

Universidad de las Ciencias Informáticas

**Procedimiento para el diseño e implantación de pruebas a partir
del análisis de los riesgos del producto.**

Tesis en opción al título de
Máster en Ciencias

Por

Autor: Ing. Heydi Menéndez Ávalos

Tutor: Dra. Ailyn Febles Estrada

Tutor: Msc. Ramsés Delgado Martínez

La Habana, Cuba

Abril de 2012

Agradecimientos

Quiero agradecer a mis padres que lo son todo, a quienes quiero con mi vida porque se las debo entera.

A mi hermana y su familia porque gracias a ella también soy mejor persona.

A mi tía y su familia por acogerme en su hogar y estar al tanto de mí hasta el cansancio.

A Ramsés por pelearme, halarme las orejas de vez en cuando y sobre todo por ser incondicional y ayudarme.

A mis compañeros del DPSW por preocuparse por la tesis y por mí, por contribuir con su trabajo y sus críticas también a este resultado que más que personal pienso nos ayude a todos en nuestra lucha por el pan, día a día.

A la Viola por el tema, y por ser mi guía por mucho tiempo, presencial y a distancia, siempre dispuesta. Gracias por ser tan violenta.

A Tay por comprender y ser la mejor jefa de por vida. Gracias por la ayuda de siempre.

A Yadira por todas las veces que discutí mis dudas con ella, quien a pesar de la poca experiencia que tiene para mí es de las más grandes en estos temas.

A Alberto por la herramienta, por las noches sin sueño, por la preocupación y el cariño por las pequeñas cosas que espero crezcan mucho y nos lleven lejos.

A mis amigos Julie, Martha, Delvis, Asnier y Yosdenis por estar siempre ahí.

A mi mamá adoptiva Ana, espero te sientas orgullosa siempre de esta hija y ojalá Anita siga mi ejemplo.

A los expertos y personal que me asesoró y apoyó en la realización del trabajo.

A las tres patas de la mesa por ser mis compañeras de siempre, aunque me hayan desplazado, yo las entiendo, lo único que quiero es que sus niñas al menos me llamen tía.

A mi compañera D y su novio, por estar presente en las buenas y malas.

A mi tutora por el liderazgo y por comprender también.

En fin...a todos los que me tienen siempre presente. Gracias.

Resumen

Las pruebas de software en su papel como requisito fundamental para garantizar la calidad de un software, deben alcanzar la variedad, nivel de exigencia y profesionalidad justa para lograr su propósito exitosamente, sin despreciar la automaticidad de las mismas que asegura el cumplimiento de un proceso lógicamente definido por especialistas que han consolidado y dominado los conocimientos necesarios para la ejecución de las pruebas. Comprobando el cumplimiento de los atributos o características de calidad según las normas y modelos internacionales.

El centro Calisoft ha definido varios procesos para garantizar la calidad del software desde el inicio del proceso de desarrollo y para todas las etapas del ciclo de vida del mismo. De igual forma ha definido un proceso de pruebas de liberación para evaluar los sistemas que se van a entregar al cliente, en caso de que cuenten con la calidad requerida.

El presente trabajo propone una alternativa al proceso de pruebas de liberación. Teniendo en cuenta algunos aspectos mejorables del proceso anterior, se definen nuevos artefactos y actividades con la garantía de minimizar el tiempo de evaluación integral del software según el cumplimiento de las características de calidad, y basar el proceso en los riesgos que puede tener la calidad del producto.

Índice de Contenidos

Introducción	1
1 Capítulo 1. Fundamentación Teórica	10
1.1 Introducción	10
1.2 Calidad de software	10
1.3 Calidad del producto	11
1.4 Normas, modelos y atributos para la calidad del producto	12
1.5 Riesgos.....	17
1.5.1 Definiciones.....	17
1.6 Gestión de riesgos.....	19
1.6.1 Gestión de riesgos de un software	20
1.6.2 Gestión de riesgos como área de proceso de CMMI.....	20
1.6.3 Herramientas para la gestión de riesgos.	21
1.7 Pruebas de software.....	21
1.7.1 Proceso de Pruebas	22
1.7.2 Tipos de Pruebas.	24
1.7.3 Técnicas de prueba	25
1.8 Herramientas para la ejecución de pruebas	28
1.8.1 Herramientas Manuales	28
1.8.2 Herramientas Automatizadas.....	28
1.9 Pruebas basadas en riesgo	29
1.10Técnicas para analizar el riesgo	31
1.10 Conclusiones	35
2 Capítulo 2.Análisis del Proceso de Pruebas de Liberación y Procedimiento propuesto para las Pruebas Basadas en Riesgos a la Calidad del Producto.	37
2.1 Antecedentes.....	37
2.2 Descripción general del funcionamiento del LIPS	37
2.3 Procedimiento para las Pruebas Basadas en Riesgos a la Calidad del Producto. ...	40
2.3.1 Fases y Actividades.....	42
2.3.2 Roles y Responsabilidades	43

2.3.3 Artefactos	43
2.3.4 Procedimiento	43
2.4 Otra propuesta de aplicación del procedimiento.....	50
2.5 Conclusiones	51
3 Validación de la Propuesta y Presentación de los Resultados.....	53
3.1 Introducción	53
3.1.1 Método Delphi para la Validación de la propuesta.....	53
3.1.2 Estudio de casos para la Validación de la propuesta	54
3.2 Validación de la Propuesta con Delphi	54
3.2.1 Objetivo de la validación.....	54
3.2.2 Muestra	55
3.2.3 Instrumento	55
3.2.4 Procedimiento	55
3.2.5 Conclusiones de la Validación por Delphi.....	61
3.3 Validación por Casos de Estudio.	62
3.3.1 Muestra	62
3.3.2 Analisis de las Pruebas Basadas en Riesgos.....	66
3.3.3 Conclusiones de la Validación	70
3.4 Conclusiones del Capítulo	71
4 Conclusiones	73
5 Recomendaciones	74
6 Referencias Bibliográficas	75
7 Anexos	77
7.1 Anexo1 Tabla para el Cálculo del Impacto del Riesgo	77
7.2 Anexo2 Tabla para el Cálculo de la Probabilidad del Riesgo	77
7.3 Anexo3 Tabla para el Cálculo de la Prioridad del Riesgo	77
7.4 Anexo4 Documento de Análisis de Riesgos	79
7.5 Anexo4 Lista de Chequeo de Riesgos.....	80
7.6 Cuestionario Definitivo.....	81

Índice de Ilustraciones

Ilustración 1.Estructura de Calisoft (Creación propia).....	2
Ilustración 2.Departamento de Pruebas de Software (Creación propia)	3
Ilustración 3.Modelo de Mc Call(1977) tomado de (Palazzolo, 2005)	14
Ilustración 4.Modelo de Boehm(1978)tomado de (Palazzolo, 2005)	15
Ilustración 5. Los trabajadores y artefactos involucrados en las pruebas tomado de (Jacobson et al. 1999).....	24
Ilustración 6.Actividades de las Pruebas Basadas en Riesgos (Creación propia).....	31
Ilustración 7.Proceso de Pruebas de Liberación tomado de (Capote,2011)	39
Ilustración 8.Proceso de Pruebas Basadas en Riesgos (Creación propia).....	41
Ilustración 9.Procedimiento de Pruebas Basadas en Riesgos para el LIPS (Creación propia).	42
Ilustración 10 Valoración de Expertos.....	57
Ilustración 11 Resultados del Cuestionario	59
Ilustración 12 Riesgos del Proyecto1	63
Ilustración 13 Riesgos del Proyecto2.....	64
Ilustración 14 Riesgos del Proyecto3.....	66
Ilustración 15 Resumen de No Conformidades	66
Ilustración 16 Resumen de No Conformidades	68
Ilustración 17 Resumen de No Conformidades	69
Ilustración 18 Comparación entre Proyectos.	69
Ilustración 19 Por Ciento de No Conformidades asociadas a riesgos.	70

Índice de Tablas

Tabla 1. Características de la ISO 9126(1997) y su correspondencia con la 25000(2005) tomado de la norma.....	16
Tabla 2. Características de la ISO 25000 tomado de la norma.....	17
Tabla 3.Comparación de las técnicas de análisis de riesgo tomado de (Black,2008)	34
Tabla 4.Resultados del cuestionario definitivo.	61
Tabla 5.Análisis de riesgos del Proyecto1.....	63
Tabla 6.Análisis de Riesgos del Proyecto2	64
Tabla 7.Análisis de Riesgos del Proyecto3	66

Introducción

Los riesgos históricamente han sido un elemento preocupante en la construcción y puesta en marcha de cualquier producto, debido a que involucra cada parte, proceso, persona. Desde cualquier punto de vista la inversión actual de cada proyecto en la gestión, mitigación, y elaboración de planes de contingencia para reducir el impacto que estos puedan causar en cualquier aspecto de la industria, resume grandes cantidades de dinero. El desarrollo de cualquier producto informático no se encuentra exento de riesgos. Estos aumentan con la complejidad, el número de participantes, esfuerzo, presupuesto y duración y están presentes en todo tipo de actividad que ha hecho el hombre. Analizando varios indicadores presentes en la bibliografía disponible se puede concluir diciendo que:

- Las probabilidades de fracaso de un proyecto de software pueden manifestarse durante todo el ciclo de desarrollo y estadísticamente están en los límites del 2% al 85%. (Jones, 1995)
- Las malas pruebas son consideradas como una de las cuatro causas fundamentales del fracaso junto con la estimación pobre, la mala planificación y el mal seguimiento del proyecto (tracking).
- La gestión de riesgos clásica dice de mitigar estos utilizando ambos métodos- proactivo (entrenamiento cruzado del personal) y reactivo (como la codificación de componentes redundantes) (reutilización).
- Los riesgos en la calidad del sistema pueden ser mitigados empleando como método proactivo las revisiones y como reactivo las pruebas.
- El análisis de riesgos de la calidad es un proceso para la identificación, el análisis, la categorización prioritaria de los problemas potenciales de la calidad de un sistema determinado. Una vez que los riesgos son identificados, cada uno es asignado a un nivel (en la medida de su grado de importancia). (Análisis de Riesgos a la Calidad, 2008)

Un riesgo es un resultado potencial indeseable. Por lo que un riesgo a la calidad del sistema es cualquier problema potencial que podría hacer que el sistema fallara en encontrar las expectativas razonables de calidad. Los riesgos se agrupan u organizan por categorías como

son: funcionalidad, performance, seguridad, etc. (Procesos Críticos de Prueba, 2003)

En el año 2002 es creada la Universidad de las Ciencias Informáticas (UCI) .Existe una tendencia a que los proyectos desarrollados en la UCI sufran afectaciones en su cronograma, en la calidad del producto, en el presupuesto y en la formación de los estudiantes. Una variable significativa en la Gestión de Proyectos es la Gestión de Riesgos y se considera que “el nivel de calidad de todo proyecto depende de la minimización del número de riesgos al que se enfrenta el mismo” (Análisis de Riesgos a la Calidad, 2008).

En el año 2004 es creada en la UCI la Dirección de Calidad de Software. Posteriormente, en el 2008, atendiendo a la evolución de sus actividades, se transformó en el Centro Nacional de Calidad de Software (Calisoft). Dicho centro tiene entre sus funciones normar el desarrollo de software, realizar revisiones técnicas a lo largo del ciclo de vida de los proyectos y revisar los productos entregables a los clientes.

Calisoft presenta una estructura de organización conformada por tres grupos, tres subdirectores y un director.

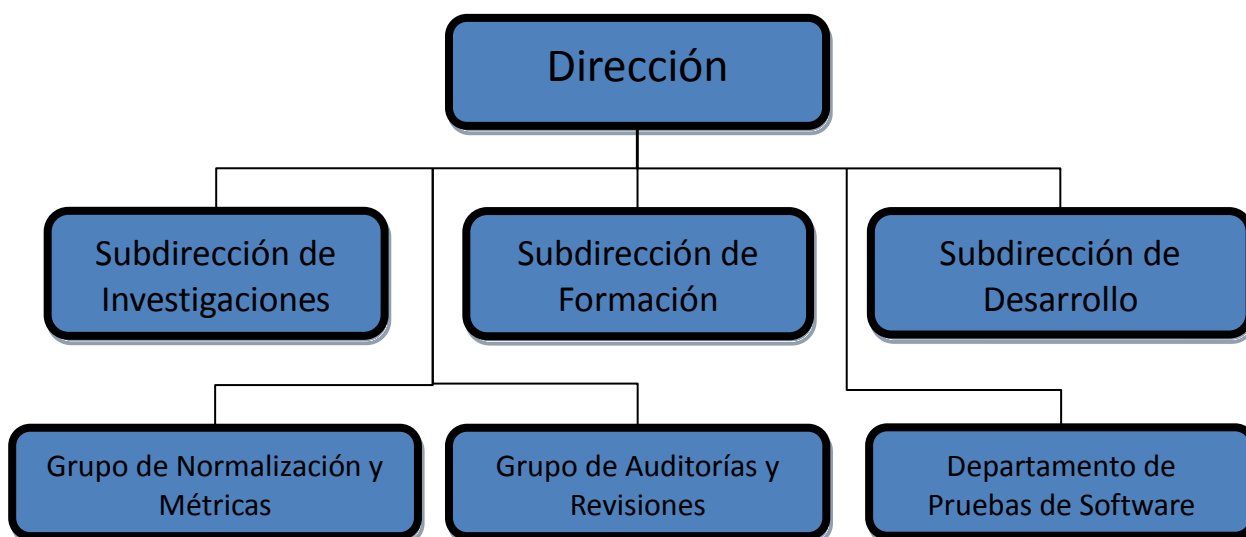


Ilustración 1. Estructura de Calisoft (Creación propia)

Dentro de las actividades de normalización y estandarización, Calisoft ha dedicado grandes esfuerzos a la definición de procesos claves para el desarrollo de software empezando por los relacionados con la Gestión de Proyectos.

La Gestión de Riesgos se considera a nivel mundial una de las disciplinas con un peso fundamental en el aseguramiento de la calidad del producto, pues permite desde los inicios establecer una visión de los problemas que puede presentar el software en sus etapas de desarrollo y puesta en marcha.

El Departamento de Pruebas de Software (DPSW) de Calisoft es quien se encarga de realizar las evaluaciones al software producido y a la documentación básica del mismo, ejecutando de esta forma distintos tipos de pruebas que validen la calidad del software en términos de funcionalidad, seguridad, fiabilidad, usabilidad, mantenibilidad, eficiencia, soporte, efectividad, productividad y satisfacción(ISO –IEC 25000).Este grupo consta de dos estructuras lógicas internas para separar los temas investigativos de los productivos.

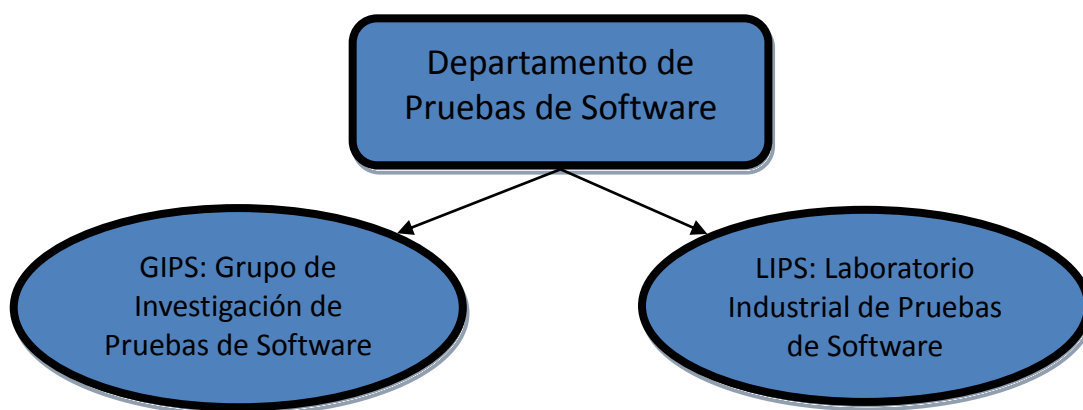


Ilustración 2. Departamento de Pruebas de Software (Creación propia)

El Grupo de Investigación de Pruebas (GIPS) está conformado por todos los especialistas del DPSW y su objetivo es investigar los tipos de pruebas a realizar a las soluciones informáticas y productos y que verifiquen las características de calidad (ISO/IEC 25000:2005). El LIPS es la estructura que funciona como un laboratorio y que ejecuta de manera industrial las pruebas que ya están definidas, empleando los métodos y herramientas definidos por el

GIPS. Las pruebas que se llevan a cabo en Calisoft se enmarcan generalmente en la verificación de los requisitos (especificados mediante casos de uso o historias de usuario) contra lo implementado. Partiendo en pocos casos de la prioridad de los casos de uso para concederle mayor o menor importancia a la prueba unitaria del mismo y que de esta forma se pueda controlar el trabajo(en cuanto a cantidad de requisitos revisados) que emplea cada especialista en dichas pruebas, incluso en algunos proyectos solo se pueden verificar las funcionalidades específicas que se implementaron lo cual permite igualmente comprobar que cada una funciona correctamente pero tampoco precisa desde un inicio el tiempo que se va a invertir en la prueba de cada una, ni cuantos especialistas u otros recursos , serán necesarios para las pruebas. Actualmente las pruebas se desarrollan teniendo en cuenta la fecha de entrega del proyecto. Esta forma de trabajo influye en que aparezcan determinadas complicaciones como son:

- La asignación de tiempo a las pruebas mediante la estimación basada en la experiencia muchas veces no se ajusta del todo al tiempo real pues no se tiene en cuenta el impacto que cada funcionalidad pueda tener por individual en el software y por ende el riesgo que esto pueda constituir para la calidad del mismo.
- La distribución de los especialistas en las pruebas en muchas ocasiones puede exceder el esfuerzo necesario para una prueba específica o en otras no bastar, pues no se calcula con exactitud el dato real teniendo en cuenta los atributos o características de calidad que se probarán y los conocimientos del probador a asignar en el tema. Para el caso de todos los proyectos solo se tiene en cuenta la disponibilidad del especialista para realizar la prueba y en algunos se asignan por el tamaño del proyecto más de un especialista de manera que la distribución del trabajo sea equitativa entre los que trabajen la prueba pero no existe un indicador que apunte hacia la experiencia del especialista y permita designarlo por sus conocimientos. Solo en pocos casos se hace el esfuerzo de mantener un especialista al frente de un proyecto porque ya ha trabajado con anterioridad en el mismo y le resultara más familiar, conociendo los antecedentes en productos liberados.
- En algunas pruebas emplean tiempo y especialistas y se prolonga la iteración de pruebas en reportes de defectos que afectan más la estética, la calidad de la interfaz

y la correspondencia con la documentación básica y de apoyo a las pruebas , que en un final son más fáciles de arreglar, que otros defectos que afectan directamente la calidad del desempeño del sistema y realmente requieren de tiempo para solucionarse y más esfuerzo para comprobar si funcionan correctamente y esto se evidenció en un estudio hecho en tres proyectos que estaban liberándose en el LIPS: un portal que tenía 35 mercados de datos para un total de 125 casos de prueba se detectaron en la 1era iteración 297 No conformidades y de ellas 176 eran de correspondencia entre los casos de uso y lo implementado, el resto se repartieron entre ortografía , formato y redacción, excepto 5 que fueron de funcionalidad. Para el caso de otro sistema analizado solo habían 18 casos de prueba y en la primera iteración se detectaron 15NC de las cuales 2, fueron referente a los requisitos de configuración del sistema, 3 de formato, 5 de validaciones y 5 de funcionalidad, este era un sistema de escritorio muy sencillo. En un tercer producto cuyo objetivo era generar reportes se detectaron en la primera iteración 19NC de las cuales, solo 5 fueron de validación, el resto estuvo relacionado con la correspondencia con otro artefacto, errores ortográficos y errores técnicos en la documentación de apoyo.

Resumen del Análisis efectuado en el LIPS con los especialistas.

El 25% de los errores detectados no son referentes a la calidad del producto.

El 15% de la asignación de los especialistas no es eficiente.

El 80% de la duración estimada para el proceso no se cumple.

- No se emplea en ningún momento la gestión de los riesgos del producto en las pruebas de software lo cual solo permite ver las funcionalidades que podrían fallar luego de implementadas en el sistema, perdiendo así la oportunidad de brindarles puntualmente atención diferenciada previo a la Prueba de Liberación.

Estas situaciones se deben en su mayoría a que no se tiene en cuenta un análisis de los riesgos que cada producto puede acarrear y sus consecuencias durante el proceso de pruebas.

La actual organización en la preparación y superación en el grupo de Ingeniería de Pruebas del DPSW, es que cada especialista o grupo de especialistas, gestione el conocimiento y la ejecución de las pruebas guiadas que atienden a una característica o atributo específico de calidad. De esta forma se verifica a fondo cada una de las funcionalidades asociadas a la característica que atiende el especialista.

Luego de una observación detallada de los servicios de pruebas de software que ofrecen varias organizaciones, en varias destacó el empleo de los riesgos como parte de las pruebas para minimizar y corregir los problemas que presenta Calisoft. Empresas como Infosys, RBCS, Acustest, Inspire Quality Services(IQS), Sonata, entre otros se inclinan por el desarrollo de las pruebas basadas en los riesgos del producto para lograr una valoración del mismo en un corto periodo de tiempo , así como detectar la mayor cantidad de errores estableciendo prioridades para su solución a partir de la probabilidad de ocurrencia de los mismos y el impacto que pueden tener sobre el sistema. En el DPSW muchas veces resulta difícil probar cada funcionalidad por lo que sería ideal escoger un grupo de todo el conjunto de pruebas a desarrollar y funcionalidades a comprobar .El Análisis de Riesgos puede contribuir a enfocar el esfuerzo puesto en las pruebas considerablemente y el DPSW cuenta con el personal calificado para realizar estas pruebas, además del conocimiento y las herramientas necesarias para acometer este servicio. Aun así existe un inconveniente de suma importancia y es que no existe una guía o forma de proceder documentada para realizar el análisis de los riesgos que se probarán y cómo será la ejecución de la prueba por parte de los especialistas que cumplen con este rol.

Atendiendo a la situación problemática identificada anteriormente se define como **Problema de la investigación:**

- ¿Cómo realizar un proceso de pruebas de software basadas en los riesgos del producto ,con mayor calidad ,que contribuyan a reducir los tiempos de duración del proceso, a partir de la identificación de las funcionalidades más vulnerables del

software , y verifique el cumplimiento con la ISO/IEC 25000:2005?

Como **Hipótesis** se plantea que:

- Con la implantación de un procedimiento para aplicar pruebas basadas en riesgos en el LIPS aumentará la calidad de las pruebas ejecutadas ,reducirá el tiempo de desarrollo de las mismas y permitirá brindar una evaluación completa en cuanto a cumplimiento de la norma ISO/IEC 25000:2005.

Para enfrentar el problema se toma como **Objeto de Estudio:**

- Proceso de gestión de riesgos en las pruebas de software

Y como **Campo de Acción:**

- Laboratorios de pruebas de software

Para responder al problema enunciado se define como **Objetivo general:**

- Definir un procedimiento para diseñar e implementar pruebas basadas en riesgos del producto que permita reducir los tiempos de duración del proceso de pruebas de software, aumente la calidad del mismo y permita brindar una evaluación del software en cuanto al cumplimiento de la norma ISO/IEC25000:2005.

Para dar cumplimiento al objetivo general se definen como **Objetivos específicos:**

- Analizar las concepciones teóricas sobre las pruebas basadas en riesgos y su aplicación práctica.
- Definir un procedimiento para aplicar las pruebas basadas en riesgo a los productos que sean solicitados al DPSW.
- Implementar artefactos que permitan determinar el alcance de las pruebas y los recursos necesarios para la misma.
- Evaluar los resultados de la implantación del procedimiento.

Tareas de la investigación:

- Estudiar las técnicas de pruebas basadas en riesgos y su aplicación práctica.
- Elaborar procedimiento para aplicar las pruebas basadas en riesgo al producto informático.
- Aplicar el procedimiento definido a un producto.

- Comparar los datos de la validación con los históricos.

Métodos de investigación:

Los métodos empíricos utilizados en la investigación son:

- El grupo focal para realizar discusiones con los especialistas del Laboratorio de Pruebas de Calisoft con el objetivo de identificar los problemas esenciales que afectan el buen desempeño de las pruebas de software y que contribuirán al refinamiento de los artefactos para la ejecución del procedimiento propuesto.
- La observación participativa para obtener información de las discusiones en sesiones Científicas y encuentros que se realicen en entornos que aporten información para la Investigación.
- El estudio de casos para investigar a fondo el problema científico y plantear la solución que más justa le sea a partir del estudio realizado.
- La encuesta para determinar la opinión de los especialistas del LIPS sobre la efectividad del procedimiento.

- El método de la entrevista para investigar sobre las técnicas de análisis de riesgo trabajadas en la Universidad y sus beneficios aportados.
- El experimental para validar los resultados obtenidos a partir de la aplicación del procedimiento.

Y los métodos teóricos:

- Método de análisis e histórico lógico para el estudio de artículos relacionados con el tema y los antecedentes de la solución actual de manera que posteriormente a la implementación el procedimiento se pueda contar con datos comparativos.

- El analítico sintético para descomponer el problema de investigación en elementos por separado y profundizar en el estudio de cada uno de ellos, para luego sintetizarlos en la solución de la propuesta.
- Enfoque sistémico, para lograr que los elementos que forman parte del modelo sean un todo que funcione de manera armónica.
- La modelación que permitirá establecer un marco a seguir en el diseño y aplicación del procedimiento que será propuesto.

Resultados a alcanzar

El trabajo servirá como guía en el Laboratorio Industrial de Pruebas de Software para la ejecución de las Pruebas Basadas en Riesgos, mediante el establecimiento de un procedimiento detallado en fases y actividades que permita ejecutar pruebas de software con mayor organización y mejoren la calidad del sistema a liberar.

Aporte Científico

En la universidad no se ha presentado este procedimiento con anterioridad donde se aborden las técnicas de análisis de riesgos ya diseñadas por empresas internacionales y que pretende proponer un modelo para las pruebas de los proyectos de la universidad evitando así la detección de errores en etapas tardías y elevando de esta forma la eficiencia, productividad y sobre todo la calidad de los productos desarrollados en la universidad.

Estructura del documento

El trabajo constará de 3 capítulos: un primer capítulo enfocado en la fundamentación teórica y estado del arte del tema investigado, un segundo capítulo donde se expondrá la propuesta de solución y un tercer capítulo donde se registrarán los resultados de la aplicación de la propuesta a un producto de un proyecto de la universidad.

1 Capítulo 1. Fundamentación Teórica

1.1 Introducción

En el presente capítulo se presentan los conceptos fundamentales relacionados con las pruebas de software y el análisis de riesgos, además se explican varias técnicas para dicho análisis resultado de la investigación efectuada para dar cumplimiento a los objetivos planteados al inicio de la misma. Se hace un estudio del tema tanto en el ámbito mundial, como en nuestro país y en la Universidad específicamente.

Además se describen procesos básicos en el aseguramiento de la calidad de un sistema como es la Gestión de Riesgos, así como sus beneficios y principales actividades. Varias técnicas para el análisis de riesgo son explicadas brevemente con el objetivo de potenciar el interés del lector en el tema y propiciar nuevas investigaciones sobre los proyectos adecuados para cada técnica específica.

1.2 Calidad de software

Existen muchas definiciones para este término debido a la variedad de ideas que se pueden resumir en el personal vinculado a estos temas, en dependencia de su punto de vista y del entorno donde se desarrolle. Es sumamente difícil establecer un concepto abarcador que satisfaga todas las variantes de lo que se entiende por calidad en el software.

- Calidad del software: es el desarrollo de software basado en estándares con la funcionalidad y rendimiento total que satisfacen los requerimientos del cliente. (Vega Lebrún Carlos et al, 2008)
- “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, 1992)

- “El conjunto de características de una entidad que le confieren su aptitud para satisfacer las necesidades expresadas y las implícitas” (ISO8402:1994)

La **calidad de un programa** se puede medir en base a tres aspectos principales:

- Sus características operativas: corrección, eficiencia, facilidad de uso e integridad.
- Su capacidad para sufrir cambios: facilidad de mantenimiento, flexibilidad y facilidad de prueba.
- Su adaptabilidad a entornos distintos: reusabilidad, facilidad de interoperación y portabilidad. (Pes, 2006)

1.3 Calidad del producto

A la hora de definir la calidad del software se debe diferenciar entre la calidad del producto software y la calidad del proceso de desarrollo de éste (calidad de diseño y fabricación). No obstante, las metas que se establezcan para la calidad del producto van a determinar los objetivos a establecer de calidad del proceso de desarrollo, ya que la calidad del primero va a depender, entre otros aspectos, de ésta. Sin un buen proceso de desarrollo es casi imposible obtener un buen producto. (Pressman, 1998) (ISO8402:1994)

Pero la calidad del producto software se diferencia de la calidad de otros productos de fabricación industrial, ya que el software tiene sus propias características específicas:

- El software es un producto mental, no restringido por las leyes de la Física o por los límites de los procesos de fabricación. Es algo abstracto, intangible.
- Se desarrolla, no se fabrica. El coste está fundamentalmente en el proceso de diseño, no en la posterior producción en serie, y los errores se introducen también en el diseño, no en la producción.
- Los costes del desarrollo de software se concentran en las tareas de Ingeniería, mientras que en la fabricación clásica los costes se acentúan más en las tareas de producción.
- El software no se deteriora con el tiempo. No es susceptible de los efectos del entorno

y su curva de fallos es muy diferente de la del hardware. Todos los problemas que surjan durante el mantenimiento estaban allí desde el principio y afectan a todas las copias del mismo; no se generan nuevos errores.

- Es artesanal en gran medida. El software, en su mayoría, se construye a medida, en vez de ser construido ensamblando componentes existentes y ya probados, lo que dificulta aún más el control de su calidad.
- El mantenimiento del software es mucho más complejo que el mantenimiento del hardware. Cuando un componente del hardware se deteriora se sustituye por una pieza de repuesto, pero cada fallo en el software implica un error en el diseño o en el proceso mediante el cual se tradujo el diseño en código máquina ejecutable.
- Es engañosamente fácil realizar cambios sobre un producto software, pero los efectos de estos cambios se pueden propagar de forma explosiva e incontrolada.
- Como disciplina, el desarrollo de software es aún muy joven, por lo que las técnicas de las que dispone aún no están perfeccionadas.
- El software con errores no se rechaza. Se asume que es inevitable que el software presente algunos errores de poca importancia. (Palazzolo, 2005)

También es importante destacar que la calidad de un producto software debe ser considerada en todos sus estados de evolución (especificaciones, diseño, códigos,...). No basta con verificar la calidad del producto una vez finalizado cuando los problemas de mala calidad ya no tienen solución o su reparación es muy costosa.

1.4 Normas, modelos y atributos para la calidad del producto

La calidad del producto software siempre ha sido un elemento cuestionable desde muchos puntos de vista. Siempre resultará difícil comprobar la capacidad de un sistema de actuar de acuerdo a lo requerido sin tener en cuenta cada aspecto por separado que se desea validar. Si es importante verificar que cada funcionalidad representa objetivamente una actividad que el usuario realiza, más importantes son aun las comprobaciones que aseguran que el sistema se va a desempeñar en todos los ambientes para los cuales fue diseñado, de forma

óptima. Para dichas comprobaciones se han definido una serie de características, o atributos en dependencia de las bibliografías, que cubren el desempeño del software atendiendo a condiciones de ejecución.

Los atributos de **McCall** son de los más difundidos y además han servido de base para otros modelos (el modelo de Boehm y el Software Quality Management –SQM- de Murine).

En general los modelos de calidad definen a ésta de forma jerárquica, o sea la calidad se produce como consecuencia de la evaluación de un conjunto de indicadores o métricas en diferentes etapas:

En el nivel más alto de la jerarquía se encuentran los factores de calidad definidos a partir de la visión del usuario del software, y conocidos también como **atributos de calidad externos**.

Cada uno de los factores se descompone en un conjunto de criterios de calidad, o sea aquéllos atributos que cuando están presentes contribuyen a obtener un software de la calidad. Se trata de una visión de la calidad técnica, desde el punto de vista del producto software y se les denomina también **atributos de calidad internos**.

Finalmente para cada uno de los criterios de calidad se definen un conjunto de métricas o medidas cuantitativas de ciertas características del producto que indican el grado en que dicho producto posee un determinado atributo de calidad.

De esta manera, a través de un modelo de calidad se concretan los aspectos relacionados con ella de tal manera que se puede definir, medir y planificar. Además el empleo de un modelo de calidad permite comprender las relaciones que existen entre diferentes características de un producto software.

En contra de los modelos de calidad pesa que aún no ha quedado demostrada la validez absoluta de ninguno de ellos.

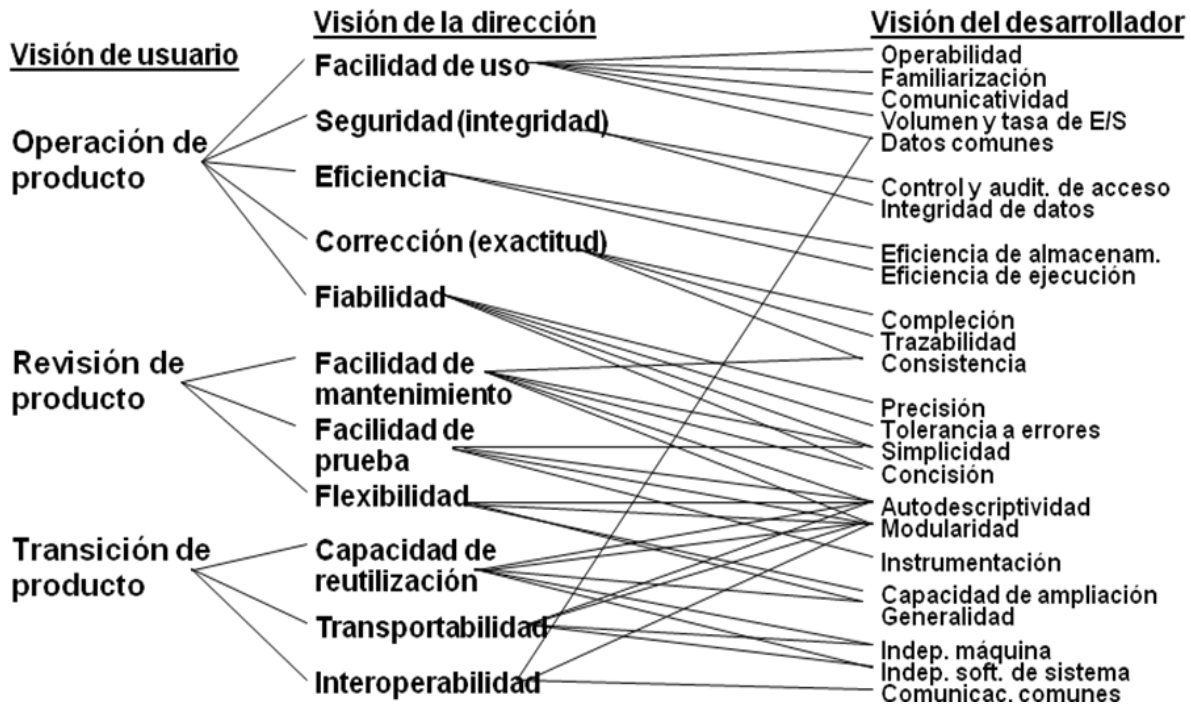


Ilustración 3. Modelo de Mc Call (1977) tomado de (Palazzolo, 2005)

Boehm

El modelo que presenta **Boehm** presenta una jerarquía de características donde cada una de ellas contribuye a la calidad global. Se centra en:

- Sus características operativas.
- Su capacidad para soportar los cambios.
- Su adaptabilidad a nuevos entornos.
- La evaluación del desempeño del hardware.

El modelo comienza con la utilidad general del software, afirmando que el software es útil, evitando pérdida de tiempo y dinero. La utilidad puede considerarse en correspondencia a los tipos de usuarios que quedan involucrados. El primer tipo de usuarios queda satisfecho si el sistema hace lo que él pretende que haga; el segundo tipo es aquel que utiliza el sistema luego de una actualización y el tercero, es el programador que mantiene el sistema.

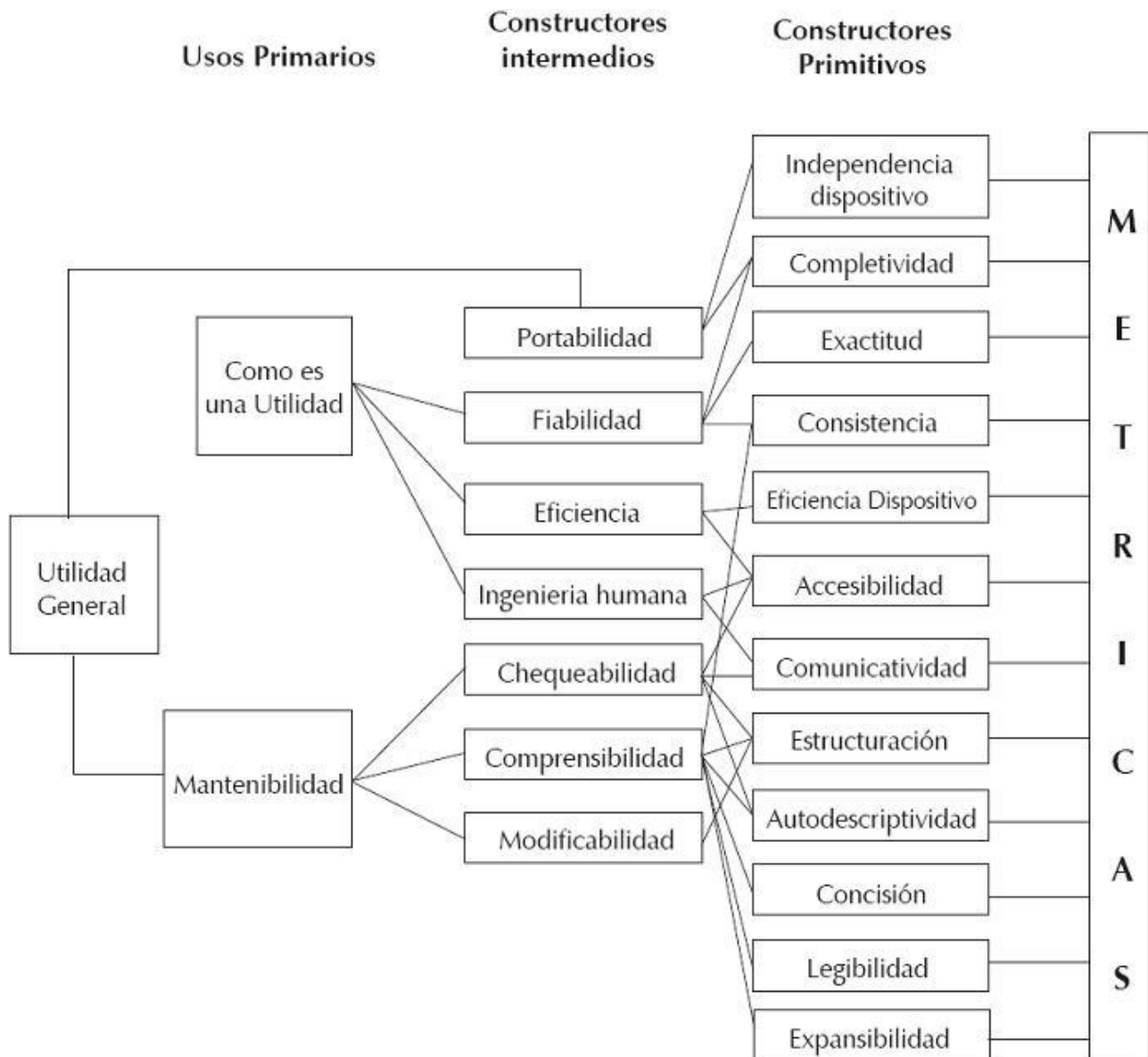


Ilustración 4. Modelo de Boehm(1978)tomado de (Palazzolo, 2005)

ISO/IEC25000:2005

Dentro las normas que forman parte del compendio ISO 25000 del 2005 se recoge la 25012 que tiene su antecedente en la 9126 del 1997 de donde se hace referencia a las características internas y externas que se vieron en el modelo de McCall como atributos de calidad (factores y criterios).Para este caso el término empleado es el de características y

sub-características del software.

Características	Sub-Características	
9126-1/25012	9126-1	25012
Funcionalidad	Interoperabilidad Seguridad	Consistencia Compleitud Precisión Aseguramiento Interoperabilidad Seguridad
Confiabilidad	Madurez Tolerancia a fallos Recuperabilidad	Disponibilidad Recuperabilidad
Usabilidad	Operabilidad Atractividad	Manejabilidad Atractividad
Eficiencia	Comportamiento del tiempo Empleo de los recursos	Empleo de los Recursos
Mantenibilidad	Facilidad de análisis Facilidad de Cambio Estabilidad Facilidad de prueba	Facilidad de análisis Facilidad de cambio
Portabilidad	Adaptabilidad Instalabilidad Co-existencia Reemplazabilidad	Adaptabilidad

Tabla 1. Características de la ISO 9126(1997) y su correspondencia con la 25000(2005) tomado de la

norma.

Características	Sub-Características	
9126-1/25012	9126-1	25012
Efectividad	-	Perdurabilidad Adecuación
Productividad	-	Relevancia Interpretación
Seguridad	-	Relevancia de seguridad
Satisfacción	-	Credibilidad Accesibilidad

Tabla 2. Características de la ISO 25000 tomado de la norma.

Independientemente de las etiquetas empleadas por los autores, el manejo de los términos vistos se hace de manera general. Y se reconoce el aporte de cada modelo en los elementos definidos para la medición del software. Teniendo en cuenta la definición más exacta y referente a los términos de verificación se empleara para el desarrollo de la propuesta la ISO/IEC25000:2005 que además viene avalada por una técnica de aplicación al análisis de riesgo con base en su antecesora la 9126.

1.5 Riesgos

1.5.1 Definiciones

La palabra "riesgo", ha tenido hasta la fecha muy diversas interpretaciones, en términos muy simples existe riesgo en cualquier situación en que no sabemos con exactitud lo que ocurrirá al futuro.

Para "La International Organization for Standardization" (ISO27000:2002) define al riesgo como:

"Combinación de la Probabilidad de un Evento y su Consecuencia" ISO aclara que el término riesgo es generalmente usado siempre y cuando exista la posibilidad de pérdidas (resultado negativo)".

El "The Institute of Internal Auditors" (IIA, 2004) define al riesgo como:

"La Posibilidad de que ocurra un acontecimiento que tenga un impacto en el alcance de los objetivos. El riesgo se mide en términos de impacto y probabilidad".

Al analizar las diversas definiciones de riesgo podemos decir, que en nuestros tiempos es muy importante que las entidades ya sean públicas o privadas, se interioricen de cómo manejar de la mejor manera este concepto, para así poder llevar a cabo su misión y por ende lograr sus objetivos planteados. Teniendo en cuenta los términos más útiles a la investigación, la definición de la IIA constituirá una base para fundamentar el procedimiento a definir para las pruebas basadas en riesgo.

Las pruebas de software se refieren a dos tipos de riesgos:

Riesgos de la calidad del producto (o del producto), que son problemas que pueden afectar a la calidad del producto en sí, como un defecto que podría provocar que un mal funcionamiento del sistema.

Riesgos del Proyecto (o de la planificación del proyecto), que son problemas que pueden afectar el éxito general del proyecto, tales como la escasez de personal que podrían retrasar la realización de un entregable.

Partiendo de un planteamiento muy bien visto y por demás comprobado en varias empresas de calidad de software que realizan pruebas a nivel mundial a los productos más recientes. Los riesgos a la calidad son mitigados mediante las pruebas y los riesgos del proyecto mediante los controles. (Rex Black et al. 2008)

Para este trabajo serán más útiles el primer grupo y se verán cuáles son las técnicas

empleadas para su análisis así como la implicación directa de estos riesgos en las pruebas de software.

1.6 Gestión de riesgos

La Gestión de riesgos es un enfoque estructurado para manejar la incertidumbre relativa a una amenaza, a través de una secuencia de actividades humanas que incluyen evaluación de riesgo, estrategias de desarrollo para manejarlo y mitigación del riesgo utilizando recursos gerenciales. Las estrategias incluyen transferir el riesgo a otra parte, evadir el riesgo, reducir los efectos negativos del riesgo y aceptar algunas o todas las consecuencias de un riesgo particular.

Algunas veces, el manejo de riesgos se centra en la contención de riesgo por causas físicas o legales (por ejemplo, desastres naturales o incendios, accidentes, muerte o demandas).

El objetivo de la gestión de riesgos es reducir diferentes riesgos relativos a un ámbito preseleccionado a un nivel aceptado por la sociedad. Puede referirse a numerosos tipos de amenazas causadas por el medio ambiente, la tecnología, los seres humanos, las organizaciones y la política. Por otro lado, involucra todos los recursos disponibles por los seres humanos o, en particular, por una entidad de manejo de riesgos (persona, staff, organización).

Todas las decisiones involucran minimizar, reducir o eliminar el riesgo, ya sea decisiones que se toman en las operaciones diarias o las decisiones acerca de políticas importantes, estrategias o nuevos proyectos. Dentro de este contexto, muchas veces debemos de tomar decisiones de forma muy rápida y a menudo basadas en la intuición, pero es muy importante pensar acerca de los riesgos que esto pudiese involucrar.

No se debe olvidar, que la Gestión de Riesgos debe ocurrir durante todo el desarrollo e implementación de un proyecto.

Con respecto a los miembros del equipo de desarrollo, el líder debe trabajar en mentalizarlos para que ellos entiendan que este tipo de gestión que va de la mano con el seguimiento y la planificación es una herramienta que debemos implementar para la consecución de los objetivos.

1.6.1 Gestión de riesgos de un software

La Gestión de Riesgos es un proceso complejo que conduce al planeamiento y aplicación de políticas, estrategias, instrumentos y medidas orientadas a impedir, reducir, prever y controlar los efectos adversos de fenómenos peligrosos sobre el proyecto; por lo que se llevan acciones integradas de reducción de riesgos a través de actividades de prevención, mitigación, preparación y atención de emergencias. Este proceso ha adquirido gran importancia en los últimos años, dado el impacto que tiene en el cumplimiento de los objetivos del proyecto y la calidad del mismo.

1.6.2 Gestión de riesgos como área de proceso de CMMI

Gestión de riesgo (RSKM)

La Gestión de riesgo es un área del proceso de la gerencia de proyecto que se ubica en el nivel 3 de la madurez.

El propósito de la **Gestión de riesgo** (RSKM) como área es identificar problemas potenciales antes de que ocurran para que pueda ser planificado el tratamiento y las actividades pertinentes a través del ciclo de desarrollo del producto o del proyecto y de esta forma atenuar impactos adversos en la realización de objetivos.

Objetivos específicos y prácticas.

- SG 1 Se prepara para la Gestión de riesgo
 - 5-El SP 1.1 determina fuentes y categorías del riesgo

6-EI SP 1.2 define parámetros del riesgo

7-EI SP 1.3 establece una estrategia para la gestión de riesgo

•SG 2 identifican y analizan riesgos

5-EI SP 2.1 identifica riesgos

6-EI SP 2.2 evalúa, categoriza, y da la prioridad a riesgos

•SG 3 atenúan riesgos

5-EI SP 3.1 desarrolla planes de la mitigación del riesgo

1.6.3 Herramientas para la gestión de riesgos.

A nivel mundial se emplean varios sistemas para la gestión de riesgos como son:

- Magerit 2.0.
- GIRO.
- ORM de Monitor PLUS.

De las herramientas mencionadas ninguna presenta funcionalidades específicas que puedan aplicarse a la gestión de riesgos del producto de software, ni arrojan los resultados necesarios para el proceso de pruebas definido en el laboratorio industrial de pruebas del DPSW de Calisoft. Por lo planteado anteriormente se decide proponer la implementación de una herramienta para gestionar los riesgos del producto que se emplearan en las pruebas y agilizar el proceso de análisis y definición del alcance de las pruebas.

1.7 Pruebas de software

Las pruebas, vistas desde el marco de un proceso de desarrollo de software, son los diferentes procesos que se deben realizar durante un desarrollo, con el objetivo de asegurar completitud, correctitud, calidad, entre otros factores de gran importancia.

- Una prueba es aquella actividad vital y muy necesaria que se llevará a cabo para

demostrar la verdad de un hecho. (DefinicionAbc,2010)

- Un hecho utilizado para demostrar una acción, tesis o teoría en ciencias.
- En informática, procesos que permiten verificar y revelar la calidad de un producto software. (Wikipedia,2010)
- Es el proceso mediante el cual se establece la existencia de errores.

Las pruebas no se deben dejar para el final de la etapa de construcción del software. Deben empezar desde la misma etapa de análisis y licitación de requerimientos, ya que desde un principio se puede incurrir en malas interpretaciones de las "reglas del negocio", lo que finalmente tendrá como consecuencia incongruencia entre lo que el cliente quiere y lo que se ha desarrollado. Las pruebas de software, en inglés testing o beta testing son los procesos que permiten verificar y revelar la calidad de un producto de software. Son utilizadas para identificar posibles fallos de implementación, calidad, o usabilidad de un programa de ordenador o videojuego. (Wikipedia,2010) Para la presente investigación el término empleado para la definición de pruebas de software que más se ajusta al tema central es el siguiente:

- Proceso realizado concurrentemente a través de las diferentes etapas de desarrollo de software que utiliza y cuyo objetivo es apoyar la disminución del riesgo de aparición de fallas y faltas en operación. (Pruebas de software, 2007)

1.7.1 Proceso de Pruebas

Las pruebas se realizan a lo largo del desarrollo del sistema y no simplemente al final. Esto significa sacar a la luz problemas no conocidos y no demostrar la perfección de programas manuales o equipo.

La prueba se realiza en subsistemas o módulos de programa conforme el trabajo avanza. Se hace en muchos niveles diferentes y a diversos intervalos. Antes de que el sistema sea

puesto en producción, todos los programas deben ser probados en el escritorio, revisados con datos de prueba y revisados para ver si los módulos trabajan juntos entre ellos, tal como se escribió.

También debe ser probado el sistema trabajando con un todo. Esto incluye probar las interfaces entre subsistemas, la corrección de la salida y la utilidad y comprensibilidad de la documentación de la salida del sistema. Los programadores, analistas, operadores y usuarios juegan papeles diferentes en los diversos aspectos de la prueba.

El flujo de trabajo de prueba dentro de RUP

En el flujo de trabajo de prueba se verifica el resultado de la implementación probando cada construcción, incluyendo tanto construcciones internas como intermedias, así como las versiones finales del sistema a ser entregadas a terceros.

Los objetivos de las pruebas son [Jacobson et al,1999]:

- Planificar las pruebas necesarias en cada iteración, incluyendo las de integración y las de sistema. Las pruebas de integración son necesarias para cada construcción dentro de la iteración, mientras que las de sistema son necesarias sólo al final de la iteración.
- Diseñarlas e implementarlas creando los casos de pruebas que especifican qué probar, creando los procedimientos de pruebas que especifican cómo realizarlas y creando, si es posible, componentes de pruebas ejecutables para automatizarlas.
- Realizar las diferentes pruebas y manejar los resultados de cada una sistemáticamente. Las construcciones en las que se encuentran defectos son probadas de nuevo y posiblemente devueltas a otro flujo de trabajo, como diseño o implementación, de forma que los defectos importantes puedan ser arreglados.

El flujo de trabajo de prueba dentro de RUP está compuesto por el papel que éstas

desempeñan en la vida del software, los artefactos que utilizan, como son los modelos de pruebas; los casos de pruebas; los procedimientos; los componentes; los planes de prueba; los defectos y la evaluación de las mismas; además, se detallan los trabajadores involucrados en este proceso como son: el diseñador de pruebas, el ingeniero de componentes, el ingeniero de pruebas de integración y el ingeniero de pruebas del sistema; y también se abordan los flujos de trabajo que este proceso contiene como: planificar las pruebas; diseñarlas; implementarlas; realizar las pruebas de integración y las de sistema y, por último, evaluar las pruebas; lo cual se detallará a continuación.

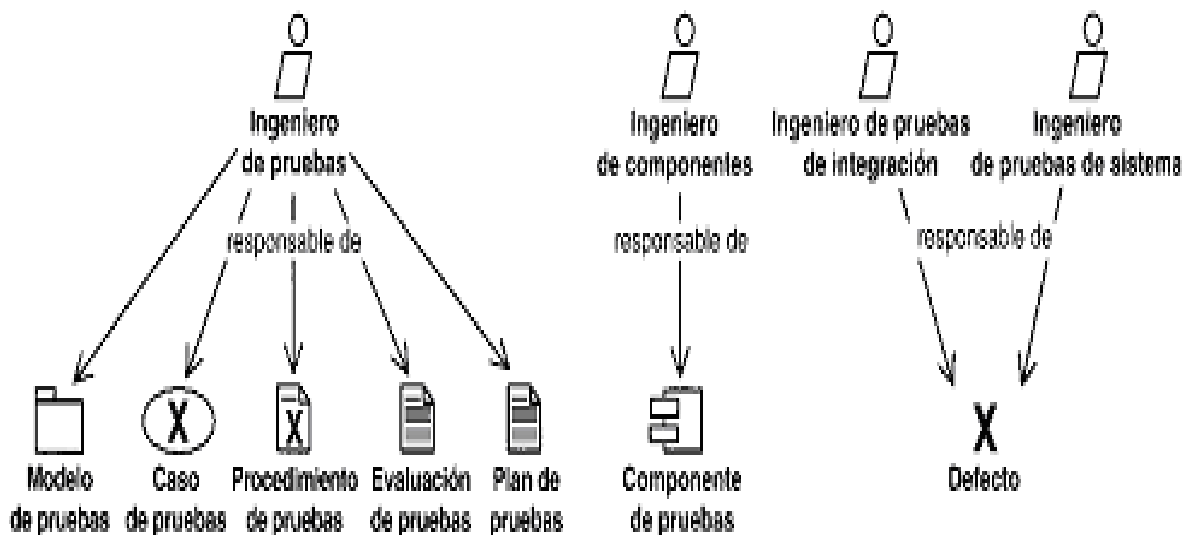


Ilustración 5. Los trabajadores y artefactos involucrados en las pruebas tomado de (Jacobson et al. 1999)

1.7.2 Tipos de Pruebas.

Las pruebas se realizan con el objetivo de comprobar el cumplimiento de los requisitos funcionales y no funcionales del sistema. En dependencia del requisito que se pruebe coincidirá con el tipo de prueba a realizar, de aquí que los tipos de pruebas se clasifican en:

- Función.

- Seguridad.
- Volúmen.
- Usabilidad.
- Integridad
- Estructura.
- Stress.
- Benchmark.

1.7.3 Técnicas de prueba

1.7.3.1 Pruebas Basadas en experiencia

Son aquellas pruebas que requieren de mayores competencias en el probador y están mayormente enfocadas a las experiencias obtenidas de pruebas anteriores.

- Adivinación de Errores
- Ataques al sistema
- Exploratorias

Pruebas exploratorias

Las pruebas exploratorias pueden ser realizadas en cualquier situación donde no sea obvio cual es la próxima prueba que se debe realizar. Puede tener varios objetivos como obtener retroalimentación rápida de cierto producto o funcionalidad, aprender el producto rápidamente investigar y aislar un defecto en particular, investigar el estado de un riesgo particular, o evaluar la necesidad de diseñar pruebas para determinada área. (Capote, 2011)

Pruebas de Caja Blanca

Las pruebas de caja blanca comprueban los caminos lógicos del software proponiendo casos de prueba que se ejerciten conjuntos específicos de condiciones y/o bucles. Se puede

examinar el estado del programa en varios puntos para determinar si el estado real coinciden con el esperado o mencionado.

Requieren del conocimiento de la estructura interna del programa y son derivadas a partir de las especificaciones internas de diseño o el código. Mediante los métodos de prueba de la caja blanca, el ingeniero de software puede obtener casos de prueba que garanticen que:

- Se ejerciten por lo menos una vez todos los caminos independientes para cada módulo.
- Se ejerciten todas las decisiones lógicas en sus vertientes verdaderas y falsa.
- Ejecuten todos los bucles en sus límites y con sus límites operacionales.
- Se ejerciten las estructuras internas de datos para asegurar su validez.

La prueba de la caja blanca, a primera vista, podría parecer impracticable puesto que no es posible aplicarla exhaustivamente para grandes sistemas, sin embargo no se debe desechar ya que se puede elegir y ejercitar una serie de caminos lógicos importantes, que invoquen además las estructuras de datos más importantes para comprobar su validez. Se pueden combinar ambos métodos para llegar a un método que valide la interfaz del software y asegure selectivamente que el funcionamiento interno del software es correcto. (Hernández et al, 2009)

Técnicas de caja blanca

- La prueba del camino básico.
- La prueba de condición.
- La prueba de flujo de datos.
- La prueba de bucles.

Pruebas de caja Negra

También conocidas como Pruebas de Comportamiento, estas pruebas se basan en la especificación del programa o componente a ser probado para elaborar los casos de prueba. El componente se ve como una “Caja Negra” cuyo comportamiento sólo puede ser determinado estudiando sus entradas y las salidas obtenidas a partir de ellas. Para seleccionar el conjunto de entradas y salidas sobre las que trabajar, hay que tener en cuenta

que en todo programa existe un conjunto de entradas que causan un comportamiento erróneo en nuestro sistema, y como consecuencia producen una serie de salidas que revelan la presencia de defectos. Para confeccionar los casos de prueba de Caja Negra existen distintos criterios. Algunos de ellos son:

- Particiones de Equivalencia.
- Análisis de Valores Límite.
- Métodos Basados en Grafos.
- Pruebas de Comparación.
- Análisis Causa-Efecto.

Estas pruebas verifican las especificaciones funcionales y no consideran la estructura interna del programa.

- Se realiza sin el conocimiento interno del producto.
- No validan funciones ocultas (por ejemplo funciones implementadas pero no descritas en las especificaciones funcionales del diseño) por tanto los errores asociados a ellas no serán encontrados.

En otras palabras, la prueba de la caja negra se refiere a las pruebas que se llevan a cabo sobre la interfaz del software. O sea los casos de prueba pretenden:

- Demostrar que las funciones del software son operativas.
- Que las entradas se aceptan de la forma adecuada y que se produce el resultado correcto.
- Así como la integrada de la información externa (por ejemplo archivos de datos) se mantiene.

Guiada por Casos de Prueba.

Las pruebas funcionales guiadas por casos de prueba representan un caso especial de las pruebas de caja negra donde se emplea el flujo central descrito por cada caso de uso y se prueba teniendo en cuenta cada variable independientemente, verificando la incidencia de las mismas en las respuestas del sistema.

1.8 Herramientas para la ejecución de pruebas

Las herramientas constituyen un elemento fundamental en las pruebas de software cuales quieran que sean, es muy importante siempre identificar cuales podrían servir de ayuda para llevar este proceso y contribuir a la mejora de las mismas y a la adecuación con relación a los resultados que queremos que estas nos reporten.

1.8.1 Herramientas Manuales

1.8.1.1 Listas de chequeo

La lectura basada en Listas de chequeo o comprobación (checklists, en inglés) proporciona un apoyo mayor mediante preguntas que los probadores deben responder mientras leen el artefacto. Es decir, esta técnica proporciona listas que ayudan al probador a saber qué tipo de faltas buscar.

1.8.1.2 Casos de prueba

Un caso de prueba es: un conjunto de entradas de pruebas, condiciones de ejecución y resultados esperados desarrollados para cumplir un objetivo en particular o una función esperada.

Los casos de pruebas deben verificar:

- Si el producto satisface los requerimientos del usuario, tal y como se describe en las especificación de los requerimientos.
- Si el producto se comporta como se desea, tal y como se describe en las especificaciones funcionales del diseño.

1.8.2 Herramientas Automatizadas

Para la realización de las pruebas directamente sobre la aplicación informática se emplean múltiples herramientas las cuales se adecuan al tipo de prueba que se va a realizar y a la técnica que aplicará. Entre las usadas en el laboratorio están:

- Selenium: Función

- Paros Proxy: Seguridad
- Nexus: Seguridad
- Test Maker: Volúmen
- Jmeter: Stress, Contención, Carga.

Estas herramientas no vienen asegurar la calidad de las pruebas como elemento clave del proceso, pero si constituyen un objeto estratégico y de apoyo fundamental para la dinámica que requiere dicho proceso. Todas requieren de conocimiento del probador para su empleo y todas aportan datos acertados y muy cercanos a las expectativas del ingeniero de pruebas. Si bien las listas de chequeo requieren de un trabajo arduo y sobre todo del producto de un vasto conocimiento en las características del elemento a revisar, las herramientas automatizadas y los casos de prueba no son más que simples mecanismos de verificación superficial, sin la adecuada preparación del especialista al frente de la prueba.

1.9 Pruebas basadas en riesgo

Como se vio anteriormente las pruebas tienen un papel fundamental en la mitigación de los riesgos que puede tener el producto en su calidad a la hora de su desempeño, puesto que detectan aquellas vulnerabilidades que pueden afectar el buen funcionamiento, o la política de acceso, por poner un ejemplo, que deben estar presentes en el software y legitimar sus condiciones de entrega al cliente. Teniendo en cuenta ambas áreas, la combinación de los riesgos y las pruebas resulta a veces interesante (por los resultados que pueden arrojar las sesiones de análisis y las lecciones que puede aprender el equipo de desarrollo a partir de dichos resultados) y en otros momentos necesaria (por la premura con que a veces se desarrollan las solicitudes de pruebas, contando con fechas cercanas previstas para las entregas de los sistemas y por el número creciente de solicitudes anuales que son recibidas en el DPSW y que deben ser respondidas con puntualidad). Analizando esta unión en estudios se define las pruebas basadas en riesgo de la siguiente forma.

Las pruebas basadas en riesgos consisten en:

- Identificar riesgos específicos a la calidad del sistema.
- Evaluar y asignar el nivel de riesgo a cada riesgo, basándose en la probabilidad de

ocurrencia (a consideración técnica) y el impacto (consideraciones del negocio).

- Asignar el esfuerzo de prueba y priorizar (la secuencia) la ejecución de prueba basada en riesgo.
- Revisar el análisis de riesgo a intervalos regulares en el proyecto incluyendo el después de las pruebas de la primera construcción. (Shaefer, 2004) (Black, 2008)

Sin importar la técnica escogida, el camino a seguir para el proceso de análisis debe ser el siguiente:

- Identificar el equipo de analistas de riesgos a la calidad.
- Seleccionar una técnica.
- Identificar los riesgos, priorizarlos, seleccionar acciones de mitigación idóneas. (Brainstorm)
- Si el equipo detecta problemas en los requerimientos o en otro artefacto, deben reportarse inmediatamente para su resolución.
- Revisar, repasar y finalizar el documento de análisis de riesgos a la calidad.
- Comprobar el documento de análisis de riesgo a la calidad en el expediente de proyecto y bajo control de versiones.

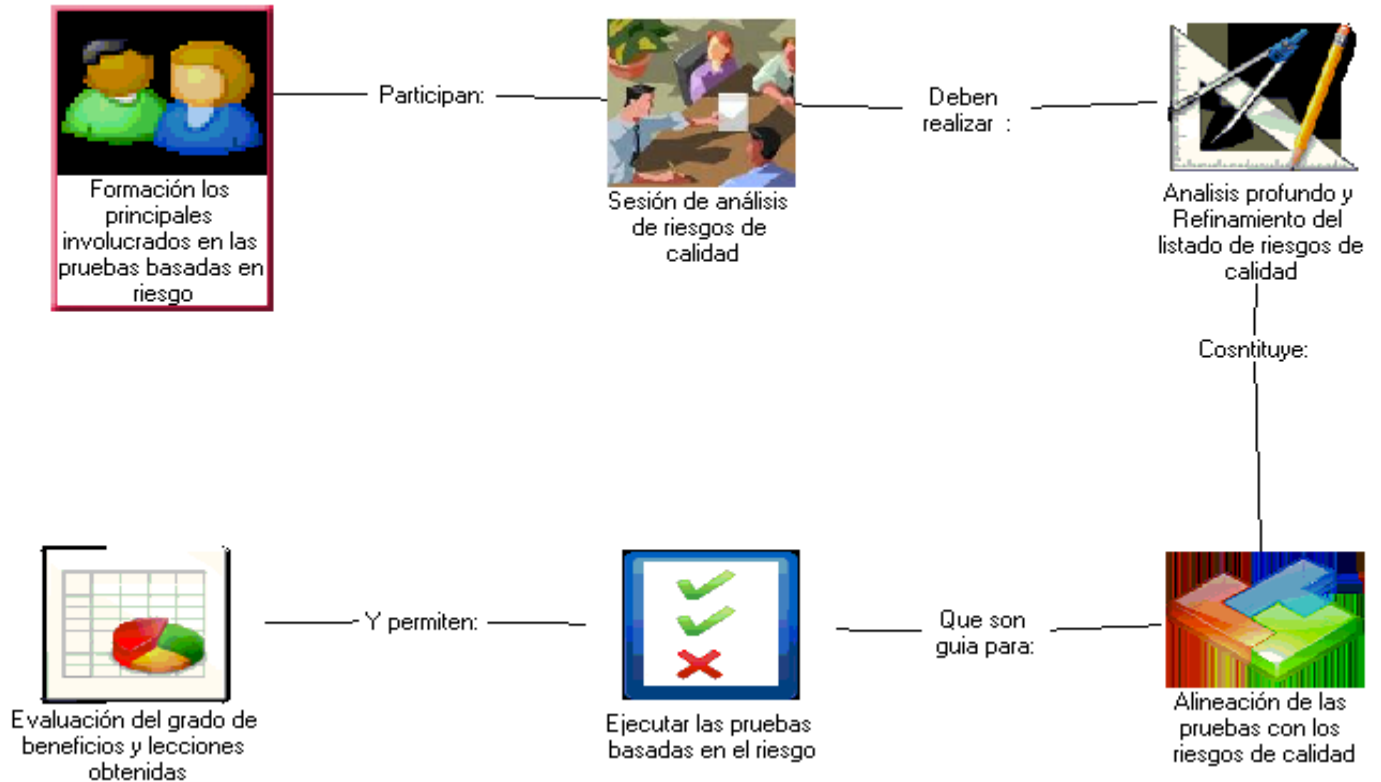


Ilustración 6. Actividades de las Pruebas Basadas en Riesgos (Creación propia)

En intervalos regulares de tiempo y unido al incremento de información del proyecto, se debe repasar y actualizar el análisis de riesgos adicionando nuevos elementos identificados y re-evaluando los niveles de riesgo para los detectados anteriormente.

1.10 Técnicas para analizar el riesgo

En el mundo existen muchas técnicas aplicables a la gestión de riesgos en la producción, a partir del análisis profundo de todas las técnicas identificadas resultaron ser las siguientes 5 las que más se adecuaron al proceso de análisis de los riesgos a la calidad del producto y a las actividades que se identificaron por separado dentro de él.

Técnica de análisis de riesgo de calidad informal (*Informal quality risk analysis*)(Black,2002)

Esta técnica plantea que se debe realizar un encuentro formal donde se listen todos los

riesgos aplicando una técnica informal para la identificación de los mismos. Ej: Lluvia de ideas.

Esta técnica proporciona un modo fácil para comenzar el análisis de riesgos de la calidad. Puesto que casi siempre se omite algunas áreas importantes de riesgo, sobre todo durante análisis tempranos de riesgo, pero esta técnica aunque informal hace posible alcanzar un mejor grado de enfoque en la prueba y una mayor cobertura que la que se alcanza normalmente sin dicho análisis.

Técnica de análisis de riesgo a la calidad según ISO 9126 (*ISO 9126 quality risk analysis*)

Para la presente técnica de análisis de riesgo se deben usar las características y subcaracterísticas descritas en la ISO 9126 como sistema estándar de categorías para el análisis de riesgos. Las principales son:

- Funcionalidad
- Fiabilidad
- Usabilidad
- Eficiencia
- Mantenibilidad
- Portabilidad

En cada categoría existe una o más subcaracterísticas, ISO 9126 provee un acercamiento más estructurado y reduce la probabilidad de omitir algunos elementos de riesgo importantes. Esto también tiende a añadir un poco en términos de tiempo y trabajo administrativo. (Black, 2008)

Nota: Dado que la norma 9126 data del año 2001, se consideró más prudente adaptar la técnica a la actualización de la misma que es la ISO25000 del 2005.

Técnica de costo de exposición para el análisis de riesgo (*Cost of exposure quality risk analysis*)(Black, 2003)

La técnica de costo de exposición se enfoca en la siguiente pregunta: ¿Cuáles son las pérdidas esperadas, asociadas a varios riesgos? Y ¿Cuánto se debe gastar en reducir esos riesgos?

Una pérdida esperada es el producto de la probabilidad de la pérdida multiplicada por el costo de la misma:

$$\text{Pérdida esperada} = \text{Probabilidad de la pérdida} * \text{Costo de la pérdida}$$

Dicha técnica permite al equipo de gestión del proyecto tomar una decisión económica dura sobre las pruebas. Para que la misma funcione bien, sin embargo, el equipo de análisis debe ser capaz de estimar con exactitud las probabilidades y los gastos de varias pérdidas.

Técnica de modo de falla y análisis de efecto (*Failure mode and effect analysis*)(Stamatis ,1995)

El método de análisis de modo de fallos y efecto va más allá de la discusión sobre riesgos. Empleando esta técnica el equipo de análisis de riesgo trata de identificar los caminos diferentes que el sistema podría fallar, y todos los posibles efectos que estas fallas puedan traer, en los clientes, negocio, sociedad, etc. Es una técnica bien detallada, puede producir una prueba finamente cargada, pero también una tonelada de papeleo y trabajo administrativo y mucho tiempo invertido en dicho análisis. (Black, 2008)

Técnica de análisis de amenazas (*Hazard analysis*)(Craig & Jaskiel,2002)

Esta técnica comparte puntos de contacto con la anterior pero se realiza a la inversa. Comenzando por los efectos se va de estos hacia las causas de los mismos. Durante el trabajo se van haciendo más claras las probabilidades de esas causas. En algunos casos, aunque existen muchas causas de diferentes tipos de malos comportamientos. Dicha técnica tiende a trabajar mejor en sistemas pequeños pues se considera que realizando el análisis a la inversa que se requiere, podrían surgir muchos caminos diferentes hacia las causas, pudiéndose entrecruzar dichos caminos y malogrando de esta forma el trabajo empleado en la determinación de las causas reales que serían mucho mayor en número y clasificación mientras más complejo sea el proyecto. (Black, 2008)

Técnica	Informal(Black, 2002)	ISO9126(ISO)	Costo de exposición(Black,2003)	Modo de fallas/análisis de	Análisis de amenazas(Craig
---------	-----------------------	--------------	----------------------------------	----------------------------	----------------------------

				efectos(Stamatis,1995)	Jaskiel,2002)
En resumen	Basada en historial, experiencias y listas de chequeo.	Siguiendo las características de calidad según el estándar industrial.	Estimación del costo de prueba contra el costo de no probar.	Identifica los errores potenciales y su efecto en los involucrados	Analiza causas de amenaza(fuentes de riesgo)
Fortalezas	Fácil, de peso ligero, flexible	Predefinida ,exhaustiva ,común	Balanceada , singular ,tradicional	Precisa, meticulosa, general	Exacta ,prudente , sistemática
Debilidades	Dependiente del personal participante, abierta, imprecisa	Potencialmente sobre estimada, sobre regimentada.	Intensivo de información , exclusivamente monetaria.	Prolongada ,pesada documentación ,difícil de aprender	Fácilmente puede abrumarse por su complejidad
Adecuada para proyectos_	De bajo riesgo o ágiles.	Comprometido con los estándares	Financiero	De alto riesgo o conservativos	Médico o embebido.
Evitar en proyectos_	Críticos o regulados en cuanto a seguridad	Muy inusuales o de estructura intolerante	De seguridad crítica- misión-	Caóticos, rápidamente cambiantes o ligados a prototipos.	Impredecibles o Complejos.

Tabla 3.Comparación de las técnicas de análisis de riesgo tomado de (Black,2008)

Todas las técnicas enunciadas aportan datos de valor al proceso de pruebas basadas en riesgo por lo que se decide adecuarla a las actividades de realización identificadas en el mismo y de esta forma logran una adaptación específica de la técnica a la actividad a desarrollar puesto que en varios casos una misma técnica puede ser empleada en varias actividades por la cantidad de datos de valía que arroja. Así como también habrá actividades que emplearan varias técnicas de manera organizada para obtener los datos deseados a emplear en la prueba.

1.10 Conclusiones

Luego del análisis de los conceptos relacionados con las pruebas de software, niveles, tipos, los modelos de calidad de software, la gestión de riesgos, y las técnicas de análisis de riesgo se definieron las actividades fundamentales para desarrollar las pruebas basadas en riesgos así como el camino a seguir para desarrollar el análisis de los riesgos que puede presentar un producto:

- A partir de la diferenciación de los tipos de riesgos fue posible enmarcar el alcance de la investigación a los que confieren únicamente al producto y la calidad del mismo. Es conveniente aclarar en el presente trabajo que todas las técnicas descritas no se aplican únicamente a este grupo únicamente la desarrollada a partir de la ISO9126, el resto puede aplicarse a otros procesos de gestión y análisis que comprendan riesgos de otra índole.
- La selección de esta técnica fue posible a partir del análisis de los diferentes modelos de calidad de software que al final quedan resumidos en ella. Y que es para este trabajo la más importante pues permitirá establecer un indicador numérico en porcentaje de la calidad del producto en términos de características de calidad.
- Las pruebas basadas en riesgos son un aspecto novedoso a nivel mundial en la industria de software y a pesar de su comprobada efectividad y el número de adeptos

e impulsores que laboran en empresas tan importantes como el ISQTB (International Software Testing Quality Board)y RBCS (Rex Black Consulting Services ,quienes son líderes en el tema) aún no han sido adoptadas por otras empresas que brindan servicios de pruebas y que bien pudieran tomar esta alternativa, debido a lo novedoso del tema , y a que desconocen en su mayoría lo eficiente que resulta aplicar este método de análisis que puede brindar resultados muy fiables en un corto tiempo y que más que nada puede ser personalizado según el tipo de producto que se prueba.

Es por eso que el estudio de las actividades desarrolladas en el LIPS permitirá alcanzar la personalización que requiere un procedimiento estándar para aplicar pruebas de este tipo y que debe permitir además implementarse en cualquier tipo de los productos desarrollados en la UCI.

2 Capítulo 2. Análisis del Proceso de Pruebas de Liberación y Procedimiento propuesto para las Pruebas Basadas en Riesgos a la Calidad del Producto.

Introducción

En el presente capítulo se abordará la propuesta de solución al problema, iniciándose en el análisis del proceso definido actualmente, enunciando el proceso propuesto para pruebas basadas en el análisis de riesgos, y por último planteando detalladamente sus actividades insertadas en el proceso de liberación actual. Además se propone una alternativa de aplicación del procedimiento en la etapa de pruebas exploratorias como método de selección de la muestra.

2.1 Antecedentes

A partir del proceso de pruebas propuesto originalmente por (Jorrín, 2005) y analizando los cambios que, producto de la experiencia, se le han sobrevenido, en el presente capítulo se expondrán como actualmente se aplican las pruebas de software a los productos a término, previo a su entrega al cliente. Teniendo en cuenta que el Proceso de Pruebas de Liberación (PPL) tiene su base en una necesidad de hace algunos años, a fuerza de satisfacer las necesidades que se han presentado, este ha ido sufriendo cambios que lo han adecuado al trabajo actual que se lleva en el Laboratorio Industrial de Pruebas de Software, debido a que el trabajo de mesa en las revisiones es realizado por estudiantes, y atendiendo al gran número de solicitudes de Liberaciones que son recibidas semanalmente y los problemas que han presentado históricamente los productos liberados y que han entorpecido el trabajo en las pruebas atrasando los cronogramas de entregas de los productos y empleando tiempo que bien podría utilizarse revisando productos con una calidad inicial más tangible.

2.2 Descripción general del funcionamiento del LIPS

Actualmente la definición en el LIPS del Proceso de Pruebas de Liberación (PPL) se ve muy ligado no solo a las necesidades ya históricas de la Universidad de las Ciencias Informáticas

(UCI) sino que se unen a estas las necesidades crecientes del país en tener un órgano evaluador a nivel nacional del software cubano.

Las actividades definidas en el PPL tienen como objetivos la organización, ejecución controlada y el flujo ininterrumpido de las pruebas, de manera que, con una buena comunicación con el equipo de desarrollo y un cumplimiento cabal del cronograma, estas avancen y se detecten el mayor número de No Conformidades posibles y su resolución asegure la calidad del sistema en cuestión.

La última versión del mismo aparece en el artículo Conceptualización e implantación de un Laboratorio Industrial de Pruebas de Software (Capote, 2011). Siendo este diagrama la guía actual empleada para realizar las PL a los productos de la UCI. En su trabajo sobre la conceptualización del LIPS el diagrama viene como un resultado de la combinación de los elementos esenciales definidos en su trabajo y que define de forma general como funciona el LIPS y cuáles son las actividades y subprocesos que se ejecutan de manera continua y armónica.

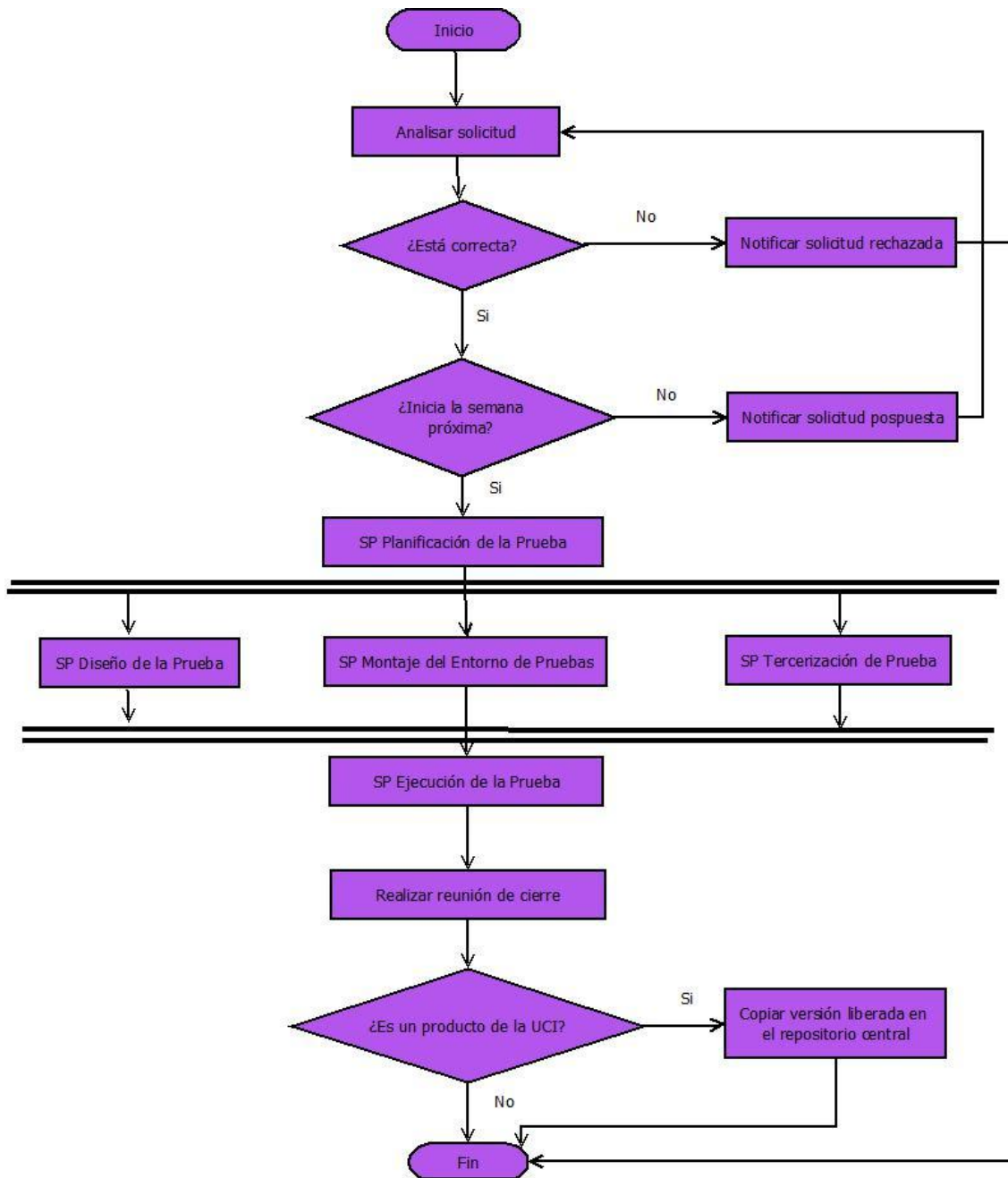


Ilustración 7. Proceso de Pruebas de Liberación tomado de (Capote, 2011)

El proceso presentado en la figura anterior está descrito con más detalles de forma textual y gráfica en el Libro de Procesos del LIPS (GNM, 2010), el cual fue elaborado utilizando la plantilla definida por el Grupo de Normalización y Métricas de Calisoft.

2.3 Procedimiento para las Pruebas Basadas en Riesgos a la Calidad del Producto.

Luego del análisis del PPL, teniendo en cuenta la premura con que deben ser atendidas la mayoría de las solicitudes recibidas en el LIPS se evidencia la necesidad de contar con métodos o elementos que permitan identificar en un corto tiempo los puntos más débiles del software que se produce y/o más propensos a provocar fallas en el mismo. Con ese propósito se define un Proceso de Pruebas Basadas en Riesgos que permita a partir del análisis y registro de los errores más comúnmente detectados en el LIPS, vistos en forma de riesgos, que presentan los sistemas informáticos de manera general, realizar un análisis que permita priorizar y probar acorde a las prioridades, las funcionalidades que más problemas pueden causar en el despliegue del sistema y puesta en práctica.

El proceso diseñado para tal función es presentado a continuación, siendo diseñado a partir del Método para Aplicar Pruebas Basadas en riesgos descrito en el trabajo Risk based testing (Shaefer, 2004) y enunciado para productos desarrollados en la UCI en Pruebas Basadas en Riesgos (Menéndez et al, 2010)

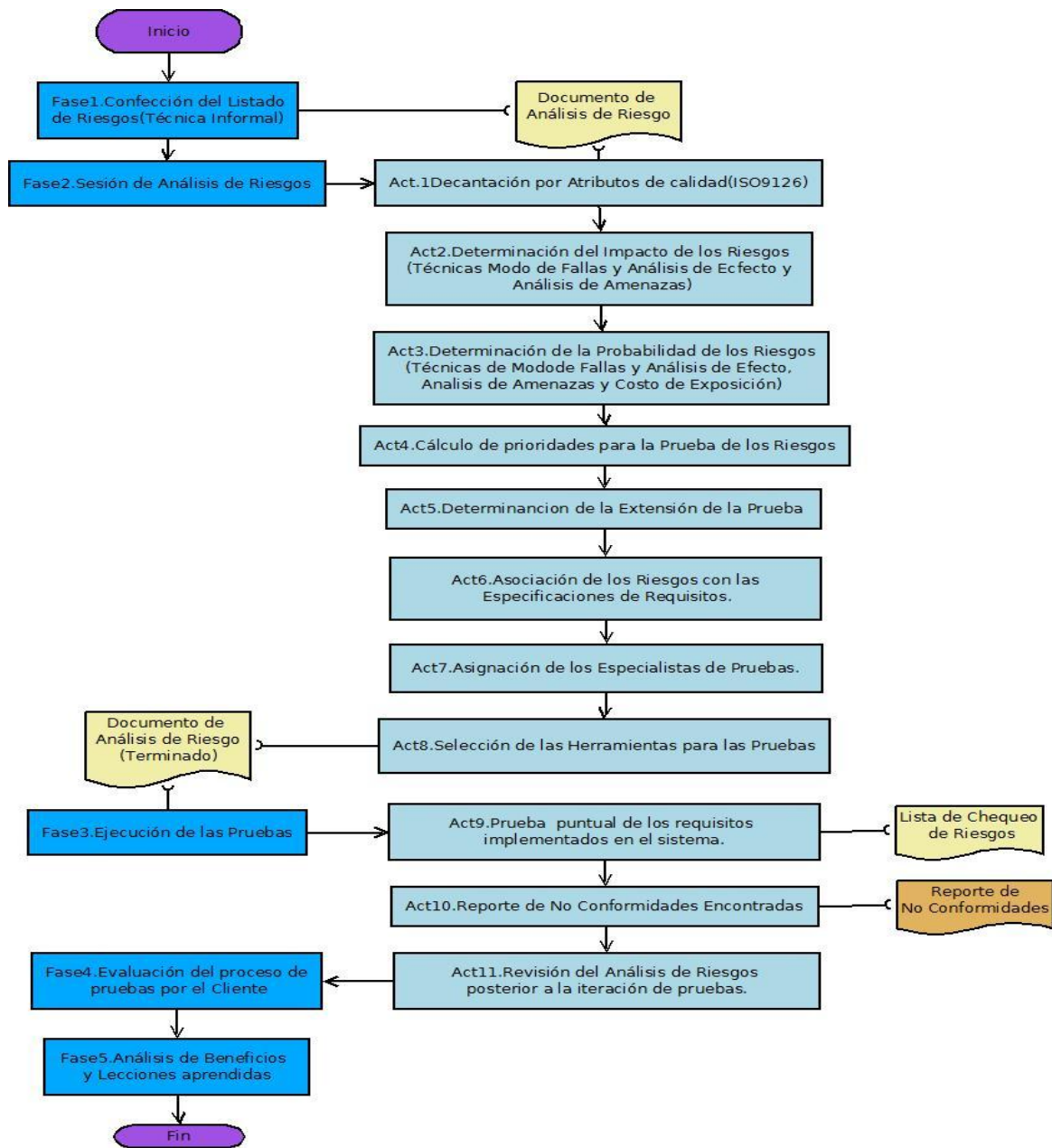


Ilustración 8. Proceso de Pruebas Basadas en Riesgos (Creación propia).

Luego de un estudio del proceso propuesto para aplicar las pruebas basadas en riesgos en los productos de la UCI y las actividades y subprocesos descritos en el PPL se insertan las actividades del proceso inicial en el PPL para complementar los objetivos que cada uno tiene trazado por separado y se resumen las actividades comunes dando como resultado un Proceso más ajustado a las necesidades del LIPS.

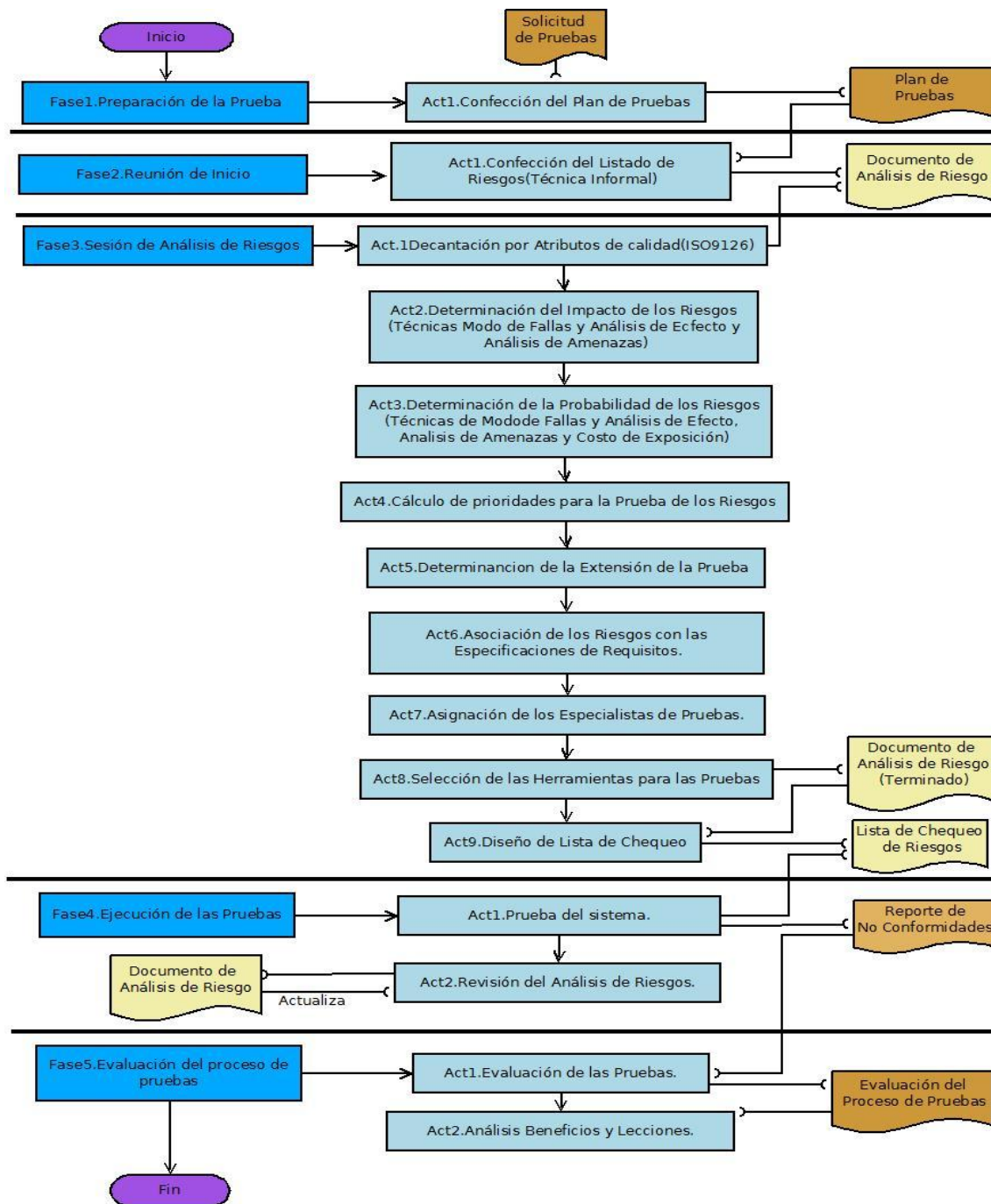


Ilustración 9.Procedimiento de Pruebas Basadas en Riesgos para el LIPS (Creación propia).

2.3.1 Fases y Actividades

Las Fases definidas para el procedimiento tienen su base fundamental en las fases establecidas en el PPL , en algunos casos se hacen acotaciones a la fase y las actividades

que aseguran su completamiento, en otros casos se definen nuevas fases así como sus respectivas actividades, teniendo en cuenta el nuevo punto de vista de los riesgos y su análisis.

2.3.2 Roles y Responsabilidades

Los Roles que se proponen para el procedimiento propuesto son los definidos por RUP y el PPL y que obedecen a las mismas definición y responsabilidades que plantea la Metodología y el Libro de procesos (GNM, 2010) únicamente se requiere la participación de algunos roles claves dentro de las actividades de listado y análisis de los Riesgos y por consiguiente en las actividades resultado de estas que serían la base del procedimiento. Los roles de forma general son: Líder de Proyecto, Analista, Especialista de Pruebas, Jefe del DPSW.

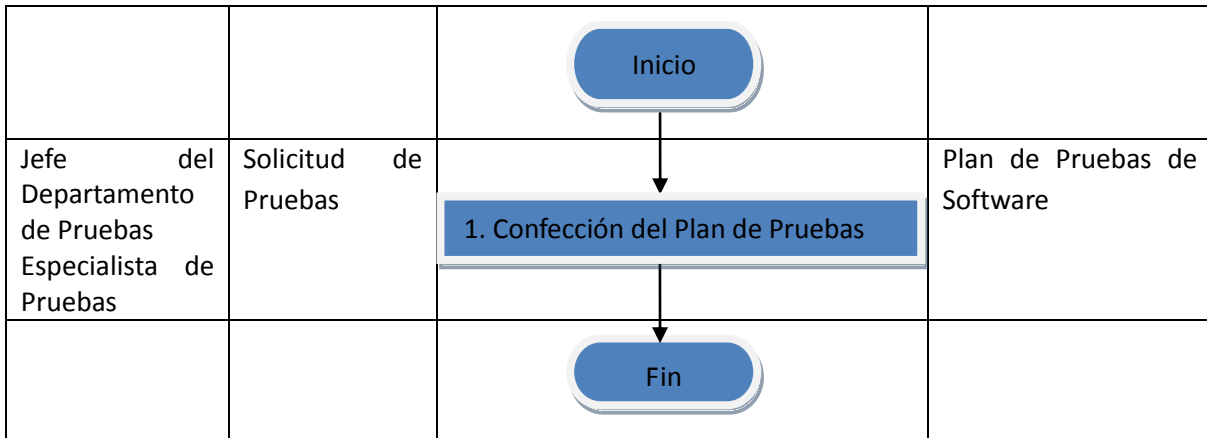
2.3.3 Artefactos

Como propuesta se definen dos nuevos artefactos uno para registrar la información resultante de cada actividad de la Fase de Análisis de Riesgos que es el Documento de Análisis de Riesgos. Y el segundo que es un lista de chequeo donde se registrarán los riesgos identificados en las sesiones de análisis, separados por categoría, para el trabajo del estudiante en el LIPS, en la misma se registran las especificaciones de la prueba y las preguntas específicas que deberán comprobar los estudiantes en el software a liberar.

2.3.4 Procedimiento

Fase 1.Preparación de la Prueba.


Descripción			
En la presente Fase se discute la solicitud de la prueba, se decide que es factible hacer pruebas basadas en riesgos se aprueba y se pasa a la citación de una reunión de inicio y se convienen sobre cuáles son los artefactos necesarios para registrar la información de las pruebas y de apoyo a la misma.			
Roles	Entradas	Actividades	Salidas



Fase 1. Preparación de la Prueba		
Actividad	Descripción	Artefactos
1.	Esta actividad describe la conformación de un documento donde se registrarán los elementos necesarios para el aseguramiento de la prueba tales como son, el objetivo, alcance, cronograma, entorno, y definición de responsabilidades para las pruebas.	Plan de Pruebas de Software

Fase 2. Reunión de Inicio.

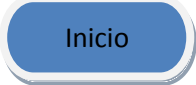
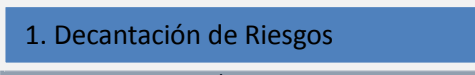
Descripción			
<p>La reunión de inicio es una actividad que se lleva a cabo de manera obligatoria y de imprescindible asistencia por parte de los involucrados en las pruebas, con motivo de la discusión del plan de pruebas y el cronograma que se define en el mismo, para de esta forma garantizar un cumplimiento puntual al 100% de las tareas ahí registradas. Esta actividad entra dentro del SP Planificación de la Prueba.</p>			
Roles	Entradas	Actividades	Salidas
Jefe del DPSW Especialista de Pruebas Líder de Proyecto			
Especialista de Pruebas Analista	Plan de Pruebas de Software	<div style="background-color: #4F81BD; color: white; padding: 5px; border: 1px solid #4F81BD;">1. Confección del Listado de Riesgos</div>	Documento de Análisis de Riesgos

Fase 2. Reunión de Inicio

Actividad	Descripción	Artefactos
1.	Esta actividad puede realizarse con éxito de diversas formas, mediante una reunión formal, implementando encuentros sistemáticos planificados entre los desarrolladores y el equipo de pruebas o empleando la técnica de tormenta de ideas garantizando siempre la participación de todos los involucrados. Teniendo en cuenta un número elevado de situaciones disimiles que pongan el sistema en problemas se van listando cada una de estas situaciones, enfocadas al problema que causaría en el funcionamiento del sistema .Y de esta forma se va conformando el Documento de Análisis de Riesgos que posteriormente se irá modificando a medida que se apliquen las técnicas para dicho análisis.	Documento de Análisis de Riesgos

Fase 3. Análisis de Riesgos a la Calidad del Producto

Descripción			
En esta fase se aplican las diferentes técnicas seleccionadas del estudio realizado con anterioridad en el Capítulo 1 para realizar cada actividad propuesta de manera exitosa.			
Roles	Entradas	Actividades	Salidas
			
Especialista de Pruebas Analista	Plan de Pruebas de Software		Documento de Análisis de Riesgos(actualizado)

Especialista de Pruebas Analista	Documento de Análisis de Riesgos	2. Determinación del Impacto	Documento de Análisis de Riesgos(actualizado)
Especialista de Pruebas Analista	Documento de Análisis de Riesgos	3. Determinación de la Probabilidad	Documento de Análisis de Riesgos(actualizado)
Especialista de Pruebas	Documento de Análisis de Riesgos	4. Cálculo de Prioridades	Documento de Análisis de Riesgos(actualizado)
Especialista de Pruebas	Documento de Análisis de Riesgos	5. Determinación de Extensión	Documento de Análisis de Riesgos(actualizado)
Especialista de Pruebas Analista	Documento de Análisis de Riesgos	6. Asociación a las E.R.	Documento de Análisis de Riesgos(actualizado)
Jefe del DPSW	Documento de Análisis de Riesgos	7. Asignación del Especialista	Documento de Análisis de Riesgos(actualizado)
Especialista de Pruebas	Documento de Análisis de Riesgos	8. Selección de las Herramientas	Documento de Análisis de Riesgos(actualizado)
Especialista de Pruebas	Documento de Análisis de Riesgos	9. Diseño de la Lista de Chequeo	Lista de Chequeo de Riesgos
		Fin	

Fase 3. Análisis de Riesgos a la Calidad del Producto

Actividad	Descripción	Artefactos
1.	A partir de los riesgos registrados en el Documento de Análisis se debe hacer una distribución de cada elemento registrado empleando la Técnica de Análisis de Riesgos a la Calidad según la ISO9126(o empleando la ISO/IEC25000:2005 alternativamente), reorganizando cada riesgo en grupos atendiendo al atributo de calidad al que atentan directamente.	Documento de Análisis de Riesgos
2.	Aplicando las técnicas de Modo de Fallas y Análisis de Efecto, y Análisis de Amenazas se determinan para cada riesgo ya asociado a un atributo de calidad específico, el	Documento de Análisis de Riesgos

	<p>impacto que su ocurrencia traería en la calidad del desempeño del sistema. El valor asignado estará ubicado en una escala de 1 a 5 para mayor seguridad en los cálculos de futuras actividades del procedimiento. Utilizando estas técnicas también se pueden obtener las probabilidades de ocurrencia de cada riesgo .El registro de estos datos se efectuaría en la próxima actividad enunciada. De este modo se actualiza además la información del Documento de Análisis de Riesgos. Ver anexo1.</p>	
3.	<p>Aplicando las técnicas de Modo de Fallas y Análisis de Efecto, y Análisis de Amenazas y Costo de Exposición se calcula la probabilidad de ocurrencia aproximada para cada riesgo. El valor de la probabilidad también debe estar ubicado en una escala de 1 a 5 para asegurar la actividad y lograr uniformidad en los cálculos para el posterior análisis de sus resultados. De este modo se actualiza además la información del Documento de Análisis de Riesgos. Ver anexo2.</p>	Documento de Análisis de Riesgos
4.	<p>En esta actividad se determina para cada riesgo el número de prioridad con que debe probarse .El cálculo se hará uniformemente teniendo en cuenta que dicha prioridad es igual al producto del impacto por la probabilidad de ocurrencia del riesgo .El número resultante debe estar entre 1 y 25.Ver Anexo3.</p>	Documento de Análisis de Riesgos
5.	<p>Teniendo en cuenta el número de prioridad calculado se deben hacer en esta actividad diversos grupos atendiendo a varios rangos de estos valores. Cada rango estará asociado a una clasificación de la extensión de la Prueba a realizar.</p>	Documento de Análisis de Riesgos
6.	<p>Empleando el Documento de Análisis de Riesgos con toda la información registrada hasta el momento, y el Documento de Especificación de Requisitos, se harán coincidir en esta actividad riesgo descrito con la especificación correspondiente a la funcionalidad o característica del sistema registrada en el segundo documento y de esta forma se designan cuáles serán las funcionalidades que se probaran y en qué orden.</p>	Documento de Análisis de Riesgos
7.	<p>Teniendo en cuenta los temas de investigación desarrollados en el LIPS por los diferentes especialistas y el trabajo de vanguardia que cada uno realiza identificando nuevas pruebas y herramientas atendiendo a un atributo de calidad específico, se asignan para cada Especialista el listado de requisitos que debe probar del sistema y en qué orden debe hacerlo, teniendo en cuenta además la</p>	Documento de Análisis de Riesgos

	extensión de la prueba definida por el número de prioridad asociado al riesgo.	
8.	Para efectuar la prueba el especialista debe seleccionar de las herramientas disponibles cuáles serán las más adecuadas y cuyos resultados serán más exactos para desarrollar la prueba lo más exitosamente posible.	Documento de Análisis de Riesgos
9.	Luego de finalizado el Documento de Análisis de Riesgos el especialista elabora una lista de chequeo para el trabajo del estudiante en el LIPS.	Lista de Chequeo de Riesgos

Fase 4.Ejecución de la Prueba

Descripción			
Esta actividad constituye un subproceso descrito en el Libro de Procesos de Calisoft .Para el proceso propuesto se hace una acotación en la actividad por el uso de un nuevo artefacto no registrado en el PPL.			
Roles	Entradas	Actividades	Salidas
		<pre> graph TD Inicio([Inicio]) --> A[1. Prueba del Sistema] A --> B[2. Revisión del Análisis de Riesgos] B --> Fin([Fin]) </pre>	
Probador	Lista de Chequeo de Riesgos	1. Prueba del Sistema	Registro de No conformidades
Especialista de Pruebas Analista	Documento de Análisis de Riesgos	2. Revisión del Análisis de Riesgos	Documento de Análisis de Riesgos(actualizado)

Fase 4. Ejecución de la Prueba

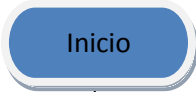
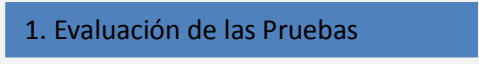

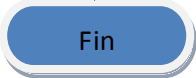
Actividad	Descripción	Artefactos
1.	Para esta actividad es necesario tener un previo conocimiento del trabajo con las herramientas del LIPS, de otra forma sería oportuno que los probadores reciban una preparación en este tema y que de ser posible trabajen con la asesoría del especialista al frente de la prueba. Para tal caso también es	Lista de Chequeo de Riesgos Registro de No conformidades

recomendable que los artefactos diseñados para la prueba, tengan los valores bien descritos y lo más entendible posible para evitar explicaciones innecesarias y ganar en tiempo.

2. Una vez culminada la primera iteración o ciclo de pruebas, si es necesario o conveniente se deben revisar y/o refinar, de manera opcional, cada dato registrado referente a un riesgo del Documento de Análisis de Riesgos, díganse el impacto, la probabilidad y por consiguiente el NPR.

Documento de Análisis de Riesgos

Fase 5.Evaluación del proceso de pruebas.

Descripción			
En esta fase se analizan los inconvenientes que hubo en el proceso y se registran las lecciones aprendidas por los involucrados.			
Roles	Entradas	Actividades	Salidas
			
Líder de Proyecto Analista	Registro de No Conformidades		Evaluación del Proceso de Pruebas
Líder de Proyecto Especialista de Pruebas	Evaluación del Proceso de Pruebas		Evaluación del Proceso de Pruebas(actualizado)
			

Fase 5. Evaluación del Proceso de Pruebas

Actividad	Descripción	Artefactos
1.	Para la evaluación de las pruebas ya se ha establecido una planilla que encuesta al cliente sobre los temas de satisfacción respecto al proceso de pruebas, vendría faltando como evaluar lo referente a la experiencia del equipo de desarrollo respecto a las sesiones de análisis de riesgos y los beneficios que esto le reportó y que servirán para futuros procesos de desarrollo de software.	Evaluación del Proceso de Pruebas
2.	Esta fase se desarrolla empleando nuevamente la Técnica Informal descrita anteriormente y se propone hacer una reunión entre los especialistas al frente de la prueba , el equipo de desarrollo y de ser posible el cliente , pues la diversidad de criterios puede influir directamente en la identificación de los beneficios resultantes del proceso de pruebas basadas en el análisis de los riesgos ,y los beneficios del análisis de riesgos mismo que de alguna forma darán visibilidad a las lecciones aprendidas del proceso y esto permitirá dejar una constancia de las acciones tomadas para la resolución de posibles problemas , a priori.	Evaluación del Proceso de Pruebas

2.4 Otra propuesta de aplicación del procedimiento.

Teniendo en cuenta la experiencia adquirida en el tiempo que lleva aplicándose el Proceso de Pruebas de Liberación, y el cúmulo de pruebas que han tenido que ser suspendidas por falta de documentación de apoyo necesaria a las pruebas, exceso de tiempo empleado por el proyecto para resolver las no conformidades detectadas, falta de cumplimiento con las fechas de las actividades definidas en el cronograma del Plan de Pruebas, se estableció una etapa de Pruebas Exploratorias como actividad del proceso, donde se aplican las pruebas de software solo a una muestra considerable del producto y los resultados son comparados contra indicadores que; de ajustarse a alguno, provocan la detención o aborto de la prueba. Estos indicadores o criterios de parada, a la hora de hacer una prueba se denominan Criterios de Criticidad y como su nombre les indica van a dar al traste con los problemas que

presente el sistema durante el proceso e indicaran cuán crítico se encuentra para su pase a las pruebas funcionales.

Las Pruebas Exploratorias deben incluir un método de selección de la muestra, y en dependencia de la efectividad del método será más o menos significativa la muestra que se tome y por consiguiente los resultados se acercarán más o menos a la realidad de los problemas que tiene el sistema. De ahí la importancia de aplicar un método de selección de la muestra que dé resultados bien exactos o que al menos se acerquen lo más posible a la realidad del sistema.

Dado que el análisis de riesgos va a detectar los problemas que podría presentar el sistema durante su puesta en marcha y posteriormente; este método podría identificar a fondo cuales son aquellas funcionalidades más propensas a fallar. Así como la probabilidad de ocurrencia del fallo y el impacto que podría tener sobre el producto. De esta forma arrojaría un resultado más significativo en cuanto a las no conformidades que podrían resultar de las pruebas funcionales y serviría como un argumento fuerte a la hora de aplicar los criterios de criticidad y tomar las decisiones respectivas a la continuidad del proceso de pruebas o la detención del mismo.

Actualmente se encuentran en investigación varios métodos de selección y ya el propuesto en el presente trabajo ha sido seleccionado como uno de los más exactos para las Pruebas Exploratorias ,así como rápido para su aplicación, de manera que se pueda optimizar en tiempo y recursos .

2.5 Conclusiones

De lo presentado en el capítulo se concluye:

- Fueron analizados los subprocesos y actividades del Proceso de Pruebas de Liberación.
- Se desarrolló la propuesta de procedimiento para desarrollar Pruebas Basadas en Riesgos en el LIPS haciendo coincidir las actividades comunes entre este y el PPL vigente.
- Se propusieron dos artefactos complementarios al PPL que serán la base de la

propuesta pues servirán de contenedor para los datos de los sistemas que serán probados.

- El documento de Análisis de Riesgo propuesto es el resultado de la combinación de varias técnicas y permitirá registrar los resultados de cada análisis específico por técnica, mientras que la Lista de Chequeo de Riesgos será la herramienta de trabajo más importante para las pruebas desarrolladas en el LIPS pues está diseñada al más básico nivel y puede ser comprendida por todos los estudiantes que ejecutan las pruebas. De esta forma servirá de guía única para los probadores independientemente de su nivel de experticia en el tema.
- Fueron descritas y detalladas las actividades del procedimiento propuesto.
- Además se propuso la variante de aplicar la propuesta como método de selección de la muestra en las Pruebas Exploratorias a realizar con anterioridad a las pruebas funcionales como parte del PPL.

3 Validación de la Propuesta y Presentación de los Resultados

3.1 Introducción

En este capítulo se realizará el estudio de los resultados alcanzados con la aplicación del procedimiento propuesto y los obtenidos a partir de la instrumentación del Método Delphi para la validación por expertos del Procedimiento para Pruebas Basadas en Riesgos.

El primer estudio se basa en los resultados obtenidos a partir de la liberación de tres proyectos que aplicaron el procedimiento, analizando la relación de NC resultantes de las pruebas funcionales y que están asociadas directamente con los riesgos detectados en la sesión de análisis. Para tal estudio se tendrán en cuenta las variables de tiempo de la prueba, calidad del proceso (expresada en el número de pruebas efectuadas y los diferentes tipos) y cumplimiento con la norma ISO/IEC 25000:2005.

3.1.1 Método Delphi para la Validación de la propuesta.

El método Delphi es un procedimiento eficaz (Linstone H.A, Turrof M, 1975) y sistemático que tiene como objeto la recopilación de opiniones de expertos sobre un tema particular con el fin de incorporar dichos juicios en la configuración de un cuestionario y conseguir un consenso a través de la convergencia de las opiniones de expertos diseminados geográficamente (Bass, 1983) (B.G.Ludwig, 1996). Desde los inicios de su utilización, atribuida (N.C.Dalkey, O.Helmer, 1963) en la década de los 50, ha sufrido algunas modificaciones. Inicialmente se partía de un cuestionario abierto sobre un tema concreto que era presentado a expertos con el tema a investigar para que cada uno presentara, de forma anónima, sus aportaciones, aunque actualmente muchos estudios utilizan la versión modificada (Young, 2009). Su principal novedad consiste en el uso desde la primera ronda de un cuestionario estructurado al que se van adicionando o modificando, si es el caso, las diferentes opiniones de los expertos en las sucesivas rondas hasta completar al menos (Linstone H.A, Turrof M, 1975). Por otra parte, este método tiene sus ventajas e inconvenientes. En cuanto a estos últimos (Bernard, 1988) destaca la selección de los elementos iniciales del cuestionario y la

selección de los expertos, que (Adranovich, 1995) recomendó se hiciera coincidir con el interés de los expertos y el tema de estudio, a fin de que su participación fuera significativa. La no existencia de directrices que marquen el consenso entre los expertos es una de las desventajas. Los resultados, son las percepciones de los expertos. El tiempo requerido para completar el cuestionario puede llevar a dar respuestas/opiniones apresuradas. En cuanto a las ventajas podemos citar que permite obtener y participar en el procedimiento de validación a expertos distribuidos geográficamente, cosa que de hacerse de forma presencial costaría mucho más tiempo y dinero. En cuanto a la subjetividad de las respuestas, éstas se verán avaladas por el análisis de contenidos que se realizará de cada una de las opiniones a fin de evitar esta circunstancia.

La utilización del Método Delphi como instrumento de validación de cuestionarios ha sido ampliamente utilizada en numerosos estudios y ámbitos de conocimiento como el de las pruebas de software. (2006,2007) (LA APLICACIÓN DE LA TÉCNICA DELPHI, PARA LA CONSTRUCCIÓN DE UN INSTRUMENTO DE ANÁLISIS CATEGORIAL DE INVESTIGACIONES E-LEARNING, 2008).

3.1.2 Estudio de casos para la Validación de la propuesta

El otro metodo empleado para la validación consiste en el estudio de dos proyectos liberados en el LIPS donde fueron aplicadas las pruebas basadas en riesgo.Los resultados de las pruebas quiadas a partir del analisis de riesgos ,aplicadas a los mismos, serán comparados con el total de NC resultantes de la liberación para comprobar de esta forma el impacto de introducir esta variable en las pruebas y como se podría ahorrar tiempo previendo las funcionalidades mas vulnerables.

3.2 Validación de la Propuesta con Delphi

3.2.1 Objetivo de la validación

Validar mediante el método Delphi un cuestionario para conocer las experiencias e interés hacia las pruebas de software basadas en riesgo con especial atención a los especialistas

del DPSW y demás especialistas vinculados a las pruebas de software ya sea mediante las clases u otra actividad docente- productiva.

3.2.2 Muestra

Para conseguir el objetivo de la investigación, en concordancia con lo que plantea el Método Delphi en sus postulados teóricos, se conformaron los dos grupos humanos encargados de validar el instrumento diseñado, en este caso el grupo coordinador y el grupo de expertos.

El primer grupo se conformó con tres especialistas en el tema: un doctor y dos másteres que dominan el tema de la investigación y realizarán la coordinación del grupo de expertos.

La selección del grupo de expertos la realizó el grupo coordinador, para lo cual el primer paso fue fijar como criterio fundamental de selección la competencia de los candidatos en el área del conocimiento en que se inserta la investigación sobre la base de su currículum personal.

Se identificaron 5 expertos en el tema de los cuales accedieron a participar 3.

3.2.3 Instrumento

El instrumento que se somete a validación es un documento de 5 aspectos donde en uno de ellos se propone la valoración de los elementos de un cuestionario sobre el procedimiento propuesto y que además contiene aspectos para evaluar las experiencias en el tema y el interés en su aplicación por parte de los expertos y demás especialistas vinculados a las actividades del LIPS. El cuestionario tiene 21 elementos y de ellos 2 son relacionados con los Riesgos, 2 sobre las Pruebas de Software, 2 sobre Pruebas Basadas en Riesgo, 2 sobre el PPL, 8 sobre comprensión y aceptación del Procedimiento propuesto, 5 sobre actividades e impresiones personales .

3.2.4 Procedimiento

Para la aplicación del método se proponen tres fases: Preliminar, Exploratoria y Final.

3.2.4.1 Fase Preliminar

En esta fase deben crearse tres condiciones fundamentales para la validación.

- Conformación del grupo coordinador.
- Selección del cuestionario.
- Selección de los expertos.

Actividades

1. Selección del cuestionario por los miembros del grupo coordinador.
2. Análisis y Discusión del cuestionario en el grupo coordinador.

En esta primera fase se conformó el grupo coordinador el cual estuvo conformado por 3 especialistas conocedores del tema, quizá no tan a fondo por la novedad del mismo pero que forman un criterio válido reconocido a nivel institucional en nuestro país en los temas de calidad de software y pruebas, de ellos 1 es doctor y el resto son másteres (Dra. Ailyn Febles, Msc. Tayche Capote y Msc. Dennis Neuland), este grupo asumió la responsabilidad de delimitar el tema de estudio concibiendo inicialmente el problema de investigación; seleccionar el grupo de expertos y conseguir su compromiso de colaboración; interpretar los resultados parciales y finales de la investigación; y supervisar la marcha correcta de la investigación pudiendo realizar ajustes y correcciones.

Resultados

Como resultados de esta fase se obtuvo una primera versión del cuestionario a aplicar para la validación de la propuesta. Además se seleccionó el grupo de expertos para la aprobación del cuestionario.

El grupo seleccionado estuvo compuesto por cuatro personas con el grado científico de máster y un alto grado de conocimientos en el tema de investigación (Msc. Ramsés Delgado, Msc. Violena Hernández, Msc. Maypher Román). A este grupo se dirigió la primera versión del cuestionario .Ver Anexo 4.6.

3.2.4.2 Fase Exploratoria

Actividades

1. Respuesta de los expertos al cuestionario.
2. Análisis estadístico cuantitativo de las respuestas de los expertos y agregación de los

comentarios por el grupo coordinador.

En esta fase se analizó el cuestionario por el grupo de expertos y fueron analizadas las respuestas de forma cuantitativa y cualitativa para el caso de los criterios agregados al instrumento por parte de los expertos. Los comentarios que más aportes valiosos hicieron a la investigación pues fueron añadidos al cuestionario definitivo. Entre el grupo de expertos hubo un promedio del grado de conocimientos del tema de 7 puntos (entre 0 y 10) lo cual no demuestra un dominio pleno por la novedad del tema, pero si un conocimiento bastante amplio de los conceptos referidos, así como un gran entendimiento del procedimiento propuesto y apoyo total para su implantación.

Resultados

Luego de la valoración de los expertos para cada elemento del cuestionario las calificaciones respecto al grado en que cada elemento es adecuado para la validación de la propuesta, según el criterio personal y la experticia de cada miembro de este selecto grupo, fueron analizadas teniendo en cuenta el número de calificaciones de un mismo tipo que recibían los elementos. Para este caso se suman además las consideraciones del grupo coordinador.

Valoración de Expertos

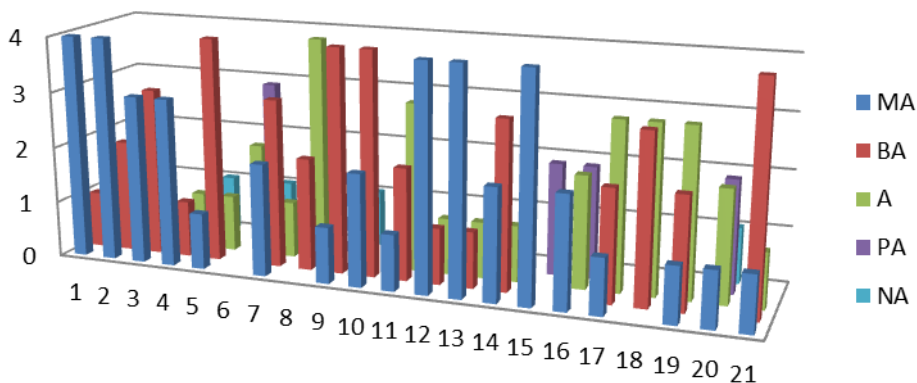


Ilustración 10 Valoración de Expertos

Como se puede ver en la gráfica ningún elemento resultó ser No Adecuado para el 50% de los expertos y solo para el caso del elemento 6 (Cómo califica sus conocimientos en el tema.) hubo mayoría de calificaciones como Poco Adecuado debido a que para el caso de los

expertos esta información no es de importancia para la validación del procedimiento pero si se considera fundamental para considerar las necesidades de capacitación de los especialistas que trabajaran el tema. No hubo elemento calificado como poco adecuado. Más del 90% de los elementos fueron calificados de Muy Adecuado, Bastante Adecuado y Adecuado. En resumen no hubo sugerencias de suprimir ningún elemento del cuestionario, dejando las preguntas de reflexión como opcionales, a tener en cuenta en análisis de apoyo al resultado de las respuestas para el resto de los elementos. De esta forma quedó dispuesto el Cuestionario Definitivo de la siguiente forma. Ver anexo 4.7.

3.2.4.3 Fase Final

Actividades

1. Aplicación del Cuestionario Definitivo.

En esta fase se aplicó el cuestionario definitivo en un número de 20 especialistas y profesores que estuvieron dispuestos a colaborar con la Validación del procedimiento para Pruebas Basadas en Riesgos.

Resultados

Los resultados de la aplicación del cuestionario reflejan el criterio de 20 especialistas en el tema, dentro de este grupo hay diversidad en los roles que desempeñan en su actual ocupación pero todos de alguna forma u otra han estado vinculados a las pruebas de software y han estado en contacto con el PPL que está vigente en el LIPS. Los resultados de las preguntas seleccionadas por el grupo experto se muestran a continuación .Las respuestas opcionales a las preguntas cualitativas (8, 11, 16, 17y18) que en este caso no fueron seleccionadas para la validación también serán empleadas como refuerzo a las conclusiones de la validación.

Resultados del Cuestionario

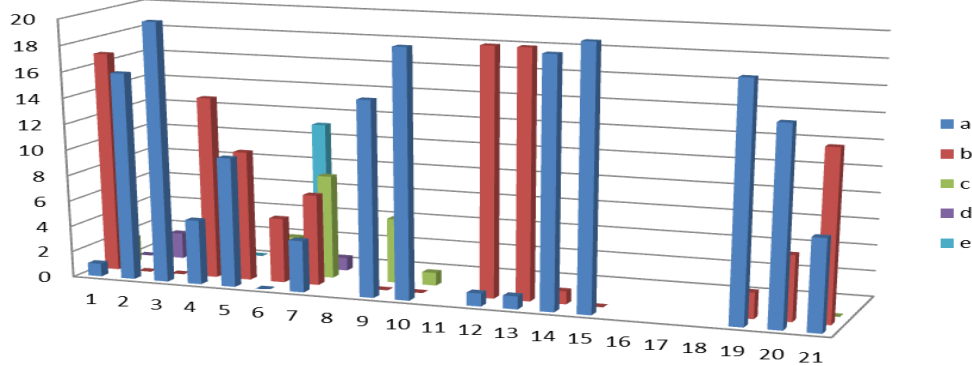


Ilustración 11 Resultados del Cuestionario

Elemento 1	Definen las pruebas	De manera significativa	Influyen muy poco	No tienen nada que ver
¿Cómo cree que influyan los riesgos de la calidad del sistema en las pruebas de liberación?	1	17	2	0
%	5%	85%	10%	0%

Elemento 2	Si	No	Quizá	Sería interesante
¿Cree usted que las actividades de mitigación de riesgos que se hacen en el ciclo de desarrollo del software, deba incluir los riesgos del producto?	16	0	2	2
%	80%	0%	10%	10%

Elemento 3	Si	No
¿Tiene conocimientos sobre Pruebas de software?	20	0
%	100%	0%

Elemento 4	Bastos	Generales	Superficiales	Oportunos	Escasos
¿Cómo califica sus conocimientos sobre las pruebas?	5	15	0	0	0
%	25%	75%	0%	0%	0%

Elemento 5	Si	No
¿Tiene conocimientos sobre el tema de Pruebas de basadas en Riesgos?	10	10
%	50%	50%

Elemento 6	Bastos	Generales	Superficiales	Oportunos	Escasos
¿Cómo califica sus conocimientos sobre el tema?	0	5	3	1	11
%	0%	25%	15%	5%	55%

Elemento 7	Si	Considerablemente	En parte	No
¿Considera usted que el actual Proceso de Pruebas de Liberación satisface las necesidades del proyecto de garantizar la calidad del software liberado?	4	7	8	1
%	20%	35%	40%	5%

Elemento 9	Si	No	No del todo
¿Entiende usted el procedimiento propuesto?	15	0	5
%	75%	0%	25%

Elemento 10	Si	No	No del todo
¿Lo ve útil?	19	0	1
%	95%	0%	5%

Elemento 12	Si	No
¿Considera que debe añadirse o restarle algún artefacto?	1	19
%	5%	95%

Elemento 13	Si	No
¿Considera que debe añadirse o restarle alguna actividad?	1	19
%	5%	95%

Elemento 14	Si	No
¿Considera que está bien enunciado?	19	1
%	95%	5%

Elemento 15	Si	No
¿Considera que satisface las necesidades del producto para garantizar su calidad?	20	0
%	100%	0%

Elemento 19	Si	No
¿Está vinculado de alguna forma al DPSW?	18	2

%	90%	10%
---	-----	-----

Elemento 20	Si	No
¿Había oído hablar sobre el tema antes?	15	5
%	75%	25%

Elemento 21	Muy útil	Si	No
¿Le pareció útil la encuesta?	7	13	0
%	35%	65%	0%

Tabla 4. Resultados del cuestionario definitivo.

3.2.5 Conclusiones de la Validación por Delphi

El método Delphi permitió establecer una comparación en los criterios de diferentes especialistas, sin despreciar la respuesta de ningún cuestionario. Los especialistas seleccionados forman partes de los centros DATEC, ISEC, FORTES , Empresa QUIMEFA y la gran mayoría labora en CALISOFT.

-Sobre los riesgos y su influencia en las pruebas el 85% de los evaluadores cree que es significativa dicha influencia, un especialista alega que las definen, y dos concuerdan en que influyen muy poco .En un segundo encuentro estos dos últimos alegaron que su criterio debía ser sumado al del 85% pues no habían comprendido que tipo de riesgos influían en las pruebas, es decir que los riesgos de manera general no pero si estaban de acuerdo con los del producto.

-Todos los especialistas tienen conocimiento sobre pruebas de software, el 25% posee conocimientos bastos y el 75% general, esto se explica perfectamente por los años de experiencia que cada cual tiene en el tema y lo cual es muy importante considerando que ninguno alcanza 10 años.

- De manera general el procedimiento tuvo una aceptación del 95% de los especialistas que evaluaron el mismo. Y el especialista que representa el 5% presentó una sugerencia muy buena que será plasmada en las recomendaciones del trabajo.

- Esta evaluación fue de cierta forma el primer acercamiento a las Pruebas Basadas en Riesgo para la mitad de los especialistas, quienes una vez leído el procedimiento llegaron a entenderlo perfectamente en un 75% y un 25% alegó que no del todo por el caso de los

artefactos que propone y que serán adaptados en mayor medida a un lenguaje más natural para la comprensión por igual de todos los especialistas que deban trabajar con ellos.

- El 100% de los evaluadores reconoció que el procedimiento satisface todos los requerimientos que deben ser vistos del producto para garantizar su calidad.

3.3 Validación por Casos de Estudio.

3.3.1 Muestra

A continuación se realizara un estudio de los resultados alcanzados con la implantación del Procedimiento para Pruebas Basadas en Riesgos. Los resultados son el reflejo de la aplicación del procedimiento a 3 productos liberados en el LIPS.

Proyecto1: El primer proyecto analizado fue el Sistema de Información Geográfica de Salud. Este sistema tiene como propósito general brindar información geográfica mediante el empleo de mapas con señalizaciones y símbolos para lugares determinados. En este caso y siempre serán resaltados en ellos las organizaciones, casas y edificaciones del sistema de salud bolivariano. En la sesión de análisis de riesgos del proyecto se detectaron 2 riesgos de Funcionalidad y 2 de Seguridad para hacer un total de 4. Donde el primer riesgo de funcionalidad estuvo asociado a 19 requisitos y el segundo a solo 1 y por su parte los de seguridad estuvieron asociados a un requisito cada uno para hacer un total de 22 requisitos identificados como vulnerabilidades del sistema y posibles afectaciones a la calidad del mismo.

No	Característica	Especificación del Riesgo	Especificación de Requisitos
Funcionalidad			
1		No se cargan los mapas en el sistema.	RF3 al RF11, RF12, RF14 al RF18
2		Los datos no se salvan correctamente.	RF12
Seguridad			

3	Fallos que atentan contra el uso adecuado del sistema teniendo en cuenta los niveles de acceso y permisos.	RF18
4	Se puede acceder al sistema solo poniendo la url en el navegador de una de las páginas internas del sistema.	RF1

Tabla 5. Análisis de riesgos del Proyecto1

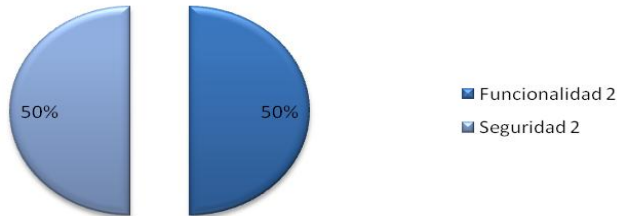


Ilustración 12 Riesgos del Proyecto1

Proyecto2: El segundo proyecto estudiado fue el Sistema de Servicios de Gestión a Servidores PostgreSQL. Este proyecto tiene su base en la necesidad de contar con reportes actualizados del estado de los servidores y gestores de Postgre que se estén usando en el centro para el desarrollo de otros proyectos, estableciendo alertas cuando estos exceden algún número definido como capacidad máxima ya sea de memoria, almacenamiento, cantidad de consultas, métricas, etc. De esta forma se evitan desbordamientos del uso de la memoria y se mantiene control sobre los medios y recursos utilizados para el desarrollo de aplicaciones con Postgre.

En la sesión de análisis de riesgos fueron detectados 5 riesgos de Funcionalidad, 1 de Seguridad, 1 de Manejo de datos 2 de Mantenibilidad y 1 de Soporte, para hacer un total de 10.

Donde los riesgos de funcionalidad estuvieron asociados a 55 requisitos, el de seguridad a todo el sistema de manera general, los de manejo de datos a 40 coincidentes con algunos de funcionalidad, los de Mantenibilidad a 4 y el de soporte se probaron en la instalación del sistema y configuración.

No	Característica	Especificación del Riesgo	Especificación de Requisitos
Funcionalidad			
1		Existen comandos de la librería de datos que no funcionan.	MA9.1-9.4
2		Fallos en las Descargas de documentos en línea.	MA6.1-6.15,A2.3,M5.1
3		No se muestran correctamente las gráficas y tablas de datos.	MA2.1-2.15,3.1-3.5,6.1-6.15,8.2,A2.2,M2.4,4.1,B1.3
4		Las búsquedas no funcionan correctamente.	MA1.1,4.1,7.1,8.1,A1.1,2.1,3.1,3.2,B1.2
5		Los reportes no se muestran correctamente.	MA2.1-2.15,3.1-3.5,6.1-6.15,8.2,A2.2,M2.4,4.1,B1.3
Seguridad			
6		Se puede acceder al sistema solo poniendo la url en el navegador de una de las páginas internas del sistema.	MA1.1
Manejo de datos			
7		Los campos de fecha no son calculados, actualizados o mostrados correctamente.	MA2.1-2.15,3.1-3.5,6.1-6.15,8.2,A2.2,M2.4,4.1,B1.3
Mantenibilidad			
8		Los procesos de respaldo del sistema afectan la disponibilidad del mismo.	MA9.1-9.4
9		Los procesos de respaldo del sistema provocan perdida de información.	MA9.1-9.4
Soporte			
10		Proceso de Instalación / configuración para la instalación falla.	Sistema

Tabla 6. Análisis de Riesgos del Proyecto2

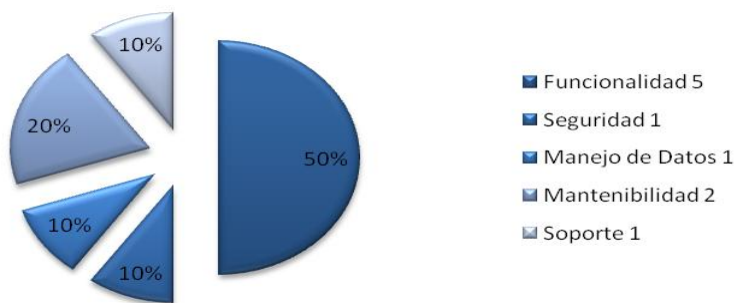


Ilustración 13 Riesgos del Proyecto2

Proyecto4: Es un sistema web que permite la gestión de documentos históricos. Surge por

la necesidad de crear un archivo histórico para salas de lectura ,lo cual es de suma importancia pues permite establecer búsquedas rápidas mediante catálogos , y garantiza la existencia de la información registrada en el mismo brindando datos fiables sobre los medios existentes en las salas de lectura el estado de los mismos y los requerimientos de manejo para su cuidado y preservación. La revisión de de este proyecto es referente a la segunda versión del proyecto, conformada por 52 por casos de uso.

En el análisis de riesgos fueron detectados 5 riesgos de Funcionalidad, 4 de Seguridad y 3 de Usabilidad, para hacer un total de 12.

Donde los riesgos de funcionalidad estuvieron asociados a 15 requisitos, los de seguridad a todo el sistema de manera general, y los de Usabilidad 10 requisitos.

No	Característica	Especificación del Riesgo	Especificación de Requisitos
Funcionalidad			
1		Existen comandos de la librería de datos que no funcionan.	
2		Los datos no se salvan correctamente.	
3		Fallos en la conexión a Internet (Existen funcionalidades que requieren de internet y otras de acceso directo).	
Seguridad			
4		Se puede acceder al sistema solo poniendo la url en el navegador de una de las páginas internas del sistema.	
5		Ejecución de código php o consultas SQL desde los formularios de la aplicación.	
6		Nivel de permisos en el servidor de aplicaciones que permita listar el código fuente.	
7		Autenticación incorrecta y no controlada a la administración del sistema.	
Usabilidad			
8		Visibilidad distorsionada en algún navegador Web.	
9		Poco entendimiento de los formularios de búsqueda.	
10		No ejecución de funciones de autocompletamiento, Hide-Show, tooltips y slideshow.	
Soporte			
11		Incompatibilidad con la versión anterior del sistema.	

Tabla 7. Análisis de Riesgos del Proyecto3

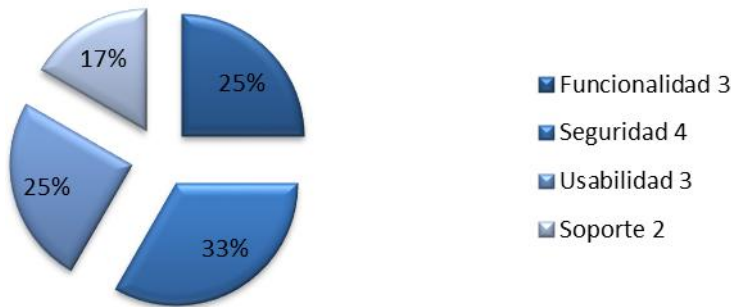


Ilustración 14 Riesgos del Proyecto3

3.3.2 Analisis de las Pruebas Basadas en Riesgos

Para el análisis de riesgos de cada proyecto estuvieron presentes los analistas quienes en conjunto con el especialista de pruebas determinaron el impacto y la probabilidad de ocurrencia para cada riesgo específico y de esta forma se calcularon las prioridades para las pruebas y la extensión de dichas pruebas. Los resultados en las Pruebas basadas en riesgos fueron los registrados a continuación.

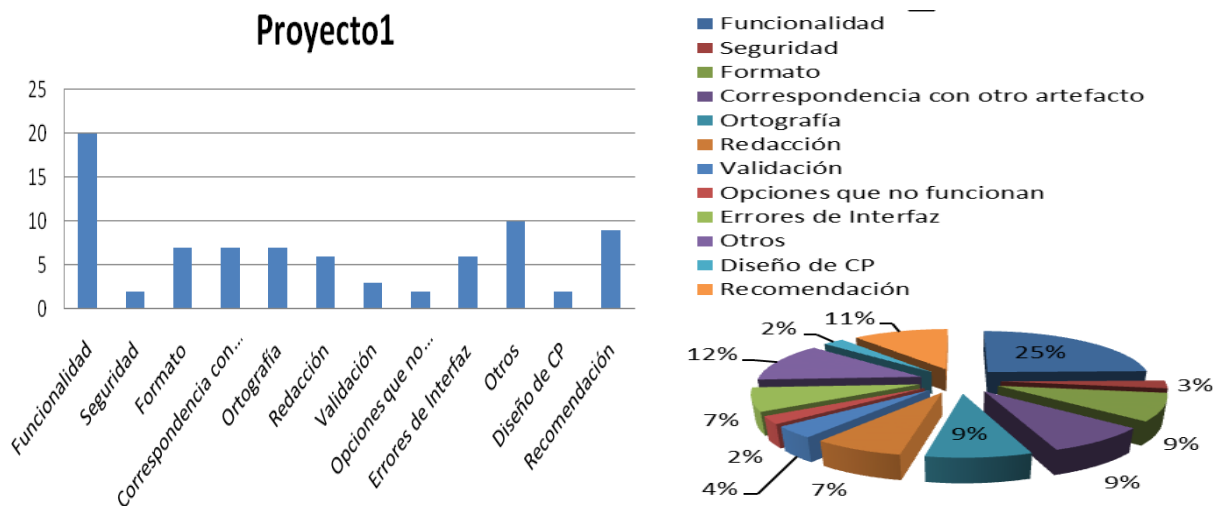


Ilustración 15 Resumen de No Conformidades

Analizando el número de NC por tipo se puede concluir diciendo que el mayor número estuvo concentrado en los problemas con las funcionalidades básicas del sistema, las cuales fueron

identificadas como riesgos importantes , asociados a 20 RF , los cuales arrojaron 20 NC en la prueba del sistema.El otro número considerable es de las NC con tipo Otros que hicieron referencia a temas de diseño y visualización de iconos , colores y apariencia del sistema en general , estas características no fueron vistas en el análisis de riesgos pues vienen dadas por el tipo de sistema que es y las exigencias que este debe cumplir teniendo en cuenta que una de las características mas importantes a cumplir es que brinde siempre la información visual correcta .Este elemento pudo tenerse en cuenta en el análisis de riesgo pero los participantes no lo consideraron determinante para la calidad del sistema en general , por eso es recomendable la presencia del cliente siempre que sea posible en la sesión de análisis pues criterios como la usabilidad del sistema y la apariencia que debe tener para el usuario muchas veces es del dominio del cliente y del equipo de desarrollo pero no se recogen en pautas de diseño o en el documento de especificación de requisitos , como requisitos de diseño de interfaz.En resumen hubo un 50% de las NC detectadas que están directamente asociadas a los riesgos identificados.Puede parecer poco significativo , pero teniendo en cuenta el sistema que se revisó y los errores graves de Ortografía y Redacción unido a los problemas de atractividad en la visualización ,pues resulta entonces muy interesante que en una sesión de análisis se hayan podido prever cuales serían los posibles fallos que el sistema podría presentar y una vez verificados, haber satisfecho estas interrogantes en tan poco tiempo y contar con un sistema más completo y con estos riesgos mitigados luego de que se resuelvan las NC .En cuanto a cumplimiento de la norma ISO/IEC25000:2005 el proyecto resumió sus problemas a un 4% de funcionalidad y un 2% de seguridad. Las otras características no aplicaban por falta de requisitos.

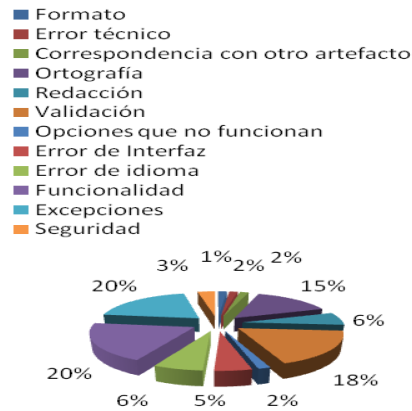
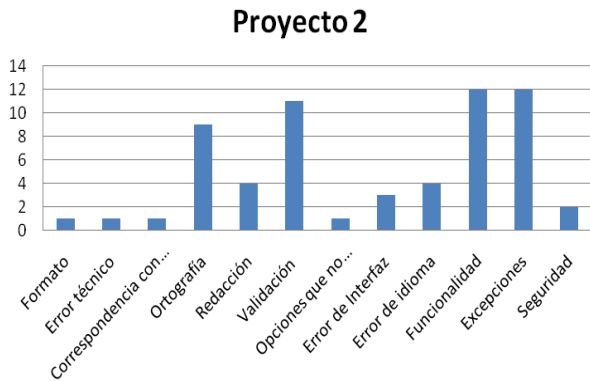


Ilustración 16 Resumen de No Conformidades

Las pruebas del Proyecto 2 fueron un poco más complejas pues habían más riesgos a comprobar y por tanto la asociación con los requisitos fue mas abarcadora.No obstante la prueba se hizo en muy poco tiempo gracias a las descripción de los casos de prueba que ya tenían bien desarrollados el equipo de proyecto y que agrupaban muchas de estas funcionalidades , facilitando de esta forma la simpleza en las pruebas y el empleo del menor tiempo posible.Como se ve en la gráfica la mayoría de las NC estuvieron concentradas en problemas de funcionalidad , validación y excepciones del sistema debido a su poca estabilidad en la primera iteracion de pruebas, sin despreciar las faltas de ortografía que tenían las páginas que también generó el otro número sobresaliente de NC.Las NC de validación también estuvieron asociadas directamente a los requisitos probados basados en los riesgos de funcionalidad detectados.Demostrando de esta forma que el gran cúmulo de errores detectados en las pruebas del sistema están asociados directamente a los riesgos del producto indetificados en la sesión de análisis.Del total de NC que fueron 61 ,las asociadas al analisis de riesgos (funcionalidad,excepciones,validación,seguridad,opciones que no funcionan) hacen un 63% del total. Dejando un 37% correspondiente a Ortografía,Redacción, Idioma,Formato,etc. En cuanto a cumplimiento de la norma ISO/IEC25000:2005 el proyecto resumió sus problemas a un 20% de funcionalidad , 1% de seguridad,0%mantenibilidad y 0% de soporte. Las otras características no aplicaban por falta de requisitos.

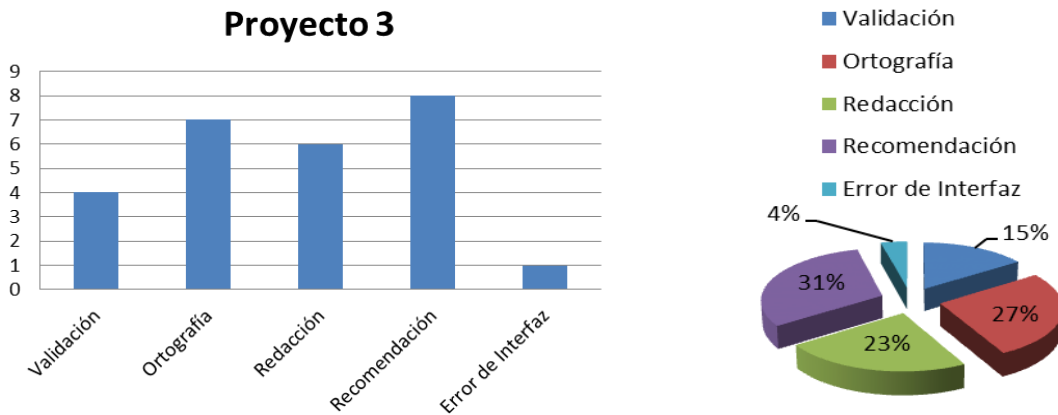


Ilustración 17 Resumen de No Conformidades

El Proyecto 3 se comportó muy diferente a los dos anteriores ,para este caso solo un 19% de las NC detectadas estuvieron asociadas a los riesgos identificados en la sesión de análisis.Es deducible que este proyecto en cuestión de revisiones internas y por el cliente ,tienen más experiencia que el resto de ahí que las NC de validación fueran las únicas referentes a la implementación que perduraran para esta etapa.

En resumen los resultados de la pruebas fueron bastante satisfactorios , sobre todo teniendo en cuenta que en las iteraciones restantes no salieron un gran numero de NC y las que resultaron nuevas fueron generadas en la etapa de resolución de las primeras.No estaban en el sistema , como debe ser.A continuación se muestran los resultados de los tres proyectos comparando para los tres el número de NC asociadas a los riesgos sobre el total de las NC detectadas. En cuanto a cumplimiento de la norma ISO/IEC25000:2005 el proyecto no presentó problemas para los riesgos detectados en las características de funcionalidad, seguridad,usabilidad y soporte . Las otras características no aplicaban por falta de requisitos.

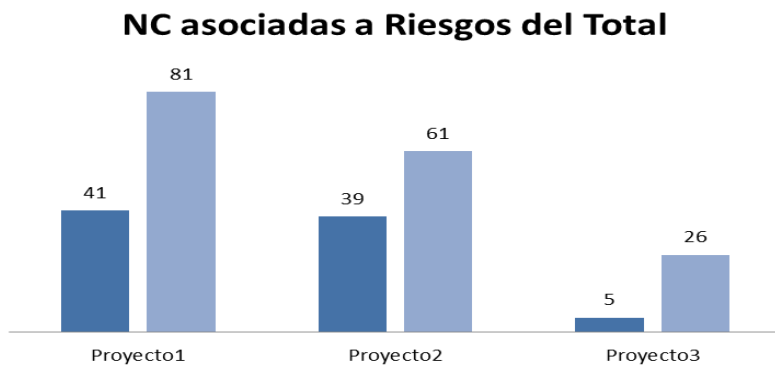


Ilustración 18 Comparación entre Proyectos.

En esta ilustración la serie uno representa las NC asociadas a los riesgos identificados mientras que la serie 2 se refiere al total de NC detectadas en la iteración. Es apreciable que en los tres casos el número de la serie 1 se comportó de forma diferente para el caso del proyecto tres ya se explicaron las causas, pero el comportamiento general debe ser muy similar a los resultados de los proyectos 1 y 2 pues más del 60% de los proyectos que entran a Liberación apenas han sido revisados integralmente una vez antes de su entrega.

La relación en por cientos para los tres proyectos fueron los siguientes:

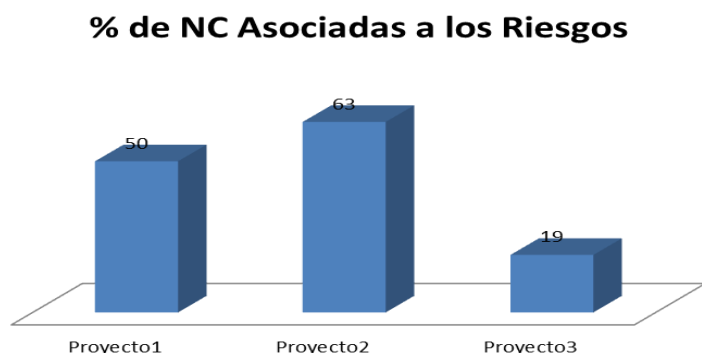


Ilustración 19 Por Ciento de No Conformidades asociadas a riesgos.

Los resultados para cada proyecto estuvieron influenciados como es lógico por el tipo de proyecto, por la estrategia de prueba desarrollada en cada equipo de desarrollo previamente a la liberación de los mismos y la profundidad del análisis en las sesiones de identificación de riesgos. Lo cual alerta de la necesidad de establecer mejores estrategias de pruebas en los proyectos 1 y 3, además de involucrar más personal en la sesión de análisis de manera que aumente en criticidad y permita formar una cultura más amplia en cuanto a las técnicas de análisis de riesgos y como asegurar que estos sean mitigados antes de llegar a la prueba de liberación en etapas de prueba previas.

3.3.3 Conclusiones de la Validación

Teniendo en cuenta los resultados de las pruebas realizadas a los proyectos 1,2 y 3 podemos concluir anotando varios aspectos interesantes.

-En dos proyectos más del 50% de las NC detectadas fueron fallos en funcionalidades asociadas directamente a los riesgos identificados en sus respectivas sesiones de análisis.

-El tiempo que tomó para revisar las funcionalidades asociadas a los riesgos fue muy poco

comparado al tiempo que normalmente se emplea para revisar estos proyectos íntegramente. (Esto por supuesto varía para cada proyecto pero por lo general en todos se redujo el tiempo considerablemente.) Normalmente un proyecto como el 2 que tiene 28 HU hubiese sido necesario emplear dos sesiones de trabajo para un probador experto como los especialistas del LIPS, o una sesión de trabajo en el LIPS con 14 puestos de trabajo cubiertos por estudiantes. Sin embargo al especialista le tomó solo dos horas, la mitad de una sesión, para verificar las funcionalidades señaladas como más riesgosas. En el resto de los proyectos, que eran más pequeños, no tomó más de dos horas a los especialistas para dar los dictámenes referentes al estado de cada sistema. Es válido aclarar que normalmente toma más de una jornada de trabajo hacer la revisión y el informe del tipo de errores con la cantidad de NC tipo para cada uno, pero partiendo de que las funcionalidades ya estaban directamente asociados a los riesgos y estos a su vez agrupados en características de calidad pues esto ya estaba adelantado para presentar el informe.

3.4 Conclusiones del Capítulo

Del presente capítulo se concluye lo siguiente:

1. Se validó la propuesta por el método Delphi lo que permitió que cada experto haciendo uso de su criterio, diera el visto bueno al procedimiento.
2. Con la prueba basada en riesgos se detectó el mayor número de NC significativas demostrada a través de la segunda validación en dos proyectos.
3. Las pruebas basadas en riesgos permitieron establecer en poco tiempo una valoración del estado de la calidad del software en cuanto a las características de calidad según la ISO/IEC25000:2005.
4. Debido a que las NC detectadas ya están asociadas a una característica de calidad específica y en el análisis se revisan todas, las pruebas se realizaron teniendo en cuenta todos los riesgos de la calidad del software y por consiguiente hay una garantía de mayor calidad en las mismas y en los artefactos liberados.

5. Se logró disminuir el tiempo de ejecución de una iteración de pruebas, lo que hace el proceso más ágil y efectivo, pues no deja de contemplar la calidad del sistema que se libera.

4 Conclusiones

Sobre el trabajo realizado se concluye lo siguiente:

1. Se diseñó e implementó un procedimiento para aplicar pruebas basadas en riesgo introduciendo nuevas técnicas para el análisis de riesgos y selección de las entradas de las pruebas de una forma diferente, teniendo en cuenta la norma internacional ISO/IEC 25000:2005.
2. Con la implantación del procedimiento se ponen en práctica los resultados de las investigaciones realizadas en el GIPS sobre las características o atributos de calidad y los diferentes tipos de pruebas complementando el servicio a brindar y elevando la calidad del proceso mismo.
3. Se diseñaron los artefactos que permitan un trabajo más sencillo y enfocado a lo que realmente se debe revisar en el sistema, teniendo en cuenta la preparación que tiene el estudiante de 2do año dado que son la principal fuerza de trabajo del LIPS.
4. El procedimiento implementado fue construido a partir de la situación actual y las necesidades persistentes del DPSW, emplea artefactos del Expediente de Proyecto v3.3 la cual está vigente (desarrollada en el Proceso de Mejora a partir del nivel 2 de CMMI) , y sus actividades y artefactos propios no modifican o sustituyen alguno existente , pero si pueden ser adaptados a proyectos determinados o probadores específicos(sin entrar en conflicto con los objetivos del procedimiento mismo o las normativas existentes en el LIPS) en dependencia de las necesidades del laboratorio de pruebas que pretenda adoptarlo.
5. Se validó el procedimiento empleando dos métodos reconocidos mundialmente, dejando constancia de la efectividad del mismo en la reducción del tiempo de evaluación del software de manera integral y en la evaluación misma respecto a las características de la norma ISO/IEC 25000:2005.
6. Se demostró la hipótesis de que con la implantación de un procedimiento para aplicar pruebas basadas en riesgos en el LIPS aumentó la calidad de las pruebas desarrolladas, disminuyó el tiempo de desarrollo de las mismas y permitió brindar una evaluación completa en cuanto a cumplimiento de la norma ISO/IEC 25000:2005.

5 Recomendaciones

El Procedimiento de Pruebas Basadas en Riesgos constituye una adaptación al proceso de Pruebas de Liberación, dicho proceso está sujeto a los cambios organizacionales que puede sufrir el centro incluso ha servido de base para otros procesos desarrollados como el de Aceptación y el de Evaluación para clientes externos y que son servicios que brinda la empresa actualmente. Por lo tanto se recomienda:

1. Completar progresivamente el análisis de los beneficios que reporta la incorporación de las técnicas de análisis de riesgos para la organización en general y sus servicios.
2. Implementar versiones del mismo para los servicios de Aceptación y Evaluación.
3. Implementar una herramienta que gestione la información de los artefactos propuestos y que permita brindar los reportes de manera automatizada.
4. Incorporar las técnicas de análisis de riesgos a la calidad no solo del producto sino del proceso mediante el estudio de los factores influyentes en la satisfacción del cliente.

6 Referencias Bibliográficas

- Adranovich, G. 1995.** *Developing community participation and consensus: The Delphi technique.* 1995.
- B.G.Ludwig. 1996.** *U.S. Extension systems—Facing the Challenge to Internationalize.* 1996.
- Bass, B. 1983.** *Organizational decision making.* Homewood,IL : s.n., 1983.
- Bernard, H. 1988.** *Research methods in cultural anthropology.* Newbury Park:Sage. 1988.
- Capote, Tayche. 2011.** *Conceptualización del Laboratorio Industrial de Pruebas de Software.* 2011.
- Critical Testing Procesess.* **Black, Rex. 2003.** 2003.
- DefinicionAbc.** Definicionabc. [En línea] <http://www.definicionabc.com/general/prueba.php>.
- GNM, Calisoft. 2010.** *Libro de procesos del LIPS.* 2010.
- IIA. 2004.** *Institute of Internal Auditors.* 2004.
- ISO. 2002.** *ISO 27000.* 2002.
- . 1994.** *ISO 8402.* 8402 1994.
- Ivar Jacobson,Booch,Rumbaugh. 1999.** *El proceso unificado de desarrollo de software.* 1999.
- Jones, Capers. 1995.** *Patterns of Large Software Systems: Failure and Success .* New York : Mc Graw-Hill, 1995.
- Jorrín, Michael González. 2005.** *Proceso de Pruebas de Liberación.* 2005.
- LA APLICACIÓN DE LA TÉCNICA DELPHI, PARA LA CONSTRUCCIÓN DE UN INSTRUMENTO DE ANÁLISIS CATEGORIAL DE INVESTIGACIONES E-LEARNING.* **Julio Cabero, Julio Barroso,Rosalía Romero, Pedro Román,Cristobal Ballestero, María del Carmen. 2009.** 28, España : EDUTEC,. **2008.** 2008.
- 2006,2007.** Lenguajes de Simulación. [En línea] 2006,2007. http://jair.lab.fi.uva.es/~pablfue/leng_simulacion/slides/0607/vv_0607_t.pdf..
- Linstone H.A,Turrof M. 1975.** *The Delphi method: Techniques and applications.* 1975.
- N.C.Dalkey,O.Helmer. 1963.** *An experimental application of the Delphi. Method.* 1963.
- Palazzolo, Cecilia. 2005.** *Calidad de Software¿Qué es la calidad de software?* 2005.
- Pes, Carlos. 2006.** CarlosPes.com. [En línea] 2006. http://www.carlospes.com/curso_de_ingenieria_del_software/01_04_calidad_del_software.ph

p.

Pressman, Roger. 1998. *Calidad de Software. Ingeniería de Software.* 1998.

—. **1992.** *Ingeniería de Software Un enfoque práctico.* 1992.

Pruebas Basadas en Riesgos. **Heydi Menéndez, Aniubis Rodríguez, Yanetsi Millet. 2010.**

La Habana : s.n., 2010.

Pruebas de software. **Choucair, María. 2007.** Bogotá : s.n., 2007.

Quality Risk Analysis. **Black, Rex. 2008.** 2008.

Rex Black, Peter Nash, Keng Young. 2008. *Case Study in Successful Risk-Based Testing at CA.* 2008.

Shaefer, Hans. 2004. *Risk Based testing.* 2004.

Vega Lebrún Carlos; Rivera Prieto Laura; García Santillán Arturo. 2008. eumed.net.

MEJORES PRÁCTICAS PARA EL ESTABLECIMIENTO Y ASEGURAMIENTO DE LA CALIDAD DE SOFTWARE. [En línea] 2008.

<http://www.eumed.net/libros/2008a/351/ESTABLECIMIENTO%20Y%20ASEGURAMIENTO%20DE%20LA%20CALIDAD%20DE%20SOFTWARE%20situacion%20actual.htm>.

Violena Hernández, Heydi Menéndez, Yadira Machado, Odelkys Betancourt. 2009. *Estrategias de Pruebas.* 2009.

2010. Wikipedia. [En línea] Fundación Wikimedia, Inc., 2010. <http://es.wikipedia.org/wiki/Prueba>.

Young, Lee. 2009. *Competencias Hended by Korean HRD Master's Graduates: A Compararison Between the ASTD WLP Competency Model and the Korean Study.* 2009.

7 Anexos

7.1 Anexo1 Tabla para el Cálculo del Impacto del Riesgo

Impacto	Valor	Comentarios
Corrección inmediata	1	Prioridad uno
Corrección Programada	2	Programación en cronograma para su corrección lo antes posible
Debe corregirse	3	No debe recibir atención mientras las otras dos categorías superiores no se hayan cubierto totalmente
Sería bueno corregirlo	4	Algo que el cliente tomaría positivamente si se arreglara pero no atenta contra el funcionamiento.
No corregir	5	Ningún esfuerzo para solucionar este problema

7.2 Anexo2 Tabla para el Cálculo de la Probabilidad del Riesgo

Probabilidad	Valor	Comentarios
Muy Probable	1	Muy cercano a pasar
Probable	2	Más cercano a pasar que a no pasar
Algo Probable	3	Pocas posibilidades de que pase
Improbable	4	Menos cercano a pasar que a no pasar
Muy Improbable	5	No debe pasar

7.3 Anexo3 Tabla para el Cálculo de la Prioridad del Riesgo

Prioridad del Riesgo	Extensión de la Prueba	Comentarios
----------------------	------------------------	-------------

1-12	Extenso	Ejecutar un número grande de pruebas de forma general y profunda, ejercitando combinaciones y variaciones de condiciones diferentes para las pruebas.
13-16	General	Ejecutar un número mediano de pruebas que ejerciten muchas condiciones diferentes de la prueba.
17-20	Superficial	Ejecutar pocas pruebas que ejemplifiquen las condiciones más probables del sistema
21-25	Oportuno	Favorecer otras actividades o realizar una prueba o dos de una condición interesante, pero sólo si se trata de una inversión muy pequeña de tiempo y esfuerzo, y sólo si se presenta la oportunidad.

7.5 Anexo4 Lista de Chequeo de Riesgos

Lista de Chequeo de Riesgos

<Nombre del Proyecto>

<Nombre del producto>

<Versión>

7.6 Cuestionario Definitivo

Cuestionario definitivo para aplicar a las personas vinculadas al desarrollo y pruebas de software.

Centro: _____ Edad: _____ Género: M F

1-¿Cómo cree que influyan los riesgos de la calidad del sistema en las pruebas de liberación?

Definen las pruebas De manera significativa Influyen muy poco No tienen nada que ver

2-¿Cree usted que las actividades de mitigación de riesgos que se hacen en el ciclo de desarrollo del software, deba incluir los riesgos del producto?

Si No Quizá Sería interesante

3-¿Tiene conocimientos sobre Pruebas de software?

Si No

4-¿Cómo califica sus conocimientos sobre las pruebas?

Bastos Generales Superficiales Oportunos Escasos

5-¿Tiene conocimientos sobre el tema de Pruebas de basadas en Riesgos?

Si No

6-¿Cómo califica sus conocimientos sobre el tema?

Bastos Generales Superficiales Oportunos Escasos

7-¿Considera usted que el actual Proceso de Pruebas de Liberación satisface las necesidades del proyecto de garantizar la calidad del software liberado?

Si Considerablemente En parte No

8-¿Por qué?(Opcional)

9-¿Entiende usted el procedimiento propuesto?

Si No No del todo

10-¿Lo ve útil?

Si No No del todo

11-¿Por qué?(Opcional)
