

SISREAD: REVISIONES Y AUDITORÍAS COMO PARTE DEL ASEGURAMIENTO DE LA CALIDAD

MSc. Arelys Vázquez Riverón, Ing. Arnoldo Domínguez Leyva

RESUMEN

El resultado final de un proyecto de *software* es un producto que toma forma a lo largo del desarrollo del proyecto. La calidad del producto final, está estrechamente ligada a la calidad del Proceso de Desarrollo de *Software*, entre otros aspectos. Establecer un adecuado sistema de aseguramiento de la calidad que contribuya a detectar los defectos en las etapas tempranas del proceso de desarrollo de *software* contribuye considerablemente a elevar la calidad de los productos resultantes. Una parte importante de este sistema lo constituyen las revisiones y auditorías, que se realizan en los diferentes momentos del ciclo de desarrollo del proyecto. Con ellas se detectan defectos que pueden ser corregidos oportunamente, de forma tal que el producto final tenga la calidad necesaria. En este trabajo se expone la propuesta de un Sistema de Revisiones y Auditorías (SISREAD), que responde a la informatización del procesamiento de las evidencias para facilitar la toma de decisiones. La propuesta está desarrollada en Django. El proceso de desarrollo del *software* estuvo guiado por el enfoque ágil.

Palabras Clave: aseguramiento de la calidad, revisiones, auditoría, SISREAD

ABSTRACT

The end result of a software project is a product that takes shape throughout the project development. The quality of the final product, is closely linked to the quality of Software Development Process, among others. Establish an adequate system of quality assurance that will help detect defects in the early stages of the software development process contributes significantly to improving the quality of the resulting products. An important part of this system are the reviews and audits, which are conducted at different stages of the development cycle of the project. With these activities flaws that can be corrected promptly,

so as detected that the process and the final product have the required quality. This paper describes a management system for reviews and audits (SYSREAD), which responds to the computerization of the processing of information to facilitate decision-making. The solution is developed in Django. The software development process was guided by the agile approach.

KeyWords: quality assurance, reviews, audit, SISREAD.

INTRODUCCIÓN

El uso de las TIC se considera un pilar fundamental para el desarrollo económico de cualquier país. La realidad del mundo actual se caracteriza por ser muy competitiva, en donde las empresas se esfuerzan por ampliar y mejorar su posición en el mercado, basadas, entre otros aspectos, en la calidad de sus productos o servicios. Para ello, dedican parte del tiempo, esfuerzo y no pocas cantidades de dinero a la adquisición de soluciones informáticas que faciliten desde la gestión de sus procesos hasta su posicionamiento en el mercado. En aras de satisfacer las necesidades de los clientes en el mercado, las industrias de *software* deben desarrollar cada vez más, productos en el menor tiempo posible y con mayor calidad y precisamente ahí es donde radica la importancia del control exhaustivo de cada una de las actividades desarrolladas durante todo el proceso de desarrollo del *software*.

El desarrollo de la Industria Nacional de *Software* es una tarea de gran prioridad para Cuba, debido a la alta perspectiva económica que posee, así como para el aseguramiento de un grupo importante de actividades del país.

Varios trabajos reflejan la importancia de establecer el proceso de revisión en las empresas de *software* sustentado en el argumento de que las dos terceras partes de los defectos de los sistemas son el resultado de errores cometidos en etapas tempranas del desarrollo del proyecto y sólo una tercera parte son el resultado de errores cometidos en etapas avanzadas, por lo que se hace necesario prevenir los defectos o detectarlos en las etapas tempranas.

MÉTODOS Y RESULTADOS

El término calidad se utiliza en todos los sectores que se relacionen con la creación de un producto o la prestación de un servicio, ya que incluyen la relación con el cliente. En informática el término se asocia tanto al producto como al proceso de desarrollo del mismo.

El Instituto de Ingenieros Eléctricos y Electrónicos o IEEE (en inglés, Institute of Electrical and Electronics Engineers) define la calidad de *software*, apreciándose una implícita consideración de la opinión del cliente sobre el producto y/o servicio prestado. La misma enuncia: “Grado con el cual el cliente o usuario percibe que el *software* satisface sus expectativas”. (Engineers, 19997)

Pressman considera la calidad de *software* como: “Concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos con los estándares de desarrollo explícitamente documentados y con las características implícitas que se espera de todo *software* desarrollado profesionalmente”. (Pressman, 2007)

La norma ISO 9000, refina el concepto de calidad considerándola como el “grado en el que un conjunto de características inherentes cumple con los requisitos.” (Colombiana, 2000)

A pesar de las diferentes acepciones hoy en día las compañías de todo el mundo industrializado reconocen que la calidad del producto se traduce en ahorro de costos y en una mejora general.

La calidad del *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 de la calidad como su control durante todas las etapas del ciclo de vida del *software*.

A partir de una revisión de las diferentes acepciones de los términos revisiones y auditorías, se decide apoyar la propuesta de solución de este trabajo en la definición extraída del estándar (Engineers, 19997), quien se refiere a las revisiones como un proceso o reunión durante la cual un producto de *software* es presentado al personal del proyecto, administradores, usuarios, clientes, usuario representativos, o a otras partes interesadas para ser comentado o aprobado. Pueden ser dinámicas y estáticas. Las primeras son las que detectan los defectos ejecutando el *software*, fundamentalmente son las ejecutadas en la fase de prueba del proyecto. Las segundas son visuales y se realizan sin necesidad de que el *software* esté ejecutándose. Ambas son de suma importancia y una combinación adecuada puede detectar gran cantidad de defectos y por tanto mejorar la calidad del producto final. Las revisiones son actividades de control de calidad, que permiten detectar defectos en los proyectos de *software*.

Según Pressman (2007), son un filtro para el proceso de ingeniería del *software* y se aplican en varios momentos del desarrollo del *software* para detectar errores y defectos que puedan

así ser eliminados.

Por su parte, las auditorías según el estándar ISO 19011:2002 se definen como un proceso sistemático, independiente y documentado para evaluar el estado actual de un proceso o un producto y evaluarlos de manera objetiva con el fin de determinar la extensión en que se cumplen los lineamientos establecidos.

El análisis de los resultados obtenidos en una revisión o auditoría es bastante exhaustivo en todos los niveles. Sin embargo resulta engorrosa la obtención de cifras estadísticas de dichas inspecciones realizadas a un proyecto o en un período determinado debido al cúmulo de información digital o manuscrito al que hay que acceder para realizar estos cálculos.

SISREAD tiene como objetivo facilitar la gestión de la información referente a las actividades de revisiones y auditorías en los proyectos que desarrollan *software*. De esta manera permite la identificación temprana de desvíos o defectos que atenten contra la calidad del *software* y la adopción de medidas correspondientes. Para el desarrollo de la aplicación fueron definidos los siguientes objetivos:

1. Diseñar el sistema para la gestión de las revisiones y las auditorías en proyectos de desarrollo de *software*.
2. Implementar el sistema para la gestión de las revisiones y las auditorías en proyectos de desarrollo de *software* utilizando herramientas libres.
3. Validar el sistema para la gestión de las revisiones y las auditorías en proyectos de desarrollo de *software* mediante pruebas de *software*.

Entre las funcionalidades que brinda hoy la aplicación se encuentra la adición de las revisiones, permitiendo la inserción de varios datos imprescindibles para la generación de las estadísticas como la selección del proyecto revisado, el período, el equipo que realiza la revisión, el estado de la revisión y el listado de las no conformidades detectadas. Permite personalizar o administrar las No Conformidades mediante nomencladores permitiendo estandarizar las mismas. Cuenta con un área de notificación de mensajes referente a acciones realizadas en el sistema como pueden ser: errores cometidos por el usuario que interactúa con la aplicación o de solicitud de confirmación para determinadas acciones como Eliminar.

Con SISREAD, es posible realizar un análisis del comportamiento de dichas actividades al final de cada trimestre u otro período de interés si durante el mismo se realizó al menos una revisión o auditoría registrando para ello la información mínima necesaria correspondiente.

La aplicación se desarrolla en Django en su versión 1.7.6, que es un framework de código abierto para el desarrollo de aplicaciones web escrito en Python y se ajusta en alguna medida a la estructura Modelo Vista Controlador, lo que en su caso se transcribe como Modelo Vista Plantilla (MTV). Facilita la creación de aplicaciones web, haciendo énfasis en la reutilización de código que caracteriza a los marcos de trabajo, así como en la forma en que se conectan los componentes dentro de la aplicación, un factor que ahorra tiempo de trabajo, por lo que su principal ventaja es permite el desarrollo rápido de las aplicaciones basado en la utilización de poco código. Es el único marco de trabajo que por defecto viene con un sistema de administración activo, que permite ser utilizado sin ningún tipo de configuración.

Se utiliza como lenguaje de programación Python en la versión 2.7. Es un lenguaje interpretado, lo que ahorra un tiempo considerable en el desarrollo del programa, pues no es necesario compilar ni enlazar; sólo hay que escribir el programa y ejecutarlo. En el desarrollo web, esto significa que se puede desarrollar código y ver los resultados inmediatamente al recargar la página en el navegador web. Tiene *tipado dinámico*, por lo que no es necesario declarar los tipos de datos de las variables. Su sintaxis es concisa pero expresiva, por tanto se necesita menos código para llevar a cabo una tarea que en otros lenguajes como Java.

PostgreSQL es un excelente sistema de gestión de base datos objeto-relacional porque es distribuido bajo la licencia BSD y con su código fuente disponible libremente. Es el sistema de gestión de bases de datos de código abierto más potente del mercado. Sus características técnicas lo hacen uno de los gestores de bases de datos más potentes y robustos del mercado. (Martínez, 2010)

PostgreSQL en su versión 9.1 se utiliza para almacenar los datos del sistema por su capacidad de almacenamiento, que es restringida únicamente por el espacio físico disponible en el disco duro de la computadora donde esté instalado y facilita la administración de un gran número de datos. Además, el framework seleccionado se integra de manera fácil con este sistema gestor de base de datos.

La arquitectura del *software*, que abarca según (James Rumbaugh, 1998) los componentes de *software*, las propiedades visibles externamente de esos componentes y las relaciones entre ellas, está basada en el patrón Model Template View (MTV), que no es más que la implementación propia del Model View Controller (MVC), tomando elementos del propio

patrón y realizando pequeñas modificaciones. Según las nuevas definiciones determinadas por este framework, el controlador adopta el nombre de View (vista) y la vista pasa a denominarse Template (plantilla).

El diseño se apoya también en el uso de algunos patrones como el bajo Acoplamiento, que se evidencia entre las Vistas y URLs, estas últimas llaman a las funciones y métodos que implementan las primeras, pero algún cambio que se realice a una función o método, no afecta la URL, Alta Cohesión y Experto.

Se considera que la aplicación optimiza el tiempo que normalmente se invierte en realizar consultas o elaborar estadísticas a partir de un cúmulo de archivos digitales o manuscritos, en los cuales mientras más antiguo o extenso sea el período a analizar o reportar, pues más difícil será obtener la información. Garantiza el acceso a la información por personas previamente autorizadas lo que permite una gestión rápida y fidedigna de las revisiones y las auditorías, eliminando así riesgos de pérdida o corrupción de archivos. La información permanece centralizada, lo que agiliza el acceso a la misma sin depender de un correo electrónico u otro dispositivo, eliminando así el riesgo de duplicidad.

Cualquier organización que se dedica al desarrollo de *software* debe considerar la calidad, ya que el cliente es cada vez más exigente. Para garantizar la calidad de SISREAD se diseñaron un conjunto de pruebas en diferentes niveles, utilizando tanto métodos de caja blanca como de caja negra. Las pruebas diseñadas y realizadas permitieron validar el correcto funcionamiento de las características y servicios brindados.

CONCLUSIONES

Después de realizar el trabajo anterior se arribó a las siguientes conclusiones:

- Es un hecho que la demanda de soluciones informáticas hoy en día exigen más resultados en menos tiempo. La solución que se propone en esta investigación posee grandes posibilidades de contribuir a las exigencias de esta industria.
- Mediante la aplicación de dicha herramienta para el procesamiento de la información, se detectan problemas existentes y comunes en varios proyectos que constituyen obstáculos para lograr un exitoso proceso de desarrollo de *software*, agilizando la toma de decisiones al respecto.
- Permite una gestión rápida y fidedigna de la información referente a las revisiones y las auditorías.

BIBLIOGRFÍA

Colombiana, N. T. (2000). NTC-ISO 9000. Fundamentos y vocabularios.

Engineers, T. I. o. E. a. E. (1997). IEEE Standard 1028 for Software Reviews.

James Rumbaugh, G. B., Ivar Jacobson. (1998). *El lenguaje unificado de modelado*.

Martínez, R. (2010). PostgreSQL. from www.postgresql.org.es

Pressman, R. (2007). *Ingeniería del Software. Un enfoque práctico*.