Universidad de las Ciencias Informáticas

Facultad 6

Título: Procedimiento para realizar Pruebas de Seguridad a la Plataforma de Transmisión Abierta de Radio y Televisión (PTARTV)

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

Autores: Maydelin Navarro Suárez.

Tutor: Ing. Roexcy Vega Prieto.

Ciudad de La Habana, Junio 2011
"Año 53 del Triunfo de la Revolución"

"La lucha por la calidad del producto es una lucha revolucionaria y de vanguardia."

Ernesto "Che" Guevara

Agradecimientos

Este trabajo está dedicado principalmente a **mi mamá**, que es la luz que ilumina todos mis días y la razón por la cual hoy soy lo que soy y puedo tener esta oportunidad. Porque mami tú me enseñaste a ser mejor persona cada día. Quiero que me disculpes si en algún momento te he hecho sentir mal con mis palabras. Por todo eso eres alguien especial.

A **mi papá**, tu sabes papi que te quiero mucho y te agradezco todo lo que hiciste por mí en este año. Por haberte portado tan bien conmigo y por aguantar todo lo que te digo algunas veces, tú sabes que te quiero.

A **mi hermanito** que tanto me ha costado estar lejos de él y aunque no se encuentre aquí en estos momentos lo extraño y lo quiero mucho.

A mi **tatico** que para mi eres alguien especial. Nunca voy a olvidar todo el apoyo incondicional que me brindaste en los momentos más difíciles. Quisiera que nunca olvides que te quiero mucho y que a pesar que soy bien malcriada tengas paciencia conmigo como siempre lo has hecho, tatico gracias.

A mi prima **Stefani** y a mi tía **Barbarita** que de una forma u otra me han apoyado incondicionalmente durante estos cinco años.

A mi **abuelita** Olga que aunque en estos momentos no se encuentra aquí y vive un poco lejos de mi casa le agradezco su preocupación hacia mis estudios.

A mi **tutor** Roexcy por soportarme todo este tiempo, tener esa paciencia que siempre tuvo y agradecerle muchísimo la ayuda que me ha brindado sin él no podría haber realizado sola esta tesis.

A mis amistades que han estado conmigo en los momentos más difíciles y me han apoyado durante estos cinco años.

Les agradezco al tribunal por haberme apoyado y darme opiniones positivas, que gracias a sus críticas y revisiones a tiempo he podido mejorar.

Maydelin

Dedicatoria

Le dedico todo mi esfuerzo durante estos cinco años de universidad a mi mamá, a mi papá, a mi hermanito que aunque no se encuentra conmigo aquí en estos momentos lo tengo muy presente, ellos han sido las personas que me han enseñado todo en la vida, los que siempre me han apoyado y me han guiado por el buen camino. Y muy especialmente a mi abuelito Juan Manuel que a pesar de no estar ya junto a mí siempre supo darme mucho amor, comprensión y siempre fue mi orgullo. Y a ti mi amor que eres una de las personas que me has apoyado todo este tiempo y aunque no estabas aquí en la escuela conmigo siempre me diste apoyo y fuiste muy comprensivo.

A mi tía Barbarita que me ha apoyado en los malos y buenos momentos

Maydelin

DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo al Departamento de Señales Digitales de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Maydelin Navarro Suárez. Ing: Roexcy Vega Prieto.

_____ _____

DATOS DE CONTACTO

Tutor: Ing. Roexcy Vega Prieto.

- Ingeniero en Ciencias Informática, Universidad de Ciencias Informática.

E-mail: rprieto@uci.cu

RESUMEN

En la actualidad, muchas aplicaciones web son objeto de amenazas crecientes como la obtención o manipulación no autorizada de información que en muchos casos conlleva a pérdidas económicas entre otras consecuencias negativas. Es por ello que los desarrolladores de sistemas deben prestarle especial atención al tema de la seguridad y tomar medidas para detectar vulnerabilidades con el fin de poder corregirlas a tiempo. A partir de esta problemática, en el presente trabajo se propone un procedimiento para realizar Pruebas de Seguridad, contribuye a la prevención de diversas formas de ataque cuando el software está completamente elaborado. La solución está dirigida a la Plataforma de Transmisión Abierta de Radio y Televisión, un producto del centro de desarrollo Geoinformática y Señales Digitales perteneciente a la Universidad de las Ciencias Informáticas. Además se sugieren algunas herramientas de apoyo a este tipo de pruebas que complementa a la documentación organizada por pasos de las pruebas aplicadas, que al realizar el proceso de revisión y eliminación de errores, el software posee una mayor calidad y seguridad.

Palabras claves: Pruebas de Seguridad, Procedimientos, Aplicaciones web, Calidad

Abstract

Currently, many web applications are subject to growing threats such as obtaining or tampering of information which often leads to economic losses and other negative consequences. That is why system developers should pay special attention to security issues and take steps to detect vulnerabilities in order to correct them in time. From this problem, this paper proposes a procedure for Security Testing, contributes to the prevention of various forms of attack when the software is fully developed. The solution is aimed at Open Air Platform for Radio and Television, a product development center Geoinformatics and Digital Signals belonging to the University of Information Sciences. You can suggest some tools to support this type of evidence that supplements the documentation organized by test steps applied that in making the review process and eliminate errors; the software has improved quality and safety.

Keywords: Security Testing, Procedures, Web Applications, Quality

ÍNDICE DE CONTENIDO

TABLAS Y FIGURAS

Tablas

Figuras

INTRODUCCIÓN

Introducción

Con el vertiginoso desarrollo de las Tecnologías de la Información y la Comunicación (TIC) el mundo comenzó una nueva era, la era de la información. Grandes volúmenes de datos recorren todo el planeta y se hace indispensable su manipulación. El uso generalizado de Internet ha provocado que el flujo de investigación que transita por la red aumente cada día, en especial relacionada con archivos multimedia, entre los que se encuentran los videos digitales.

Debido al gran avance en la industria del software son muchos los métodos que a nivel mundial se utilizan con el objetivo de desarrollar un producto. Donde se manipulan técnicas y metodologías adecuadas que permiten una mayor productividad. Además se garantizan que sea confiable, preciso, flexible, rápido de usar y que esté bien documentado.

Actualmente y desde inicios de los 80, la calidad está en aumento en las empresas de todo tipo a nivel mundial. Estas lanzan constantemente compromisos acerca de la mejora de los productos, los servicios que desarrollan y ofrecen al mercado. La garantía se evidencia, cuando se haya tenido la capacidad de cumplir con las exigencias del cliente que juega un papel importante dentro del desarrollo de las operaciones que influyen positivamente en la decisión a la hora de escoger lo que necesita.

Cuba presenta en este momento un crecimiento en la producción de software con un interés por garantizar la calidad del producto que se origina. Teniendo en cuenta la importancia de la vinculación de todas las ramas de la economía cubana con el mundo informático es de primordial interés para el estado cubano, lo cual aporta grandes beneficios para el país, en cuanto a la creación de productos de software.

El país ha seguido los pasos de este avance tecnológico mediante la informatización de las diferentes ramas de la sociedad (económicas, políticas y sociales). La producción de software no sólo trae beneficios desde el punto de vista del desarrollo de sistemas para el uso interno, sino también es una manera de introducirse en el mercado a escala mundial aprovechando su perspectiva económica. Este proceso requiere de una mejora constante del producto que garantice la calidad del mismo. Con este objetivo se cuenta con varias entidades productoras de software, dentro de ellas la UCI.

INTRODUCCIÓN

La UCI, institución surgida al calor de la Batalla de Ideas, en estos momentos cuenta con disímiles Proyectos de Investigación, Desarrollo e Innovación, que son fuentes de ingresos a la economía del país. Se dedica al desarrollo de productos informáticos, ya que es necesario preservar activamente la calidad de estos. Este centro forma parte de la industria cubana de software. Sus estudiantes tienen la peculiaridad de vincular en sus actividades el estudio con los proyectos, ayudando de esta forma a dar respuesta a las solicitudes de aplicaciones que llegan continuamente a la universidad. Las cuales pueden ser de origen nacional e internacional, por lo que se han desarrollado una serie de actividades de mucho interés para la universidad con el objetivo de garantizar que los productos que se realizan cumplan con la calidad requerida.

En los centros de desarrollo de la UCI los proyectos se realizan a partir de convenios firmados con diferentes empresas, tanto en el ámbito nacional como internacional. El objetivo que estos presentan es obtener como resultado final productos de software con una mayor calidad, a través de la transparencia y control total de la eficiencia utilizada por los desarrolladores. Esto tiene como ventaja que un cliente satisfecho solicitará más servicios.

La seguridad, como parte necesaria del desarrollo de software, es un tema que hasta hace muy poco tiempo no era de sumo interés para los desarrolladores de sistemas. Actualmente se presentan evidencias relacionadas con los medios informáticos donde aparecen graves problemas detectados, como el robo de información; toman el IP de la máquina y pueden enviar virus, configurar o borrar algún programa importante, ya que constantemente se está enviando y recibiendo información de gran valor.

Si algún individuo extraño a la *Personal Computer*[1] (PC) obtiene de alguna forma o manipula información que puede ser de gran valor, lo mínimo que podría pasar sería pérdida en términos económicos. Para darle seguimiento y control a la seguridad que se diseña en los diferentes sistemas se pueden encontrar las Pruebas de Seguridad, las cuales determinan las vulnerabilidades potenciales en su estructura de seguridad de la información.

Los clientes exigen en sus contratos el desarrollo y liberación de un software que garantice la confidencialidad, integridad, autenticidad, confiabilidad, corrección, fiabilidad, mantenibilidad y seguridad de la información sensible almacenada en formato electrónico. Por esta razón a partir del año 2005 es creado el Laboratorio Industrial de Pruebas de Software (LIPS) en la UCI con el fin de certificar con un sello de calidad a los productos que estén listos para ser comercializados internacionalmente.

[1] *Computadora Personal*

Además se crearon en algunas facultades grupos de calidad con el objetivo de garantizar que exista documentación, lograr mejoras en el proceso de desarrollo y garantizar que se prueben las aplicaciones que se realizan en los proyectos. La eficiencia de un software es una tarea a la que se dedican muchos esfuerzos. Todo proyecto tiene como objetivo desarrollar productos de alta disposición, tratar de ser lo mejor posible y que supere las expectativas de los clientes.

Una de las facultades que tiene en funcionamiento estos grupos de calidad es la Facultad 6 de la universidad, en ella se encuentra el Centro Geoinformática y Señales Digitales (GEYSED), compuesto por dos departamentos: Señales Digitales y Geoinformática. Este último desarrolla sistemas que facilitan la geo-referenciación de objetos sobre mapas para su posterior análisis y consulta de acuerdo a características específicas de los solicitantes.

Por su parte Señales Digitales tiene como misión el procesamiento digital de imágenes y señales, aprovechando las tecnologías de código abierto, a través de la actividad docente – científico – productiva. En el departamento mencionado anteriormente uno de los proyectos que se encuentra es la Plataforma de Transmisión Abierta de Radio y Televisión (PTARTV), este se dedica a integrar y organizar todos los procesos que se realizan en la Dirección de Televisión Universitaria (DTU) y brinda soluciones a los problemas existentes, mediante funcionalidades que faciliten la programación, la gestión de medias, la transmisión, la transferencia y el entorno web de la aplicación. Esta plataforma es capaz de difundir las informaciones en un sitio o un televisor, haciendo uso de las nuevas tecnologías, logrando que se pueda acceder al contenido audiovisual para el tratamiento y la gestión de medias.

En PTARTV como en otros proyectos de desarrollo de software del centro de la facultad 6, tradicionalmente han padecido de problemas para garantizar la calidad y seguridad, tanto en el propio proceso de desarrollo como en los productos que entregan. Esta problemática tiene su origen en las habituales desviaciones de plazos, esfuerzo sobre los valores previstos, en la frecuente aparición de fallos durante la implantación y operación de los productos resultantes.

Lo anteriormente planteado da lugar a ciertas dificultades, como es la escasa adherencia a los plazos y esfuerzos previstos, lo que denota o pone de manifiesto la baja acción de la calidad en el proceso de gestión, la falta de calidad de los productos desarrollados: cuanto menor es la seguridad y calidad de los productos, mayor es el número de defectos, consecuentemente, mayor será el número de fallos que aparecerán durante la ejecución del software. Cuanto más se logre evitar estos conflictos más se ganará en calidad del software. Un componente importante del *Software Quality Assurance* (SQA)[2] son

[2] *Aseguramiento de la Calidad del Software.*

las actividades del proceso de Verificación y Validación (V&V) de software que se realizan durante las diferentes fases que componen el ciclo de desarrollo de los sistemas.

Entre las técnicas del proceso de V&V se encuentran las pruebas de software, un conjunto de herramientas, técnicas y métodos que conciben mejorar el desempeño de un programa. Las técnicas para descubrir problemas son numerosas y van desde el descubrimiento por uso del ingenio del personal de prueba hasta novedosas herramientas automatizadas que ayudan a aliviar el peso y el costo de tiempo de esta actividad.

El Plan de Prueba es un documento donde se deja por escrito todo los pasos necesarios para la ejecución de las pruebas que se vayan a realizar. El del proyecto PTARTV está encaminado a verificar los requisitos funcionales del sistema. Las pruebas que se realizan son de Caja Negra que se centra en los requisitos funcionales y las de Caja Blanca que se ajusta a comprobar la interacción interna de los componentes que ayudan a definir conjuntos de casos aplicando ciertos criterios. El plan no cuenta con actividades que permitan realizar otros tipos de pruebas, en el no están definidos los roles que intervienen en las pruebas de seguridad ni las herramientas necesarias.

En el proyecto PTARTV del centro GEYSED la seguridad que esta implementada permite que los usuarios que tienen acceso restringido, puedan acceder a los recursos no autorizados por los administradores. El proyecto cuenta con una base de datos a la cual se le pueden añadir datos o hacer modificaciones a sus tablas, no tiene medidas de seguridad para impedir que personas ajenas puedan acceder a esta base de datos. Aún las contraseñas que se definieron en la aplicación son débiles porque no cumplen con las políticas de seguridad establecidas por el proyecto. Se puede afirmar que no se le da un seguimiento adecuado a la seguridad del proyecto.

INTRODUCCIÓN

Teniendo en consideración que el proyecto PTARTV tiene gran importancia para la DTU, es necesario que el mismo cuente con la calidad y seguridad requerida. A partir de la situación planteada se ha identificado el siguiente **problema a resolver:** ¿Cómo contribuir a mejorar la seguridad del producto PTARTV del Centro GEYSED?

Por lo cual, el **objeto de estudio** lo constituye las Pruebas de Seguridad y el **campo de acción** son los procesos de las Pruebas de Seguridad a la PTARTV.

Se persigue con esta investigación lograr el siguiente **objetivo general**: Definir un procedimiento para realizar Pruebas de Seguridad a la PTARTV**.**

Como **idea a defender** se plantea que con la definición y posterior aplicación de un procedimiento para el desarrollo de Pruebas de Seguridad a la PTARTV, se obtendrá un sistema con mayor seguridad.

Para darle cumplimiento al **objetivo general** se definen diferentes **tareas de investigación** las cuales se relacionan a continuación:

- Caracterizar el estado actual sobre las Pruebas de Seguridad en el mundo.
- Diagnosticar el proceso de pruebas que se le realizan a la PTARTV.
- Caracterizar las diferentes técnicas utilizadas para desarrollar Pruebas de Seguridad.
- Caracterizar las herramientas internacionalmente utilizadas en las Pruebas de Seguridad.
- Identificar y describir los elementos de un procedimiento.
- Definir el procedimiento para el desarrollo de Pruebas de Seguridad a la PTARTV.
- Validar el procedimiento para el desarrollo de Pruebas de Seguridad a la PTARTV.

Posibles resultados que presupone la investigación:

- Procedimiento para realizar Pruebas de Seguridad a la PTARTV.

INTRODUCCIÓN

Entre los **métodos de investigación** que sostienen científicamente el presente trabajo se encuentra dentro del nivel teórico al:

- **Analítico-Sintético:** el mismo favoreció el estudio de los fundamentos teóricos del problema que se aborda, así como los resultados obtenidos en la aplicación de la investigación, luego se hace una síntesis de los elementos más significativos del proceso de prueba.
- **Histórico–Lógico:** este método permite ir observando los posibles resultados durante todo el proceso de desarrollo del software, además facilita extraer la información más importante del objeto de estudio de la investigación.

La presente investigación está compuesta por 3 capítulos, a continuación se brinda una breve descripción sobre lo que aborda cada uno en específico.

Capítulo 1: "Fundamentación Teórica". En este capítulo están sentadas las bases de la investigación, brindando una serie de conceptos y caracterizando los diferentes tipos de pruebas según la metodología que se lleva a cabo en el proyecto PTARTV.

Capítulo 2: "Técnicas y procedimientos a aplicar". En este capítulo se seleccionan las pruebas que se le van a aplicar al software, para luego definir los procedimientos que se llevarán a cabo en la ejecución de dichas pruebas al Módulo de Programación de la PTARTV.

Capítulo 3: "Diseño, ejecución y evaluación de las pruebas". Se diseñan y aplican los casos de prueba al Módulo de Programación de la PTARTV y finalmente con los resultados obtenidos durante la aplicación de los casos de prueba diseñados se emite una evaluación final.

Capítulo 1: Fundamentación Teórica

1. Introducción

Este capítulo muestra los argumentos fundamentales que se abordan a lo largo de la investigación. Se mencionan conceptos relacionados con la calidad de software, los tipos de pruebas así como, otros aspectos relacionados con estos temas. Tiene como objetivo fundamental profundizar en diferentes contenidos que se utilizarán como soporte teórico para la investigación.

1.1. En el presente epígrafe se van a definir los principales conceptos acerca de la calidad del software.

1.1.1. Calidad.

Calidad es una definición que se hace necesario tenerla presente en cualquier actividad de la vida, no es algo que pueda ser determinado fácilmente, su significado depende del marco en que se analice. A partir esta enunciación se lleva a cabo en la realización de los proyectos de software. Muchos han sido los autores que han aportado sus ideas sobre la calidad y cómo debe de ser aplicada.

La palabra calidad se usa cada vez más, con mayor frecuencia en las corporaciones, ya sea en los sectores de alimentos, industria o servicios y especialmente en el sector de Tecnología Informática (TI). El cliente debe interpretar de modo amplio como beneficiario del producto, la entrega de algo material o inmaterial (servicio). Tiene como meta definir los términos principales como: comprender los objetivos y los métodos de la implementación de un procedimiento de la misma.

La calidad se puede definir como la capacidad de lograr objetivos de operación buscados. La norma ISO 8402-94 la define como: El conjunto de características de una entidad que le otorgan la capacidad de satisfacer necesidades expresas e implícitas (1).

La ISO 9000 no deja de ser un estándar de calidad, es decir una norma aplicada por igual a los que la poseen y pasan una certificación por una entidad registrada. En concreto, las normas ISO 9000 son referentes a los Sistemas de Calidad y permiten certificar que la empresa que conserva el certificado y lo implementa en toda su estructura, es decir, que se orienta de cara a satisfacer las expectativas de sus clientes (1).

Para Deming[3] (2) calidad significa ofrecer productos, servicios confiables y satisfactorios a bajo costo. En tanto para Juran, lo importante es que sea adecuado para su uso, para Crosby[4] (2), es el

[3] *William Edwards Deming: Estadístico estadounidense, profesor universitario, autor de textos, consultor y difusor del concepto de calidad total*

[4] *Philip Crosby: Es uno de los pensadores sobre calidad más destacados de los Estados Unidos.*

cumplimiento de los requerimientos de cada compañía, o dicho de otra forma, dar cumplimiento a las especificaciones. Partiendo de una perspectiva diferente se puede llegar a la conclusión que es un compromiso ético con la excelencia, porque sólo una empresa que ha definido en sus valores supremos el generar beneficios y favores, estará realmente comprometida en su consecución. Si no se genera interna, puede ofrecerse que sea apropiada a los agentes externos. Por ello se construye y genera en cada actividad, tarea y proceso de la compañía.

Calidad es un concepto manejado con bastante frecuencia en la actualidad, pero a su vez, su significado es percibido de distintas maneras. Al hablar de bienes y/o servicios de calidad, la gente se refiere normalmente a bienes de lujo o excelentes con precios elevados. Su significado sigue siendo ambiguo y muchas veces su uso depende de lo que cada uno entiende por ella.

1.1.2. Calidad de software.

La calidad del producto debe ser medida a lo largo de todo el período de desarrollo. Sin embargo el software puede medirse después de elaborado el producto, pero esto puede resultar muy costoso si se detectan problemas derivados de imperfecciones en el diseño, por lo que es imprescindible tener en cuenta tanto la obtención del control durante todas las etapas del ciclo de vida del proyecto. A pesar de no tener una definición estándar, cada autor da su punto de vista al respecto. Permitiendo así que cada quien adecue sus conceptos a su favor.

Se toma como definición de calidad de software: la concordancia con los requerimientos funcionales y de rendimiento explícitamente establecidos, con los estándares de desarrollo evidentemente documentados y con las características implícitas que se espera de todo software desarrollado profesionalmente (3).

Se puede decir que el software tiene calidad si cumple o excede las expectativas del usuario en cuanto a (4):

- Funcionalidad (que sirva a un propósito).
- Ejecución (que sea práctico).
- Confiabilidad (que haga lo que debe).
- Disponibilidad (que funcione bajo cualquier circunstancia).

A modo de resumen se dice que la calidad de software se refiere a: Los factores de un producto que contribuyen a la satisfacción completa y total de las necesidades de un usuario u organización. Debido a la importancia que tiene esta se consigue explicar que es el desarrollo de software basado en estándares con la funcionalidad y rendimiento total que satisfacen los requerimientos del cliente.

Para resumir de alguna manera la amplitud de este concepto, se hace referencia que la calidad de software ha sido usada desde un simple argumento de venta, hasta verdaderos estudios formales y usos de métricas para el desarrollo. Extrañamente dentro de la ingeniería, esta es muy complicada de definir y de enmarcar en un simple concepto teórico, permiten describirla en los elementos que le importan específicamente al diseñador de software.

Uno de los desafíos que se impone en la producción de un software en la esfera de la computación es la calidad del software. Todo proyecto tiene como objetivo producir software con la mejor calidad posible para que supere las expectativas de los usuarios.

Una aplicación informática con calidad es aquella que tiene un alto grado de confiabilidad, aceptación y cumple con los requerimientos explícitos e implícitos. A continuación se verán características que le permite tener a un software una mayor eficacia (5).

Mantenibilidad: El software debe ser diseñado de tal manera, que permita ajustarlo a los cambios en los requerimientos del cliente. Esta característica es crucial, debido al inevitable cambio del contexto en el que se desempeña un software.

Confiabilidad: Incluye varias características además de la confiabilidad, como la seguridad, control de fallos.

Eficiencia: Tiene que ver con el uso eficiente de los recursos que necesita un sistema para su funcionamiento.

Usabilidad: El software debiera ser utilizado sin un gran esfuerzo por los usuarios para los que fue diseñado, documentado.

Corrección: El grado en que una aplicación satisface sus especificaciones y consigue los objetivos encomendados por el cliente.

Fiabilidad: El grado que se puede esperar de una aplicación lleve a cabo las operaciones especificadas y con la precisión requerida. Se refiere a la capacidad del sistema de funcionar permanentemente sin fallos y de mantener la integridad de los datos.

Portabilidad: La capacidad del producto de software para ser transferido de un entorno a otro. El entorno puede incluir la organización, hardware o software.

Integridad: El grado con que puede controlarse el acceso al software o a los datos a personal no autorizado y se mide la capacidad del sistema a resistir ataques contra su seguridad.

Facilidad de uso: El esfuerzo requerido para aprender el manejo de una aplicación, trabajar con ella, introducir datos y conseguir resultados.

Seguridad: Consiste en asegurar que los recursos del sistema de información (material informático o programas) de una organización sean utilizados de la manera que se decidió y que el acceso a la información allí contenida, así como su modificación sólo sea posible a las personas que se encuentren acreditadas y dentro de los límites de su autorización.

1.1.3. Control de la calidad

Para lograr el control de la calidad es necesario realizar un seguimiento constante al proceso de desarrollo del software, el cual consiste en realizar una observación sobre el cumplimiento de las tareas que pueden ofrecer un valor al producto que se está desarrollando, o sea, una vigilancia permanente a todo el proceso y el ciclo de vida del mismo. En cada uno de los hitos debe ejecutarse por personal especialmente dedicado a esta función, debe escogerse los técnicos de mayor experiencia.

Son las actividades de carácter operativo, utilizadas para satisfacer los requisitos relativos a la calidad, centradas en dos objetivos fundamentales (6):

- Mantener bajo control un proceso.
- Eliminar las causas de los defectos en las diferentes fases del ciclo de vida.

1.1.4. Sistema de calidad

Estructura organizativa, procedimientos, procesos y recursos necesarios para implantar la gestión de calidad (6)

- El sistema se debe adecuar a los objetivos de la empresa.
- La dirección de la empresa es la responsable de fijar la política de calidad y las decisiones relativas a iniciar, desarrollar, implantar y actualizar el sistema.
- Un sistema de calidad consta de varias partes (6).
    - Manual de calidad. Es el documento principal para establecer e implantar un sistema de calidad. Puede haber manuales a nivel de empresa, departamento, producto, específicos (compras, proyectos).

Parte física

- Locales
- Herramientas
- Ordenadores.

Aspectos humanos

- ❖ Formación de personal.
- ❖ Creación y coordinación de equipos de trabajo.

- ➢ Normativas:
  - ❖ ISO.
  - ❖ ISO 9000: Gestión y SQA (conceptos y directrices generales) (6).

Con el transcurso de los años ha aumentado el desarrollo tecnológico y económico, por lo que es más favorable evitar los fallos de la calidad, que corregirlos o lamentarlos. Los sistemas de calidad se basan en una serie de inspecciones, revisiones y pruebas utilizadas a lo largo del proceso del software, para asegurar que cada producto cumple con los requisitos que le han sido asignados.

1.2. Concepto de procedimientos

Procedimiento: es un método de ejecución o pasos a seguir, en forma secuenciada y sistemática, en la consecución de un fin. Un conjunto de procedimientos con un mismo fin, se denomina sistema. Se puede decir que las instrucciones que otorga un programa para ejecutar una tarea determinada. Se hace referencia a este concepto ya que el Plan de Prueba descrito es parte del mismo y es uno de los parámetros que el utiliza para su funcionamiento (7).

A través de un procedimiento se puede especificar cómo realizar uno o varios diseños de casos de prueba o partes de estos. Este puede ser una instrucción para un individuo sobre cómo va a realizar un caso de prueba manualmente o puede ser una especificación de cómo interactuar manualmente con una herramienta de automatización de las pruebas para crear componentes ejecutables de pruebas.

Un procedimiento de pruebas cuenta con 5 fases muy importante una de ella es el Plan de Pruebas dentro de ella se especifica los roles que van a intervenir, el cronograma de las pruebas, las observaciones que se realizan mediante la ejecución de las pruebas, cómo segunda fase se encuentra el diseño de casos de pruebas, la tercera fase es: la ejecución de las pruebas, la cuarta es: la documentación de los errores y la última depuración de los errores que es aquí donde se reúnen el equipo de desarrollo con el equipo revisor para discutir las no conformidades encontradas. Este procedimiento puede variar en cuanto al producto que se le aplique y las características del proyecto donde se vaya a utilizar este proceso.

1.3. Pruebas de calidad de software

Prueba del software es el proceso usado para determinar la calidad. Es una investigación técnica empírica conducida para proveer de poseedores de apuestas la información sobre el producto o el servicio, con respecto al contexto en el cual se piensa para funcionar. Esto incluye, pero no se limita a ejecutar un programa. La prueba puede establecer totalmente la corrección del software arbitrario; está facilitada a crítica o comparación al estado y el comportamiento de la ganancia contra una especificación. Un punto importante es el software que debe ser distinguido de la disciplina separada de la garantía del SQA, que abarca todas las áreas del negocio. (8).

Las pruebas, vistas desde el marco de un proceso de desarrollo de software, son los diferentes procesos que se deben realizar durante un desarrollo, con el objetivo de asegurar completitud, correctitud y calidad de gran importancia, estas no se deben dejar para el final de la etapa de construcción del software. Las pruebas se deben empezar a realizar desde la misma etapa de análisis y de requerimientos, ya que desde un principio se puede incurrir en malas interpretaciones de las reglas del negocio, lo que finalmente tendrá como consecuencia incongruencia entre lo que el cliente quiere y lo que se ha desarrollado.

- Las pruebas de software, en inglés *testing* o *betatesting* son los procesos que permiten verificar y revelar la calidad de un producto de software. Son utilizadas para identificar posibles fallos de implementación, calidad o usabilidad de un programa de ordenador o videojuego.
- Proceso realizado concurrentemente a través de las diferentes etapas de desarrollo de software que utiliza y mantiene el *testware*[5] y cuyo objetivo es apoyar la disminución del riesgo de aparición de fallas y faltas en operación (8).

La fase de pruebas es una de las más costosas del ciclo de vida del software. En sentido estricto, deben realizarse de todos los artefactos generados durante la construcción de un producto, lo que incluye especificaciones de requisitos, casos de uso, diagramas de diversos tipos, el código fuente y el resto de productos que forman parte de la aplicación (9).

Estas fases de pruebas consisten en una actividad de control de la calidad, además son un elemento crítico para el SQA, pues representan una revisión final de las especificaciones, el diseño y la codificación. Las pruebas son las encargadas de verificar la interacción y la integración adecuada de los componentes, comprueba que todos los requisitos se han implementado correctamente, identificar y

[5] *Software creado con el propósito de probar otro software.*

asegurar que los defectos encontrados se han corregido antes de entregar el software al cliente, que sistemáticamente saquen a la luz diferentes clases de errores, haciéndolo con la menor cantidad de tiempo y esfuerzo.

El único instrumento adecuado para determinar el *status* de la calidad de un producto software es el proceso de pruebas. En este se ven como son dirigidas a componentes del software o al sistema en su totalidad, con el objetivo de medir el grado en que cumple con los requerimientos. En ellas se usan casos de prueba, especificados de forma estructurada mediante sus técnicas. El proceso, sus objetivos, los métodos y técnicas usadas se describen en el plan realizadas por ella (10).

1.3.1. Estrategia de pruebas

En cierta manera la estrategia de pruebas vela porque el producto de software que se está construyendo o modificando, reúna los requerimientos de lógica del negocio que el cliente ha pedido realizar mediante el debido contrato de desarrollo. El objetivo fundamental es detectar los errores que presenta las operaciones que se está construyendo. Se deben realizar en cada iteración[6] para de esta forma poder eliminar lo antes posible los defectos encontrados.

Presenta las líneas de guías del equipo de un proyecto de desarrollo de software. Estas son escritas por el diseñador, en un documento, que será entregado formalmente al director del proyecto, para su posterior revisión y aprobación. Se debe tener claridad en el transcurso de escribir las no conformidades encontradas, debe ser visto y analizado por el dirigente del proyecto, éste corresponde a ser examinado como un artefacto especialmente creado para reunir las ideas más representativas del proceso que se llevará a cabo.

El grupo de calidad de la facultad 6 presenta una estrategia de prueba que tiene como objetivo: definir un procedimiento para homogenizar las pruebas en la facultad 6 y como alcance: define cómo se efectuará el esfuerzo contra uno o más aspectos de los sistemas que se desarrollan en los diferentes Proyectos de Investigación, Desarrollo e Innovación.

1.3.2. Estrategias de pruebas definidas.

Para realizar pruebas efectivas un equipo de software debe efectuar revisiones técnicas formales y efectivas (11)**.**

- La prueba comienza al nivel de componentes y trabaja hacia fuera, hacia la integración de los componentes.

[6] *Es un mini proyecto que tiene como resultado una versión interna de cada uno de los artefactos que pueden ser generados en un proceso de desarrollo de software.*

- Diferentes técnicas de prueba son apropiadas en diferentes momentos.
- La prueba la dirige el desarrollador del software y un grupo independiente de pruebas.
- La prueba y la depuración son actividades diferentes, pero la segunda se debe incluir en la estrategia de pruebas.
- La estrategia debe incluir pruebas de bajo nivel y de alto nivel.
- Las pruebas son el último bastión para la evaluación de la calidad y el descubrimiento de errores.
- Las pruebas no garantizan la calidad del producto.
- Si el producto no tiene la calidad requerida, las pruebas demostrarán únicamente la falta de calidad del producto evaluado.

1.3.3. Roles de pruebas de software.

Las metodologías son las que definen los tipos de roles con que se vayan a trabajar. La presente investigación tomará como punto de partida la metodología RUP, en esta se definen varios roles:

- Diseñador de pruebas.
- Gestor de prueba.
- Analista de pruebas.
- Verificador.

Estos roles pueden ser desempeñados tanto por profesores como estudiantes que estén a cargo de las pruebas de software dentro del proyecto.

Diseñador de pruebas: Este rol dirige la identificación de las técnicas, herramientas y directrices apropiadas para implementar las pruebas necesarias y para proporcionar orientación al esfuerzo de prueba sobre los requisitos de recursos correspondientes.

Gestor de prueba: Este rol dirige el apoyo de calidad y prueba, la planificación, gestión de recursos y la resolución de cuestiones que impiden el esfuerzo de prueba.

Analista de pruebas: Este rol identifica y define las pruebas necesarias, supervisa el proceso de prueba y los resultados de cada ciclo de prueba y evalúa la calidad global. El rol también representa a los interesados que no tienen una representación directa o regular en el proyecto.

Verificador: Este rol realiza pruebas y registra los resultados de las pruebas.

1.3.4. Tipos de pruebas de software

En la comúnmente llamada fase de pruebas, se encuentran diferentes niveles y tipos prueba (12).

- **Pruebas de Integración:** Se comprueba la compatibilidad y funcionalidad de los interfaces entre las distintas partes que componen un sistema, estas pueden ser módulos, aplicaciones individuales o aplicaciones cliente/servidor. Este tipo de pruebas es especialmente relevante en aplicaciones distribuidas.
- **Pruebas de Validación:** Son las pruebas realizadas sobre un software completamente integrado para evaluar el cumplimiento con los requisitos especificados.
- **Pruebas de Aceptación:** Son las que hará el cliente, se determina que el sistema cumple con lo deseado y se obtiene la conformidad del cliente.
- **Pruebas de Sistema:** El software ya validado se integra con el resto del sistema, en los cuales existen varios tipos de pruebas pertenecientes al sistema que son:
  - ✓ **Rendimiento:** determinan los tiempos de respuesta, el espacio que ocupa el módulo en disco o en memoria, el flujo de datos que genera a través de un canal de comunicaciones.
  - ✓ **Resistencia:** determinan hasta donde puede soportar el programa determinadas condiciones extremas.
  - ✓ **Robustez:** determinan la capacidad del programa para soportar entradas incorrectas.
  - ✓ **Seguridad:** se determinan los niveles de permiso de usuarios, las operaciones de acceso al sistema y acceso a datos.
  - ✓ **Usabilidad:** se determina la calidad de la experiencia de un usuario en la forma en la que éste interactúa con el sistema, se considera la facilidad de uso y el grado de satisfacción del usuario.
  - ✓ **Instalación:** se determinan las operaciones de arranque y actualización del software.

Esta correspondencia entre fases del desarrollo y tipos de pruebas produce el llamado “modelo en V”, del que se muestra un ejemplo en la Figura 1 (12).

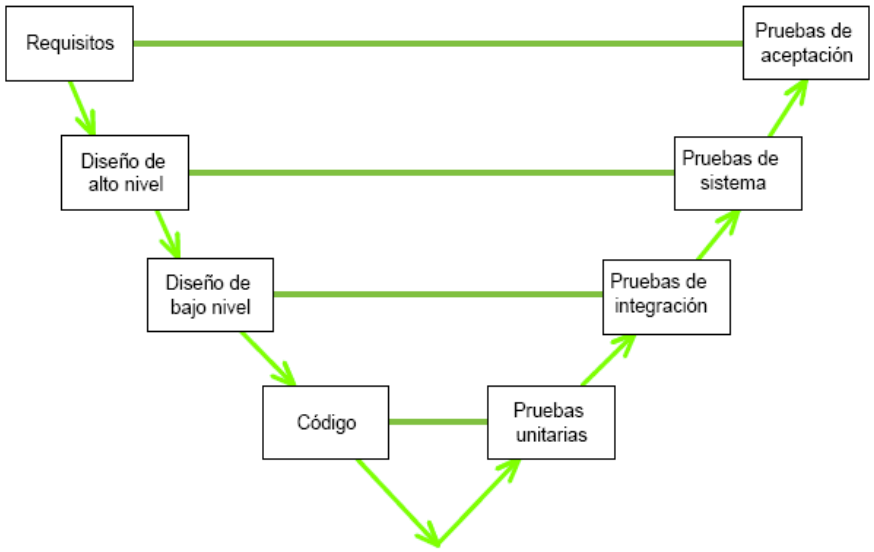

Figura 1: Modelos de Pruebas

Asociado a los tipos de pruebas existen técnicas de pruebas que ayudan a definir conjuntos de casos de pruebas aplicando ciertos criterios, como son (12):

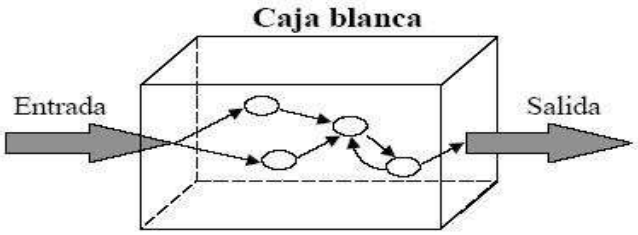

Figura 2: Pruebas de caja blanca.

- **Pruebas de caja blanca:** Se centra en comprobar la interacción interna de los componentes del sistema.

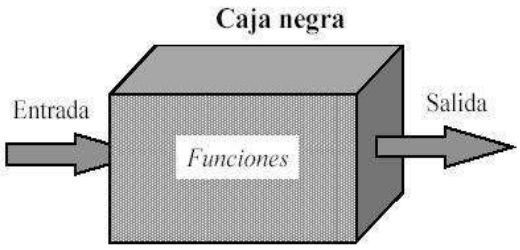

Figura 3: Pruebas de caja negra.

- **Pruebas de caja negra:** Se centran en los requisitos funcionales del software. O sea, la prueba de de caja negra permite al ingeniero del software obtener un conjunto de condiciones de entrada que ejerciten completamente todos los requisitos.

Cada uno de estos temas de prueba junto con otros como las estrategias de prueba, el plan de prueba, casos de prueba entre otros pudieran ser abordados con más profundidad.

1.3.5. Plan de Pruebas software

Un Plan de Pruebas está constituido por un conjunto de pruebas. Cada una de ellas debe (13):

- Dejar claro qué tipo de propiedades se quieren probar (corrección, robustez, fiabilidad, amigabilidad) y cómo se mide el resultado.
- Especificar en qué consiste la prueba (hasta el último detalle de cómo se ejecuta).
- Definir cuál es el resultado que se espera (identificación, tolerancia).

Las pruebas carecen de utilidad, tanto si no se sabe exactamente lo que se quiere probar, o si no está claro cómo se prueba.

Estas mismas ideas se suelen agrupar diciendo que un caso de prueba consta de 3 bloques de información:

- El propósito de la prueba.
- Los pasos de ejecución de la prueba.
- El resultado que se espera.

1.4. Pruebas de Seguridad.

Las Pruebas de Seguridad consisten en asegurar que los recursos del sistema de información (material informático o programas) de una organización sean utilizados de la manera que se decidió y que el acceso a la información allí contenida, así como su modificación sólo sea posible a las personas que se encuentren acreditadas y dentro de los límites de su autorización.

En la actualidad dicho concepto debe ser incorporado de manera obligatoria en el proceso de desarrollo de un software, desde las fases iniciales de su diseño hasta su puesta en funcionamiento, la seguridad es una aplicación que se realiza a un producto desde las fases iniciales de su diseño hasta que el mismo sea terminado. El desarrollador no sólo debe concentrarse únicamente en los usuarios y sus requerimientos, sino también en los eventos que puedan interferir con la integridad del software y la información que este maneja.

A nivel mundial, las Pruebas de Seguridad se llevan a cabo de una manera mucho más metódica. Esto es totalmente necesario, pues la existencia de un diseño seguro, avanzado y su enfoque de desarrollo, están reduciendo el número de defectos escondidos, por lo que la detección de vulnerabilidades durante las pruebas es mucho más difícil. La seguridad del software es demasiado importante como para dejarla en manos de un pequeño grupo de intrusos o hackers. Esta debe ser fácil de enseñar y ordenar de modo que pueda aplicarse en una amplia variedad de circunstancias (14).

Las pruebas giran en torno a la variación, es decir, detectar los aspectos del software y su entorno que pueden haber variado y ver cómo responde al mismo. El objetivo consiste en garantizar que el funcione de una manera confiable y segura en escenarios de producción. Por lo que la planeación más importante que debe llevar a cabo un evaluador es conocer los aspectos que pueden variar y de qué formas necesita configurarse para las pruebas (14).

Actualmente las organizaciones encaran el doble riesgo de ataques externos a sus activos digitales y el abuso interno de los privilegios de acceso. A pesar de que muchas de sus conexiones a Internet están protegidas por Firewalls, muchas no tienen sistemas de detección de intrusos o políticas de seguridad adecuadas. Debido a esto, los hackers[7] siguen causando devastación a redes de computadoras, las consecuentes pérdidas económicas, de imagen y credibilidad de sus víctimas.

Una Prueba de Seguridad de Aplicación es independientemente usada para auditar aplicaciones web y se simulan posibles acciones de usuarios no autorizados, internos y externos. Con ellas es posible descubrir si este es seguro antes de que los hackers descarguen datos sensitivos, comentan un crimen o pongan en riesgo su negocio.

Desde el punto de vista de la seguridad, el entorno, la entrada de usuario son los lugares principales donde dicha variación puede revelar problemas de seguridad. El entorno consiste en los archivos,

[7] **Hackers:** Es toda aquella persona con elevados conocimientos informáticos independientemente de la finalidad con que los use.

aplicaciones, recursos del sistema y otros locales de red que la aplicación use. Cualquiera de ellos podría ser el punto de un ataque. La entrada de usuario la constituyen los datos que se originan con entidades externas (normalmente no confiables) que el software analiza y los usa. Los datos y lógica internos son las variables y rutas almacenadas internamente que tienen cualquier número de enumeraciones potenciales (14).

1.4.1. Definición de las Pruebas de Seguridad

Un equipo de expertos de confianza intenta practicar una abertura en sus perímetros en línea y físicos de la compañía. Puesto que han integrado cada vez más pesadamente la tecnología de la información en sus negocios, hay una amenaza creciente del ataque del intento de la gente contra el robo de esa información (15).

El propósito de la Prueba de Seguridad: es una manera ética de determinar las vulnerabilidades potenciales en su estructura de la seguridad de información. Determina el modo en que usted se pueda defender mejor contra todas las formas de ataques. Estas realizan los mismos métodos que los piratas informáticos expertos. Por esta razón, ellas necesitan ser realizadas por los usuarios con el conocimiento y las habilidades necesarias en este tipo de prueba.

1.4.2. Resultados y necesidades de usar Pruebas de Seguridad

Después de una Prueba de Seguridad completa, los especialistas preparan un informe con vulnerabilidades potenciales en su sistema entero. El documento se escribe en varias formas, de modo que los empleados técnicos del nivel del personal puedan entender y apreciar las amenazas encontradas en el negocio sobre una base diaria. Los expertos revisarán las respuestas del personal que es: la ingeniería social y su huella electrónica.

Posteriormente de un estudio de las Pruebas de Seguridad es importante destacar que se pueden utilizar para comprobar la eficacia, validez de las políticas de seguridad y los procedimientos existentes. Al asumir el papel de un hacker y luego examinar la red, los administradores pueden identificar los puntos fuertes de seguridad y las posibles deficiencias que se pueden encontrar. Estas pruebas permiten ampliar los conocimientos de los administradores acerca del sistema y redes. Descubre algunas herramientas disponibles que pueden ser utilizados para este tipo de prueba. Es valioso resaltar que la información obtenida que presenta es para solicitar el apoyo financiero de los sistemas de detección de intrusos, firewall y soluciones.

Según lo indicado previamente, los piratas informáticos pueden ser aficionados o profesionales, su objetivo puede ser robar datos sensibles o el dinero. La información necesita ser protegida profesionalmente contra ataque. Tener un cortafuego y regularmente el cambio de contraseñas no es

bastante para evitar que estas personas que son expertos en hacer daño en la rama de la informática puenteen sus medidas de seguridad en la red.

Las debilidades pueden existir en el software que usted utiliza y también dentro de sus protocolos y procedimientos de seguridad. Sin una Prueba de Seguridad completa los piratas informáticos podrían acceder y sustraer toda la información confidencial que deseen y realizar daños en la PC. Algunas compañías podrían incluso hacer frente a la demanda legal o aún al encierro si sus sistemas de seguridad de la información no se conforman con los modelos legales.

1.5. Conclusiones parciales

Se puede afirmar que para que un producto software cumpla con las especificaciones para las que fue creado debe ser verificado por un Plan de Pruebas elaborado por el Administrador de Prueba que es responsable que las mismas se ejecuten correctamente. Para validar el cumplimiento con las especificaciones antes mencionadas es necesario conocer los diferentes tipos de pruebas y técnicas para asegurar la calidad del software. En este capítulo se han mencionado los factores a tener en cuenta para medir la calidad de un sistema, también se han mencionado los diferentes tipos de pruebas conocidas y como deben ser usadas para obtener un resultado óptimo y favorable. Por esta razón los esfuerzos que se realizan para conseguirla nunca son suficientes y el equipo no puede quedarse conforme, siempre hay que hacer más con mayor eficacia.

Capítulo 2: Herramientas y Procedimientos

2. Introducción

En el presente capítulo se realiza estudio de los procedimientos y las herramientas a utilizar en el capítulo 3. Se hace mención a los tipos de Pruebas de Seguridad existentes, a las categorías de las mismas, a las técnicas y fases existentes conocidos actualmente. Tiene como objetivo fundamental profundizar en diferentes contenidos que se utilizarán como base para llegar a una solución final.

2.1 Prueba de Seguridad

Se pueden observar algunas Pruebas de Seguridad existentes las cuales son (15):

Prueba de Intrusión de Aplicación Web: es un método de evaluación de la seguridad de un sistema de ordenadores o una red mediante la simulación de un ataque. Está enfocada solamente a evaluar la seguridad de esta. El proceso conlleva un análisis activo de la aplicación en busca de cualquier debilidad, fallos técnicos o vulnerabilidades. Cualquier incidencia que sea encontrada será presentada al propietario del sistema, junto con una evaluación de su impacto y a menudo con una propuesta para su mitigación o una solución técnica.

Pruebas de Gestión de Configuración de la Infraestructura: A menudo, los análisis de la infraestructura y topología de la arquitectura pueden revelar una gran cantidad de datos sobre la aplicación web. Se pueden obtener así informaciones como el código fuente, métodos HTTP, funcionalidad administrativa, sistemas de autenticación y configuraciones. No puede ser tan completo como la información recopilada mediante un análisis más amplio de ella.

Pruebas del Receptor de Escucha de la BBDD: Durante la configuración de un servidor de base de datos, muchos administradores no toman en consideración adecuadamente la seguridad del componente receptor la base de datos. Este puede revelar información sensible, así como ajustes de configuración o instancias de la base de datos en ejecución, si está configurado de forma insegura y se comprueba con técnicas manuales o automatizadas. La información revelada a menudo será de utilidad, sirviendo como una puerta de entrada a otras pruebas derivadas, de mayor impacto.

Pruebas de Gestión del Caché de Navegación y de Salida de Sesión: El objetivo de esta prueba es comprobar que la función de cierre de sesión está correctamente implementada y que no es posible reutilizar una sesión después del cierre. También se comprueba que la aplicación automáticamente cierra la sesión de un usuario cuando ha estado inactivo durante un intervalo de tiempo y que ningún dato sensible permanece en el caché del navegador.

Pruebas de Gestión de Sesiones: En el núcleo de toda aplicación web se encuentra el sistema en que la aplicación mantiene los estados y por lo tanto controla la interacción del usuario con el sitio. La gestión de sesiones cubre ampliamente todos los controles que se realizan sobre el usuario, desde la autenticación hasta la salida de la aplicación, lo que significa que los servidores web responden a las peticiones de clientes sin enlazarlas entre sí. Incluso la lógica de una simple aplicación requiere que las múltiples peticiones de un usuario sean asociadas. Para ello se necesitan soluciones externas disponibles en el mercado y del servidor web.

Pruebas de Cifrado y Vulnerabilidades por Reutilización de Testigos de Sesión: La protección frente al acceso a la información se realiza habitualmente mediante el cifrado SSL, pero puede incorporar otro sistema de túnel o cifrado. Debe destacarse que el cifrado de los identificadores de sesión deberá de considerarse por separado en clave, así que el identificador de sesión estará protegido en sí mismo y no los datos representados por él. La principal sesión puede ser presentado por un atacante para obtener acceso a la aplicación, entonces deberá de ser protegido en el tránsito para reducir este riesgo. Por lo tanto garantizarse que el cifrado en ambos por defecto, esté forzado para cualquier petición o respuesta donde se transmita el identificador, independientemente del mecanismo.

Pruebas de Firma de Aplicaciones: La obtención de la firma digital de un servidor web es una tarea esencial para la persona que realiza una Prueba de Intrusión. Saber el tipo y versión del servidor en ejecución le permite determinar vulnerabilidades conocidas y los *exploits* apropiados a usar durante la realización de este tipo de pruebas. Actualmente existen multitud de vendedores y de versiones de servidores web en el mercado. Saber el tipo exacto de servidor que estás comprobando ayuda considerablemente en el proceso de análisis y puede cambiar el curso de las pruebas. Esta información puede inferirse mediante el envío de órdenes específicos al servidor web y posteriormente analizar su respuesta, ya que cada versión del software de servidor web puede responder de forma diferente a un mismo orden.

Prueba de Fuerza Bruta: Consiste en enumerar sistemáticamente todas las posibilidades candidatas como solución a un problema y comprobar para cada una de ellas si satisface el problema enunciado. En las pruebas de aplicaciones web, el problema al que se va a enfrentar la mayor parte de las veces está relacionado con la necesidad de disponer de una cuenta de usuario válida para acceder a la parte interna de la aplicación. Por tanto, se comprueba en diferentes tipos de sistema de autenticación y la efectividad de diferentes ataques de fuerza bruta.

Inyección SQL: Se habla de pruebas de Inyección SQL cuando se intenta inyectar una determinada consulta SQL directamente en la base de datos, sin que la aplicación haga una validación adecuada de los datos. El objetivo es manipular los datos, un recurso vital para todas las empresas. Una inyección SQL explota el siguiente patrón: Entrada -> Consulta SQL == Inyección SQL.

Pruebas de Escalación de Privilegios: Esta sección describe el problema de escalada de privilegios de una etapa a otra. Durante esta fase, el auditor debe verificar que no es posible para un usuario modificar sus privilegios/roles dentro de la aplicación de manera que le permita realizar un ataque de escalada de privilegios. Esta ocurre cuando un usuario logra acceder a más recursos o funcionalidades de las que le fueron asignadas generalmente y tales cambios deberían haber sido prevenidos por la aplicación. Esto es usualmente causado por una falla. El resultado es que el estudio, realiza acciones con más privilegios de los cuales fueron intencionados por el desarrollador o administrador del sistema.

Después de un estudio realizado profundamente de las Pruebas de Seguridad, se ha determinado que las pruebas que se van a utilizar en el proyecto de PTARTV del Centro GEYSED son aquellas que permiten brindar una mayor seguridad y calidad al proyecto, las cuales son más fáciles a la hora de ejecutarlas en las aplicaciones web del proyecto y las que permiten detectar mayor cantidad de errores en un sistema de seguridad. Se eligieron las Pruebas de Firma Aplicaciones, Fuerza Bruta, Escalación de Privilegios e Inyección SQL, ya que las herramientas con que cuentan son libres y utilizan códigos abiertos las cuales traen un ahorro económico a la universidad.

2.2 Categorías de las Pruebas de Seguridad

Recopilación de información: Comprobar si la aplicación brinda datos sensibles que puedan ser utilizados por cualquier atacante.

Es esencial al hacer investigación saber cómo localizar los trabajos previos relativos al área de investigación de su interés, para eso debe conocer:

- Las fuentes de información que contienen los trabajos anteriores o información sobre ellos.
- Los organismos que generan, recopilan u organizan ese tipo de información.
- La forma en que se puede tener acceso a esa información.
- Los procedimientos correspondientes para obtenerla, tanto en el país de origen como en el extranjero; el tiempo que tardaría en tenerla en sus manos y el costo aproximado de los servicios más inmediatos para obtener la información.

Técnicas de investigación documental: Centran su principal función en todos aquellos procedimientos que conllevan al uso óptimo y racional de los recursos documentales disponibles en las fuentes de información.

Se presentan 3 técnicas de recopilar información

- **Método de análisis en grupos:** Las siguientes clases de proyectos de sistemas son probablemente los que se pueden beneficiar más del método en grupos: Un sistema que afecte a varios grupos de usuarios involucrados en actividades diferentes y con intereses diversos. Servirá como una nueva función de la empresa con la cual no existe una experiencia previa.
- **El cuestionario:** Puede ser utilizado en el trabajo de sistemas para obtener un consejo, para identificar una dirección o un área para un estudio más profundo, para realizar una auditoría posterior a la implementación y para identificar requerimientos específicos pero variables. Los analistas deben identificar lo que desean saber, preparar y entregar el cuestionario a la persona que lo va a contestar.
- **Observación:** Otra técnica con que cuenta el analista durante la indagación de hechos que consiste en observar a las personas en el momento de ejecutar su trabajo. El propósito de la observación es múltiple. Le permite al analista determinar lo que se está haciendo, la forma en que se hace, quien lo realiza, cuándo, cuánto tiempo requiere, dónde se hace y por qué (16).

En este tema se desarrollarán los tipos de Pruebas de Seguridad acorde con las categorías planteadas anteriormente estas son:

- **Descubrimiento de aplicaciones:** Es el proceso destinado a identificar aplicaciones web instaladas en una infraestructura dada.
- **Análisis de códigos de error:** El factor más importante para esta actividad es enfocar la atención en los errores encontrados en las aplicaciones.
- **Pruebas de SSL/TLS:** SSL y TLS son dos protocolos que proveen, con el apoyo de criptografía, de canales de transmisión de datos seguros, para la protección, confidencialidad y autenticación de la información transmitida. Teniendo en cuenta la criticidad de estas implementaciones de seguridad, es importante verificar el uso de un algoritmo de cifrado robusto y su implementación adecuada.

Comprobación de las reglas del negocio: que consiste en comprobar las reglas del negocio definidas en la aplicación.

Las reglas del negocio o conjunto de reglas de negocio describe las políticas, normas, operaciones, definiciones y restricciones presentes en una organización y que son de vital importancia para alcanzar los objetivos misionales.

Las reglas de negocio deben ser:

- Declarativas.
- Atómicas.
- Expresadas en lenguaje natural.
- Orientadas al negocio.

Especificación Formal: Las reglas del negocio pueden ser expresadas en un lenguaje formal de acuerdo a la naturaleza de la organización. Los lenguajes más ampliamente utilizados desde el 2008 incluyen: UML, *Business Process Modeling Notation(BPMN)*.

En ella se utiliza la siguiente Prueba de Seguridad:

Comprobación de las reglas del negocio: La lógica de negocio puede contener fallos de seguridad que permiten a un usuario hacer algo no permitido por el negocio.

Comprobación de la autenticación: Poner a prueba el sistema de autenticación de la aplicación. Se presenta como servicio de autenticación algo que verifica la identidad elegida y con un término en desuso (por el motivo expuesto en autenticación) referido a la integridad de los datos.

La comprobación de esa autenticación se hará de tres formas:

- **Usando código directo:** En el propio código de la aplicación se tendrán los nombres y las claves.
- **Los nombres de los usuarios y las claves estarán en el fichero:** Para mayor nivel de seguridad no se guardara la clave, sino un valor HASH[8] que servirá para comprobar si la clave es buena. Para este caso, se necesitará una pequeña utilidad para generar ese valor.
- **Los nombres y claves se guardan en una base de datos:** Para un buen nivel de seguridad, se recomienda que las claves se guarden como valores HASH, no se podrá guardar directamente, el problema en este caso es que no se podrá saber la clave del usuario, salvo que en lugar de dejar que se genere automáticamente (17).

[8] *Una función o método para generar claves o llaves que representen de manera casi unívoca a un documento.*

El tipo de prueba a utilizar es:

- **Fuerza bruta:** Consiste en averiguar el usuario y contraseña válidos de un individuo registrado en el sistema, mediante el intento reiterado de diferentes combinaciones de usuarios y contraseñas.

Validación de datos: Verificar que todas las entradas de datos estén validadas.

Algunos puntos que se deben de saber para validar datos:

- Debe verificarse la exactitud de los datos críticos, independientemente de si fueron ingresados a mano o transferidos electrónicamente.
- Los chequeos deben ser parte de procedimientos rutinarios para identificar errores.
- Deben existir procedimientos estándar para definir datos sin procesar, seguridad para la entrada de datos y revisión.
- Los resultados finales deben ser traceables a quien ingresó los datos o al instrumento desde el cual se incorporaron automáticamente. Los datos deben ser validados por personal calificado y autorizado siguiendo un procedimiento operativo estándar.

Antes de aprobar o rechazar se debe chequear:

- Correcta identificación de las muestra.
- Posibilidad de transmisión de errores.
- Plausibilidad.
- Consistencia.

El tipo de prueba a utilizar:

- **Inyección de procedimientos almacenados:** El simple uso de procedimientos almacenados no asiste en la mitigación de inyecciones SQL. Si no se tratan adecuadamente, las consultas dinámicas de SQL en los procedimientos almacenados pueden ser tan vulnerables a inyecciones de estas como lo son las consultas dinámicas desde una página web.
- **Inyección XML:** Se habla de pruebas de inyección XML cuando se trata de inyectar un determinado documento de éste tipo en la aplicación: Si el intérprete XML falla al realizar una validación adecuada de los datos, la prueba resultará positiva. Una inyección XML explota el siguiente patrón: Entrada -> documento XML == Inyección XML.

- **Inyección de código:** Estas pruebas pueden tener como objetivos diversos motores de *scripting*[9] del lado del servidor, como pueden ser ASP o PHP. Para protegerse frente a estos ataques será preciso emplear unas medidas adecuadas de validación y programación segura.
- **Validación de Datos:** La debilidad más común en la seguridad de aplicaciones web, es la falta de una validación adecuada de las entradas procedentes del cliente o del entorno de la aplicación. Esta debilidad conduce a casi todas las principales vulnerabilidades en aplicaciones, como inyecciones sobre el intérprete, sobre el sistema de archivos y desbordamientos de búfer.

Métricas

Una parte importante de un buen programa de seguridad es la habilidad para determinar si las cosas están mejorando. Es elemental seguir la pista de los resultados de los trabajos de prueba y desarrollar métricas que podrán revelar la evolución de la confirmación de la aplicación en la organización. Estas pueden mostrar si son necesarias para la educación y formación, si hay algún mecanismo en particular que no es comprendido con claridad y si el número total de problemas relacionados con la seguridad que se han encontrado cada mes se está reduciendo. Consistente en ser generadas automáticamente a partir del código fuente disponible ayudarán también a la organización en la evaluación de la eficacia de los mecanismos introducidos para reducir el número de errores de eficiencia en el desarrollo de software.

2.3 Fases de las Pruebas de Seguridad

Se podrán observar algunas fases de este tipo de prueba de seguridad que es tan necesario en todo momento a la hora de trabajar con documentos muy importantes, ellas son (18):

- **Fase de recopilación de información:** Utiliza los buscadores, herramientas de análisis de *Domain Name System* (DNS)[10] y herramientas para obtener información acerca de los hacker o algún intruso. Además se hace una exploración de metadatos de los documentos, imágenes y otros tipos de archivos que es estén manejando en esos momentos para obtener mayor seguridad.
- **Fase de enumeración:** Consigue direcciones IP de las víctimas, nombres de usuarios, contraseñas válidas de su entorno, nombres de servicios, aplicaciones accesibles, y todo aquello que luego pueda ayudar a lanzar un ataque. Esta fase al igual que la primera, solo es

[9] *Lenguaje de programación interpretado.*

[10] *Sistema de Nombres de Dominio*

de investigación, no se llevará a cabo ningún ataque, será más bien la realización los elementos existentes.

- **Fase de análisis:** Se empieza a actuar sobre los sistemas encontrados, se analizan en busca de vulnerabilidades, ya sea en la infraestructura, los sistemas operativos, los servicios disponibles o las aplicaciones existentes.
- **Fase de explotación:** En esta fase se realiza la intrusión en el sistema y se obtienen evidencias de la intrusión detectada para la posterior documentación o la demostración que se ha realizado la intrusión.
- **Fase de documentación**: En esta fase se deja por escrito de forma entendible y accesible todos los descubrimientos. Se registra todo lo que se ha descubierto sobre los sistemas, se realiza informes de las intrusiones y las evidencias de estas, se efectúa una presentación concisa y resumida de los resultados y se señala aquellos puntos que requieren especial importancia o que provocan los problemas más graves o inmediatos.

2.4 Técnicas de las Pruebas de Seguridad:

Los principales pasos que usan los intrusos para poder explorar los sistemas son: Reconocimiento del sistema y Escaneo de sistema.

Desde finales de los años 90 crecieron y mejoraron las técnicas empleadas por los usuarios. Entre las técnicas más conocidas y empleadas por los intrusos se encuentra: *barrido de puertos*, que es una de las técnicas favoritas desarrolladas para proporcionar la versión de los sistemas operativos, así como huecos potenciales a explotar en cada sistema.

Se hace referencia a los métodos y técnicas más importantes de este tipo de prueba:

- Identificar cada tipo de usuario, las funciones y datos a los que se debe autorizar.
- Crear pruebas para cada tipo de usuario, verificar cada permiso, creando transacciones específicas para cada tipo de cliente.
- Modificar tipos de usuarios y volver a ejecutar las pruebas.
- Si se conoce el funcionamiento interno del producto, se aplica este tipo de prueba para asegurarse de que todas las piezas encajan, es decir, que las operaciones internas se realizan de acuerdo a las especificaciones y que se han probado todos los componentes internos de manera adecuada

- Examina algún aspecto funcional de un sistema que tiene poca relación con la estructura lógica interna del software.
- Prueban las rutas lógicas del software y la colaboración entre componentes, al proporcionar casos de prueba que ejerciten conjuntos específicos de condiciones, bucles o ambos.

2.5 Herramientas

Las herramientas automatizadas dan soporte a las actividades de calidad, pruebas de integración, pruebas de sistema, diagnóstico o afinado de las aplicaciones en los entornos de certificación y los de producción. Generalmente son difíciles de realizar de una forma integrada, comprenden herramientas de análisis estático, automatización de pruebas funcionales, de carga, de rendimiento y de estrés.

Estas herramientas guían hacia múltiples ventajas y desventajas, por ello cuando se trata de pruebas, es necesario conocerlas para así seleccionar o no diversos caminos de solución a problemas que se puedan presentar.

Algunas ventajas de estos tipos de pruebas automatizadas:

- Rapidez en la ejecución.
- Realización de un mayor número de pruebas. Algunos de los problemas hallados por la automatización, tal vez no hubieran sido encontrados utilizando solo pruebas manuales, debido a limitantes de tiempo.
- Mayor confiabilidad en los resultados, mientras las pruebas o su información no cambie, deben de obtener siempre el mismo resultado; son consistentes, confiables y repetibles. Como seres humanos te cansas, preocupas o simplemente apresuras en sacar tu trabajo a tiempo. Todo esto lleva a simples errores humanos que afectan tu capacidad de ser eficiente en pruebas rutinarias. La automatización de pruebas repetitivas que requieren una ejecución frecuente, te permite tiempo para integrar pruebas más complejas, probar nuevas funciones dentro de la aplicación y su integración con el resto del sistema.
- Programación de la ejecución de pruebas en horarios no laborales como la noche y los fines de semana. Muchas herramientas brindan prestaciones que permiten la programación automática de un sinnúmero de pruebas, utilizando así el horario de trabajo para idear posteriores planes de prueba.

Estas herramientas también traen algunas desventajas las cuales son:

- Por lo general una gran cantidad de aplicaciones contienen elementos que no son compatibles con las herramientas que se utilizan para ponerlas a prueba, en consecuencia esto requiere la búsqueda de soluciones creativas para que éstas se adapten a la aplicación, lo cual constituye un obstáculo al que se tendrá que enfrentar el equipo de trabajo.
- Problema del mantenimiento. Cuando los sistemas sufren algún tipo de cambio, como ya se ha visto anteriormente, las pruebas también tienen que cambiar. El esfuerzo empleado en la actualización de las mismas es a veces causa de abandono de la iniciativa de su automatización y de un retorno a la ejecución manual.

Después de llevada a cabo la investigación se concretó que solo se ejecutarán las pruebas que utilicen herramientas automatizadas, que resulten muy difíciles realizarlas de forma manual. De la lista de herramientas analizadas se ha hecho énfasis total de las libres y de código abierto, pues las mismas representan un ahorro económico y amplias posibilidades de personalización a los intereses tecnológicos del centro. Las herramientas elegidas es de acuerdo a los tipos de pruebas que se van a validar, teniendo en cuenta que existe una documentación de ella bastante abarcadora sobre su modo de empleo y los requerimientos no funcionales que ella requiere.

La herramienta ***Netcraft*** se va a utilizar en la realización de las pruebas de Firmas de Aplicaciones que se van a emplear en el proyecto. Una de sus características principales que no ocupa mucho espacio (ni físico, ni virtual), ofrece beneficiosamente un servicio práctico y fundamental en los tiempos que está en ejecución. Esta herramienta es un sistema *anti-phishing*[11], que trabajará perfectamente integrado en el navegador *Internet Explorer*, con el fin de mantenerte totalmente aislado de los fraudes relacionados con el robo de los datos bancarios (números de cuenta, o de tarjetas de crédito, contraseña y datos personales). El mecanismo de actuación está basado en el registro y en la difusión de las URL marcadas como fraudulentas por parte de los usuarios. La versión de *Netcraft* es 3.8.5.

[11] *Es una modalidad de estafa diseñada con la finalidad de robar la identidad.*

Ventajas:

- Es una aplicación gratuita para Windows, que instala una barra en el navegador de Internet Explorer que indica el nivel de riesgo de las páginas visitadas y avisa si detecta que alguna de ellas está destinada al *phishing.*
- Su instalación es sencilla, muestra información adicional acerca de la antigüedad de la página y del servidor que está en ejecución.
- Se puede reportar páginas que no estén marcadas como maliciosas y así de esta manera poder avisar a los demás usuarios de esta barra.
- El Netcraft es una las auditoras más reconocidas del sector a nivel internacional, que nos coloca dentro de los informes que publica cada mes como la empresa de hosting.
- Utiliza el DNS para identificar qué sitios hay en la Web y comprobar cuántos de ellos están en una localización particular. Comprueba además cuál es el sistema operativo y cuál el software del servidor Web, para terminar publicando su información en un informe mensual.

Desventajas:

- Se tarda para cargar los programas que vayas a trabajar.
- Es un poco complejo en el período que se va a trabajar con él, por funcionar a través de líneas de comandos.
- Los comandos y la bibliografía que utiliza es en ingles.
- Las actualizaciones requieren en ocasiones tener conocimientos profundos del sistema.
- Configurar algunos servicios de red requiere de más tiempo que en Windows.
- Mayor coste del personal.

¿Cómo funciona esta herramienta?

La posición de riesgo mostrada por la barra de herramientas Netcraft ofrece un nivel adicional de la protección contra nuevos sitios que no están todavía en la base de datos de Netcraft.

- Dispone de un formato de fácil acceso, que brinda la posibilidad de consulta al público en general, ofreciéndole la oportunidad de crear y almacenar en una galería cuadros, gráficos, mapas temáticos, en informes propios del usuario, exportables a programas de Microsoft Office o una página web, facilitando la diseminación de información a través de medios magnéticos (CD-ROM), Internet.

- Los indicadores contenidos en las bases de datos vienen acompañados de su respectiva información técnica que es necesaria para facilitar su adecuada interpretación y análisis, siendo accesibles de manera dinámica durante el proceso de manejo y consulta de la información.
- Su acceso puede realizarse tanto como aplicación en escritorio, como por vía web, permitiendo acceder a las actualizaciones de las bases de datos, siendo esto último unas de las principales novedades del sistema.
- Dispone de un módulo de personalización que permite hacer cambios en los colores, idioma, fuentes para darle un diseño acorde a las necesidades del usuario.

La barra de dicha herramienta presenta varias funciones como:

- Atrapa URLs sospechosos con conteniendo de caracteres que no tienen ningún objetivo común además de engañar.
- Permite la demostración del navegador de una barra de herramientas y una barra de direcciones en todas las ventanas, para detectar y dar a conocer si alguna de estas presentan algún tipo de peligro a la hora de trabajar con ellas.
- Se puede evaluar las direcciones URLs fraudulentas que se vayan a visitar y te permite identificar las que sean seguras a la hora de visitarlas.

Al utilizar este tipo de herramienta se podrá conocer el tipo de servidor que tiene el sistema y la versión en que se encuentra. Al aplicar esta herramienta se obtiene como resultado si existe alguna falla en el servidor utilizado y muestra como resultado las direcciones URLs que sean seguras a la hora de trabajar con ella.

La prueba de Fuerza Bruta utiliza la herramienta llamada *Brutus:* Es uno de los más rápidos a distancia y más flexible, es una herramienta libre. Brutus se hizo por primera vez a disposición del público en octubre de 1998 y desde entonces ha habido por lo menos 70.000 descargas y más de 175.000 visitantes a esta página. El desarrollo sigue tan nuevas versiones estarán disponibles en un futuro próximo. Bruto fue originalmente escrito para ayudar a ver *routers* de forma predeterminada y contraseñas comunes (22).

Algunas características de Brutus (22)**:**

Utiliza la versión actual AET2 Brutus e incluye los tipos de autenticación siguiente:

- HTTP (autenticación básica).
- HTTP (Formulario HTML/CGI).
- POP3
- FTP
- SMB
- Telnet

La versión actual incluye las siguientes funcionalidades: (22).

- Presenta varias etapas del motor de autenticación.
- Realiza 60 conexiones simultáneas de destino.
- Presenta N nombres de usuario único y nombres de usuarios de múltiples modos.
- Contiene listas de contraseñas, combinado (usuario/contraseña) y configura los modos de la fuerza bruta.
- Tiene secuencia de autenticación personalizable.
- Importa y Exporta tipos de autenticación personalizadas como archivos sin problemas.

Ventajas:

- Dispone de un formato de fácil acceso, que brinda la posibilidad de consulta al público en general, ofreciéndole la oportunidad de crear y almacenar en una galería cuadros, gráficos, mapas temáticos, en informes propios del usuario, exportables a programas de Microsoft Office o una página web, facilitando la información a través de medios magnéticos (CD-ROM), Internet.
- Los indicadores contenidos en las bases de datos vienen acompañados de su respectiva información técnica (metainformación), necesaria para facilitar su adecuada interpretación y análisis, siendo accesibles de manera dinámica durante el proceso de manejo y consulta de la información.

- Su acceso puede realizarse tanto como aplicación en escritorio, como por vía web, permitiendo acceder a las actualizaciones de las bases de datos, siendo esto último unas de las principales novedades del sistema.
- Dispone de un módulo de personalización que permite hacer cambios en los colores, idioma, fuentes para darle un diseño acorde a las necesidades del usuario.

Desventajas:

- No tiene modo de fuerza bruta
- Para que funcione en windows necesitas de otra herramienta que es: cygwin.

Esta herramienta es un programa de fuerza bruta capaz de adivinar contraseñas mediante diccionarios de palabras, los diccionarios están en un .DOC y se puede crear manualmente. No admite diccionarios en formato .DOC, solo en formato de texto plano.

El papel de las herramientas automatizadas.

Existen varias compañías que comercializan herramientas de análisis y comprobación de seguridad automatizadas. Es necesario recordar las limitaciones de estas herramientas, de modo que puedan emplearlas para aquello en lo que son más útiles.

Lo más importante es que estas herramientas son genéricas, es decir, que no están diseñadas para un código específico, sino para aplicaciones en general. Lo que significa que aunque se pueden encontrar algunos problemas genéricos, no tienen el conocimiento suficiente sobre la aplicación como para permitirles detectar la mayoría de los fallos. Las incidencias de seguridad más serias son aquellas que no son genéricas, sino profundamente confusas en la lógica del negocio y en el diseño de la aplicación.

2.6 Conclusiones parciales

Con la realización de este capítulo se demuestra la importancia que tiene la aplicación de pruebas de software para lograr obtener un producto con la calidad requerida. Se ha explicado las pruebas a utilizar a la PTARTV, así como la caracterización de los diferentes tipos de herramientas, sus principios, se dejó planteado el procedimiento para la realización de las pruebas y cómo deben ser usadas para obtener un resultado óptimo y favorable.

Capitulo 3: Propuesta y validación de procedimientos de las Pruebas de Seguridad.

3.1 Introducción

Las Pruebas de Seguridad que se realizan al software tienen un impacto importante en la calidad del producto final, por esta razón en el presente capítulo se ofrece una propuesta de solución para garantizar la realización de Pruebas de Seguridad al Módulo de Programación del proyecto PTARTV. Se va a realizar una validación acerca del procedimiento propuesto para realizar las Pruebas de Seguridad en dicho proyecto.

3.2 Entorno de las pruebas

Las pruebas serán llevadas a cabo en Sistema Operativo Linux en la versión Ubuntu 10.7, ya que así fue acordado por los clientes en un inicio, de esta forma se probará el módulo simulando el entorno donde se va a utilizar. Las computadoras cliente tendrán estas características: Procesador (CPU): Intel (R) Pentium (R) 4. Velocidad de CPU 3.0 GHz Y 1 GB de Memoria tipo RAM, conectándose a través de la red hacia el servidor, el cual cuenta con las mismas características de las ordenadores de los clientes, el mismo corre sobre el sistema operativo GNU/Linux en una computadora que tiene como.

Descripción

Este procedimiento define de forma detallada los pasos para llevar a cabo las Pruebas de Seguridad a una aplicación Web. En el proyecto PTARTV se hace necesario hacer pruebas que validen la calidad durante todo el proceso de desarrollo. Primeramente se deben especificar las fases en las que se aplicarán las pruebas, para ello se tiene en cuenta que estas, según la metodología basada en RUP, se aplican en la fase Inicio, Elaboración, Construcción y Transición se puede observar en la (Figura 4). En la fase Inicial se diseñarán los casos de pruebas para definir la misión de evaluación. Luego en la de Elaboración se trazará el Plan de Pruebas definiendo también la misión de evaluación, además de probar y evaluar pruebas, verificando así el enfoque de la misma. Por último y no menos importante, en la fase de Construcción se aplicarán las Pruebas de Seguridad para así detectar los fallos que esta pudiera tener.

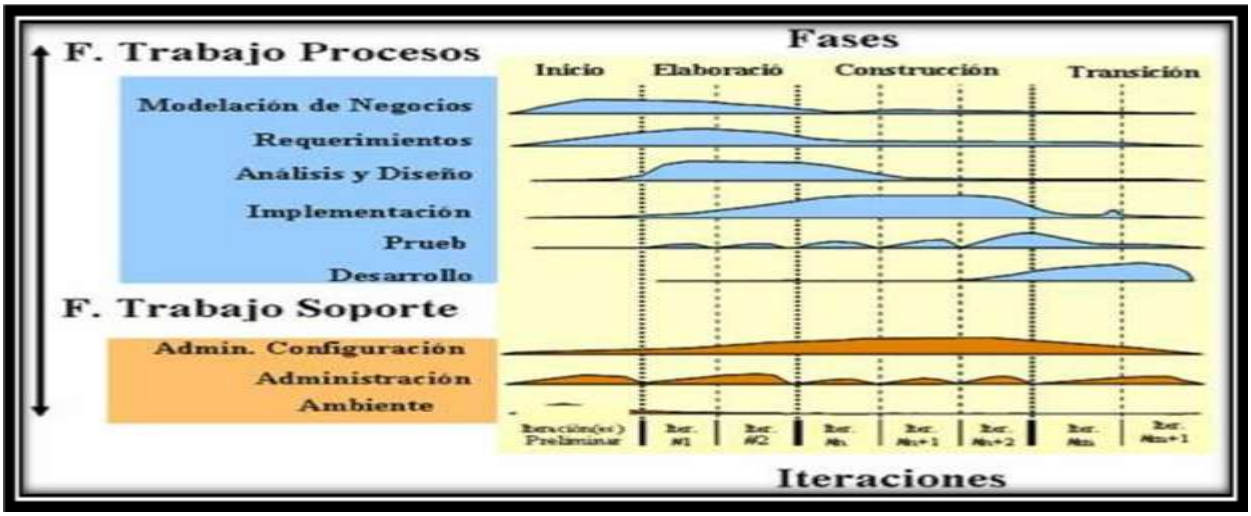

Figura 4: Fases de RUP

Alcance

El objetivo de este Plan de Pruebas es guiar las actividades referentes a la realización de las pruebas al Módulo de Programación perteneciente a la PTARTV, el cual se le realizarán dos iteraciones de pruebas. Realizada la primera iteración deben de obtenerse las no conformidades, las cuales serán entregadas al equipo de desarrollo, para luego ser corregidas en dependencia del nivel de complejidad que posean.

3.3.1 Descripción de las actividades del Procedimiento.

El procedimiento cuenta con cinco fases fundamentales, se pueden observar en la Tabla 1:

Tabla 1: Fases de Pruebas

Código	Actividad	Descripción
A.1	Planificación de las pruebas.	Coordinar el comienzo de las pruebas. Se desarrollará un Plan de Pruebas de Seguridad.
A.2	Diseño de las pruebas.	Diseñar las listas de chequeo y los casos de prueba.
A.3	Ejecución de la pruebas.	Ejecutar las pruebas mediante el uso de las listas de chequeo y los casos de prueba.
A.4	Documento e informe de los resultados.	Documentar y realizar un informe de las no conformidades encontradas.
A.5	Depuración de errores.	Solución de los errores encontrados.

3.3.1.1 Planificación de las pruebas

En este paso se planificarán las pruebas que se le realizarán a la aplicación, así como se determinará los roles y los recursos que intervendrán en la ejecución de las pruebas. Se desarrollará el Plan de Pruebas con el propósito de explicar el alcance, enfoque, recursos requeridos, calendario, responsables y manejo de riesgos de un proceso de pruebas. Para el diseño del plan de pruebas se va a utilizar la plantilla que se utilizó es la definida por el Centro de Calisoft, la cual cuenta con la organización del equipo que ejecutaran las pruebas, la arquitectura técnica que es la que muestra el diagrama de despliegue del Módulo de Programación, las especificaciones del software y hardware (Ver Anexo 1). El Plan de Prueba identificará los elementos de estas, los recursos necesarios para la ejecución, se va a definir, recomendar las estrategias y el cronograma de las pruebas que se encuentra en la Tabla 2

Tabla 2: Cronograma de Pruebas

No.	Tarea	Fecha	Responsable	Participantes	Etapa	Observaciones
1	Pruebas de Fuerza Bruta.	La 1era iteración desde el 29/11/2010 al 1/12/2010 La 2da iteración desde el 19/05/2011 al 23/05/2011	Maydelin Navarro Suárez	Maydelin Navarro Suárez	Prueba	
2	Pruebas de Escalación de Privilegios.	La 1era iteración desde el 29/11/2010 al 1/12/2010 La 2da iteración	Maydelin Navarro Suárez	Maydelin Navarro Suárez	Prueba	

		desde el 19/05/2011 al 23/05/2011				
3	Pruebas de Firma de Aplicaciones.	La 1era iteración desde el 29/11/2010 al 1/12/2010 La 2da iteración desde el 19/05/2011 al 23/05/2011	Maydelin Navarro Suárez	Maydelin Navarro Suárez	Prueba	
4	Pruebas de Inyección SQL	La 1era iteración desde el 29/11/2010 al 1/12/2010 La 2da iteración desde el 19/05/2011 al 23/05/2011	Maydelin Navarro Suárez	Maydelin Navarro Suárez	Prueba	
5	Pruebas al sistema del Módulo de Programación	Desde el 19/05/2011 al 23/05/2011	Maydelin Navarro Suárez	Maydelin Navarro Suárez y Yandys Menéndez Delgado	Prueba	

3.3.1.2 Diseño de las pruebas

En esta actividad se diseñarán los casos de pruebas correspondientes a las categorías de pruebas Comprobación de las Reglas del Negocio y Validación de Datos. Además se elaborarán las listas de chequeo que se utilizarán en las categorías de pruebas Recopilación de Información y Comprobación de la Autenticación. Estas listas de chequeos y casos de pruebas que se explicarán a continuación la forma general de medir los aspectos que se deben tener en cuenta para la realización de esta. En caso de surgir nuevos aspectos a medir estos se adicionarán (Ver Anexo 2).

A continuación se van a especificar los requerimientos de los casos de uso que se van a utilizar para diseñar los casos de pruebas los cuales se van a utilizar para la ejecución de las pruebas:

- **Autenticar usuario** se inicia cuando el actor introduce los datos que se solicitan para acceder a la aplicación, estos se verifican y el mismo finaliza dándole los permisos, en caso de poseer los privilegios necesarios.
- **Gestionar planificaciones web** comienza cuando el programador web accede al sistema para poder adicionar o eliminar una programación en la web. Este caso de uso finaliza cuando el programador web haya realizado satisfactoriamente o no alguna de las operaciones mencionadas anteriormente.
- **Visualizar medias publicadas** comienza cuando el programador web accede al sistema para poder visualizar las medias que se encuentran en publicación y a su vez pueda finalizar esta publicación. Este caso de uso finaliza cuando el programador web haya visto las medias que se encuentran publicadas y haya eliminado o no una publicación.
- **Gestionar planificación de radio** comienza cuando el programador de radio accede al sistema para poder adicionar, modificar o eliminar una planificación de radio. Este caso de uso finaliza cuando se haya realizado satisfactoriamente o no alguna de las operaciones mencionadas anteriormente.
- **Gestionar programación de radio** comienza cuando el programador de radio accede al sistema para poder adicionar, modificar o eliminar una programación de radio. Este caso de uso finaliza cuando se hayan realizado satisfactoriamente o no alguna de las operaciones mencionadas anteriormente.

- **Exportar programación de radio a PDF** comienza cuando el programador de radio accede al sistema para poder exportar al formato PDF el listado de programaciones de radio. Se deben listar todas las programaciones y de ahí seleccionar las que se desee exportar. Este caso de uso finaliza cuando se hayan exportado o no las programaciones de radio deseadas.
- **Gestionar espacio de radio** comienza cuando el programador de radio accede al sistema para poder adicionar, modificar o eliminar un espacio de radio. Este caso de uso finaliza cuando se hayan realizado satisfactoriamente o no alguna de las operaciones mencionadas anteriormente.
- **Gestionar planificación de televisión** comienza cuando el programador de televisión accede al sistema para poder adicionar, modificar o eliminar una planificación de televisión. Este caso de uso finaliza cuando se hayan realizado satisfactoriamente o no alguna de las operaciones mencionadas anteriormente.
- **Gestionar programación de televisión** comienza cuando el programador de televisión accede al sistema para poder adicionar, modificar o eliminar una programación de televisión. Este caso de uso finaliza cuando se haya realizado satisfactoriamente o no alguna de las operaciones mencionadas anteriormente.
- **Exportar programación de televisión a PDF** comienza cuando el programador de televisión accede al sistema para poder exportar al formato PDF el listado de programaciones de televisión. Se deben listar todas las programaciones y de ahí seleccionar las que se desee exportar. Este caso de uso finaliza cuando se hayan exportado a PDF las programaciones de televisión deseadas.
- **Gestionar espacio de televisión** comienza cuando el programador de televisión accede al sistema para poder adicionar, modificar o eliminar un espacio de televisión. Este caso de uso finaliza cuando se haya realizado satisfactoriamente o no alguna de las operaciones mencionadas anteriormente.

3.3.1.3 Ejecución de las pruebas

Se ejecutan las categorías de pruebas definidas haciendo uso de las listas de chequeo y los casos de prueba diseñados en los epígrafes anteriores. La ejecución de las pruebas se va a realizar comenzando por la Recopilación de Información, luego de esta prueba se pueden realizar las de Comprobación de la Autenticación, Comprobación de las Reglas del Negocio y Validación de Datos en paralelo o en el orden que el especialista de prueba decida. El probador ejecuta las pruebas y al mismo tiempo va registrando las no conformidades detectadas.

3.3.1.4 Documentación e informe de los errores

Como se mencionó anteriormente los probadores ejecutan las pruebas y en paralelo van a ir registrando las no conformidades encontradas en el registro de defectos y dificultades. Luego que se haya registrado todas las no conformidades encontradas en la ejecución de las pruebas, el especialista al frente de las pruebas las revisa y las guarda en el informe de resultados que también se describe a continuación (Ver Anexo 3).

Al analizar la 1era iteración al Módulo de Programación de la plataforma PTARTV se le detectaron un grupo de no conformidades. Las cuales se muestran en la figura 4.

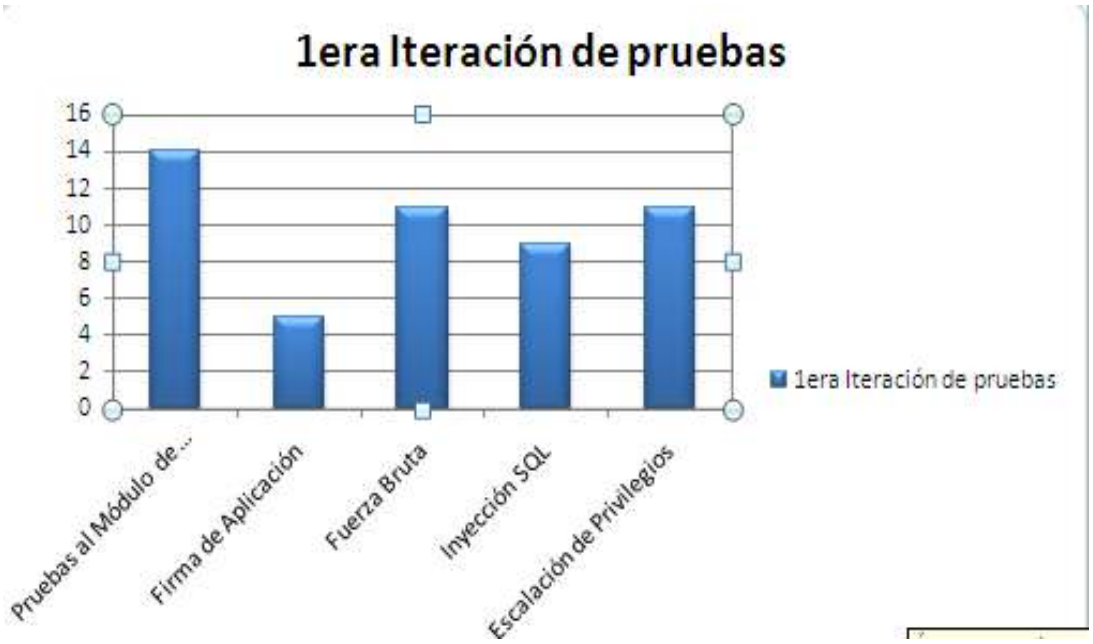

Figura 5: 1era iteración de pruebas

Después de haber realizado esta 1era iteración de pruebas se realizó una reunión entre el equipo de desarrollo y el equipo revisor. Para analizar las no conformidades detectadas en las pruebas.

Una vez concluido el periodo de tiempo que le dio el equipo revisor para que resolvieran las no conformidades encontradas en la 1era iteración. Se realizó una 2da iteración arrojando las no conformidades que se muestran en la figura 5.

Figura 6: 2da Iteración de pruebas.

Al realizar estas dos iteraciones al Módulo de Programación se pudo observar como disminuyeron los errores de este sistema, el cual se encuentra en fase de desarrollo actualmente. En la figura 6 se observa que existió una reducción entre las no conformidades de la 1era iteración y la 2da concluyendo que el equipo de desarrollo resolvió un grupo importante de los defectos detectados.

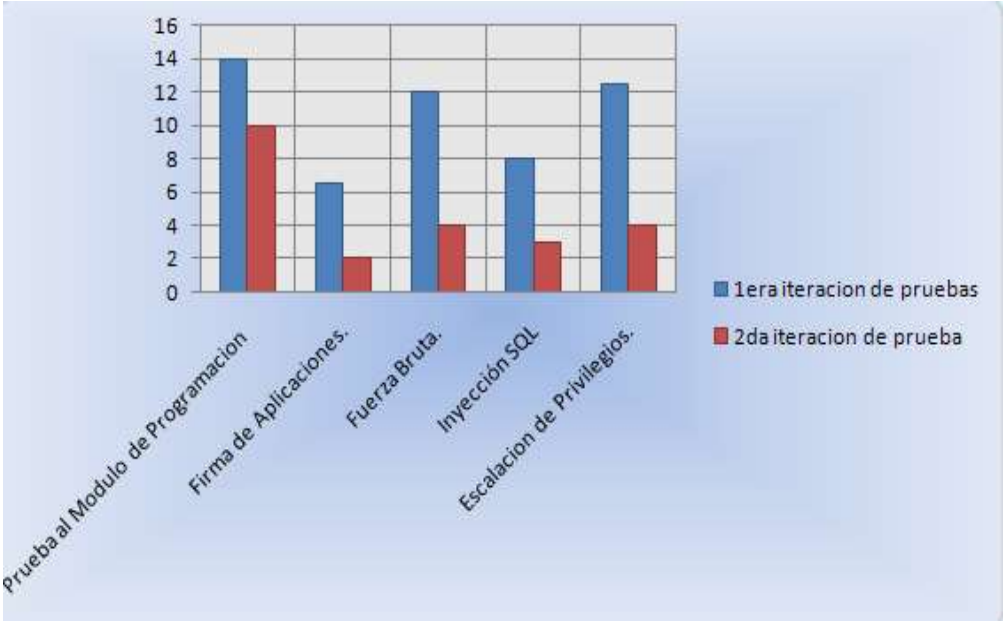

Figura 7: Comparación entre las iteraciones realizadas.

3.3.1.5 Depuración de los errores

En esta fase el equipo de desarrollo se reúne, ven las no conformidades encontradas en la fase anterior y las resuelve. Esto se realiza mediante una reunión del equipo de desarrollo de calidad con los directivos del proyecto para explicar los errores encontrados.

3.4 Conclusiones parciales

De manera general el procedimiento para Pruebas de Seguridad en aplicaciones web fue evaluado mediante pruebas que pertenecen a unas series de actividades para valorar en qué estado se encuentra el proyecto en cuanto a la seguridad y calidad. Al efectuar las pruebas quedó evidenciado que la solución propuesta mejora la seguridad del Módulo de Programación del proyecto PTARTV, pues detecta mayor cantidad de no conformidades y garantiza que el código quede bien escrito y estructurado.

CONCLUSIONES

Conclusiones

Luego de realizar un estudio del proceso de pruebas que se realiza a la PTARTV se llegó a la conclusión que este es un proyecto al cual solo se han realizado pruebas de Caja Negra y Caja Blanca. Teniendo en cuenta este aspecto y que el Módulo de Programación perteneciente a este proyecto es un punto clave para asegurar la calidad del sistema, se determinó aplicar Pruebas de Seguridad. Con este fin se propuso un procedimiento para guiar el proceso de las pruebas a seguir.

Este es un procedimiento propuesto para aplicaciones Web debido a que el Módulo de Programación es una aplicación de este tipo. Con el objetivo de determinar vulnerabilidades en el sistema de forma más rápida y segura se realizó un estudio de diferentes herramientas que existen en el mundo y se determinó usar la herramienta Netcraft para la prueba de Firma de Aplicaciones y para las pruebas de Fuerza Bruta se utilizó Brutus. Estas herramientas fueron escogidas ya representan un ahorro económico y amplias posibilidades de personalización a los intereses tecnológicos del centro. Teniendo en cuenta que existe una documentación de ella bastante abarcadora sobre su modo de empleo y los requerimientos no funcionales que ella requiere. Utilizan software libre y son de código abierto.

El procedimiento propuesto fue aplicado al Módulo de Programación. Con la aplicación de las Pruebas de Seguridad y con los resultados obtenidos se pudo llegar a la conclusión que el sistema no cuenta con una seguridad que logre mantener la integridad de la información que se maneja en el proyecto PTARTV.

Una vez determinado estas vulnerabilidades el proyecto trabaja para eliminarlas logrando así de esta forma un sistema más seguro con mayor calidad en cuento a este aspecto. Esto permite cumplir con el objetivo general de la investigación el cual propone “Definir un procedimiento para realizar Pruebas de Seguridad a la PTARTV” y contrastar la idea de defender que asegura “que con la definición y posterior aplicación de un procedimiento para el desarrollo de Pruebas de Seguridad a la PTARTV, se obtendrá un sistema con mayor seguridad.” Estos resultados fueron comunicados a los miembros del proyecto, dando paso a la validación de la propuesta.

RECOMENDACIONES

Recomendaciones

Existen algunos aspectos que se desglosan de este trabajo investigativo en los cuales serían útiles seguir profundizando y otros que se recomiendan tener en cuenta como son:

- Aplicar la propuesta del procedimiento para realizar Pruebas de Seguridad a aplicaciones web en varios proyectos de la UCI y se analicen los resultados como una validación práctica del mismo.
- Emplear las siguientes herramientas: Netcraft y Brutus para las Pruebas de Seguridad.
- Profundizar en el estudio de las Pruebas de Seguridad en aplicaciones web.
- Probar y evaluar nuevas herramientas para realizar las Pruebas de Seguridad.
- Incorporar conocimiento de pruebas de software en el programa docente educativo desde segundo año de la carrera, el cual permitirá formar tempranamente estudiantes con cultura de pruebas y las herramientas necesarias para su desempeño como probadores.
- Ampliar el estudio de las herramientas de apoyo a la realización de pruebas donde se incluyan herramientas libres y multiplataforma que permitan probar aplicando las Pruebas de Seguridad o Intrusión.

Bibliografía Referenciada

1. Soto**, prof Lauro.** Definicion Calidad De Software. *Definicion Calidad De Software.* [En línea] [Citado el: 15 de octubre de 2010.] http://www.mitecnologico.com/Main/DefinicionCalidadDeSoftware.

2. **Altozano, Esteban.** ¿ Qué es la Calidad ? *¿ Qué es la Calidad ?* [En línea] [Citado el: 15 de 11 de 2010.] http://www.agoratel.com/recursos/docs_calidad/calidad.htm.

3. **Lefcovich, Mauricio.** Calidad total. *Calidad total.* [En línea] 14 de 12 de 2005. [Citado el: 17 de 11 de 2010.] http://www.degerencia.com/articulo/calidad_total.

4. **McCall.** Calidad del Software. *Calidad del Software.* [En línea] [Citado el: 18 de 11 de 2010.] http://www.usfx.edu.bo/biblioteca/practicos/archivos/SIS301/CALIDAD.PDF.

5. **Carlos, Vega Lebrún.** BIBLIOTECA VIRTUAL de Derecho, Economía y Ciencias Sociales. *BIBLIOTECA VIRTUAL de Derecho, Economía y Ciencias Sociales.* [En línea] [Citado el: 19 de 01 de 2011.] http://www.eumed.net/libros/2008a/351/Calidad%20de%20Software.htm.

6. **Agüero, Dennis Neuland.** Áreas del aseguramiento de la calidad. [En línea] 2008. [Citado el: 9 de 2 de 2010.] http://calisoft.uci.cu/tmp/documentos/articulos/articulo_sqa.pdf.

7. **Lovelle, Juan Manuel Cueva.** Calidad del Software. *Calidad del Software.* [En línea] 21 de 10 de 1999. [Citado el: 18 de 11 de 2010.]

http://gidis.ing.unlpam.edu.ar/downloads/pdfs/Calidad_software.PDF.

8. **WorldLingo.** Prueba del software . *Prueba del software .* [En línea] 2002. [Citado el: 24 de 11 de 2010.] http://www.worldlingo.com/ma/enwiki/es/Software_testing.

9. **Usaola, Dr. Macario Polo.** Mantenimiento Avanzado de Sistemas de Información Pruebas del Software. *Mantenimiento Avanzado de Sistemas de Información Pruebas del Software.* [En línea] [Citado el: 15 de 10 de 2010.] http://alarcos.inf-cr.uclm.es/doc/masi/doc/lec/parte5/polo-apuntesp5.pdf.

10. Gestión de Calidad y Pruebas de Software. *Gestión de Calidad y Pruebas de Software.* [En línea] 2005. [Citado el: 10 de 11 de 2010.] http://www.pruebasdesoftware.com/laspruebasdesoftware.htm.

11. **Mirelis, Gabriel Alberto Garcia.** Pruebas de software. *Pruebas de software.* [En línea] 4 de 2008. [Citado el: 18 de 11 de 2010.] http://www.mat.uson.mx/mireles/PruebaDelSoftware20081_archivos/frame.htm.

BIBLIOGRAFÍA REFERENCIADA

12. **Prado, Elena Raja.** Casi todas las pruebas del software. *Casi todas las pruebas del software.* [En línea] 2007. [Citado el: 12 de 10 de 2010.] http://www.sistedes.es/sistedes/pdf/2007/pris-07-raja-ctps.pdf.

13. **Mañas, Jose A.** Prueba de Programas. *Prueba de Programas.* [En línea] 16 de 3 de 1994. [Citado el: 25 de 11 de 2010.] http://www.lab.dit.upm.es/~lprg/material/apuntes/pruebas/testing.htm#sB.

14. **Whittaker, James A.** Pruebas de Seguridad. *Pruebas de Seguridad.* [En línea] 5 de 2008. [Citado el: 18 de 11 de 2010.] http://msdn.microsoft.com/es-es/magazine/cc507646.aspx#S3.

15. *Prueba de Seguridad penetrante.* **Barney, Lee.** 2005-2010, Prueba de Seguridad penetrante, pág. 1.

16. unidad 5 tecnicas de recopilacion de informacion. *unidad 5 tecnicas de recopilacion de informacion.* [En línea] 2011. [Citado el: 24 de 02 de 2011.]

17. **ASP.NET.** Tutorial para la creación de un sitio Web con autenticación mediante formulario. *Tutorial para la creación de un sitio Web con autenticación mediante formulario.* [En línea] [Citado el: 24 de 02 de 2011.] http://www.elguille.info/NET/ASPNET/tutorialLogin/tutorialLogin.htm.

18. Fases de un test de penetración. *Fases de un test de penetración.* [En línea] 04 de 08 de 2010. [Citado el: 23 de 02 de 2010.] http://infow.wordpress.com/2010/08/04/fases-de-un-test-de-penetracion/.

19. **WorldLingo.** Prueba de Seguridad . *Prueba de Seguridad .* [En línea] 2011. [Citado el: 23 de 02 de 2011.] http://www.worldlingo.com/ma/enwiki/es/Penetration_test#Web_application_penetration_testing.

20. softonic. *softonic.* [En línea] 2010. [Citado el: 22 de 02 de 2011.] http://shadow-security-scanner.softonic.com/.

21. Iguna, Herramienta gratuita para realizar de Pruebas de Seguridad. *Iguna, Herramienta gratuita para realizar de Pruebas de Seguridad.* [En línea] 08 de 08 de 2007. [Citado el: 22 de 02 de 2011.] http://www.dragonjar.org/iguna-herramienta-gratuita-para-realizar-de-Pruebas-de-Seguridad.xhtml.

22. **referencias acerca de brutus.** [En línea] 30 de 04 de 2001. [Citado el: 22 de 02 de 2011.] http://translate.google.com/translate?hl=es&sl=en&u=http://www.hoobie.net/brutus/&ei=amxkTfzJCs6hOoGAmfcF&sa=X&oi=translate&ct=result&resnum=4&ved=0CEMQ7gEwAw&prev=/search%3Fq%3Dbrutus%26hl%3Des%26biw%3D1148%26bih%3D659%26prmd%3Divns.

23. **Fyodor.** Las 75 Herramientas de Seguridad Más Usadas. *Las 75 Herramientas de Seguridad Más Usadas.* [En línea] [Citado el: 22 de 02 de 2011.] http://insecure.org/tools/tools-es.html.

BIBLIOGRAFÍA REFERENCIADA

24. **Soto, prof Lauro.** Definicion Calidad De Software. *Definicion Calidad De Software.* [En línea] [Citado el: 20 de 10 de 2010.] http://www.mitecnologico.com/Main/DefinicionCalidadDeSoftware.

25. **DRAE.** *Diccionario de la Real Academia Española.* 2005.

26. **Cueva, Juan Manuel.** Calidad del Software. [En línea] 21 de 10 de 1999. [Citado el: 15 de 1 de 2010.] http://www.itescam.edu.mx/principal/sylabus/fpdb/recursos/r35043.PDF.

27. **GCCF.** GESTIÓN DE LA CALIDAD CONCEPTOS Y FILOSOFÍAS. [En línea] 2000. [Citado el: 2 de 11 de 2010.] http://www.scribd.com/doc/2628724/GESTION-DE-LA-CALIDAD-CONCEPTOS-Y-FILOSOFIAS.

28. **ISO 9000:2000.** Escuela de Ingeniería Electrónica Universidad Nacional de Rosario. [En línea] [Citado el: 2010 de 1 de 15.]

http://www.eie.fceia.unr.edu.ar/ftp/Gestion%20de%20la%20calidad/ISO%209000-2000(ES).pdf.

29. **Juran, Joseph Manuel.** SITIO DEL INSTITUTO PANAMERICANO DE GESTIÓN DE LA SALUD. [En línea] 2002. [Citado el: 2 de 11 de 2010.] http://www.gerenciasalud.com/art483.htm.

30. **ISO 8402.** Términos Generales. [En línea] [Citado el: 2 de 11 de 2010.] http://ver.megared.net.mx/~jccz/iso8402.html.

31. **Carrasco, O.** [En línea] 1995. [Citado el: 28 de 10 de 2010.] http://bvs.sld.cu/revistas/aci/vol3_3_95/aci05395.htm.

32. **Calero, D.C.** Modelos de Calidad. [En línea] 2007. [Citado el: 29 de 10 de 2010.] http://bdigital.eafit.edu.co/bdigital/PROYECTO/P005.14CDP613/marcoTeorico.pdf.

33. **Pressman, R S.** *Ingeniería del Software, un enfoque práctico.* 2005.

34. **Vazquez, J P.** Verificación con XML. [En línea] 2001. [Citado el: 29 de 10 de 2010.] http://gredos.usal.es:800/getblob?blobid=4002222026466598..

35. **ISO 9241.** Términos Generales. [En línea] 2006. [Citado el: 15 de 11 de 2010.] http://ver.megared.net.mx/~jccz/iso8402.htm.

ANEXOS

Anexos

Anexo 1: Plan de Pruebas

1. Introducción Este documento se confecciona con el objetivo de definir el Plan de las Pruebas de Seguridad al Módulo de Programación del proyecto PTARTV.

1.1 Objetivos Los objetivos de este Plan de Pruebas de Seguridad consisten en:

- Identificar los elementos de pruebas.
- Identificar los recursos y configuraciones necesarias para llevar a cabo las pruebas.
- Describir y recomendar las estrategias de las pruebas a ser empleadas.
- Definir el cronograma de las pruebas.

1.2 Alcance Este procedimiento va a ser utilizado por los estudiantes del laboratorio de pruebas de liberación para realizar Pruebas de Seguridad a aplicaciones web.

2 Recursos

2.1 Selección de Roles

Para la aplicación de este procedimiento de Pruebas de Seguridad se realiza la elección de un equipo de trabajo seleccionado por el jefe de proyecto que será el encargado de revisar la realización de las pruebas, en compañía del jefe del módulo de calidad. Durante el procedimiento de pruebas es necesario asumir cuatro roles fundamentales por el equipo que desarrollará el trabajo, estos roles son:

- Administrador o Jefe de pruebas
- Diseñador de las pruebas
- Analista de Prueba
- Probador

Antes de la realización de las tareas por roles es importante saber las competencias básicas necesarias para seleccionar cada uno de estos roles. Estas son:

- Conocer cómo funciona el flujo de pruebas de la metodología RUP.
- Conocer a plenitud la aplicación a probar.
- Tener toda la documentación necesaria para aprender a usar la aplicación y dominarla antes de la planificación y realización de las pruebas.

Las competencias que tiene un Administrador o Jefe de pruebas es que tiene que saber cómo realizar una buena planificación de pruebas como es: conocer el alcance de las pruebas que se van a ejecutar, el propósito, saber repartir bien las tareas a los integrantes del equipo de desarrollo, evaluar el

progreso y la efectividad de las pruebas. Una de la competencia más importante es saber dirigir el proceso de pruebas con la mejor aptitud y sabiduría posible.

Diseñador de pruebas tiene que tener claro los procedimientos de las pruebas a ejecutar ya que las competencias de él son: saber identificar un caso de uso, saber diseñar los casos de pruebas que se van a utilizar, tener el dominio de la realización del diseño de las listas de chequeos a utilizar y definir la configuración del entorno de las pruebas donde se van aplicar.

El Analista de pruebas unas de las competencias principales que tiene que saber es llevar el control y verificar las pruebas las pruebas que están en ejecución.

Probador es el que tiene que saber cómo preparar y ejecutar las pruebas. Uno de los conocimientos que debe tener es saber realizar y conformar el documento de los errores encontrados al terminar el proceso de ejecución de pruebas.

La estrategia de la metodología basada en RUP, contempla la presencia de cuatro roles responsables de la aplicación de las pruebas al software. Los mismos serán ocupados de acuerdo a los conocimientos mínimos que deba poseer el individuo. Seguidamente se refieren dichos roles y responsabilidades:

Tabla de distribución de tareas por roles:

Rol	Responsabilidades
Administrador de Prueba	➢ Asegurar la planificación apropiada y dirección de los recursos de las pruebas. ➢ Evaluar el progreso y efectividad del proceso de pruebas. ➢ Evaluar resultados obtenidos.
Diseñador de Prueba	➢ Realizar el procedimiento de las pruebas, decidir los objetivos de prueba apropiados. ➢ Identificar, priorizar, seleccionar y describir los casos de pruebas y los procedimientos correspondientes. ➢ Identificar las técnicas apropiadas, herramientas y pautas para llevar a cabo las pruebas. ➢ Definir la configuración del entorno para realizar las pruebas.
Analista de Pruebas	

	➢ Verificar y llevar el control de la ejecución de las pruebas.
Probador	➢ Preparar y ejecutar las pruebas. ➢ Conformar documentación al culminar el proceso.

Tabla 3: Roles y Responsabilidades.

Escenarios de Pruebas

Se colocan las especificaciones de hardware necesarias para las pruebas y una imagen que represente la configuración, puede ser un diagrama de despliegue o una imagen hecha en Visio, este tópico es responsabilidad del equipo de desarrollo.

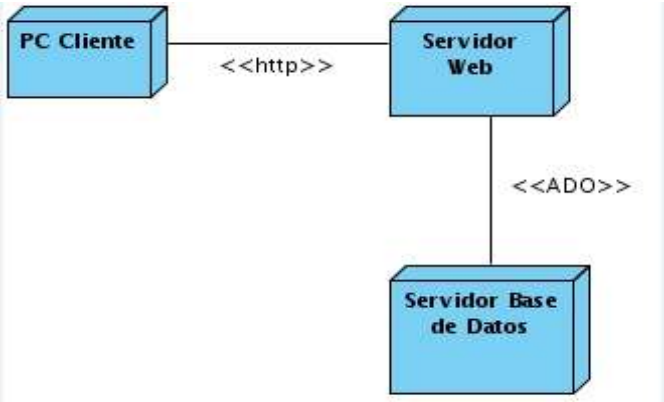

Figura 8: Diagrama de Despligue.

ANEXOS

Tabla 4: Recursos del sistema

Recursos	Tipo
Servidores de Base de datos	Características del software: Mysql 5.0.2. Características de hardware. Procesadores a 1.6 GHz 1 GB de Memoria RAM. Dos discos duros de 72GB a 7200 rpm en RAID1. Tarjeta de Red de 10/100/1000 Giga bit.
Servidor de Aplicación	Características de software: Apache 6.2.1. Características de hardware: 2 Procesadores a 3 GHz 8 GB de Memoria RAM. Dos discos duros de 72GB a 7200 rpm en RAID1. Tarjeta de Red de 10/100/1000 Giga bit.
PC Clientes para prueba	Características de hardware: Pentium III 450 MHz o superior. HDD 40 GB. Tarjeta de Red Fast Ethernet 100 Mbps. 256 MB de Memoria RAM. Características del software: SO: Linux, Ubuntu 10.4 con soporte para navegadores web Navegador Web compatible con IE 4.0
Requerimientos especiales	Algún periférico, necesario para las pruebas: impresora, escáner, cámara digital.
Red o subred	Tipo de red e: inalámbrica, TCP/IP.

ANEXOS

Descripción del flujo de trabajo.

El flujo de trabajo se inicia cuando los especialistas comienzan a diseñar las listas de chequeo y los casos de pruebas para su posterior utilización. Luego si todas las condiciones están creadas se pasa a la 1ra iteración de pruebas, se realizan las pruebas. En la medida que detecten errores, molestias o incomodidades en el trabajo con el producto, estas serán anotadas. Al finalizar el día estas no conformidades son revisadas por el especialista de pruebas. Al concluir la iteración de prueba se realizará el Informe Final de Resultados, atendiendo a los informes diarios.

Una vez aprobado es entregado al líder del proyecto para que este comience a ejecutar los arreglos acordados. En la figura 8 se muestra de forma general el flujo de trabajo.

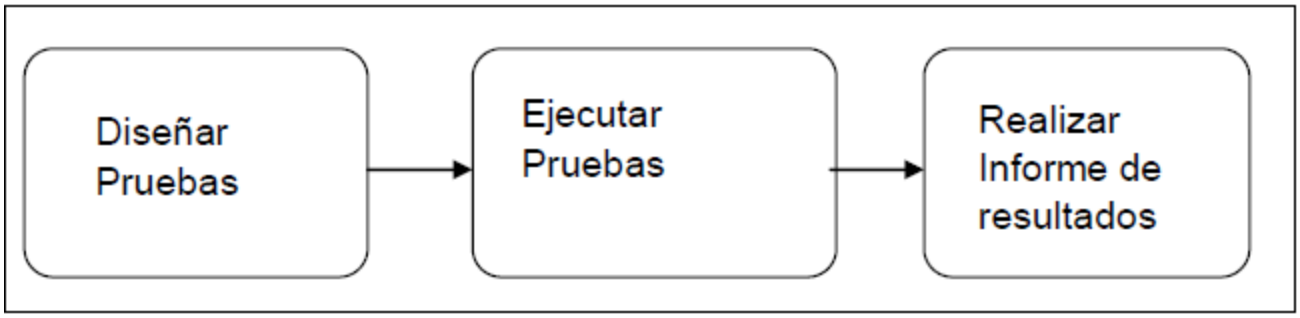

Figura 9: Descripción del Flujo del trabajo.

Descripción de las estrategias y tipos de pruebas

Todas las pruebas se realizarán de forma manual o utilizando herramientas en dependencia del tipo.

- **Primera Fase:**

1. Organización del escenario de pruebas, capacitación del equipo de pruebas, Diseño de los Casos de Prueba y elaboración de Listas de Chequeo.

- **Segunda Fase:**

1. Realización de la categoría de las pruebas de recopilación de la Información: su objetivo es verificar si la aplicación brinda datos sensibles que puedan ser utilizados por cualquier atacante.
2. Realización de la categoría de pruebas Comprobación de las reglas del Negocio: su objetivo es verificar las reglas del negocio definidas en la aplicación.
3. Realización de la categoría de pruebas Comprobación de la Autenticación: Su objetivo es poner a prueba el sistema de autenticación de la aplicación.

4. Realización de la categoría de las Pruebas de Validación de Datos: Su objetivo es verificar que todas las entradas de datos estén validadas.

Anexos2 Lista de chequeo para la Recopilación de Información:

Tabla 5: Lista de chequeo para la Recopilación de Información

Indicadores a evaluar	Descripción	Resultado Esperado	Resultado real	Herramienta
1.1 ¿Se identifican más aplicaciones web instaladas en el servidor web?	Es el proceso destinado a identificar aplicaciones web instaladas en una Infraestructura dada.	Ver las aplicaciones que están instaladas en el servidor.	No existe ningún proceso para identificar aplicaciones web instaladas en una infraestructura dada.	Netcraft
1.2 ¿Se identifican todos los puertos abiertos en el IP del servidor y los servicios asociados a esos puertos?	Es el proceso para encontrar que aplicaciones específicas se encuentran instaladas en un servidor a partir de todos los puertos abiertos en el IP del servidor y los servicios asociados.	Ver los puertos abiertos en el IP del servidor.	No se puede identificar bien claro los puertos abiertos en el IP del servidor y los servicios asociados a esos puertos.	Netcraft
1.3 ¿El código de error de la aplicación muestra información sobre el sistema operativo o base de datos del	Durante la configuración de un servidor de base de datos, muchos administradores de BBDD no toman en consideración adecuadamente la	Ver el código error de la aplicación.	No tiene una adecuada configuración de un servidor de base de datos, no se toma en consideración una adecuada	Netcraft

servidor web?	seguridad del componente receptor de escucha de la base de datos.		seguridad del componente receptor de escucha de la base de datos.	

Tabla 6: Lista de chequeo para la Autenticación

Indicadores a evaluar	Descripción	Resultado esperado	Resultado Real	Herramientas
¿Puede obtenerse mediante el ataque de fuerza bruta el usuario y la contraseña de un usuario privilegiado en la aplicación	Consiste en averiguar el usuario y contraseña válidos de un individuo registrad en el sistema.	Obtener el usuario y la contraseña de un usuario privilegiado en la aplicación.	No se puede conseguir fácilmente el usuario y la contraseña, ya que el sistema cumple con las normas de Autenticación	Brutus
¿El sistema envía la contraseña por correo sin hacerle alguna pregunta?	Enviar la contraseña (o un enlace al reset de la contraseña) a la dirección de email del usuario sin realizar primero una pregunta secreta significa confiar al 100% en la seguridad de esa dirección de email, algo que no es	Enviar la contraseña por correo mediante el sistema.	El sistema no envía ninguna contraseña por correo	Brutus

	adecuado si la aplicación requiere de un nivel de Seguridad alto.			
¿Se le realizan al usuario dos o más preguntas?	A menudo un sistema de reset ofrece la elección entre varias preguntas; esta es una buena señal para el posible atacante, porque le ofrece opciones.	Realizar más de dos preguntas.	El sistema no ofrece ninguna pregunta a los usuarios y no presenta opciones para realizarlas	Brutus
¿Las preguntas pueden tener varias respuestas?	Alternativamente (o adicionalmente), la aplicación podría pedir al usuario responder a una o más "preguntas secretas", que son escogidas por el usuario entre un conjunto de preguntas posibles.	Dar varias respuestas a las preguntas.	El sistema no ofrece ninguna pregunta a los usuarios, por lo que presenta una buena señal para el posible atacante, porque le ofrece opciones.	Brutus

Anexo 3 Documentación e informe de los errores:

Tabla 7: Pruebas al Sistema del Módulo de Programación

Elemento	No	No conformidad	Aspecto correspondiente	Etapa de detección	Clasificación	Estado NC	Respuesta de Equipo Desarrollo
SC 1: Adicionar espacio de Radio.	1	En el EC 1.1: Adicionar espacio de Radio correctamente. No se verifica que se hayan llenado todos los campos correctamente, ya que el campo de nombre acepta números u otro tipo de caracteres	Debe arreglarse el campo Nombre que solo acepte textos.	Prueba	S: Significativa	PD: Pendiente. 10/05/2011	
SC 1: Adicionar espacio de Radio.	2	En el EC 1.2: Adicionar espacio de Radio con campos vacíos. No se muestra un mensaje de error diciendo que se deben llenar todos los campos.	Mostrar un mensaje que diga: “Se debe llenar todos los campos”.	Prueba	R: Recomen daciones	PD: Pendiente. 10/05/2011	

SC 2: Modificar espacio de Radio	3	En el EC 2.1: Modificar espacio de Radio exitosamente. Él sistema no muestra ningún mensaje diciendo que el espacio fue modificado correctamente y no modifica los nuevos datos, el campo de los días no lo actualiza con los nuevos valores	Mostrar mensaje diciendo: “El espacio fue modificado correctamente”. Debe actualizarse el campo de los Días.	Prueba	S: Significativa	PD: Pendiente. 10/05/2011	
SC 2: Modificar espacio de Radio	4	En el EC 2.3: Modificar espacio de Radio con campos vacios. Él sistema no muestra ningún mensaje de error diciendo que todos los campos deben de ser llenados	Mostrar mensaje diciendo: “Todos los campos deben de ser llenados”.	Prueba	R: Recomendaciones	PD: Pendiente. 10/05/2011	

SC 3: Eliminar Espacio de Radio.	5	En el EC 3.1: Eliminar Espacio de Radio correctamente. El sistema no muestra ningún mensaje diciendo que el espacio de radio fue eliminado correctamente	Mostrar mensaje diciendo: “El espacio de radio fue eliminado correctamente”.	Prueba	S: Significativa	PD: Pendiente. 10/05/2011	
SC 3: Eliminar Espacio de Radio.	6	EC 3.2: Cancelar Eliminar Espacio de Radio. El sistema no vuelve a la vista anterior, ya que no existe la opción cancelar en esa interfaz	Debe tener una opción que diga Cancelar para volver a la página anterior.	Prueba	R: Recomendaciones	PD: Pendiente. 10/05/2011	
SC 1: Adicionar espacio de TV.	7	EC 1.1: Adicionar espacio de TV satisfactoriamente. En el sistema el campo Nombre	Debe permitir que el campo Nombre solo acepte texto y ningún carácter extraño.	Prueba	S: Significativa	PD: Pendiente. 10/05/2011	

		acepta cualquier tipo de caracteres					
SC 1: Adicionar espacio de TV	8	EC 1.2: Adicionar espacio de TV con fallo. El sistema no muestra ningún mensaje de error diciendo que se deben llenar todos los campos.	Mostrar un mensaje diciendo: “Se deben llenar todos los campos”.	Prueba	R: Recomendaciones	PD: Pendiente. 10/05/2011	
SC 2: Modificar espacio de TV	9	EC 2.1: Modificar espacio de TV satisfactoriamente. El sistema no muestra ningún mensaje de error señalando que debe seleccionar un espacio de TV, ni diciendo que todos los campos deben de ser llenados	Mostrar mensajes diciendo: “Por favor selecciona un espacio de TV” y otro mensaje: “Deben de ser llenados todos los campos”.	Prueba	S: Significativa	PD: Pendiente. 10/05/2011	

SC 3: Eliminar espacio de TV	10	EC 3.2: Eliminar espacio de TV con fallo. El sistema no vuelve a la lista anterior ya que no existe ninguna opción que sea de Cancelar	Debe existir una opción que sea Cancelar para que vaya a la lista anterior.	Prueba	R: Recomendaciones	PD: Pendiente. 10/05/2011	

GLOSARIO DE TÉRMINO

PTARTV: Plataforma de Transmisión Abierta de Radio y Televisión.

GEYSED: Geoinformática y Señales Digitales.

TIC: Tecnologías de la Información y la Comunicación.

IP (Protocolo de Internet): la forma estándar de identificar un equipo que está conectado a Internet.

LIPS: Laboratorio Industrial de Pruebas de Software.

DTU: Dirección de Televisión Universitaria.

SQA (Aseguramiento de la Calidad): conjunto de actividades planificadas y sistemáticas necesarias para proporcionar confianza en que el producto de software satisfará los requisitos dados de calidad.

Calidad: Calidad de software. Satisfacción de las necesidades del usuario.

Caso de prueba: Conjunto de entrada, condiciones de ejecución y resultados esperados desarrollados para un objetivo particular, por ejemplo, ejercitar un camino concreto de un programa o verificar el cumplimiento de un determinado requisito. También se puede referir a la documentación en la que se describen las entradas, condiciones y salidas de un caso de prueba.

Fases: son los pasos en que se descomponen las metodologías. Cada fase puede o no estar subordinada a otra fase pudiendo existir entre ellas relaciones de dependencia.

Herramienta: Subprograma o módulo encargado de funciones especificas y afines entre sí para realizar una tarea. Una aplicación o programa puede contar con múltiples herramientas a su disposición. Por ejemplo, corrector ortográfico puede ser una herramienta en una aplicación para redactar documentos, pero no es una aplicación en sí misma.

Prueba: Prueba de software. Ejecución de un sistema bajo condiciones especificas, se observan y se analizan los resultados realizándose una evaluación de los mismos.

Procedimiento: forma específica de llevar a cabo la actividad. En muchos casos los procedimientos se expresan en documentos que contienen el objeto y el campo de aplicación de una actividad; que debe hacerse y quien debe hacerlo, cuando, donde y como se debe de llevar a cabo; que materiales, equipos y documentos deben utilizarse; como debe controlarse y registrarse.

Proyecto: Elemento organizativo a través del cual se gestiona el desarrollo del software, el resultado de un proyecto es una versión del producto.

GLOSARIO DE TÉRMINO

Proceso: es un conjunto de tareas, actividades o acciones interrelacionadas entre sí que, a partir de una o varias entradas de información, materiales o de salidas de otros procesos, dan lugar a una o varias salidas también de materiales (productos) o información con un valor añadido.

Roles: papel que ejerce un actor en una actividad o proyecto.

Requerimientos: una condición o capacidad que debe estar presente en el sistema o componentes del sistema para satisfacer un contrato estándar, especiación u otro documento formal.

Recursos: son todos aquellos elementos necesarios, tanto tangibles como intangibles, para que una organización cumpla con sus objetivos.

Sistema: conjunto de elementos relacionados que interactúan entre sí para lograr un fin determinado.

Status: estado.

HTTP (Protocolo de Transferencia de Hipertexto): el método mediante el cual se transfieren las páginas web a un ordenador.

BBDD (Base de Datos): consiste en un almacén de datos relacionados con diferentes modos de organización.

Pruebas de Seguridad: es el término profesional para evaluar un tipo de seguridad.

SSL (Protocolo de Transacciones Seguras): permite a sus clientes conectarse al sitio web de su propiedad y establecer un canal de comunicación seguro.

TLS (Seguridad para Capa de Transporte): provee el apoyo de criptografía, de canales de transmisión de datos seguros, para la protección, confidencialidad y autenticación de la información transmitida.

Exploits: explotar.

SQL (Lenguaje de Consulta Estructurada): es un estándar en el lenguaje de acceso a bases de datos.

Hash (picar y mezclar): una función o método para generar claves o llaves que representen de manera casi unívoca a un documento.

XML (Lenguaje de Marcas Extensible): es un lenguaje abierto que se ha diseñado y optimizado para mejorar la funcionalidad de la Web.

Scripting: secuencias de comandos.

GLOSARIO DE TÉRMINO

ASP (Servidor de Páginas Activas): es una tecnología dinámica funcionando del lado del servidor

PHP (Personal Home Page): es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas.

DNS (Sistema de Nombres de Dominio): su utilidad principal es la búsqueda de direcciones IP de sistemas centrales basándose en los nombres de estos.

URL (Localizador Uniforme de Recursos): la dirección Internet de un recurso Web (página, elementos incorporados.) entendido e interpretado por los navegadores.

FTP (Protocolo de Transferencia de Archivos): su misión es permitir a los usuarios recibir y enviar ficheros de todas las máquinas que sean servidores FTP.

SMB (Bloque de Mensajes de Servidor): es usado por las redes de Microsoft Windows para acceder a sistemas de archivos de otras máquinas.

Telnet: es el acrónimo de Telecommunication Network. Se trata del nombre de un protocolo de red que se utiliza para acceder a una computadora y manejarla de forma remota.