dicha IP y por lo tanto se detiene el ataque. La Fig. 5 muestra el diagrama de secuencias del Proceso de mitigación cuando se trata del ataque de fuerza bruta.

Cabe señalar que al aplicarse este demonio, y al repetir los ataques descritos en la sección 4 en la topología de experimentación utilizando tecnologías de virtualización, los resultados muestran la funcionalidad de esta investigación que reduce las amenazas y vulnerabilidades de las redes en producción.

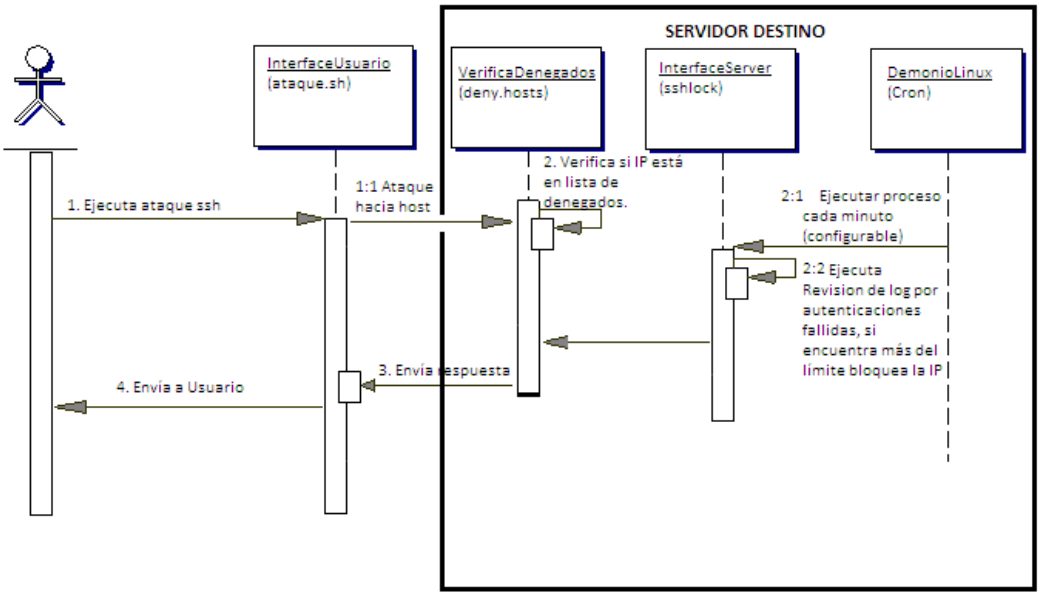

Figura 5. Diagrama de secuencias del algoritmo en shell script para contrarrestar el ataque de fuerza bruta.

5 TRABAJOS RELACIONADOS.

Aunque exista una diversidad de trabajos relacionados, en esta sección se han incluido los más relevantes, que se han encontrado durante la investigación:

En lo que se refiere al ámbito educativo, el trabajo desarrollado por Keller y Naues [4.], expone la implementación de un laboratorio colaborativo de seguridad utilizando máquinas virtuales. Esta investigación permite realizar las tareas de administración de seguridad mediante un Shell remoto, además cuenta con otra interfaz Web que permite saber los resultados de su práctica de laboratorio, y tareas pendientes. En este mismo ámbito, Li y Mohammed [5.], proponen la integración de las tecnologías de virtualización para la instrucción de seguridad en redes implementando un laboratorio remoto de detección de intrusiones. Adicionalmente, [6.][7.] han utilizado el concepto de *Honeynet* basada en máquinas virtuales, como una herramienta de seguridad cuyo propósito es el estudio de las técnicas y motivaciones de los atacantes al romper los sistemas de seguridad. En este mismo contexto, El trabajo propuesto por Damiani [8.], describe un laboratorio virtual basado en tecnología de código abierto, utilizando la plataforma Xen, que tiene como objetivo la configuración de un firewall para proteger el servidor de ataques externos mediante Iptables. Todos estos trabajos han sido utilizados como insumos en esta investigación.

En relación a soluciones de recuperación de desastres mediante virtualización, el trabajo propuesto por [9.], demuestra que el uso de esta tecnología como una opción, debido a que minimizan el uso de servidores y liberan a los administradores del hecho de tener el mismo ambiente de hardware que los servidores en operación, representando una mayor flexibilidad y costos mucho menores de mantenimiento y administración.

En un contexto más cercano al nuestro, el trabajo propuesto por Ferrie [10.], utilizó código malicioso y ataques de denegación de servicio contra máquinas virtuales VMware, VirtualPC, Paralles e Hydra. Sin embargo en este estudio solo se recomiendan pero no se han desarrollado soluciones tangibles. Comparando este trabajo con el nuestro existen dos diferencias fundamentales, la primera hemos realizado la evaluación de diversos ataques de redes y hemos desarrollado e implementado un demonio que permita detectar, controlar y mitigar los ataques evaluados.

6 CONCLUSIONES Y TRABAJO FUTURO

La presente investigación se enfocó en la evaluación de diversos ataques reales de redes IP utilizando plataformas de virtualización. Durante esta investigación, se diseñó e implementó varias topologías de experimentación basadas en entornos virtuales de red. Los tipos de ataques evaluados por ser tradicionales fueron escaneo de puertos, fuerza bruta y suplantación de identidad, tanto en una red de área local como en una red de área extendida. Para cada topología, se utilizó diferente software libre tanto para producir el ataque como para obtener el flujo de tráfico, evaluándose las vulnerabilidades de la red virtual. Para contrarrestar dichos ataques, se desarrolló un set de programas en Shell script que detectó, controló y mitigó los ataques mencionados de manera programable y constante. Los resultados redujeron las amenazas y vulnerabilidades de los ataques en redes en experimentación.

Como trabajo futuro se planea evaluar ataques distribuidos de denegación de servicio, utilizando otros mecanismos de mitigación como la encriptación, sistemas de detección de intrusos y VPNs en un entorno de red virtualizado.

Referencias Bibliográficas

[1.] H. Tipton, M. Krause, "Information Security Management Handbook", Auerbach Publications. Fifth Edition. ISBN: 08493-1997-8

[2.] S. Garfinkel with Gene Spafford Web Security, Privacy & Comerse.. O'Really. Second Edition. ISBN 0-596000-456

[3.] W. Fuertes, J. E. Lopez de Vergara, F. Meneses, "Educational Platform using Virtualization Technologies: Teaching-Learning Applications and Research Uses Cases", In proceedings of II ACE Seminar: Knowledge Construction in Online Collaborative Communities, Albuquerque, NM - USA, October 2009.

[4.] J. Keller, R. Naues, "A Collaborative Virtual Computer Security Lab," e-science, In Proc. Second IEEE International Conference on e-Science and Grid Computing, pp. 126, CA, USA, Dec. 2006

[5.] P. Li, T. Mohammed, "Integration of Virtualization Technology into Network Security Laboratory", In Proc. 38th ASEE/IEEE Frontiers in Education Conference, Saratoga, NY, October, 2008.

[6.] F. Abbasi, R. Harris, "Experiences with a Generation III virtual Honeynet", In Proccedings of the Telecommunication Networks and Applications Conference (ATNAC), 2009 Australasian, Canberra, ACT , ISBN: 978-1-4244-7323-6. May 2009.

[7.] Fermín Galán, David Fernández, "Use of VNUML in Virtual Honeynets Deployment", IX Reunión Española sobre Criptología y Seguridad de la Información (RECSI), Barcelona (Spain), pp. 600-615, September 2006. ISBN: 84-9788-502-3.

[8.] E. Damiani, F. Frati, D. Rebeccani, "The open source virtual lab : a case study". In proceedings of the workshop on free and open source learning environments and tools, hosted by: FOSLET 2006; pp. 5-12, Italy nel 2006.

[9.] Co-innovation lab Tokyo, "Disaster Recovery Solution Using Virtualization Technology", White paper, http://www.cisco.com/en/US/prod/collateral/ps4159/ps6409/ps5990/N037_COIL_en.pdf.

[10.] P. Ferrie, Attacks on Virtual Machine Emulators, Symantec White Paper, 2008.

[11.] F. Galán, D. Fernández, W. Fuertes, M. Gómez and J. E. López de Vergara, "Scenario-based virtual network infrastructure management in research and educationnal testbeds with VNUML,", Annals of Telecommunications, vol. 64(5), pp. 305-323, May 2009.

[12.] Matthews, J., Hapuarachi, W., Deshane, Hu, M. T., Quantifying the Performance Isolation Properties of Virtualization Systems. In Proc. of Workshop on Experimental computer science ExpCS'07, 13–14 June, 2007, San Diego, CA.

[13.] W. Fuertes and J. E. López de Vergara, "An emulation of VoD services using virtual network environments,". In Proc. GI/ITG Workshop on Overlay and Network Virtualization NVWS'09, Kassel-Germany, March 2009.

[14.] W. Fuertes and J. E. López de Vergara, "A quantitative comparison of virtual network environments based on performance measurements,", in Proceedings of the 14th HP Software University Association Workshop, Garching, Munich, Germany, 8-11 July 2007.

[15.] VMware home page, [Online:] http://www.vmware.com

[16.] VirtualBox home page [Online:] http://www.virtualbox.org

[17.] C. Lee, C. Roedel, E. Silenock, "Detection and Characterization of Port Scan Attacks", [Online:] "http://cseWeb.ucsd.edu/users/clbailey/PortScans.pdf

[18.] Hacking: VII Ataques por Fuerza Bruta. [Online:]: http://jbercero.com/index.php?option=com_content&view=article&id=71:hacking-vii-ataques-por-fuerza-bruta&catid=40:hacking-tecnicas-y-contramedidas&Itemid=66

[19.] Laboratorios: Hacking, Técnicas y contramedidas, Ataques por fuerza bruta (Brute Force) III. [Online:] http://labs.dragonjar.org/laboratorios-hacking-tecnicas- fuerza-bruta-brute-force-iii

[20.] Jhon the Ripper 1.7.6., [Online:] www.openwall.com/jhon/

[21.] F. Callegati, W. Cerroni, and M. Ramilli, "Man-in-the-middle attack to the HTTPS protocol," IEEE Security and Privacy, vol. 7, no. 1, pp. 78–81, 2009

[22.] Jacobson, V., Leres, C., and McCanne, S. Tcpdump. Available at anonymous@ftp.ee.lbl.gov

[23.] Nmap, www.nmap.org. Ultima comprobación Octubre de 2010.

[24.] Nemesis, http://nemesis.sourceforge.net/. Ultima comprobación, 20 de octubre de 2010.

[25.] Ethercap, http://ettercap.sourceforge.net/. Ultima comprobación, 21 de octubre de 2010

Gateway para el envío masivo de mensajes cortos de texto (SMS)

R. Montaquiza, F. Romero, R. Fonseca, R. Delgado
Departamento de Ciencias de la computación, Escuela Politécnica del Ejército, Sangolquí, Ecuador
ramiro.montaquiza, felipe.romerol {@gmail.com}, rfonseca, rdelgado { @espe.edu.ec}.

RESUMEN: La ESPE, institución líder en el sistema nacional de educación superior, cuenta con infraestructura tecnológica para solventar de manera efectiva todos los servicios que brinda, sin embargo surgen nuevas necesidades como la oferta de servicios móviles SMS, que es relativamente nueva en el mercado ecuatoriano y más aún en una institución educativa. Estos servicios poseen gran potencial de crecimiento por los beneficios que brindan, tales como movilidad, comodidad, facilidad de uso. Para aprovechar este canal de comunicación, se propone el diseño de un gateway de mensajería SMS sobre una plataforma Web, que permita la difusión masiva de contenido personalizado. Para llevarlo a cabo se ha desarrollado una interfaz utilizando metodologías ágiles sobre plataforma Web con herramientas bajo estándares libres. Como contribución se aspira que la ESPE al disponer de la tecnología para brindar servicios móviles, se posicionaría como pionera entre las Instituciones Educativas. Los resultados de las pruebas demuestran que se logra una comunicación efectiva e inmediata con el usuario.

Palabras claves: servicios móviles, mensajería SMS, metodologías ágiles.

ABSTRACT: ESPE (Army Polytechnic School in Spanish) is a leader establishment in the national higher education system. It has Technological Infrastructure (TI) to effectively support all the services it provides. Nevertheless, there are new needs such as SMS mobile services, which not only are relatively new in the Ecuadorian market, let alone in an educational institution. These services have a great growth potential due to the benefits associated with them, such as mobility, comfort and usability. In order to take advantage of this communication way, this paper proposes a Web based SMS gateway design. This design allows massive broadcast of customized content. With the aim of carrying it out, an interface has been developed using Web based agile development methodologies via free software. As a result of this design, it is aspired that ESPE, having technology for mobile services, becomes a pioneer among educational institutions. Test results have shown an effective and immediate communication with the users.

Keywords: mobile services, SMS messaging, Agile Methodologies

1. INTRODUCCIÓN

El ser humano desde siempre ha encontrado la forma de comunicarse, y paulatinamente ha mejorado esa capacidad ayudado por la tecnología. En este trabajo se describe el servicio SMS desde un punto de vista práctico, haciendo hincapié en cómo es posible realizar aplicaciones que utilicen este servicio.

En este contexto, según [1] el promedio de envío de SMS en el mundo por usuario activo por día es de 4 SMS. En el Ecuador es de 9 SMS por día por usuario activo. Así mismo, de acuerdo a datos de de la Superintendencia de Telecomunicaciones (Supertel) [2] hasta junio 2010 había 14'162.931 millones de abonados de telefonía celular, quiere decir que 99 de cada 100 personas en Ecuador tienen teléfono celular [3].

Actualmente el 98% de los estudiantes de la Carrera de Ingeniería en Sistemas e Informática de la ESPE usan un teléfono celular[4], en el cual encontramos un sin número de servicios que abarcan desde entretenimiento hasta organización personal, de los cuales el más usado después del servicio de voz, es el de SMS.

Frente a este escenario surge la necesidad de integrar a la comunidad de la ESPE a las comunicaciones SMS, por ser este es uno de los medios más efectivos, cómodos y personalizados para poder entregar y recibir información. Por tanto, como contribución, el presente artículo se basa en la definición de una técnica para el envío masivo (Bulk Messaging) de contenido al teléfono celular de los estudiantes.

Para llevarlo a cabo, se propone la implementación de un Gateway SMS. Para ello se ha escogido el uso de Metodologías Ágiles de desarrollo [5] sobre una plataforma Web, con productos de software bajo estándares libres [6].

El resto del artículo se ha organizado como sigue: La sección 2 describe brevemente la tecnología para el envío y recepción de mensajes de texto, así como la metodología usada. La sección 3 detalla el diseño e implementación del gateway SMS. En la sección 4 se muestran los resultados experimentales. En la sección 5, se analizan algunos trabajos relacionados. Finalmente, en la sección 6, se presentan las conclusiones y líneas de trabajo futuro sobre la base de los resultados obtenidos.

2. SERVICIO DE MENSAJES CORTOS DE TEXTO (SMS)

2.1 Generalidades

El servicio SMS permite transferir un mensaje de texto, a través de un Centro de Servicio, entre una estación móvil (celular) y otra entidad, que puede ser otro celular, o puede estar situada en una red fija, en el caso de envío de un mensaje entre dos celulares, ambas partes son estaciones móviles.

Los SMS llegan automáticamente al celular y se almacenan en este, debido a que la transmisión de los SMS se realiza usando los canales de control de GSM, es posible cursar una llamada y recibir un mensaje en forma simultánea.

En la actualidad el organismo responsable del desarrollo y mantenimiento de los estándares GSM y SMS es el 3GPP (Third Generation Partnership Project) [7].

La información que se puede enviar en un SMS es limitada [8]. Un mensaje SMS contiene a lo sumo 140 bytes (1120 bits) de datos, o el equivalente a: **i)** 160 caracteres si la codificación utilizada es: 7-bit. (La codificación 7-bit se utiliza para codificar caracteres Latinos como el alfabeto Inglés); **ii)** 70 caracteres si la codificación utilizada es: 16-bit Unicode UCS2. (Se usa para caracteres no Latinos).

A continuación se describe algunos elementos importantes definidos durante esta investigación:

a) Número Corto para SMS

Son números de teléfono especiales, en Ecuador tienen 3 o 4 dígitos, están diseñados para ser de fácil recordación y de fácil lectura. Se puede utilizar el mismo número en todas las operadoras, ya que depende exclusivamente de la red a la que pertenece, estos códigos son ampliamente utilizados para servicios móviles de valor agregado. La difusión de un gran número de SMS se denomina Bulk.

b) Integrador SMS
Es un intermediario, el Gateway SMS consume un WebService del Integrador SMS, y este a su vez tiene conexión con todas las operadoras, se encarga del enrutamiento los mensajes a la operadora correspondiente. La Fig. 1 muestra el trayecto de un SMS desde que sale del Gateway hasta que llega al teléfono celular:

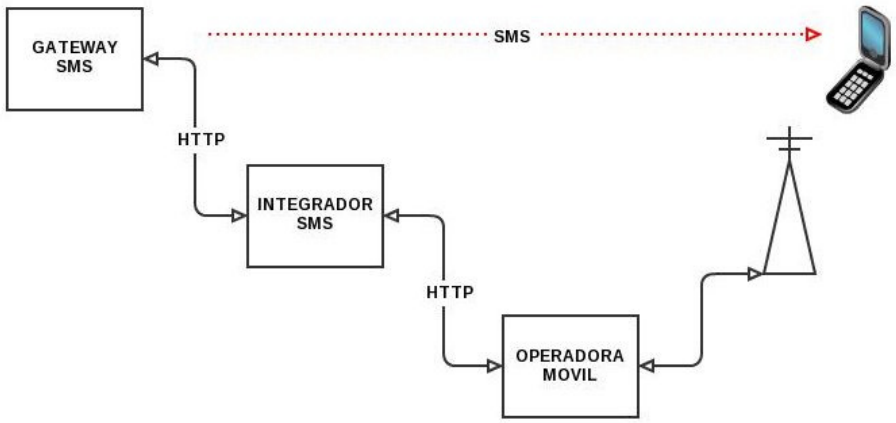

Figura 1. Ruta de un SMS desde el Gateway SMS hasta el teléfono celular

2.2 Tipos y Servicios

Existen dos tipos de SMS disponibles: mensajes de difusión (Cell Broadcast), usados para enviar información de control unidireccionales a los terminales móviles que tengan habilitado el servicio dentro de una celda (point-to-area), y mensajes punto a punto (point-to-point) [9].

Este servicio es del tipo "store and forward", lo cual permite que: si el equipo del destinatario se encuentra apagado, o fuera del área de cobertura, el mensaje sea almacenado en la red hasta que pueda ser entregado al destinatario, el tiempo de almacenamiento depende de la operadora móvil. Se divide en dos servicios básicos: ***i)*** **SM MT** (Short Message Mobile Terminated Point-to-Point). Servicio de entrega de un mensaje desde el Centro de Servicios hasta un celular; ***ii)*** **SM MO** (Short Message Mobile Originated Point-to-Point). Servicio de envío de un mensaje desde un teléfono celular hasta el Centro de Servicios.

2.3 Arquitectura Básica de Red para SMS

SMS proporciona un mecanismo para la transmisión de mensajes cortos hacia y desde dispositivos móviles. El servicio hace uso de un CSMS (Centro SMS), que actúa como un sistema de almacenamiento y transmisión de mensajes cortos.

La red inalámbrica proporciona los mecanismos necesarios para encontrar una o varias estaciones de destino y transporta los mensajes cortos entre los CSMS's y las estaciones inalámbricas (celulares).

En contraste con otros servicios existentes de trasmisión de mensajes de texto, tales como paginación alfanumérica, los elementos de servicio son diseñados para proporcionar la entrega garantizada de los mensajes de texto hasta el punto de destino.

Además, SMS soporta varios mecanismos de entrada que permiten la interconexión con diferentes fuentes y destinos mensaje. Una característica distintiva de este servicio es que un teléfono móvil activo es capaz de recibir o enviar un mensaje corto en cualquier momento, independientemente de si una llamada de voz o de datos está en marcha (en algunas implementaciones, esto puede depender de la capacidad de MSC o CSMS).

SMS también garantiza la entrega de los mensajes cortos por la red, si hay fallas temporales debido a que no se identifican las estaciones receptoras, el mensaje corto es almacenado en el CSMS hasta que el dispositivo de destino esté disponible.

SMS se caracteriza por la entrega de paquetes out-of-band (fuera de la banda) y transferencia de mensajes low-bandwidth (bajo ancho de banda), que se traduce en un medio muy eficiente para la transmisión de breves ráfagas de datos.

La Fig. 2 ilustra la arquitectura de red básica para el envío/recepción de mensajes cortos. A continuación se detallan sus elementos:

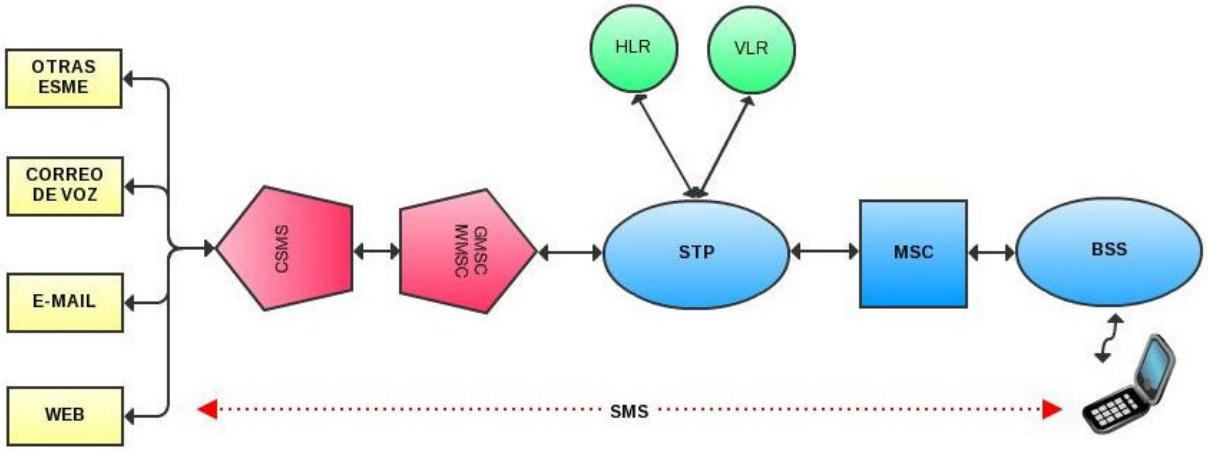

Figura 2. Arquitectura de red básica para el envío/recepción de mensajes cortos

2.3.1 External Short Messaging Entities (ESME)

Un ESME (Entidad Externa de Mensajes Cortos) es un dispositivo que puede recibir o enviar mensajes cortos. La ESME puede estar situada en la red fija, un dispositivo móvil, u otro centro de servicio, y pueden ser: **i)** Correo de Voz; **ii)** WEB; **iii)** E-mail.

2.3.2 Centro SMS (CSMS)

CSMS es una combinación de hardware y software encargado del almacenamiento, la transmisión y reenvío de un mensaje corto entre una SME y el dispositivo móvil.

Normalmente, una solución IN (Intelligent Network : Red Inteligente) como es la IS-41 [10] puede soportar otras aplicaciones en una sola plataforma de hardware y compartir recursos, otro factor a considerar es la facilidad de operación y mantenimiento de la aplicación, así como la flexibilidad necesaria para activar nuevos servicios y la actualización a nuevas versiones de software.

2.3.3 Signal Transfer Point (STP)

Es un elemento de red normalmente disponible en desarrollos IN que permite la interconexión de IS-41 sobre SS7 (Signaling System 7 : Sistema de Señalización 7) [11] con múltiples elementos de la red. Se comunica con los siguientes elementos:

a) Home Location Register (HLR)

Es una base de datos utilizada para el almacenamiento permanente y gestión de las suscripciones y perfiles de servicio, tras ser interrogado por el CSMS, el HLR proporciona la información de enrutamiento para el suscriptor indicado. Si la estación de destino no estaba disponible cuando se intentó entregar el mensaje, el HLR informa al SMSC que la estación es ahora reconocida por la red móvil para ser accedida, y por lo tanto el mensaje puede ser entregado.

b) Visitor Location Register (VLR)

Es una base de datos que contiene información temporal sobre los abonados encasillados en un HLR que hacen roaming en otro HLR. Esta información es necesaria para el MSC para servir a los suscriptores visitantes.

2.3.4 Mobil Switching Center (MSC)

Realiza las funciones de conmutación del sistema y controla las llamadas hacia y desde otros teléfonos y sistemas de datos, el MSC entregará el mensaje corto al móvil de un abonado específico a través de la propia estación base.

2.3.5 Base Station System (BSS)

Todas las funciones relacionadas con la transmisión de señales de radio electromagnéticas entre el MSC y los dispositivos móviles se realizan en la estación base (BS Base Station).

El BS consta de los Controladores de Estación Base (BSC Base Station Controllers) y la estación base de transceptor (BTS Base Transceiver Stations), también conocida como celdas. El BSC puede controlar uno o varios BTSs y se encarga de la correcta asignación de recursos, cuando un suscriptor se mueve de un sector de una BTS a otro, independientemente de que el próximo sector se encuentre dentro de la misma BTS, o en otra distinta.

2.3.6 Tasa de Servicio Efectiva en SMS.

Interesa conocer la tasa efectiva de servicio (throughput) en el envío de un SMS. El throughput está dado por la ecuación (1), donde T_{inf} corresponde al tiempo mínimo necesario para enviar los B_{inf} bits de Información.

$$S=\frac{B_{inf}}{T_{inf}} \quad (1)$$

Dado que en un sub-canal es posible encapsular 184 bits de información, y solamente se puede disponer de sólo un canal para el envío de un SMS, se tiene un límite en caracteres para el uso de un canal. Si el SMS tiene entre 1 y 26 caracteres (codificados a 7 [bits/caracter]) es posible enviar el SMS completamente en un sub-canal. Si la cantidad de caracteres aumenta, también aumenta la cantidad de canales que se van a tener que utilizar para su transmisión, disminuyendo evidentemente el throughput debido al aumento en T_{inf} . Esto es debido a que para el caso de utilizar más de un sub-canal, se deberá esperar la siguiente multi-trama para seguir transmitiendo la información.

2.3.7 Metodología

Para la realización del presente trabajo se han seguido las normativas de la metodología XP (Extreme Programming) [12], en comparación a otras metodologías como RUP es mucho más rápida, ya que la metodología XP conlleva menos protocolo.

En la Tabla 1 se muestran los valores y las prácticas fundamentales de la Metodología XP.

TABLA 1. "Valores y prácticas de la Metodología XP**"**

Valores	Prácticas
Comunicación	Planificación incremental
Coraje	Testing
Simplicidad	Programación en parejas
Retroalimentación	Refactorización
	Diseño simple
	Propiedad colectiva del código
	Integración continua
	Cliente en el equipo
	Releases pequeñas
	Semanas de 40 horas
	Estándares de codificación
	Uso de Metáforas

El Manifiesto Ágil [13] pondera:
- Los individuos e interacciones son más importantes que los procesos y herramientas.
- Software que funcione es más importante que documentación exhaustiva.
- La colaboración con el cliente es más importante que la negociación de contratos.
- La respuesta ante el cambio es más importante que el seguimiento de un plan.

3. DISEÑO E IMPLEMENTACIÓN

En esta sección se explica el procedimiento que se utilizó para implementar la interfaz:

3.1 Especificación de requerimientos

Aquí se muestra la Especificación de Requisitos de Software (ERS) para un Gateway SMS destinado a la carrera de Ingeniería de Sistemas e Informática de la ESPE. Este documento toma como base la norma IEEE Practicas Recomendadas para la Especificación de Requerimientos de Software. (830 -1998).

a) Propósito

Presentar especificación de requisitos de software de un Gateway SMS utilizando software libre, para el envío de notificaciones y procesamiento de requerimientos, aplicado a la carrera de Ingeniería en Sistemas e Informática de la ESPE.

b) Alcance

El Gateway SMS es un producto a través del cual se realizará el envío de mensajes de texto informativos a los estudiantes mediante la utilización de un Gateway SMS. Esto implica la utilización de un Número Corto provisto por las operadoras móviles.

La aplicación será alimentada con los datos proporcionados por el operador del Gateway SMS. Estos datos corresponden básicamente al número de teléfono celular, nombres y apellidos, de los alumnos matriculados en la carrera de sistemas e informática de la ESPE.

c) Perspectiva del Producto

La aplicación del Gateway SMS no se integra ni forma parte de otras aplicaciones del entorno escolástico de la carrera de Ingeniería en Sistemas de la ESPE, es totalmente independiente y autónoma, los datos utilizados para el envío de SMS son proporcionados por el operador del Gateway SMS.

d) Funciones del sistema

La función principal del Gateway SMS es difundir vía mensajes de texto la información suministrada por el operador, la cual es de carácter netamente informativo para los estudiantes de la carrera de Ingeniería de Sistemas e Informática de la ESPE.

El sistema debe cumplir las siguientes características funcionales: **i)** Registro e identificación en el sistema; **ii)** Administración; **iii)** Transacciones; **iv)** Reportes.

e) Características del Usuario

Los usuarios del producto serán netamente de formación media en el uso de tecnología de la información, los cuales según el rol serán definidos como administradores y operadores del sistema, se detalla a continuación los roles: **i) Administrador** Será el encargado del mantenimiento; **ii) Encargado** Podrá realizar tareas de mantenimiento de Plantillas ya creadas; **iii) Usuario** Persona que se encarga de la adaptación y carga de información al sistema.

f) Restricciones generales

El sistema es un medio de masificación de mensajes de texto, por lo cual, el mensaje de texto no deberá sobrepasar los 150 caracteres (restricción del integrador) y el contenido del mismo debe ser netamente informativo, de ninguna manera promocional, tampoco debe ser de carácter sexual, político, religioso, o atentatorio contra la moral.

g) Suposiciones y Dependencias

Se debe tomar en cuenta lo siguiente: **i) Suposiciones** Base de datos actualizada con el n{umero celular del alumno; **ii) Dependencias** Depende de acceso a Internet; además de la interacción con el Integrador de servicios, así como la señal de telefonía celular.

h) Requisitos de las interfaces externas.

Los requisitos son los siguientes: **i) Interfaces de usuario** El sistema estará desarrollado sobre plataforma Web, por lo cual es usuario podrá interactuar con el mismo a través de un navegador Web que soporte la arquitectura definida; **ii) Interfaces de Hardware** El sistema es netamente transaccional, a nivel de software, es decir no se necesita de un hardware en especial para que el sistema funcione; **iii)** Interfaces de software Las interfaces de software necesarias para el sistema tienen que ver con servidores, servicios y navegadores Web, ya que la plataforma es totalmente orientada a la Web; **iv) Interfaces de comunicaciones** La plataforma de comunicación desde el Gateway SMS hacia el integrador de servicios es el Internet sobre protocolo TCP/IP.

i) Requisitos funcionales.

El sistema Gateway SMS permitirá:

- Autenticar usuarios de acuerdo a un rol para interactuar con el sistema.
- Crear, mantener y consultar la información de usuarios en el sistema SMS.
- Crear, mantener y consultar la información de los clientes en el sistema SMS.
- Crear, mantener y consultar la información de los números cortos registrados en el sistema SMS.
- Crear, mantener y consultar la información de las plantillas del sistema SMS.
- Crear, mantener y consultar la información de las operadoras del sistema SMS.

3.2 *Diseño*

a) Arquitectura

La arquitectura de software utilizada en el sistema ESPE-SMS es Modelo-Vista-Controlador [14] sobre plataforma Web, es decir el cliente será un usuario en cualquier parte del mundo que accede al sistema mediante un navegador Web con conexión a Internet.

b) Interfaz de la aplicación

Para la creación de la interfaz de la aplicación se utilizó SAVANT3 [15], el cual es un potente pero ligero sistema de plantillas orientado a objetos para Php5.

3.3 Diagrama de Clases

La Fig. No. 3 representa el Diagrama de Clases del Gateway SMS. Tal como se puede apreciar, este diagrama permite visualizar las principales clases y las relaciones entre las clases que involucran el sistema, las cuales pueden ser asociativas, de herencia, de uso y de contención. Entre las principales clases se distinguen: cliente, usuario, número corto, mensaje, envío, rol, plantilla y por su puesto los atributos y métodos por cada clase.

3.4 Implementación

Para la implementación del ambiente de pruebas no fue necesaria la utilización de recursos costosos, ya que se centró en la utilización de software y herramientas bajo estándares libres.

Usamos un equipo de escritorio con un procesador Intel(R) Core(TM)2 Duo a 2.66GHz, con 2Gb de memoria RAM y una capacidad en disco de 40 GB, para integrarnos a la red empleamos una tarjeta 3Com.

En cuanto al software para el servidor de la aplicación se optó por Centos 5.4 como sistema operativo, el servidor Web utilizado es Apache versión 2.2.3, se utilizó Php5 como lenguaje de programación y Mysql se encarga de guardar los datos, para facilidad en el desarrollo se utilizó el framework Savant3.

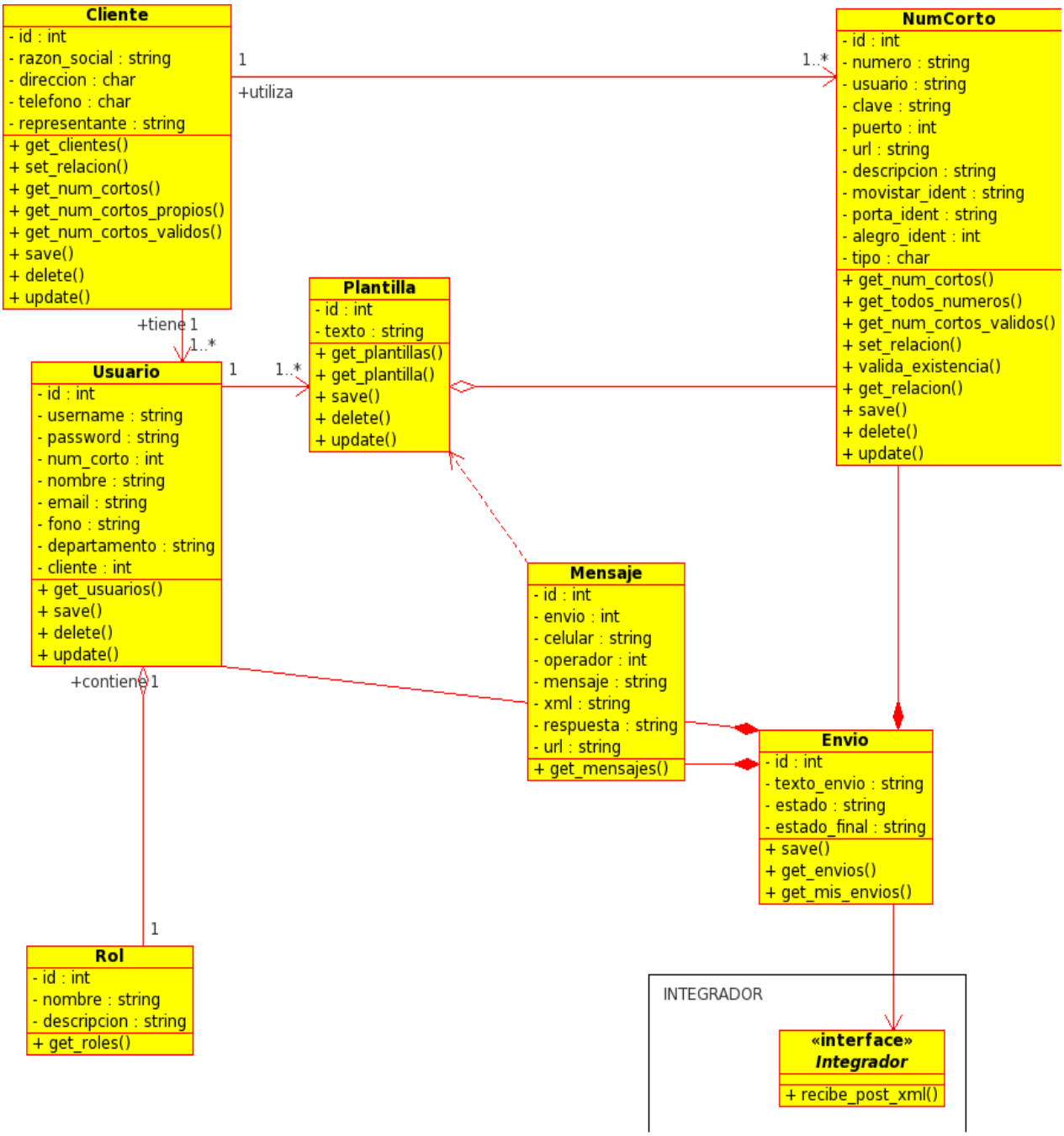

Figura 3. Diagrama de Clases – Gateway SMS

3.5 Diagrama de secuencia de un SM MT:

La Fig. 4 muestra el Diagrama de secuencias de envío de un SM-MT desde el gateway SMS hacia un teléfono celular:

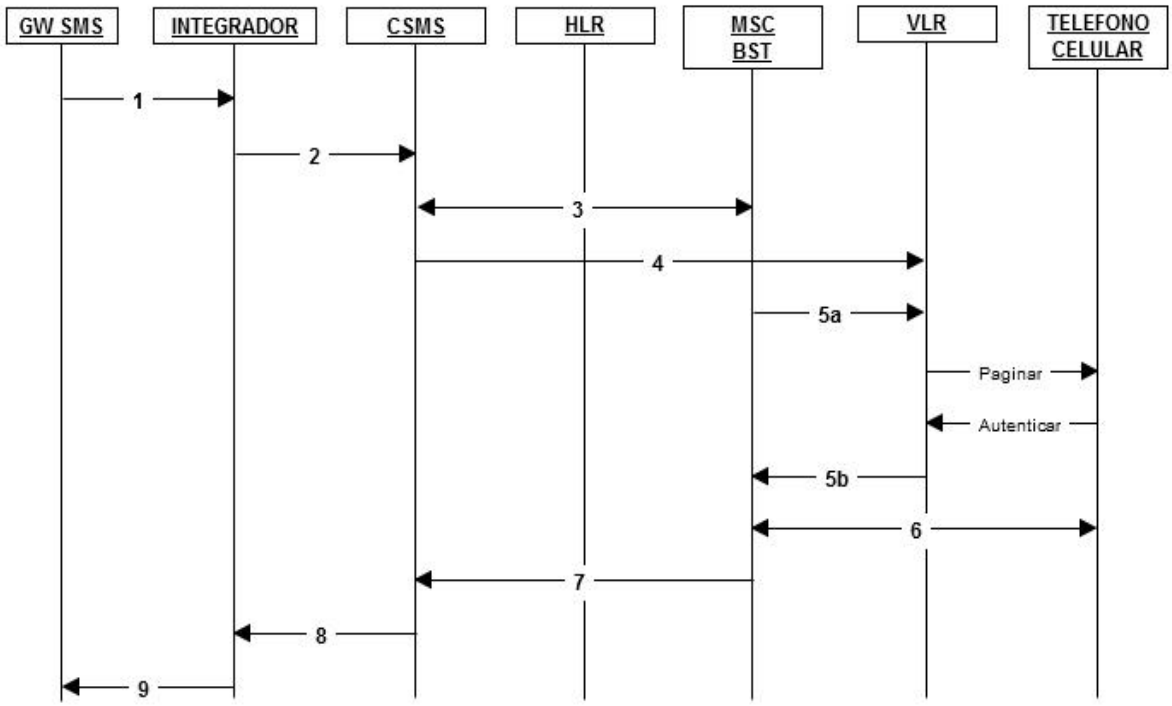

Figura 4. Escenario MT - GSM

A continuación se detallan sus elementos:

1) Invocar servicio Web
2) Enviar SMS
3) Enviar información de enrutamiento para SMS
4) Hacer seguir el mensaje corto
   a. Enviar información para MT-SM
   b. Enviar información para MT-SM (ACK)
5) Transferencia del Mensaje
6) Entrega de reporte
7) Reporte de estatus
8) Invocar servicio Web (ACK)

4. RESULTADOS EXPERIMENTALES

Los resultados obtenidos fueron calculados a partir de una muestra de 15 mensajes cortos.

4.1 Velocidad de Envío

Los aspectos principales a tomar en cuenta para determinar el tiempo de envío son: **a)** La velocidad del acceso a Internet; **b)** La capacidad de gestión del Integrador SMS; **c)** La capacidad del CSMS de cada operadora.

El tiempo total usado fue de 48 segundos, por lo cual cada mensaje demoró en promedio 3.13 segundos en llegar al teléfono celular desde su salida del Gateway SMS. En la Fig. 5 se muestra la relación entre el tiempo utilizado y el número de mensajes enviado:

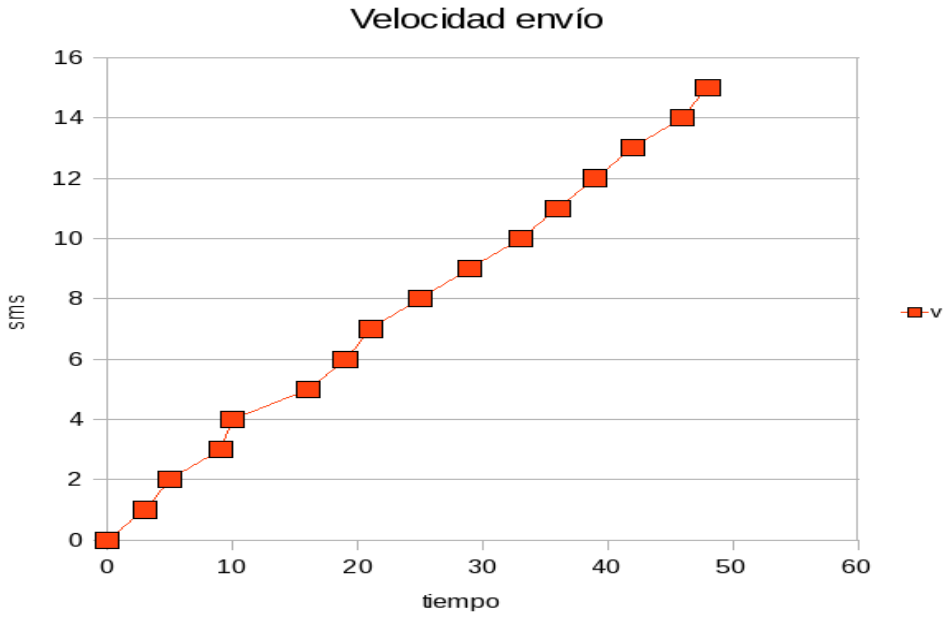

Figura 5. Promedio de Velocidad de Envío de SMS

4.2 Efectividad en los Envíos

Tomando en cuenta el número de mensajes salientes desde el Gateway SMS con respecto al número de mensajes recibidos en los celulares, encontramos que la efectividad en el envío fue del 100%. La Fig. 6 muestra el porcentaje de aciertos y errores del envío:

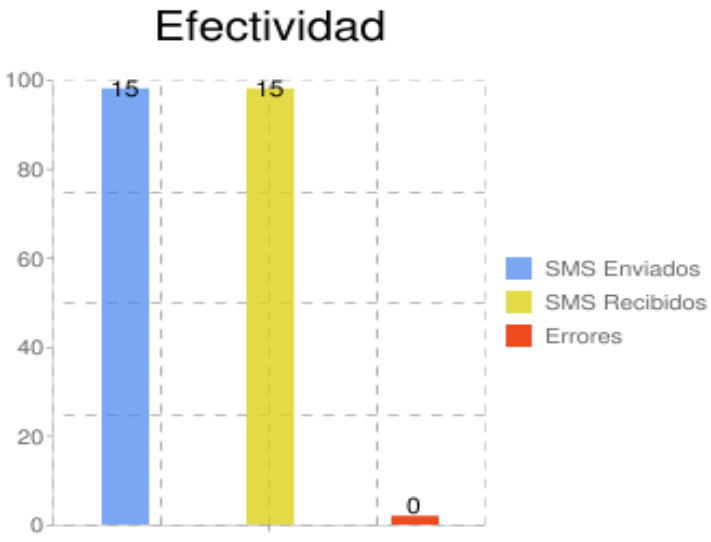

Figura 6. Efectividad en el envío de SMS

5. TRABAJOS RELACIONADOS

En relación con el mercado internacional, existe una gran variedad de productos generalmente asociados a empresas cuyo giro de negocio es la venta de SMS, como por ejemplo Clickatell [16], que es el proveedor No. 1 a nivel mundial, pudiendo ser utilizado el servicio en más de 221 países en el mundo. En este mismo ámbito la empresa Trusenses [17], localizada en Suiza, cubre 69 países en el mundo con su servicio.
Estos proveedores ofertan sus servicios basados en tecnologías Web, con conexiones a través de Internet a las operadoras de los distintos países donde prestan el servicio. Comparado a nuestro trabajo estos servicios no identifican el mensaje con un número corto y su costo es similar al ofrecido localmente.

En referencia al mercado local, existen Integradores SMS que monopolizan el tráfico de mensajes de texto utilizando números cortos, haciendo estrictamente necesaria la utilización de su servicio que es bajo en prestaciones y cerrado, lo que imposibilita la parametrización de acuerdo a los requerimientos propios. En este ámbito, la empresa Mobile Business [18] utiliza el mismo Integrador de servicios que el utilizado por el presente trabajo, de igual manera brinda sus servicios a través de tecnologías Web. Debido a la naturaleza del servicio, todos los gateway SMS deben conectarse al integrador SMS, con los mismos métodos y por los mismos canales, la diferencia radica en el software utilizado para la plataforma y el desarrollo.

6. CONCLUSIONES Y TRABAJO FUTURO

En esta investigación se ha comprobado que el envío de mensajes cortos (SMS) asegura una comunicación masiva, efectiva e instantánea. Ayudado por el uso cotidiano de la mensajería SMS por parte de los usuarios la adaptación al servicio propuesto es instantánea. Así mismo se optimiza el tiempo de quien lo utiliza. Este servicio puede actuar como un producto mercadeo y servicio al cliente, generando sentido de pertenencia hacia la institución y ayudando en la fidelización de clientes. También es una poderosa herramienta para la gestión de cobros. En general, debido a la facilidad de uso y efectividad, las aplicaciones que se le pueden dar están limitadas por las necesidades del cliente. El desarrollo Web de la aplicación permite: **i)** Usar el software como servicio; **ii)** Usar el software desde cualquier lugar donde se disponga de una conexión a Internet, **iii)** No requiere software cliente en el equipo del usuario, se usa el navegador en cualquier sistema operativo, **iv)** Las actualizaciones se hacen en el servidor, sin necesitar la intervención del cliente.

Como trabajo futuro se planea realizar el estudio al fin de determinar los requerimientos necesarios para que la ESPE se convierta en un Integrador SMS.

Referencias Bibliográficas

[1] Tomi T. Ahonen, "Communities Dominate Brand", [Online:] http://communities-dominate.blogs.com/brands/2009/02/bigger-than-tv-bigger-than-the-Internet-understand-mobile-of-4-billion-users.html?cid=6a00e0097e337c88330133f30f402f970b, USA, Febrero, 2009.

[2] Estadísticas, Superintendencia de Telecomunicaciones del Ecuador (Supertel), [Online:] www.supertel.gov.ec, Ecuador, Junio, 2010.

[3] Instituto Nacional de Estadísticas y Censos (Inec), [Online:] www.inec.gov.ec, Ecuador, Junio, 2010.

[4] Felipe Romero, Ramiro Montaquiza, "Encuesta realizada a 200 estudiantes de la Carrera de Ingeniería de Sistemas e Informática de la ESPE", Ecuador, Junio, 2009.

[5] Phil Brock, Agile Alliance, [Online:] http://www.agilealliance.org/, USA, 2010

[6] Libertades del Software Libre, Free Software Foundation (FSF), [Online:]

http://www.fsf.org/, USA, Mayo, 2008.
[7] Third Generation Partnership Project, [Online:] http://www.3gpp.org/, Francia, 2010.
[8] Finn Trosby, "SMS, the strange duckling of GSM", [Online:] http://www.telenor.com/telektronikk/volumes/pdf/3.2004/Page_187-194.pdf, Noruega, Marzo, 2004.
[9] European Telecommunications Standards Institute (ETSI) http://www.etsi.org/
[10] Randall A. Snyder, Michael D. Gallagher, "Wireless telecommunications networking with ANSI-41" , McGraw-Hill, 2001, ISBN 0-07-135231-7
[11] Signaling System 7 http://www.protocols.com/pbook/pdf/ss7.pdf
[12] Extreme Programming http://www.extremeprogramming.org, USA, 2010.
[13] Kent Beck, Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin, Steve Mellor, Ken Schwaber, Jeff Sutherland, Dave Thomas, "Principios del Manifesto Ágil", [Online:] http://www.agilemanifesto.org/iso/es/principles.html, USA, Febrero, 2001.
[14] Modelo Vista Controlador, [Online:] http://es.wikipedia.org/wiki/Modelo_Vista_Controlador, Usa, 2010.
[15] Savant3, Framework de Desarrollo, [Online:] http://www.phpsavant.com/, USA, 2010.
[16] Proveedor SMS, Clickatell, [Online:] http://www.clickatell.com, SudAfrica, 2010.
[17] Proveedor SMS, Truesenses, [Online:] http://www.truesenses.com/, Suiza, 2010
[18] Proveedor SMS, Mobile Business, [Online:] http://www.mensajea.net/, Ecuador, 2010.

Sistema de control y seguridad para casas inteligentes orientado a la Web 2.0 bajo Linux desarrollado con JEE de Java.

C. Ortiz, A. Solórzano, R. Fonseca, J. Andrango
Departamento de Ciencias de la Computación, Escuela Politécnica del Ejército, Sangolquí, Ecuador
marycris_86@hotmail.com, andres_solorzano85@hotmail.com, efonseca@espe.edu.ec, ***jaime_ac@yahoo.com***

RESUMEN: Actualmente en el Ecuador existen problemas de seguridad debido al incremento delincuencial especialmente en las viviendas. Esto se produce dado que muchas personas por su trabajo o porque viajan constantemente no pueden permanecer en sus hogares todo el tiempo necesario, y es de suma importancia que las familias posean un sistema de seguridad instalado en sus casas. Ante este problema, el presente artículo propone el desarrollo de un sistema domótico el cual se enfoque no solo en lo referente al hardware, sino también al software, brindando al usuario la posibilidad de tener un sistema distribuido e interactivo acoplado a sus necesidades, haciendo uso de tecnologías actuales como Java EE y Richfaces, incorporando los beneficios que brinda Jboss® para obtener una aplicación Web 2.0 enfocada principalmente en la seguridad del hogar. Los resultados experimentales muestran que, el sistema permite controlar dispositivos desde una aplicación orientada a la Web, permitiendo a los usuarios administrar las luces eléctricas, cerraduras y alarmas en tiempo real, de la misma manera se pueden programar tareas para lograr un control automático del hogar.

Palabras clave: Sistema domótico, sistema distribuido, sistema interactivo, Web 2.0

ABSTRACT: Nowadays in Ecuador there are many security issues due to increased crime especially in homes. This takes place provided that many people for his job or because they travel constantly cannot remain in his homes all the necessary time, and it is important that families possess a security system installed in his houses. In front of this problem, this paper proposes the development of a home automatic system which focuses not only in what concerns the hardware, but also the software, offering the user the possibility of having a distributed and interactive system connected to his needs, using current technologies such as Java EE and RichFaces, incorporating the benefits that JBoss® offers to obtain a Web 2.0 application focused principally on the homeland safety. The experimental results show that, the system allow to control devices from a Web application, allowing users to manage the lights, locks (doors and windows) and alarms at real time, in the same way permit schedule tasks to achieve automatic control of home.

Keywords: Domotic system, distributed system, interactive system, Web 2.0

1 INTRODUCCIÓN

Actualmente en el mundo se han producido cambios significativos no solo en el ámbito económico sino también en el área social. Los precios de todos los productos han aumentado y se vive en una crisis mundial [1.].

El aumento de la delincuencia ha supuesto un deterioro de la seguridad pública [2.]. No hay ciudadano en el país que no tenga este momento un pariente o al menos un amigo cercano, que no haya sido víctima de los delincuentes. Y no hay peor sentimiento de inseguridad y desconcierto que aquel derivado de la violación de los espacios personales como: robo de casa, secuestros, asaltos. Todo eso, en conjunto, genera un ambiente de incertidumbre y temor social, que a la postre genera violencia.

La integración de las tecnologías de la información en el hogar, se denomina "domótica", y es la incorporación de tecnología a la vivienda que permite su control a distancia, brindando de esta manera seguridad, bienestar y ahorro de energía, racionalizando los distintos consumos [3.].

En los últimos años, se ha observado un creciente interés hacia la seguridad del hogar, es por ello que existen empresas dedicadas a brindar servicios de seguridad. Entre ellos se puede citar: *i) DIGITAL HOME* es una empresa que realiza proyectos a la medida de las necesidades y presupuesto del cliente, se le asesora en la construcción de la vivienda y si ya se encuentra construida se implementa el sistema [4.]. *ii) ISDE ECUADOR* posee una línea de productos llamada SICOV, Sistema de Control de Vivienda Centralizado de Bajo Costo, fácil de instalar. Sistema Adecuado para promoción de viviendas [5.]. *iii) GENERAL DOMOTIC* es una empresa que se dedica a la automatización de viviendas, edificios, colegios, hospitales, hoteles e industria así como a todo tipo de obra civil. Ofrece servicios de consultoría, diseño y ejecución de proyectos de obra civil y telecomunicaciones [6.]. *iv) SOLAIE* es una empresa radicada en Ambato, dedicada al desarrollo, producción y comercialización de Sistemas de Control para la automatización de la industria así como de las viviendas [7.].

De la misma manera existen estudios realizados por compañeros de otras universidades del país, entre los que podemos destacar: *i)* "Estudio de los sistemas Web embebidos y su aplicación en un sistema de control domótico con microcontroladores", que permite monitorizar una vivienda a través de Internet a partir de microcontroladores. *ii)*"Diseño de un sistema de control, automatización y monitoreo para viviendas o negocios utilizando multimedia sobre IP". *iii)* "Diseño e implementación del control de acceso y seguridad del laboratorio de instrumentación utilizando el protocolo X-10", que usa los diferentes módulos X-10, y Visual Basic como interfaz de usuario.

Frente a este escenario el presente proyecto, se basa en el concepto de domótica y propone ayudar a solucionar problemas de seguridad de los hogares, controlando y administrando actividades como: el acceso al hogar, ahorro energético, interactividad usuario-hogar y alarmas.

El resto del artículo ha sido organizado como sigue: La sección 2 describe la metodología y las herramientas de software y hardware utilizadas. En la sección 3 se redacta el diseño y la implementación. La sección 4 se refiere a los resultados y discusión. La sección 5 contiene trabajos relacionados y finalmente, en la sección 6, se presentan las conclusiones y el trabajo futuro.

2 METODOLOGÍA Y HERRAMIENTAS DE DESARROLLO

2.1 Rational Unified Process RUP

El Proceso Unificado de Rational (*Rational Unified Process, RUP*) es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos. El RUP no es un sistema con pasos firmemente establecidos, sino un conjunto de metodologías adaptables al contexto y necesidades de cada organización [8.].

Mediante esta metodología se logró definir los requerimientos y procesos a seguir para el desarrollo del sistema de casas inteligentes (véase Fig. 1).

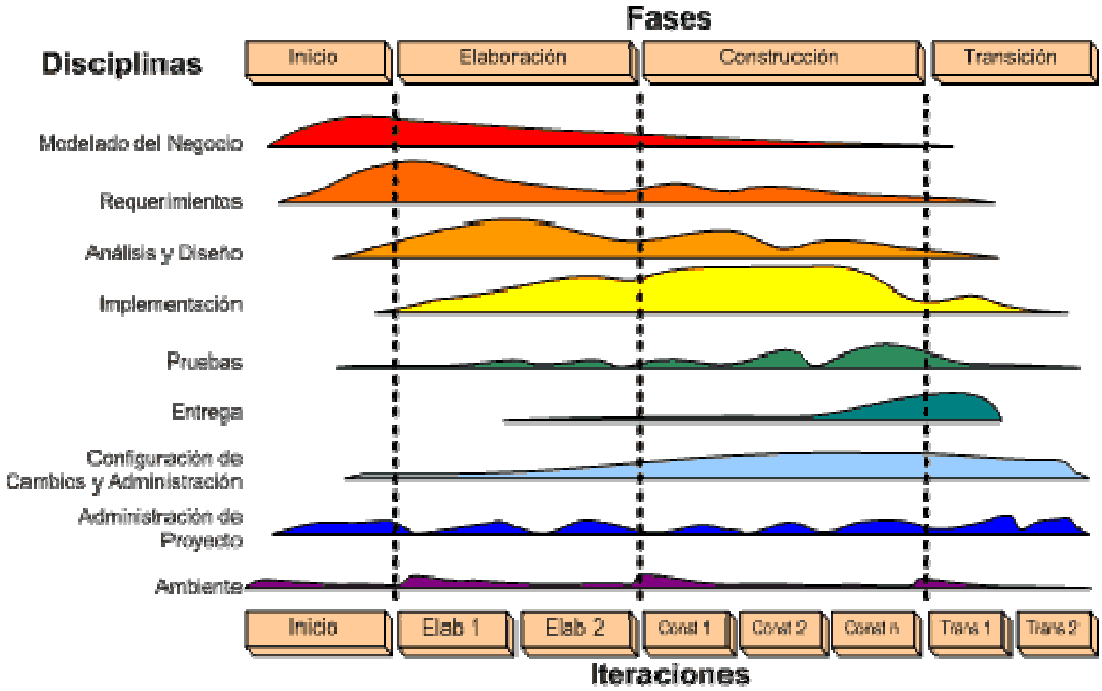

Figura 1. Fases del RUP Fuente: **[9.]**.

Como se puede observar en la Fig. 1, RUP divide el proceso en cuatro fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace un mayor o menor hincapié en las distintas actividades.

Las fases de inicio y elaboración se enfocan hacia la comprensión del problema y la tecnología, la delimitación del ámbito del proyecto, la eliminación de los riesgos críticos, y al establecimiento de la línea base de la arquitectura. En la fase de construcción, se lleva a cabo la construcción del producto por medio de una serie de iteraciones. En la fase de transición se pretende garantizar que se tiene un producto preparado para su entrega a la comunidad de usuarios.

2.2 Herramientas para el desarrollo de software.

2.2.1 Java Server Faces

Uno de los patrones más conocidos en el desarrollo Web es el patrón MVC (Modelo Vista Controlador). Este patrón permite separar la lógica de control, la lógica de negocio y la lógica de presentación. Utilizando este tipo de patrones se consigue: más calidad, mejor mantenibilidad, teniendo un patrón a seguir al empezar un proyecto, pero una de las cosas más importantes es la normalización y estandarización del desarrollo de Software.

JavaServer Faces (JSF) es una tecnología y framework de desarrollo basado en el patrón MVC (Modelo Vista Controlador) para aplicaciones Java basadas en Web que simplifica el desarrollo de interfaces de usuario en aplicaciones Java EE. JSF usa JavaServer Pages (JSP) como la tecnología que permite hacer el despliegue de las páginas.

2.2.2 RichFaces

RichFaces es una biblioteca de componentes para JSF y un avanzado framework para la integración de AJAX con facilidad en la capacidad de desarrollo de aplicaciones de negocio.

El framework es implementado como una biblioteca de componentes el cual incorpora Ajax en las páginas existentes, de esta manera los desarrolladores no necesitan escribir código JavaScript o reemplazar los componentes existen con nuevos Ajax widgets. Además, RichFaces habilita un amplio soporte Ajax para las páginas en lugar del tradicional soporte de componentes. Finalmente, RichFaces permite a los desarrolladores definir (con JSF tags) las diferentes partes de la páginas JSF que se desea actualizar mediante una petición Ajax, y provee una serie de opciones para enviar la petición Ajax al servidor. De la misma manera la página JSF no deja de ser una página regular JSF por lo que no se debe introducir código JSF a mano.

2.2.3 Enterprise JavaBeans

Los EJB son componentes del contexto de servidor que cubren la necesidad de intermediar entre la capa Web y diversos sistemas empresariales. En este sentido conviene releer la arquitectura J2EE para ver el papel de interfaz que juegan estos componentes. Los EJB nacen para encapsular la lógica de negocio de una forma integrada, no quedando dispersos su representación en una pléyade de sistemas empresariales. Los EJB están especialmente pensados para integrar la lógica de la empresa que se encuentra en sistemas distribuidos, de tal forma que el desarrollador no tenga que preocuparse por la programación a nivel de sistema (como control de transacciones, seguridad, etc.), sino que se centre en la representación de entidades y reglas de negocio.

Existen tres tipos de EJBs: ***i)*** *EJB de Entidad* (*Entity EJBs*): su objetivo es encapsular los objetos del lado del servidor que almacena los datos. Los EJB de entidad presentan la característica fundamental de la persistencia. ***ii)*** EJB de Sesión (*Session EJBs*): gestionan el flujo de la información en el servidor. Generalmente sirven a los clientes como una fachada de los servicios proporcionados por otros componentes disponibles en el servidor. ***iii)*** EJB dirigidos por mensajes (*Message-driven EJBs*): son los únicos beans con funcionamiento asíncrono. Usando el *Java Messaging System* (JMS), se suscriben a un tema (*topic*) o a una cola (*queue*) y se activan al recibir un mensaje dirigido a dicho tema o cola. No requieren de su instanciación por parte del cliente [10.].

2.2.4 Seam Framework

JBoss Seam es un nuevo framework para desarrollar aplicaciones Web 2.0 de próxima generación, unificando e integrando tecnologías como JavaScript asincrónico y XML (AJAX), Java ServerFaces (JSF) y Enterprise Java Beans(EJB3), Java Portlets y Manejo de procesos del negocio (MPN).

Seam introduce el concepto de contextos, cada componente de Seam existe dentro de un contexto. El contexto conversacional por ejemplo, captura todas las acciones del usuario hasta que éste sale del sistema o cierra el navegador, inclusive puede llevar un control de múltiples pestañas y mantiene un comportamiento consistente cuando se usa el botón de regresar del navegador.

2.3 Herramientas de hardware

2.4 Microcontrolador PIC

Un microcontrolador es un circuito integrado o chip que incluye en su interior las tres unidades funcionales de una computadora: unidad central de procesamiento, memoria y unidades de E/S (entrada/salida). Debido a su reducido tamaño es posible montar el controlador en el propio dispositivo al que gobierna. En este caso el controlador recibe el nombre de controlador empotrado (véase Fig. 2).

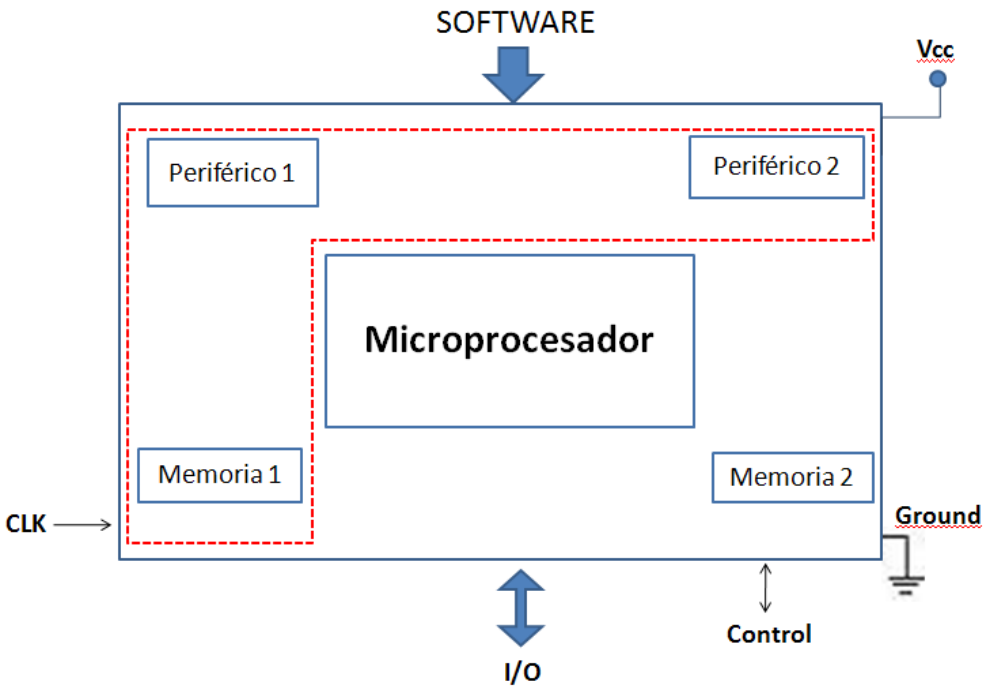

Figura 2. Esquema de un Micro controlador, Fuente: **[11.]**

En la Fig. 2, se observar al microcontrolador dentro de un encapsulado de circuito integrado, con su procesador (CPU), buses, memoria, periféricos y puertos de entrada salida. Fuera del encapsulado se ubican otros circuitos para completar periféricos internos y dispositivos que pueden conectarse a los pines de entrada/salida.

Los PIC (PICMicro o ***P**eripheral **I**nterface **C**ontroller*) son una familia de microcontroladores tipo RISC fabricados por Microchip Technology Inc [12.].

La arquitectura de los microcontroladores esta basada en la arquitectura Harvard la cual dispone de dos memorias independientes una, que contiene sólo instrucciones y otra, sólo datos. Ambas disponen de sus respectivos sistemas de buses de acceso y es posible realizar operaciones de acceso (lectura o escritura) simultáneamente en ambas memorias.

2.5 Programación del PIC

Para transferir el código de un ordenador al PIC normalmente se usa un dispositivo llamado programador. Este proceso corresponde a utilizar un programa en el PC que toma el código ensamblado (.hex, .o, .bin, .coff) para el microcontrolador específico, y lo envía mediante algún puerto (para el caso específico USB) a un dispositivo que lo escribe en la memoria del microcontrolador.

Para realizar la fase de software en el microcontrolador, se ha programado en el compilador PICC que maneja el protocolo USB y permite al PC maestro detectar al dispositivo esclavo periférico como un dispositivo serial RS-232 en forma virtual.

3 DISEÑO E IMPLEMENTACIÓN

El sistema de Casas Inteligentes está dividido en dos partes: la parte de software y la parte Electrónica, esta ultima incluye la implementación de circuitos eléctricos que permitirá tener un mejor control y

administración de las Casas Inteligentes, como el encendido y apagado de luces internas o externas de la casa, o como poder asegurar o quitar el seguro a puertas y ventanas de la casa.

3.1 Proceso de control de dispositivos en la casa inteligente

El proceso se inicializa cuando el usuario se registra en el sistema de casa inteligente, se listará una lista de opciones que permitirá administrar tanto a los usuarios y sus permisos como los dispositivos y sus acciones (véase Fig. 3).

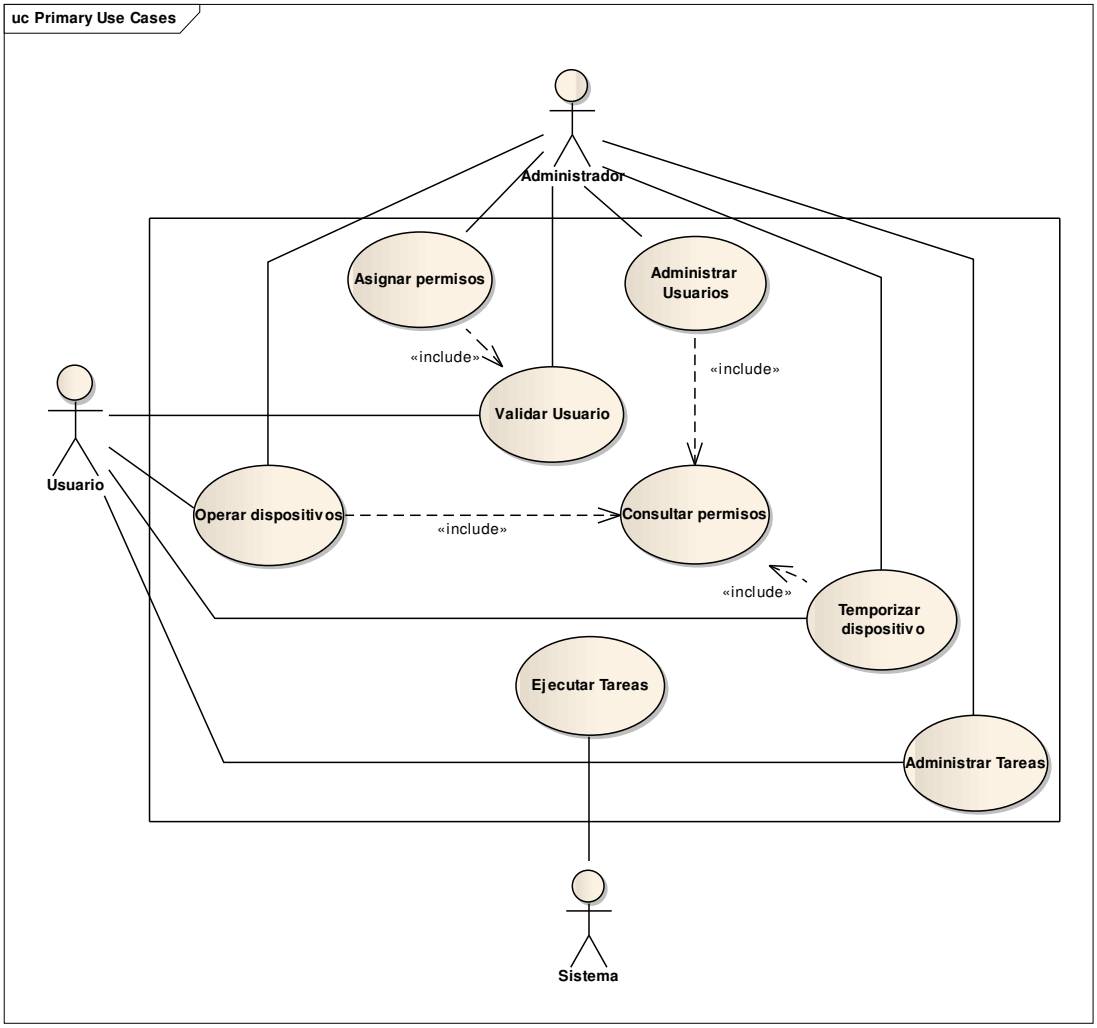

Figura 3. Diagrama general de casos de uso del sistema

En la Fig. 3 se puede observar el diagrama general de casos de uso del sistema en el que se detalla cada uno de los procesos que realiza el sistema de casa inteligente.

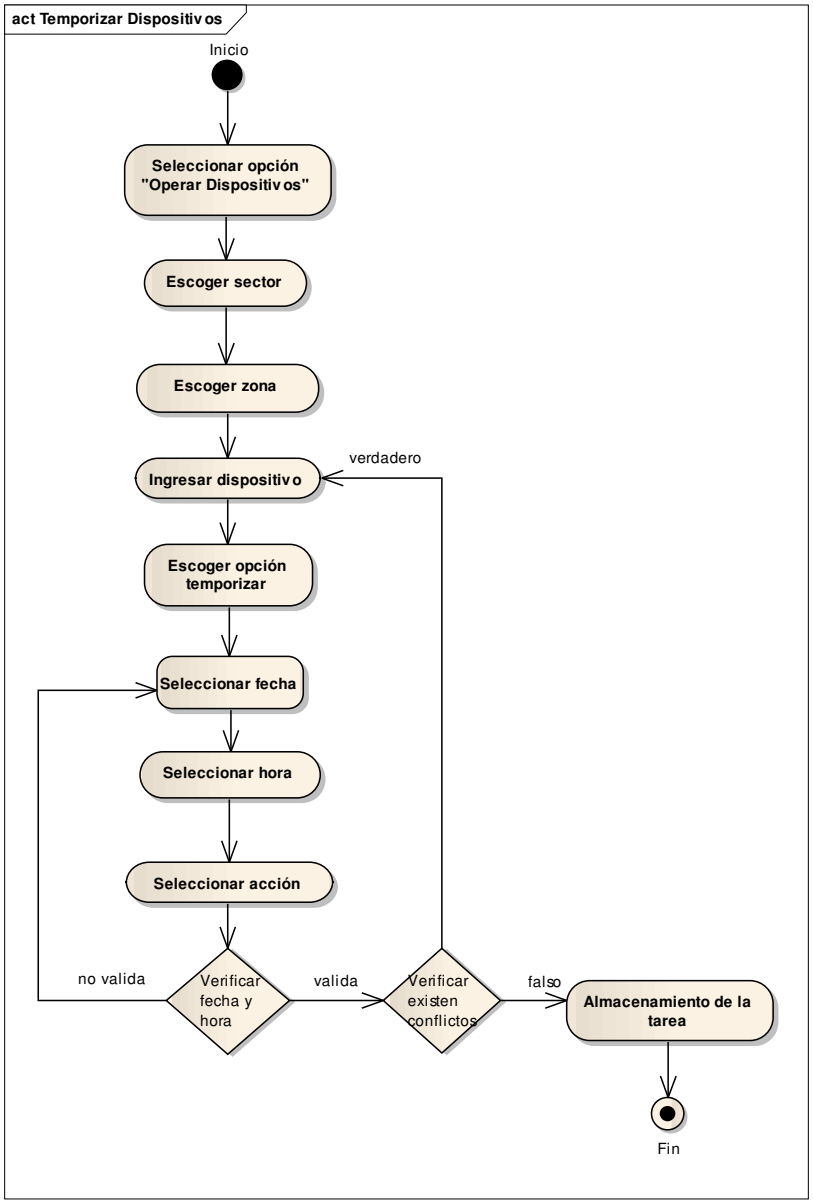

Figura 4. Diagrama de Actividad Temporizar Dispositivos

El proceso principal del sistema se basa en el control de las actividades realizadas por los dispositivos en el hogar, para lo cual se dispone de pantallas mediante las cuales el usuario podrá visualizar en tiempo real los cambios producidos en cada dispositivo e interactuar con cada uno de ellos, el proceso de cómo se temporiza un dispositivo se lo puede observar en la Fig. 4 del presente documento.

3.2 Base de Datos PostgreSQL

Para que el sistema pueda reconocer que tipos de dispositivos puede manipular, se ha diseñado una Base de Datos en PostgreSQL que almacena todos los dispositivos de la casa inteligente con sus estados actuales y su respectivo sector o ubicación dentro de la casa.

En el momento en el que el usuario del sistema realice una operación en la casa inteligente, se almacenará en la base de datos el nuevo estado del dispositivo, de tal manera que la siguiente vez que el

usuario desee realizar una operación pueda visualizar el estado actual de cada dispositivo, esto para brindarle información de los estados de los dispositivos en tiempo real.

Como se puede observar en la Fig. 5 se dispone de tablas específicas en las cuales se almacenará los datos y estados del dispositivo, y en otra las tareas programadas para los dispositivos, la parte primordial a tomarse en cuenta es la tabla de permisos, ya que los usuarios tendrán acceso limitado a los sectores de la casa.

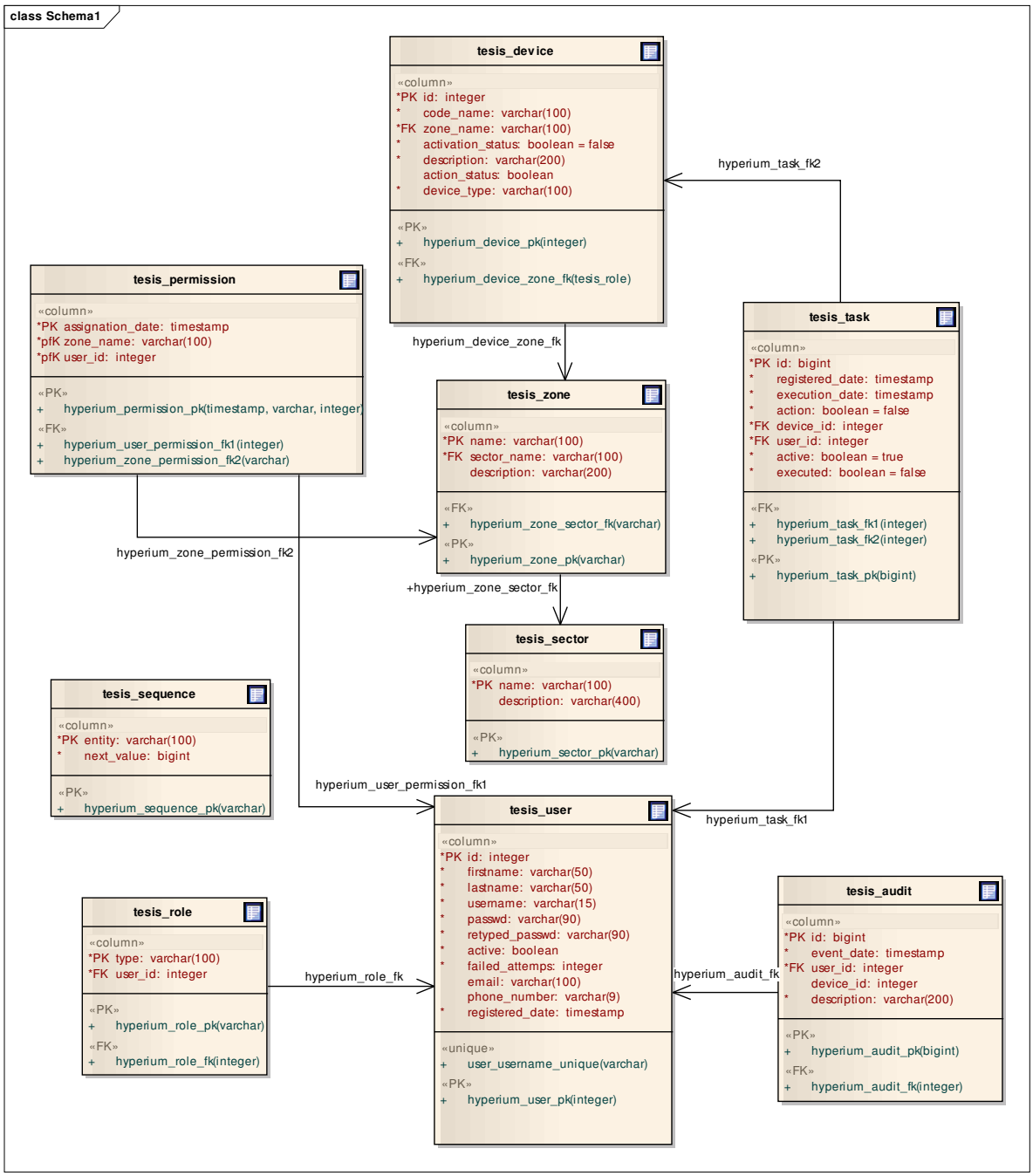

Figura 5: Diagrama Entidad-Relación de la solución.

4 RESULTADOS Y DISCUSIÓN

Como resultado del desarrollo del sistema, se implementó una solución distribuida basada en la Web que permite al usuario mantener un control de los dispositivos remotamente en tiempo real, de la misma manera el usuario tiene la facilidad de programar tareas las cuales desea que sean ejecutadas a una hora específica del día, aun estando fuera del hogar.

En la Fig. 6 se observa la pantalla de administración en la que se listan todos los dispositivos existentes en cada zona, el sensor de movimiento en el momento se encuentra apagado, lo que significa que no ha detectado actividad en la zona.

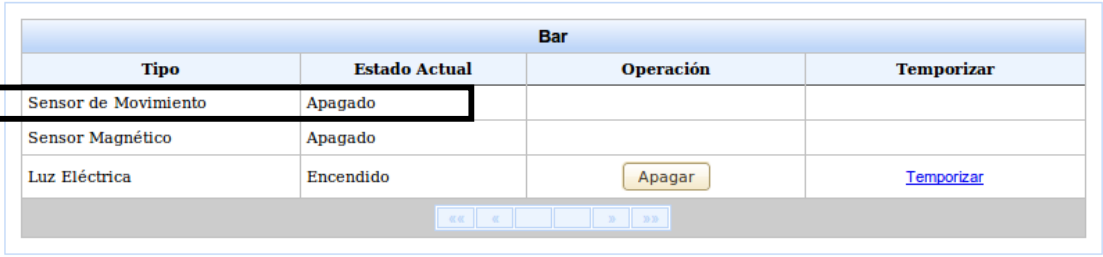

Figura 6: Pantalla de administración de dispositivos, sensor movimiento apagado.

Al detectarse actividad en la zona, un mensaje de información es presentado en la consola del servidor, como se puede observar en la Fig. 7.

```
20:21:00,453 INFO  [STDOUT] Se activó el dispositivo: Sensor de Movimiento, de la zona: Bar
20:21:00,456 INFO  [STDOUT] SECUENCIAL DE AUDITORIA: 54
20:21:00,458 INFO  [STDOUT] SECUENCIAL DE AUDITORIA: 54
```

Figura 7: Consola del servidor, sensor de movimiento activado

De la misma manera en la pantalla de administración el estado del dispositivo es actualizado como se observa en la Fig. 8.

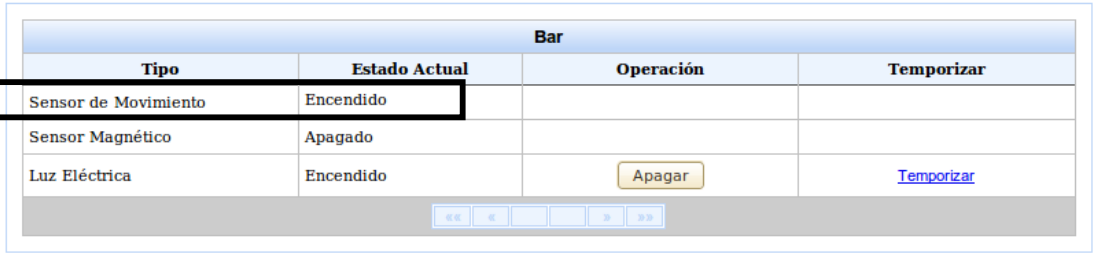

Figura 8: Pantalla de administración de dispositivos, sensor de movimiento encendido

Sin embargo según pruebas observamos un retraso de 500 milisegundos de retraso en la actualización del estado del dispositivo en la pantalla de administración.

De la misma manera aunque el sistema maneja la opción de temporizar dispositivos, esta no se aplica para los sensores, dado que estos se encuentran en constante estado inactivo, hasta que un movimiento lo active.

A pesar de que la mayoría de sistemas domóticos poseen cámaras Web como forma adicional de controlar la seguridad del hogar, por costos y por falta de drivers en Linux no se logró implementar.

El sistema desarrollado así como los trabajos publicados tienen componentes de hardware que actúan como cerebro de la casa, los cuales permiten administrar y controlar los distintos dispositivos existentes en el hogar.

5 TRABAJOS RELACIONADOS

Existen en el mercado una variedad de empresas dedicadas a realizar proyectos relacionados al sistema desarrollado, de la misma manera existen proyectos de investigación los cuales servirán como punto de referencia, en esta sección se han incluido los más relevantes que se han encontrado durante la investigación:

En lo referente al trabajo presentado en [13.], describe el uso de sistemas embebidos como interfaz directa con el hardware e intermediario entre el software de alto nivel y las funciones del hardware, su lenguaje de programación es de medio y bajo nivel, de la misma manera detalla como al estar estos sistemas dedicados a una tarea específica, se puede optimizar los costos de producción, el tamaño del producto y el consumo de potencia. Asimismo describe la aplicación de Basic Stamp 2 microcontrolador que ejecuta programas en lenguaje PBASIC, el cual es capaz de almacenar entre 500 y 600 instrucciones de alto nivel y ejecuta un promedio de 4000 instrucciones/segundo. La autora muestra como el sistema diseñado es una alarma de arquitectura centralizada, ya que dispone de un controlador centralizado que envía la información a la interfaz del programa. Sin embargo aunque el resultado final es eficiente, la interfaz no es amigable, haciéndolo difícil de manejar y entender para los usuarios.

En lo que se refiere al estudio [14.], se habla del concepto de una pasarela residencial el cual es un dispositivo que conecta las infraestructuras de telecomunicaciones de la vivienda a una red pública de datos, como por ejemplo el Internet, ya que la red domótica es totalmente independiente de la red de datos y multimedia, pero gracias a la pasarela residencial es posible interactuar entre ellos, la investigación detallada no demuestra resultados claros, pero nos da una idea de la aceptación de un proyecto de esta magnitud en el mercado actual.

Respecto al trabajo propuesto en [15.], se usa como plataforma de desarrollo el protocolo estándar de comunicación X-10, el cual permite enviar información a través de la red eléctrica de bajo voltaje presente en las viviendas. El software desarrollado en Visual Basic permite el manejo de sensores, cámaras de video, módulos actuadores y un módulo de interface de computador. De la misma manera el sistema actúa en tiempo real, mostrando continuamente la información recibida por los sensores de vigilancia de acuerdo a los datos recibidos. Dados los inconvenientes que tiene el país con respecto a la generación de energía eléctrica, sus altas y bajas de corriente, estos dispositivos tienden a quemarse por lo que las viviendas deberán disponer de reguladores de corriente para que dichos módulos trabajen de forma adecuada.

6 CONCLUSIONES Y TRABAJO FUTURO

Con el desarrollo del software se comprobó que la aplicación de una metodología facilita el desarrollo de las aplicaciones y genera mejores resultados. El uso del estándar IEEE 830 y RUP, ayudó a cumplir los objetivos planteados y optimizar los recursos y costos. De la misma manera gracias a las facilidades de acoplamiento entre el IDE de Jboss® y Seam se pudo obtener el mayor provecho de dicha herramienta y de esta manera crear un sistema interactivo, estable y acoplado a sus necesidades del usuario. El programa PICC permitió desarrollar la lógica de acción de cada uno de los dispositivos y la manera en la que van a actuar dependiendo de la situación en la que se encuentre.

Como trabajo futuro se pretende que el sistema controle cámaras Web y que pueda ser accedido mediante dispositivos móviles. De la misma manera se aumentará el número de dispositivos.

Referencias Bibliográficas

[1.] El Universo, “Mercados presentan aumento de precios en productos de la sierra”, Enero 2010, [Online: http://www.eluniverso.com/2010/01/19/1/1356/mercados-presentan-aumento-precios-productos-sierra.html]

[2.] C. Estarellas Velásquez, “La delincuencia en el Ecuador”, Junio 2010, [Online:http://www.desdemitrinchera.com/2010/06/07/la-delincuencia-en-el-ecuador/]

[3.] Balcones Paraiso del Valle, [Online: http://www.casalindaecuador.com/balcones-del-Paraiso-Quito.html]
[4.] Digital Home, Quito-Ecuador, [Online: http://www.digitalhome.com.ec]
[5.] ISDE-ECUADOR, Edificios Inteligentes, Quito-Ecuador, 2005 [Online: http://isde-ecuador.com/].
[6.] GeneralDomotic S.A, Quito-Ecuador, [Online: http://www.generaldomotic.com]
[7.] SOLAIE, Ambato-Ecuador, [Online: http://solaie.com/]
[8.] IBM- Rational Unified Process, [Online: www.ibm.com/software/awdtools/rup/]
[9.] ConsolidaIT, Soluciones en Tecnología de Información, [Online: http://consolida-it.com/marco.htm]
[10.] Wikipedia, Oracle-Sun Microsystems, "Enterprise Java Beans", 2006, [Online: http://es.wikipedia.org/wiki/Enterprise_JavaBeans]
[11.] Wikipedia, "Microcontrolador", [Online: http://es.wikipedia.org/wiki/Microcontrolador]
[12.] Wikipedia, "Microcontrolador PIC", [Online: http://es.wikipedia.org/wiki/Microcontrolador_PIC]
[13.] I. Robalino "Estudio de los Sistemas Web embebidos y su aplicación en un sistema de control domótico con microcontroladores". Escuela Politécnica del Chimborazo. Riobamba-ECUADOR, 2009.
[14.] M. Molina, D. Montesdeoca, E. Leyton "Diseño de un sistema de control, automatización y monitoreo remoto para viviendas o negocios utilizando multimedia sobre IP". Ingeniería Electrónica y Telecomunicaciones. ECUADOR, 2005.
[15.] W. Chamorro, D. Guerrón "Diseño e Implantación del control de acceso y seguridad del laboratorio de instrumentación utilizando el protocolo X-10". Escuela Politécnica Nacional. Quito-ECUADOR, Marzo 2008.

Evaluación de ataques de Denegación de servicio DoS y DDoS, y mecanismos de protección

D. Narváez, C. Romero, M. Núñez
Departamento de Ciencias de la Computación, Escuela Politécnica del Ejército
manager571@hotmail.com, caromero@espe.edu.ec, mvnunezn@gmail.com

RESUMEN: Una de las grandes amenazas en seguridades informáticas son los potenciales ataques de denegación de servicio Dos y DDos. Ante este problema, el presente artículo propone la evaluación de las técnicas para la ejecución de ataques, basado en herramientas, además del desarrollo de una herramienta personalizada usando tecnología de múltiples hilos o multi-threads, aplicado en un escenario de pruebas reales. En consecuencia, se han implementado mecanismos de monitoreo ante este tipo de ataques, los cuales nos permitirán examinar el tráfico de red con el propósito de decidir que existe un ataque de denegación de servicio por parte de una o varias direcciones realizadas. Se han implementado las seguridades en base de software, específicamente mediante la implementación de políticas de filtrado de tipo IPTables, módulos extras en los servicios levantados que resultan más eficaces a la hora de proteger nuestra red de un posible ataque. Los resultados experimentales muestran que el nivel de eficacia de las protecciones aplicadas frente a este tipo de ataques; si bien es cierto, no serán 100% eficaces en todos los tipos de ataques actuales y probablemente menos los futuros; pero hoy por hoy, resultan ser herramientas de primera mano, conjuntamente con estrategias de seguridad adecuadas.

Palabras clave: Seguridades en Redes, ataques de denegación de servicio, mecanismos de monitoreo

SUMMARY: One of the great threats in computer science securities is the potential attacks of two refusal on watch and DDos. To carry owl this problem, the present article proposes the evaluation of the techniques for the execution of attacks, based on tools, besides the development of a customized tool using thread technology multiple or multi-threads, applied in a scene of real tests, consequently, monitoring mechanisms have been implemented before this type of attacks, which will allow us to examine the network traffic in order to decide that an attack of refusal on watch on the part of one exists or several realised directions. The securities on the basis of software have been implemented, specifically by means of the implementation of filtrate policies of IPTables type, extra modules in the raised services that are more effective at the time of protecting our network of a possible attack. The experimental results show that the level of effectiveness of the protections applied against this type of attacks; although it is certain, they will not be effective 100% in all the types of present attacks and probably except the futures; but at the present time, they turn out to be tools of first hand, jointly with suitable strategies of security.

Keywords: Network Security, Denial of service attacks,

1. INTRODUCCIÓN

El amplio desarrollo de manera exponencial que han venido teniendo las redes de comunicaciones y los sistemas de información actuales plantean inevitablemente la cuestión de su seguridad, que se ha convertido en un tema de preocupación creciente para la sociedad.

Un ataque de denegación de servicio se define como "una acción que priva o interrumpe parcial o totalmente tanto al sistema o a los usuarios, de los recursos requeridos para efectuar su normal funcionamiento" [1]. Por lo general, los ataques DoS son mecanismos que aprovechan la fuerza bruta con el propósito de "echar abajo" el sistema o convertirlo en indisponible o inutilizable, mediante una sobrecarga de la capacidad de procesamiento de paquetes y datos en sus servidores o de la pila de peticiones de la red. El primer tipo de ataque se denomina por vulnerabilidad y el segundo por inundación. Los ataques efectuados a servidores en muchas ocasiones pueden solucionarse aplicando las configuraciones y parches adecuados para limitar o incluso bloquear la excesiva carga del sistema en condiciones poco favorables [2]. Por ejemplo, los ataques mediante el envío de paquetes enmascarados o por difusión, son prácticamente imposibles de detener, a no ser que se desconecte el sistema de Internet. Puede ser que no "echen abajo" el sistema, pero seguramente se logrará saturar la conexión a Internet.

Frente a estos escenarios de pruebas se realiza los ataques a dispositivos como servidores Mandriva implementado servicios de Web (HTTP) Apache, correo (SMTP / POP / IMAP), DNS, FTP. El aporte que presenta este trabajo es darnos una mejor visión acerca de los niveles de seguridad que se deberían tener en cuenta al momento de implementar de los sistemas de información, ya que esta investigación abarca una buena cantidad de aspectos y conceptos de seguridad de redes relativamente nuevos, que además presentan una evolución constante en cuanto a sus contenidos y estrategias.

Sin embargo, es necesario recalcar, que actualmente no existe ningún sistema completamente inmune a todos los ataques de denegación de servicio, especialmente aquellos que se realizan de manera distribuida entre cientos, miles o quizás millones de equipos atacantes también víctimas de manera indirecta.

El resto del documento ha sido organizado de la siguiente manera: La sección 2 muestra los fundamentos de seguridades en las redes de comunicación. La sección 3 detalla el diseño e implementación de los ataques en un escenario real. La sección 4. La implementación y aplicación de una herramienta propia en Java e implementación de las protecciones frente a ataques DoS y DDoS. La sección 5 se evalúa las protecciones implementadas y los resultados experimentales. En la sección 6 se analiza trabajos relacionados y Finalmente en la sección 7, se presenta las conclusiones sobre los resultados obtenidos.

2. LOS ATAQUES DE DENEGACION DE SERVICIO Y LA SEGURIDAD

2.1. Políticas y modelos de seguridad

A las políticas de seguridad se las puede definir como explicitas cuando constituyen unas reglas bien documentadas, registradas y disponibles para su consulta por parte de un potencial ejecutor de la política[3]. Además, existirán las denominadas políticas implícitas, aquellas que establecen criterios que no están documentados pero que se asumen bien por su obviedad, o bien por costumbre. Para garantizar la implementación de los diferentes servicios de seguridad existen tres campos de trabajo que deben ser considerados: prevención, detección y respuesta.

2.2. Servicios básicos de la seguridad

Protege las comunicaciones ante determinadas violaciones de la política de seguridad ante los usuarios. Entre ellos se puede citar: **i)** "Servicio de autenticación o autentificación", garantiza que una entidad comunicante sea realmente quien dice ser; **ii)"**Servicio de confidencialidad de los datos", previene la divulgación no autorizada de los datos del sistema; **iii)**"Servicio de integridad de los datos", detecta

cualquier modificación, inserción, borrado o repetición de los datos, se puede tener integridad de conexión con recuperación, integridad de conexión sin recuperación, integridad de recuperación en campos selectos, integridad en modo no-conexión e integridad en modo no-conexión en campos selectos; **iv)**"Servicio de no repudio", no permite a un emisor negar haber enviado un mensaje, ni permite a un receptor el negar haber recibido un mensaje; **v)**"Servicio de control de acceso", es una protección contra el uso no autorizado del sistema; **vi)**"Servicio de privacidad", consigue que la identidad del elemento que realiza una determinada operación permanezca oculta ante algunos de los sistemas, actores o servicios presentes en dicha operación; **vii)**"Servicio de disponibilidad", se relaciona a la capacidad de la red, sistema o servicio para estar disponible en cualquier momento y para recuperarse con premura a partir de la ocurrencia de un evento de interrupción del mismo.

2.3. Clasificación y tipos de ataque

Se entiende como ataque a la seguridad o amenaza a una condición del entorno del sistema de información que, dada una oportunidad, podría dar lugar a que se produjese una violación de la seguridad. Esta señal supone la existencia de un flujo o comunicación desde un origen o emisor de la información a un destino o receptor, utilizando un canal intermedio o una red de comunicación. En este contexto, es posible concebir cuatro categorías generales de ataques a la seguridad: **i)**" Ataques por Interrupción", Un elemento del sistema es destruido o se vuelve no disponible. Este es un ataque contra la disponibilidad.; **ii)**" Ataques por Intercepción", Cuando una entidad no autorizada consigue acceso a un recurso. Este es un ataque contra la confidencialidad; **iii)**"Ataques por Modificación", Cuando una entidad no autorizada no sólo consigue acceder a un recurso, sino que es capaz de cambiarlo. Se produce así un ataque contra la integridad; **iv)**"Ataques por Falsificación (Phising)", Cuando una entidad no autorizada inserta objetos falsificados en el sistema. Este es un ataque contra la autenticidad.

2.4. Métodos de defensa contra ataques de negación de servicio

Las maniobras de prevención tienen el propósito de intentar eliminar la posibilidad de que un ataque se realice antes de que este se lleve a cabo de manera real. Estos acercamientos permiten implantar cambios en los protocolos, aplicaciones y sistemas para robustecerlos contra los intentos de ataque. La prevención, referida a los ataques DoS, tiene como objetivo disminuir el riesgo de sufrir algunos de los ataques de vulnerabilidad, además de dificultar al atacante la tarea de conseguir una cantidad de agentes elevados y reduce las probabilidades de éxito del ataque. Pero, aunque la prevención juega un papel primordial para la seguridad, de ninguna manera elimina la amenaza que suponen los ataques de denegación de servicio. En el campo de la prevención de ataques DoS, se podrían clasificar las posibles medidas en cuatro grandes grupos: **i)**" Mecanismos de seguridad del sistema", son mecanismos que tratan de incrementar la seguridad global del sistema, mediante la defensa contra accesos ilegítimos, eliminando bugs en las aplicaciones, actualizando las implementaciones de los protocolos para evitar intrusiones y la utilización del sistema con fines delictivos; **ii)**" Mecanismos de seguridad en protocolos", Son aquellos que abordan el problema de un diseño defectuoso en los protocolos de comunicaciones; **iii)**" Mecanismos de supervisión de recursos", Son aquellos que controlan el acceso de cada usuario a los recursos, fundamentándose en los privilegios que posee dicho usuario y en su conducta; **iv)**" Mecanismos de multiplicación de recursos", Son aquellos que pretenden dotar de abundantes recursos a los sistemas para debilitar la amenaza que supone el agotamiento de los mismos por parte de un posible ataque DoS.

3. ESCENARIO DE PRUEBAS DE ATAQUES DoS.

El escenario fundamental sobre el cual se desarrollarán los ataques, está compuesto por una red de computadoras primaria, que en ciertos casos hará las veces de Internet (red externa); además, se dispondrá de una red interna, compuesta principalmente un Servidor de tipo Mandriva Linux. El mencionado equipo, trabajará implementado como servidor de Web (HTTP) Apache 2.2.1, correo (SMTP / POP / IMAP), DNS,

FTP, entre los servicios más importantes. La red sobre la que será implementado el escenario de pruebas, consiste en un red con topología en estrella de tipo fast-Ethernet con una velocidad de trabajo en modalidad auto negociada 10/100 Mbps. Se trata de una red cableada con dos enrutadores independientes, que servirán para separar igualmente dos redes que servirán para efectuar los ataques respectivos. Para efectuar los ataques externos se utilizará como medio principal, la primera red mencionada anteriormente. En cambio, para los ataques internos a nivel de la intranet, se usará una máquina conectada físicamente a la segunda red. Ya que ambas redes poseen un enrutador independiente para cada una; el enrutador de la primera red fungirá como si se tratase de un ISP, debido principalmente a que estará conectado directamente a Internet, haciendo de Gateway por defecto al enrutador de la segunda red. Finalmente, las dos redes se conectan a través de un switch 10/100 no administrable. Las principales razones por la que se decidió la implementación de este pequeño laboratorio de pruebas de ataques DoS fueron: Por un lado, la facilidad de efectuar cualquier tipo de ataques sin representar una amenaza real a un servicio o proveedor verdadero, ya que se trata de una experimentación con propósitos educativos, además de éticos. Por otro lado, el hecho de poder tener una disponibilidad directa e inmediata de los equipos, facilita enormemente su configuración, monitoreo, activación y desactivación de servicios, etc. Con este escenario se tiene un conocimiento más profundo del funcionamiento de los ataques como se ve en la Fig. 1.

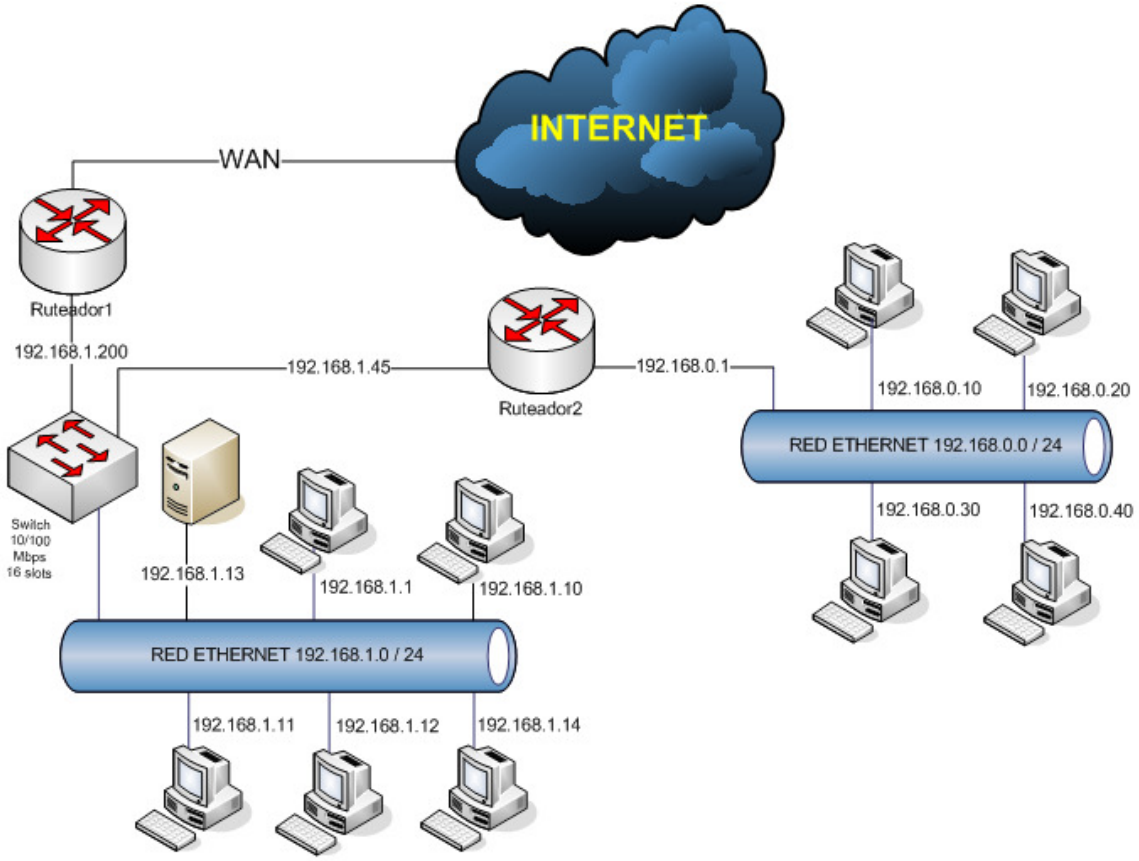

Figura 1:Diagrama del escenario de pruebas

Con el propósito de obtener un esquema de monitoreo del rendimiento inicial de nuestro servidor, se utilizó la herramienta de monitoreo de red Wireshark. La Fig. 2 muestra la captura de pantalla del rendimiento y el nivel de carga normal del servidor antes de efectuar los ataques. Se puede observar que el

pico promedio oscila en un valor cercano a 100 paquetes con tendencia a la baja, con una carga pico máxima de alrededor de 300 paquetes y una mínima de 50 paquetes.

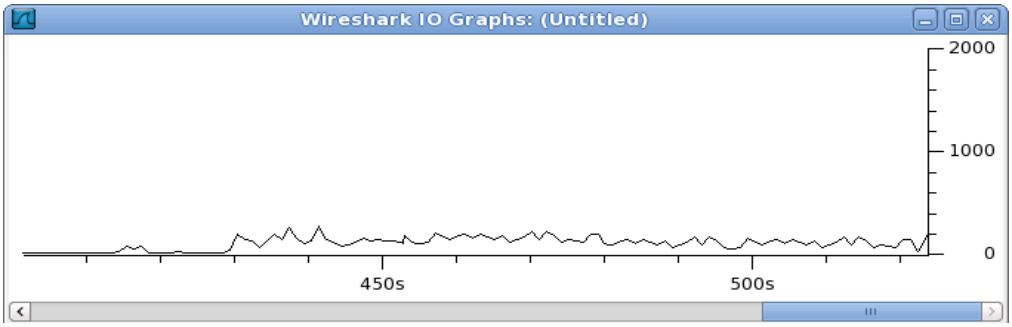

Figura 2. Captura de pantalla antes del ataque DoS y DDoS

3.1. Ejecución de los ataques con diferentes herramientas a diferentes servicios y protocolos.

En los tipos de ataques que se realizaron se utilizaron herramientas para los ataques de inundación y vulnerabilidad, para el ataque a través del protocolo ARP se realizó sin ninguna herramienta especializada, sino únicamente con el comando ARP, mediante la alteración de las tablas ARP. Este protocolo, se encarga de almacenar dentro de una caché interna, la lista de equivalencias MAC – IP de los diferentes equipos o hosts con quienes se ha comunicado. Este ataque está basado en modificar la tabla de traducción, a medida que se va estableciendo contacto con los hosts de la LAN específicamente con la Mac de la maquina atacada. Una vez que ya no se ha vuelto a establecer contacto con un determinado host, dentro de un tiempo prudencial, ARP automáticamente elimina el registro de su tabla de traducciones, con el propósito de dejar espacio a nuevas direcciones de otros hosts.

3.2. Ataque por Inundación HTTP (Flood)

Con el propósito de desarrollar un ataque más efectivo, se utilizo la herramienta [4] DoSHTTP v2.5.1 desde varias máquinas (DDoS), de tal manera que, se obtuvo una denegación de servicio de manera casi inmediata, ya que, por un lado el ancho de banda del objetivo fue inferior al ancho de banda de los atacantes, y por otro lado el ataque simultáneo de manera distribuida complico aún más la situación de la víctima. Una vez inicializado la herramienta en algunas máquinas atacantes, se ingresa la dirección correspondiente a la víctima, se escoge el agente de usuario o cliente Web que será visible en el momento de efectuar el ataque. Suministrada esta información, se especifica la cantidad de Sockets que abrirá nuestra herramienta, además que el envío de paquetes fue de manera continúa. La Fig. 3 muestra los componentes de la herramienta. Una vez configurado se procede a iniciar el ataque. Al acceder hacia el servidor HTTP de tipo Apache, la pagina Web de pruebas, ya no se encuentra disponible. Ya que el servicio fue saturado de peticiones, teniendo como resultado la demora del tiempo de respuesta del servicio (véase Fig. 3).

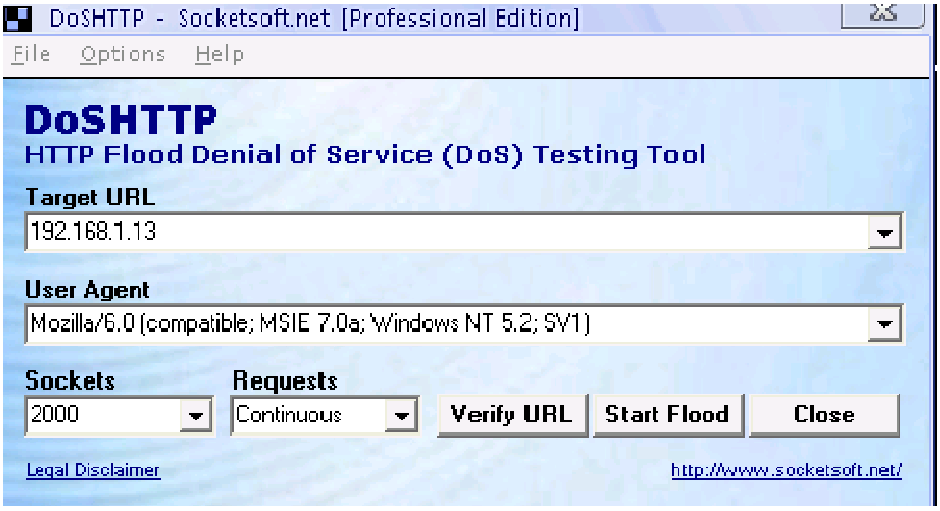

Figura 3. Inicializando la herramienta DoSHTTP

3.3. Ataque DoS por suplantación de direcciones con EtterCAP

Para efectuar este tipo de ataque, se utilizo una eficaz herramienta, que efectúa una inyección de paquetes de manera arbitraria hacia una máquina víctima, la misma que automáticamente alterará su tabla de direcciones ARP, capturando su tráfico mediante un ataque de tipo "Man in the Middle" o redirigiendo su tráfico a un punto muerto de una red o host no existente. Para que este ataque funcione se uso las herramientas: La conocida interfaz Winpcap 2.0, estándar de la industria para acceder a la conexión entre capas de red en entornos Windows y EtterCAP 0.7.3 para inyectar los paquetes. La Fig. 4 muestra los componentes y procedimiento para realizar el ataque.

En este ataque su nivel de efectividad fue muy elevado, pero también depende mucho del esquema de protección del que disponga el equipo o víctima a ser atacada. De hecho, este mismo tipo de ataque puede tener ciertas variantes o combinaciones dependiendo del tipo de resultado que se quiera obtener.

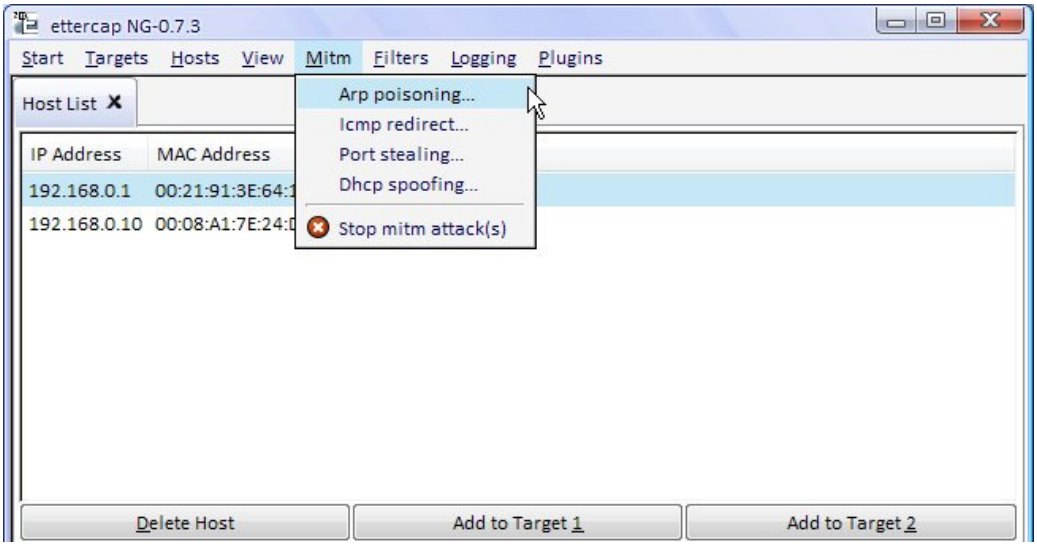

Figura 4. Iniciando el ataque MitM

3.4. *Ataques a DNS por medio de DNS Spoofing.*

Este ataque se realizo con el elementos DNS Spoofing, es un mecanismo muy poderoso de suplantación de un servidor de nombres de dominio; mediante esta característica se hizo pasar por un determinado host de Internet, y la víctima nunca se dio cuenta que en realidad está accedió a una dirección falsa forjada por la herramienta EtterCAP que trabajo con algunos plug-ins, que extiende ciertas funcionalidades del programa que normalmente no son posibles. Se suplanto una dirección específica apuntada en el servidor DNS legítimo. En este caso, se suplanto de manera temporal en la víctima la conexión de acceso hacia la página www.microsoft.com, de tal manera que se mostro otra página diferente en lugar de aquella, o también se puede realizar que aparezca bloqueada o fuera de servicio. En este tipo de ataque todo el resto de páginas no fueron afectadas, de tal manera que el usuario piensa que su conexión no ha sido intervenida de manera alguna, pasando de esta forma el ataque prácticamente desapercibido.

3.5. Ataques de redirección falseada del Servidor SMTP

Para lograr este ataque se manipulo el direccionamiento de los registros DNS impactando de esta manera las actividades normales, la productividad y por ende, la seguridad de las víctimas de este tipo de ataque. Utilizando el DNS Spoofing contra los registros hacia los cuales apuntan los servidores de correo electrónico de Google, atacando los puertos de una cuenta de gMail creada a propósito para esta tesis. Efectivamente, después de haber efectuado el ataque, el servidor SMTP de pruebas, apunto a una dirección no existente, por lo cual, el servidor de envío de mensajes gMail aparece como fuera de servicio o inaccesible; confirmándose de esta manera la denegación del servicio para la víctima del ataque.

3.6. Ataque a enrutador por medio de inundación

Para llevar a cabo este objetivo, se trabajo con dos herramientas, se utiliza Angry IP escanner v 2.2 como un eficaz escaneador de direcciones y puertos disponibles en nuestra red, podiendo observarla en la Fig. 5 donde nos despliega los puertos abiertos. Y una herramienta para ataques por inundación, llamada Server Attack. Se sometió a la red de pruebas al proceso de escaneo de puertos abiertos; excepto el caso del enrutador mismo que ha sido habilitado para redireccionar mediante NAT hacia una máquina ubicada dentro de la red local.

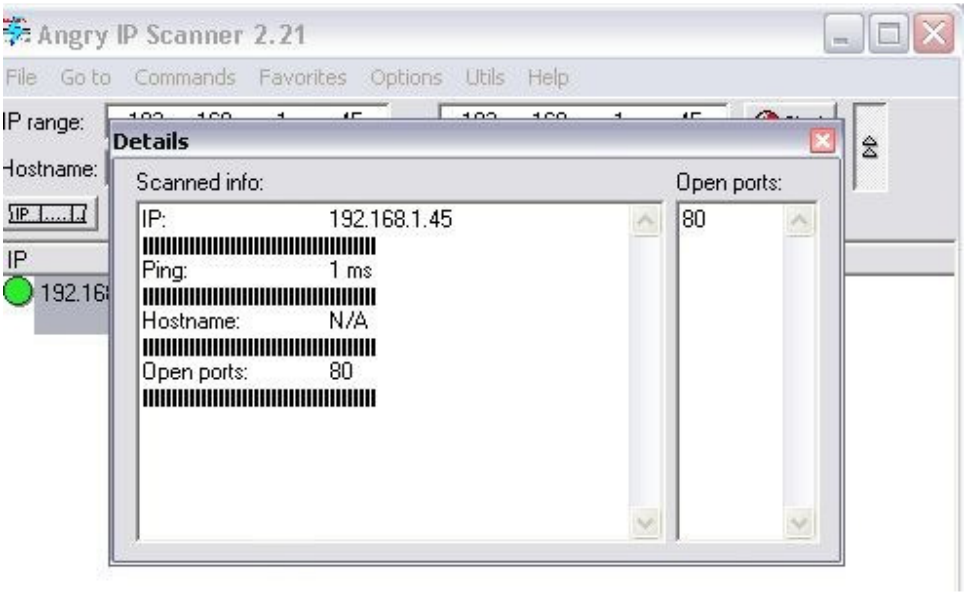

Figura 5. Escaneado de puertos con Angry IP

Dado que se conoce como se encuentra distribuida la red, se espero que el escaneador de puertos aparezca con el resultado. A continuación se ejecuto la aplicación Server Attack, con el cual, se realiza el ataque por inundación hacia el puerto abierto en el enrutador. Para tal efecto, se ingreso la dirección Ip como el puerto de la víctima que se ataco.

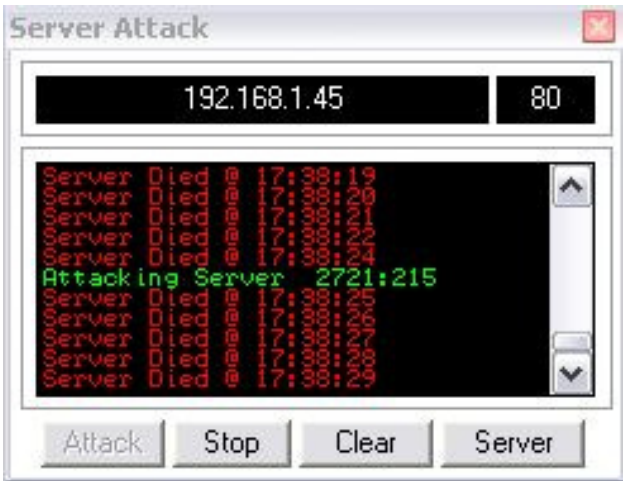

Figura 6. Ataque con la herramienta Server Attack

En la Fig. 6, se demuestra que el ataque se está llevando a cabo, se observa que aparecen unos mensajes de estado en color rojo; estos mensajes indican claramente que el ataque está siendo exitoso ya que no hay respuesta de parte del servidor, y por esa razón el log o bitácora de seguimiento de procesos aparece pintado de rojo, este ataque se desarrolla desde diferentes maquinas ejecutando algunos procesos en cada equipo atacante. Con el propósito de efectuar una verificación sobre el éxito del ataque, se procede a realizar un intento de acceso hacia el servidor y los resultados son negativos.

4. DISEÑO E IMPLEMENTACION DE UNA HERRAMIENTA Y SU APLICACION

La herramienta creada utiliza el Framework de desarrollo Java NetBeans IDE v 6.8 de Sun. El nombre del paquete de aplicación en cuanto se refiere a la clase base sobre la cual está definido este proyecto Java es URLFlooder. Adicionalmente, aprovecha la característica del trabajo con múltiples hilos o threads para efectuar su ataque de manera simultánea a medida que se procesan los Threads, dándole a la herramienta una capacidad de procesamiento paralelo, si se la ejecuta desde una máquina con 2 o más procesadores trabajando a la vez.

La interfaz de trabajo de la herramienta está compuesta por cinco campos de ingreso de datos, cuatro campos de visualización o resumen, un campo para controlar la duración de ataque en curso y tres botones que permitirán iniciar un ataque, detenerlo o cerrar la aplicación. Dentro del campo denominado URL o IP a atacar, se deberá ingresar la URL o dirección IP de la víctima a la que deseamos atacar, se tiene los campos para ingresar la cantidad de Threads (hilos) que se va a ejecutar, así como también la cantidad de Sockets por cada Thread que se ejecute, el indicador del número de puerto (por defecto 80), y finalmente, el tiempo de espera máximo permitido, mostrado en la Fig. 7.

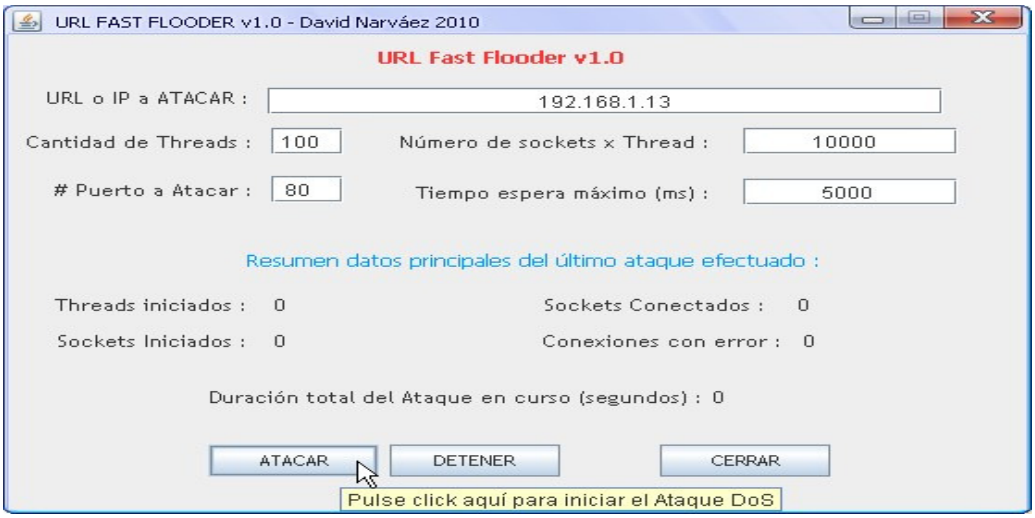

Figura 7. Ejecución de la herramienta Java

En el diagrama de secuencia que se muestra en la Fig. 8, se describen los eventos generados por los actores externos y su orden. El usuario ingresa los datos en el sistema y este envía la petición al servidor. El servidor envía los resultados y los despliega en pantalla al usuario (véase Fig. 8).

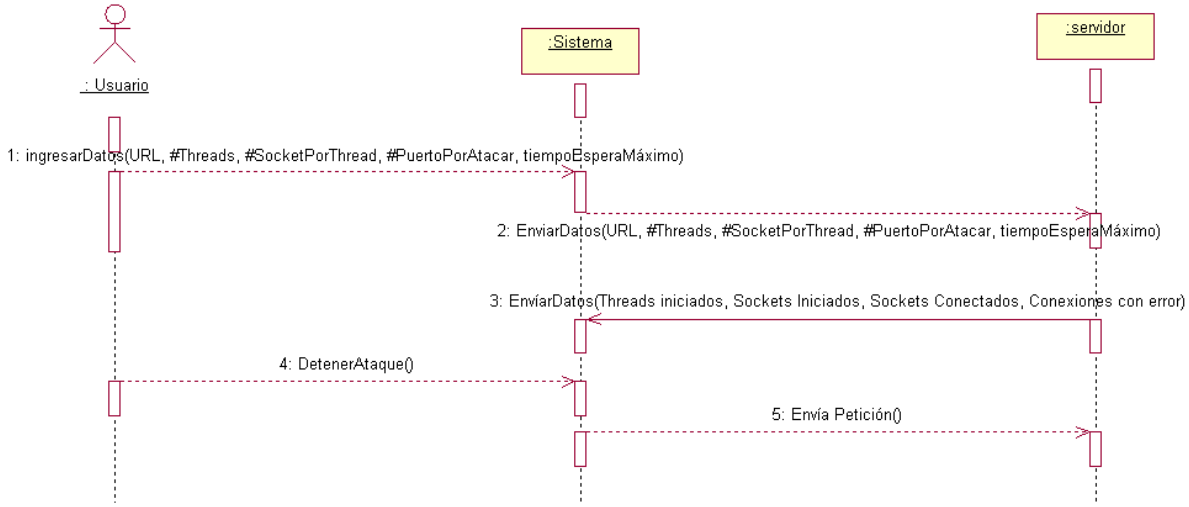

Figura 8. Diagrama de Secuencia de la herramienta

4.1. Implementación de Protección frente a ataques DoS y DDoS

Frente a estos ataques bien efectivos se optaron por mecanismos que se tiene dentro del servidor, es el firewall, con IPTABLES, que es el firewall más poderoso y difundido a nivel de servidores. Después de configurar del firewall con las reglas más adecuadas para establecer un nivel de seguridad óptimo para el servidor; se configura también las opciones inherentes a la distribución Linux sobre la cual se encuentra instalado y trabajando el Servidor de pruebas. Para lo cual, hay que dirigirse al centro de control principal de la interface Gnome instalada en este sistema, e ingresar a las opciones de seguridad correspondientes al servidor y configurarlas en niveles altos. Se limito los recursos utilizados por los usuarios en el sistema

como el uso de la memoria, la cantidad de procesos y sesiones concurrentes permitidos a éstos, incluso se limita las cuotas de espacio de disco disponibles para cada uno. Habilitar las entradas Proc y bloqueos de seguridad, básicamente corresponde a las opciones que se configuran mediante /proc/sys/net/ipv4, este set de entradas es un grupo de comandos para activar funcionalidades de seguridad que protegen al sistema contra varios tipos de ataques y vulnerabilidades conocidas; entre ellas el envenenamiento de ARP o la suplantación de direcciones y DNS basado en spoofing. Además de agregar seguridades extras como la instalación y configuración del módulo de Apache llamado MOD_EVASIVE, es una contribución desarrollada de manera independiente por Jonathan Zdziarski's[6]. Un científico e investigador que ha hecho este y otros aportes a la comunidad GNU. Además se activo algunos complementos que son paquetes de instalación RPM disponibles en la configuración de Mandriva Linux. Los paquetes, corresponden explícitamente a apache_devel y glibc_devel, los mismos que son requeridos obligatoriamente para poder compilar el archivo fuente suministrado por el creador de mod_evasive. Por último, y para dar por finalizado el proceso de aseguramiento del servidor de pruebas, se efectuó una protección extra hacia los ataques DNS Spoofing, activando la configuración que dentro del archivo de control al servidor DNS.

5. EVALUACION DE RESULTADOS

Con la implementación de políticas de filtrado de tipo IPTables configuradas, y módulos extras en los servicios levantados se volvió a realizar todos los ataques. Por lo cual, en todos los intentos de ataque, la configuración aplicada a MOD_EVASIVE, procedió a limitar la carga de la misma página hasta un máximo permitido de 2 veces por segundo. De sobrepasarse ese intervalo de tiempo, MOD_EVASIVE procedió a bloquear todos los paquetes cuyo origen sea el host que está sobrecargando las peticiones y que estén destinados al puerto 80 (http) del servidor. El equipo sospechoso de ser un atacante, ingresa automáticamente a la lista negra de MOD_EVASIVE por un intervalo de 10 minutos. Una vez transcurrido el período de cuarentena de un host puesto en lista negra, vuelve a permitir acceso nuevamente; En el caso de que el equipo atacante volviese a reincidir en la sobrecarga de paquetes, volverá a estar bloqueado de nuevo, hasta un máximo de 10 bloqueos sucesivos; posterior a lo cual, será bloqueado del servidor. Paralelamente al mecanismo de bloqueo, MOD_EVASIVE envia un e-mail de advertencia al administrador del servidor, con el propósito de informarlo acerca de las direcciones que fueron bloqueadas de los posibles atacantes, ya que probablemente se tratase de un ataque DDoS. Realizando una evaluación entre la activación de MOD_EVASIVE, se desprenden las siguientes diferencias:

En la Fig. 9 donde MOD_EVASIVE no está activo; el pico promedio oscila en un valor cercano a 800 paquetes / segundo, con una carga pico máxima de alrededor de 1500 paquetes / segundo y una mínima de 100 paquetes / segundo. En este ataque se registró el contingente de algunas máquinas atacantes.

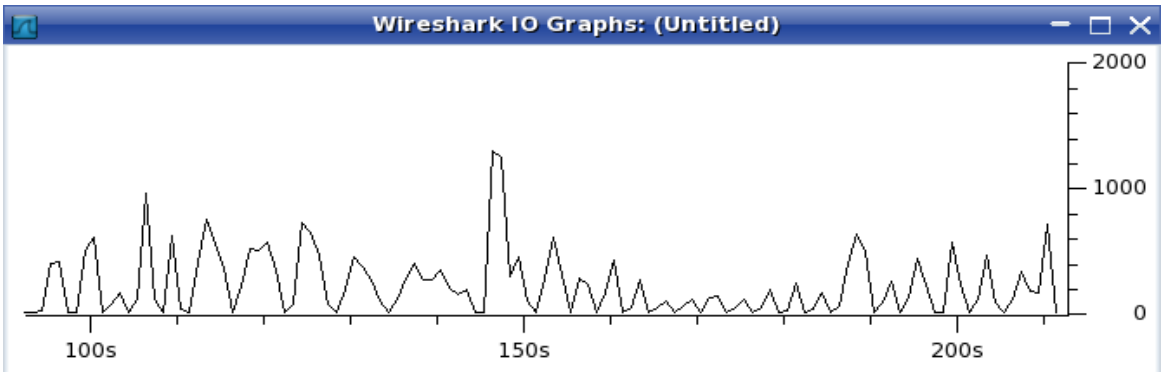

Figura 9. Ataque al Servidor Apache sin MOD_EVASIVE

En la Fig. 10 donde MOD_EVASIVE si está activo; se observa que el pico promedio oscila en un valor cercano a 250 paquetes/segundo con tendencia a la baja, con una carga pico máxima de alrededor de 550 paquetes/segundo y una mínima de 100 paquetes/segundo. En este ataque se registró igualmente el contingente de algunas máquinas atacantes.

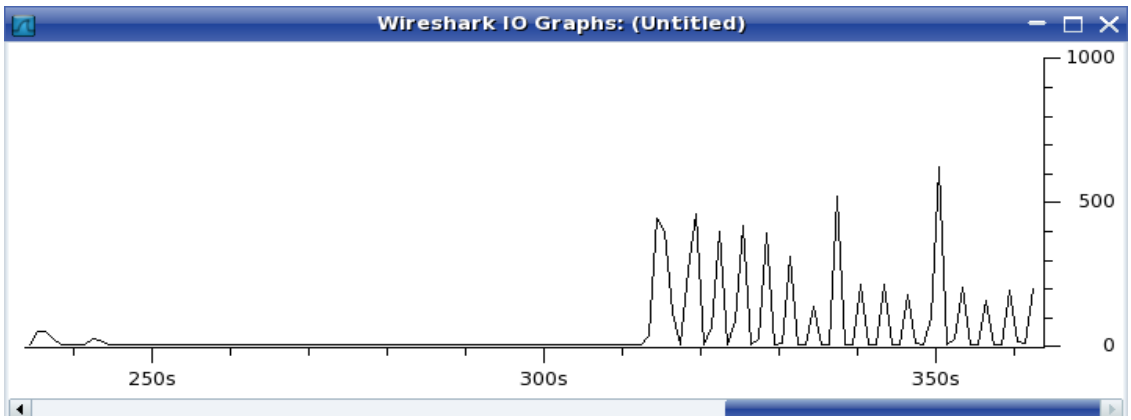

Figura 10. Ataque al Servidor Apache con MOD_EVASIVE activo

En efecto MOD_EVASIVE resultó eficaz para frenar los paquetes atacantes, reduciendo prácticamente en un 80% el nivel de carga del servidor mostrado en la Fig. 11, debido principalmente al bloqueo de los atacantes en base a la restricción de la cantidad de peticiones permitidas a un mismo host por segundo.

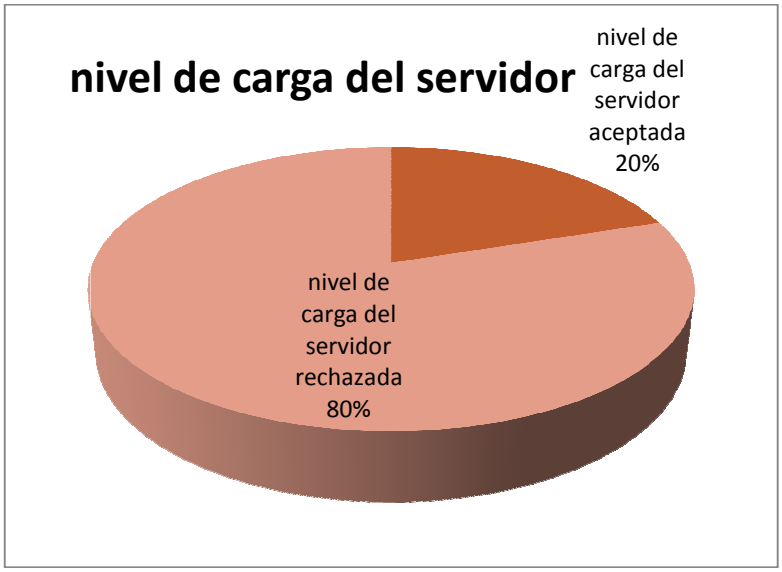

Figura 11. Conexiones rechazadas por MOD_EVASIVE

Se comprueba el nivel de eficacia de las protecciones aplicadas a nivel del servidor frente a este tipo de ataques; si bien es cierto, no serán 100% eficaces en todos los tipos de ataques actuales y probablemente menos los futuros; pero hoy por hoy, resultan ser herramientas de primera mano, conjuntamente con estrategias de seguridad adecuadas como análisis y monitoreo periódico del tráfico de red, mantenimiento, actualización de servicios y parches, respaldo de datos sensibles, protección de antivirus, malware y rootkits, entre otras.

6. TRABAJOS REALACIONADOS

Aunque exista una diversidad de trabajos relacionados, en esta sección se han incluido los más relevantes, que se han encontrado durante la investigación:

En lo que se refiere al tema de seguridad y ataques de DoS en redes IPv6, el trabajo presentado en [7] describe que el tráfico IPv6 está empezando a ser notable y la tendencia irá en aumento a medida que los operadores y proveedores de contenidos lo implementen en sus redes y servicios. IPv6 no es más inseguro que su predecesor IPv4, todo lo contrario. Sin embargo como cualquier otra tecnología, IPv6 ofrece la posibilidad de que gente maliciosa idee diversas formas de sacarle partido para realizar actividades fraudulentas. En lo que se refiere a métodos planteado por RR. Talpade [8] que consiste en el planteo de un sistema escalable de monitorización NOMAD, el cual, detecta los absurdos realizando un análisis estadístico de la información contenida en las cabeceras IP. Se puede utilizar para detectar anomalías en una red local, pero no nos permite clasificar el tráfico agregado que venga procedente de diferentes fuentes. Por los resultados obtenidos, se puede deducir que, como el método no reconoció en base a la información en las cabeceras de otras redes, podría no ser muy efectivo, en razón de que los ataques por inundación saben cómo saturar el ancho de banda.

Respecto a investigaciones basadas en técnicas de minería de datos (data minning) planteada por Stolfo [9], que revela los patrones en las características de un sistema que evaluara el comportamiento de los programas y los usuarios. Mediante aquello, se creó un clasificador que reconoció anomalías e intrusiones. Este procedimiento utiliza como información base a las variables o parámetros medidos en el propio sistema y no en los paquetes de información que viajan por la red. Con el propósito de mejorar esta técnica se utilizan los resultados provenientes de múltiples modelos para corregir la detección. Esta es una investigación a gran escala y aporto mucho para mi tesis ya que realiza las pruebas en escenarios reales. Respecto a investigaciones basadas detección de estructura de ataques de red, hicieron la formulación de que una estructura de datos heurística [10] recopila información fundamentándose en las direcciones IP de origen o destino. Cada elemento de una red recopila información estadística en una estructura de tipo multinivel, de tal forma que únicamente cuando una dirección o rango de direcciones supera un cierto nivel de tasa de tráfico dado se inician a compilar datos con mayor nivel de detalle. Es así que este sistema permite detectar el origen del ataque y a su vez las máquinas víctimas de un ataque. Una de sus desventajas es que requiere la reconfiguración de los encaminadores, mucha memoria y que no es capaz de detectar ataques con spoofing aleatorio generado por una sola fuente, o por un número lo suficientemente elevado de agentes, en nuestro proyecto el MOD_EVASIVE realiza el mismo trabajo sin usar tantos recursos. Para finalizar se recalca que todas las herramientas usadas fueron evaluados con algunos prototipos existentes, pero la gran diferencia fue la funcionalidad y eficacia para ejecutar los ataques, son herramientas súper básicas en su programación aprovecha la característica del trabajo con múltiples hilos para efectuar su ataque.

7. CONCLUSIONES Y TRABAJO FUTURO

En este artículo, se ha presentado la metodología general a seguir para la ejecución del ataque, ilustrando las implicaciones que ello conlleva, tanto en el lado del servidor atacado como en el del atacante. Con la aplicación creada, se ha probado y evaluado la alta capacidad y eficiencia obtenida por estos ataques, como se ha demostrado a través de la experimentación aportada en el trabajo. Para ello, una vez desarrollada las soluciones prácticas que posibilita dicha ejecución, se han propuesto mejoras a la misma y se ha comprobado el beneficio de sus efectos para el escenario de pruebas atacado.

Como trabajo futuro se podría evaluar los ataques en tráfico IPv6. Por tanto IPv6 sólo representa un nuevo canal por el que se podrían aprovechar vulnerabilidades de equipos, aplicaciones y también la seguridad en la Web 2.0.

Referencias Bibliográficas

[1] Macías Fernández Gabriel, "Ataques de denegación de servicio a baja tasa contra servidores". Tesis Doctoral Universidad de Granada ,2007. Pp(s): 115 – 139.

[2] Mieres Jorge, "Buenas prácticas en seguridad informática". Analista ESET La, [online:], abril 2010. http://www.eset-la.com/press/informe/buenas_practicas_seguridad_informatica.pdf.

[3] Limón Martínez Fernando, "Sistemas Distribuidos de Denegación de Servicio". Madrid, Junio de 2000, [online:] http://fi.upm.es/~flimon

[4] Web de la Agencia Europea de Seguridad de las Redes y de la Información (ENISA), [online,]http://www.enisa.europa.eu/, varios temas consultados

[5] Modelo de Clases , Objetos y Secuencia, [online:]http://www.dcc.uchile.cl/~psalinas/uml/modelo.html

[6] Jonathan Zdziarski's, "mod_evasive", [online:]http://www.zdziarski.com

[7] Miguel Angel Díaz Fernández, Cesar Olvera y Álvaro Vives, "Seguridades y ataques de DoS en redes IPv6". XII Congreso iberoamericano de Internet, telecomunicaciones y sociedad de la información, octubre 2009.

[8] Talpade R.R. Kim. S Khurana, "Simposium IEEE. Computers and Comm.",Pp(s): 324 – 335, 2006 [online] http://portal.acm.org/citation.cfm?id=1100973

[9] J. Stolfo Salvatore, Lee Wenke, Chan Philip K., Wei Fan, Eleazar Eskin. "Data mining-based intrusion detectors: an overview of the columbia IDS project". ACM Portal, December 2007, [online], http://www.cc.gatech.edu/~wenke/publications.html

[10] Thomer M. Gil y Massimiliano Poletto," MULTOPS" Simposium de Redes y Sistemas Distribuídos, November 2006 [online:] http://portal.acm.org/citation.cfm?id=1179542.1179557&coll=GUIDE&dl=GUIDE&CFID=93496650&CFTOKEN=73142641

Uso de la Web 2.0 en el Proceso Educativo para mejorar el Rendimiento Académico del Idioma Inglés

J. Aguas.
Departamento de Lenguas, Escuela Politécnica del Ejército, Sangolqui, Ecuador
Juan.aguas364@gmail.com

RESUMEN: La enseñanza tradicional del idioma Inglés se ha basado en el uso de recursos didácticos tales como: El pizarrón, textos, la grabadora y otros relacionados. Como consecuencia el Rendimiento Académico refleja un desequilibrio de las 'Competencias Lingüísticas'. Muchos estudiantes son buenos lectores 'Reading', escritores 'Writing', pero no muy buenos entendedores 'Listening' ni parlantes 'Speaking' del idioma Inglés (L2) [1]. Para muchos entender e interactuar con nativos hablantes ha sido una tarea difícil debido a que en el caso del texto por ejemplo, este no enseña el uso real del idioma. El/a estudiante aprende mucha gramática y vocabulario pero no pueden entender completamente lo que un/a nativo/a hablante dice, debido a que el lenguaje del usuario a diario incluye muchas expresiones idiomáticas y argot. Al observar este proceso educativo se advierte que cuando se aprende un idioma es necesario incluir otro tipo de herramientas (TICs) que de manera creativa le permitan al hablante del idioma español (L1) tener experiencias que lo pongan en contacto con el exterior (L2). Después de todo en esta era de la información quien enseña no solo es el docente sino el compañero, el amigo y hasta una comunidad entera [2]. Los resultados experimentales del uso de las TICs en el proceso educativo de idiomas muestran cambios fundamentalmente en las competencias lingüísticas vinculadas a la interacción. Sin embargo, en este campo queda mucho por hacer todavía especialmente, con el claustro docente.

Palabras clave: Web 2.0, rendimiento académico, enseñanza el idioma inglés, competencias lingüísticas

ABSTRACT: The English language teaching has been based on the use of traditional educational resources such as blackboard, texts, the recorder and other related elements. Consequently, the result has been an Academic Performance which reflects an imbalance of Language Skills. Many students are good readers (Reading), writers (Writing), but neither good listeners (Listening) nor speakers (Speaking) of the English language (L2) [1]. For many people understanding and interacting with native speakers has been a difficult task because in the case of text, for example, it does not teach the real use of English. Students learn too much grammar and vocabulary but cannot understand what real people say because native speakers use many idioms and slang. Therefore, this article based on this type of educational process considers using non-traditional resources resulting from the NCITs seeking to promote Meaningful Learning. By reason of it, the most popular media Web resources have been selected to add them an educational value to allow students to use the network as a platform where teaching is not only done by teachers but a partner, a friend or even an entire community [2]. Experimental results show some academic achievement in language skills primarily in those related to interaction. However, in the field of NCITs there is much yet to do especially with the faculty.

Keywords: Web 2.0, academic performance, teaching English, language skills

1. INTRODUCCIÓN

En la actualidad existe una creciente demanda por la enseñanza del idioma inglés (L2) y para satisfacer esta instancia se han creado muchos productos y servicios educativos que ofrecen instructores calificados y recursos para sustentar en forma eficiente esta necesidad. El estudiante es atraído a través de programas que ofrecen el dominio del idioma. No obstante, el efecto de algunos factores como la utilización de recursos didácticos tradicionales en el campo del proceso educativo del (L2) propone una evaluación de la situación en la que se encuentran quienes egresan de este tipo de programas y la dirección que según la época en que vivimos se debe avanzar.

Para entender cómo se desarrolla un proceso educativo de idiomas tradicional actual basta con recordar nuestros días como estudiantes en las aulas de colegio, de repente, aunque sea somero, tenemos una idea de un/a docente cuyo accionar era el uso constante del texto, el pizarrón, la radio grabadora y otros recursos como recortes de revistas, Xerox copias y uno que otro documento que con el pasar del tiempo se quedó olvidado en medio del texto de trabajo.

Esta inferencia es el sello característico de algo llamado el aprendizaje. Si bien la idea es generar en los estudiantes una base cognitiva que en su mayoría conserve información sensorial de forma permanente, el presente sistema de educación de idiomas favorece a que el conjunto de datos recogidos por el cerebro durante todo el proceso de formación se quede en una memoria de corto plazo donde el estudiante recuerda lo aprendido solamente, para las evaluaciones (orales, escritas) mas no así, para desenvolverse en un entorno donde el idioma de contacto es el inglés, por sorprendente que parezca, se están formando usuarios de habla española (L1) con exiguas habilidades, especialmente en las competencias lingüísticas del (L2) de la interacción (Listening & Speaking).

Los recursos utilizados tienen un valor característico en el proceso educativo, cuando oímos hablar de personas que interactúan en un medio donde el idioma de contacto es el inglés, sin haber asistido a un lugar de enseñanza regular de (L2) es entonces que tendemos a pensar que existe algún tipo de dispositivo biológico que hace que el momento del aprendizaje sucedan situaciones diferentes. Mucho se preguntan si existe algún tipo de capacidad intelectual diferente o si es que el área del cerebro donde se almacena las palabras, las frases y los conceptos gramaticales es diferente de una persona a otra. Gardner diría que las estructuras de la mente en cada individuo esta en un rango de siete a nueve categorías y un neurocirujano tal vez nos diría que cada cerebro tiene su propia individualidad.

El aprendizaje de idiomas como en cualquier otro campo del conocimiento requiere de innovaciones que permitan que los individuos se conecten al momento histórico y que además utilicen los recursos de la era actual y más importante aún que sus necesidades sean cubiertas. El hablar un idioma diferente al (L1) trae consigo implícito una necesidad, el sistema actual está fundado sobre una serie de recursos que no favorecen los objetivos educativos de inicio del milenio los cuales debería alcanzar la forma como el mundo se desenvuelve en el presente y el prominente futuro. Sin embargo, este estudio trata de concluir que no hay nada que indique que la enseñanza de idiomas tradicional en su mayoría, ya que no todo lo tradicional es malo, sea el mejor modelo en el proceso educativo a seguir.

Hoy en día, muchos usuarios especialmente en el ámbito de la educación universitaria inevitablemente se encuentran con los recursos que al momento ofrece. Las TICs (Tecnologías de la Información y Comunicación) agrupan los elementos y las técnicas utilizadas en el tratamiento y la transmisión de las informaciones, principalmente de Informática, Telecomunicaciones e Internet. En este último especialmente, los estudiantes se sirven de un sin número de recursos para crear redes sociales virtuales que son multi- propósitos (Facebook, Google, Twitter, Youtube, etc.). Sus motivaciones son diferentes pero se ha determinado que la necesidad personal esta seguida por un fenómeno social que le da la oportunidad al/los usuario/s de resolver un sin número de problemas en forma colectiva y dinámica mediante el uso compartido de la información.

Un estudio llevado a cabo en el continente europeo reportó que el acceso a redes sociales, a la búsqueda de información, escuchar música, subir y bajar videos, compartir mensajes instantáneos, etc., ha disminuido do el rendimiento académico de los estudiantes y esto sugiere que muchos de estos recursos deben restringirse durante el proceso educativo ya que son distractores. Si es así, entonces, se impediría

que las tecnologías de la información y la comunicación puedan lograr su principal objetivo que es poner al servicio de cada persona las herramientas para llegar a los objetivos de Desarrollo del Milenio. Muchos de los instrumentos de la red existentes actualmente no fueron creados para ser utilizados como recursos didácticos, no obstante el uso del Internet (Web 2.0) tiene un gran potencial cuando se habla de 'Aprendizaje Significativo', dónde los sistemas informáticos se constituyen en una plataforma para sacarle más provecho a lo aprendido por parte de los estudiantes.

La Web 2.0 es un fenómeno social que se logra a partir de diferentes aplicaciones en la Web y que le da la oportunidad al usuario de pasar de ser consumidor a un productor de información. Estas herramientas para ser aplicadas en el proceso educativo necesitan como agregado un valor didáctico y un protocolo para su utilización, pues en caso contrario se supone una distracción de la atención del objeto de estudio.

El gran reto frente a este paradigma de la tecno-ciencia, es hacer que esta nueva forma de utilizar el Internet proporcione recursos didácticos que le permita al docente crear un ambiente de enseñanza que incorpore una característica significativa y a cada estudiante generar un tipo de codificación en el momento del aprendizaje para que la recuperación de la información sirva no solo para el momento del examen sino para resolver problemas en su vida diaria [3].

Así como en el caso de otras áreas de la ciencia, también en la lingüística aplicada al idioma inglés se quiere explorar el camino a un nuevo neologismo en evolución del Web llamado Web 3.0. Ello incluye, la transformación de la red en una base de datos, un movimiento hacia hacer los contenidos accesibles por múltiples aplicaciones, el empuje de las tecnologías de Inteligencia Artificial, la Web semántica, la Web Geoespacial, o la Web 3D.

El resto del artículo ha sido organizado como sigue: La sección 2 describe aspectos del aprendizaje basado en herramientas de la Web 2.0. La sección 3 detalla el diseño e implementación del experimento. En la sección 4 se muestran los resultados experimentales. En la sección 5, se analizan algunos trabajos relacionados. Finalmente, en la sección 6, se presentan las conclusiones y líneas de trabajo futuro sobre la base de los resultados obtenidos.

2. EL APRENDIZAJE BASADO EN HERRAMIENTAS WEB 2.0

En un ambiente educativo, los estudiantes aprenden contenidos de diferentes categorías pero también desarrollan habilidades intelectuales asociadas a esos aprendizajes tales como representar la realidad, elaborar juicios de valor, razonar, inventar o resolver problemas de varios tipos, al tiempo de que aprenden otras habilidades comunicacionales que son importantes en su proceso de socialización [4]. Con la incorporación de las nuevas tecnologías al ambiente educativo se supone una formación holística que permita tanto a docentes y estudiantes ampliar conocimientos en una forma que era imposible hasta tan solo unos años atrás. Sin embargo, los ambientes de aprendizaje no se dan de manera automática, no surgen como generación espontánea ni son tampoco resultado de las nuevas tecnologías [8].

Es por esto que un diseño pedagógico que tome con seriedad a las Nuevas Tecnologías en Información y Comunicación es imperativo. Cuando se diseñan ambientes de aprendizaje basados en el computador se debe generar al mismo tiempo cambios en la naturaleza tradicionalista de autoridades, docentes y dicentes, incluso del gobierno y la sociedad respecto al proceso educativo y sus diferentes aristas. Esto implicaría la modificación completa del entorno educativo para adaptarlo a un verdadero enfoque de uso de tecnologías.

"La enseñanza en el siglo XXI, supone un ambiente rodeado de tecnología despojado de las limitaciones del pasado, en donde el acceso a la información era extremadamente restringido. Hoy el aprendizaje esta bañado por los grandes avances científicos. Un entorno de aprendizaje utilizando la Web se fundamenta en aspectos pedagógicos como: la actividad, la individualización, progresión, retroalimentación inmediata, el valor del error, y aplicación inmediata de lo aprendido. Esto valida y fundamenta la idea que la educación y la capacitación con visión holística es un sistema centrado en el estudiante mediante el uso didáctico de las TICs. Esto, por su puesto posibilitaría la individualización del proceso educativo centrando la metodología al aprendizaje y no a la enseñanza cuyo resultado final sería el aprendizaje autónomo [5].

La individualización que se crea en un ambiente educativo cuya plataforma son las TICs se adapta a cada persona y le permite avanzar al ritmo que pueda o desee llevar. A si mismo, está de por medio la progresión que permite al estudiante dosificadamente adquirir conocimientos que van desde lo más simple hasta lo más complejo. La retroalimentación inmediata proporciona información eficaz y precisa sobre cada una de las respuestas del usuario, lo que aumenta su nivel de refuerzo y motivación. El valor del error es la evaluación de los errores de los usuarios (estudiantes) y se convierte en el mejor camino para aprender. Finalmente, la aplicación inmediata de lo aprendido está pensada para realizarse en el puesto de trabajo (en la universidad), por lo que las posibilidades de utilizar los conocimientos aprendidos aumentan.

Para emplear las TICs en el aula hay que tener una idea clara y definida de cómo organizar situaciones de clase apoyadas en el uso de la tecnología [6]. Centrarse en el ambiente de aprendizaje no puede reducirse al análisis de la organización del espacio y tiempo educativos. La unidad básica de espacio educativo (salón de clase) y la unidad básica de tiempo ("la clase") se ven afectadas por la aparición de nuevas tecnologías de la información. Algunas consideraciones son importantes especialmente para el docente al momento de implementar en el proceso educativo el uso del computador. Lo relevante debe ser siempre lo educativo, no lo tecnológico. Por ello, un docente cuando planifique el uso de las TIC_s siempre debe tener en mente qué es lo que van a aprender los alumnos y en qué medida la tecnología sirve para mejorar la calidad del proceso educativo que se desarrolla en el aula. [7]

Los nuevos entornos llevarán a que el docente deje de ser el transmisor exclusivo de información, pasando a desempeñar el rol de diseñador de situaciones mediadas de aprendizaje y creador de hábitos de destreza en los estudiantes para la búsqueda, selección y tratamiento de la información [8]. El docente debe ser consciente de que las TIC_s no tienen efectos mágicos sobre el aprendizaje ni generan automáticamente innovación educativa. Es el método y estrategia del uso de los recursos didácticos junto con las actividades planificadas las que promueven el tipo de aprendizaje.

Se deben utilizar las TIC_s de forma que el estudiante aprenda "haciendo cosas" con la tecnología. Es decir, se debe organizar en el aula experiencias de trabajo para que el estudiante desarrolle tareas mediante las TIC_s, observando aspectos tales como: la fuente de los datos, la manipulación de objetos digitales, los valores ético digitales, etc. Al tener entornos más abiertos y flexibles, habrá un desempeño mayor y una tendencia más amplia a adquisición de nuevas competencias, destacando el rol pasivo, memorización y repetición de la información para la solución cognitiva de problemas, la localización, reflexión y discriminación de la información, el control activo de los recursos de aprendizaje, y la adquisición de una actitud positiva para la interacción con y desde las tecnologías.

Las TIC_s pueden ser utilizadas tanto como herramientas para la búsqueda, consulta y elaboración de información, como para relacionarse y comunicarse con otras personas. Es decir, debemos propiciar que el estudiante desarrolle con las TIC_s tareas tanto de naturaleza intelectual como social. Esto le servirá para desenvolverse en la sociedad del futuro que será una sociedad de aprendizaje y del aprendizaje a lo largo de toda la vida. Las TIC_s deben ser utilizadas tanto para el trabajo individual como para el desarrollo de procesos de aprendizaje colaborativo. Se debe pensar en un proceso de aprendizaje que pasa por experiencia, meditación, formación de conceptos y comprobación de conceptos.

Cuando se planifica una lección, unidad didáctica, proyecto o actividad con las TIC_s debe hacerse explícito no sólo el objetivo y contenido de aprendizaje curricular, sino también el tipo de competencia o habilidad tecnológica / educativa (lingüística) que se promueve en el estudiantado. Esto incluye la definición de aprender como adquirir conocimiento o habilidad. Es saber por qué, es la parte conceptual del aprendizaje, por qué algo ocurre o funciona. La habilidad es el saber cómo, es la parte de aplicación, tener la habilidad para utilizar el saber por qué para hacer que algo ocurra.

Cuando llevemos al estudiante al uso de las TIC_s debe evitarse la improvisación, esto con el fin de evitar la frustración y aumentar la cognición de los estudiantes. Es muy importante tener planificados el tiempo, las tareas o actividades, los agrupamientos de estudiantes, el proceso de trabajo, etc. De esta manera el docente sabe qué enseñar, cómo enseñar y aprende informaciones relevantes del estudiante proporcionando un aprendizaje personalizado.

Usar las TIC_s no debe considerarse ni planificarse como una acción ajena o paralela al proceso de enseñanza habitual. Es decir, las actividades de utilización de los ordenadores tienen que estar integradas y

ser coherentes con los objetivos y contenidos curriculares que se están enseñando. Así el ordenador es concebido como un dispositivo economizador de trabajo, orientado a la cognición.

El buen uso didáctico de las TIC_s siempre enriquece el proceso educativo. Además, situados en esta sociedad de la información que exige una fuerte disminución de las prácticas memorísticas/reproductoras en favor de las metodologías socio-constructivistas centradas en los estudiantes y en el aprendizaje autónomo y colaborativo, los entornos sociales para la interacción que ofrecen las aplicaciones de la Web 2.0 constituyen un instrumento idóneo para ello. No obstante, hay que tener en cuenta que en general constituyen herramientas avanzadas que solamente las utilizará en las aulas el docente que disponga de recursos, formación y experiencia en el uso educativo de las TIC_s.

Con el término Web 2.0, subrayamos un cambio de paradigma sobre la concepción de Internet y sus funcionalidades, que ahora abandona su marcada uni-direccionalidad (Web 1.0) y se orientan más a facilitar la máxima interacción entre los usuarios y el desarrollo de redes sociales (tecnologías sociales) donde pueden expresarse y opinar, buscar y recibir información de interés, colaborar y crear conocimiento (conocimiento social), compartir contenidos. En este contexto, podemos distinguir: *i)* Aplicaciones para expresarse, crear, publicar y difundir: Blog, Wiki; *ii)* Aplicaciones para publicar, difundir y buscar información: *Youtube, Flickr, Slideshare, Cálameo, Bookmarks*; *iii)* Aplicaciones para buscar y acceder a información de la que nos interesa estar siempre actualizados: *RSS, blog, GoogleReader*; *iv)* Redes sociales: *Facebook, Twitter*.

Tecnológicamente, las aplicaciones Web 2.0 son servicios de Internet, por lo que no es necesario tener instalado un software en el ordenador. Así, la plataforma de trabajo es la propia página Web, que suministra herramientas on-line siempre disponibles y proporciona espacios de trabajo colaborativo. Las implicaciones educativas de la Web 2.0 en definitiva permiten: buscar, crear, compartir e interactuar on-line [8].

3. IMPLEMENTACION DE LA WEB 2.0 EN EL PROCESO EDUCATIVO

En esta sección se describe el método utilizado para la implementación de las herramientas de la Web 2.0 como recursos didácticos en el proceso educativo del (L2). La Fig. 1 muestra las etapas y procedimiento de la implementación que fueron utilizados y que serán analizados a continuación, de manera secuencial:

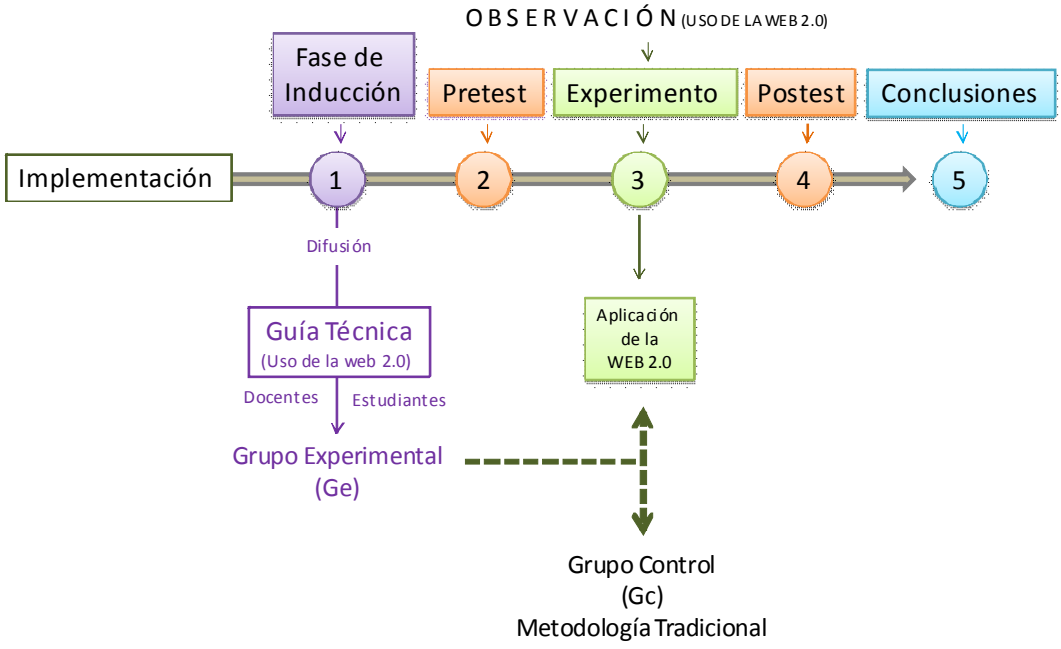

Figura 1. Procedimiento para la implementación de la Web 2.0, durante el experimento

3.1 Inducción

En la primera fase llamada de inducción se preparó una guía técnica para el uso de las herramientas Web 2.0 dirigida al docente y estudiantes del grupo experimental. El contenido de este documento es una guía interactiva que les permite a los usuarios entender cómo funcionan estos elementos mediante videos demostrativos (inglés y español) y además con links que les permiten crear cuentas en la red. Además, se incluyeron alternativas metodológicas que combinadas con el programa estándar de idiomas y recursos Web para cada competencia lingüística, haga que el estudiante mientras trabaje con los programas, mejore sus resultados. Finalmente, y para la obtención de datos numéricos se proveyó en la guía del docente un sin número de propuestas para la evaluación de los resultados en *Writing, Reading, Listening & Speaking* (escribir, leer, escuchar y hablar) mediante el uso de matrices de valoración (rubrics, en inglés).

3.2 *Pretest y Postest*

La evaluación diagnóstica (Pretest) se llevó a cabo con la finalidad de medir el nivel de dominio de la lengua considerando las cuatro competencias del idioma (*Writing, Reading, Listening & Speaking*). El procedimiento de elaboración del pretest se construyó considerando las siguientes etapas:

- *Prueba piloto*.- Aquí se realizó la administración de la prueba, su duración, las instrucciones, el contenido, con un grupo reducido de personas. También la prueba de verificación de los ítems de evaluación a hablantes nativos de la lengua (L2) con el objetivo de controlar el contenido y revisar la calidad de cada pregunta formulada.
- *Ensayo general*.- En él se comprobó de nuevo la administración, la duración, las instrucciones, el contenido y la clave, pero esta vez con tantas personas como sea posible y de nuevo con hablantes nativos del idioma.

Transcurrido el tiempo previsto para la aplicación de las TICs en el proceso educativo del grupo experimental (Ge) y de la metodología tradicional en el grupo control (Gc) se aplicó un Postest a los dos grupos, esta fue del tipo sumativa ya que se aplicó al final del experimento. La evaluación final estuvo enmarcada en las normas que el Marco Común Europeo dicta para medir el nivel de comprensión y expresión oral y escrita de la lengua (L2) para usuarios de nivel A1: acceso.

La preprueba y posprueba determinaron datos donde se calculan las estadísticas de distribución, es decir: *media*, la nota media de una prueba; *moda*, la nota obtenida por el mayor número de estudiantes; *mediana*, la nota que se encuentra en la mitad de los resultados obtenidos por la totalidad de los estudiantes; *rango*, la diferencia entre las notas más altas y las más bajas de una prueba y *desviación típica*, la cantidad media aproximada en que la puntuación de cada estudiante se desvía (o difiere) de la media.

La estructura de los dos test se puede apreciar en la Tabla 1 (véase Tabla 1):

3.3 Observación metodológica

El instrumento estuvo dirigido a observar a los docentes y estudiantes de los grupos control y experimental durante el estudio, haciendo referencia a las actividades tanto dentro y fuera del aula mediante las actividades vinculadas a los recursos didácticos. La observación realizada fue cuantitativa como un registro sistemático, valido y confiable. Consistió en un registro de la tasa de frecuencia del uso de los recursos didácticos y de incidentes de comportamiento en relación a las dos variables de esta investigación (V_I: TICs / V_D: Rendimiento Académico) en el curso normal del proceso educativo.

El observador fue una persona totalmente ajena a la investigación (ciego) y su rol fue solo de registro y no participante durante el experimento. Previa a su labor se llevó a cabo un periodo de introducción del manejo del instrumento de observación en relación a las variables, sus indicadores, forma de registro y rol durante el experimento en los dos grupos. Adicionalmente, se le capacitó para identificar claramente cuáles son las formas en las que pueden presentarse las variables para que no se realice el doble registro de un mismo evento. El observador realizó su actividad directamente y en forma presencial. Esto quiere decir, que ingresó a las dos aulas en los diferentes períodos asignados a los grupos y se ubicó en la parte

posterior de la clase. A los estudiantes se les informo de su presencia más no de su actividad. Para la observación se construyeron 17 matrices de observación para el grupo experimental y 17 matrices para el grupo control. Las dos medidas de observación del instrumento fueron la ocurrencia (si o no) y cuantas veces (frecuencia) utiliza los componentes (indicadores) de las dos variables expuestas. Cada variable está representada con todos sus indicadores para observar y registrar al grupo experimental y de control.

TABLA 1: Indicadores del pre y post test

NOMBRE DE LA HABILIDAD	INDICADORES	N. de preguntas	Forma del Test
LISTENING	El estudiante puede entender oraciones y expresiones cortas con claridad.	5	Opción multiple
	El estudiante puede entender diálogos cortos con claridad.		Opción multiple
	El estudiante puede abstraer información específica para entender la realidad		Opción multiple
	El estudiante puede utilizar "el contexto" para entender las circunstancias que relacionan un hecho.		Opción multiple
	El estudiante puede modelar el sonido de palabras, oraciones y preguntas de acuerdo a la entonación y acento		Opción multiple
SPEAKING	El estudiante realiza expresiones de la vida diaria con oraciones básicas	5	Oral
	El estudiante puede presentarse a sí mismo y proveer posee. información personal y de índole general y particular acerca de las cosas él/ella		Oral
	El estudiante puede interactuar con otras personas en forma simple pero lenta y claramente.		Oral
	El estudiante utiliza el vocabulario, gramática y funciones en contextos hablados con significado.		Oral
READING	El estudiante puede entender una variedad de material con expresiones básicas.	5	Opción multiple
	El estudiante puede extraer información específica		Opción multiple
	y general de todo tipo de contexto escrito en forma básica		Opción multiple
	El estudiante puede leer y utilizar esta para responder		Opción multiple
	preguntas abiertas, cerradas, selección múltiple, etc.		Opción multiple
WRITING	El estudiante puede escribir ideas y colocarlas juntas para formar textos simples e informales	5	Escritura
	El estudiante puede utilizar información visual, de audio y lectura para realizar piezas de discurso escrito.		Escritura
	El estudiante puede escribir artículos sobre la base del vocabulario utilizado		Escritura
TOTAL		**20**	

4. RESULTADOS EXPERIMENTALES

La Fig. 2 muestra los puntajes (rendimiento académico) obtenidos por los estudiantes de los grupos control y experimental en la evaluación diagnostica (pre-test), evaluación formativa y evaluación sumativa (pos test). En la evaluación diagnóstica (pre-test), las condiciones iniciales muestran una diferencia del 0.41de punto, el grupo control tiene una media de 14.23 puntos y el grupo experimental una media de 14,64 puntos. En la evaluación formativa (durante la ocho unidades didácticas), esto es evidenciar si los objetivos planteados en el proceso educativo se alcanzan o no. Los dos grupos tienen una tendencia a incrementar sus rendimientos académicos no obstante, hay una media el 0.8 de punto entre los puntajes obtenidos por los dos grupos entre cada unidad didáctica (**pre-test**: 0.41e+, **u1**: 0.98e+, **u2**:0.73 c+, **u3**:0.58e+, **u4**:0.79e+, **u5**:0.85e+, **u6**:0.77 e+, **u7**: 1.06 e+, u8:0.91e+, **pos-test:** 1.00e+) estos puntos medios entre las calificaciones de los grupos en cada unidad demuestran que el grupo experimental es el que tuvo mayor progreso en cuanto al proceso educativo.

Finalmente, en la evaluación sumativa (post-test) en el cual se juzga el aprendizaje para diferentes fines, la diferencia entre los grupos es de 1,00 puntos. El grupo control termina su proceso educativo con una media de 16,00 puntos y el grupo experimental con una media de 17,00 puntos.

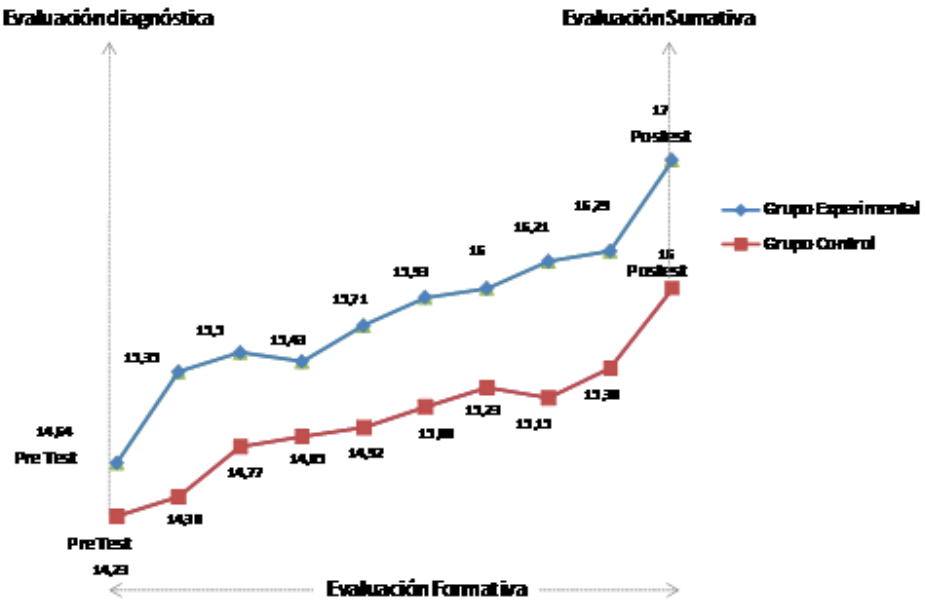

Figura 2. Resultados del experimento, evaluación diagnostica, formativa y sumativa

La Fig. 3 muestra los resultados del grupo del pre y post test de los grupos control y experimental. En Listening hay una diferencia porcentual del 28%, en el pre-test el porcentaje es del 42% y el 70% del pos test. En Speaking la diferencia es del 11%, en el pre-test el porcentaje es del 59% y pos test es del 70%. En Reading la diferencia es del 4%, el pre-test tiene un porcentaje del 64% y el pos test el 60%. En Writing los resultados tienen una diferencia del 1%, en el pre-test el porcentaje es del 57% y el post del 58%.

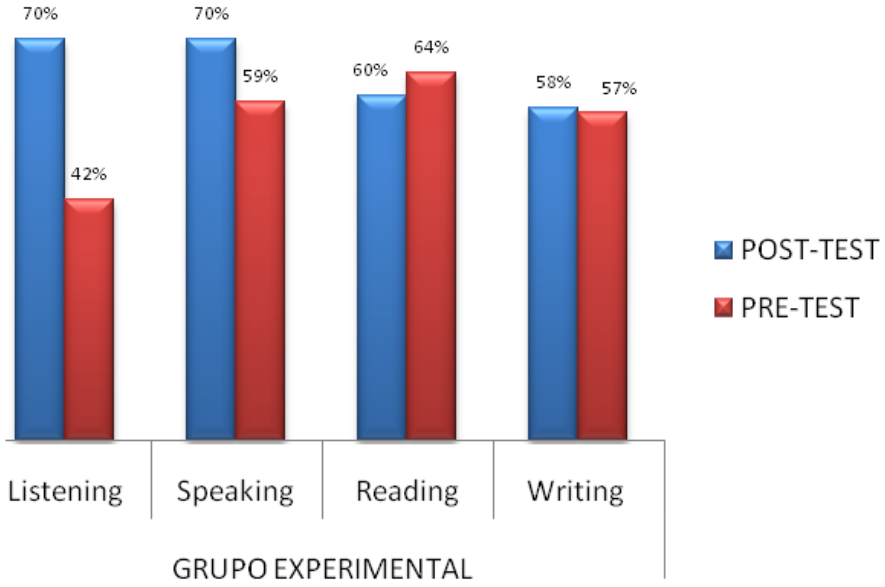

Figura 3. Resultados del experimento, por competencias lingüísticas (Grupo experimental)

La Fig. 4 muestra los resultados del pre y pos test del grupo control. En Listening la diferencia porcentual es del 28%, el resultado del pre-test es del 47% y pos test del 19%. En Speaking la diferencia porcentual es del 15%. En Reading la diferencia es del 10%, el pre-test registra el 44% y el pos test el 54%. Finalmente, Writing la diferencia es del 19%, el pre test es del 38% y el porcentaje del post test es del 38% y el porcentaje del post test es del 57%.57%.

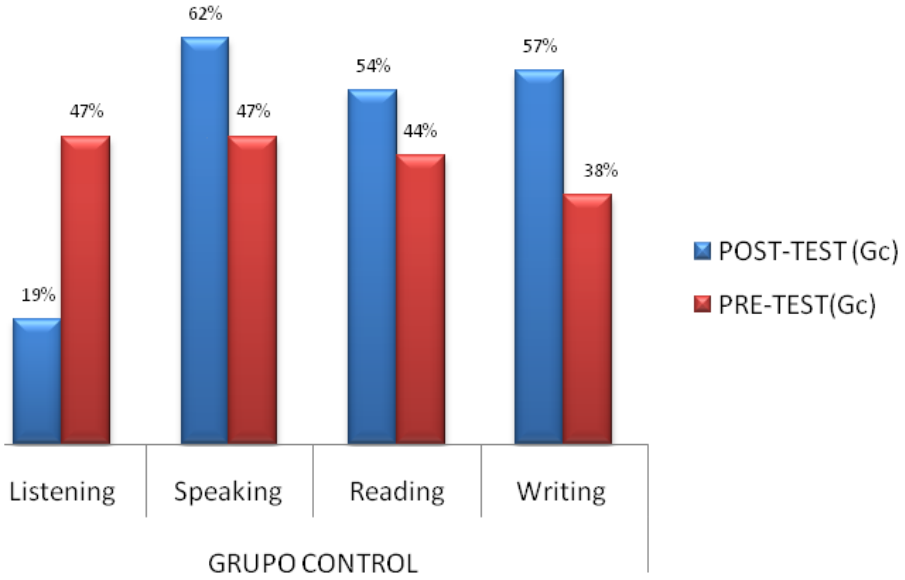

Figura 4. Resultados del experimento, por competencias lingüísticas (Grupo Control)

A continuación se presentan los resultados obtenidos en las pruebas post-test, prueba f y prueba t para varianzas desiguales del grupo control y experimental del estudio (Véase TABLAS 2, 3 y 4).

TABLA 2: Post test de los grupos control y experimental

POST - TEST	
GRUPO CONTROL	**GRUPO EXPERIMENTAL**
16	17
19	18
17	17
16	17
16	17
17	17
15	16
16	16
16	16
14	16
14	18
17	16
15	18
	18
	17

TABLA 3: Prueba F para varianzas de dos muestras (pos-test)

	CONTROL	*EXP*
Media	16	17,0933333
Varianza	1,83333333	0,66780952
Observaciones	13	15
Grados de libertad	12	14
F	2,74529378	
P(F<=f) una cola	0,03741262	
Valor crítico para F (una cola)	2,53424325	

TABLA 4: Prueba t para dos muestras suponiendo varianzas desiguales (pos-test)

	CONTROL	*EXP*
Media	16	17,0933333
Varianza	1,83333333	0,66780952
Observaciones	13	15
Diferencia hipotética de las medias	0	
Grados de libertad	19	
Estadístico t	-2,53820367	
P(T<=t) una cola	0,01002714	
Valor crítico de t (una cola)	1,72913279	
P(T<=t) dos colas	0,02005428	
Valor crítico de t (dos colas)	2,09302405	

5. TRABAJOS RELACIONADOS

Un artículo publicado por la Royal Economic Society en Londres, informó que "Los estudiantes que usan menos el computador obtienen mejores calificaciones". Para llegar a esta conclusión los alemanes Thomas Fuchs y Ludger Wossman, de la Universidad de Munich se basaron en análisis del rendimiento académico y sostienen que cuanto más acceso a una computadora tienen los estudiantes en su casa es menor el rendimiento escolar, en parte porque habitualmente las PC (Internet) los distraen de sus tareas. Asimismo, afirman que cuantos más sistemas informáticos hay en la escuela menor es la productividad de los alumnos, porque la enseñanza computarizada es menos eficaz respecto de la forma tradicional".

Este análisis, es para esta investigación el punto de partida ya que las conclusiones a las que llegan estos dos expertos europeos deberían ser evaluadas, tomado en cuenta las condiciones en las que el proceso educativo de nuestro sistema de educación universitaria 'local' se desarrolla.

Si bien el estudio europeo desmiente los resultados de una investigación anterior que abogaba por la progresiva informatización de la instrucción y métodos de enseñanza desde la llegada de la PC no sería apropiado, previa una investigación, inscribir esta realidad extranjera a la de nuestro país. Desafortunadamente, no existe una fuente especializada en el campo universitario que señale que hacen los docentes y estudiantes en un entorno educativo que proporciona un recurso tan importante en la formación académica como es el "Internet". Las tecnologías en la educación tienen un valor característico que necesita de un análisis ampliado. Es por esto que esta investigación utilizó fuentes de medición en línea y se recabará información para retro y prospectivamente tener una visión del mundo hacia nuestro entorno local en lo que se refiere al uso del computador conectado a la Internet.

6. CONCLUSIONES

En esta investigación se ha determinado que el uso de la Web 2.0 en el proceso educativo del idioma inglés como segunda lengua (L2) incide positivamente en el rendimiento académico de los estudiantes, especialmente en las competencias lingüísticas de la interacción que son el Listening y Speaking. La incidencia positiva de las TICs sobre el rendimiento académico se debe a que la enseñanza asistida por computadoras fortalece la individualización, la progresión, retroalimentación, el valor del error y la aplicación inmediata de lo aprendido. Se detectó que existen beneficios especialmente en la memoria a largo plazo que le capacitan al/la estudiante a utilizar los recursos Web para realizar la gestión de su propio conocimiento mediante diferentes formas para encontrar, aplicar y resolver un problema. El rendimiento académico es el producto de una conciencia educativa que nace del docente para hacer más atractivo, adecuado y exitoso el proceso de aprendizaje. Los alumnos que utilizaron el entorno Web 2.0 mejoran su capacidad de transferencia del conocimiento. Un estudiante consciente de sus propias estrategias meta cognitivas maneja: un *Control consciente de la atención;* las metas y objetivos de su propia formación; la reestructuración cognoscitiva que es archivar la información y recuperarla cada vez que sea necesario; y la *Autoevaluación* que es la propia supervisión de progreso hacia una meta. El uso de la Web 2.0 fortalece el mecanismo para codificar, almacenar y recuperar información de la memoria ya que el aprendizaje deja de ser mecánico o memorístico.

La información sensorial pasa directamente a la memoria permanente como consecuencia de un reforzamiento permanente de la sinapsis debido a la activación de ciertos genes y a la síntesis de las proteínas correspondientes. Este fenómeno se da por que el estudiante básicamente genera su conocimiento por interés o motivación personal más no por imposición. Las herramientas Web 2.0 proporcionan a los estudiantes la oportunidad de revisar el conocimiento y sus aplicaciones cada vez que sea necesario y hace que la información sea más consistente ya que gradualmente se va incorporando a la memoria de largo plazo. En cuanto a las competencias lingüísticas, los recursos didácticos provenientes de la Web 2.0 permiten que los estudiantes directamente participen con un mundo que se comunica por el Internet. El estudiante puede oír, escribir, leer e incluso hablar en tiempo sincrónico y asincrónico con otros nativos y no nativos que utilizan el inglés para la vida diaria o negocios.

Se concluye a partir del análisis de los datos procesados, que el uso de las herramientas Web 2.0 como recursos didácticos en el proceso educativo, permite que el estudiante mejore sus formas de resolución de problemas y el desarrollo de sus capacidades cognitivas en cuanto al proceso de adquisición del conocimiento. Este nuevo modelo de proceso educativo utiliza las novedosas y poco utilizadas herramientas de la Web 2.0 junto a técnicas enseñanza y de gestión del conocimiento, pues integra conceptos e ideas innovadoras provenientes de la didáctica, habitualmente aplicadas en recursos didácticos tradicionales pero poco estudiados en los no tradicionales.

Referencias Bibliográficas

[1] JONES B.F, PALINSCAR A.S. Ogle. D, S. & Carr. E.G, Learning and thinking in strategy & teaching and learning: Cognitive instruction in the content areas, Alexandria, 1987.

[2] SLEEMAN D, BROWN JS. Intelligent tutoring systems. London: Academic, 1982.

[3] JOHN MEDINA, , “Los doce principios del cerebro”, Bogotá, Primera edición, 2008.

[4] LÓPEZ OSTIO J. Sistemas Tutoriales Inteligentes (ITS). Conferencia mecanografiada. San Sebastián, España: 1993.

[5] ARNÁIZ SÁNCHEZ, P. ET (coord.) "intercultur@-net. formación telemática en interculturalidad. "diversidad para convivir: educar para no discriminar"(2002)

[6] SLEEMAN D, BROWN JS. Intelligent tutoring systems. London: Academic, 1982.

[7] ALONSO TAPIA, Jesús " Motivación y estrategias de aprendizaje. Principios para su mejora en alumnos universitarios". En GARCÍA-VALCARCEL, Ana (2001).

[8] ALONSO, LUIS "¿Cuál es el nivel o dificultad de la enseñanza que se está exigiendo en la aplicación del nuevo sistema educativo? Revista EDUCAR, 26, (2000).