

Tipo de artículo: Artículo original
Temática: Soluciones Informáticas
Recibido: 10/05/16 | Aceptado: 18/06/16

Sistema para la auditoría y control de los Activos Fijos Tangibles

System for audit and control of Fixed Assets Tangible

Omar Mar Cornelio^{1*}, Barbara Bron Fonseca¹, Yanisleidis González Caballeros¹

¹ Facultad 6, Universidad de las Ciencias Informáticas. Carretera a San Antonio de los Baños Km 2 ½ reparto Torrens Boyeros. {omarmar, bbron, yanisleidydis}@uci.cu

* Autor para correspondencia: omarmar@uci.cu

Resumen

Como parte del sistema de control interno, se establece como norma que 10% de los Activos Fijos Tangible AFT sean auditados mensualmente, representa una tarea primordial cada mes en las organizaciones. En la Facultad 6 de la Universidad de las Ciencias Informática UCI, se realiza a través del testeado manual de los medios contra un inventario previamente impreso, lo que genera un proceso engorroso e ineficiente con altos gastos de recurso humanos y materiales. El presente trabajo describe una solución a dicha problemática con la implementación de una aplicación Web codificada con PHP sobre Entorno de Desarrollo Integrado NetBeans, con framework de desarrollo Symfony, gestor de base de datos PostgreSQL y un sistema de adquisición de datos a través de un cliente desktop que permite obtener los medios físicos de un local mediante etiquetas en código barras. El sistema genera además un conjunto de reportes que contribuyen a elevar el control de los AFT y su proceso de identificación.

Palabras clave: Activos Fijos Tangibles; auditoría; control; códigos barra.

Abstract

As part of the internal control system, established a rule that 10% of Tangible Fixed Assets are audited monthly AFT represents a major task each month in organizations. In 6 of the School of Computing Sciences University UCI is done through manual means testing against a pre-printed inventory, creating a cumbersome and inefficient process with high human resource costs and materials. This paper describes a solution to this problem by implementing a coded Web application with PHP on NetBeans Integrated Development Environment with Symfony development

framework, manager PostgreSQL database and a data acquisition system through a client desktop that allows for the physical media via a local bar code labels. The system also generates a set of reports which contributes to increase the AFT control and identification process.

Keywords: *Tangible Fixed Assets; audit; control; bar codes.*

Introducción

Las Tecnologías de la Información y las Comunicaciones (TIC) evolucionan de manera que el desarrollo social en el planeta no está a la altura del actual desarrollo científico en el ámbito de la informática y las comunicaciones. El ritmo de cambio acelerado que ha vivido la sociedad, ha obligado a las organizaciones, instituciones y empresas a asumir un proceso de cambio y control constante, en una búsqueda permanente de mejorar su actividad (SÁNCHEZ and VALDÉS 2009), (GARAY 2015).

Las grandes evoluciones económicas y sociales de los últimos tiempos han conformado un nuevo modelo socio técnico, impulsado principalmente por un sorprendente desarrollo tecnológico. Estos eventos han provocado un obligatorio cambio en el propio contenido de la organización de la información en la gestión empresarial y su control (RODRÍGUEZ 2012), (ROSSO and SORIANO 2008).

Debido a la importancia que se le amerita los temas de control, en agosto de 2009 la Asamblea Nacional del Poder Popular, crea la Contraloría General de la República, la que tiene entre sus funciones establecidas, según lo señalado en el Artículo 31 inciso 1), normar, supervisar, evaluar los sistemas de control interno; manifestar las recomendaciones necesarias para su mejoramiento y perfeccionamiento continuo (CONTRALORÍA 2009).

El sistema de control interno establece que se deben establecer un proceso que permita auditar al menos el 10 % de los Activos Fijos Tangibles AFT mensuales en una organización. Durante dicho proceso se comparan las existencias físicas con los registros contables para verificar su coincidencia y prevenir riesgos de sustracción, despilfarro, uso indebido u otras irregularidades (VICTORIA 2005).

Las auditorías son procesos informacionales, desde las económicas financieras hasta las de conocimiento, de ahí que pueden ser tratadas de manera conjunta si se poseen las herramientas y metodologías apropiadas, ellas interactúan con la estrategia de la organización, tomando los elementos primarios para comenzar la revisión de los procesos (MAR 2014), los recursos y la propia estrategia, y una vez cumplidos sus objetivos, enriquecen esta última, formulan recomendaciones, acciones correctivas, cronogramas y planes de implementación, seguimiento y control (GONZÁLEZ, M 2011), (JARAMILLO *et al.* 2012).

Los Activos Fijos Tangibles (AFT), se definen como los bienes que una empresa utiliza de manera continua en el curso normal de sus operaciones; representan al conjunto de servicios que se recibirán en el futuro a lo largo de la vida útil de un bien adquirido se caracterizan por ser físicamente tangible, tener una vida útil relativamente larga, ser utilizado en la producción o comercialización de bienes y servicios, para ser alquilado a terceros, o para fines administrativos (MILLONES 2011), (GONZÁLEZ, J and MAR 2015). El medio existe con la intención de ser usado en las operaciones de la empresa de manera continua y no para ser destinado a la venta en el curso normal del negocio (CUNHA *et al.* 2010), (SUTZ 2000).

La Facultad 6 cuenta con 620 estaciones de trabajo destinadas a los Procesos Docentes - Productivos para una aproximación de 2408 AFT lo que provoca: Insuficiencia para generar un control de los AFT, imposibilidad de cumplir con la cantidad de medios que representa el 10% de los AFT de la organización e incapacidad para generar informes resultantes de las auditorías realizadas.

Teniendo en cuenta la insuficiente gestión que se describe del proceso de auditoría de los AFT, se decide implementar un sistema informático que permita automatizar el proceso de auditoría y control de los AFT.

Materiales y métodos o Metodología computacional

El sistema para la auditoría y control de los AFT de la Facultad 6 de la UCI en su Versión 1.0, está destinado a soportar el proceso de gestión de auditoría y control. El mismo, cuenta con varias funcionalidades que rigen el funcionamiento del sistema dentro de los que se encuentran:

Autenticar usuario, Gestionar usuario, Gestionar organización, Gestionar centro de costo, Gestionar área, Gestionar activo fijo, Gestionar tipo de medio, Gestionar activos, Generar código de barra, Cargar documento Excel, Generar acta de responsabilidad material, Realizar movimiento de medios, Capturar datos de auditoría, Realizar auditoría, además cuenta con cliente encargado de realizar auditorías.

Para la gestión del sistema se definen tres actores fundamentales, administrador con el máximo nivel de privilegio encargado de realizar las actividades de administración, especialista encargado del trabajo con las diferentes funcionalidades del sistema y el rol auditor encargado de realizar auditorías de los activos a los diferentes locales.

Se asume Open Up por ser una metodología de modelado, extensible, dirigido a gestión y desarrollo de proyectos de software basados en desarrollo iterativo, ágil e incremental apropiado para proyectos pequeños o de bajos recursos, aplicable a un conjunto amplio de plataformas de desarrollo. Basada en la metodología de Proceso Unificado de Rational (Rational Unified Process) RUP (FOUNDATION. 2010). Es el subconjunto de esta última que contiene el conjunto mínimo de prácticas que ayudan a un equipo de desarrollo de software a realizar un producto de alta calidad.

Utiliza un punto de vista pragmático, una filosofía ágil que se centraliza en la naturaleza colaborativa del proceso de desarrollo del software.

Es un lenguaje de modelado visual que se utiliza para especificar, visualizar, construir y documentar artefactos de un sistema de software, es Lenguaje Unificado de Modelado (Unified Modeling Language) UML 2.0 por incluir aspectos conceptuales tales como procesos de negocio, funciones del sistema, aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables (YIKUN and KAKOLA 2014).

Es un lenguaje gráfico para:

Visualizar: Permite expresar de una forma gráfica un sistema de forma que otro lo puede entender. Especificar: Permite especificar cuáles son las características de un sistema antes de su construcción. Construir: A partir de los modelos especificados se pueden construir los sistemas diseñados. Documentar: Los propios elementos gráficos sirven como documentación del sistema desarrollado que pueden servir para su futura revisión.

Como herramienta de modelado se utilizó Visual Paradigm en su versión 8.0, porque soporta el ciclo de vida completo del desarrollo de software: Análisis y Diseño orientados a objetos, Construcción, Pruebas y Despliegue.

El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. La herramienta UML CASE también proporciona abundantes tutoriales de UML, demostraciones interactivas de UML y proyectos UML (JACOBSON 2000).

Desde el comienzo de Internet, surgieron diferentes demandas por los usuarios y se dieron soluciones mediante lenguajes estáticos. Con el transcurso del tiempo, las tecnologías se desarrollaron y surgieron nuevos problemas a dar solución (DART and ELLISON 2008). Esto dio lugar a desarrollar lenguajes de programación para la Web dinámica, que permitieran interactuar con los usuarios y utilizaran sistemas de bases de datos. Se utiliza PHP en su versión 5.2

Es un lenguaje orientado a objeto, lo que permite la reutilización de código y un rápido desarrollo. Posee compatibilidad con las bases de datos más comunes, como MySQL, Oracle y PostgreSQL, entre otros. Proporciona soporte para diferentes protocolos de comunicación conocidos entre los cuales se tienen:

HTTP (Protocolo de Transferencia de Hipertexto), IMAP (Protocolo de Acceso a Mensajes de Internet), FTP (Protocolo de Transferencia de Archivos), LDAP (Protocolo Ligero de Acceso a Directorios), entre otros.

Con los framework de desarrollo, se pueden encontrar soportes de programas, bibliotecas, un lenguaje interpretado, entre otros software, para facilitar el desarrollo, unir las diferentes estructuras y componentes de un proyecto. Se selecciona Symfony en su Versión 1.4.10 diseñado para aplicaciones Web desarrollado en PHP5. Acelera la creación, mantenimiento de las aplicaciones Web, y permite la reutilización de código, lo cual agiliza el desarrollo de la

aplicación.

Su diseño permite optimizar las aplicaciones y mantener una organización de la misma. Permite reducir el tiempo de desarrollo de una aplicación pues proporciona varias herramientas y clases las cuales pueden ser reutilizadas por los desarrolladores. Permite el desarrollo por capas, ya que se basa en el patrón arquitectónico Modelo Vista Controlador (MVC). Es compatible con la mayoría de los gestores de base de datos como PostgreSQL (POTENCIER and ZANINOTTO 2010).

Para desarrollar, normalmente solo es necesario un editor de texto, un intérprete o compilador y una terminal de líneas de comando, pero siempre es más rápido y fácil si se usan algunas herramientas o un Entorno de Desarrollo Integrado (IDE). Esto simplifica el trabajo y ahorra tiempo de desarrollo. Se selecciona NetBeans en su versión 6.8 por ser un producto libre y gratuito sin restricciones de uso, dispone de soporte para crear interfaces gráficas de forma visual, desarrollo de aplicaciones Web, control de versiones, colaboración entre varias personas, creación de aplicaciones compatibles con teléfonos móviles y resaltados de sintaxis.

Se selecciona PostgreSQL por ser un sistema de gestión de base de datos relacional orientada a objetos, de software libre, publicado bajo la licencia de Distribución de Software Berkeley (Berkeley Software Distribution) BSD, con características de los mejores sistemas de bases de datos comerciales. PostgreSQL es libre y su código fuente completo está disponible.

Para el diseño de las interfaces del sistema se propone Ext JS en su versión 3.1 por ser un framework JavaScript del lado del cliente, que permite crear aplicaciones Web con interfaces muy similares a la de una aplicación de escritorio. Entre sus características se pueden encontrar que es multiplataforma, completamente orientado a objetos, y posee múltiples posibilidades para el trabajo con las validaciones y manejo de errores en el cliente.

Arquitectura del sistema

Los patrones de arquitectura de diseño de software brindan soluciones a problemas de arquitectura en ingeniería de software. Describen los elementos y el tipo de relación que tienen junto con un conjunto de restricciones sobre cómo pueden ser usados. Expresa un esquema de organización estructural esencial para un sistema, que consta de subsistemas, sus responsabilidades e interrelaciones. Tienen una escalera más grande en comparación con los patrones de diseño (MAR *et al.* 2016).

En el sistema planteado se implementa mediante plantillas que describen la estructura y organización de los casos de uso con se expresa en:

Control de acceso: Permite garantizar un correcto acceso a datos y funcionalidad limitado al rol del usuario. Desde un punto de vista arquitectónico, acceder a determinadas partes del software debe tener un riguroso control.

Concurrencia: Permite manejar múltiples tareas de forma que simule el paralelismo. **Distribución:** Forma en que los subsistemas o componentes se comunican con los otros en un entorno distribuido.

Persistencia: Los datos persistentes son almacenados en bases de datos o archivos y pueden ser leídos o modificados por otros procesos más adelante.

El Modelo Vista Controlador (MVC) es un patrón para el desarrollo del software que se basa en separar los datos (por un lado), la interfaz del usuario (por otro) y la lógica interna (por un último lado). Es mayormente usado en aplicaciones Web, dónde la vista es la página HTML, el modelo es el Sistema de Gestión de Base de Datos y la lógica interna, y el controlador es el responsable de recibir los eventos y darles solución (SALAS and CERÓN 2014).

El Modelo: Es la representación de la información en el sistema. Trabaja junto a la vista para mostrar la información al usuario y es accedido por el controlador para añadir, eliminar, consultar o actualizar datos.

La Vista: Presenta al modelo en un formato adecuado para que el usuario pueda interactuar con él, casi siempre es la interfaz de usuario.

El Controlador: Es el elemento más abstracto. Recibe, trata y responde los eventos enviados por el usuario o por la propia aplicación. Interactúa tanto con el modelo como con la vista. En la Figura 1, se puede observar el funcionamiento de un sistema implementado que usa el patrón arquitectónico Modelo Vista Controlador (MVC).

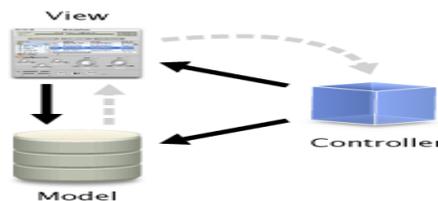


Figura. 1 Representación del Patrón Arquitectónico Modelo Vista Controlador.

A través del diagrama de componente referido en la Figura 2, se muestra la estructura general del sistema y el comportamiento del servicio que estos proporcionan y utilizan a través de las interfaces. Se describen los elementos físicos, las organizaciones y dependencias lógicas entre componentes de software así como las relaciones de dependencias que se utilizan para indicar que un componente se refiere a los servicios ofrecidos por otro componente.

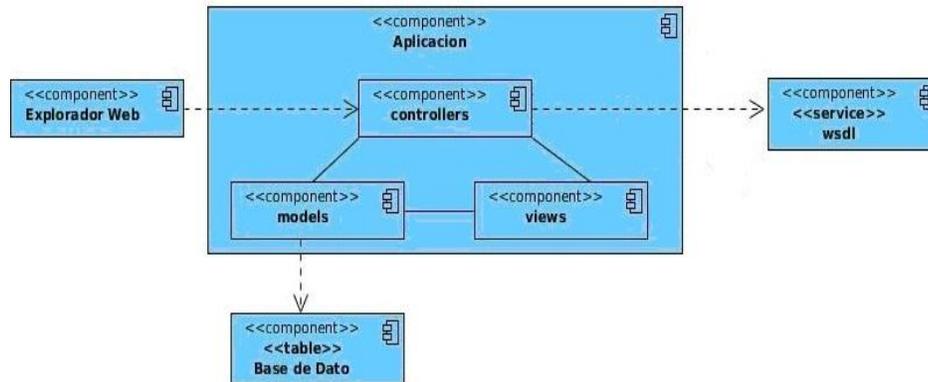


Figura. 2 Diagrama de Componentes de la Aplicación.

Se concreta la forma de acceso a los datos, la interacción con los diferentes servicios Web con que cuenta la institución para devolver los datos de una persona.

Para el correcto funcionamiento del sistema propuesto, es necesario garantizar como requisitos de Software para el servidor un Sistema Operativo Linux, Windows o Unix. Servidor Web Apache 2.2.4 o superior, Servidor de Bases de Datos PostgreSQL 9.1 o superior, PHP 5.2.3 o superior. Para la correcta visualización del sistema por las estaciones clientes se requiere Sistema operativo Linux o Windows, un navegador Internet Explorer (versión 8 o superior) y Mozilla (versión 8 o superior). Para el módulo de escritorio es necesario Net framework 2.0 o superior y un lector de código de barras.

Dentro de los requerimientos de hardware definidos para garantizar el correcto funcionamiento, el servidor por su parte como requisitos mínimos debe tener un microprocesador Pentium IV, una memoria operativa de 1.0 Gigabytes (Gb), una capacidad de almacenamiento de disco de 80 Gb y una tarjeta de red de 100 Megabytes (Mb), las estaciones cliente deberán tener como requisitos mínimos 256 Mb de memoria operativa con un Pentium IV o superior y una tarjeta de red de 10 Mb o superior.

Resultados y discusión

A través del diagrama de despliegue Figura 3, se muestra la configuración de los nodos que participan en la ejecución de los componentes que residen en ellos. Describen la arquitectura física de la aplicación durante la ejecución, en términos de: procesadores, dispositivos, componentes de software. La vista de despliegue representa la disposición de las instancias de componentes de ejecución en instancias de nodos conectados por enlaces de comunicación.

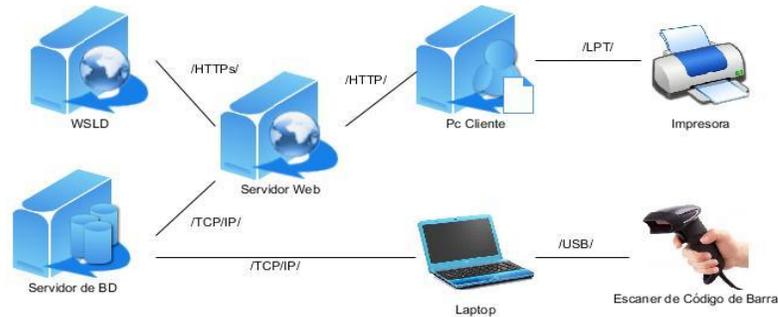


Figura 3. Diagrama de Despliegue de la Aplicación.

A continuación se describen las principales funcionalidades de la propuesta presentada:

Para garantizar el proceso de gestión de los activos que son manejados por el sistema, se cuenta con la funcionalidad gestionar activos. La Figura 4, muestra una vista desde la cual el usuario autorizado puede, agregar o quitar un medio dado un área, generar etiquetas para uno o varios medios. El proceso de gestión sobre los activos, representa la base sobre las acciones de auditoría y control.

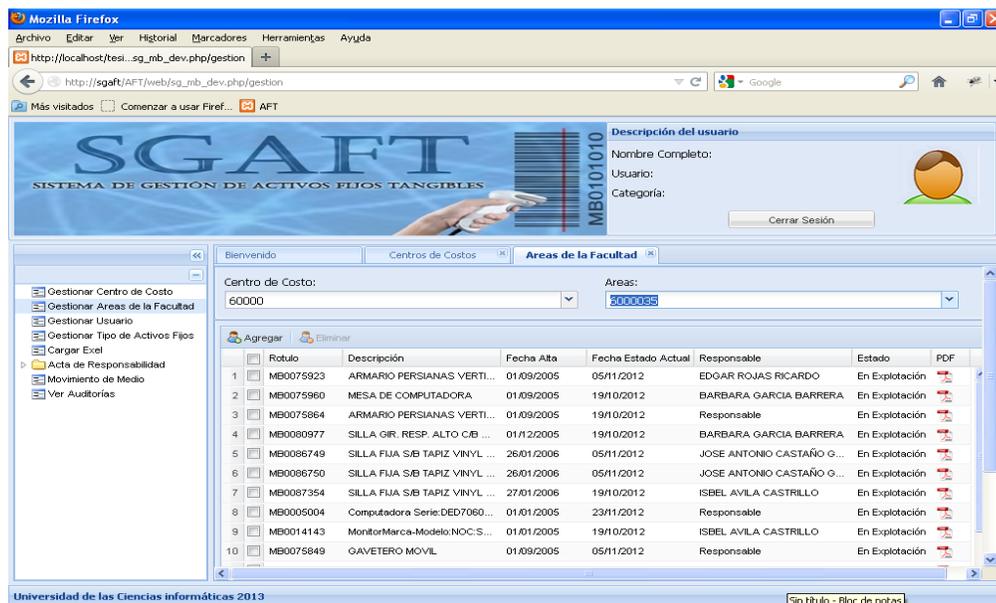


Figura 4. Vista gestionar Activos de un Área.

Una vez cargado los datos que son necesarios para el proceso de auditoría en la base de datos, es posible realizar el proceso de control. Para soportar el proceso de control, el sistema cuenta con un módulo realizar auditorías. La Figura 5. Visualiza la pantalla inicial del cliente auditoria. Desde la presente interface, es seleccionado por el usuario auditor el centro de costo y el área de responsabilidad que será objeto de revisión.



Figura 5. Cliente auditoría, solicitar datos.

Una vez determinado el área y centro objeto de auditoría, para realizar la captura de datos durante el proceso de revisión, son escaneados los rótulos de los activos codificados previamente mediante etiquetas con los valores barras correspondientes. La Figura 6, muestra la vista realizar control del cliente auditoria, donde son mostrados todos los activos correspondientes al área auditada. Se visualiza un menú con los listados de los medios sobrantes y faltantes que van siendo revisados durante el proceso. El sistema brinda además la posibilidad de generar un informe una vez concluido el proceso.



Figura 6. Cliente auditoría, realizar control.

Dentro de las características obtenidas se encuentran los requisitos de Usabilidad, Seguridad, Confidencialidad, Integridad, Disponibilidad, Soporte, Interfaz; los cuales son descritos a continuación:

Usabilidad: El sistema puede ser utilizado por cualquier persona que tenga al menos conocimientos básicos en el manejo de la computadora, navegación y exploración de los sitios Web en sentido general, las operaciones se realizan con bajo nivel de complejidad.

Seguridad: Es el requerimiento más complejo y difícil de garantizar, a su vez permitirá garantizar que la aplicación será utilizada correctamente por cada usuario según sus niveles permitidos.

Confidencialidad: Se debe tratar el manejo de permisos de forma que solamente se acceda a la información autorizada de acuerdo a los niveles de permisos que debe tener cada usuario del sistema. De esta forma se garantiza que la información no sea expuesta a personal indebido.

Integridad: Se debe tratar el manejo de la información de forma tal, que la información no sea modificada por personal ajeno. Esto evitará alteraciones en los resultados planteados en la documentación.

Disponibilidad: Se deberá garantizar el acceso pleno de cada usuario con facultades para el uso de la aplicación las 24 horas del día.

Respaldo y recuperación de base de datos: El sistema debe permitir la realización de copias de respaldo hacia otros dispositivos de almacenamiento para prevenir pérdida de información. Esta base de datos debe ser posible de recuperar mediante las copias de respaldo previamente creadas.

Soporte: Debe presentar un diseño que sea capaz de la realización de pruebas que permitan la explotación de la aplicación de forma eficiente. Se incluye la posibilidad de brindar asistencia técnica que permita la solución de problemas en tiempo real de ejecución que garantice la solución de fallas que pueda presentar. Debe ser garantizada su adaptabilidad y compatibilidad con los distintos sistemas operativos.

Interfaz: El diseño de la interfaz visual debe ser amigable para los usuarios que interactúan con la aplicación, de forma tal que permita el fácil entendimiento de las funcionalidades que brinda, además de poseer colores amigables y refrescantes para una mejor interacción entre el usuario y la aplicación.

El diseño de la interfaz permitirá mostrar mensajes para la guía de usuarios en caso de errores en entradas inválidas de los datos o de confirmación de realización de actividades.

Los resultados obtenidos con la aplicación del sistema para la auditoría y control de los AFT, logran satisfacer el objetivo planteado sobre la base de la problemática existente. Con la utilización de la herramienta informática como soporte al proceso se contribuye eficientemente con el proceso objeto de estudio.

Los datos almacenados durante el proceso de revisión posibilitan investigar tendencias y comportamientos históricos para la toma de decisiones ante una situación de descontrol.

La herramienta desarrollada aporta para práctica cotidiana en las labores diarias de directivos y especialistas que laboran en la gestión de los AFT, los hallazgos e informaciones necesarias para elevar los mecanismos de control de la organización lo que tributa con impacto, novedad y actualidad científica para la universidad.

Si bien es cierto que un sistema de información posibilita la gestión y contribuye a la toma de decisiones en las organizaciones, es válido resaltar que requiere de sistematicidad en su implementación. La decisión de aplicar las normativas y procedimientos vigentes depende de personas, quienes poseen la responsabilidad de mantener visibles y

actualizadas las etiquetas barras tomadas como referencia en el proceso de adquisición de datos.

A pesar de los numerosos esfuerzos realizados por especialistas y directivos en su labor educativa diaria, se pudo demostrar que aún existen diferencias identificándose un conjunto de violaciones, las cuales con la gestión oportuna de los directivos fueron disminuyendo considerablemente en el tiempo.

Conclusiones

Para que el sistema de control interno logre cumplir sus objetivos de control en organizaciones que posean gran números de AFT, requiere de la implementación de herramientas informática que gestione el flujo de información en la auditoría y el control.

Con la implementación del sistema informática que gestione el proceso de auditoría y control de los AFT, se disminuir considerablemente el gasto de recursos materiales y humanos derivados de la ejecución del proceso.

Con la utilización del sistema informático para la auditoría y control de los AFT, se logra incrementar el por ciento de medios a revisar en una organización, siendo posible llegar hasta el 100% del total representando un excelente índice de control en la entidad.

Referencias

- CONTRALORÍA. *Contraloría General de la República de Cuba*, [En línea]. 2009. [Disponible en: http://www.contraloria.cu/index.php?option=com_content&view=article&id=84&Itemid=22]
- CUNHA, P.; R. KLANN, *et al.* Procedimientos de auditoria aplicados pelas empresas de auditoria independente de Santa Catarina em entidades do terceiro setor *Revista de Contabilidade e Organizações*, 2010, Vol.4(No.10): pp. 65-85.
- DART, S. and R. ELLISON Software Development Environments *IEEE Computer*, 2008, Vol.20 (No.11): 18-28.
- FUNDATION. *OpenUp Basic. Eclipse Foundation*, [En línea]. 2010. [Disponible en: <http://epf.eclipse.org/wikis/openupsp/>]
- GARAY, M. Interfaces Inteligentes en el aprendizaje de la Modelación *Ingeniería Industrial*, 2015, Vol. XXXVI(No. 2): 187-201.
- GONZÁLEZ, J. and O. MAR Algoritmo de clasificación genética para la generación de reglas de clasificación *Publicaciones*, 2015, Vol.8(No.1): 1-14.
- GONZÁLEZ, M. Mirada contextual a los nexos entre las auditorías de información y las auditorías de conocimiento *Ciencias de la Información*, 2011, Vol.42(No.1): pp. 31 - 37.

- JACOBSON, I. El Proceso Unificado de Desarrollo de Software. Madrid *Pearson Educación S.A.*, 2000.
- JARAMILLO, M.; M. GARCÍA, *et al.* Factores que determinan los honorarios de auditoria: Análisis empírico para México *Revista Venezolana de Gerencia*, 2012, Vol.17(No.59): pp. 387-406.
- MAR, O. Procedimiento para determinar el índice de control organizacional *Revista Infocencia*, 2014, Vol.18(No.2).
- MAR, O.; L. ARGOTA, *et al.* Módulo para la evaluación de competencias a través de un Sistema de Laboratorios a Distancias *RCCI*, 2016, Vol.10(No.2): 132-147.
- MILLONES, L. *Los Activos Fijos*, [En línea]. 2011. [Disponible en: <http://www.bibliomaster.com/pdf/2363.pdf>
- POTENCIER, F. and F. ZANINOTTO. *Symfony 1.2, la guía definitiva*. 2010. pp.10-31 p. ISSN: 978-1590597866
- RODRÍGUEZ, Y. Auditoria de información y conocimiento en la organización Ingeniería Industrial *Ingeniería Industrial*, 2012, Vol. XXXIII(No.3): pp. 260-271.
- ROSSO, A. and J. SORIANO Sistema de control inteligente para un grupo de elevadores ciencia e ingeniería *Ciencia e Ingeniería*, 2008, Vol.18(No.2): pp. 177-130.
- SALAS, M. and C. CERÓN Sistema Web para Evaluar las Competencias mediante Pruebas Objetivas en Educación Superior *Revista Iberoamericana para la Investigación y el Desarrollo Educativo*, 2014, (No.12).
- SÁNCHEZ, B. and Y. VALDÉS Diseño de Sistemas de Información Documental. Consideraciones teóricas *Ciencias de la Información*, 2009, Vol.39(No.3).
- SUTZ, R. *La Universidad Latinoamericana del Futuro*, [En línea]. 2000. [Disponible en: <http://www.udual.org/CIDU/ColUDUAL/11/ColUDUAL11.pdf>
- VICTORIA, N. Factores que inciden en el control interno de una organización *Actualidad contable*, 2005, Vol.8(No.11): pp 87 - 92.
- YIKUN, L. and T. KAKOLA. *An Information System Design Product Theory for the Abstract Class of Integrated Requirements and Delivery Management Systems*. System Sciences (HICSS), 2014 47th Hawaii International Conference on, 2014. 3677-3686 p.