

"Revista Virtual Universidad Católica del Norte". No. 34, (septiembre-diciembre de 2011, Colombia), acceso: [<http://revistavirtual.ucn.edu.co/>], ISSN 0124-5821 - Indexada Publindex-Colciencias (B), Latindex, EBSCO Information Services, Redalyc, Dialnet, DOAJ, Actualidad Iberoamericana, Índice de Revistas de Educación Superior e Investigación Educativa (IRESIE) de la Universidad Autónoma de México.

## **Metodología integrada al proceso de construcción de software para aplicar inspecciones y pruebas<sup>1</sup>**

### **A Methodology Integrated to the Software Construction Process to Perform Inspections and Tests**

### **Méthodologie intégrée au processus de construction des logiciels pour appliquer inspections et tests**

#### **Clara Patricia Avella Ibáñez**

Ingeniera de Sistemas

Especialista en Informática para Gerencia de Proyectos

Magíster en Ingeniería de Sistemas y Computación

Universidad de Boyacá

Correos: [cpavella@gmail.com](mailto:cpavella@gmail.com), [cpavella@uniboyaca.edu.co](mailto:cpavella@uniboyaca.edu.co)

#### **Juan Federico Gómez Estupiñán**

Ingeniero de Sistemas

Especialista en Telemática

Msc(c) en Ciencias de la Información y las Comunicaciones

Universidad de Boyacá

Correos: [jfgomeze@gmail.com](mailto:jfgomeze@gmail.com), [jfgomez@uniboyaca.edu.co](mailto:jfgomez@uniboyaca.edu.co)

**Tipo de artículo:** Investigación científica y tecnológica.

**Recepción:** 2011-04-02

**Revisión:** 2011-08-08

**Aprobación:** 2011-08-17

---

<sup>1</sup> Artículo resultado del proyecto de investigación: "Definición de un método integrado para aplicación de inspecciones y pruebas de software", desarrollado en el programa de Ingeniería de Sistemas de la Universidad de Boyacá. Financiado por esta universidad. Línea de investigación: Calidad de Software, Grupo de Investigación en Procesos y Calidad de Software (GIPOCAS). Fecha de inicio: marzo de 2007, finalizado: diciembre de 2009.

“Revista Virtual Universidad Católica del Norte”. No. 34, (septiembre-diciembre de 2011, Colombia), acceso: [<http://revistavirtual.ucn.edu.co/>], ISSN 0124-5821 - Indexada Publindex-Colciencias (B), Latindex, EBSCO Information Services, Redalyc, Dialnet, DOAJ, Actualidad Iberoamericana, Índice de Revistas de Educación Superior e Investigación Educativa (IRESIE) de la Universidad Autónoma de México.

---

## Contenido

1. Introducción
2. Generalidades sobre inspecciones y pruebas de software
3. Método
4. Resultados del proyecto de investigación
5. Conclusiones
6. Lista de referencias

### Resumen

El proyecto de investigación surgió al observar que en los procesos de desarrollo de software se aplicaban pruebas e inspecciones como tareas puntuales e independientes, pero no se evidenciaba una metodología que integrara estas actividades. La presente indagación inició con la revisión del estado del arte en cuanto a pruebas e inspecciones de software, así como los estándares existentes para aplicar estos procesos. Posteriormente se definieron las etapas de la metodología, incluyendo actividades y herramientas para cada una de ellas. El resultado fue una metodología integral para aplicar inspecciones y pruebas dentro del proceso de construcción de software. Como conclusión se destaca que los procesos de inspecciones y pruebas no deben ser aplicados de forma independiente y excluyente, sino de manera complementaria a lo largo del proceso de construcción de software, dado que ninguno sustituye al otro, y al integrarlos se obtienen mejores resultados en cuanto a la calidad del producto final.

### Palabras clave

Aseguramiento de calidad de software, Calidad de software, Inspecciones de software, Proceso de construcción de software, Pruebas de software.

### Abstract

This research project emerge from observing that during the software development processes were performed tests and inspections as specific and independent tasks, but was not evident a methodology that integrated those activities. This study started with a review on the state-of-the-art regarding to software tests and inspections, and about the existing standards to perform these processes. Subsequently the methodology stages were defined, including tools and activities for each of them. The result was a comprehensive methodology to perform inspections and tests during the software construction process. As a conclusion we emphasize on the fact that inspections and tests must not be performed

"Revista Virtual Universidad Católica del Norte". No. 34, (septiembre-diciembre de 2011, Colombia), acceso: [<http://revistavirtual.ucn.edu.co/>], ISSN 0124-5821 - Indexada Publindex-Colciencias (B), Latindex, EBSCO Information Services, Redalyc, Dialnet, DOAJ, Actualidad Iberoamericana, Índice de Revistas de Educación Superior e Investigación Educativa (IRESIE) de la Universidad Autónoma de México.

independently nor in an exclusive way, but in a complementary way in the course of the software construction process, because none of them substitutes the other, and when they become integrated better results are obtained as regards final product quality.

### **Keywords**

Software quality assurance, Software quality, Software inspections, Software construction process, Software tests.

### **Résumé**

Le projet de recherche a surgi quand nous avons remarqué que pendant les processus de développement des logiciels étaient appliqués test et inspections comme tâches ponctuelles et indépendantes, mais n'était pas évident une méthodologie qu'intègre ces activités. Cette recherche a commencé avec une révision de l'état de l'art par rapport à tests et inspections sur logiciels, ainsi que des standards existants pour appliquer ces processus. Après ça on a défini les étapes de la méthodologie, en incluant activités et outils pour chacune. Le résultat est une méthodologie intégrale pour appliquer inspections et tests pendant le processus de construction de logiciel. En guise de conclusion nous remarquons que les processus d'inspections et tests ne doivent pas être appliqués d'une manière indépendant et exclusive, mais de manière complémentaire pendant le processus de constructions de logiciel, parce que aucun remplace l'autre, et quand ils deviennent intégrés sont obtenus par rapport à la qualité du produit final.

### **Mots-clés**

Affermissement de la qualité des logiciels, Qualité des logiciels, Inspections des logiciels, Processus de construction des logiciels, Tests de logiciel.

## **1. Introducción**

Los procesos de inspecciones y pruebas de software deben ser planeados adecuadamente, dedicando el tiempo y esfuerzo necesarios en la búsqueda de la calidad del producto software. Mediante estos procesos se puede hacer un seguimiento y control de todas las actividades que se adelantan en el desarrollo de cualquier producto.

"Revista Virtual Universidad Católica del Norte". No. 34, (septiembre-diciembre de 2011, Colombia), acceso: [<http://revistavirtual.ucn.edu.co/>], ISSN 0124-5821 - Indexada Publindex-Colciencias (B), Latindex, EBSCO Information Services, Redalyc, Dialnet, DOAJ, Actualidad Iberoamericana, Índice de Revistas de Educación Superior e Investigación Educativa (IRESIE) de la Universidad Autónoma de México.

Aunque la mayoría de equipos de desarrollo de software aplican pruebas en su proceso de construcción, existe poca información respecto a la planeación y a la aplicación de los procesos de inspecciones y pruebas, enmarcados en las etapas de construcción del software.

Este artículo presenta la síntesis de una metodología estructurada e integral, aplicable a procesos de inspecciones y pruebas de software, que es el producto del proyecto de investigación denominado "Definición de un método integrado para aplicar inspecciones y pruebas de software", realizado en el programa de Ingeniería de Sistemas de la Universidad de Boyacá.

Esta metodología incluye elementos que han sido adaptados teniendo en cuenta estándares internacionales referentes a calidad, inspecciones y pruebas de software: el estándar IEEE Std 1028-1997 sobre revisiones de software, el estándar IEEE Std 829-2008 relacionado con documentación de pruebas de software y el estándar ISO/IEC 9126-1 sobre especificaciones de calidad. También se utilizaron como referentes fundamentales las propuestas desarrolladas por Rivera (1998) y Dustin (2002).

El artículo expone los conceptos generales asociados a los procesos de calidad, inspecciones y pruebas de software, posteriormente diserta acerca del equipo de trabajo necesario para estos procesos. Seguidamente se presenta la metodología propuesta y se describe su aplicación en cada una de las etapas en el proceso de construcción de software, finalizando con las conclusiones del proyecto de investigación.

## **2. Generalidades sobre inspecciones y pruebas de software**

Producir software de alta *calidad* es el propósito fundamental de la ingeniería de software, por ende, todos los estándares, normas, metodologías y herramientas definidas se dirigen al logro de este propósito. La búsqueda de la calidad del producto debe estar presente en cada una de las etapas del proceso de construcción de software (Avella, Gómez & Caro, 2011).

A continuación se presentan de manera general los conceptos básicos relacionados con el tema de investigación.

“Revista Virtual Universidad Católica del Norte”. No. 34, (septiembre-diciembre de 2011, Colombia), acceso: [<http://revistavirtual.ucn.edu.co/>], ISSN 0124-5821 - Indexada Publindex-Colciencias (B), Latindex, EBSCO Information Services, Redalyc, Dialnet, DOAJ, Actualidad Iberoamericana, Índice de Revistas de Educación Superior e Investigación Educativa (IRESIE) de la Universidad Autónoma de México.

## **2.1 Calidad de software**

Pressman (2005) define la *calidad de software* como la concordancia del producto de software 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.

Un referente internacional es el estándar ISO/IEC 9126-1 (2001), que provee un marco de trabajo para la evaluación de la calidad del software y establece un modelo general aplicable a cualquier tipo de software. La primera parte del estándar define la especificación de calidad que incluye seis características claves a través de las cuales se describe y se evalúa la calidad del producto. Estas características son: funcionalidad, fiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad (Lucero, 2001).

## **2.2 Aseguramiento de la calidad de software**

El *Aseguramiento de la Calidad del Software* (SQA Software Quality Assurance) es una actividad que debe aplicarse durante todo el proceso de construcción de software. Como su nombre lo indica, busca asegurar que el producto en desarrollo –en este caso el software– se ajuste a todos los requisitos de calidad esperados. SQA es un conjunto de actividades planificadas y sistemáticas indispensables para asegurarle al cliente que el producto software cumplirá cabalmente con los requisitos de calidad previamente establecidos (Cueva, 1999; Van Zeist, 1996).

## **2.3 Inspecciones de software**

Las *inspecciones de software*, también conocidas como revisiones técnicas formales, son procedimientos de revisión realizados conjuntamente en todas las etapas del proceso de construcción de software, sobre los documentos de soporte que se van generando en cada una de las actividades del proceso y en el código fuente del producto software, con el propósito de detectar errores, omisiones o inconsistencias y realizar las acciones correctivas correspondientes (Avella, Gómez & Caro, 2011; Díez, 2000).

Uno de los propósitos fundamentales de las inspecciones de software es encontrar oportunamente los errores y evitar su propagación y posterior conversión en defectos de software. La detección temprana de los errores permitirá realizar las acciones correctivas pertinentes, que se traducen en la reducción de costos, tiempo y esfuerzo en el proceso de construcción de

"Revista Virtual Universidad Católica del Norte". No. 34, (septiembre-diciembre de 2011, Colombia), acceso: [<http://revistavirtual.ucn.edu.co/>], ISSN 0124-5821 - Indexada Publindex-Colciencias (B), Latindex, EBSCO Information Services, Redalyc, Dialnet, DOAJ, Actualidad Iberoamericana, Índice de Revistas de Educación Superior e Investigación Educativa (IRESIE) de la Universidad Autónoma de México.

software. Entre más rápido sean detectados y corregidos los errores, menor será su impacto en términos financieros y de tiempo (Rivera, 1998).

## **2.4 Pruebas de software**

Las *pruebas de software* son procedimientos manuales o automáticos consistentes en la ejecución del producto de software a examinar, suministrando diversos datos de entrada para encontrar la mayor cantidad de errores posibles en el mismo, antes de que éste sea puesto en producción. (Avella, Gómez & Caro, 2011; McGregor, 2001).

A diferencia de las inspecciones, las pruebas son una técnica dinámica de verificación y de validación del software, las cuales se aplican directamente sobre el producto software en funcionamiento. Tienen dos propósitos fundamentales: demostrar que el software satisface todos los requisitos y descubrir los errores del software que evitan que el sistema funcione correctamente (Sommerville, 2005; Guzmán, 2004).

El estándar IEEE Std 829-2008 (2008) describe un conjunto de documentos básicos que deben ser utilizados en las etapas del proceso de pruebas de software. Este estándar especifica la forma y el contenido de los documentos para pruebas individuales, pero no determina los detalles de las pruebas que se requieren para un caso en particular. Actualmente se está desarrollando un nuevo estándar de pruebas identificado como ISO/IEC 29119, cuyos objetivos son: unificar los estándares anteriores, cubrir el ciclo de vida completo incluyendo los aspectos no considerados por otros estándares, que sea aplicable a todo tipo de sistemas software y que sea consistente con otros estándares de la ISO (Tuya, 2009).

## **2.5 Inspecciones versus pruebas**

Aunque las inspecciones tienen múltiples ventajas, es fundamental que éstas sean realizadas con disciplina y rigor para garantizar su eficacia. Además, no significa que las inspecciones reemplacen completamente las pruebas del sistema, sino que son un proceso de verificación inicial para encontrar errores del sistema. Por otro lado, el uso de inspecciones durante el proceso de desarrollo, tampoco garantiza la detección de todos los errores. Por esta razón es necesario complementar el uso de inspecciones con técnicas de generación de casos de prueba que permitan maximizar la detección de errores durante la fase de pruebas (Rivera, 1998).

"Revista Virtual Universidad Católica del Norte". No. 34, (septiembre-diciembre de 2011, Colombia), acceso: [<http://revistavirtual.ucn.edu.co/>], ISSN 0124-5821 - Indexada Publindex-Colciencias (B), Latindex, EBSCO Information Services, Redalyc, Dialnet, DOAJ, Actualidad Iberoamericana, Índice de Revistas de Educación Superior e Investigación Educativa (IRESIE) de la Universidad Autónoma de México.

Las inspecciones permiten verificar si el sistema concuerda con una especificación, pero no pueden validar el comportamiento dinámico, es decir, cuando el sistema está en funcionamiento, esto sólo es posible mediante la aplicación de pruebas. Las inspecciones no pueden verificar características no funcionales como el desempeño, la fiabilidad, la usabilidad y la validación de la interfaz con el usuario; esto sólo es posible con la aplicación adecuada de los casos de prueba (Sommerville, 2005).

En conclusión, las inspecciones y las pruebas no son opuestas, sino actividades complementarias; ambas deben utilizarse conjuntamente durante la validación y la verificación del proceso de construcción de software, con el fin de buscar el aseguramiento de la calidad del producto (SQA).

### **3. Método**

Para el desarrollo del proyecto de investigación se llevaron a cabo las siguientes actividades, conducentes al logro de los objetivos propuestos:

- Revisión del estado del arte sobre pruebas, inspecciones, tipos de pruebas, errores y defectos de software, calidad de software y temas afines. En esta actividad se utilizaron fuentes bibliográficas como libros sobre el tema, estándares internacionales y artículos científicos.
- Identificación de las etapas del proceso de construcción de software y definición de cada una de las actividades de inspección y pruebas a realizar en cada etapa. Para la realización de esta actividad fue necesario revisar las diferentes propuestas existentes en cuanto al proceso de construcción de software y a procesos de inspecciones y pruebas, para estructurar la metodología propuesta en la investigación.
- Definición de la estructura del equipo responsable de pruebas e inspecciones de software, determinando los roles, las funciones y responsabilidades de cada uno de sus integrantes. Esta actividad se realizó tomando como base diferentes propuestas de autores sobre los roles necesarios para aplicar inspecciones y pruebas. Como estos procesos por lo general se presentan por separado, la propuesta de la

"Revista Virtual Universidad Católica del Norte". No. 34, (septiembre-diciembre de 2011, Colombia), acceso: [<http://revistavirtual.ucn.edu.co/>], ISSN 0124-5821 - Indexada Publindex-Colciencias (B), Latindex, EBSCO Information Services, Redalyc, Dialnet, DOAJ, Actualidad Iberoamericana, Índice de Revistas de Educación Superior e Investigación Educativa (IRESIE) de la Universidad Autónoma de México.

metodología consistió en plantear un solo equipo que involucrara los roles necesarios para ambos procesos.

- Diseño y elaboración de listas de chequeo para aplicación de inspecciones y formatos de casos de prueba. Para la realización de esta actividad se consultaron estándares internacionales y propuestas de diferentes autores sobre listas de chequeo para inspecciones, de otra parte se tomó como base la propuesta de Rivera (1998) en cuanto a diseño de casos de prueba y se realizaron algunos cambios a la misma.
- Aplicación de prueba piloto utilizando la metodología propuesta. Esta prueba se aplicó a dos cursos de Ingeniería de Software durante tres semestres consecutivos, toda vez que en esta asignatura se desarrolla un producto de software como proyecto de curso y es necesario aplicar los procesos de inspecciones y pruebas de software.
- Refinamiento de la metodología de acuerdo con los resultados de la prueba piloto. La aplicación de esta prueba piloto arrojó algunos ítems a modificar, específicamente en los formatos propuestos, tanto para inspecciones como para pruebas y fue necesario realizar ajustes a los mismos. De otra parte, se vio la necesidad de incluir algunas actividades en la metodología, que inicialmente no estaban.
- Elaboración del documento final de investigación. Esta actividad se realizó teniendo en cuenta las normativas de la Universidad de Boyacá en cuanto a la presentación de informes finales de investigación.
- Publicación de un libro acerca de la metodología propuesta. El producto principal de la investigación fue el libro denominado *Aplicación de inspecciones y pruebas de software*, Ediciones Universidad de Boyacá, año 2011, ISBN 978-958-8642-09-3.



"Revista Virtual Universidad Católica del Norte". No. 34, (septiembre-diciembre de 2011, Colombia), acceso: [<http://revistavirtual.ucn.edu.co/>], ISSN 0124-5821 - Indexada Publindex-Colciencias (B), Latindex, EBSCO Information Services, Redalyc, Dialnet, DOAJ, Actualidad Iberoamericana, Índice de Revistas de Educación Superior e Investigación Educativa (IRESIE) de la Universidad Autónoma de México.

## **4. Resultados del proyecto de investigación**

La siguiente es la propuesta metodológica para la aplicación de inspecciones y pruebas de software.

### **4.1 Conformación del equipo de trabajo para inspecciones y pruebas**

Para emprender un proceso de inspecciones y de pruebas, es necesario conformar previamente un equipo de trabajo. En el estándar IEEE Std 1028-1997 (1998) se definen los roles y las responsabilidades para el equipo de inspecciones. Rivera (1998) también presenta una propuesta de equipo de trabajo para las actividades de inspección y Dustin (2002) expone los roles y las responsabilidades de los integrantes del equipo de pruebas.

Como propone Dustin (2002), un equipo de pruebas eficaz o eficiente incluye la combinación de experiencia técnica y un dominio relevante del problema de software planteado. Es decir, no sólo se requiere que el equipo tenga todas las habilidades técnicas necesarias, también es necesario que esté bien estructurado y que conozca ampliamente el problema de software a resolver.

Los inspectores y los probadores se clasifican de acuerdo con su tiempo de experiencia, es así que se encuentra: *Inspectores y/o probadores junior* e *Inspectores y/o probadores senior*. Algunas veces, estas categorías se subdividen en razón a la complejidad de las aplicaciones a verificar.

La conformación del equipo de inspecciones y pruebas requiere que previamente se definan los roles y responsabilidades de sus integrantes.

Para el proceso de inspecciones se debe contar con los siguientes roles: administrador de inspecciones, inspector o revisor, autor o desarrollador y lector o secretario.

Para el proceso de pruebas es necesario contar con los siguientes roles: administrador de pruebas, líder de prueba, ingeniero de pruebas de usabilidad (facilidad de uso), ingeniero de pruebas manuales, ingeniero de pruebas automáticas, ingeniero de pruebas de red, ingeniero de pruebas de

"Revista Virtual Universidad Católica del Norte". No. 34, (septiembre-diciembre de 2011, Colombia), acceso: [<http://revistavirtual.ucn.edu.co/>], ISSN 0124-5821 - Indexada Publindex-Colciencias (B), Latindex, EBSCO Information Services, Redalyc, Dialnet, DOAJ, Actualidad Iberoamericana, Índice de Revistas de Educación Superior e Investigación Educativa (IRESIE) de la Universidad Autónoma de México.

seguridad, especialista del ambiente de prueba y especialista en configuración y documentación de pruebas (Dustin, 2002).

Las responsabilidades y competencias específicas para cada rol se encuentran detalladas en el libro *Aplicación de inspecciones y pruebas de software* (Avella, Gómez & Caro, 2011).

El equipo de trabajo debe estar conformado por personal altamente calificado, capaz de responder a cabalidad con las responsabilidades y con los roles asignados. No obstante, una misma persona puede ser capaz de desempeñar múltiples roles de manera simultánea. Por ejemplo, una persona con suficiente experiencia puede administrar las inspecciones y las pruebas.

El número de integrantes del equipo de inspecciones y de pruebas estará supeditado a la complejidad del producto que será desarrollado, a los recursos humanos y financieros disponibles y a la estructura del equipo de desarrollo, teniendo en cuenta que existen organizaciones que tienen su propia unidad de desarrollo de software y otras cuyo objetivo primordial es la construcción de software a nivel comercial.

## **4.2 Etapas de la metodología para la aplicación de inspecciones y pruebas de software**

La propuesta metodológica para la aplicación de inspecciones y de pruebas se enmarca en el modelo general del proceso de construcción de software mostrado en la figura 1. La propuesta, resultado de la investigación, presenta las etapas generales y para cada una de ellas las actividades a desarrollar, las técnicas y las herramientas a utilizar, así como los formatos de listas de chequeo y casos de prueba. En el presente artículo sólo se muestran las actividades establecidas en cada una de las etapas de la metodología.

**Figura 1.** Modelo general de proceso de construcción de software.

"Revista Virtual Universidad Católica del Norte". No. 34, (septiembre-diciembre de 2011, Colombia), acceso: [<http://revistavirtual.ucn.edu.co/>], ISSN 0124-5821 - Indexada Publindex-Colciencias (B), Latindex, EBSCO Information Services, Redalyc, Dialnet, DOAJ, Actualidad Iberoamericana, Índice de Revistas de Educación Superior e Investigación Educativa (IRESIE) de la Universidad Autónoma de México.

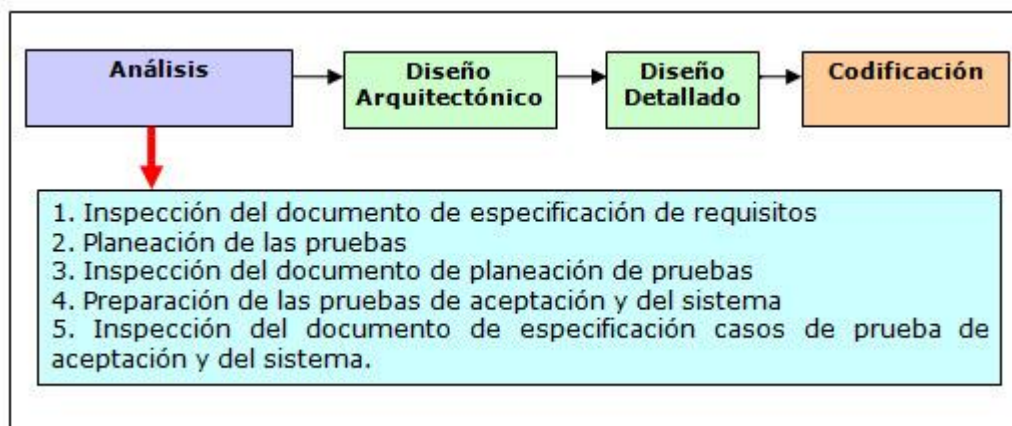


Fuente: Avella, Gómez y Caro (2011). p. 60.

#### 4.2.1 Etapa de análisis

El análisis de sistemas define los objetivos y los límites del sistema objeto de estudio, caracteriza su estructura y su funcionamiento, marca las directrices que permitan alcanzar los objetivos propuestos y evalúa sus consecuencias (ITLP, 2007). La figura 2 muestra las tareas de la metodología, propuestas para esta etapa.

**Figura 2.** Aplicación de inspecciones y pruebas en la etapa de análisis.



Fuente: Avella, Gómez y Caro (2011). p. 61.

Teniendo en cuenta que el resultado sustancial de esta etapa es el documento de especificación de requisitos, la primera actividad de la metodología propuesta es la inspección de este documento. Para la aplicación de esta inspección se definió una lista de chequeo basada en el estándar IEEE STD-830-1998 (1998), (Avella, Gómez & Caro, 2011; UPM, 2008).

"Revista Virtual Universidad Católica del Norte". No. 34, (septiembre-diciembre de 2011, Colombia), acceso: [<http://revistavirtual.ucn.edu.co/>], ISSN 0124-5821 - Indexada Publindex-Colciencias (B), Latindex, EBSCO Information Services, Redalyc, Dialnet, DOAJ, Actualidad Iberoamericana, Índice de Revistas de Educación Superior e Investigación Educativa (IRESIE) de la Universidad Autónoma de México.

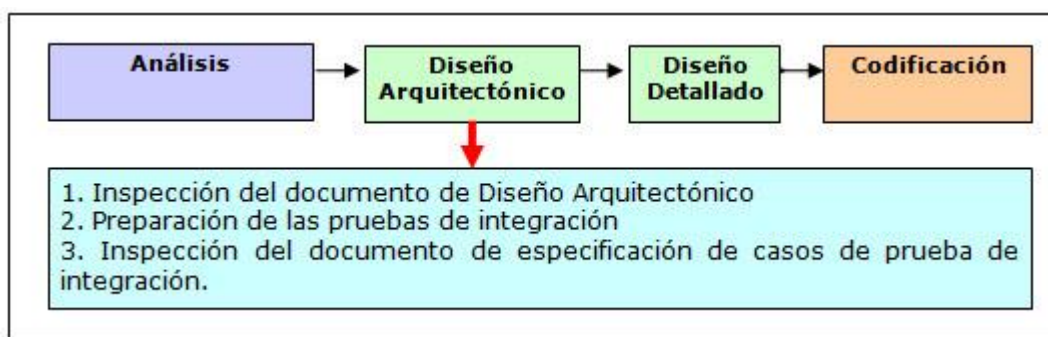
A nivel de pruebas, en esta etapa se planea el proceso, generando el documento correspondiente, de otra parte, se preparan los casos de *pruebas de aceptación y del sistema*. Estas pruebas tienen como propósito detectar errores respecto al cumplimiento de los requisitos del sistema. Los documentos generados deben ser inspeccionados con el fin de evidenciar su validez.

#### 4.2.2 Etapa de diseño arquitectónico

El diseño arquitectónico es la definición de la estructura del sistema en términos de sus componentes y subcomponentes y la relación establecida entre ellos (Sommerville, 2005). Es importante tener en cuenta la arquitectura del sistema para preparar las pruebas de integración.

Dustin (2002) indica que es necesario que el equipo de inspecciones y de pruebas conozca el sistema en su interior y desde etapas tempranas de su ciclo de vida, pues esto le permitirá diseñar mejores pruebas y diagnosticar los defectos con mayor efectividad. En la figura 3 se identifican las actividades realizadas en esta etapa.

**Figura 3.** Aplicación de inspecciones y pruebas en la etapa de diseño arquitectónico.



Fuente: Avella, Gómez y Caro (2011). p. 78.

En la etapa de diseño arquitectónico se genera el documento que plasma la arquitectura del sistema, así que la metodología propone hacer una inspección detallada de este documento, para lo cual se proponen las listas de chequeo a utilizar. Por otra parte, se preparan los casos de *pruebas de*

"Revista Virtual Universidad Católica del Norte". No. 34, (septiembre-diciembre de 2011, Colombia), acceso: [<http://revistavirtual.ucn.edu.co/>], ISSN 0124-5821 - Indexada Publindex-Colciencias (B), Latindex, EBSCO Information Services, Redalyc, Dialnet, DOAJ, Actualidad Iberoamericana, Índice de Revistas de Educación Superior e Investigación Educativa (IRESIE) de la Universidad Autónoma de México.

*integración*. Estas pruebas tienen como propósito detectar errores al integrar los componentes del sistema. El documento de diseño de casos de prueba de integración que se genera en esta etapa, también es objeto del proceso de inspección con el fin de verificar su consistencia.

#### 4.2.3 Etapa de diseño detallado

En esta etapa se propone la solución del sistema en términos de estructuras de datos y de algoritmos que implementarán cada uno de los requisitos del sistema, de acuerdo con la arquitectura definida en la fase anterior. La figura 4, presenta las actividades a realizar en esta etapa, en cuanto a inspecciones y pruebas.

**Figura 4.** Aplicación de inspecciones y pruebas en la etapa de diseño detallado.



Fuente: Avella, Gómez y Caro (2011). p. 83.

Las inspecciones en este nivel verifican la completitud de las especificaciones de diseño detallado para cada uno de los componentes del sistema, es así que en la metodología definen las listas de chequeo requeridas para realizar la inspección al documento de diseño detallado. En esta etapa también se diseñan los casos de las *pruebas de unidad*. Estas pruebas tienen como finalidad encontrar errores de lógica, de estructura de datos y de algoritmos en el interior de las unidades funcionales de código del sistema. Como en las etapas anteriores, es necesario realizar una inspección al documento generado, que en este caso corresponde a los casos de prueba producidos.

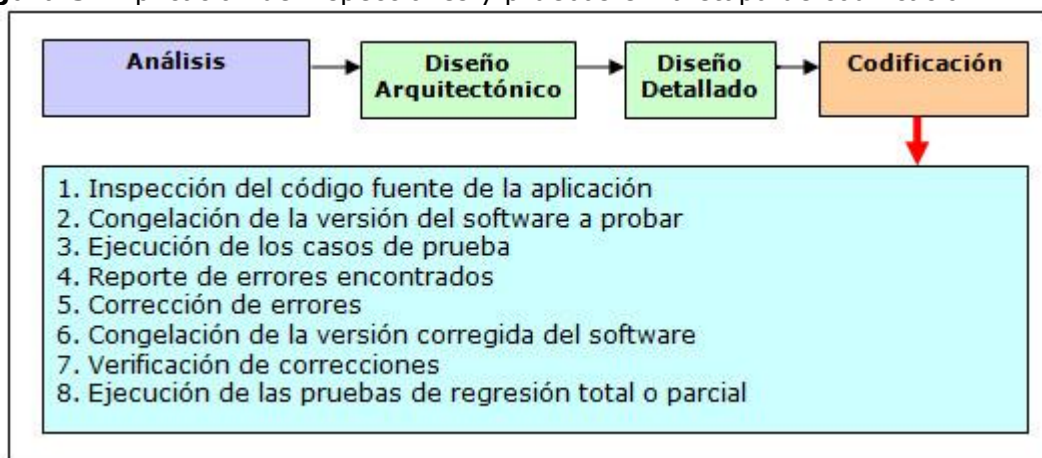
"Revista Virtual Universidad Católica del Norte". No. 34, (septiembre-diciembre de 2011, Colombia), acceso: [<http://revistavirtual.ucn.edu.co/>], ISSN 0124-5821 - Indexada Publindex-Colciencias (B), Latindex, EBSCO Information Services, Redalyc, Dialnet, DOAJ, Actualidad Iberoamericana, Índice de Revistas de Educación Superior e Investigación Educativa (IRESIE) de la Universidad Autónoma de México.

#### 4.2.4 Etapa de codificación

En esta etapa se genera el código del sistema que será implementado a partir de las especificaciones de diseño y donde los probadores ejecutan las pruebas soportadas con los diseños de las mismas y elaboradas en las etapas anteriores. Para facilitar el desarrollo de las pruebas de unidad, los desarrolladores deberían considerar que éstas se realicen sobre la interfaz. Para esto es una buena práctica desarrollar la interfaz antes de hacer las pruebas (Avella, Gómez & Caro, 201; Dustin, 2002).

En la figura 5 se ilustran las actividades que realiza el equipo de inspecciones y de pruebas en la etapa de codificación del sistema, basadas en la propuesta de Rivera (1998).

**Figura 5.** Aplicación de inspecciones y pruebas en la etapa de codificación.



Fuente: Avella, Gómez y Caro (2011). p. 88.

El producto generado en esta etapa es el código fuente del sistema, por lo tanto, se propone hacer un primer filtro sobre este código mediante un proceso de inspección. Posteriormente se ejecutan las pruebas, utilizando los casos de prueba definidos en las etapas anteriores. De otra parte, es indispensable realizar la corrección de los errores encontrados por los responsables del desarrollo, y finalmente es competencia del grupo de inspecciones y pruebas, verificar que se realicen las correcciones respectivas y aplicar las *pruebas de regresión* total o parcial, las cuales tienen como

"Revista Virtual Universidad Católica del Norte". No. 34, (septiembre-diciembre de 2011, Colombia), acceso: [<http://revistavirtual.ucn.edu.co/>], ISSN 0124-5821 - Indexada Publindex-Colciencias (B), Latindex, EBSCO Information Services, Redalyc, Dialnet, DOAJ, Actualidad Iberoamericana, Índice de Revistas de Educación Superior e Investigación Educativa (IRESIE) de la Universidad Autónoma de México.

objetivo repetir total o parcialmente los casos de prueba diseñados, para comprobar que las correcciones no perjudicaron ningún elemento funcional de la versión anterior del software.

## **5. Conclusiones**

- Los procesos de inspecciones y pruebas apoyan el aseguramiento de la calidad de los productos de software. Estos procesos se presentan generalmente como técnicas independientes, sin embargo, la investigación realizada ha permitido determinar, que aunque tengan enfoques diferentes, se pueden utilizar de forma complementaria, teniendo como finalidad la búsqueda de la calidad del producto.
- Las inspecciones y pruebas de software, como técnicas fundamentales en el aseguramiento de la calidad de un producto, deben considerarse como parte integral del proceso de construcción de software y no como tareas yuxtapuestas e independientes de este proceso.
- Las inspecciones y pruebas deben ser planeadas, diseñadas y documentadas desde las etapas iniciales del proceso de construcción de software, con el objetivo de filtrar errores oportunamente, que de no ser detectados, se pueden propagar e incrementar a lo largo del desarrollo del producto.
- Es importante utilizar un conjunto de formatos de listas de chequeo y casos de prueba estandarizados para aplicar la metodología propuesta, es así que los formatos diseñados en la investigación, son el resultado de un análisis de diferentes autores y estándares generalmente aceptados en la Ingeniería de Software.

"Revista Virtual Universidad Católica del Norte". No. 34, (septiembre-diciembre de 2011, Colombia), acceso: [<http://revistavirtual.ucn.edu.co/>], ISSN 0124-5821 - Indexada Publindex-Colciencias (B), Latindex, EBSCO Information Services, Redalyc, Dialnet, DOAJ, Actualidad Iberoamericana, Índice de Revistas de Educación Superior e Investigación Educativa (IRESIE) de la Universidad Autónoma de México.

## 6. Lista de Referencias

- Avella Ibáñez, C. P., Gómez Estupiñán, J.F. y Caro Pineda, S. (2011). *Aplicación de inspecciones y pruebas de software*. Tunja, Colombia: Ediciones Universidad de Boyacá.
- Cueva Lovelle, J. M. (1999). *Calidad de Software*. Recuperado de: [http://gidis.ing.unlpam.edu.ar/downloads/pdfs/Calidad\\_software.pdf](http://gidis.ing.unlpam.edu.ar/downloads/pdfs/Calidad_software.pdf)
- Díez, E. (2000). *Pruebas de software inspecciones formales*. Recuperado de: [http://www.itba.edu.ar/capis/rtis/articulosdeloscuadernosetapaprevia/DIEZ\\_-\\_INSPECCIONES.pdf](http://www.itba.edu.ar/capis/rtis/articulosdeloscuadernosetapaprevia/DIEZ_-_INSPECCIONES.pdf)
- Dustin, E. (2002). *Effective Software Testing: 50 Specific Ways to Improve your Testing*. Boston, United States of America: Addison Wesley.
- Guzmán Cortés, O. H. (2004). Aplicación práctica del diseño de pruebas de software a nivel de programación. *Revista sistemas y telemática*. Universidad ICESI, 2004, 83-128.
- Institute of Electrical and Electronics Engineers. IEEE Computer Society. (1998). *IEEE Std 830-1998. Estándar de especificaciones de Requisitos de Software SRS*. New York. USA.
- Institute of Electrical and Electronics Engineers. IEEE Computer Society. Software Engineering Standards Committee. (1998). *IEEE Std 1028-1997. Standard for Software Reviews*. New York. USA.
- Institute of Electrical and Electronics Engineers. IEEE Computer Society. Software Engineering Technical Committee. (2008). *IEEE Std 829-2008. Standard for Software Test Documentation*. New York. USA.
- Instituto Tecnológico de la Paz. Departamento de Sistemas y Computación. (2007). *Prueba de Programas*. Recuperado de: <http://sistemas.itlp.edu.mx/tutoriales/analisis/37.htm>
- International Organization for Standardization/International Electrotechnical Commission. (2001). *ISO/IEC 9126-1. 2001. Software Engineering-Product Quality. Part 1: Quality model*. Switzerland.
- Lucero Manresa, J. L. (2001). Auditoría de la Calidad. En: Piattini, M. y Del Peso, E. *Auditoría Informática. Un enfoque práctico* (2 ed.) (pp. 361-388). México: Alfaomega Grupo Editor.



"Revista Virtual Universidad Católica del Norte". No. 34, (septiembre-diciembre de 2011, Colombia), acceso: [<http://revistavirtual.ucn.edu.co/>], ISSN 0124-5821 - Indexada Publindex-Colciencias (B), Latindex, EBSCO Information Services, Redalyc, Dialnet, DOAJ, Actualidad Iberoamericana, Índice de Revistas de Educación Superior e Investigación Educativa (IRESIE) de la Universidad Autónoma de México.

McGregor, J. D. (2001). *Testing a Software Product Line. Technical report CMU/SEI-2001-TR-022 ESC-TR-2001-022. Software Engineering Institute – Carnegie Mellon University*. Recuperado de: <http://www.sei.cmu.edu/>

Pressman, R. (2005). *Ingeniería del software, un enfoque práctico*. (6 ed.). México: Mc. Graw Hill.

Rivera, G. (1998). Inspecciones y pruebas de software. En: Centro de Investigación de la Facultad de Ingeniería – Informática. Universidad de los Andes. *Método de desarrollo de sistemas de información e ingeniería de software* (pp. 12-1, 12-52). Bogotá.

Van Zeist, R.H.J. y Hendriks, P.R.H. (1996). Specifying Software Quality with the Extended ISO Model. *Software Quality Journal. Springer Netherlands*, 5(4), 273-284.

Sommerville, I. (2005). *Ingeniería de software*. (7 ed.). México: Addison Wesley.

Tuya, J. (2009). *Hacia el nuevo estándar de pruebas ISO/IEC 29119. Ponencia presentada en la XI Jornada de Innovación y Calidad de Software*. Alcalá de Henares. España. Recuperado de: <http://in2test.lsi.uniovi.es/qt26/presentations/ISO-29119-Javier-Tuya-JICS2009.pdf>

Unidad Docente de Ingeniería del Software. Facultad de Informática. Universidad Politécnica de Madrid. (2008). *Especificación de requisitos según el estándar de IEEE 830*. Recuperado de: <http://ayudantiasubb.files.wordpress.com/2008/06/ieee.pdf>