

Tipo de artículo: Artículo original
Temática: Ingeniería y gestión de software
Recibido: 26/01/2016 | Aceptado: 01/04/2016

Metodología orientada al Proceso de Testing Funcional en la división DATYS-Santiago

Oriented Methodology to Functional Testing Process in DATYS-Santiago division

Deysi Indira Aleaga Bravo ^{1*}, Andrés Veloso Jardón ¹

¹ DATYS, Patricio Lumumba S/N Altos de Quintero Santiago de Cuba, Cuba, {deisy.aleaga, andres.veloso}@datys.cu

* Autor por correspondencia: deisy.aleaga@datys.cu

Resumen

En este artículo, se expone una metodología orientada al proceso para realizar testing funcional de un producto de software. El proceso es aplicado en la división DATYS-Santiago, y se emplea para realizar las pruebas de liberación a los productos. Se presenta como caso de estudio, su aplicación en un proyecto de la división DATYS-Santiago. Las principales características de esta metodología son: se basa en el análisis de riesgos del producto a probar; se prueban primero las partes del producto cuyo funcionamiento incorrecto representan mayor impacto negativo para el negocio; la planificación del proyecto de testing se realiza en función de los ciclos de prueba: cada ciclo de prueba está asociado a una versión del producto a probar. Al finalizar el proceso de pruebas se evalúa, el estado del producto determinando si está listo para su liberación.

Palabras clave: testing funcional, proceso de pruebas, metodología, ciclos de prueba

Abstract

In this article, a process-oriented methodology is presented to perform functional testing of a software product. The process is applied in the division DATYS-Santiago, and is used for testing the product release. It is presented as a case study, his application in a division DATYS-Santiago's project. The main features of this methodology are: is based on risk analysis of product to be tested; product parts which represent malfunction greater negative impact on the business are tested first; project planning of testing is performed according to the test cycles: each test cycle is

associated with a version of the product to be tested. Upon completion of the testing process is evaluated, product's status, determining if it is ready for release.

Keywords: *functional testing, testing process, methodology, testing cycles*

Introducción

Desde el punto de vista del cliente y los usuarios, la calidad de un producto de software es percibida principalmente por las fallas que encuentran en el producto y por la gravedad que éstas tienen para el negocio del cliente. Para ser competitivas, las empresas desarrolladoras de software necesitan asegurarse de la calidad de sus productos previo a su instalación en el ambiente del cliente (Pérez, 2006)

El testing se define como la verificación dinámica del comportamiento de un programa usando un conjunto finito de casos de prueba, seleccionados desde el dominio infinito de ejecución, contra el comportamiento esperado. La verificación dinámica implica que para realizar las pruebas siempre hay que ejecutar el programa para los datos de entrada (Kit, 1995). El testing funcional tiene como objetivo validar si el comportamiento observado del software probado cumple o no con sus especificaciones (Kit, 1995).

Los conceptos, las estrategias, las técnicas y las mediciones del testing son integrados en un proceso definido y controlado que es realizado por personas. Este proceso apoya las actividades de testing y provee guías para el equipo que lo lleva a cabo de forma tal que se alcancen los objetivos propuestos de forma rentable (SWEBOK, 2004)

DATYS¹, es una empresa cubana, de alta tecnología, especializada en el desarrollo de aplicaciones informáticas, ofrece soluciones propias a problemas tecnológicos complejos. Forma parte de una plataforma de integración con universidades y centros de investigación que garantiza el ciclo completo investigación + desarrollo + comercialización, agregando un indiscutible valor a las soluciones que propone.

En la división de Santiago de Cuba se desarrollan los productos dedicados a la Minería de Datos (Hernández, Ramírez, Ferri, 2004). Con el objetivo de lograr mayor impacto de sus productos en el mercado, se decide crear un grupo encargado de realizar pruebas de liberación. Para lograr un trabajo efectivo de este grupo es necesario definir un proceso que guíe las pruebas a realizar y permita evaluar la liberación o no de los productos, ajustándose al

¹ <http://www.datys.cu/>

proceso de desarrollo existente en la división. En este trabajo se propone una metodología para el proceso testing funcional, a emplear en el grupo de Pruebas y Calidad de la división DATYS-Santiago.

Trabajos Relacionados

Los aportes para este trabajo surgen de diferentes fuentes, a continuación, se describen las principales referencias utilizadas.

Rex Black (Black, 2009) presenta una propuesta enfocada en los aspectos administrativos y de gestión del testing: su planificación, seguimiento y control, sin entrar en detalles de cómo realizar el testing.

Por otro lado Beatriz Pérez (Pérez, 2006) , propone ProTest, una guía pensada para una organización que dedicada a la evaluación de productos de software desarrollados por terceros y es el proceso aplicado en el Laboratorio de Testing Funcional del Centro de Ensayos de Software (CES)² de Uruguay. Consta de etapas y actividades detalladas, para la negociación, planificación, seguimiento y control del proceso de testing.

De manera similar a ProTest se describe en la propuesta de Yeniset León, Asnier Góngora Rodríguez y Ailyn Febles (León, Góngora, Febles, 2013), el marco general del proceso de pruebas, empleado en el Laboratorio Industrial de Pruebas de Software (LIPS) del Departamento de Pruebas de Software de (Calisoft). Está diseñado para una organización que realiza testing funcional de productos desarrollados por terceros.

Todas estas propuestas se focalizan en el testing realizado por una organización independiente al equipo de desarrollo. En la metodología que se propone en este trabajo se combinan las actividades, las guías y técnicas de estas propuestas, para ajustarlas a un equipo de pruebas que, aunque pertenece a la organización que desarrolla el software, interviene en este proceso antes de la fecha de liberación.

Etapas y actividades de la metodología

La metodología propuesta consta de tres etapas; Planificación, Ciclos de Prueba y Evaluación del Proyecto, como se muestra en la figura 1. En este capítulo describiremos las actividades y su interacción en el tiempo.

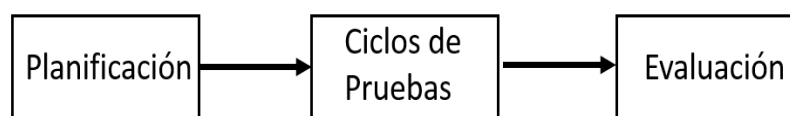


Figura. 1: Etapas de la metodología para el proceso de pruebas.

² <http://www.ces.com.uy/>

Planificación

El objetivo de esta etapa es planificar todo el proyecto de pruebas de un producto. No es posible probar todo, la decisión de qué probar se basa en los riesgos de cada parte del sistema (Pérez, 2006). En la actividad Evaluación de los Riesgos del Proyecto se realiza una lista priorizada de las funcionalidades del producto, junto con los desarrolladores.

Las actividades a realizar en esta etapa son:

- Evaluación de criterios mínimos para comenzar el proceso de pruebas. En esta actividad se verifica que el equipo de desarrollo tenga listo todos los artefactos entregables del producto.
- Evaluación de los riesgos del producto. Al finalizar esta etapa, se tendrá una evaluación de cada funcionalidad del producto según su prioridad y criticidad, en ALTO, MEDIO y BAJO. Permitiendo definir cuáles funcionalidades se probarán y cuáles no.
- Definición del plan de pruebas. Una vez realizada la evaluación de riesgo del proyecto, se definirán los ciclos de pruebas necesarios, y qué funcionalidad será objeto de prueba y en cuál ciclo, como se muestra en la tabla 1. En este análisis se tendrá en cuenta el tiempo mínimo para verificación de errores resueltos y pruebas de regresión.

Como resultado final de esta etapa debe quedar establecido el Plan de Pruebas para la versión del producto. Donde quedará reflejado, fecha de inicio y fin de cada ciclo de prueba, técnicas que se usarán para diseñar los casos de prueba, requerimientos de software y hardware del producto, artefactos que serán entregados por el equipo de desarrollo, al comienzo de cada ciclo y las métricas que se emplearán para evaluar el avance del proyecto y el desempeño del equipo de pruebas.

Tabla 1. Evaluación de funcionalidades y ciclo correspondiente de prueba

Funcionalidad	Prioridad	Criticidad	C1	C2	C3
F1	ALTA	ALTA	x	x	x
F2	MEDIA	ALTA	x	x	
F3	BAJA	MEDIA			x
F4	BAJA	BAJA			

Ciclos de Prueba

Esta etapa tiene como objetivo, ejecutar las pruebas al producto y determinar posibles defectos, antes de su liberación. Durante los ciclos planificados, el equipo de desarrollo se enfocará en la resolución de errores detectados en este ciclo de pruebas o anteriores, de manera tal que, en cada fecha acordada en la etapa de planificación, se le entregue al equipo de pruebas, una versión candidata a liberar del producto con defectos corregidos, ver figura 2.

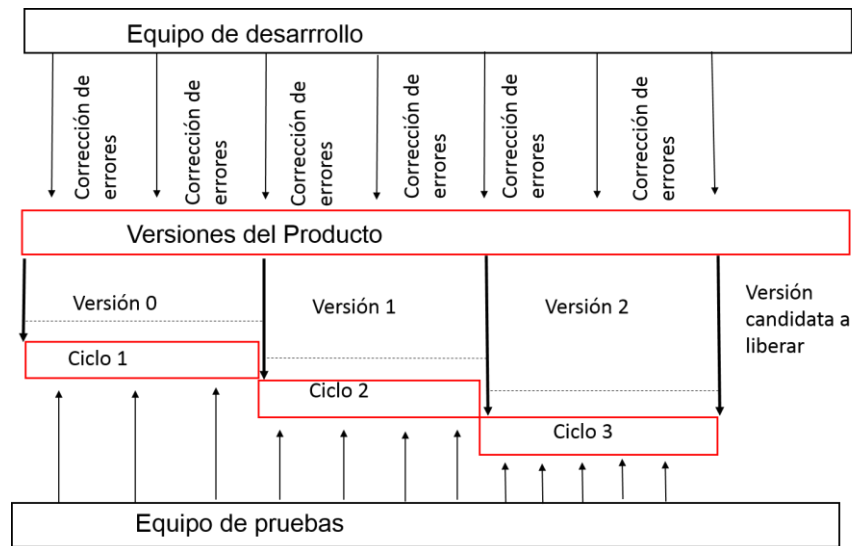


Figura 2. Esquema general de los ciclos de prueba.

La duración de cada ciclo y la cantidad a ejecutarse, varían de acuerdo a la complejidad del producto y al tiempo y personal disponible para realizar las pruebas.

Sin embargo, una semana es un tiempo razonable para ejecutar un número de baterías de prueba y la realización de pruebas exploratorias al producto (Black, 2009).

Las actividades a desarrollar en esta etapa son:

- Configuración del entorno de pruebas. Esta actividad, es de suma importancia, pues el entorno de ejecución del producto, debe ser independiente del entorno de desarrollo y simular lo más posible, el entorno real de los clientes.
- Diseño de las pruebas. Utilizando las fuentes de especificación de requerimientos (manuales, historias de usuario, descripción de las funcionalidades, pruebas de aceptación, etc), cada tester deberá diseñar las pruebas que

ejecutará durante este ciclo.

- Ejecución de pruebas de humo (Standard Glossary, 2012). Son un conjunto de pruebas aplicadas a cada nueva versión, su objetivo es validar que las funcionalidades básicas de la versión se cumplen según lo especificado. Estas pruebas buscan grandes inestabilidades o elementos claves faltantes o defectuosos, que hacen imposible realizar el testing como fue planificado para el ciclo. Si la versión no pasa las pruebas de humo, no se comienza la ejecución de las pruebas de la versión, quedando comprometida la liberación del producto (Kaner, 2002).

Ejecución de las pruebas. En esta actividad deben ejecutarse los casos de prueba diseñados para este ciclo, así como reportar los defectos encontrados. Especificando su categoría (Error o Sugerencia), su severidad (Alto, Medio o

- Bajo) y las condiciones para su reproducción, en caso de ser posible.
- Ejecución de pruebas exploratorias. Esta técnica le permite al tester interactuar con el producto sin restricciones, usando la información que el sistema provee, para cambiar el curso de las pruebas y en general explorar las funcionalidades (Whittaker, 2009). Identificando salidas del sistema, que no han sido identificadas en las etapas de planificación (Crispin, 2009).
- Revisión de las correcciones. Como en cada versión el equipo de desarrollo deberá corregir un conjunto de defectos, durante esta tarea es responsabilidad del equipo de pruebas validar si ciertamente fueron corregidos.
- Ejecución de pruebas de regresión. Los tester deberán asegurarse que la corrección de errores no ha introducido algunos nuevos. Para ello deberán seleccionar un subconjunto de casos de pruebas, asociados a las funcionalidades donde se han corregido errores y ejecutarlos. Los defectos encontrados deberán reportarse (Pressman, 2005).

Evaluación del proyecto

Esta etapa tiene como objetivo evaluar y emitir un informe del estado del proyecto, así como realizar las mejoras necesarias al proceso de pruebas. Para ello se realizarán las siguientes actividades:

- Cálculo de las métricas del producto. En esta actividad se recopilarán los datos de las pruebas realizadas durante cada uno de los ciclos pruebas y de ser necesario, de las pruebas realizadas en versiones previas del producto. Se calcularán las métricas propuestas en el plan de pruebas aprobado en la etapa de planificación.
- Cálculo de las métricas del proceso de pruebas. En esta actividad se recopilarán todos los datos referentes al comportamiento de las pruebas con el objetivo de ir calibrando el proceso de pruebas.
- Elaboración del informe final del proceso de pruebas. En esta actividad el jefe del equipo de pruebas,

elaborará un informe con los resultados del proceso de pruebas del proyecto. Este informe debe incluir, todos los incidentes reportados que pudieron afectar el proceso de pruebas, los resultados de las métricas calculadas (tanto del producto como del proceso), un criterio de liberación o no del producto, así como las principales deficiencias detectadas en el proceso.

Aplicación de la metodología

Esta metodología es aplicada en el Grupo de Pruebas y Calidad de la división DATYS-Santiago. Todos los productos que se desarrollan en la división, han sido evaluados empleando esta metodología desde la creación del grupo.

El sistema que utilizaremos como caso de estudio en este artículo tiene poco tiempo en producción. Cuenta con manual de usuario y de instalación actualizados, y como material de apoyo un conjunto de pruebas de aceptación. No existe un documento con la especificación de requerimientos. Se aplicaron para este proyecto cada una de las etapas y actividades de la metodología propuesta, como se describe a continuación.

Durante la etapa de Planificación, el análisis de los riesgos del proyecto es la actividad más importante, para planificar el trabajo de testing y definir los ciclos de prueba del producto. Para cada ciclo de prueba debe definirse qué funcionalidades se probarán y cuándo comienza y termina el ciclo. La definición de las funcionalidades a probar durante el proceso de pruebas, se realiza agendando en los ciclos, las funcionalidades según la importancia asignada en el análisis de riesgo del producto, dado que cada ciclo se corresponde con una versión del producto. Se definieron 4 ciclos con 1 semana de duración cada uno.

Las funcionalidades para el proyecto se clasificaron de conjunto con el jefe del equipo de desarrollo según su Prioridad y Criticidad, y se asignaron en cada ciclo, agrupadas según el módulo al que pertenece en el sistema como muestra la tabla 2. En total fueron analizadas 80 funcionalidades. En el ciclo 4 no se incluyeron funcionalidades, fue dejado exclusivamente para la revisión de la corrección de errores y pruebas de regresión.

La definición del Plan de Pruebas ha resultado de gran utilidad para que tanto el equipo de desarrollo como el de pruebas, tenga visibilidad respecto al trabajo del proyecto de testing, así como lograr acuerdos respecto a responsabilidades.

Tabla 2. Distribución de funcionalidades agrupadas en Módulos por cada ciclo.

Prioridad	Criticidad	C1	C2	C3
ALTA	ALTA	55	41	41
MEDIA	ALTA		8	
MEDIA	BAJA		15	

BAJA	BAJA			2
------	------	--	--	---

Una vez planificado el proyecto de prueba, se comenzaron los Ciclos de Prueba. En cada uno de ellos, se realizó la instalación y configuración del producto en el ambiente de pruebas. El diseño de las pruebas se realizó, a partir de las pruebas de aceptación y los resultados de las pruebas exploratorias. Se empleó para el control de las ejecuciones y el mantenimiento de los casos de prueba, la herramienta Klaros-TestManagement³. En la tabla 3 se resume el número de defectos encontrados en cada ciclo, según su severidad. Estos defectos fueron previamente aceptados por el equipo de desarrollo.

Tabla 3. Defectos encontrados en cada ciclo de prueba.

Ciclo	Altos	Medios	Bajos
C1	4	10	19
C2		5	9
C3		14	15
C4	2	1	6

En el Plan de Pruebas de este proyecto, fue acordado el uso de las métricas: Densidad de Defectos (Kan, 2002), Pesado de la Densidad de Defectos, Porcentaje de Resolución de Defectos (Pandian, 2005) y Tiempo Promedio de Resolución de Defectos. En conjunto permiten determinar el estado del producto y la efectividad del equipo en la resolución de defectos.

En la etapa de Evaluación, se recolectaron los datos necesarios para el cálculo de las métricas acordadas. Para mostrar los resultados obtenidos, utilizaremos el Pesado de la Densidad de Defectos, la cual pondera el impacto real de los defectos encontrados según su severidad, en función del tamaño del producto, fórmula (1).

$$\frac{(5 * \text{Altos} + 3 * \text{Medios} + \text{Bajos})}{\text{Tamaño}} \quad (1)$$

donde:

Altos: Defectos de alta prioridad reportados en el ciclo,

Medios: Defectos de media prioridad reportados en el ciclo,

³ <http://www.klaros-testmanagement.com/>

Bajos: Defectos de baja prioridad reportados en el ciclo y

Tamaño: Tamaño del producto, medido en cantidad de funcionalidades.

En la figura 3, se representan los resultados del proyecto, durante los ciclos de prueba, utilizando la métrica Pesado de la Densidad de Defectos, a partir de los defectos pendientes, una vez concluida la actividad de Revisión de las correcciones en cada ciclo.

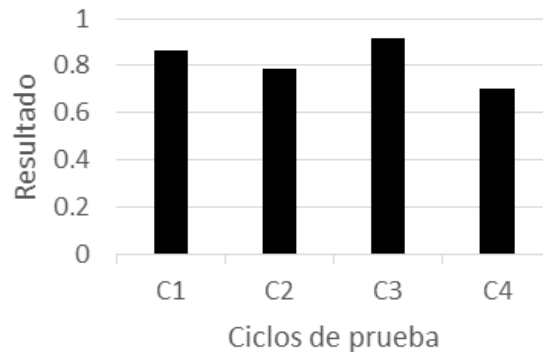


Figura 3. Cálculo de la métrica Pesado de Densidad de Defectos, durante los ciclos de prueba.

A medida que avanzaba el proyecto de pruebas, se esperaba, que el producto tendiera a disminuir la cantidad de defectos reportados. Ese es el caso de los valores que muestra la Figura 3, en los ciclos 1, 2 y 4. Sin embargo en el ciclo 3, el proyecto alcanzó un resultado superior a los anteriores. Este tipo de variaciones, pueden presentarse cuando, se comienzan las pruebas de un conjunto de funcionalidades, que no han sido probadas previamente y su análisis de riesgo para el proyecto, no fue acertado. Al corregirse un defecto, no se realizan las pruebas unitarias y de integración correspondientes, entre otros casos. Estas circunstancias deben ser controladas rápidamente, pues podría comprometerse la liberación del producto.

Finalmente, el equipo de desarrollo logró solucionar el 69 % de los defectos reportados, disminuyendo el Pesado de la de Densidad de Defectos de 2.16, al iniciar el proceso, a 0.74 lo que representa aproximadamente 2 defectos de baja prioridad por cada 3 funcionalidades. Este valor se consideró aceptable por la administración de la división y se decidió liberar el producto.

Conclusiones

En este trabajo se propone una guía para realizar testing funcional de un producto de software y su aplicación en el Grupo de Pruebas y Calidad de la división DATYS-Santiago. Las principales características de esta metodología se listan a continuación:

Se basa en el análisis de riesgo del producto para priorizar las funcionalidades a probar, esto requiere involucrar a los desarrolladores en ese análisis y responsabilizarlos de tomar decisiones durante las pruebas. Los ciclos de prueba son el punto central de cada proyecto de testing. La decisión de las pruebas a ejecutar en cada ciclo se realiza a partir del análisis de riesgo del producto. En cada ciclo de prueba se realiza testing exploratorio además del testing planificado para el ciclo.

Al finalizar el proceso de testing se emite una evaluación del producto empleando métricas que ponderan la cantidad de defectos encontrados, con el tamaño del producto y la eficiencia en la resolución. Como trabajo futuro se encuentra añadir en la etapa de Evaluación un conjunto de criterios de liberación que agrupen las métricas aplicadas y otros criterios de interés administrativo.

Referencias

BLACK R.: Managing the Testing Process, 3rd Edition. Ed. Willey, 2009.

CRISPIN, L., GREGORY, J.: Agile Testing: A Practical Guide for Testers and Agile Teams. Ed. Addison-Wesley, 2009.

Guide to the Software Engineering Body of Knowledge (SWEBOK), pp. 73-86. IEEE Computer Society, 2004.

HERNÁNDEZ J., RAMÍREZ MA. J., FERRI C.: Introducción a la Minería de Textos, Pearson Educación S.A, 2004

KAN, S. H.: Metrics and Models in Software Quality Engineering. Ed. Addison Wesley, 2002.

KANER C., BACH J., PETTICHORD B.: Lessons learned in Software Testing. Ed. Willey, 2002.

KIT E.: Software Testing In The Real World: Improving The Process. Ed. Addison Wesley, 1995.

LEÓN, Y., GÓNGORA, A., FEBLES, A.: “Aplicando métricas de calidad a proyectos y procesos durante las pruebas exploratorias”, Revista Cubana de Ciencias Informáticas, Vol. 7, No. 2, pp. 36-48, La Habana, 2013.

PANDIAN, C. R.: Software Metrics: A Guide to Planning, Analysis, and Application. Ed. Auerbach Publications, 2005.

PÉREZ, B.: "Proceso de Testing Funcional Independiente", Tesis de maestría, Facultad de Ingeniería Universidad La República, Montevideo Uruguay, 2006.

PRESSMAN, R. S.: Ingeniería del software: Un enfoque práctico (6ta Edición). Ed. McGraw Hill, 2005.

Standard Glossary of terms used in Software Testing. Version 2.2. International Software Testing Qualifications Board, 2012.

WHITTAKER J.: Exploratory Software Testing. Ed. Addison Wesley, 2009.