

Universidad de las Ciencias Informáticas
UCI

Estrategia para elevar la eficiencia del servicio de
evaluación de productos de software en el
Laboratorio Industrial de Pruebas de Software de
CALISOFT.

Tesis en opción al título de
Máster en Calidad de Software.

Por

Autor: Ing. Heney Díaz Pérez

Tutoras: MsC. Tayché Capote García

Dra. Ailyn Febles Estrada

La Habana, Cuba.
Junio de 2014.

DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor del presente trabajo. Autorizo al Centro Nacional de Calidad de Software (CALISOFT) y a la Universidad de las Ciencias Informáticas (UCI) a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Ing. Heney Díaz Pérez
Autor

MsC. Tayché Capote García
Tutora

Dra. Ailyn Febles Estrada
Tutora

DEDICATORIA

A mis padres, por todo el apoyo y sacrificio para llegar hasta aquí.

A mi esposa, por ser mi reina, el regalo más lindo que me ha dado la vida.

A mi hermano, porque cuando sea grande quiero ser como él.

AGRADECIMIENTOS

Quiero agradecer a mi esposa por acompañarme en todo momento, por su paciencia y sobre todo, por su confianza.

A mis padres por todo su apoyo, porque les debo tantas enseñanzas que me han permitido llegar hasta aquí.

A mi familia y a la familia de mi esposa, por estar al tanto de mi superación y de la conclusión de la maestría.

A mis tutoras, a Tayché por todo su apoyo y por toda la admiración y el respeto que me inspira y a Ailyn, por ser para todos nosotros, la madre de la calidad de software en Cuba.

A todos los profesores y profesoras que conforman el claustro de la Maestría en Calidad de Software.

Al profesor Mauricio, por su preocupación y colaboración.

A mis amigos y hermanos, a Lisset, Yoan, Celia, Yudisbel, Yeni, Alionuska, Yenier, Yanet, Kariné, Pepe, Liana, Idalis. Gracias por ser mis amigos y por aguantarme.

A mis compañeros de trabajo, a los chicos y chicas del LIPS, del departamento, a todos los que juntos trabajamos en la concepción y materialización del Laboratorio.

A todos los que de una manera u otra, contribuyeron al desarrollo de esta investigación. Sinceramente... Muchas Gracias...

RESUMEN

Las actividades de verificación y validación favorecen la obtención de aplicaciones informáticas con elevados indicadores de calidad. La gestión de pruebas permite desarrollar estas actividades de manera ordenada, para contribuir a elevar la eficiencia y así obtener mejores resultados. En esta investigación se presenta el diseño de una estrategia que permita elevar la eficiencia del servicio de evaluación de productos en el Laboratorio Industrial de Pruebas de Software (LIPS) del Centro Nacional de Calidad de Software (CALISOFT). Se parte de un estudio de los principales referentes teóricos asociados a los procesos de pruebas de software. Se realiza una caracterización de las actividades que se desarrollan en el LIPS, partiendo de la identificación y comportamiento de los elementos que favorecen la gestión de las pruebas. Luego se presenta la estrategia en dos etapas. La primera determina el establecimiento del proceso de evaluación de productos, con la descripción de las actividades, roles y responsabilidades y la segunda establece las actividades de seguimiento y control del proceso. Luego se realiza un análisis del comportamiento de la gestión de las pruebas y la eficiencia del servicio para validar la estrategia, con una triangulación de métodos, a partir de la aplicación de un cuestionario, seguido de la valoración de su implementación parcial y de los criterios de un grupo de expertos. La validación evidenció el cumplimiento del objetivo de la investigación al lograr elevar la eficiencia del servicio de evaluación de productos en el LIPS.

Palabras claves: Evaluación de productos, Gestión de pruebas, Proceso.

ÍNDICE

Introducción	1
Capítulo I. Fundamentación teórica	9
1.1 Introducción	9
1.2 La definición de Proceso. Su aporte a la gestión de pruebas de software.....	9
1.2.1 Componentes de los procesos.....	11
1.3 Procesos de pruebas de Software	14
1.4 Estructuras organizativas para el desarrollo de pruebas de software.....	17
1.4.1 Los desarrolladores son probadores.....	17
1.4.2 Equipos de pruebas integrados	18
1.4.3 Equipos de pruebas independientes	18
1.4.4 Grupos de aseguramiento de la calidad.....	18
1.4.5 Factoría de pruebas.....	19
1.5 Modelos relacionados con la gestión de pruebas.....	21
1.5.1 CMMI for Development	21
1.5.2 TMM – Test Maturity Model	22
1.5.3 TMMi – Test Maturity Model Integration	22
1.5.4 TMap – Test Management Approach.....	23
1.5.5 TPI – Test Process Improvement.....	24
1.6 Otras iniciativas para la gestión de pruebas de software	26
1.7 Caracterización del entorno	27
1.7.1 Principios generales del funcionamiento del LIPS.....	29
1.8 Conclusiones parciales del capítulo	32
Capítulo II. Planteamiento de la estrategia.....	34
2.1 Introducción	34
2.2 Definición de la estrategia.....	34
2.2.1 Etapa I. Definición del proceso Evaluación de Productos de Software... 35	
Primeros acercamientos al proceso de evaluación de productos de software .. 36	
Descripción del proceso Evaluación de Productos de Software	39
Subproceso Gestionar Solicitudes	40
Subproceso Planificar Evaluación de Productos de Software	41
Subproceso Ejecutar Evaluación de Productos de Software.....	42
Subproceso Cerrar Evaluación de Productos de Software.....	43

Definición de roles y responsabilidades.....	45
Roles administrativos.....	46
Roles ejecutivos.....	48
2.2.2 Etapa II. Definición de las actividades de seguimiento y control	52
Actividades de seguimiento.....	53
Actividades de control de calidad	55
Actividades de revisión.....	56
Reunión de Seguimiento	58
2.3 Conclusiones parciales del capítulo	58
Capítulo III. Validación de la estrategia	59
3.1 Introducción	59
3.2 Evaluación del establecimiento del proceso Evaluación de Productos de Software	59
3.3 Análisis de la eficiencia del servicio Evaluación de Productos	63
3.4 Evaluación de la estrategia mediante valoración de expertos	67
3.4.1 Selección de los expertos	67
3.4.2 Escalamiento de Likert	68
3.5 Triangulación de los resultados	69
3.5.1. Triangulación de métodos y evaluación de los resultados.	70
3.6 Conclusiones parciales del capítulo	71
Conclusiones	72
Recomendaciones	73
Bibliografía.....	74
Anexos.....	80

ÍNDICE DE FIGURAS

Figura 1. Representación gráfica del concepto Proceso.	11
Figura 2. Estructura del Departamento de Pruebas de Software.....	28
Figura 3. Estructura del Departamento de Evaluación de Productos.	31
Figura 4. Etapas de la estrategia.	35
Figura 5. Proceso del LIPS.	36
Figura 6. Proceso general para las Pruebas de Liberación.	37
Figura 7. Proceso Evaluación de Productos de Software.....	39
Figura 8. Subproceso Gestionar Solicitudes.	40
Figura 9. Subproceso Planificar Evaluación de Productos de Software.	41
Figura 10. Subproceso Ejecutar Evaluación de Productos de Software.....	43
Figura 11. Subproceso Cerrar Evaluación de Productos de Software.....	44
Figura 12. Roles que intervienen en la evaluación de productos.....	46
Figura 13. Actividades de seguimiento y control.	52
Figura 14. Actividades asociadas a la revisión.....	57

ÍNDICE DE TABLAS

Tabla 1. Ventajas y desventajas de las estructuras organizativas para las pruebas de software.	20
Tabla 2. Responsabilidades de los roles administrativos.	47
Tabla 3. Responsabilidades de los roles ejecutivos.	49
Tabla 4. Responsabilidades del rol Cliente.	51

INTRODUCCIÓN

El grado de desarrollo alcanzado por la industria de software a nivel internacional, con la introducción de las tecnologías de la información y las comunicaciones en prácticamente todos los sectores productores de bienes y servicios, permite sustentar que se trata de un sector estratégico y de constante crecimiento. Por tal razón, insertarse en el mercado internacional requiere altos niveles de competitividad.

Según Hernández, existen dos razones por las que la industria de software y servicios puede ser una fuente de crecimiento económico. Por un lado, en una economía que se basa cada vez más en la información y el conocimiento, el software es una herramienta esencial para el aumento de la productividad de otros sectores. Por otra parte, esa industria pasa por una etapa de fuerte expansión y ofrece grandes oportunidades de exportación, al generar empleos calificados y exportaciones de bienes y servicios producidos a distancia, particularmente como resultado de los avances tecnológicos en las comunicaciones y la arquitectura de sistemas. (Hernández 2009)

Los países desarrollados continúan marcando la diferencia al afianzar su liderazgo en cuanto a la disponibilidad, uso y desarrollo de las nuevas tecnologías. Según el Informe Global de las Tecnologías de la Información, presentado por el Foro Económico Mundial, los diez primeros lugares de 134 países, en lo que a disponibilidad de las nuevas tecnologías respecta, son ocupados por economías del primer mundo (Foro Económico Mundial 2008). Independientemente de lo que muestra este informe, algunos países tercermundistas se han insertado en este mercado aumentando sus ingresos por concepto de exportación de software, demostrando que existe un mercado creciente y por explotar, que favorece el aumento de la competitividad. La primera ola de expansión destaca la conformación del núcleo de Israel, Irlanda y la India. En los años 90, se sumaron algunos países clave como Brasil, Corea del Sur, China, Costa Rica, México y Filipinas. En el caso de los tres primeros, cabe destacar que han traducido su creciente éxito en las tecnologías de la información -en especial las exportaciones de software- en generación de empleo, ganancias, divisas y crecimiento económico. (Durán, Piore, y Schrank 2005)

En la primera onda de expansión consta que Irlanda, por ejemplo, ingresó en el 2005, 1.300 millones de dólares por concepto de exportaciones. El ingreso de la India en el mismo año alcanzó 7.500 millones de dólares, a partir de un conjunto importante de inversiones en telecomunicaciones y de certificaciones de calidad, para ganar credibilidad. En 2008 exportó 40.400 millones de dólares por el mismo concepto de

exportaciones, llegando a convertirse en potencia industrial de las comunicaciones y el software. (Foro Económico Mundial 2008)

Para las empresas que se dedican al desarrollo de software, la calidad es uno de los factores de competitividad más importantes. Cuando no hay calidad en un software se pueden presentar las siguientes situaciones (Rodríguez 2010):

- Programas que no hacen exactamente lo que se espera.
- Proyectos que no terminan nunca.
- Sistemas informáticos que no se utilizan por la dificultad de su manejo.
- Productos software que son imposibles de mantener cuando desaparece la persona o personas que lo desarrollaron.
- Software poco seguro.

La calidad de software debe ser construida desde el comienzo, no es algo que puede ser añadido después (Humphrey 1997). Es imprescindible tener en cuenta tanto la obtención de la calidad como su control durante todas las etapas del ciclo de vida del software (Carrasco, León, y Benavides 2004). En la industria de software, el control de la calidad se encuentra íntimamente relacionado con los procesos de validación y verificación (Pressman 2002). En estos se perfilan las pruebas de software como actividad que permite asegurar que el producto realice correctamente todas sus funciones y que cumple correctamente con los requisitos del cliente.

Dependiendo de la estructura organizativa de la institución, las actividades relacionadas con las pruebas software se desarrollan siguiendo diferentes modelos. Entre ellos destacan los siguientes (Runeson, Andersson, y Thelin 2006):

- Desarrolladores que hacen las funciones de probadores.
- Equipo de pruebas integrado en el equipo de desarrollo.
- Equipos de pruebas independientes que realizan las labores de verificación y validación.
- Grupos de SQA que dedican parte de su esfuerzo a realizar pruebas.

Otros autores como T. Koomen hacen referencia a otra estructura organizativa conocida como *outsourcing* de las pruebas de software, o lo que es lo mismo, subcontratar las pruebas de software con el objetivo de reducir costes, tiempos y mejorar tanto el proceso de pruebas como la calidad del producto. Esta estructura se ha convertido en una tendencia actual (T. Koomen et al. 2006).

Como se ha explicado la calidad del software se ha convertido en un factor crítico, teniendo en cuenta un entorno cada vez más competitivo y clientes más exigentes. El desarrollo de software es una actividad que posee un impacto directo en todos los ámbitos de la sociedad moderna. La buena calidad de los desarrollos y la conciencia colectiva en el sector empresarial de la importancia de las áreas de pruebas, han contribuido a que el mercado internacional proponga nuevos estándares de aceptación. El mercado de las pruebas de software crecerá hasta 100.000 millones de euros en 2014, según la consultora francesa PAC (Cusmai 2011).

Dentro de la calidad, las pruebas de software como actividad fundamental del control de la calidad, juegan un papel protagónico. Como tendencia se ha generalizado la creación de entidades que se dediquen a probar software, bajo el concepto de la tercerización del servicio.

En las organizaciones que se dedican al servicio de pruebas de software se observa que para la consecución de sus objetivos, definen sus procesos, infraestructura, herramientas y técnicas (Capote García 2011). Partiendo del principio de que una organización será tan eficiente como lo son sus procesos, la industria de software, en especial el sector dedicado a las pruebas, no está ajena a las bondades y beneficios que aporta la gestión por procesos. Este enfoque le permite a la organización analizar sus limitaciones en función de mejorar la competitividad, a partir de reconocer la existencia de procesos internos, identificarlos y relacionarlos con los factores críticos para el éxito o que proporcione una ventaja competitiva. La gestión por procesos permite a las organizaciones de pruebas de software, enfocarse en las necesidades del cliente externo y orientar todo el esfuerzo hacia su satisfacción.

Internacionalmente, Cuba alcanza resultados muy discretos como país productor de software. No fue hasta la 10ma Feria Internacional Informática 2004 que se comienza a crear cierta imagen al mercado externo, a través de la presentación internacional de Incusoft (Industria Cubana del Software). Incusoft fue creada con el objetivo de aunar los esfuerzos individuales que han venido realizando diversas instituciones del país en este campo, para alcanzar una fortaleza que permita incursionar, con más efectividad, en los mercados extranjeros.

No existen evidencias de que en las empresas productoras de software en Cuba, se hayan aplicado procesos de certificación que permitan valorar el estado de la organización en cuanto a la madurez de sus procesos. Partiendo del alto potencial de personal calificado en el área de la informática, se considera que debe trabajarse para lograr organizaciones de software maduras, con personal comprometido, diestro y con

procesos planificados, ordenados y consistentes. Esto sin dudas, permitirá satisfacer el mercado interno y la demanda de informatización de la sociedad, además de aumentar la capacidad de adquisición de clientes provenientes del mercado internacional, a partir de la certificación formal de sus organizaciones.

La Universidad de las Ciencias Informáticas (UCI), acometió un proyecto de mejora de procesos basado en el Modelo de Integración de Madurez y Capacidades (CMMI por sus siglas en inglés), con la contratación de los servicios de consultoría del SIE Center del Tecnológico de Monterrey. Dicho proyecto estuvo encaminado a lograr el nivel 2 del modelo CMMI en tres centros de desarrollo, para convertirse así, en la primera organización cubana en lograr una certificación de este tipo.

CMMI, como modelo de referencia para el crecimiento de las capacidades y la madurez, se enfoca en los procesos de administración como de ingeniería de sistemas y de software. Con su implantación se prevé alcanzar beneficios tales como: (Carnegie Mellon Software Engineer Institute 2006)

- Calendarios y presupuestos predecibles en los proyectos.
- Mejora del ciclo de vida del desarrollo del software.
- Mayor productividad.
- Mayor calidad de los productos y servicios que ofrece la universidad a sus clientes en función de garantizar su satisfacción.
- Mejorar la moral del personal que labora en el centro.

El Centro Nacional de Calidad de Software (CALISOFT), adscrito al Ministerio de Comunicaciones (MINCOM), se constituye como organización enfocada a contribuir al desarrollo de la industria, proponiendo al Ministerio, normativas técnicas, políticas, procedimientos y estándares para las entidades que desarrollan aplicaciones informáticas. CALISOFT tiene como objeto social además, la responsabilidad de evaluar la conformidad con las normas y emitir certificación a productos informáticos nacionales o importados; brindar servicios de consultorías asociadas a la ingeniería y calidad de aplicaciones informáticas; controlar y auditar el uso de normativas técnicas, procedimientos, documentos estandarizados y buenas prácticas para el desarrollo de aplicaciones y sistemas informáticos en el país y brindar servicios de formación en los temas de calidad e ingeniería de aplicaciones informáticas. (Ministerio de Economía y Planificación 2012)

Dentro de este centro se encuentra un laboratorio de pruebas, el cual ha cambiado su concepción hasta convertirse en el Laboratorio Industrial de Pruebas de Software

(LIPS). Junto al Grupo de Ingeniería de Pruebas de Software, el LIPS forma parte del Departamento de Evaluación de Productos de Software, donde un conjunto de especialistas dirigen y coordinan los procesos de pruebas de liberación y evaluación de productos de software.

El LIPS se organiza a partir del análisis realizado de los principales problemas que se iban presentando con la organización anterior del laboratorio. Partiendo de un grupo artesanal de pruebas de software, se consideró que el proceso de pruebas de liberación debía ser como un proceso industrial donde se tuvieran elementos de entrada a un grupo de actividades consecutivas que fueran ejecutadas por los diferentes roles que intervienen en el proceso, y como salida se tendría un artefacto liberado. De esta manera, es un proceso que puede ejecutarse constantemente, pudiendo utilizar los recursos de manera óptima y no condicionando un grupo de probadores a una prueba específica como sucedía antes. Se contaba así con una fuerza permanente para las pruebas y se asignaba el trabajo en función de los cronogramas de los procesos de liberación que se estaban ejecutando (Capote García 2011).

A pesar de las definiciones y la madurez alcanzadas, a partir del propósito del LIPS y de los años de experiencia en el trabajo y con la ampliación de los servicios a toda la industria nacional de software, persisten un conjunto de problemas que atentan contra la eficiencia en la prestación de servicios de pruebas de liberación de productos de software. Un sondeo realizado a más del 50% de los miembros de la organización, reveló que se utiliza mucha documentación que atenta contra la fluidez del proceso. Algunos especialistas coinciden en que existen muchas actividades colaterales, planificadas o no, que interrumpen la ejecución de las pruebas de liberación que se estén desarrollando, que en muchos casos, son más de un proceso los que atiende un especialista. Se refiere además como dificultad, el incumplimiento de los cronogramas, los cuales son irreales en muchos casos al ser afectados frecuentemente por factores externos y otros riesgos que no son debidamente gestionados. Existe también el criterio de que debido a la especialización en los diferentes tipos de prueba, es evidente la dependencia entre los especialistas para el desarrollo de las actividades de pruebas. Otra dificultad detectada, es que los equipos de desarrollo de software y los proyectos no conocen el proceso de pruebas de liberación, ni los criterios de criticidad donde se definen los casos en los que se detiene o aborta un proceso. Algunos señalan, que existe falta de experiencia y poco conocimiento en algunos casos, del trabajo a desarrollar.

Con el objetivo de conocer el criterio de los especialistas del LIPS sobre los procesos que se desarrollan para una satisfactoria ejecución de las pruebas de liberación, se detectó que un porcentaje significativo no conoce si están identificados todos los procesos, o si lo conocen, consideran que se pueden introducir mejoras. Algunos señalan directamente que no están descritos todos los procesos que soportan la ejecución de las pruebas y que, o bien se encuentran desactualizados, o no se tiene conocimiento de dónde consultarlos. Al requerir si los especialistas saben qué hacer en cada momento durante una prueba de liberación, cerca de la mitad de los especialistas consideró que sí, pero daban oportunidad a que podía trabajarse para aumentar el conocimiento al respecto, como sucedió también con el conocimiento sobre dónde consultar los procesos del laboratorio ante una duda de algún especialista. Esto denota que las personas de alguna manera conocen cómo actuar en cada momento, pero no se dispone de los procesos formalmente escritos y establecidos. La gran mayoría no conoce si se controla la ejecución de los procesos, si se realizan mediciones de los mismos y si existe alguna evidencia de que se gestionen o registren las variaciones acaecidas anteriormente. Los especialistas no tienen conocimiento de la posibilidad de modificar determinado proceso.

Todo lo anteriormente expuesto constituyen elementos que acentúan la deficiente gestión de los procesos en el LIPS, que atenta con el cumplimiento de los objetivos estratégicos de la organización y que lleva a plantear como **problema de investigación**, ¿cómo contribuir a la gestión del proceso de pruebas en el Laboratorio Industrial de Pruebas de Software, para elevar la eficiencia del servicio de evaluación de productos de software?

Para resolver esta interrogante, se parte del análisis del **objeto de estudio**: la gestión de procesos de pruebas. Como **campo de acción** se plantea: el servicio de evaluación de productos de software, y se define como **objetivo**, diseñar una estrategia que permita elevar la eficiencia del servicio de evaluación de productos de software en el Laboratorio Industrial de Pruebas de Software.

Entre los **objetivos específicos** a dar cumplimiento, se relacionan:

1. Definir el marco teórico referencial de las pruebas de software, los procesos de pruebas en las organizaciones y la gestión de los procesos para sentar las bases del desarrollo de la investigación.
2. Realizar un diagnóstico sobre el estado de los procesos internos del Laboratorio Industrial de Pruebas de Software, para profundizar en las principales dificultades.

3. Diseñar una estrategia para elevar la eficiencia del servicio de evaluación de productos de software en el Laboratorio Industrial de Pruebas de Software.
4. Validar la estrategia para elevar la eficiencia del servicio de evaluación de productos de software en el Laboratorio Industrial de Pruebas de Software.

Para apoyar la investigación se plantea como **hipótesis** que, con el diseño de una estrategia se logrará gestionar las pruebas para elevar la eficiencia del servicio de evaluación de productos de software. De aquí se identifica como **variable independiente** la estrategia y como **variables dependientes**, la gestión de las pruebas y la eficiencia del servicio de evaluación de productos.

En función del desarrollo de la investigación, se definieron como métodos de trabajo científico los siguientes:

Métodos teóricos.

- *Histórico-Lógico*, para determinar las necesidades históricas y tendencias actuales de la gestión de procesos.
- *Analítico-Sintético*, para profundizar en la esencia de la gestión de procesos, partiendo de lo que plantea la teoría al respecto y del análisis del problema, para luego sintetizar todos sus componentes en una estrategia para la gestión de los procesos internos del laboratorio.
- *Inductivo-deductivo*, para arribar a conclusiones sobre el problema de la investigación, a partir de la generalización y especificación de los resultados parciales que se obtengan.

Métodos empíricos.

- *Encuestas*, para recolectar información que permita, luego de su procesamiento y análisis, identificar la esencia del problema y proponer la solución, así como evaluar el impacto de la utilización de la estrategia.
- *Observación participante*, para obtener información del seno de la organización, a partir de las discusiones científicas y de los encuentros que se realicen en entornos que aporten información para la investigación.
- *Experimentación*, para la aplicación del proceso de evaluación de productos propuesto y evaluar el impacto sobre el objeto de estudio.

Resultados a alcanzar:

Como resultado de la presente investigación, se obtendrá una estrategia para elevar la eficiencia del servicio de evaluación de productos de software en el Laboratorio Industrial de Pruebas de Software. Para este fin, se obtendrá la identificación y el

establecimiento de las actividades que se desarrollan en el laboratorio, en la prestación de servicios de pruebas de software. Se alcanzará además, la definición del proceso que soporte dicha estrategia, que brindarán el qué hacer ante la necesidad de mejora del proceso existente, así como las actividades de monitoreo y control que se precisan para su establecimiento.

La implantación de esta estrategia, permitirá aumentar la eficiencia en la prestación de servicios en el laboratorio, mejorando el uso de recursos y el tiempo de ejecución de las pruebas.

Estructura del documento:

El documento se encuentra estructurado en tres capítulos. En el Capítulo 1 se realiza un estudio de los principales referentes teóricos de los procesos, los componentes de los procesos, profundizando en los procesos de pruebas de software. Se realiza una caracterización de las actividades del Laboratorio Industrial de Pruebas de Software, partiendo de la identificación y comportamiento de los elementos que ejercen influencia en el objeto de estudio. En el Capítulo 2, se presenta la estrategia para elevar la eficiencia del servicio de evaluación de productos, profundizando en los elementos novedosos y esenciales que se le incorporan.

En el Capítulo 3 se exponen los resultados de la validación de la estrategia, a partir del análisis del comportamiento de las variables planteadas en la hipótesis: la gestión de las pruebas y la eficiencia del servicio de evaluación de productos en el Laboratorio Industrial de Pruebas de Software. Por último se presentan las Conclusiones y Recomendaciones, resultados del desarrollo de la investigación y se exponen la Bibliografía y los Anexos, para apoyar la comprensión y complementar el trabajo realizado.

CAPÍTULO I. FUNDAMENTACIÓN TEÓRICA

1.1 INTRODUCCIÓN

En el presente capítulo se desarrollan los elementos conceptuales que sustentan la investigación. Estos están relacionados esencialmente con la definición del término Proceso y sus principales componentes. Se precisan los elementos claves de procesos de pruebas de software, analizando las estructuras organizativas para el desarrollo de pruebas o actividades de verificación y validación de software. Se relacionan los principales modelos de referencia para identificar los elementos que puedan tributar al cumplimiento del objetivo. Por último, se realiza un estudio para caracterizar el entorno que rodea la investigación, abordando los principios generales del funcionamiento del LIPS.

1.2 LA DEFINICIÓN DE PROCESO. SU APORTE A LA GESTIÓN DE PRUEBAS DE SOFTWARE

La mejor manera de aprovechar y controlar todos los recursos disponibles en cualquier organización, parte de realizar un análisis profundo sobre todos los procesos que influyen en el resultado del producto o servicio. De esta manera, se logra una mejor comprensión que facilitará la toma de decisiones necesaria, para economizar esfuerzos y energía.

El término “proceso” proviene del latín *processus*, que significa avance o progreso. Según la Real Academia de la Lengua Española, proceso se define como acción de ir hacia adelante, transcurso del tiempo, conjunto de la fases sucesivas de un fenómeno natural o de una operación artificial. («Diccionario de la lengua española» 2014)

El modelo de la Fundación Europea para la Gestión de la Calidad (EFQM por sus siglas en inglés), define un proceso como la “secuencia de actividades que van añadiendo valor mientras se produce un determinado producto o servicio, a partir de determinadas aportaciones” (Sánchez y Castro 2005). La ISO/IEC 9001-2008 define proceso como actividad que utiliza recursos, y que se gestiona con el fin de permitir que los elementos de entrada se transformen en resultados. Frecuentemente, el resultado de un proceso constituye directamente el elemento de entrada del siguiente proceso. (NC-ISO/IEC 9001: 2008).

Boned Pascual, reconoce como proceso la sucesión sistemática de acciones interrelacionadas, que transforman una serie de elementos de entrada en requerimientos o necesidades; que posee un determinado destinatario o usuario (interno o externo) con el fin de conseguir o alcanzar los objetivos establecidos,

añadiendo valor en cada clase (Boned Pascual 2007). Boned, a diferencia de las definiciones anteriores, toma en cuenta la finalidad de los procesos y su visión muestra un giro conceptual hacia los servicios, además de las delimitaciones de objetivos concretos que estarían formando parte del propio proceso, y de la definición de usuario interno o externo.

Siguiendo esta misma línea, otras fuentes consultadas por el autor, definen proceso como el conjunto de actividades orientadas a generar un valor agregado sobre los insumos o entradas para conseguir un resultado, conocido como salida, a través de una operación de transformación y que a su vez satisfaga las necesidades y expectativas del cliente (Ruiz Beltrán y Suárez Maya 2011). Un proceso, es el conjunto de actividades de trabajo interrelacionadas que se caracterizan por requerir ciertos insumos y tareas particulares que implican valor añadido, con mira a obtener ciertos resultados. Freddy Fabián y Renato David lo definen también como la gestión de todas las actividades de la empresa que generan un valor añadido; o bien, conjunto de actividades mutuamente relacionadas o que interactúan, las cuales transforman elementos de entrada en resultados. (Fuentes Moreno y Paz y Miño Espinoza 2011)

Todos los autores coinciden en que los procesos constituyen actividades ordenadas y gestionadas, en la que se transforman un conjunto de entradas en otro grupo de salidas con un valor agregado, que responde a la satisfacción del cliente. En un artículo publicado en la revista científica EÍDOS, se define proceso como “Secuencia ordenada de actividades repetitivas que se realizan en la organización por una persona, grupo o departamento, con la capacidad de transformar unas entradas en (Inputs) en salidas o resultados (ouputs) para un destinatario (dentro o fuera de la empresa que lo ha solicitado y que son los clientes de cada proceso) ejecutado de una manera eficaz y eficiente para obtener un valor agregado. Los procesos generalmente, cruzan repetidamente las barreras funcionales, fuerzan a la cooperación y crean una cultura de la empresa distinta (más abierta, menos jerárquica, más orientada a obtener resultados que a mantener privilegios), están centrados en las expectativas de los clientes, las metas de la organización, son dinámicos, variables y el punto de concreción de los indicadores diseñados para el control” (Alberto Medina León et al. 2010).

El principal aporte de dicha definición, está en añadirle el componente cultural y de cooperación necesario, para lograr la misión de la organización. En su conceptualización transgrede el enfoque positivista de procesos, en la aspiración de integrar en su concepción, control, metas, expectativas y resultados.

El concepto asumido por el autor de a presente investigación define al proceso como la consecución de actividades que convierten entradas en salidas, aportándoles un valor agregado. En la Figura 1 se representa dicho concepto, asumido a los efectos de la investigación y a partir de los elementos citados anteriormente.



Figura 1. Representación gráfica del concepto Proceso.
Fuente: Elaboración Propia.

En todo proceso se distingue una serie de componentes fundamentales. La definición de cada uno de estos contribuye al entendimiento del proceso, así como a la gestión de su desarrollo, en función de obtener resultados con un valor agregado de alto impacto en la satisfacción del cliente.

1.2.1 COMPONENTES DE LOS PROCESOS

Los procesos tienen elementos componentes, aunque se da el caso de que existen algunos en los que estos elementos no han sido identificados correctamente. Sangüesa refiere como componentes de los procesos, el elemento procesador, la secuencia de actividades, las entradas, las salidas, los recursos, los clientes del proceso, las expectativas del cliente del proceso con relación al flujo de salida, el indicador y el responsable del proceso (Sangüesa 2001).

En otra descripción de los elementos de los procesos, se establecen como componentes las Entradas o Insumos, Recursos y estructuras, Producto, Transformación, Sistema de Medidas de Control del Funcionamiento y los Límites (Acuña Bermeo 2008). En esta oportunidad se definen como componentes, las condiciones de frontera de un proceso, marcadas por la referencia a dónde inicia y dónde termina el proceso, pasando por los proveedores, las entradas, las salidas y el cliente.

En el artículo “La gestión por procesos: un enfoque de gestión eficiente”, Mallar establece como componentes de los procesos las Entradas (*Inputs*), los recursos o factores que se transforman, el flujo real de procesamiento o transformación y las Salidas (*Outputs*) (Mallar 2010), sintetizando las definiciones dadas hasta el momento.

Otros autores, como es el caso de Fuentez Marquez, relacionan como componentes las Entradas, los Subprocesos, actividades y operaciones, Salidas, resultados o productos y los Clientes (Fuentez Marquez et al. 2010). Según Pérez Fernández de Velasco, las entradas de un proceso constituyen un producto que proviene de un suministrador (externo o interno); es la salida de otro proceso (precedente en la cadena de valor) o de un proceso del proveedor o del cliente. La existencia de una entrada a un proceso es lo que justifica su ejecución sistemática. (Pérez Fernández de Velasco 2007)

Las entradas de un proceso pueden clasificarse como Recursos o Insumos. Los Recursos proporcionan las facilidades para desarrollar las operaciones o tareas del proceso. Pueden ser tangibles (materiales) o intangibles (no materiales). Los insumos por su parte, constituyen todo bien material que va a ser procesado (ensamblado o transformado). Los datos sin procesar, contenidos en medios magnéticos o de forma electromagnética, constituyen insumos. Las entradas, tanto recursos como insumos, pueden ser iniciales o intermedias. Serán iniciales si es que se incorporan con el inicio del proceso. Serán intermedias si es que se van incorporando con el desarrollo del proceso.

Pérez Fernández plantea además, que las operaciones son subprocesos, actividades, tareas, procesos de menor jerarquía, pues, de manera individual o colectiva, también hacen uso de los recursos, transformándolos o agregándoles valor dentro del sistema de gestión particular.

Las operaciones constituyen una secuencia ordenada de actividades que se realizan dentro de un proceso (Sangüesa 2001). La realización de una operación, subproceso, actividad, constituye una acción que está produciéndose y siendo consumida de manera simultánea para lograr al término, un servicio asociado a un bien tangible o intangible.

Las salidas, por su parte, son resultados o productos que genera el proceso y que pueden constituir entradas de un siguiente proceso cuando el cliente es interno, o constituir el producto final cuando el cliente es externo. Según Mallar, las salidas son básicamente de dos tipos: Bienes y Servicios. Los primeros pueden ser tangibles, almacenables, transportables. La producción se puede diferenciar de su consumo. Es posible además una evaluación de su grado de calidad de forma objetiva y referida al producto. Los Servicios por su parte, son intangibles, constituyen una acción sobre el cliente. La producción y el consumo son simultáneos y su calidad depende básicamente de la percepción del cliente (Mallar 2010).

Los resultados o salidas de un proceso se dirigen a las personas, áreas o procesos, clientes o usuarios. El término cliente denota a quien se atiende una o más de una vez. El término usuario denota a quien usa o se beneficia del servicio o bien que resulta del proceso. Dependiendo de su aparición durante el proceso y de cómo se ha definido su alcance, los clientes o usuarios pueden ser internos o externos. Son internos si forman parte del sistema de gestión del proceso y externos si no forman parte de este sistema. (Fuentez Marquez et al. 2010)

Las actividades, operaciones o tareas dentro de todo proceso, requieren contar con criterios, instrucciones e instrumentos para detectar probables irregularidades y medir el desempeño del proceso en sus puntos críticos. Se precisa además, controlar, corregir o suprimir dichas irregularidades y evaluar el desarrollo del proceso a partir de las implicaciones de los cambios introducidos.

El monitoreo permite estar atento al desarrollo del proceso, del producto o servicio, y saber cómo está percibiéndolo el cliente. Para esto, se requieren instrumentos que permitan medir los avances, desarrollos o evoluciones y de la toma de acciones correctivas pertinentes.

Un proceso, en resumen, se inicia con un ingreso de insumos (materiales, financieros, personal o información), los que a través de una serie de actividades de transformación agregan (o no) valor y culmina con uno o varios productos (bienes y/o servicios) orientados a la satisfacción de las necesidades de un cliente (sea interno o externo). Es necesario también identificar a su(s) responsable(s) y los criterios (indicadores) para evaluar su rendimiento. (Medina-Giopp 2003)

Como resultado del análisis conceptual de los diferentes autores, se puede decir que muchos coinciden en señalar que los principales componentes de un proceso son los siguientes:

- Entradas.
- Operaciones.
- Salidas.
- Clientes (internos, externos).
- Sistema de monitoreo, control y evaluación.

Para el desarrollo de la presente investigación, el autor asume como componentes de los procesos, los relacionados anteriormente, a partir de las consideraciones de cada uno de los autores consultados. Una vez que ya ha sido identificado qué es un proceso y sus principales componentes, corresponde evaluar cómo se puede

asociar la definición de proceso a las actividades de verificación y validación de productos de software.

1.3 PROCESOS DE PRUEBAS DE SOFTWARE

Según Zamora Hernández, los procesos de pruebas de software destacan como el proceso de establecer datos de entrada al producto software e inspeccionar los resultados obtenidos. Fundamentalmente, las pruebas del software responden a dos preguntas, ¿se ha obtenido un buen producto?, ¿se ha desarrollado de forma correcta? Lo cual asegura, que las pruebas software cubren el ciclo completo de vida del producto, desde su desarrollo hasta su mantenimiento. Este concepto da lugar al proceso de verificación y validación del software (Zamora Hernández 2011).

El proceso de pruebas de software constituye uno de los procesos más relevantes dentro del ciclo de vida de desarrollo de software, ya que estas permiten incrementar la calidad de los productos de un modo relativamente rápido. La verificación y validación es un tema crítico para el desarrollo de productos modernos de alta tecnología (Belt 2009).

Por mucho tiempo, probablemente la prueba haya sido la etapa menos comprendida del ciclo de vida del desarrollo de software, e inclusive, a la que menos tiempo se dedica en los proyectos. Esto se puede deber a la falta de conocimiento, de experiencia o de presupuesto necesarios para aplicarla. (Serna y Serna 2014)

Según Sanz Esteban y García, de Amescua y Velasco, numerosos estudios han demostrado la gran cantidad de beneficios que se obtienen realizando actividades de verificación y validación de productos software, mediante la utilización de procesos formales. Disponer de un proceso de pruebas de software correctamente definido contribuye a incrementar la calidad de los productos (Sanz Esteban 2012) (García, De Amescua, y Velasco 2006). Los beneficios más destacables son los siguientes:

- Un incremento de la satisfacción del cliente debida al decremento de la tasa de fallos en el producto entregado,
- un incremento de la eficiencia del proceso de desarrollo,
- se facilita la definición y cumplimiento de los objetivos de calidad,
- se incrementa la satisfacción de los trabajadores debido a que se proporcionan herramientas y recursos apropiados para la realización eficiente del trabajo,
- y se incrementa el número de defectos encontrados con la consecuente reducción del tiempo y el coste de retrabajo, entre otros.

Existen muchos procesos que prescriben la manera de generar y aplicar las pruebas. En el libro de Utting y Legeard, se describen los más usados en la industria, entre los cuales se encuentran: manual, de grabación, basado en scripts, automatizado dirigido por datos y basado en modelos.

- Proceso de pruebas manual: Es el estilo más antiguo para probar una aplicación, y aún se usa ampliamente. Tanto el diseño del caso de prueba como su ejecución se hacen de manera manual, y los detalles de bajo nivel respecto de la interacción con el sistema bajo prueba, se dejan a cargo del “probador-ejecutor”. Es sencillo y económico de iniciar, de aplicación flexible, pero muy costoso en la ejecución, con la evolución del código las pruebas de regresión son muy limitadas y se dificulta medir la cobertura.
- Proceso de pruebas de grabación: Consiste en grabar la interacción con el sistema bajo prueba, para luego usarla como plantilla que se puede ejecutar varias veces con diferentes valores, o adaptarla según el objetivo de la prueba. La grabación se hace por medio de herramientas especializadas. Es flexible en la aplicación, la primera aplicación es costosa, permite pruebas automáticas de regresión, no son muy exigentes y se dificulta medir la cobertura. Para su ejecución es necesario que el sistema bajo prueba tenga una funcionalidad mínima.
- Proceso de pruebas basado en secuencias de comandos (“scripts”): Se basa en un fichero que contiene un conjunto de comandos que ejecutan uno o más casos de prueba. Requiere programar la prueba, ejecuta la prueba de manera automática, facilita las pruebas de regresión automáticas, el nivel de exigencia de la prueba depende del nivel de abstracción.
- Proceso automatizado de pruebas dirigido por datos claves: Consiste en escribir los ficheros de comandos de manera genérica y luego durante la ejecución usar los datos de prueba para particularizar la prueba. Para ello el caso de prueba se define lo más abstracto posible y luego se adapta a los objetivos de la prueba durante su ejecución. La ejecución automática permite aplicar pruebas de regresión, no es un proceso para pruebas robustas.
- Proceso de pruebas basado en modelos: En este proceso se usan herramientas especializadas para generar, a partir de los modelos arquitectónico del sistema bajo prueba, los casos de prueba. Así el diseño de las pruebas es automático, permite la ejecución automática de pruebas de regresión, cobertura sistemática, medidas de cobertura del modelo y de los requisitos; como desventajas destacan que causa sobrecarga de modelos, no

hay suficientes herramientas y no se dispone de modelos para todo el código.
(Utting y Legeard 2010)

Muchas son las iniciativas que se encuentran en la literatura sobre procesos de pruebas de software. Los esfuerzos son tangibles desde los proyectos de desarrollo de software que apuestan por el aseguramiento de la calidad de sus producciones (Idrobo Burbano y Jojoa López 2012) (Valderrama 2014) hasta la adopción de modelos, normas y estándares de calidad y procesos de mejora. (Gutiérrez et al. 2006) Las soluciones no solo se concentran en la industria sino también se ubican en la formación desde la academia. (Astigarraga et al. 2010) (Anaya 2012)

Para incrementar la calidad de los productos de software, las organizaciones han de tratar de llevar a la práctica iniciativas de mejoras de procesos en el área de pruebas de software. No obstante, esto supone un gran esfuerzo así como inversiones considerables en personal, tiempo y dinero. (Ngwenyama y Nørbjerg 2010). Una actividad clave para poner en marcha estas iniciativas, es la definición del proceso. Sin embargo, se puede decir que la definición de procesos está en un nivel poco maduro. Por consiguiente, la tarea de definir procesos es difícil y costosa puesto que cada vez que se aborda la definición de un nuevo proceso se parte de cero. (Medina Domínguez 2010)

Las pruebas del software se diferencian del resto de actividades del proceso de desarrollo software en que éstas son un proceso destructivo. El objetivo de quien realiza las pruebas, es el de descubrir errores que no se han detectado antes. El trabajo de probador de software resulta demasiado especializado por lo que todo el mundo no está preparado para llevarlo a cabo, debido a que la naturaleza humana es mayormente constructiva, no destructiva, según plantea Zamora. (Zamora Hernández 2011).

Otro factor que resulta especialmente condicionante para ejecutar pruebas, es la dificultad de probar el código desarrollado por uno mismo. La persona que ha desarrollado un código establece un punto de partida desde el que considera que todo funciona de forma correcta. Resulta difícil pensar que lo que ha hecho uno mismo está mal. Esta es la razón por la cual los desarrolladores y los probadores normalmente se encuentran distribuidos en grupos distintos y por lo que también no suelen llegar a tener un mutuo entendimiento. Es necesario comprender que los probadores se encuentran añadiendo un valor al producto, descubriendo errores e informando de ellos tan pronto como sea posible. Su intención no es la de perjudicar la producción. (Zamora Hernández 2011)

Si se pretende sintetizar todo lo teóricamente desarrollado sobre los procesos de pruebas de software, también conocidos como procesos de verificación y validación, es preciso identificar cuáles son las principales estructuras organizativas en que se centra esta actividad. Estas condicionan la implementación de procesos de pruebas de software, por lo que es necesario conocerlas para determinar qué modelo de referencia se ajusta mejor a cada una de estas estructuras organizativas.

1.4 ESTRUCTURAS ORGANIZATIVAS PARA EL DESARROLLO DE PRUEBAS DE SOFTWARE

Existen muchas formas de organizar el desarrollo de las pruebas de software, desde estructuras sencillas en las que es el programador el que las realiza, hasta organizaciones que proporcionan servicios orientados al desarrollo de las actividades de verificación y validación de productos software. La decisión sobre la estructura que mejor se adapta a cada organización depende de la política y cultura de la misma, así como del conocimiento y habilidades de los recursos humanos implicados en el proceso de pruebas software con que cuenten (Sanz Esteban 2012a).

Varios autores (Burnstein, Suwanassart, y Carlson 1996) (Craig y Jaskiel 2002) (Gonzalez, Gasco, y Llopis 2005) (Sanz Esteban 2012) señalan que las principales estructuras organizativas son:

- Los desarrolladores son probadores.
- Equipos de pruebas integrados.
- Equipos de pruebas independientes.
- Grupos de aseguramiento de la calidad.
- Factoría de pruebas.

A continuación se realiza un análisis de dichas estructuras, destacando las ventajas y desventajas de cada una de ellas.

1.4.1 LOS DESARROLLADORES SON PROBADORES

En la estructura organizativa en que los desarrolladores son probadores, para el desarrollo de pruebas de software, ambos juegan el mismo papel. La principal ventaja de esta estructura es que los desarrolladores comprenden completamente el producto que están probando y tienen pleno conocimiento del diseño y código. Sin embargo, ésta constituye también uno de los principales inconvenientes, este conocimiento provoca que, de forma inconsciente, las pruebas se diseñen para ser pasadas sin dar lugar a errores y no para detectar defectos (Craig y Jaskiel 2002). En el Anexo 1 se muestra un esquema donde se representa esta estructura organizativa.

1.4.2 EQUIPOS DE PRUEBAS INTEGRADOS

Este tipo de estructura organizativa está conformado por desarrolladores y probadores unidos a través del jefe de proyecto. La ventaja principal radica en que al estar integrados en el equipo de desarrollo, la comunicación entre ambos es más fluida y las relaciones menos tensas, comparten recursos y trabajan como un equipo (Anexo 2).

Aunque conseguir que trabajen como un único equipo es una tarea muy compleja que no siempre se desarrolla con éxito, el principal inconveniente se produce cuando el equipo comienza a estar bajo presión, la actitud del jefe de proyecto habitualmente apoya a terminar el producto sacrificando la calidad del mismo, es decir, disminuyendo las actividades de pruebas (Craig y Jaskiel 2002).

1.4.3 EQUIPOS DE PRUEBAS INDEPENDIENTES

Un equipo independiente de pruebas es un equipo cuya función principal es la prueba. Pueden ejecutar pruebas a un solo producto o a muchos. Los equipos de pruebas independientes han existido durante mucho tiempo, pero realmente ganaron popularidad a partir de principios de 1980.

La principal ventaja es que los integrantes del equipo poseen los conocimientos y habilidades necesarias para desempeñar sus tareas correctamente (Burnstein, Suwanassart, y Carlson 1996), es decir, el equipo se compondrá por personal experimentado en el área de pruebas. Sin embargo, la creación de este equipo frecuentemente constituirá la creación de un muro entre desarrolladores y probadores, lo que dificulta en gran medida la comunicación necesaria entre ambos equipos (Craig y Jaskiel 2002).

Un elemento importante que caracteriza esta estructura organizativa según Sanz Esteban, es que el equipo de pruebas carece de conocimiento sobre el producto, por lo que necesita que éste sea transferido por el equipo de desarrollo. Si la comunicación entre ambos equipos no es fluida éste constituirá un problema muy importante (Sanz Esteban 2012) (Anexo 3).

1.4.4 GRUPOS DE ASEGURAMIENTO DE LA CALIDAD

En el tipo de estructura organizacional de grupos de aseguramiento de la calidad, las organizaciones asignan al personal de aseguramiento de la calidad las funciones de verificación y validación. Esto se debe a que las habilidades de los integrantes del grupo de aseguramiento de la calidad, son similares a las habilidades que poseen los probadores. La principal ventaja según Craig y Jaskiel, es que estos grupos trabajan en la organización y tienen habilidades para desempeñar las actividades de pruebas

software. Hoy existen muchos grupos que se autodenominan de aseguramiento de la calidad, pero no realizan ninguna función de aseguramiento tradicional, solo realizan pruebas de software. El principal inconveniente es que el grupo de aseguramiento de la calidad no sólo realiza pruebas, sino que tiene otras tareas que desempeñar. Estas responsabilidades adicionales provocan que no puedan realizar su trabajo de forma efectiva (Craig y Jaskiel 2002) (Anexo 4).

1.4.5 FACTORÍA DE PRUEBAS

Las factorías de pruebas basan su funcionamiento en el concepto de outsourcing abordado por McCarthy y Anagnostou (McCarthy y Anagnostou 2004). Consisten en asignar o subcontratar todas o algunas de las actividades de pruebas a otra organización (Anexo 5). El outsourcing ha recibido mucha visibilidad en los últimos años y es una forma de obtener ayuda rápidamente. La clave para el outsourcing de las pruebas es tener un buen contrato, tener entregables bien definidos y normas de calidad, y mantener una buena supervisión de los resultados de las pruebas (Craig y Jaskiel 2002).

La principal ventaja de este enfoque, es que las organizaciones especializadas disponen del personal experimentado y la infraestructura necesaria para llevar a cabo de una forma controlada y adecuada, las diferentes actividades relacionadas con las pruebas software. Los inconvenientes se basan en el desconocimiento de la visión funcional de los aspectos de negocio de la organización contratante, según plantea (Sanz Esteban 2012).

Diferentes autores, según plantea Sanz Esteban, soportan que el outsourcing de las pruebas software constituyen una buena forma de probar los productos con un elevado nivel de calidad (Sanz Esteban 2012). Sin embargo, esta estructura no está libre de inconvenientes, aunque ya se ha citado alguno, la principal desventaja es la necesidad de realizar una buena gestión de las actividades externalizadas (Craig y Jaskiel 2002), esta constituirá el factor clave para el éxito del outsourcing de pruebas software. Además, en este tipo de entornos aparece también otro riesgo importante que es la excesiva dependencia en el proveedor, que puede llevar a la pérdida de control sobre las actividades de prueba o la pérdida de “*know-how*” sobre la ejecución de dichas actividades (Gonzalez, Gasco, y Llopis 2005).

Todas estas estructuras organizativas para la realización de las actividades de pruebas poseen un conjunto de ventajas y desventajas. Son empleadas en función de las necesidades de la organización. En la Tabla 1 se resume estos aspectos positivos y negativos que acompañan a las estructuras relacionadas.

Tabla 1. Ventajas y desventajas de las estructuras organizativas para las pruebas de software.

Fuente: Elaboración propia.

Estructura organizativas	Aspectos positivos	Aspectos negativos
Desarrolladores como probadores	<ul style="list-style-type: none"> - Expertos en software. - No hay conflicto con los probadores. 	<ul style="list-style-type: none"> - Carencia de una perspectiva fresca, diferente. Falta de la visión y el conocimiento desde el negocio. - Pueden carecer de habilidades en pruebas de software. - Existencia de presión para entregar, se centran en el código. - Requieren procedimientos rigurosos y disciplina.
Equipos de prueba integrados	<ul style="list-style-type: none"> - Trabajo en equipo. - Intercambio de recursos. - Facilita el comienzo temprano. 	<ul style="list-style-type: none"> - Presión por la entrega, en detrimento de la calidad.
Equipos de pruebas independientes	<ul style="list-style-type: none"> - Probadores profesionales con puntos de vista frescos. 	<ul style="list-style-type: none"> - Potencial conflicto entre los desarrolladores y probadores. - Difícil de comenzar las pruebas con tiempo suficiente.
Grupos de aseguramiento de la calidad	<ul style="list-style-type: none"> - Organización existente. - Una cierta habilidad existente. - Infraestructura. 	<ul style="list-style-type: none"> - Más de qué preocuparse además de las pruebas.
Factorías de pruebas	<ul style="list-style-type: none"> - Probadores profesionales. - No necesitan contratar o retener al personal. 	<ul style="list-style-type: none"> - Aún requiere la gestión de subcontratistas, tienen que tener un buen contrato.

Existen diferentes modelos de referencia, con el propósito de mejorar los procesos de pruebas de software y por tanto, la calidad de los productos. La aplicabilidad de dichos modelos está condicionada por la estructura organizativa que implemente la organización. En el siguiente epígrafe se relacionan los modelos estudiados por el autor de la presente investigación, que le permiten conocer los elementos positivos que se puedan emplear para el diseño de la estrategia, destacando aquellos modelos que son aplicables a la estructura organizativa de factorías de pruebas.

1.5 MODELOS RELACIONADOS CON LA GESTIÓN DE PRUEBAS

Los modelos definen el conjunto de buenas prácticas a realizar para llevar a cabo una mejora de procesos. Estos modelos proporcionan el marco de referencia necesario para determinar las fortalezas y debilidades de los procesos implementados en una organización, y determinar a partir de las mismas, las acciones de mejora a realizar.

La revisión bibliográfica arrojó que existen modelos enfocados al desarrollo de actividades aplicadas tanto a productos como a servicios. Existen otros dirigidos específicamente al proceso de pruebas de software. En los primeros se distinguen áreas destinadas a actividades de validación y verificación, como se analizará en los siguientes epígrafes.

1.5.1 CMMI FOR DEVELOPMENT

CMMI es un modelo de referencia que cubre el desarrollo y mantenimiento de las actividades aplicadas tanto a productos como a servicios. Surgió como resultado de la combinación de los tres modelos más importantes, cuyo uso independiente generaba un problema ya que incrementaban los esfuerzos, al implantarlos por disciplinas específicas. Estos fueron el modelo de capacidad y madurez para software (SW-CMM), modelo de capacidad para ingeniería de sistemas (SECM) y el modelo de capacidad y madurez para el desarrollo integrado de productos (IPD-CMM). Estos tres modelos fueron elegidos por su amplio reconocimiento y adopción en las comunidades de ingeniería de software y de sistemas, así como por sus diferentes enfoques para mejorar los procesos de una empresa (Parada Gélvez 2010) (Sanz Esteban 2012).

CMMI define un conjunto de buenas prácticas que cubren un conjunto de áreas de proceso involucradas en el desarrollo y mantenimiento, permitiendo establecer tanto el nivel de madurez como el nivel de capacidad existente en la organización. Permite dos representaciones diferentes: continua y por etapas. La primera define seis niveles de capacidad: Incompleto, Ejecutado, Gestionado, Definido, Cuantitativamente gestionado y Optimizado; que permiten conocer la capacidad de los procesos que sean seleccionados para su mejora. Se centra en un área de proceso y en adquirir un nivel de capacidad para el mismo. La segunda define cinco niveles de madurez: Inicial, Gestionado, Definido, Gestionado cuantitativamente y Optimizado, que permiten conocer la madurez de los procesos que se realizan en la organización; centrándose en la madurez de toda la organización.

Según Sanz Esteban, CMMI for Development no contiene el soporte adecuado para el proceso de pruebas. Esto se debe a que únicamente define 3 áreas de proceso relacionadas con el mismo: *Product Integration*, *Validation*, y *Verification* (Integración

del producto, Validación y Verificación); puesto que en esta propuesta no se detalla cómo deben de definirse e implementarse los procesos y omite muchas de las actividades a llevar a cabo, por lo que requiere del uso de modelos específicos relacionados con las pruebas de software (Sanz Esteban 2012).

1.5.2 TMM – TEST MATURITY MODEL

El modelo TMM-Test Maturity Model según Burnstein, ha sido desarrollado por el Instituto Tecnológico de Illinois como una guía para la mejora del proceso de pruebas. Al igual que CMM, modelo al que complementa, utiliza el concepto de niveles de madurez para la evaluación y mejora del proceso de pruebas; aunque a diferencia de su antecesor no contempla los niveles de capacidad (Burnstein 2003).

TMM consta de 5 niveles de madurez que reflejan el grado de madurez de la organización en el proceso de pruebas, es decir, representan la evolución hacia un proceso de pruebas maduro. Cada nivel tiene definidas un conjunto de áreas de proceso, y cada una de ellas contiene las actividades relacionadas con la misma que es necesario realizar para contribuir a la mejora del proceso.

TMM podría ser utilizado en organizaciones que utilizan una estructura organizativa de pruebas basada en equipos independientes o grupos de aseguramiento de la calidad. En el caso de que los desarrolladores sean los probadores o equipos integrados de pruebas este modelo de referencia resulta demasiado complejo y burocrático como para ser utilizado; tampoco es apropiado para una factoría de pruebas ya que no contempla procesos de gestión de servicios. En el análisis realizado por Sanz Esteban evidencia que TMM no profundiza en la propia definición de los procesos y actividades a llevar a cabo para la validación y verificación de productos software, por lo que es la propia organización la que se tiene que ocupar de la definición de los mismos (Sanz Esteban 2012).

1.5.3 TMMi – TEST MATURITY MODEL INTEGRATION

El modelo TMMi – Test Maturity Model Integration según Van Veendaal y Cannegieter, ha sido desarrollado por la TMMi Foundation como una guía y un marco de referencia para la mejora del proceso de pruebas, siendo un modelo complementario a CMMI. Al igual que este último, TMMi define una representación por etapas y hace uso del concepto de niveles de madurez para evaluar y mejorar las diferentes áreas de proceso de cada uno de los niveles. Además de las áreas de proceso, identifica el conjunto de objetivos a alcanzar y las prácticas que deben realizarse para ello. El objetivo de TMMi es soportar las actividades de prueba y la mejora del proceso de prueba. Al igual que CMMI, TMMi trata de proporcionar a las organizaciones la mejora

de procesos, en este caso de procesos de prueba, para lo cual es sustancial contar con un mecanismo de evaluación que permita identificar las oportunidades de mejora (Van Veenendaal y Cannegieter 2010).

Aunque puede llegar a utilizarse de forma aislada, TMMi tiene un gran peso como complemento de CMMi. Como resultado, en muchos casos un nivel de TMMi necesita del soporte específico de las áreas de proceso de CMMI de su mismo nivel o de las superiores. (Zamora Hernández 2011)

TMMi podría ser utilizado en organizaciones que utilizan una estructura organizativa de pruebas basada en equipos independientes o grupos de aseguramiento de la calidad. En el caso de que los desarrolladores sean los probadores o equipos integrados de pruebas, este modelo de referencia resulta inapropiado, debido a la dimensión reducida de estos equipos, la carga de trabajo sería excesiva. En el caso de las factorías de pruebas software no resulta apropiado ya que no contempla lo procesos de gestión de servicios (Sanz Esteban 2012).

Al igual que el modelo CMMI, la desventaja para ser empleado TMMi en las organizaciones es que no presenta una definición de los procesos. Sólo proporciona directrices a partir de las cuales las organizaciones deben de definirlos. Por tanto, ni TMM ni TMMi, se corresponden con modelos orientados a un eficiente servicio de evaluación de productos de software, o a estructuras organizativas de factorías de pruebas. En cambio, pueden ser utilizados como una fuente para la definición de los procesos o como un mecanismo para la evaluación de los mismos, una vez que ya están siendo utilizados.

1.5.4 TMAP – TEST MANAGEMENT APPROACH

En la investigación de Sanz Esteban se registra que TMap es una aproximación para la gestión de las pruebas que ha sido desarrollado por la empresa Sogeti. Considera pruebas dirigidas a resultados de productos software, tanto a alto como a bajo nivel (Sanz Esteban 2012).

El contenido específico que propone TMap considera cuatro elementos fundamentales:

- Gestión de pruebas dirigida al negocio.
- Proceso de pruebas estructurado.
- Proporcionar las herramientas adecuadas.

La gestión de pruebas dirigida al negocio, reconoce que el esfuerzo total está relacionado con los riesgos del sistema que va a ser probado. La estimación y planificación del proceso de pruebas está relacionada con la estrategia de pruebas definida, y hay que implicar al cliente en el proceso de pruebas. Para esto, estructura el proceso de pruebas en cuatro partes: un plan de pruebas maestro, pruebas de aceptación y de sistema, pruebas de desarrollo y procesos de soporte relacionados con la gestión de la configuración, gestión del cambio, gestión de incidencias, gestión de problemas y datos.

La ejecución del proceso estructurado de TMap se sustenta en un conjunto de herramientas que relaciona las técnicas, la infraestructura y la organización. Pérez Lamancha resume el enfoque de TMap para definir un proceso de prueba estructurado, basado en 4 componentes:

- *Ciclo de vida*, que describe las actividades de test que deben realizarse, consistentes con el ciclo de vida del software.
- *Actividades Organizacionales*, que tienen dos componentes:
 1. La organización del equipo de test, donde cada uno debe tener tareas y responsabilidades.
 2. La incorporación del equipo de test en la organización del proyecto.
- *Infraestructura y herramientas*. El entorno de pruebas debe ser estable, controlable y representativo.
- *Técnicas aplicables al proceso de prueba*. Las técnicas permiten la ejecución de las actividades en forma estructurada y repetible (Pérez Lamancha 2006).

El autor coincide con Sanz Esteban al considerar que TMap, es una aproximación para la gestión de las pruebas que considera pruebas dirigidas a resultados de productos software. Define de forma detallada actividades y tareas a realizar, pero no es en sí mismo un modelo de procesos que sea directamente utilizable por las organizaciones, si no que requiere de trabajo de consultoría para adaptarlo a las necesidades específicas de cada empresa. Su gran extensión hace que sea difícil de implementar, especialmente en pequeñas y medianas empresas donde los recursos dedicados a las pruebas son muy escasos (Sanz Esteban 2012).

1.5.5 TPI – TEST PROCESS IMPROVEMENT

TPI -Test Process Improvement fue definido para una mejora controlada y gradual del proceso de pruebas de software, basándose en el conocimiento y la experiencia de sus autores, consultores de pruebas de Sogeti (Koomen y Pol 1999).

TPI define un conjunto de 20 áreas clave, que son: Estrategia de pruebas, Modelo del ciclo de vida, Momento de implicación, Estimación y planificación, Técnicas de especificación de pruebas, Técnicas de pruebas estáticas, Métricas, Automatización de pruebas, Entorno de pruebas, Entorno de trabajo, Compromiso y motivación, Funciones de prueba y entrenamiento, Alcance de la metodología, Comunicación, Informes, Gestión de defectos, Gestión de elementos de prueba, Gestión del proceso de pruebas, Evaluación y Pruebas de bajo nivel (Koomen y Pol 1999).

TPI establece una matriz de madurez de pruebas que define las diferentes relaciones. El propósito de esta matriz es mostrar los puntos fuertes y débiles del actual proceso de pruebas, así como ofrecer ayuda a la hora de determinar la prioridad de las acciones de mejora; es decir una matriz con datos reales ofrece a todos los participantes una visión clara de la situación actual del proceso y ayuda a definir y seleccionar las propuestas de mejora. La matriz se maneja de izquierda a derecha, de tal manera que las áreas claves poco maduras sean mejoradas en primer lugar (Sanz Esteban 2012).

Para la aplicación de TPI, se requiere que se tenga implementada la metodología que establece TMap. TPI podría ser utilizado en organizaciones que utilizan una estructura organizativa de pruebas basada en equipos independientes o grupos de aseguramiento de la calidad que utilicen la metodología TMap. En el caso de que los desarrolladores sean los probadores o equipos integrados de pruebas, donde los equipos no son muy numerosos, la carga de trabajo sería excesiva. En relación a las factorías de software no se contempla la gestión del servicio por lo que no es aplicable a esta tipología.

TPI debe ser considerado como una herramienta para estructurar las acciones de mejora del proceso de pruebas y como un medio para la comunicación, pero no es un mecanismo que contribuya a la definición de los procesos de prueba y, por tanto, no es directamente implementable en organizaciones que tratan de definir sus procesos de prueba. TPI, será únicamente utilizable de cara a mejorar los mismos en etapas posteriores. Se coincide con Sanz Esteban en plantear que este hecho se resalta aún más cuando se analiza su estructura, puesto que no define actividades ni tareas a ser ejecutadas para la realización del proceso de pruebas, ya que tan sólo define áreas clave y niveles que debería de tener la organización, pero a nivel de evaluación del proceso existente. (Sanz Esteban 2012)

1.6 OTRAS INICIATIVAS PARA LA GESTIÓN DE PRUEBAS DE SOFTWARE

En la consulta de los materiales bibliográficos relacionados con la gestión de las pruebas de software, se ha logrado destacar en los epígrafes anteriores, la relación que guardan las diferentes estructuras organizativas para enfrentar las pruebas y los modelos de referencia en la industria. Existen otros materiales que por la novedad y el aporte teórico a este propósito, el autor consideró necesario analizar como fundamento para el desarrollo de la estrategia. En primer lugar, se encuentra el Proceso de Testing Funcional Independiente propuesto por Beatriz Pérez Lamancha en 2006 y el Marco Metodológico para la mejora de las actividades de Verificación y Validación de Productos Software, de Ana Sanz Esteban en su tesis doctoral en 2012.

Entre las tareas desarrolladas en la investigación de Pérez Lamancha, destaca el análisis bibliográfico que realiza sobre el ciclo de vida específico para el desarrollo de pruebas de software, que en general, separa en distintas etapas las actividades de planificación, diseño y las de ejecución de las pruebas. El proceso de prueba que propone es el denominado ProTest, basado en el estudio de los antecedentes respecto a los procesos de prueba existentes y del estudio de los modelos de mejora de calidad y los procesos. Es un proceso validado en el Centro de Ensayos de Software de Uruguay, que cuenta con un Laboratorio de Testing Funcional (Pérez Lamancha 2006).

La principal desventaja de esta propuesta en función de la estrategia del autor de la presente investigación, se centra en la dedicación del proceso definido para las pruebas funcionales solamente. No obstante, su análisis fue válido ya que permitió conocer un proceso de pruebas independiente a la metodología usada para el desarrollo y enfocado en realizar pruebas funcionales sobre una versión ejecutable del producto, sin contar con el código fuente. Esta es una experiencia similar a la que se expone en la presente investigación, ya que se trata de un proceso definido para una organización que se dedica a las pruebas independientes de productos de software.

Otro aporte importante desde el punto de vista de procesos, lo constituye el Marco Metodológico para la mejora de las actividades de Verificación y Validación de Productos Software. Su principal aporte se basa en que define de forma completa los procesos para llevar a cabo para la verificación y validación de productos software, además, es aplicable a cualquier tipo de estructura organizativa. Para ello, los procesos se han diseñado y definido de forma modular, permitiendo que cada organización implemente únicamente aquellos que se ajusten a sus necesidades de acuerdo a su tamaño y estructura organizativa (Sanz Esteban 2012).

Esta solución integra un modelo de procesos y un modelo de competencias orientados a incrementar la calidad de los productos software mediante la reducción de defectos. El modelo de procesos integra el conjunto de buenas prácticas que permiten una detección temprana del mayor número de defectos, con el propósito de evitar que se propaguen a las etapas finales del ciclo de vida, e incrementen el número de defectos encontrados en el producto final, para que la versión que se entrega al cliente se encuentre libre de errores. Para ello incluye actividades relacionadas con la gestión de proyectos, el aseguramiento de la calidad y la gestión de configuración, entre otros. Al ser tan modular esta propuesta, repite actividades y divide el proceso de pruebas en varios que podrían estar recogidos en uno solo. La experiencia de esta iniciativa, teniendo en cuenta la necesidad, la situación problemática y el objetivo de la presente investigación, constituye un referente importante para la definición de los elementos que componen la estrategia para elevar la eficiencia del servicio de evaluación de productos en el LIPS.

Teniendo en cuenta los referentes teóricos que describen los procesos de pruebas de software, asociado a las diferentes estructuras organizativas y los modelos relacionados con la gestión de pruebas, se propone estudiar los elementos que sustentan la necesidad de la estrategia que propone el autor de la presente investigación. El objetivo es identificar los puntos de contacto, a partir de los cuales se formula la estrategia. A continuación se realiza una caracterización del entorno, evidenciando la evolución de la organización y los principios de funcionamiento del LIPS.

1.7 CARACTERIZACIÓN DEL ENTORNO

La creación del Centro de Calidad para Soluciones Informáticas (CALISOFT) en la UCI, trajo consigo un replanteo de la estructura de la organización, consolidando su misión y sus objetivos en nuevos entornos. En el ámbito de las pruebas, caracterizado por el crecimiento de la cantidad de especialistas, la tendencia al aumento de la cantidad de artefactos o productos a probar de los proyectos de la Universidad y el marcado interés de otras empresas desarrolladoras de software evaluar sus productos aprovechando las experiencias de CALISOFT, se toma la decisión de crear el Departamento de Pruebas de Software (DPSW) para enfrentar tal demanda.

Para el desarrollo de su actividad, el DPSW se estructura manteniendo el laboratorio de pruebas como Laboratorio Industrial de Pruebas de Software (LIPS), con una concepción diferente y se crea el Grupo de Ingeniería de Pruebas (GIPS), tal y como se muestra en la Figura 2.

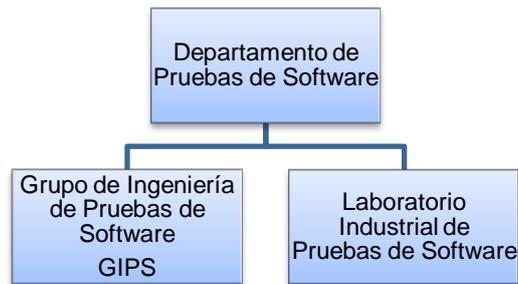


Figura 2. Estructura del Departamento de Pruebas de Software.
Fuente: Elaboración propia.

El GIPS se creó con la finalidad de definir y diseñar los tipos de pruebas que deben realizarse en el laboratorio. Reúne a un equipo de especialistas organizados a partir de las características de calidad que define la norma NC-ISO/IEC 9126:2005. Esta organización permite lograr un mayor grado de especialización en el DPSW, garantizando que se puedan definir y llevar a la práctica tipos de pruebas asociados a las características que deben evaluarse en un producto de software, para determinar su calidad. Permite además, estudiar y crear un stand de herramientas manuales y automatizadas para enfrentar los tipos de pruebas asociados a las características de calidad que establece la norma.

La concepción del LIPS según Capote García (Capote García 2011), tiene lugar como materialización de una nueva propuesta de laboratorio, a partir de la evolución del Laboratorio de Pruebas existente en la antigua Dirección de Calidad de Software de la UCI. Este fue evolucionando en función de los cambios que se realizaban en la vida productiva y docente de la Universidad, así como el aumento de la demanda en las solicitudes de pruebas de liberación. El aumento de la demanda estaba determinado por el crecimiento de los proyectos de exportación de los centros de desarrollo y el hecho de que se incorporaran a este proceso de manera paulatina, los productos y soluciones informáticas de los proyectos nacionales y de informatización. El LIPS funciona de manera similar a un proceso industrial donde se tienen entradas que son transformadas en un ciclo de actividades continuas y se obtienen salidas con valor agregado para el cliente. En este caso, se consideran entradas la solicitud de evaluación de productos y el producto o artefacto en cuestión; dentro del ciclo de transformación destaca la secuencia de actividades que se realizan para probar o evaluar el artefacto, el cual va sufriendo transformaciones a partir de las No Conformidades (NC) que se detecten; y como salidas se obtendrían el producto o artefacto con mayor calidad, el Acta de Liberación y una Minuta de Cierre con los elementos principales del proceso (Capote García 2011).

Encargado de ejecutar los procesos sustantivos del departamento, fundamentalmente el proceso de pruebas de productos de software o Pruebas de Liberación (PL), el Laboratorio guarda una relación directa con el nuevo modelo de integración formación, producción e investigación de la UCI, al formar en los estudiantes de 2do año, habilidades como probadores. Las clases del LIPS constituyen parte de su práctica profesional y forman parte de las asignaturas Proyecto de Investigación y Desarrollo I y II. Los estudiantes durante su formación, interactúan con herramientas para la gestión de la prueba como son el sistema de control de versiones SVN y el GESPRO. Además, se introdujo como elemento novedoso el uso de herramientas automatizadas, tanto para la gestión de las pruebas como para su ejecución.

1.7.1 PRINCIPIOS GENERALES DEL FUNCIONAMIENTO DEL LIPS

En el DPSW se encuentran definidos un conjunto de requisitos que rigen el funcionamiento del Laboratorio Industrial de Pruebas de Software. Estos van a establecer los modos de actuación para llevar a cabo el procesos de pruebas de software.

Primeramente se organizan turnos de trabajo de 4 horas distribuidos en sesiones de la mañana y la tarde, de lunes a viernes, y la sesión de la mañana del sábado. Estos turnos no siempre tendrán disponibilidad de probadores, pues se ajustarán a la planificación docente de las facultades.

En un turno intervienen los probadores, los profesores de CALISOFT en su rol de guía de trabajo, formador y evaluador, el especialista coordinador del proceso de PL que es un especialista del DPSW y un miembro del equipo de desarrollo del producto que esté siendo sometido a pruebas en ese turno. Estos últimos, deben explicar a los probadores los elementos esenciales relacionados con el negocio del proyecto, supervisando cómo se efectúa el desarrollo de la prueba y aclarando de manera oportuna las dudas que puedan surgir.

Durante la ejecución de una PL en el Laboratorio, se utilizan métricas para determinar, a partir de los resultados registrados de tipos de pruebas realizados, el nivel de cumplimiento o adherencia a las características y subcaracterísticas de calidad definidas en la NC-ISO/IEC 9126:2005. Para favorecer el desempeño de los procesos de pruebas, garantizando la protección y optimización del uso de los recursos disponibles, se establecen criterios de criticidad que definen parámetros para declarar un producto del desarrollo de software sometido a pruebas, en estado crítico de terminación. Estos criterios son aplicables durante todo el proceso de pruebas de software.

Se utiliza para el desarrollo de la actividad del Laboratorio, un conjunto de herramientas que favorecen el trabajo de los especialistas. En cada turno de trabajo se gestionan las NC con la ayuda del GESPRO. Las órdenes de trabajo de conjunto con el portafolio de los probadores, se publican previamente a través del SVN. Ante cada nuevo proceso, se utilizan herramientas de virtualización para el montaje de los servidores necesarios para afrontarlos, de manera que se logre un uso más eficiente de los recursos de los que se disponen. Se genera del mismo modo, un expediente de prueba, que es gestionado también con el SVN. Para el desarrollo de pruebas de funcionalidad, eficiencia y seguridad, se utilizan un conjunto de herramientas automatizadas para su realización, como los constituyen el JMeter, Selenium IDE, entre otras.

La ejecución de las pruebas se realiza utilizando como probadores a los estudiantes de 2do año de la UCI. Desempeñan prácticas laborales abarcando ocho tipos de pruebas: Funcionales, Regresión, Seguridad (nivel I), Recuperación y Tolerancia a Fallas, Usabilidad, Instalación y Configuración y Evaluación Estática.

A partir de la alta demanda de solicitudes de pruebas a aplicaciones y sistemas informáticos y su tendencia al aumento, con el paso de CALISOFT de centro de la UCI a unidad presupuestada del Ministerio de las Comunicaciones, se perfila lo que fue el Departamento de Pruebas de Software en un nuevo departamento, con otra estructura organizativa (Figura 3). El Departamento de Evaluación de Productos de Software (DEPSW) se crea con la intención de brindar un grupo importante de servicios a la Industria Cubana del Software, los que se relacionan a continuación.

1. Evaluación de productos de software con la modalidad de pruebas de liberación para la UCI.
2. Certificación de productos.
3. Pruebas de Aceptación y/o Pruebas Piloto.
4. Pruebas de Aceptación en adquisiciones de productos.

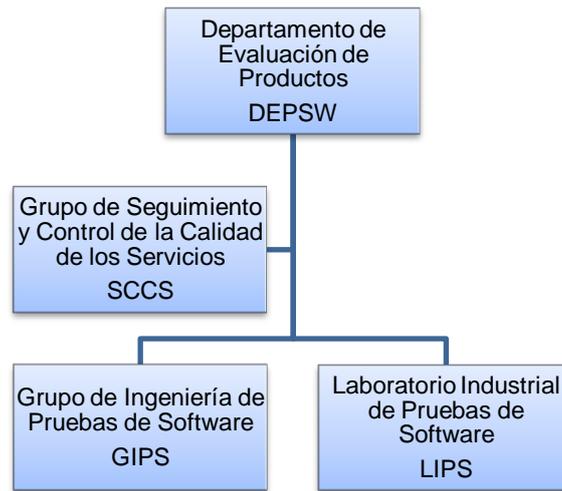


Figura 3. Estructura del Departamento de Evaluación de Productos.
Fuente: Elaboración propia.

Hasta este momento, en el DEPSW no existía ningún mecanismo para evaluar la calidad del servicio. Al estar constituido en su mayoría por especialistas que aprendieron juntos con el hacer diario y que fueron fundadores del laboratorio, existía una cierta confianza en la calidad del trabajo. Al incrementar la plantilla de los especialistas, esta confianza se mantuvo. No se pensó en evaluar la calidad del servicio que se estaba prestando, ni siquiera de aquellos procesos de evaluación cuyos coordinadores tenían una experiencia de hasta dos años o menos.

A partir de un chequeo realizado para evaluar la calidad de las pruebas que se efectuaban, se detectó un número considerable de defectos en un producto de software en un proceso de evaluación a punto de terminar. Esto provocó una alerta que movilizó a la administración. A partir de ese momento se tomaron un conjunto de acciones destinadas a verificar el estado de los procesos de evaluación que se encontraban en ejecución.

Entre las acciones se encontraba la realización de una evaluación del encargado de coordinar el proceso de evaluación, a partir de una serie de elementos que se estarían evaluando a lo largo del proceso. Esta evaluación estaría incidiendo de manera directa en el pago por resultados establecido en la entidad, lo cual provocó un descontento entre los especialistas al resultar una acción radical y reactiva. No existía ningún análisis que permitiera evaluar la situación existente y que permitiera a su vez, trazar una estrategia para lograr prestar servicios de calidad en el departamento. Este análisis, por ejemplo, no estaba teniendo en cuenta las condiciones materiales con la que cuenta el departamento y que inciden directamente en la calidad del servicio.

Otro elemento de incidencia lo constituye el cúmulo de actividades que generó para los especialistas que integran el nuevo grupo de Seguimiento y Control de la Calidad de los Servicios, creado en la nueva estructura del departamento (Figura 3). Estos, además de ejercer como coordinadores de las evaluaciones de productos, tenían que realizar actividades propias de gestión de configuración, gestión de entornos de pruebas, gestión del servicio y seguimiento de cada proceso de evaluación. Incluso, existían especialistas que tenían que ejecutar pruebas a todos los productos en los servicios de evaluación que estaban a punto de terminar. Esto provocó una sobrecarga importante en ese grupo de especialistas que, en aras de trabajar por lograr un producto de calidad, y a partir de las condiciones materiales relacionadas anteriormente, estaba incidiendo directamente en la calidad de su trabajo en la prestación del servicio.

1.8 CONCLUSIONES PARCIALES DEL CAPÍTULO

Una estrategia para elevar la eficiencia del servicio de evaluación de productos, permite alcanzar objetivos en la organización a través el establecimiento y del uso ordenado de recursos para la consecución de dichos objetivos, que le permita alcanzar ventajas competitivas con el logro de mejores índices de eficiencia. El estudio de los principales conceptos relacionados con el objeto de estudio de la presente investigación, permitió arribar a las siguientes conclusiones:

1. La definición de un proceso favorece la consecución de los objetivos de la organización y la definición de una estrategia, al ordenar los recursos y las actividades a desarrollar para transformar las entradas en salidas. Este constituye la base para diseñar una estrategia para elevar la eficiencia del servicio de pruebas.
2. Los procesos de pruebas de software permiten incrementar la calidad de los productos de un modo relativamente rápido. Existen diversas iniciativas, tanto en la industria como en la academia, que van desde los proyectos de desarrollo de software que apuestan por el aseguramiento de la calidad de sus producciones, hasta la adopción de modelos, normas y estándares de calidad y procesos de mejora.
3. Para el desarrollo de las actividades de prueba o verificación y validación, históricamente han existido estructuras organizativas que dependen de la necesidad, del entorno y del tipo de organización de que se trate. Un tipo u otro de estructura organizativa puede resultar provechoso para unos y en desventaja para otros.

4. Existe un amplio marco de modelos de referencia y otros aportes que poseen elementos que favorecen la adopción de una estrategia para elevar la eficiencia del servicio de pruebas de software, en una organización que funciona como una factoría de pruebas.
5. El servicio de evaluación de productos de CALISOFT se sustenta en la estructuración de un Departamento de Evaluación de Productos de Software que cuenta con un Laboratorio Industrial de Pruebas de Software, un Grupo de Seguimiento y Control de la Calidad de los Servicios y un Grupo de Ingeniería de Pruebas de Software.
6. El estudio del entorno en el que se desarrolla el LIPS determina las condiciones y la necesidad de organizar todas las iniciativas, en una estrategia para elevar la eficiencia del servicio de evaluación de productos de software.

CAPÍTULO II. PLANTEAMIENTO DE LA ESTRATEGIA

2.1 INTRODUCCIÓN

El Laboratorio Industrial de Pruebas de Software de CALISOFT es el resultado de más de 5 años de trabajo, centrados fundamentalmente en la ejecución de pruebas a aplicaciones informáticas desarrolladas en la UCI, de proyectos de exportación, nacionales o de informatización de la universidad. Su evolución ha estado marcada por el incremento de la demanda del servicio de evaluación de productos de software. Esto ha llevado a replantearse en varias ocasiones los procesos y procedimientos en aras de lograr un servicio de mayor calidad y satisfacer las necesidades de los clientes.

En el presente capítulo se determina una estrategia para la gestión del proceso de evaluación de productos de software en el LIPS, que parte de un diagnóstico al estado real del problema presentado en el capítulo anterior, que se toma como base para el desarrollo de la investigación. Se plantea el objetivo general así como los metas y objetivos a alcanzar con la estrategia. Se establecen las actividades y acciones a desarrollar. Se planifican los recursos y métodos, para luego desarrollar una evaluación de los resultados y del avance en el desarrollo de la estrategia.

2.2 DEFINICIÓN DE LA ESTRATEGIA

Según Francés (Francés 2006), la estrategia se refiere a la combinación de los medios a emplear para alcanzar los objetivos, en presencia de incertidumbre. La estrategia tiene que ver básicamente con la búsqueda de la ventaja competitiva. Las ventajas competitivas se buscan pero no siempre se encuentran, y esto es lo que diferencia a las organizaciones de éxito con las que no lo tienen. En el libro “Estrategia: de la visión a la acción” escrito por Juan Carrión Maroto, define que la estrategia tiene que ver con los objetivos a conseguir y los planes para llegar a ellos, se relaciona con el planteamiento de objetivos a largo plazo y la asignación de los recursos disponibles para el logro eficiente de dichos objetivos (Carrión Maroto 2007).

Labarca considera (Labarca 2008) que la estrategia adoptada representa la mejor apuesta, pero nada garantizará su éxito. Cuando existe plena certeza acerca de la eficacia de los medios para alcanzar los objetivos, no se requiere de estrategia. La relación entre objetivos y medios se vuelve puramente técnica.

Por lo anterior, se puede llegar a la definición de que la estrategia permite alcanzar objetivos en la organización, a través del establecimiento y del uso ordenado de

recursos para la consecución de dichos objetivos, que le permita alcanzar ventajas competitivas.

A partir de lo estudiado en el capítulo anterior y teniendo en cuenta las características del entorno, el autor de la presente investigación determinó fundamentar la estrategia a partir del desarrollo de dos etapas. Para lograr una gestión de procesos primeramente se hace necesario definir y establecer el proceso que sustenta el servicio de Evaluación de Productos de Software. Es precisamente el establecimiento de este proceso, la primera etapa que se propone en la estrategia. Luego de definir y establecer el proceso, se propone dar pasos en función de gestionar dicho proceso y que tribute a elevar la eficiencia del servicio. Se estaría hablando entonces, de la segunda etapa que se propone en la estrategia, determinada por la definición de las actividades de seguimiento y control del proceso Evaluación de Productos de Software.

En la siguiente figura (Figura 4), se puede observar de manera gráfica las dos etapas que definen la presente estrategia y que permitirán llevar a este servicio de un estado inicial a un estado deseado. En este caso el estado deseado se corresponde con el problema de la presente investigación, el cual está dirigido a elevar la eficiencia del servicio de Evaluación de Productos de Software.

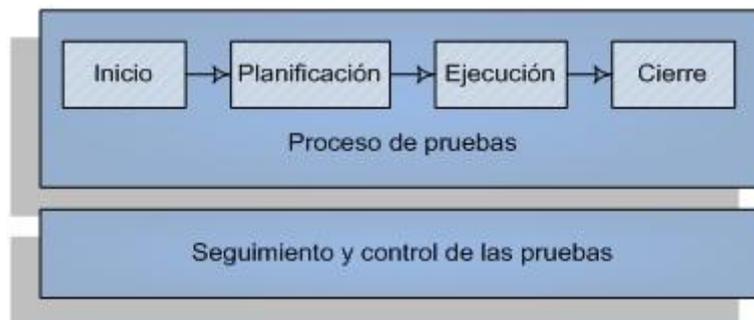


Figura 4. Etapas de la estrategia.
Fuente: Elaboración propia.

A continuación se describirá en detalles en qué consiste cada una de estas etapas. En un primer momento se definirá el proceso y las acciones para su establecimiento y posteriormente se estarán describiendo las actividades de seguimiento y control con la relación de los involucrados y los responsables de darle cumplimiento.

2.2.1 ETAPA I. DEFINICIÓN DEL PROCESO EVALUACIÓN DE PRODUCTOS DE SOFTWARE

El proceso de evaluación de productos de software será el sustento para la prestación de este servicio en el departamento. Se debe partir de la premisa de que se dificulta

brindar un buen servicio si no están definidas las actividades sucesivas que permitirán evaluar un producto de software determinado. Estas actividades deben estar en correspondencia con la realidad objetiva del departamento, teniendo en cuenta el punto de vista de todos los involucrados. A partir de lo anterior, el autor determinó que para el establecimiento del proceso, el primer paso corresponde con su definición. Luego se deben ejecutar un conjunto de actividades que permitan establecer dicho proceso, que sea transmitido, ejecutado, y validado desde la trasmisión y la ejecución.

PRIMEROS ACERCAMIENTOS AL PROCESO DE EVALUACIÓN DE PRODUCTOS DE SOFTWARE

Para la definición del proceso se desarrolló un análisis del primer acercamiento que se realizó a partir de la conceptualización e implantación del Laboratorio Industrial de Pruebas de Software, desarrollado por Capote García (Capote García 2011). La autora conceptualizó los elementos esenciales que componen el LIPS y desarrolló una descripción general de su funcionamiento, a partir de la relación de las actividades y subprocesos que se ejecutan de manera continua y armónica. Partiendo desde un análisis de solicitud hasta una reunión de cierre, se perfilaban cinco actividades y cinco subprocesos fundamentales (Figura 5).

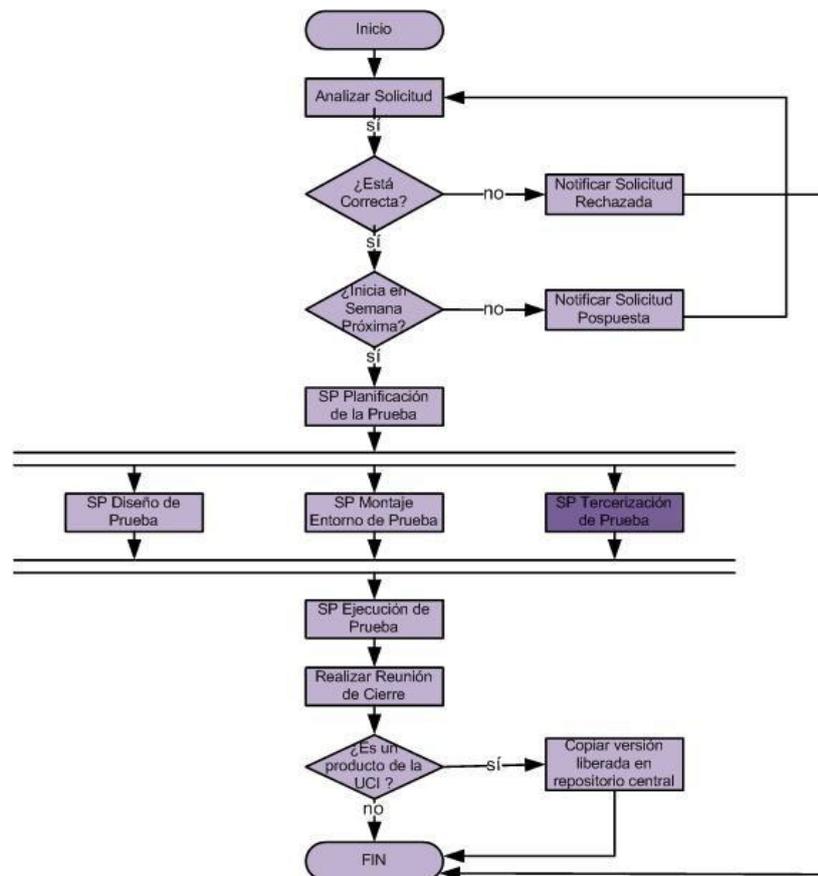


Figura 5. Proceso del LIPS.
Fuente: (Capote García 2011).

Los subprocesos relacionados son: Planificación de la Prueba, Diseño de la Prueba, Montaje del Entorno de Prueba, Tercerización de la Prueba y Ejecución de la Prueba. En el informe de la investigación de Capote García no se describen las actividades que componen los subprocesos relacionados anteriormente. Solo se limita a relacionar las consecución de las actividades y subprocesos. No se relacionan las entradas y las salidas de cada actividad ni los responsables de su ejecución.

A partir del proceso que se presenta, se hace referencia entonces al Libro de Procesos del LIPS, el cual fue elaborado utilizando la plantilla definida por el Grupo de Normalización y Métricas de CALISOFT, como parte del Programa de Mejora que se desarrollaba en la UCI, para lograr la certificación nivel 2 de CMMI, donde estaría descrito el proceso de forma textual y gráfica (Capote García 2011). Este proceso en el Libro de Procesos del LIPS, estaba compuesto por cuatro etapas fundamentales, como se muestra en la (Figura 6).

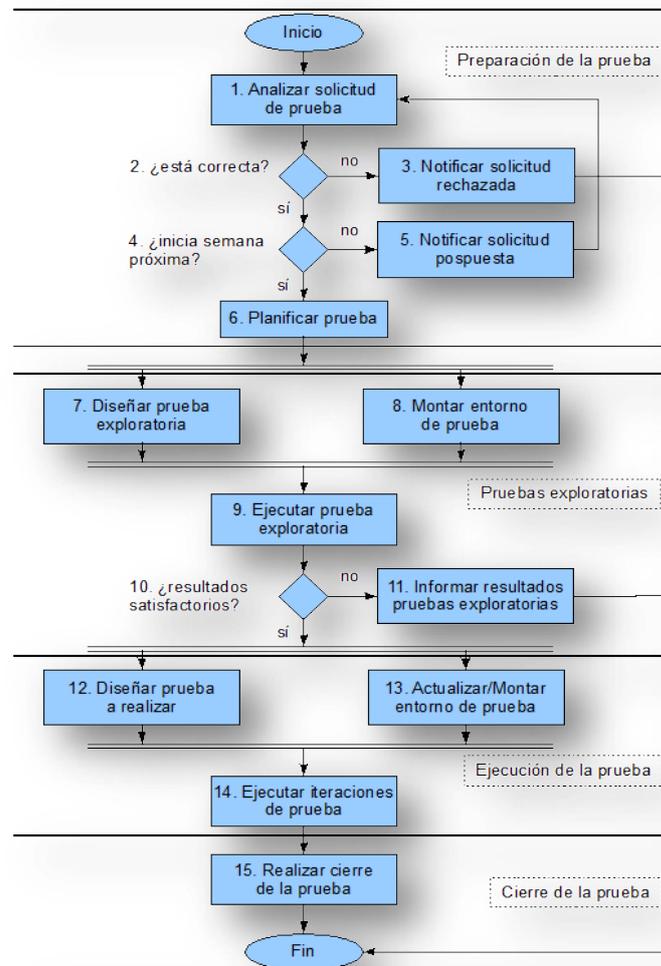


Figura 6. Proceso general para las Pruebas de Liberación.
Fuente: Manual de procesos del LIPS, 2011.

Las etapas en las que se divide el proceso son: Preparación de la prueba, Pruebas exploratorias, Ejecución de la prueba y Cierre de la prueba. Se basa en el desarrollo de 15 actividades que van desde el análisis de la solicitud hasta realizar el cierre de la prueba.

Partiendo del análisis de este proceso se pudo determinar que el mismo no contenía todas las actividades que se realizaban en el laboratorio. Es un proceso que contempla la realización de evaluaciones estáticas de artefactos o productos que no constituyen software, sino que se generan como soporte o apoyo en el proceso de desarrollo. Tal es el caso, por ejemplo, de las especificaciones de requisitos o manuales de usuario. Otra deficiencia que se pudo constatar es que no se contemplan los encuentros entre clientes y proveedores para negociar los intereses de ambas partes, crear las bases del acuerdo que se pretende establecer y definir las actividades a realizar, así como los derechos y deberes de ambas partes. En este caso se refiere a los proyectos o entidades que realizan solicitudes de evaluación de productos a CALISOFT y el LIPS como proveedor de este servicio.

Como resultado del análisis se pudo constatar además que no se tiene en cuenta en el proceso, la utilización de herramientas de gestión para la ejecución de las evaluaciones de productos. Se dedican dos actividades a la preparación del entorno de pruebas, pero no se tienen en cuenta las herramientas de gestión de no conformidades, de gestión de la configuración o de gestión de expedientes de prueba, que no forman parte del entorno de pruebas y que constituyen uno de los elementos aportados y conceptualizados por Capote García (Capote García 2011).

En el proceso se definen las pruebas exploratorias como una etapa previa e independiente a la ejecución de las pruebas. Como se puede observar en la representación del proceso, se hace una diferenciación entre una etapa y la otra, lo cual no da lugar a que se pueda realizar una exploración de la calidad del producto en cualquier momento durante el proceso de pruebas. De acuerdo a las características del LIPS y según León Perdomo, el propio proceso de pruebas del laboratorio requiere que se realicen en dos momentos fundamentales: las Pruebas Exploratorias Iniciales (PEI) y las Pruebas Exploratorias Parciales (PEP). Las PEI, como su nombre lo indica, serán las primeras que se realicen antes de la primera iteración de pruebas funcionales, mientras que las PEP serán aplicadas de manera simultánea con el resto de las evaluaciones, en cualquier otro momento, pero fundamentalmente en la segunda iteración de pruebas funcionales (León Perdomo 2012).

Tomando como punto de partida este análisis realizado, se propone entonces un proceso de evaluación de productos de software que brinde solución a las deficiencias detectadas y que responda a las necesidades del LIPS. Este proceso va a constituir el primer paso en función de lograr la prestación de un servicio eficiente. Para su establecimiento, luego de la descripción de las actividades, las correspondientes entradas y salidas y los responsables para su ejecución, será necesario además el desarrollo de una capacitación que garantice en primer lugar su validación y la estandarización de los modos de hacer. Luego se realizaría la implementación del proceso, la cual va a estar apoyada en el desarrollo de las actividades de seguimiento y control que se definen como parte de la presente estrategia.

DESCRIPCIÓN DEL PROCESO EVALUACIÓN DE PRODUCTOS DE SOFTWARE

El proceso Evaluación de Productos de Software está constituido por cuatro subprocesos fundamentales como se muestra en la Figura 7. Para su ejecución se precisa de la realización de una solicitud de servicio. Esta constituye el criterio de entrada para dar paso a la ejecución del primer subproceso, el de Gestionar solicitudes.

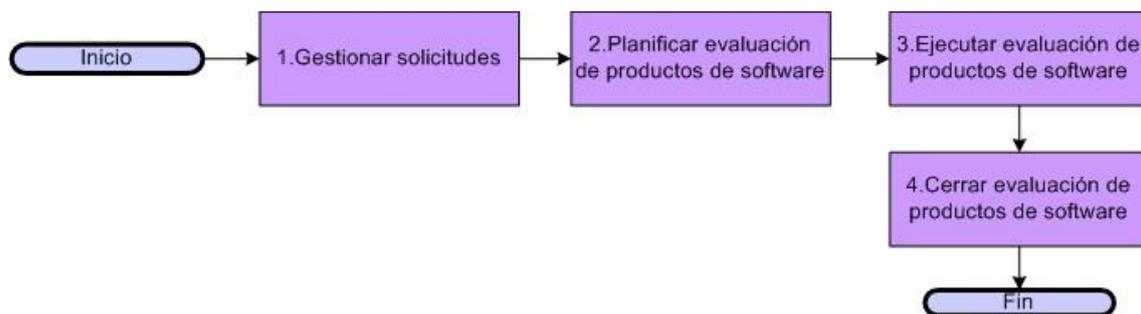


Figura 7. Proceso Evaluación de Productos de Software.
Fuente: Elaboración propia.

Una vez aceptada la solicitud, tiene lugar el subproceso Planificar evaluación de productos de software. Este es el momento en que se acuerda entre las partes los requisitos de la evaluación, los recursos necesarios, las responsabilidades de los involucrados y se preparan los entornos de pruebas y de gestión de la evaluación para dar lugar al tercer subproceso, el de Ejecución de la evaluación de productos de software. En este subproceso se describen todas las actividades relacionadas con la ejecución de la evaluación, donde se tienen en cuenta los tipos de pruebas a realizar, el desarrollo de pruebas exploratorias y las iteraciones de pruebas. Se gestionan las no conformidades detectadas en cada iteración hasta obtener la evaluación del producto, la cual puede estar fundamentada en la emisión de un dictamen técnico o un acta de liberación. La última etapa o el último subproceso es el denominado Cierre de la evaluación de productos de software. En este subproceso se realizan todas las

actividades encaminadas a comunicar, aprobar y dejar constancia de los resultados de la evaluación.

A continuación se describen cada uno de los subprocesos que corresponden al proceso de Evaluación de Productos, haciendo énfasis en las actividades y conceptos que se proponen.

Subproceso Gestionar Solicitudes

Este subproceso se encuentra alineado con la definición de los procesos de Gestión del Servicio de (Sanz Esteban 2012), en su Marco Metodológico para la mejora de las actividades de Verificación y Validación de Pruebas Software. Se corresponde específicamente con el proceso de Adquisición del Servicio. Sanz Esteban define que este proceso se ejecutará en el momento que un cliente se encuentre interesado en contratar uno o varios servicios ofertados. Al gestionar solicitudes se está dando curso justamente al interés de un cliente determinado de ejecutar el servicio de evaluación de productos de software.

El subproceso inicia cuando el cliente realiza la solicitud del servicio, la cual es analizada en el DEPSW para determinar si se puede asumir o no, a partir de los datos que recoge. De poseer la información necesaria se le asigna a un especialista del departamento, quien atenderá dicha solicitud para dar continuidad al proceso. Como resultado de esta actividad, se le notifica al cliente la realización de un encuentro denominado Reunión de Inicio (Díaz Pérez y Pérez Serrano 2012b). De esta manera finaliza el subproceso para dar lugar a la planificación de la evaluación (Figura 8).

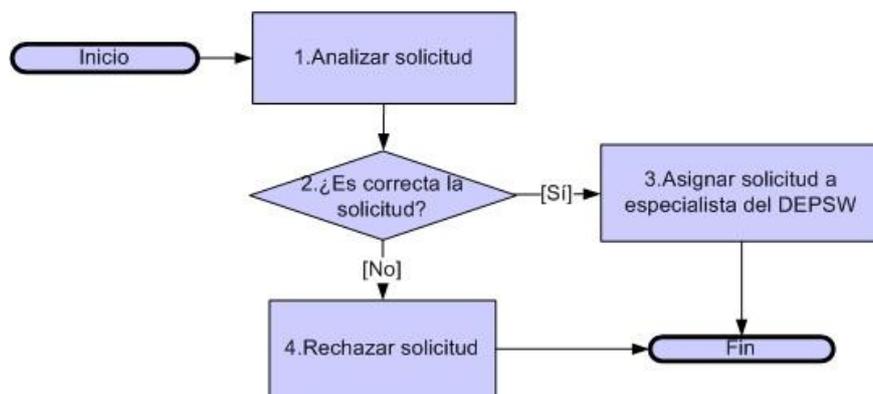


Figura 8. Subproceso Gestionar Solicitudes.
Fuente: Elaboración propia.

Si de otro modo, la solicitud no posee la información requerida para poder asumir el proceso, se procede a rechazarla. Finaliza de manera abrupta el proceso Evaluación de Productos de Software, realizando una notificación de negación de la solicitud.

Subproceso Planificar Evaluación de Productos de Software

Cuando una solicitud ha sido asignada para la evaluación de productos, se inicia el proceso de planificación que se muestra en la Figura 9. Este proceso comienza con el desarrollo de la Reunión de Inicio donde las partes acuerdan el plan de evaluación a seguir, el entorno de gestión de la evaluación y el entorno de pruebas en caso de ser necesario.

A partir de este momento y durante toda la ejecución de la evaluación, se tienen en cuenta un conjunto de criterios que ayudan a determinar la factibilidad de continuar. Estos criterios han sido denominados en el departamento, Criterios de Criticidad (Velázquez Cintra 2013). Establecen los parámetros para declarar un producto del desarrollo de software en estado crítico de terminación, a partir de los cuales puede determinarse si se detiene o finaliza la evaluación de manera abrupta. En este caso, se notifica la determinación de abortar el proceso de evaluación y en un encuentro con el cliente se explican las causas.

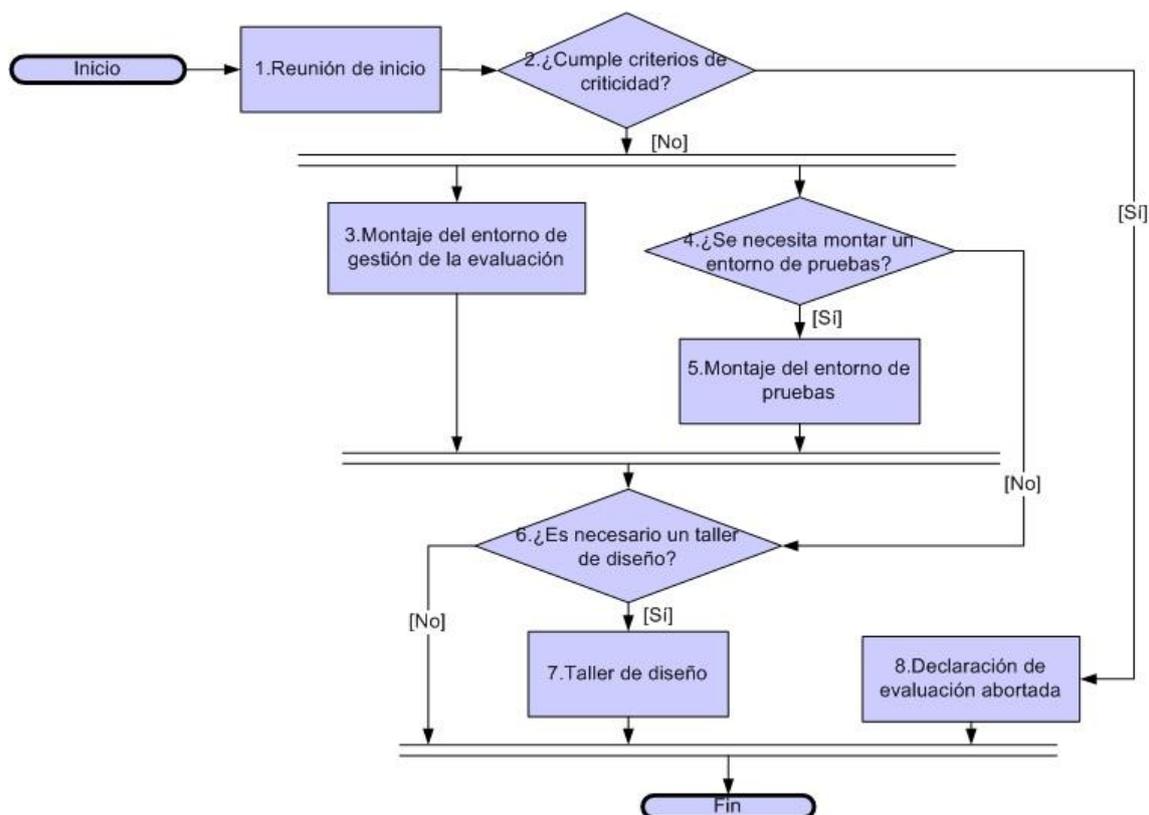


Figura 9. Subproceso Planificar Evaluación de Productos de Software.
Fuente: Elaboración propia.

Luego de la Reunión de Inicio se evalúa si se cumple algún criterio de criticidad de los relacionados anteriormente. De continuar con el proceso se procede inmediatamente al montaje del entorno de gestión de la evaluación y de ser preciso, el entorno de pruebas. Este último está en dependencia del artefacto o producto del desarrollo de

software sometido a evaluación (Anexo 6). Si se trata de un artefacto que no constituye software, sino que un entregable de soporte al proceso de desarrollo o complemento del software, no es preciso montar un entorno de pruebas.

Para culminar la planificación de la evaluación se desarrolla un taller para diseñar la ejecución de los diferentes tipos de pruebas solicitados. En este momento se realizan los últimos ajustes al plan de la evaluación. Dado el caso de que el (los) tipo(s) de prueba(s) a desarrollar no requiera(n) un diseño detallado de la evaluación, por la poca complejidad del (de los) artefacto(s) como puede ser el caso, se puede prescindir de este momento y dar paso a la ejecución de la evaluación de productos de software.

Subproceso Ejecutar Evaluación de Productos de Software

Hasta este momento del proceso, se encuentra montado el entorno de gestión de la evaluación, y en caso de ser una aplicación, se encuentra listo el entorno de pruebas. Ha sido gestionado y aprobado también el plan de evaluación, donde se establecen los criterios para su ejecución, los roles, responsabilidades de las partes, la estrategia de evaluación y se ha especificado un cronograma detallado de las actividades de la evaluación. Corresponde entonces ejecutar todas las actividades que fueron planificadas para el desarrollo de la evaluación del producto de software.

La ejecución de la evaluación está basada en el desarrollo de tres actividades y un nuevo subproceso. Comienza con el desarrollo de las PEI (León Perdomo 2012), actividad a partir de la cual se evalúa el cumplimiento de los Criterios de Criticidad (Velázquez Cintra 2013).

Se introducen dos nuevos conceptos relacionados con la prestación del servicio de evaluación de productos de software. Se trata de dos modalidades en que se puede prestar el servicio. La primera y más simple se corresponde con la ejecución de una evaluación a partir de una solicitud realizada por cualquier entidad o ente independiente. Esta se caracteriza por ejecutar una iteración de pruebas (Díaz Pérez y Pérez Serrano 2012a), a partir de los tipos de pruebas solicitados (Anexo 7) y registrar las no conformidades (Góngora Rodríguez et al. 2009) para elaborar un dictamen técnico en la etapa de cierre de la evaluación.

La segunda modalidad es la de pruebas de liberación (Díaz Pérez y Pérez Serrano 2012a), heredada del DPSW cuando CALISOFT era un centro de la UCI. Esta, en comparación con la modalidad anterior, posee un mayor grado de complejidad. La misma requiere de la ejecución de un conjunto de actividades y procesos contenidos en el subproceso Ejecutar iteraciones de pruebas, como se muestra en la Figura 10.

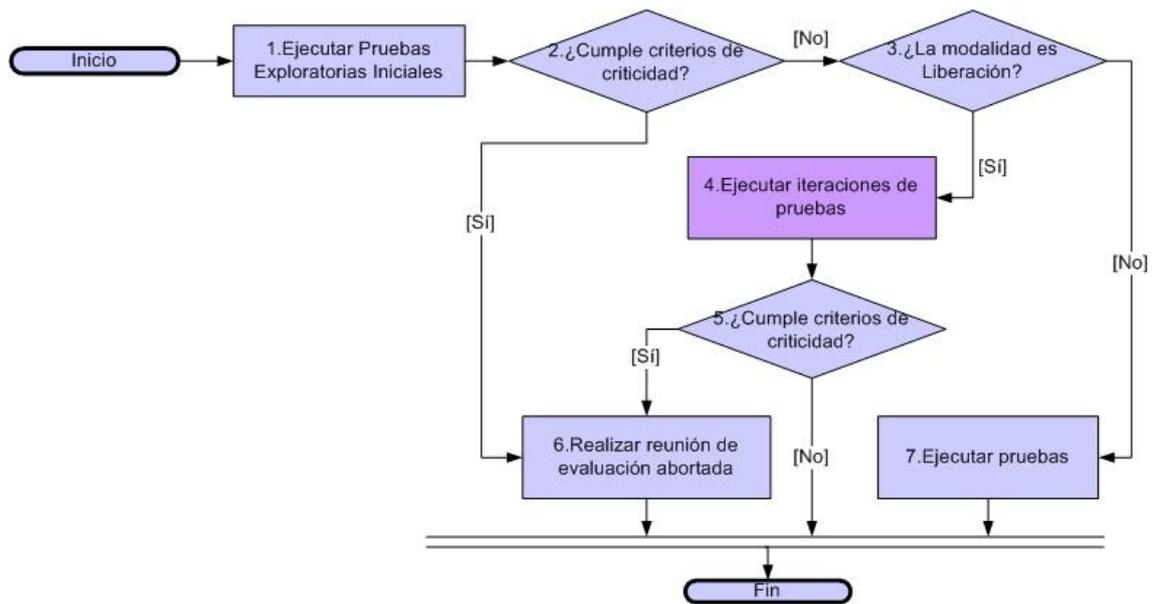


Figura 10. Subproceso Ejecutar Evaluación de Productos de Software.
Fuente: Elaboración propia.

En este subproceso se realizan las PEP y se ejecutan los tipos de pruebas solicitados (Anexo 7). Luego se gestionan todas las no conformidades (Góngora Rodríguez et al. 2009) y se vuelven a ejecutar los tipos de pruebas solicitados hasta tanto se hayan realizado por tercera vez o no sean necesarios para finalizar la evaluación. En caso de ser necesario también, se puede ejecutar una evaluación final que comprende la realización de una prueba de integración y/o prueba final.

La gestión de las no conformidades y la ejecución de una evaluación final, constituyen subprocesos que se realizan como parte de la ejecución de las iteraciones de pruebas. Dentro de la gestión de las no conformidades se ejecutan actividades relacionadas con la entrega de las no conformidades y la realización de las pruebas de regresión (Pressman 2002).

Al finalizar cada una de las actividades descritas anteriormente, se realiza una evaluación de los Criterios de Criticidad (Velázquez Cintra 2013) para evaluar el estado del artefacto o producto y valorar la viabilidad de continuar el proceso de evaluación.

Subproceso Cerrar Evaluación de Productos de Software.

Una vez que ha sido ejecutada la evaluación del producto, en correspondencia con la estrategia de pruebas definida en el plan de evaluación y en función de la modalidad de la evaluación, se tiene un registro de las no conformidades que se detectaron. Estas deben ser gestionadas debidamente de manera tal que estén creadas las condiciones para culminar el proceso de evaluación. Puede darse el caso que no se

hayan detectado no conformidades, lo que puede ser una premisa también para el inicio de este cuarto y último subproceso, el de Cerrar Evaluación de Productos de Software.

Se diseñaron cinco actividades que van desde la preparación del cierre de la evaluación hasta la actualización del registro de evaluaciones de productos de software (Figura 11).

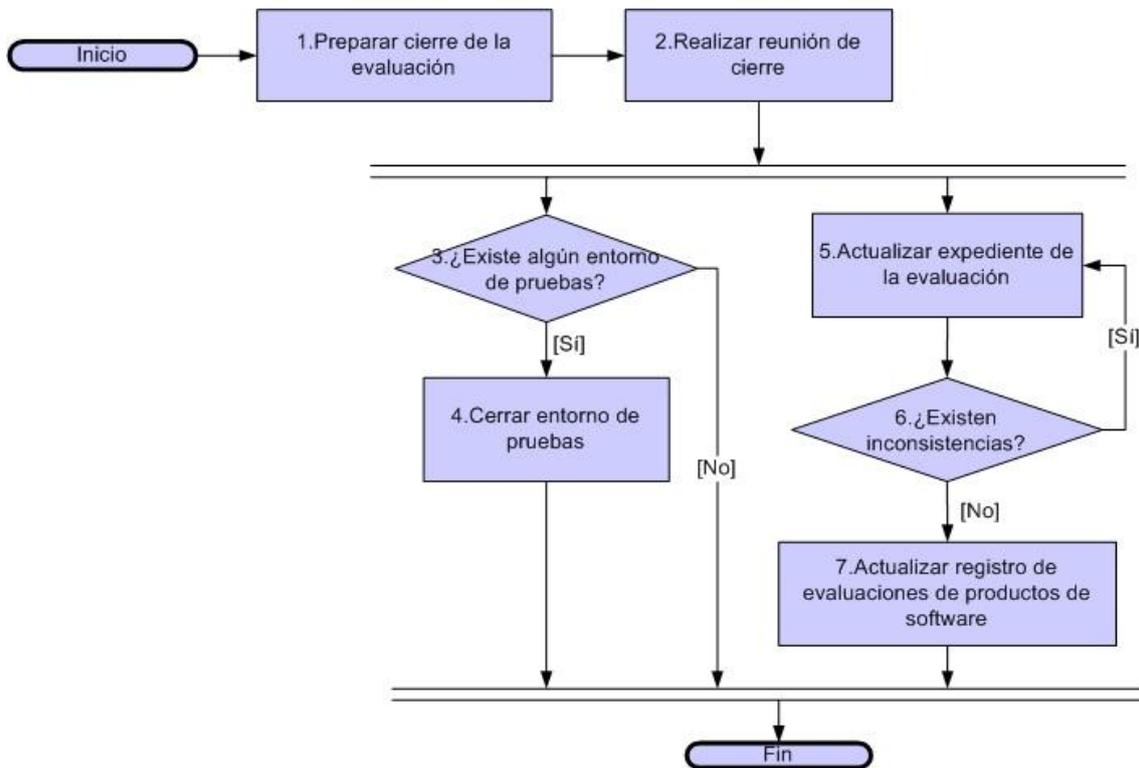


Figura 11. Subproceso Cerrar Evaluación de Productos de Software.
Fuente: Elaboración propia.

Al ejecutar el cierre de la evaluación, la primera actividad que se realiza es la preparación de este cierre. En la misma, se elaboran el dictamen técnico o el acta de liberación en dependencia de los resultados de la evaluación y de la modalidad en cuestión. Se prepara también un informe de la evaluación, el que será presentado en un encuentro con el cliente, denominado Reunión de Cierre. Para su desarrollo, y como resultado también de la actividad de preparación del cierre, se envía una notificación a los involucrados para efectuar este encuentro.

Ejecutada la reunión de cierre donde se presentaron los resultados de la evaluación, se ejecutan dos actividades simultáneas. Dado el caso de que se haya montado el entorno de pruebas, se procede a cerrar este entorno y a actualizar el expediente de la evaluación a partir del cierre del entorno de gestión. Se realiza una salva de ambos entornos y del expediente de la evaluación, como constancia de la ejecución de la

evaluación y se procede entonces a actualizar el registro de evaluaciones de productos de software del DEPSW. Finaliza de esta manera, la ejecución del proceso Evaluación de Productos de Software.

Una vez que se han definido las actividades que sustentan el proceso de evaluación de productos de software diseñado y los principales artefactos que se generan, corresponde proponer los roles que intervienen y las responsabilidades que le competen a cada uno, como parte del diseño de la estrategia. En el siguiente epígrafe se presentarán estos roles, complementando la definición del proceso en función de su establecimiento, como parte de la estrategia que se propone en la presente investigación.

DEFINICIÓN DE ROLES Y RESPONSABILIDADES

En el marco metodológico y tecnológico para la mejora de las actividades de verificación y validación de productos software de Sanz Esteban, se presenta un modelo de competencias en el que se definen las competencias generales comunes a los cuatro roles que identificó y las competencias técnicas propias de cada rol (Sanz Esteban 2012). A partir de esta experiencia, el autor de la presente investigación propone diez roles, definiendo las responsabilidades de cada uno en el proceso Evaluación de Productos de Software diseñado.

La diferencia de la cantidad de roles propuestos, en relación a los cuatro roles que propone Sanz Esteban, responde a la especialización de funciones a partir de la concepción del LIPS como fábrica, proceso industrial. Otra razón es que el proceso está definido en función de las características de una organización independiente de pruebas. Esta mejora organizativa propuesta por Capote García (Capote García 2011) ha permitido la especialización de los profesionales del DEPSW, favoreciendo el aumento de la capacidad de brindar el servicio de pruebas a la industria de software en Cuba.

Para enfocar la responsabilidad que le atañe a cada rol, estos se representaron en dos grupos fundamentales. Por una parte, velando por la correcta ejecución del proceso, en función de lograr la eficiencia en la prestación del servicio de evaluación de productos, se encuentran los roles administrativos. Estos van a formar un componente importante para la toma de decisiones durante la prestación del servicio. Por otro lado se representan los roles ejecutivos, que constituyen los actores fundamentales para desarrollar el proceso de evaluación de productos propuesto. No de menor importancia, en el centro de la representación, se ubica al rol cliente, no solo como la razón del servicio sino como ente activo en el desarrollo del proceso. En unos

momentos asume características de las que agrupan los roles ejecutivos y en otro momento las de los administrativos (Figura 12). En los siguientes apartados se describen los roles antes mencionados, con sus respectivas funciones o responsabilidades.

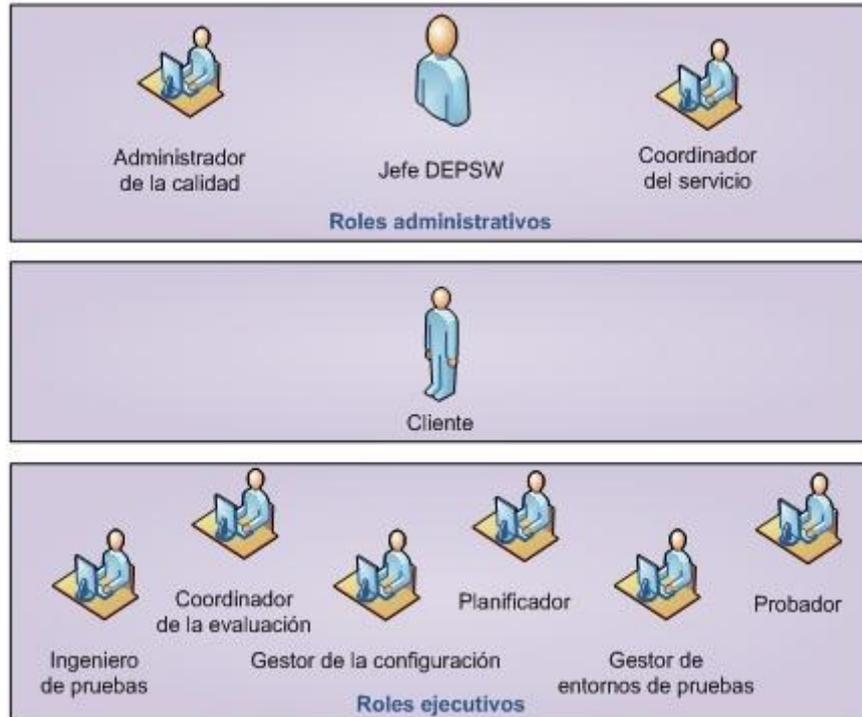


Figura 12. Roles que intervienen en la evaluación de productos.
Fuente: Elaboración propia.

Roles administrativos.

Los roles administrativos agrupan a tres de los propuestos. Estos son el **Jefe DEPSW**, el **Administrador de la calidad** y el **Coordinador del servicio**. Las competencias técnicas de los roles administrativos se corresponden con las definidas por Sanz Esteban (Sanz Esteban 2012) para el rol *Responsable de la Unidad Organizativa*.

El rol **Jefe DEPSW** es una figura que se asume desde la estructura organizativa de CALISOFT. Su función dentro del proceso que se ha propuesto está enfocada a tomar decisiones desde el punto de vista administrativo, dirigidas fundamentalmente a la gestión de los recursos de los que dispone el departamento. El **Administrador de la calidad** por su parte, es un rol asumido de la estructura del DEPSW, donde ejerce como jefe del grupo de Seguimiento y Control de la Calidad de los Servicios (Figura 3). Sus funciones están orientadas a velar por la calidad en la prestación de los servicios que se brindan en el departamento, por lo que, además de las actividades relacionadas con el proceso de evaluación de productos, su actividad se centra en las

actividades de seguimiento y control que serán definidas en la segunda etapa de la presente estrategia. El tercer rol agrupado dentro de los roles administrativos es el rol **Coordinador del servicio**. Su responsabilidad está dirigida a gestionar el proceso y el servicio de de evaluación de productos de software, a partir de la alta responsabilidad que posee en su establecimiento. Constituye la interfaz directa de la organización para con los clientes y es el responsable de su satisfacción al proponer e implementar mejoras al proceso. Se puede consultar las responsabilidades de cada uno en la Tabla 2.

Tabla 2. Responsabilidades de los roles administrativos.

Fuente: Elaboración propia.

<p>Jefe DEPSW</p> <p>Administrar las solicitudes de evaluación de productos de software.</p> <ul style="list-style-type: none"> ○ Aprobar las solicitudes de evaluación de productos de software. ○ Rechazar las solicitudes de evaluación de productos de software. <p>Administrar los recursos humanos de los que dispone el DEPSW.</p> <ul style="list-style-type: none"> ○ Asignar especialistas a los procesos de evaluación de productos de software. ○ Mover especialistas a los procesos de evaluación de productos de software. <p>Administrar los recursos materiales de los que dispone el DEPSW.</p> <p>Analizar el estado de los procesos que se ejecutan y tomar decisiones sobre su curso.</p> <ul style="list-style-type: none"> ○ Detener procesos de evaluación de productos de software. ○ Abortar procesos de evaluación de productos de software. <p>Establecer políticas para la correcta prestación del servicio de evaluación de productos de software.</p> <ul style="list-style-type: none"> ○ Definir y mantener los Criterios de Criticidad. ○ Definir y mantener los criterios de inicio y fin de una evaluación.
<p>Administrador de la calidad</p> <p>Presentar el servicio de evaluación de productos de software al cliente.</p> <p>Gestionar el proceso de evaluación de productos de software.</p> <ul style="list-style-type: none"> ○ Definir y mantener el proceso de evaluación de productos de software. ○ Definir y mantener las plantillas de los documentos que conforman el expediente de la evaluación. ○ Definir y mantener los roles y las responsabilidades de los que intervienen en el proceso de evaluación de productos de software. <p>Monitorear el desempeño del proceso de evaluación de productos de software.</p> <ul style="list-style-type: none"> ○ Definir y mantener indicadores para evaluar el desempeño del proceso de evaluación de software. <p>Evaluar el desempeño de los especialistas del DEPSW como coordinadores de las evaluaciones de productos.</p> <ul style="list-style-type: none"> ○ Participar en las reuniones de inicio y de cierre de las evaluaciones. <p>Mantener a la administración actualizada sobre el estado de la ejecución de los procesos de evaluación de productos de software.</p> <ul style="list-style-type: none"> ○ Emitir informes sobre el avance de los procesos de evaluación de productos de software. <p>Capacitar a los roles ejecutivos para el desarrollo del proceso de evaluación de productos de software.</p>
<p>Coordinador del servicio</p> <p>Presentar el servicio de evaluación de productos de software al cliente.</p> <p>Gestionar el proceso de evaluación de productos de software.</p> <ul style="list-style-type: none"> ○ Definir y mantener el proceso de evaluación de productos. ○ Definir y mantener las plantillas de los documentos que conforman el expediente de la evaluación. ○ Definir y mantener los roles y las responsabilidades de los que intervienen en el proceso de evaluación de productos de software. <p>Monitorear el desempeño del proceso de evaluación de productos de software.</p>

- Definir y mantener indicadores para evaluar el desempeño del proceso de evaluación de software.
- Evaluar el desempeño de los especialistas del DEPSW como coordinadores de las evaluaciones de productos de software.
- Participar en las reuniones de inicio y de cierre de las evaluaciones.
- Mantener a la administración actualizada sobre el estado de la ejecución de los procesos de evaluación de productos de software.
- Emitir informes sobre el avance de los procesos de evaluación de productos de software.
- Capacitar a los roles ejecutivos para el desarrollo del proceso de evaluación de productos de software.

Roles ejecutivos.

Los roles ejecutivos agrupan a seis roles vitales para la ejecución del proceso que propone el autor de la presente investigación. El **Coordinador de la evaluación**, el **Planificador**, el **Ingeniero de pruebas**, el **Gestor de la configuración**, el **Gestor de entornos de pruebas** y el **Probador**, juegan un papel activo en la evaluación de productos.

El rol **Coordinador de la evaluación** es el responsable de la ejecución de un proceso de evaluación de productos determinado. Es asumido por un especialista del DEPSW, una vez que le ha sido asignada una solicitud de evaluación previamente aprobada. Sus competencias técnicas se corresponden con las definidas por Sanz Esteban para el rol *Gestor de Pruebas* (Sanz Esteban 2012). El rol **Planificador**, como su nombre lo indica, es el encargado de planificar el desarrollo de las actividades relacionadas con el servicio de evaluación de productos de software. Su responsabilidad se mueve desde la gestión de la disponibilidad de especialistas para asumir procesos de evaluación, hasta la planificación de turnos de trabajo en el LIPS. Quien se encarga de la realización de las actividades de ejecución de las pruebas o evaluación técnica de los artefactos en el LIPS es el rol **Probador**. Su responsabilidad está definida y se corresponde con las competencias técnicas definidas por Sanz Esteban para el rol *Tester* (Sanz Esteban 2012). Es importante destacar la particularidad del rol **Probador** de ser asumido por estudiantes de segundo año de la UCI, como parte de su desarrollo curricular y a partir de conceptualización del LIPS ofrecida por Capote García (Capote García 2011). Lo anterior no elimina la posibilidad de que el rol en cuestión pueda ser asumido por personas con competencias técnicas de mayor nivel, por profesionales. En ocasiones puede ser asumido por los mismos especialistas, en función del proceso de evaluación de productos de software en cuestión, a partir de decisiones adoptadas por los roles administrativos (Ver Tabla 3).

Otro de los roles ejecutivos que se proponen es el rol **Ingeniero de pruebas**. Sus competencias técnicas están alineadas con las definidas por Sanz Esteban (Sanz

Esteban 2012) para este mismo rol. En el proceso de pruebas que el autor de la presente investigación propone, este rol está determinado por los tipos de pruebas que se ejecutan en cada evaluación de productos de software.

Tabla 3. Responsabilidades de los roles ejecutivos.
Fuente: Elaboración propia.

Coordinador de la evaluación
<p>Planificar la ejecución de la evaluación de productos de software.</p> <ul style="list-style-type: none">○ Definir la estrategia de evaluación de productos de software.○ Definir los tipos de pruebas a ejecutar a partir de la solicitud de evaluación.○ Definir las responsabilidades de las partes.○ Definir los recursos técnicos para el desarrollo de la evaluación.○ Definir el cronograma para la ejecución de la evaluación. <p>Ejecutar evaluación de productos de software.</p> <ul style="list-style-type: none">○ Desarrollar la estrategia de evaluación de productos de software prevista.○ Ejecutar los tipos de pruebas planificados.○ Velar por el cumplimiento de las responsabilidades de los involucrados.○ Velar por el uso de los recursos técnicos previstos para el desarrollo de la evaluación.○ Dar seguimiento al cronograma de ejecución de la evaluación. <p>Evaluar el cumplimiento de los Criterios de Criticidad durante la ejecución del proceso.</p> <ul style="list-style-type: none">○ Proponer la detención temporal o definitiva del proceso de evaluación a partir del cumplimiento de algún criterio de criticidad. <p>Gestionar el expediente de la evaluación de productos de software.</p> <ul style="list-style-type: none">○ Crear el expediente de la evaluación de productos de software.○ Actualizar el expediente de la evaluación de productos de software. <p>Gestionar las no conformidades que se detectan durante la ejecución de la evaluación.</p> <ul style="list-style-type: none">○ Registrar las no conformidades, ya sea a través de informes de no conformidades o del sistema GESPRO para la gestión de no conformidades.○ Actualizar el estado de las no conformidades para el caso de la ejecución de una evaluación de productos de software con modalidad de prueba de liberación. <p>Planificar intercambios con los clientes para el desarrollo de la evaluación.</p> <ul style="list-style-type: none">○ Preparar el plan de evaluación para la prestación del servicio.○ Preparar el acta de liberación o dictamen técnico según corresponda.○ Preparar el informe de evaluación del proceso de evaluación. <p>Desarrollar intercambios con los clientes durante el desarrollo de la evaluación.</p> <ul style="list-style-type: none">○ Presentar el plan de evaluación que regirá la prestación del servicio.○ Actualizar el plan de evaluación.○ Presentar el acta de liberación o dictamen técnico según corresponda.○ Presentar el informe de evaluación del proceso. <p>Mantener informado a los involucrados e interesados sobre el desarrollo de la evaluación.</p> <ul style="list-style-type: none">○ Informar sobre el avance de la evaluación a la administración cuando se le solicite.○ Informar constantemente al cliente sobre el inicio y fin del desarrollo de cada etapa de la estrategia de pruebas durante la ejecución del proceso de evaluación.○ Notificar de la necesidad de intercambiar de forma presencial con el cliente. <p>Gestionar la ejecución de las pruebas o de la evaluación del (de los) artefacto(s) o producto(s) en el LIPS.</p> <ul style="list-style-type: none">○ Notificar al cliente la planificación de estas actividades para solicitar su presencia.○ Favorecer las herramientas necesarias para la ejecución de las pruebas o de la evaluación en el LIPS.○ Presenciar la ejecución de la evaluación en el LIPS.○ Apoyar las actividades que se realicen durante la ejecución de la evaluación en el LIPS.
Planificador
<p>Gestionar la disponibilidad de los especialistas para asumir evaluaciones de productos de software.</p> <ul style="list-style-type: none">○ Mantener actualizado un balance de carga con los procesos que asume cada especialista del DEPSW como Coordinador del servicio. <p>Planificar la realización de reuniones de inicio a partir de las solicitudes de evaluación de productos de software que son aceptadas.</p>

<ul style="list-style-type: none">○ Mantener un plan de reuniones de inicio actualizado con los datos de la ejecución de los intercambios. <p>Planificar la evaluación de los artefactos correspondientes a los procesos de evaluación en ejecución en el LIPS.</p> <ul style="list-style-type: none">○ Mantener una planificación sistemática de los turnos en el LIPS disponibles para la evaluación de los artefactos o productos. <p>Brindar información actualizada y pertinente sobre las actividades de planificación que gestiona.</p> <ul style="list-style-type: none">○ Informar a la administración sobre la disponibilidad de especialistas para asumir procesos de evaluación de productos de software.○ Notificar las fechas de ejecución de las reuniones de inicio.○ Informar a los coordinadores de evaluaciones para ejecutar las actividades de pruebas o evaluación de los artefactos en el LIPS.
Probador
<p>Desarrollar la ejecución de la evaluación de productos a partir del trabajo asignado a su puesto de trabajo en el LIPS.</p> <ul style="list-style-type: none">○ Interactuar con el (los) artefacto(s) o producto(s) que le(s) corresponde evaluar.○ Ejecutar los casos de prueba o aplicar las listas de chequeo según corresponda.○ Detectar no conformidades durante la ejecución de los casos de pruebas o de la aplicación de las listas de chequeo.○ Registrar en un informe de no conformidades o a través del sistema GESPRO, las no conformidades detectadas.○ Clasificar las no conformidades detectadas (Góngora Rodríguez et al. 2009).
Ingeniero de pruebas
<p>Diseñar el tipo de prueba planificado por el coordinador de la evaluación, a partir de la solicitud de evaluación de productos de software.</p> <ul style="list-style-type: none">○ Entender las especificidades del (de los) artefacto(s) o producto(s) sometido(s) a evaluación.○ Elaborar la estrategia de pruebas a seguir para ejecutar el tipo de prueba especificado.○ Actualizar o preparar las herramientas necesarias, ya sean automatizadas o manuales, para la ejecución del tipo de prueba en cuestión. <p>Participar en la ejecución del tipo de prueba diseñado durante la ejecución de la evaluación.</p> <ul style="list-style-type: none">○ Ejecutar el tipo de prueba diseñado a partir del uso de las herramientas definidas.○ Asistir al coordinador de la evaluación en la ejecución del tipo de pruebas diseñado.
Gestor de la configuración
<p>Gestionar las herramientas de control de versiones (SVN para el caso del LIPS).</p> <ul style="list-style-type: none">○ Mantener actualizada la estructura del repositorio.○ Mantener actualizados los usuarios y sus respectivos permisos para la interacción con el repositorio.○ Brindar soporte al SVN durante la interacción con el repositorio, de los roles definidos. <p>Gestionar las herramientas de gestión de no conformidades (GESPRO en el LIPS).</p> <ul style="list-style-type: none">○ Crear proyectos en el GESPRO a partir de las solicitudes que han sido aprobadas y planificadas.○ Mantener actualizados los usuarios y sus respectivos permisos para la interacción con el sistema de gestión de no conformidades.○ Brindar soporte al sistema GESPRO durante la interacción con el sistema, de los roles definidos. <p>Mantener actualizado el registro de evaluaciones de productos de software del departamento.</p> <ul style="list-style-type: none">○ Definir y mantener un sistema de salvadas de los expedientes de las evaluaciones, de los servicios de evaluación de productos de software que se han prestado en el departamento.
Gestor de entornos de pruebas
<p>Administrar los servidores para los entornos de pruebas de las evaluaciones de productos de software en el departamento (Servidores virtuales en el LIPS).</p> <ul style="list-style-type: none">○ Preparar los servidores para el montaje de los entornos de pruebas.○ Mantener salvadas de todos los entornos de pruebas que se utilizan en el desarrollo de las evaluaciones de productos de software. <p>Brindar soporte a los servidores para los entornos de pruebas de las evaluaciones de productos de software.</p>

- Mantener el acceso restringido a los servidores para la actualización de los entornos de pruebas.
- Mantener registro del acceso a los servidores para la actualización de los entornos de pruebas

El rol **Gestor de la configuración** es el encargado de la administración de las herramientas de gestión para el desarrollo de las evaluaciones de productos de software. Su responsabilidad está bien determinada y constituye el soporte, desde el punto de vista de infraestructura tecnológica, para la prestación de servicios en el departamento. Por su parte, el rol **Gestor de entornos de pruebas**, al igual que el Gestor de la configuración descrito anteriormente, tiene una responsabilidad definida que constituye soporte tecnológico para la prestación del servicio de evaluación de productos de software en el departamento. Su responsabilidad está enmarcada, como su nombre lo indica, en la gestión y administración de los servidores para el montaje de los entornos de pruebas que corresponden a cada nuevo proceso de evaluación.

Hasta este momento se han relacionado los roles que se agrupan entre los roles administrativos y los roles ejecutivos, que constituyen nueve de los diez propuestos por el autor. El décimo es el rol **Cliente**. La peculiaridad de este rol radica en que puede ser asumido por cualquiera de los miembros de los proyectos de desarrollo de software, que solicitan el servicio de evaluación de productos de software a CALISOFT. Estos serán definidos en función de la actividad del proceso en cuestión. Sus responsabilidades se describen en la siguiente tabla (Tabla 4).

Tabla 4. Responsabilidades del rol Cliente.
Fuente: Elaboración propia.

Cliente
Participar en la planificación de la evaluación de productos de software. <ul style="list-style-type: none">○ Revisar y aprobar la estrategia de evaluación propuesta.○ Revisar y aprobar los tipos de pruebas planificados.○ Reconocer su responsabilidad en al desarrollo del proceso de evaluación.○ Proveer información pertinente para la definición de los recursos técnicos para el desarrollo de la evaluación.○ Revisar y aprobar el cronograma para la ejecución de la evaluación.
Participar en la ejecución de la evaluación de productos de software. <ul style="list-style-type: none">○ Velar por el cumplimiento de la estrategia de evaluación planificada.○ Velar por el desarrollo de todos los tipos de pruebas previstos.
Proveer la información solicitada para el desarrollo de la evaluación de productos de software. <ul style="list-style-type: none">○ Facilitar y participar en la preparación de las herramientas para la ejecución de los tipos de pruebas planificados.○ Mantener actualizado el expediente de la evaluación con la información pertinente que se le solicite.
Gestionar las no conformidades que se detectan durante la ejecución de la evaluación para el caso de la modalidad liberación. <ul style="list-style-type: none">○ Cumplir con el cronograma para la gestión de las no conformidades.
Participar en los intercambios desarrollados con el coordinador de la evaluación. <ul style="list-style-type: none">○ Informar de manera pertinente cualquier inconveniente que presente para participar en los intercambios con el coordinador de la evaluación.○ Participar en la ejecución de las actividades de la evaluación en el LIPS.

Con la descripción de las responsabilidades del rol **Ciente**, se completó con la totalidad de los roles propuestos por el autor en la presente investigación. Para comprender mejor su relación con el proceso de evaluación de productos de software, se representa en el Anexo 8, la relación entre las actividades del proceso de evaluación con los roles propuestos.

2.2.2 ETAPA II. DEFINICIÓN DE LAS ACTIVIDADES DE SEGUIMIENTO Y CONTROL

Las actividades de seguimiento y control constituyen el componente regulador del proceso de evaluación de productos, en función de una prestación eficiente del servicio. Como parte de la estrategia, definen el cómo de la gestión del proceso, para identificar a tiempo las posibles desviaciones que se puedan presentar y trabajar entonces en una temprana corrección.

Para esta definición, se identificaron tres momentos que centrarán todas las actividades de seguimiento y control. El primer momento es el denominado **Monitoreo**, basado en la ejecución de dos actividades: el seguimiento de la ejecución de los procesos y el control de la calidad. Un segundo momento corresponde al **Análisis**, en el cual, partiendo de las alertas determinadas en el paso anterior, permitirá analizar las causas que están incidiendo en la desviación. A partir de aquí, se podrán identificar entonces las acciones correctivas a desarrollar, lo que corresponde al momento de **Interacción**. En la Figura 13 se realiza una representación de estas actividades de seguimiento y control que complementan la estrategia.



Figura 13. Actividades de seguimiento y control.
Fuente: Elaboración propia.

En la etapa de Monitoreo, se proponen dos actividades fundamentales, cada una con sus objetivos bien definidos. La primera actividad, el Seguimiento, está enfocada a evaluar la ejecución del proceso de evaluación. Su objetivo es identificar cualquier desviación relacionada con el tiempo de ejecución de la evaluación, a partir de la identificación de los hitos de ejecución. La segunda actividad, que se ejecuta de

manera simultánea, es la de Control de calidad, la cual está orientada a evaluar la calidad del proceso de pruebas. Su objetivo está orientado a medir la eficiencia en la detección de no conformidades durante la ejecución de las pruebas.

A continuación se describirá la etapa de Monitoreo, detallando las dos actividades que la componen: Seguimiento y Control de calidad.

ACTIVIDADES DE SEGUIMIENTO

La actividad Seguimiento parte de la definición de los hitos del proceso de evaluación del producto de software. Para su determinación, se analizaron las actividades del proceso de pruebas propuesto por el autor de la presente investigación, para identificar así los estados por los que puede transitar la ejecución de una evaluación de productos de software.

De los cuatro subprocesos definidos en el proceso propuesto por el autor de la presente investigación, se toma el subproceso Ejecutar evaluación de productos de software, ya que constituye el momento en que se desarrolla la evaluación según la planificación efectuada en la etapa anterior. A partir de dicha planificación es que se podrá comprobar la correcta ejecución del proceso, darle seguimiento. Por lo anterior, es preciso tomar como referencia el Plan de Evaluación del Producto y en este, la estrategia de pruebas y el cronograma de la evaluación específicamente.

Teniendo en cuenta las actividades que se desarrollan en la ejecución de la evaluación de productos, los estados por los que transita la evaluación que se definieron son:

- Pruebas exploratorias iniciales.
- Primera iteración.
- Espera de respuesta de las no conformidades de la primera iteración.
- Actualización del entorno de pruebas para la regresión de la primera iteración.
- Pruebas de regresión de la primera iteración.
- Segunda iteración.
- Espera de respuesta de las no conformidades de la segunda iteración.
- Actualización del entorno de pruebas para la regresión de la segunda iteración.
- Tercera iteración.
- Espera de respuesta de las no conformidades de la tercera iteración.
- Actualización del entorno de pruebas para la regresión de la tercera iteración.
- Pruebas de regresión de la tercera iteración.
- Prueba de integración.
- Espera de respuesta de las no conformidades de la integración.

- Actualización del entorno de pruebas para la regresión de la integración.
- Pruebas de regresión de la integración.
- Prueba final.

En función de la estrategia de pruebas definida en el Plan de Evaluación del Producto y teniendo en cuenta la modalidad en la que se presta el servicio, se podrán determinar los estados por los que debe transitar la evaluación. Para el caso de los servicios en modalidad de liberación, la evaluación puede transitar por todos los estados definidos anteriormente, si es que en la estrategia, en correspondencia con las características del producto, se precisa realizar pruebas de integración y/o prueba final. Por otro lado, está el caso de la modalidad de evaluación de productos de software, en la cual se efectúa una sola iteración, y de ser preciso, se desarrollan pruebas de integración.

Las estrategias de prueba definidas en el plan de evaluación, parten de la realización de una PEI. Luego, en dependencia del tipo de artefacto, si se trata de documentación o aplicación, se desarrollan una prueba de sistema y/o revisión técnica, según corresponda. En cualquiera de los casos, se define el orden de las actividades que se van a desarrollar para lograr la completa evaluación del artefacto. Finalizada esta etapa, se desarrollan las pruebas de integración y/o prueba final, lo cual dependerá de la complejidad del artefacto en cuestión. Se debe tener en cuenta que la prueba de integración tiene el objetivo de identificar los errores introducidos por la combinación de programas probados unitaria y/o modularmente, teniendo en cuenta las dependencias establecidas entre los componentes de software, el correcto funcionamiento de los procesos que informatiza el sistema y que se hayan logrado las especificaciones de diseño. La prueba final por su parte, tiene el objetivo de verificar el correcto funcionamiento del sistema, a partir de comprobar la solución de alguna no conformidad detectada en la tercera iteración y de verificar el cumplimiento de la totalidad de los requisitos definidos para la evaluación.

El cronograma de pruebas realizado por el Coordinador de la evaluación, reflejado también en el plan de evaluación, registrará la planificación de las actividades desde el inicio hasta el fin de la evaluación de productos. En el mismo se especifican las personas involucradas, el responsable del desarrollo de cada actividad, la fecha prevista y las observaciones necesarias para su comprensión (Anexo 9). Esta planificación va a tener en cuenta la estrategia de pruebas a desarrollar, a partir del tipo de artefacto en cuestión.

Teniendo en cuenta los hitos definidos y el cronograma de pruebas establecido para la realización de cada evaluación, se podrá desarrollar entonces el seguimiento a cada proceso. Para el mismo se propone la utilización del sistema de partes semanales establecido en el DEPSW, el cual se creó con el objetivo de mantener informada a la dirección y a los clientes sobre el estado de avance de los procesos. En cada parte se notifica el estado de la evaluación, que se corresponde con los hitos definidos anteriormente. En el DEPSW existe un registro de todos los partes semanales elaborados, pero no existe evidencia de que hayan sido utilizados en función de la gestión del proceso de evaluación de productos de software. Aprovechando entonces este sistema de partes como fuente potencial de información, se propone evaluar el avance de los procesos de evaluación en ejecución contra los cronogramas de pruebas.

Cada semana, un proceso de evaluación determinado debe transitar de un estado a otro estado superior. Cuando se observa que de una semana a otra no ha variado el estado, se genera entonces una alerta que servirá de premisa para pasar del momento de **Monitoreo** al de **Análisis** (Figura 13).

ACTIVIDADES DE CONTROL DE CALIDAD

La actividad Control de calidad está asociada a la definición de un indicador para evaluar la eficacia en la detección de no conformidades. Para esto se precisa establecer una serie de premisas que se relacionan a continuación:

- En una iteración de pruebas se ejecutan todos los tipos de pruebas planificados.
- En una iteración de pruebas se da cobertura a la ejecución de todos los casos de pruebas o requisitos a evaluar.
- En un proceso de evaluación se planifican hasta tres iteraciones de pruebas.
- En cada iteración de pruebas son registradas todas las no conformidades detectadas.
- El inicio de la segunda y tercera iteración depende de la corrección o gestión de todas las no conformidades detectadas en la iteración anterior.

Si se establece Efc como la eficacia en la detección de no conformidades, TNC_n como el total de no conformidades detectadas en una iteración n y TNC_{n-1} para definir la cantidad total de no conformidades detectadas en una iteración anterior, se propone entonces calcular la eficacia a partir de:

$$Efc = \frac{TNc_n}{TNc_{n-1}}$$

donde

n: número de la iteración, tal que $2 \leq n \leq 3$, $n \in \mathbb{Z}$,

TNc_n : total de no conformidades detectadas en la iteración n,

TNc_{n-1} : total de no conformidades detectadas en la iteración n-1.

Efc puede tomar un valor que puede ser menor, igual o mayor que 1. En función del valor que tome se puede definir lo siguiente:

- $Efc < 1$, la detección de no conformidades es eficaz. Mientras más se acerque n al valor cero, más eficaz habrá sido la detección de no conformidades.
- $Efc \geq 1$, la detección de no conformidades no es eficaz.

Este indicador se evalúa en la segunda y tercera iteración durante la evaluación de productos de software. En este caso se realiza para la modalidad de liberación ya que es en la que se realiza más de una iteración de pruebas. Si luego de evaluar este indicador se obtiene que la detección de no conformidades no ha sido eficaz, se genera entonces una alerta para dar paso también al momento del **Análisis** (Figura 13).

ACTIVIDADES DE REVISIÓN

La actividad Revisión corresponde al segundo momento propuesto para las actividades de seguimiento y control, el **Análisis**. Para su ejecución se precisa que se haya generado una alerta en cualquiera de las dos actividades del momento **Monitoreo**. En la Figura 14 se representan las dos actividades que se proponen para desarrollar la revisión; realizar primeramente una recopilación y análisis de la información, para luego proponer acciones correctivas. Estas están en correspondencia con el proceso de seguimiento del servicio, propuesto por Sanz Esteban (Sanz Esteban 2012).

Dado el caso que se haya detectado una desviación en el cronograma de ejecución de la evaluación o que el proceso de detección de no conformidades no esté siendo debidamente eficaz, se requiere realizar un análisis de los equipos de pruebas o de los recursos asignados para enfrentar la evaluación. Partiendo de que el motivo de la revisión sea la desviación en el cronograma de pruebas, se procede en este momento a analizar las causas de tal desviación. Pueden estar incidiendo desde los recursos

materiales asignados, puestos de trabajo, hasta el incumplimiento de las responsabilidades de las partes involucradas.

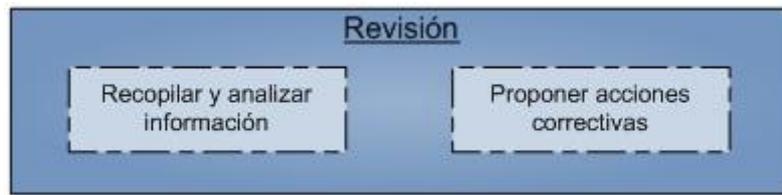


Figura 14. Actividades asociadas a la revisión.
Fuente: Elaboración propia.

Si se determina que la causa está asociada a la disponibilidad de los recursos, ya sean humanos o materiales, se realiza una revisión de la posibilidad de reasignar los recursos. Se evalúa el balance de carga de los especialistas del departamento y la disponibilidad de puestos de trabajo o turnos en el LIPS, considerando la posibilidad de reforzar el proceso de evaluación.

Si la alerta que originó el desarrollo de la revisión está determinada por la ineficacia en la detección de no conformidades, se propone entrar a analizar la composición del equipo de pruebas. En este momento debe ser considerado el nivel de preparación de los probadores, el estado de las herramientas de pruebas, el nivel de asimilación del negocio del sistema que se está evaluando y si ha existido algún flujo o variación de la composición del equipo de pruebas.

Cualquiera que sea el caso, una vez efectuada la recopilación y análisis de la información, se determinan cuáles son las acciones correctivas que se pueden desarrollar para mitigar los efectos de cualquier desviación. Para esto, se propone que se elabore un informe detallado donde se reflejen el resultado del análisis efectuado y las acciones que se deben ejecutar. Este informe constituye la entrada para el desarrollo de la próxima etapa en las actividades de seguimiento y control, la **Interacción**.

Otro elemento importante a destacar es la evaluación en todo momento de los criterios de criticidad relacionados en la descripción del proceso propuesto por el autor de la presente investigación (Descripción del proceso Evaluación de Productos). Estos permiten valorar el estado de la evaluación, a partir de la responsabilidad del cliente con la gestión de las no conformidades y de la calidad del producto evaluado. A partir de dicha evaluación, se proponen un conjunto de decisiones sobre la pertinencia de continuar con la evaluación de productos de software.

REUNIÓN DE SEGUIMIENTO

La Reunión de seguimiento corresponde al momento de **Interacción**, definido para el seguimiento y control del proceso de evaluación de productos. Para su desarrollo es preciso que se haya elaborado un informe de seguimiento, resultado del momento anterior.

A partir del informe, se propone que se reúnan los involucrados en el proceso de evaluación, para dar curso a las actividades correctivas propuestas. En este momento se comunican las desviaciones, sus causas, a partir de las cuales se deben desarrollar las acciones propuestas.

En este momento de la reunión se decide cómo dar seguimiento a las correcciones propuestas. Se recomienda que se detallen las responsabilidades y fechas de cumplimiento, para así controlar que las correcciones se ejecuten de manera satisfactoria y se pueda completar el proceso de evaluación de productos de software.

En toda la etapa de ejecución de las actividades de seguimiento y control, el rol Administrador de la calidad propuesto en la descripción del proceso de evaluación de productos, es el máximo responsable de su ejecución. Es el ente principal que garantiza el seguimiento a la prestación del servicio de evaluación de productos, de conjunto con el Coordinador del servicio. Ambos interactúan durante el desarrollo de estas actividades con el Coordinador del servicio, el Planificador y el Cliente.

2.3 CONCLUSIONES PARCIALES DEL CAPÍTULO

Con el planteamiento de la estrategia se arribó a las siguientes conclusiones:

1. La estrategia planteada se basa en la ejecución de dos etapas. La primera es el establecimiento del proceso Evaluación de Productos de Software y la segunda la definición de actividades de seguimiento y control de dicho proceso.
2. El proceso Evaluación de Productos de Software está orientado a la ejecución de cuatro subprocesos que transitan por la gestión, planificación, ejecución y cierre de las evaluaciones.
3. Se proponen diez roles y se definen sus responsabilidades, representándolos en dos grupos fundamentales orientados a su función: los roles administrativos y los roles ejecutivos. El décimo rol es el Cliente, en el centro de la representación, no solo como la razón del servicio sino como ente activo en el desarrollo del proceso.
4. La etapa de monitoreo y control se sustenta en el desarrollo de actividades de monitoreo, análisis e interacción.

CAPÍTULO III. VALIDACIÓN DE LA ESTRATEGIA

3.1 INTRODUCCIÓN

En el presente capítulo se muestran los resultados de la validación de la estrategia. Se parte de su implementación parcial, con el establecimiento del proceso Evaluación de Productos de Software, evaluando su impacto en el LIPS. Para esto se desarrolló un proceso de intercambio y capacitación, que permitió, a través de una encuesta al 69% de los especialistas, obtener una valoración del proceso. Luego se procedió a evaluar la eficiencia en la prestación del servicio, con la implementación del proceso, tomando en consideración la variable: tiempo. Con tal objetivo se seleccionó una muestra intencional no probabilística de los procesos de evaluación de productos de software que se ejecutaron en el laboratorio antes y después de la propuesta. Para dar continuidad a la validación se somete la estrategia a la valoración de un grupo de expertos para finalmente proceder a la triangulación de los resultados, evaluando la propuesta desde varias perspectivas.

3.2 EVALUACIÓN DEL ESTABLECIMIENTO DEL PROCESO EVALUACIÓN DE PRODUCTOS DE SOFTWARE

A partir de la definición de la estrategia se procedió a desarrollar un intercambio con los especialistas del LIPS, para considerar sus criterios sobre la conformidad con el proceso Evaluación de Productos de Software. Se desarrolló un encuentro en dos sesiones con los grupos que componen el departamento. En la primera sesión se convocaron a los especialistas del Grupo de Ingeniería de Pruebas (GIPS) y en la siguiente a los especialistas de los grupos Laboratorio Industrial de Pruebas de Software (LIPS) y Seguimiento y Control de la Calidad de los Servicios (SCCS), para completar un total de 48 miembros del departamento que debían asistir. Se logró reunir a un total de 33 especialistas, lo que representa una muestra significativa del 69%.

El principal objetivo del encuentro, además de presentar el proceso y las responsabilidades de cada rol definido, fue el de conocer si los involucrados consideraban que el proceso debía ser cambiado y en qué medida. Para esto se realizó un cuestionario de tres preguntas, las cuales se relacionan a continuación:

- ¿Considera que el proceso debe ser cambiado parcialmente, totalmente o no debe ser cambiado?
- ¿Qué considera propicio incluir en una próxima versión del proceso?
- ¿Considera que debe omitirse algo de lo presentado en una próxima versión del proceso? ¿Qué?

La conformidad con el proceso diseñado se representa en el siguiente gráfico, a partir de las respuestas de la primera pregunta.

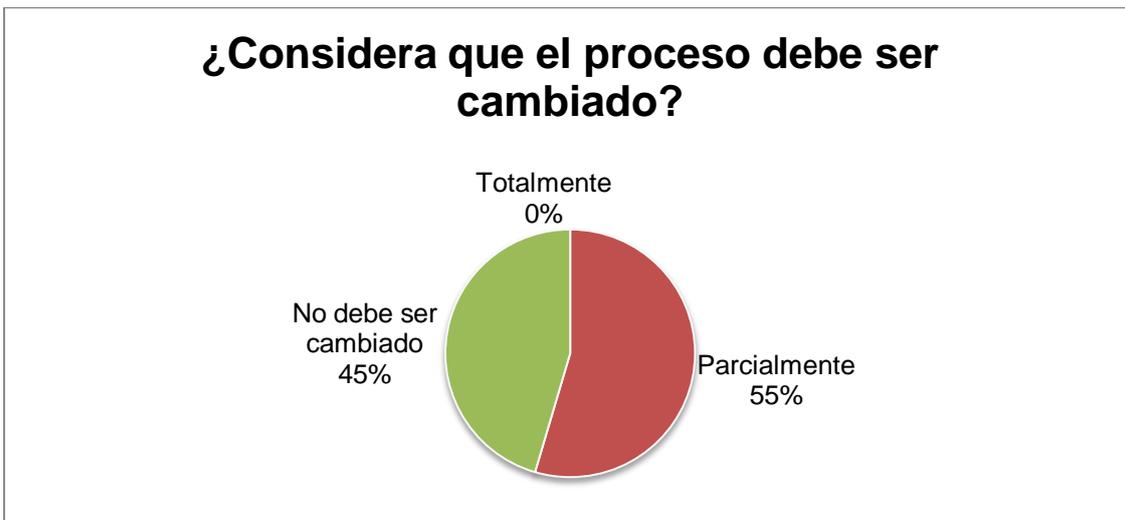


Gráfico 1

Resultados de la pregunta 1 de la encuesta de evaluación del proceso. Fuente: Elaboración propia.

Analizando las respuestas de los encuestados a esta pregunta, destaca que ninguno considera que el proceso deba ser cambiado en su totalidad. Esto evidencia que en alguna medida todos están de acuerdo con el proceso, están de acuerdo en que responde a las necesidades del departamento de prestar un servicio de evaluación de productos de software.

Del total de los encuestados, el 45% considera que no debe ser cambiado el proceso. Por otro lado, un 55% consideran que el proceso debe ser cambiado parcialmente. Esto se debe a que realizaron alguna recomendación, o que señalaron elementos que debían agregar, omitir o tener en cuenta para la mejora del proceso propuesto.

Para conocer qué considera ese porcentaje que se debe agregar o suprimir del proceso diseñado, se pueden tomar como elementos que enriquecen la información, las respuestas a la segunda y tercera interrogantes de la encuesta. Entre los criterios de mayor incidencia sobre lo que se debe omitir, destaca que con el proceso se genera demasiada documentación. Cuatro personas coinciden con este criterio especificando que atenta contra el tiempo de ejecución de las evaluaciones. Del mismo modo, existen tres criterios que coinciden en que el Diario de la Evaluación dentro del artefacto Historial de Pruebas es innecesario y que debe omitirse. También existen dos criterios que coinciden en que el Taller de diseño no es preciso, del mismo modo que el Historial de Pruebas a criterio de uno de los encuestados.

Estas consideraciones denotan que se debe prestar especial atención a la gestión documental en la gestión de los procesos de evaluación. La automatización de este proceso contribuiría en gran medida a disminuir el esfuerzo empleado en la ejecución de las evaluaciones y favorecería de manera positiva a la gestión eficiente del servicio Evaluación de Productos de Software. En el caso del artefacto Historial de Pruebas, es preciso contar con un registro de lo que va sucediendo con la evaluación para poder controlar y dar seguimiento a la ejecución de las evaluaciones con la presente propuesta. Se acordó en el intercambio con los especialistas que el Diario de la Evaluación sí constituye un trabajo exhaustivo y que para evitarlo solo se registrarían las incidencias que provocaran un desvío o retraso en el cumplimiento del cronograma inicialmente aprobado para cada evaluación. De esta manera, no se perdería información relevante que constituye la base para el desarrollo de las actividades de Seguimiento y Control. Con relación al Taller de diseño se acordó que este se realizaría solo en casos necesarios, en dependencia de las necesidades para la evaluación.

Existen otros criterios aportados por los encuestados, que constituyen recomendaciones a considerar. Los criterios emitidos se relacionan a continuación:

- El proceso descrito y los subprocesos que lo complementan, deben especificarse de manera que resulte más entendible. Existen elementos que no se explican y que deben aparecer en guías.

En el proceso se hace referencia a algunas guías que recogen el proceder de cada rol en cada momento durante el proceso. Se describen cuáles son las tareas y responsabilidades de todos los involucrados en cada una de las actividades del proceso.

- Se recomienda que no se acepte ninguna solicitud que no tenga los datos del entorno de pruebas correctos. Que de no ser así, se detenga o aborte el proceso a partir de un criterio de criticidad.

Se planteó que en muchos casos las solicitudes llegaban a una Reunión de Inicio sin tener los datos del entorno de pruebas correctos o acordes con la realidad. Se debe considerar en ese momento si con los datos proporcionados se puede efectuar la evaluación o si se debe actualizar la solicitud y detener el proceso por una semana, hasta que se tenga correctamente especificado el entorno de pruebas que se debe utilizar.

- Se recomienda que exista mayor claridad en el orden en que se hace el Taller de diseño y el montaje del sistema.

En el proceso se propone que primero se realice el Montaje del entorno de pruebas y posteriormente el Taller de diseño. El entorno de pruebas se puede ajustar en cualquier momento de la evaluación a solicitud del Coordinador de la evaluación, previo acuerdo con el rol Cliente.

- Se recomienda incluir la revisión de los expedientes de prueba de manera periódica.

A esta recomendación se le da salida con la ejecución de las actividades de Seguimiento y Control que se proponen en el Capítulo 2.

- Se recomienda analizar los criterios de criticidad en la Reunión de Inicio.

Los criterios de criticidad se evalúan en cualquier momento durante la ejecución del proceso de evaluación, desde el inicio y hasta el fin. En la Reunión de Inicio, además de evaluar el cumplimiento de estos criterios, se dan a conocer al Cliente para que sea de su conocimiento y se registran o anexan al Plan de Evaluación del Producto.

Existen otro grupo de especialistas que recomiendan que se tengan en cuenta la cantidad de no conformidades detectadas en las evaluaciones, para definir el costo en función de poner una tarifa al servicio. Cuando se defina el costo de los servicios, recomiendan no abortar las pruebas ya que sería tiempo y esfuerzo invertido. Otro criterio recogido es que debía existir solamente el servicio Evaluación de Productos de Software sin la modalidad de Pruebas de Liberación. Estos planteamientos se han clasificado como Otros, ya que están condicionados a un entorno diferente, donde sea favorable su implementación (Ver Gráfico 2).

Como resultado del cuestionario aplicado, se puede observar que las modificaciones propuestas no afectan el flujo del proceso, ni implican cambios en las actividades o en los roles definidos. Los cambios propuestos no son significativos y están clasificados en el orden de las sugerencias. Se puede concluir que el proceso es correcto y que corresponde con las necesidades del departamento para la prestación del servicio Evaluación de Productos de Software.

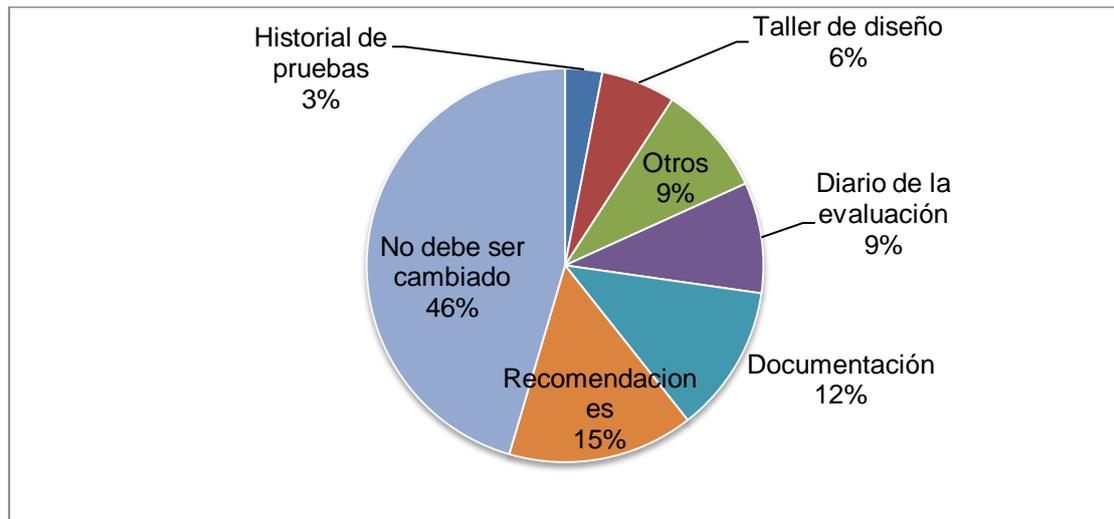


Gráfico 2

Resultados de las preguntas 2 y 3 de la encuesta de evaluación del proceso. Fuente: Elaboración propia.

3.3 ANÁLISIS DE LA EFICIENCIA DEL SERVICIO EVALUACIÓN DE PRODUCTOS

Partiendo de la hipótesis planteada para el desarrollo de la investigación, se propone desarrollar un análisis del comportamiento de la variable: eficiencia del servicio de evaluación de productos de software. Para esto se utilizó el método de experimentación a partir de una medición de su comportamiento antes y después de diseñada e implementada la estrategia.

La palabra eficiencia proviene del latín *efficientia* que quiere decir: acción, fuerza, producción. Se define como la capacidad de disponer de alguien o de algo para conseguir un determinado efecto. Según Bravo Carrasco, la eficiencia es un índice que se obtiene de la división entre producción (output) y recursos (input). ¿Qué recursos? Horas-hombre, además de infraestructura, insumos y costos generales. En el análisis horas-hombre ya aparece el tiempo, aunque de forma indirecta (Bravo Carrasco 2010).

A los efectos de la presente investigación se definió la eficiencia como la relación entre el tiempo de ejecución de las evaluaciones de productos de software y los recursos empleados, dígame cantidad de especialistas y recursos materiales (puestos de trabajo). Los recursos materiales, en las mediciones realizadas, se mantenían invariables, por lo tanto, se realizó el análisis en función del tiempo empleado.

Para evaluar la duración de las evaluaciones de productos de software se seleccionó una muestra intencional no probabilística. Se tuvieron en cuenta los procesos iniciados en los meses de octubre y noviembre del año 2012, antes de aplicar la estrategia y octubre noviembre de 2013, inmediatamente después de establecido el proceso de evaluación de productos de software. En cada muestreo se representa la relación

entre la fecha planificada de finalización de las evaluaciones y la fecha real de finalización. Los resultados de la primera medición se muestran a continuación.



Gráfico 3
Evaluaciones iniciadas en octubre – noviembre 2012. Fuente: Elaboración propia.

En los meses de octubre – noviembre de 2012 (Gráfico 3) se iniciaron un total de 13 evaluaciones de productos. De estas, solo una se ejecutó en el tiempo planificado, lo que evidencia un 92.3% de incumplimiento con los cronogramas aprobados en los planes de evaluación y por lo tanto, un alto índice de deficiencia en la ejecución de la evaluación de productos de software.

Se realizó un análisis de las principales causas que incidieron en los incumplimientos de los cronogramas de evaluación y destacan fundamentalmente tres de ellas. La primera, teniendo en cuenta el nivel de incidencia, resultó ser la priorización de otra evaluación que requirió prescindir de los recursos humanos por espacio de una semana, para trasladarlos a dicha evaluación. Con esta causa se vieron afectados un total de cinco procesos de evaluación. La segunda lo constituyó la demora en solucionar las no conformidades detectadas por parte de los clientes, para dar paso a la siguiente iteración, destacando cuatro evaluaciones afectadas por esta causa. La tercera que incidió en el alto índice de incumplimiento de los cronogramas de evaluación, fueron los problemas presentados con la disponibilidad de los entornos de prueba, ya sea por no realizar el montaje en el momento planificado o por problemas con los servidores montados en el entorno de los proyectos. Cuatro fueron los proyectos que registraron esta afectación. En menor medida pero no menos

importante, se detectaron otras causas como son algunos problemas con la actualización de los artefactos para la evaluación; incongruencia con los diseños de casos de pruebas y los artefactos; considerable número de no conformidades de iteraciones anteriores no resueltas y cambios en la aplicación por solicitud del cliente.

Una vez establecido el proceso Evaluación de Productos de Software a partir de su definición, se desarrollaron un conjunto de actividades en función de lograr que fuera asumido este por los especialistas en el LIPS. Luego se efectuó una segunda medición para comprobar el comportamiento de la variable: eficiencia del servicio de evaluación de productos. Para esta medición se tomó, del mismo modo que en la anterior, una muestra intencional no probabilística, en la que se tuvieron en cuenta los proyectos iniciados en los meses de octubre y noviembre del año 2013.

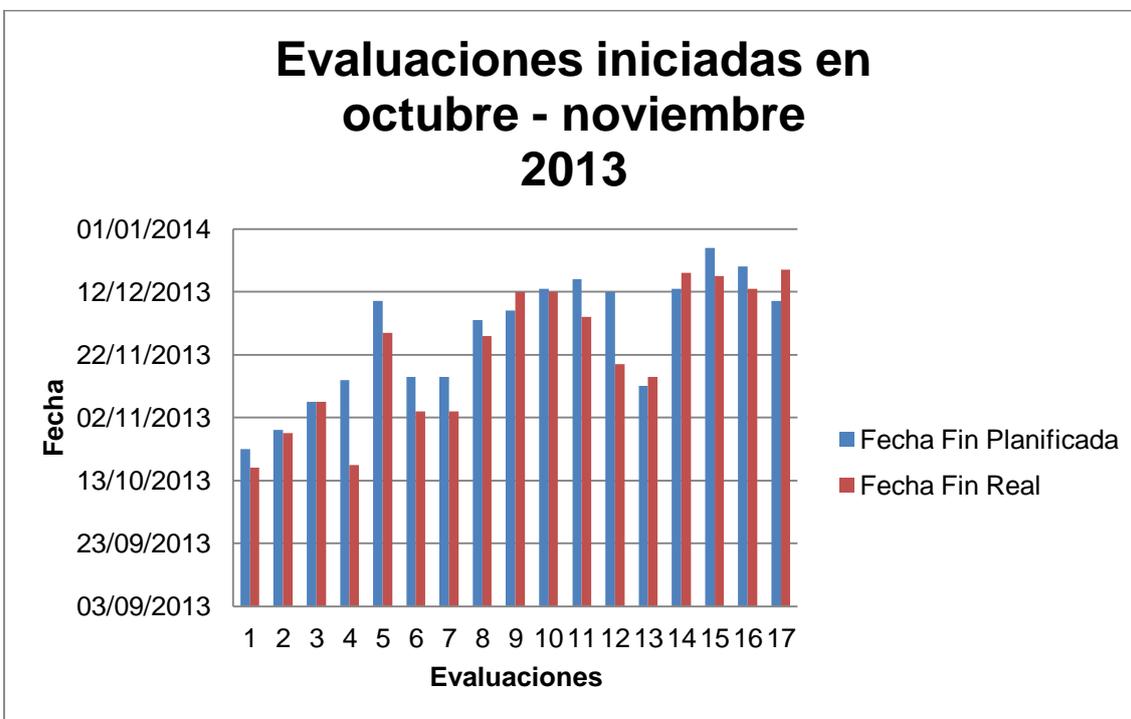


Gráfico 4
Evaluaciones iniciadas en octubre – noviembre 2013. Fuente: Elaboración propia.

En los meses de octubre – noviembre de 2013 (Gráfico 4) se iniciaron un total de 17 evaluaciones de productos de software. De estas evaluaciones, solo cuatro no se ejecutaron en el tiempo planificado, lo que representa una disminución sustantiva del incumplimiento de los cronogramas del 92.3% en igual periodo del año anterior, a solo un 23.5% por ciento en el 2013.

Se realizó un análisis de las dificultades que incidieron en que los cuatros procesos de evaluación no cumplieran con los cronogramas establecidos. En esta oportunidad, el elemento de mayor incidencia lo constituyó las dificultades presentadas durante las

pruebas en el LIPS, tales como los problemas de conectividad y de fluido eléctrico, que provocaron que no se pudieran completar y que por lo tanto, se atrasaran los cronogramas. Durante la ejecución de las pruebas en el LIPS, también se presentaron problemas con el entorno de pruebas, lo que destaca como una de las causas de los incumplimientos en la planificación. En menor medida subrayan otras incidencias como problemas de instalación en el montaje y/o actualización de los entornos de prueba. Cuando se planificaban estas tareas debían interrumpirlas en ocasiones por la carencia de algunas herramientas necesarias por parte de los proyectos. En algunos procesos también tuvieron lugar afectaciones por encontrarse los especialistas involucrados en otras evaluaciones, como es el caso de las aceptaciones de productos de software.

Para evaluar la eficiencia del servicio de evaluación de productos de software a partir de las mediciones anteriores, se consideró la razón existente entre la cantidad de procesos ejecutados y la cantidad de procesos ejecutados en tiempo, cumpliendo con los cronogramas. Se definió la siguiente fórmula:

$$Eficiencia = \frac{TPET}{TPP} \times 100$$

donde TPET representa el Total de Procesos Ejecutados en Tiempo y TPP representa el Total de Procesos Planificados.



Gráfico 5
Eficiencia del servicio Evaluación de Productos. Fuente: Elaboración propia.

Aplicando la fórmula anterior en la medición efectuada en el 2012, antes de la implementación de la estrategia, se obtuvo un valor de eficiencia del 7.69%. Este valor está asociado a una baja capacidad para disponer de los recursos utilizados en la

ejecución de la evaluación de productos, para lograr cumplir con los tiempos planificados. En la medición efectuada en el 2013, con el nuevo proceso de evaluación establecido, se obtuvo un valor de eficiencia del 76.47% (Gráfico 5). Esto denota un proceso de evaluación de productos de software más proactivo, que gestionó a tiempo los elementos que podían atentar contra el cumplimiento de los cronogramas. Las actividades de seguimiento y control fueron oportunas, viabilizando aquellas deficiencias que pudieran ir presentándose durante la ejecución del proceso. Se evidenció un proceso de evaluación de productos organizado, con un mayor grado de madurez, más eficiente. Se puede decir que aumentó también, en alguna medida, la capacidad de asumir procesos de evaluación de productos al aumentar la cantidad de procesos ejecutados de 13 a 17 en igual periodo de tiempo entre un año y otro. Esto demuestra un ligero aumento en la productividad, determinado por una mejor gestión de las evaluaciones y del tiempo.

Es preciso señalar que del mismo modo, se debe trabajar en brindar solución a las dificultades detectadas. Es necesario lograr planificaciones más ajustadas a la realidad. Se debe realizar un plan de mitigación de riesgos, a partir del conocimiento que tiene la institución de aquellos elementos de mayor incidencia que permita la toma de decisiones más oportuna y elevar más la eficiencia del proceso de evaluación de productos de software.

3.4 EVALUACIÓN DE LA ESTRATEGIA MEDIANTE VALORACIÓN DE EXPERTOS

Para validar la estrategia para elevar la eficiencia del servicio de evaluación de productos de software en el LIPS se tuvo en cuenta también el criterio de expertos los que fueron analizados a través de la escala de Likert. Para esto se seleccionó un grupo de expertos que consideraron una serie de cuestiones enfocadas a validar el cumplimiento de la hipótesis de la investigación.

3.4.1 SELECCIÓN DE LOS EXPERTOS

La selección de expertos fue una muestra intencional, que estuvo dirigida a la selección de profesionales con experiencia en pruebas de software y en gestión de procesos de evaluación de productos de software, dentro y fuera del ámbito nacional. Se convocó a especialistas del LIPS en el DEPSW de CALISOFT que tuvieran más de 5 años de experiencia y con categoría científica. Se tuvo en cuenta el criterio de profesionales de la Universidad Central de Las Villas “Martha Abreu” con experiencia en el desarrollo de investigaciones. Aceptaron participar además como expertos, especialistas y profesores de la Universidad de Costa Rica, de la Corporación Universitaria Remington de Medellín, así como del Instituto Antioqueño de

Investigaciones (IAI) en Colombia, todos estos con más de 15 años de experiencia en el tema.

Determinado el número posible de expertos, se realizó un análisis del coeficiente de conocimiento, el coeficiente de argumentación y el coeficiente de competencia mediante autovaloración de cada encuestado (Cabero Almenara y Barroso Osuna 2013) (Anexo 10). Esto permitió desestimar a un experto cuyo coeficiente de competencia resultó inferior al 0.8. De los siete expertos seleccionados, cuatro eran extranjeros y seis poseían la categoría científica de Máster. Los años de experiencia de los expertos osciló entre 3 y 25, con un 42.8% que poseían más de 15 años vinculados a la ingeniería, gestión y pruebas de software.

3.4.2 ESCALAMIENTO DE LIKERT

Para analizar la opinión de los expertos sobre la estrategia propuesta en la investigación, se utilizó la escala de Likert para evaluar el grado de concordancia con las ideas planteadas en función de valorar el cumplimiento de la hipótesis de la presente investigación. Esta escala permitía a los entrevistados emitir una valoración de Muy Adecuado (MA), Bastante Adecuado (BA), Adecuado (A), Poco Adecuado (PA) e Inadecuado (I). A partir de sus respuestas se calculó el nivel de coincidencia entre ellas a través del cálculo de un índice porcentual (IP) determinado por:

$$IP = \frac{(5 \times \% \text{ de MA}) + (4 \times \% \text{ de BA}) + (3 \times \% \text{ de A}) + (2 \times \% \text{ de PA}) + (1 \times \% \text{ de I})}{5}$$

La entrevista recogía las siguientes preguntas Anexo 11:

- ¿Qué opinión le merece la estrategia, considerando si con su aplicación se logrará gestionar el proceso de evaluación de productos en el Laboratorio Industrial de Pruebas de Software, para elevar la eficiencia del servicio de evaluación de productos de software?
- Valore las etapas definidas para el desarrollo de la estrategia, en función de lograr la gestión del servicio de evaluación de productos en el LIPS.
- ¿Cómo evaluaría el proceso definido teniendo en cuenta la gestión del proceso de evaluación de productos de software en el LIPS?
- ¿Considera que los roles definidos en la primera etapa de la estrategia, favorecen la gestión del proceso de evaluación de productos de software?
- ¿Cómo valora usted la definición de las actividades de seguimiento y control para la gestión del proceso de evaluación de productos de software en el laboratorio?

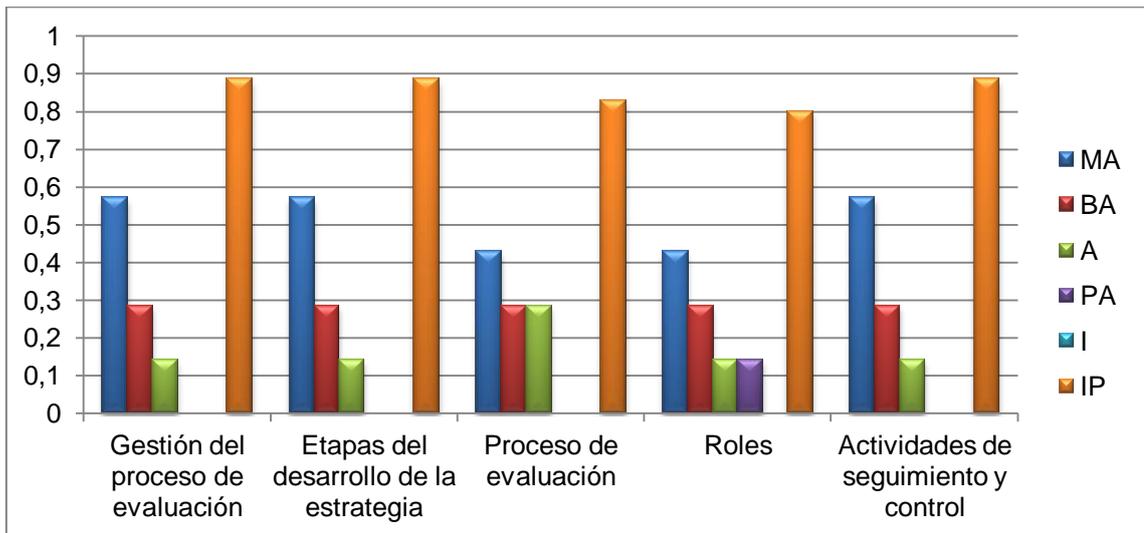


Gráfico 6
Resultado de la aplicación de la escala de Likert. Fuente: Elaboración propia.

En el Gráfico 6 se muestra el resultado del procesamiento realizado a partir del escalamiento de Likert. Esta evaluación evidencia que el índice porcentual de concordancia supera el 0.8 en todos los casos. Muestra por otra parte, que la estrategia es valorada en un alto grado como muy adecuada para lograr la gestión del proceso de evaluación de productos de software, a partir además de la valoración positiva de las etapas definidas, el proceso de evaluación, los roles definidos y las actividades de seguimiento y control. Los criterios de mayor índice porcentual de coincidencia fueron la gestión del proceso de evaluación, las etapas definidas para el desarrollo de la estrategia y las actividades de seguimiento y control, sobrepasando el 0.85%. Esto evidencia que los expertos coinciden en considerar que con la estrategia propuesta se logra gestionar satisfactoriamente el proceso de evaluación y que las etapas definidas en la estrategia se corresponden con esta intención. Las actividades de seguimiento y control tienen un alto nivel de aceptación también por los expertos, lo cual constituye un aporte importante de la investigación a la solución del problema inicialmente planteado.

3.5 TRIANGULACIÓN DE LOS RESULTADOS

La triangulación de los resultados de la investigación favorece el análisis desde diversos puntos de vista para tener un resultado más amplio del objeto en cuestión. El autor de la presente investigación coincide con la definición tomada de Silvio Donolo, quien dice que: “es la aplicación y combinación de varias metodologías de la investigación en el estudio de un mismo fenómeno” (Silvio Donolo 2009).

La modalidad de validación empleada más frecuentemente es la triangulación de métodos. Su fundamento radica en la idea de que los métodos son instrumentos para

investigar un problema y facilitar su entendimiento (Rodríguez Ruiz 2005). En consonancia con lo anterior, se utiliza en la presente investigación la triangulación intermétodos, para medir el grado de validez externa de los datos. Esta trata de comprobar que los resultados no son consecuencia de la utilización de un método particular, a partir del estudio del fenómeno mediante el empleo de métodos cuantitativos y cualitativos. Se realiza una evaluación que integra desde la implementación del proceso Evaluación de Productos de Software con la utilización de una encuesta, pasando por el estudio de casos y culminando con la validación por expertos y el escalamiento de Likert.

3.5.1. TRIANGULACIÓN DE MÉTODOS Y EVALUACIÓN DE LOS RESULTADOS.

Como resultado de la triangulación de métodos en la presente investigación, según los resultados tanto cuantitativos como cualitativos, se puede concluir que existe una correspondencia satisfactoria entre los resultados de la variable evaluada, que evidencia el alto grado de satisfacción y aceptación de los usuarios con el proceso Evaluación de Productos de Software propuesto. Queda de manifiesto además, el impacto directo en la eficiencia en la prestación del servicio, a partir de una disminución del tiempo de ejecución de las evaluaciones y por lo tanto, una gestión eficiente de los recursos humanos y tecnológicos. Todo esto, acompañado por la aplicabilidad de la estrategia, valorada a partir de expertos con un índice porcentual superior al 0.85 en la mayoría de los casos, evidencia la validez de la estrategia para elevar la eficiencia del servicio de evaluación de productos de software. Lo anterior se apoya en la valoración de que la hipótesis planteada en la investigación fue comprobada, lográndose cumplir los objetivos propuestos.

La implementación de la estrategia proveerá a la organización de un medio para reducir los gastos de operación en la prestación del servicio de evaluación de productos de software. Esto se logra al contar con un proceso definido, establecido y que al mismo tiempo es evaluado y monitoreado. El departamento será capaz de controlar el cumplimiento de los planes propuestos a favor del cumplimiento de los objetivos estratégicos anuales. Los recursos utilizados en la Evaluación de Productos de Software serán gestionados de manera que se evite la sobreexplotación y la subutilización y la carga de trabajo de las personas involucradas, se podrá balancear en correspondencia con la responsabilidad que adquieren en la ejecución del proceso y del nivel de especialización que posean en el desempeño de cada rol.

3.6 CONCLUSIONES PARCIALES DEL CAPÍTULO

A partir de los resultados de la validación de la presente estrategia de evaluación de productos se arribó a las siguientes conclusiones:

1. Mediante la implementación parcial de la estrategia, con el establecimiento del proceso Evaluación de Productos de Software, se pudo constatar a través de una encuesta, el impacto positivo del proceso, el evaluarse como correcto y considerarse en correspondencia con las necesidades del departamento para la prestación del servicio.
2. A través de la utilización de la estrategia en procesos pilotos de prestación del servicio, se evidenció un incremento significativo en la eficiencia por encima del 76%, dada por la reducción de los tiempos de ejecución y del cumplimiento con los cronogramas de evaluación.
3. La valoración de un conjunto de expertos seleccionados por su coeficiente de competencia en el ámbito de las pruebas de software y de la gestión de los procesos de evaluación de productos de software, demostró que la estrategia es altamente adecuada para lograr la gestión del proceso de evaluación de productos de software, considerando positivas las etapas de la estrategia definidas, el proceso de evaluación, los roles definidos y las actividades de seguimiento y control.
4. La triangulación de métodos utilizados para la evaluación de la estrategia en la presente investigación, constató la convergencia de los resultados obtenidos de la validación, evidenciando que la solución propuesta es adecuada y resuelve el problema de la investigación.

CONCLUSIONES

El desarrollo de la presente investigación permitió al autor arribar a las siguientes conclusiones:

1. Existe un amplio marco de modelos relacionados con la gestión de pruebas que aportan elementos que favorecen la adopción de una estrategia para elevar la eficiencia del servicio de pruebas de software, en una organización que funciona como una factoría de pruebas.
2. En el LIPS, con el análisis de las acciones implementadas para prestar un servicio de mayor calidad y de las principales deficiencias, se dieron las condiciones y la necesidad para organizar todas las iniciativas en una estrategia para elevar la eficiencia del servicio de evaluación de productos de software.
3. La estrategia planteada se basa en la ejecución de dos etapas. La primera es el establecimiento del proceso Evaluación de Productos de Software y la segunda la definición de actividades de seguimiento y control de dicho proceso.
4. Mediante la implementación parcial de la estrategia, con el establecimiento del proceso Evaluación de Productos de Software, se pudo constatar su impacto positivo con la realización de una encuesta. La estrategia se evaluó en procesos pilotos de prestación del servicio que evidenció un incremento significativo en la eficiencia.
5. Las etapas de la estrategia definidas, el proceso de evaluación, los roles y las actividades de seguimiento y control, fueron evaluados de positivos, constatándose mediante triangulación de métodos, la convergencia de los resultados, evidenciando que la solución propuesta es adecuada y resuelve el problema de la investigación.

RECOMENDACIONES

El resultado de la presente investigación permitió al autor proponer las siguientes recomendaciones:

1. Desarrollar la implementación total de la estrategia e incluir mejoras que evidencien su adaptabilidad, a partir de los resultados de la implementación.
2. Evaluar la satisfacción del cliente como otra forma de validación de la estrategia, que permitirá identificar elementos que puedan ser objetos de mejora para futuras propuestas.
3. Proponer un proceso de planificación de las evaluaciones basado en el uso del razonamiento basado en casos, a partir de la conformación del cuerpo de conocimientos resultantes de la implementación de la estrategia.
4. Identificar y evaluar otros indicadores que contribuyan a elevar la eficiencia del servicio Evaluación de Productos de Software propuesto.

BIBLIOGRAFÍA

- Acuña Bermeo, Cristina Fernanda. 2008. «Diseño, propuesta de mejora y estandarización de los procesos orientados al cliente de la empresa Edinun». <http://eelalnx01.epn.edu.ec/handle/15000/997>.
- Alberto Medina León, Dianelys Nogueira Rivera, Arialys Hernández Nariño, y Jorge Viteri. 2010. «Relevancia de la Gestión por Procesos en la Planificación Estratégica y la Mejora Continua». *Revista EÍDOS* Número 2 (febrero). <http://www.ute.edu.ec/posgrados/EIDOS2.pdf>.
- Anaya, Raquel. 2012. «Una visión de la enseñanza de la ingeniería de software como apoyo al mejoramiento de las empresas de software». *Revista Universidad EAFIT* 42 (141): 60–76.
- Astigarraga, Tara, Eli M. Dow, Christina Lara, Richard Prewitt, y Maria R. Ward. 2010. «The emerging role of software testing in curricula». En *Transforming Engineering Education: Creating Interdisciplinary Skills for Complex Global Environments, 2010 IEEE*, 1–26. IEEE. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5508833.
- Belt, Pekka. 2009. «Improving verification and validation activities in ICT companies— product development management approach». *Acta Universitatis Ouluensis C Technica* 324. <http://herkules.oulu.fi/isbn9789514291487/isbn9789514291487.pdf>. ISBN 978-951-42-9147-7
- Boned Pascual, Carlos Javier. 2007. «La gestión por procesos». *Instalaciones deportivas XXI* (151): 68–71.
- Bravo Carrasco, Juan. 2010. *Gestión de procesos (La participación es la clave)*. Tercera edición rediseñada. Santiago Chile: Evolución SA. ISBN 978-956-7604-17-3
- Burnstein, Ilene. 2003. *Practical software testing: a process-oriented approach*. Springer. <http://www.google.com/books?hl=es&lr=&id=0v6HSeqA00oC&oi=fnd&pg=PR15&dq=%22Practical+Software+Testing%22&ots=9FMZghaSfD&sig=JIKhINXC51dgYZ7khH9G2yM59DE>.
- Burnstein, Ilene, Taratip Suwanassart, y Robert Carlson. 1996. «Developing a testing maturity model for software test process evaluation and improvement». En *Test Conference, 1996. Proceedings., International*, 581–589. IEEE. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=557106.
- Cabero Almenara, Julio, y Julio Barroso Osuna. 2013. «La utilización del juicio de experto para la evaluación de TIC: el coeficiente de competencia experta».

- Academia.edu* (febrero 19).
https://www.academia.edu/4023370/La_utilizacion_del_juicio_de_experto_para_la_evaluacion_de_TIC_el_coeficiente_de_competencia_experta. ISSN 0210-5934
- Capote García, Tayché. 2011. «Conceptualización e implantación de un Laboratorio Industrial de Pruebas de Software». Tesis de Maestría, La Habana: Universidad de las Ciencias Informáticas.
- Carnegie Mellon Software Engineer Institute. 2006. «CMMI® for Development v1.2».
- Carrasco, O. M. F., D. G. León, y A. B. Benavides. 2004. «Un enfoque actual sobre la calidad del software». *Revista ACI*.
- Carrión Maroto, Juan. 2007. *Estrategia: de la visión a la acción*. ESIC Editorial.
http://www.google.com/books?hl=es&lr=lang_es&id=8_PwloGOa6QC&oi=fnd&pg=PA13&dq=%22definici%C3%B3n+de+estrategia%22&ots=BaRtXiTCQh&sig=7OoGJZSIE89c8e3sAvAmgrxQVI8.
- Craig, Rick David, y Stefan P. Jaskiel. 2002. *Systematic software testing*. Artech House.
http://books.google.com/books?hl=es&lr=&id=2_gbZYzCZXgC&oi=fnd&pg=PR19&dq=%22Systematic+Software+Testing%22&ots=sU_IFL3FG4&sig=PRAS7NCPWmFRcOKx5LZVcek61yk.
- Cusmai, Marcelo. 2011. «Calidad de Software desde Latinoamérica». *Infotesting. Un espacio para testers profesionales. Revista digital*, abril.
- Díaz Pérez, Heney, y Daira Pérez Serrano. 2012a. «Case Based Reasoning to Infer the Behavior of a Release Testing of Software Products». En *Proceedings Latin American Congress on Requirements Engineering & Software Testing LACREST*, 106. Medellín, Colombia.
<http://lacrest.remington.edu.co/files/memoriaslacrest.pdf#page=106>. ISBN 978-958-46-0577-1
- . 2012b. «La Reunión de Inicio. Acuerdos para la provisión de Servicios de Pruebas de Liberación de Productos de Software». *Latin American and Caribbean Consortium of Engineering Institutions* (julio 23).
<http://www.laccei.org/LACCEI2012-Panama/RefereedPapers/RP087.pdf>.
- «Diccionario de la lengua española». 2014. Accedido junio 10.
<http://lema.rae.es/drae/?val=proceso>.
- Durán, C.R., M. Piore, y A. Schrank. 2005. «Los retos para el desarrollo de la industria del software». *Comercio Exterior* 55 (9): 744–753.
- Foro Económico Mundial. 2008. «Informe de competitividad mundial 2008-2009». Ginebra, *Informe de competitividad mundial*.

- Francés, Antonio. 2006. *Estrategia y planes para la empresa: con el cuadro de mando integral*. Pearson Educación.
http://www.google.com/books?hl=es&lr=lang_es&id=yAmLG-Vr8BkC&oi=fnd&pg=PA21&dq=definici%C3%B3n+de+estrategia+para+elevar+indicadores+de+eficiencia&ots=D1wXS9ABh5&sig=MULiRIMiSt3UCuTyNXg_WVQ1z-k.
- Fuentez Marquez, Ruben, Luis Fabricio García Serrato, Ivan González Hueto, Mario Iturbide Mondragón, y Jose Alfredo Luviano Pérez. 2010. «Algoritmo del proceso para la atención a usuarios del sistema del programa de apoyo al empleo web (SISPAEW)». Instituto Politécnico Nacional.
<http://itzamna.bnct.ipn.mx:8080/dspace/bitstream/123456789/5690/1/im179.pdf>.
- Fuertes Moreno, Freddy Fabián, y Renato David Paz y Miño Espinoza. 2011. «Diseño del plan estratégico y estandarización de los procesos de la Cooperativa de Ahorro y Crédito Politécnica Limitada». QUITO/EPN/2011.
<http://eelalnx01.epn.edu.ec/handle/15000/3872>.
- García, Javier, Antonio de Amescua, y Manuel Velasco. 2006. «TOP 10 de factores que obstaculizan la mejora de los procesos de verificación y validación en organizaciones intensivas en software». *Revista Española de Innovación, Calidad e Ingeniería del Software* 2 (2).
http://www.researchgate.net/publication/26575536_TOP_10_de_factores_que_obstaculizan_la_mejora_de_los_procesos_de_verificacin_y_validacin_en_organizaciones_intensivas_en_software/file/79e4150a92985ec8fb.pdf. ISSN: 1885-4486
- Góngora Rodríguez, Asnier Enrique, Alionuska Velazquez Cintra, Martha Nieves Borrero, Maikel Perez Castro, Heney Díaz Pérez, Lisset Rosas Moreno, y Yanetsi Millet Lombida. 2009. «Descripción de las clasificaciones de las no conformidades en un proceso de pruebas de liberación». En . *Novena Semana Tecnológica. Las TIC: Presente y Futuro*. Ciudad de la Habana: FORDES.
- Gonzalez, Reyes, Jose Gasco, y Juan Llopis. 2005. «Information systems outsourcing risks: a study of large firms». *Industrial management & Data systems* 105 (1): 45–62.
- Gutiérrez, Javier J., María J. Escalona, Manuel Mejías, y Antonia M. Reina. 2006. «Modelos de pruebas para pruebas del sistema». *Taller de Desarrollo de Software Dirigido por Modelos. XIII Jornadas sobre Ingeniería del Software y Bases de Datos JISBD*.
http://www.researchgate.net/publication/220776011_Modelos_de_Pruebas_para_Pruebas_del_Sistemas/file/3deec5260deaf51d49.pdf.

- Hernández, Vismar Santos. 2009. «La Industria Del Software. Estudio A Nivel Global Y América Latina». *Observatorio de la Economía Latinoamericana* (116).
- Humphrey, Watts S. 1997. *Introduction to the Personal Software Process SM*. Addison-Wesley Professional. http://www.google.com/books?hl=es&lr=&id=_g0te-9WuLYC&oi=fnd&pg=PR11&dq=Introduction+to+the+Personal+Software+Process&ots=4Qv9NDBvvJ&sig=yY_CO7gayv534I5aYHjpkQQJSqc.
- Idrobo Burbano, Gina Lorena, y Ingri Lorena Jojoa López. 2012. «Proceso para gerenciar proyectos de pruebas de software en empresas especializadas de servicios de aseguramiento de la calidad de software». http://bibliotecadigital.icesi.edu.co/biblioteca_digital/handle/10906/71260.
- Koomen, Tim, y Martin Pol. 1999. *Test Process Improvement: A practical step-by-step guide to structured testing*. Lulu. com.
- Labarca, Nelson. 2008. «Evolución del pensamiento estratégico en la formación de la estrategia empresarial». *Opción: Revista de Ciencias Humanas y Sociales* (55): 47–68. ISSN: 1012-1587
- León Perdomo, Yeniset. 2012. «Estrategia de Pruebas Exploratorias para mejorar el rendimiento del Laboratorio Industrial de Pruebas de Software de Calisoft». Tesis de Maestría, La Habana: Universidad de las Ciencias Informáticas.
- Mallar, Miguel Ángel. 2010. «La Gestión por procesos: un enfoque de gestión eficiente». *Visión de Futuro* No. 13 (No. 1) (junio): 127- 145. ISSN 1668 - 8708
- McCarthy, Ian, y Angela Anagnostou. 2004. «The impact of outsourcing on the transaction costs and boundaries of manufacturing». *International journal of production economics* 88 (1): 61–71.
- Medina Domínguez, Fuensanta. 2010. «Marco Metodológico para la Mejora de la Eficiencia de Uso de los Procesos Software». <http://e-archivo.uc3m.es/handle/10016/7433>.
- Medina-Giopp, Alejandro. 2003. «Gestión por Procesos y creación de valor público, un enfoque analítico». Tesis doctoral sin publicar. Departamento de Política y Administración y Dirección de Empresas. Barcelona, España, ESADE.
- Ministerio de Economía y Planificación. 2012. *Resolución No. 444/2012*.
- NC-ISO/IEC 9001: 2008. *Sistemas de Gestión de la Calidad—Fundamentos y Vocabulario [ISO 9001: 2008,(Traducción Certificada), IDT]*. NC-ISO/IEC.
- Ngwenyama, Ojelanki, y Jacob Nørbjerg. 2010. «Software process improvement with weak management support: an analysis of the dynamics of intra-organizational alliances in IS change initiatives». *European Journal of Information Systems* 19 (3): 303–319.

- Parada Gélvez, Hugo Alexer. 2010. «Contribución a la gestión de los procesos de pruebas de software y servicios». Telecomunicacion. <http://oa.upm.es/4019/>.
- Pérez Fernández de Velasco, José Antonio. 2007. *Gestión por procesos*. ESIC Editorial.
- Pérez Lamancha, Beatriz. 2006. «Proceso de Testing Funcional Independiente». *Facultad de Ingeniería. Montevideo Uruguay, PEDECIBA Informatica Instituto de Computación (InCo) Facultad de Ingeniería*. http://www.ces.com.uy/documentos/imasd/Tesis-Beatriz_Perez_2006.pdf.
- Pressman, Roger. 2002. *Ingeniería de Software. Un enfoque práctico*. 5ta ed.
- Rodríguez, Moisés. 2010. «Calidad de Procesos y Productos de Software - ISO/IEC 25000», julio. alarcos.inf-cr.uclm.es/per/fruiz/.../santander/mrodriguez-iso25000-update.pdf.
- Rodríguez Ruiz, Oscar. 2005. «La Triangulación como Estrategia de Investigación en Ciencias Sociales». *Revista de Investigación en Gestión de la Innovación y Tecnología. LA I+D QUE TENEMOS. Número 31 (septiembre)*. <http://www.madrimasd.org/revista/revista31/tribuna/tribuna2.asp>.
- Ruiz Beltrán, Paola Alexandra, y Jéssica Alexandra Suárez Maya. 2011. «Diseño y estandarización de los procesos institucionales de la Escuela Politécnica Nacional». *Repositorio Digital Escuela Politécnica Nacional de Ecuador*. <http://bibdigital.epn.edu.ec/handle/15000/2728>.
- Runeson, P., C. Andersson, y T. Thelin. 2006. «What do we know about defect detection methods?» *IEEE Software*.
- Sánchez, Miguel Ferrando, y Javier Granero Castro. 2005. *Calidad total: modelo EFQM de excelencia*. FC Editorial. http://www.google.com/books?hl=es&lr=&id=ZxYPb_6NcXsC&oi=fnd&pg=PA9&dq=EFQM&ots=qRfKHIPJ7Z&sig=T1JTePMXanWzVced5wLYs2DgcZs.
- Sangüesa, Marta. 2001. «Manual de Gestión de la Calidad». *Universidad de Navarra y DEIOC: Departamento de estadísticas, investigación operativa y computación. Universidad de Laguna [en línea]* <http://mediateca.rimed.cu/media/document/2216.pdf>.
- Sanz Esteban, Ana. 2012. «Marco metodológico y tecnológico para la mejora de las actividades de verificación y validación de productos software». Tesis Doctoral, Leganés: Universidad Carlos III de Madrid. <http://e-archivo.uc3m.es/handle/10016/15464>.
- Serna, Edgar, y Alexei Serna. 2014. «Escenarios, valores de entrada y contexto de uso en las pruebas de software Scenarios, input values and context of use in software testing». *I Conferencia Científica Internacional UCIENCIA 2014*.

- Accedido mayo 18. https://uciencia.uci.cu/sites/default/files/public/p2790-ponencia-501_0.pdf.
- Silvio Donolo, Danilo. 2009. «Triangulación: Procedimiento incorporado a nuevas metodologías de investigación». *Revista Digital Universitaria*, agosto 10. <http://www.revista.unam.mx/vol.10/num8/art53/int53.htm>. ISSN: 1607-6079
- T. Koomen, L. van der Aalst, B. Broekman, y M. Vroon. 2006. «TMap Next for result-driven testing». *UTN Publishers. Netherlands*.
- Utting, Mark, y Bruno Legeard. 2010. *Practical model-based testing: a tools approach*. Morgan Kaufmann. http://www.google.com/books?hl=es&lr=&id=8hAGtY4-oOoC&oi=fnd&pg=PP2&dq=%22Practical+Model-Based+Testing:+A+Tools+Approach%22&ots=OR2nSiDGDO&sig=bsZif_nZQC NFsnC8eeziRBBkohc.
- Valderrama, Francisco Oyarce. 2014. «Propuesta de Mejora del Proceso de Testing en Pequeñas Empresas». Accedido mayo 19. http://www.infonorchile2012.uta.cl/download.php?file=infonor2012_17.pdf.
- Van Veenendaal, Erik, y Jan Jaap Cannegieter. 2010. «Test Maturity Model integration (TMMi)». *TMMi Foundation*. http://www.tmmi.org/pdf/e-book_TMMi.pdf.
- Velázquez Cintra, Alionuska. 2013. «Estrategia de satisfacción del cliente para aumentar la calidad de los servicios de pruebas de software de CALISOFT, desde la perspectiva de los clientes». La Habana: Universidad de las Ciencias Informáticas.
- Zamora Hernández, Jorge. 2011. «Análisis de los procesos de verificación y validación en las organizaciones software». <http://orff.uc3m.es/handle/10016/12880>.

ANEXOS

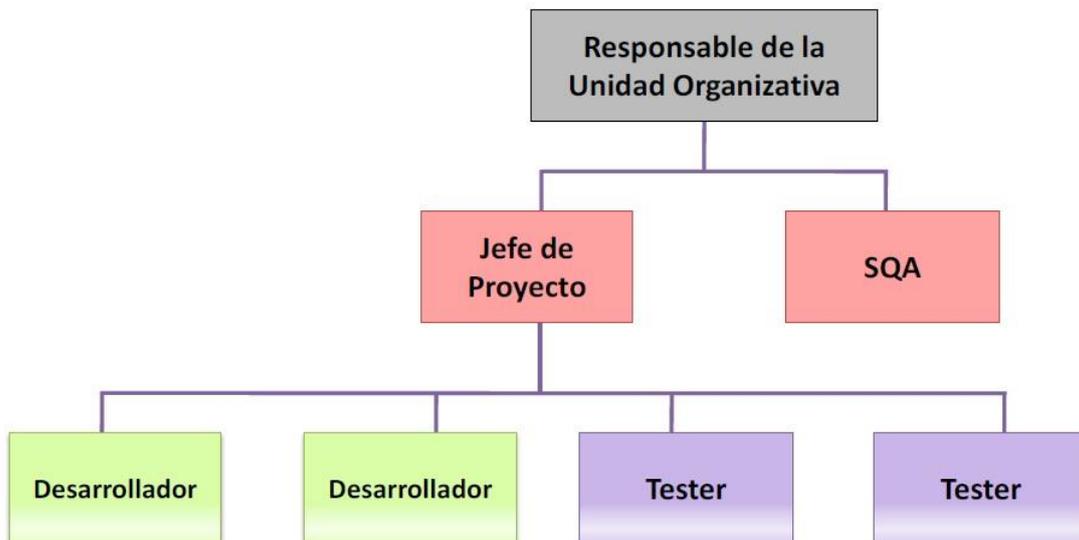
Anexo 1

Representación de los desarrolladores como probadores. Fuente: (Sanz Esteban 2012).



Anexo 2

Representación de los equipos de pruebas integrados. Fuente: (Sanz Esteban 2012).



Anexo 3

Representación de los equipos de pruebas independientes. Fuente: (Sanz Esteban 2012).



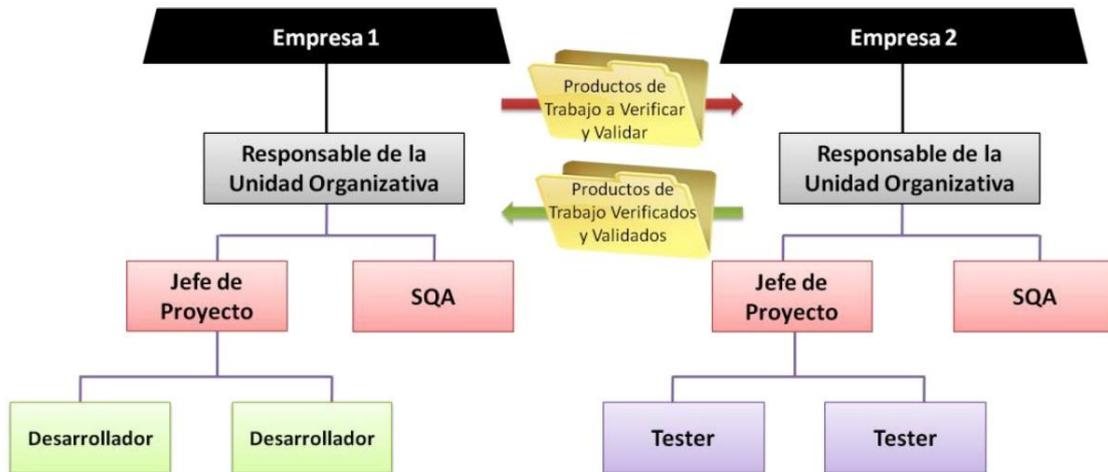
Anexo 4

Representación del grupo de aseguramiento de la calidad. Fuente: (Sanz Esteban 2012).



Anexo 5

Representación de las factorías de pruebas. Fuente: (Sanz Esteban 2012).



Anexo 6

Relación de artefactos o productos del desarrollo de software sometidos a evaluación en el LIPS y la documentación de apoyo. Fuente: Elaboración propia.

Artefactos a probar	Documentación necesaria
- Instalador de la aplicación	- Manual de Instalación
- Portal Web	- Pautas del diseño - Diseño de la Arquitectura de Información - Documento de Arquitectura de Software - Especificación de requisitos funcionales y no funcionales - Especificaciones de CU - Contenido (en caso de ser necesario) - Reglas del negocio
- Aplicación Web	- Pautas del diseño - Diseño de la Arquitectura de Información - Documento de Arquitectura de Software - Especificación de requisitos funcionales y no funcionales - Especificaciones de CU - Reglas del negocio
- Aplicaciones Escritorio	- Pautas del diseño - Diseño de la Arquitectura de Información (si lo tiene) - Documento de Arquitectura de Software - Reglas del negocio

	<ul style="list-style-type: none"> - Especificación de CU - Especificación de requisitos funcionales y no funcionales
- Multimedia	<ul style="list-style-type: none"> - Especificación de requisitos funcionales y no funcionales - Especificación de CU (si lo tiene) - Diseño de la Arquitectura de Información - Documento de Arquitectura de Software - Guión de Contenido
- Juegos	<ul style="list-style-type: none"> - Especificación de requisitos funcionales y no funcionales - Especificación de CU (si lo tiene) - Diseño de la Arquitectura de Información - Reglas del negocio - Guión de Contenido

Anexo 7

Tipos de pruebas de sistema. Fuente: Elaboración propia.

A continuación se describen los tipos de pruebas del sistema que se realizan en el LIPS, los cuales fueron definidos en función de las características de calidad identificadas en la norma ISO/IEC 9126-1 Parte 1: Modelo de Calidad, homologada en Cuba, y teniendo en cuenta las sub-características que se definen por cada una, que ya han sido mejoradas en la norma ISO/IEC 25000, Software Product Quality Requirements and Evaluation (SquaRE)

Características de calidad	Tipos de pruebas del sistema
Funcionalidad	<p>Funcionalidad: Consisten en la revisión de los requisitos aceptados por el cliente contra las funcionalidades presentes en la aplicación.</p> <p>Seguridad: Asegurar que los datos o el sistema solamente es accedido por los actores definidos según niveles de acceso. En 3 niveles.</p> <p>Volumen: Enfocada en verificar las habilidades de los programas para manejar grandes cantidades de datos, tanto como entrada, salida o residente en la BD.</p>
Confiabilidad	<p>Recuperación y tolerancia a fallas: Verificar que los procesos de recuperación (manual o automática) restauran apropiadamente la base de datos, aplicaciones y sistemas, y los llevan a un estado conocido o deseado.</p> <p>Comparativa de código: Se comparan dos ficheros, o partes del código, de manera que se puedan identificar los cambios que han sido realizados en el artefacto.</p>

Usabilidad	<p>Usabilidad: Prueba enfocada a factores humanos, estéticos, consistencia en la interfaz de usuario, ayuda sensitiva al contexto y en línea, asistente documentación de usuarios y materiales de entrenamiento.</p> <p>Estructura: Enfocada a la valoración a la adherencia a su diseño y formación. Este tipo de prueba es hecho a las aplicaciones Web asegurando que todos los enlaces están conectados, el contenido deseado es mostrado y no hay contenido huérfano.</p>
Eficiencia	<p>Contención: Enfocada a la validación de las habilidades del elemento a probar para manejar aceptablemente la demanda de múltiples actores sobre un mismo recurso (registro de recursos, memoria, etc.).</p> <p>Carga: Usada para validar y valorar la aceptabilidad de los límites operacionales de un sistema bajo carga de trabajo variable, mientras el sistema bajo prueba permanece constante. La variación en carga es simular la carga de trabajo promedio y con picos que ocurre dentro de tolerancias operacionales normales.</p> <p>Estrés: Enfocada a evaluar cómo el sistema responde bajo condiciones anormales. (extrema sobrecarga, insuficiente memoria, servicios y hardware no disponible, recursos compartidos no disponible)</p> <p>Rendimiento: Enfocadas a monitorear el tiempo en flujo de ejecución, acceso a datos, en llamada a funciones y sistema para identificar y direccionar los cuellos de botellas y los procesos ineficientes.</p>
Portabilidad	<p>Configuración: Enfocada a asegurar que funciona en diferentes configuraciones de hardware y software. Esta prueba es implementada también como prueba de rendimiento del sistema.</p> <p>Instalación: Enfocada a asegurar la instalación en diferentes configuraciones de hardware y software bajo diferentes condiciones (insuficiente espacio en disco, etc.).</p>

Anexo 8

Relación de los roles definidos con las actividades del proceso de evaluación de productos de software. Fuente: Elaboración propia.

	Jefe del DEPSW	Administrador de la calidad	Coordinador del servicio	Cliente	Coordinador de la evaluación	Planificador	Probador	Ingeniero de Pruebas	Gestor de la configuración	Gestor de entornos de pruebas
Analizar solicitud	X	X				X				
Asignar solicitud a especialista del DEPSW	X	X				X				
Rechazar solicitud	X									
Reunión de inicio			X	X	X					
Montaje del entorno de gestión de la evaluación									X	
Montaje del entorno de pruebas										X
Taller de diseño				X	X			X		
Declaración de evaluación abortada			X	X	X					
Ejecutar PEI				X	X					
Ejecutar iteraciones de pruebas				X	X		X	X		
Realizar reunión de evaluación abortada		X	X	X	X					
Ejecutar pruebas					X		X	X		
Preparar cierre de la evaluación					X					
Realizar reunión de cierre			X	X	X					
Cerrar entorno de pruebas										X
Actualizar expediente de la evaluación					X					
Actualizar registro de evaluaciones de productos de software									X	

Anexo 9

Planificación de actividades de la evaluación de productos de software. Fuente: Elaboración propia.

No.	Actividad	Fecha	Responsable	Participantes	Observaciones
1.	Elaboración del Plan de Evaluación del Producto.	[DD/MM/AAAA]	[Nombre del coordinador de la evaluación]	[Nombre del coordinador de la evaluación.]	[Esta actividad se realiza antes de la R/Inicio.]
2.	Realización de la Reunión de Inicio.	[DD/MM/AAAA]	[Nombre del coordinador de la evaluación]	[Involucrados en el proceso de evaluación]	Se aplican los Criterios de Criticidad.
3.	Realizar el diseño de la evaluación.	[DD/MM/AAAA]	[Nombre del coordinador de la evaluación]	[Involucrados en el proceso de evaluación]	Se realiza el Taller de Diseño.
4.	Preparación del entorno de pruebas.	[DD/MM/AAAA]	[Nombre del coordinador de la evaluación]	[Especialistas del LIPS]	Se dictamina que el entorno fue preparado según los requisitos de la evaluación.
5.	Instalación de la aplicación.	[DD/MM/AAAA]	[Equipo de desarrollo]	[Especialistas del equipo de desarrollo y del LIPS]	
6.	Etapa 1: Ejecución de las Pruebas Exploratorias Iniciales (PEI) o Pruebas Modulares.	[DD/MM/AAAA]	[Nombre del coordinador de la evaluación]	[Probadores del LIPS y coordinador de la evaluación]	Se aplican los Criterios de Criticidad.
7.	Reunión de Conciliación de los resultados de las PEI o Pruebas Modulares.	[DD/MM/AAAA]	[Nombre del coordinador de la evaluación]	[Involucrados en el proceso de evaluación]	Se recoge la Minuta de R/Conciliación PEI.
8.	Etapa 2: 1ra iteración de las pruebas. En esta iteración se ejecutarán los tipos de pruebas de	[DD/MM/AAAA]	[Nombre del coordinador de la evaluación]	[Probadores del LIPS y coordinador de la evaluación]	Se aplican los Criterios de Criticidad.

	sistema siguientes: <Especificar las pruebas que se realizarán en esta iteración.>				
9.	Notificación del fin de la iteración de prueba. Entrega de las NC detectadas al equipo de desarrollo.	[DD/MM/AAAA]	[Nombre del coordinador de la evaluación]	[Nombre del coordinador de la evaluación]	Vía correo electrónico.
10.	Solución de las NC en los artefactos en evaluación y actualización de los Casos de Prueba y artefactos de apoyo.	[DD/MM/AAAA]	[Líder de Proyecto]	[Equipo de desarrollo]	
11.	Conciliación de las NC que fueron declaradas como No Procede. Inicio de las Pruebas de Regresión.	[DD/MM/AAAA]	[Nombre del coordinador de la evaluación]	[Nombre del coordinador de la evaluación y especialistas del equipo de desarrollo]	Se aplican los Criterios de Criticidad.
12.	2da iteración de las pruebas. En esta iteración se ejecutarán los tipos de pruebas de sistema siguientes: <Especificar las pruebas que se realizarán en esta iteración.>	[DD/MM/AAAA]	[Nombre del coordinador de la evaluación]	[Probadores del LIPS y coordinador de la evaluación]	Se inicia la prueba con la revisión de los artefactos que hayan sido solicitados. Se aplican los Criterios de Criticidad.
13.	Notificación del fin de la iteración de prueba. Entrega de las NC detectadas al equipo de desarrollo.	[DD/MM/AAAA]	[Nombre del coordinador de la evaluación]	[Nombre del coordinador de la evaluación]	Vía correo electrónico.

14.	Solución de las NC en los artefactos en evaluación y actualización de los Casos de Prueba y artefactos de apoyo.	[DD/MM/AAAA]	[Líder de Proyecto]	[Equipo de desarrollo]	
15.	Conciliación de las NC que fueron declaradas como No Procede. Inicio de las Pruebas de Regresión.	[DD/MM/AAAA]	[Nombre del coordinador de la evaluación]	[Nombre del coordinador de la evaluación y especialistas del equipo de desarrollo]	Se aplican los Criterios de Criticidad.
16.	3ra iteración de las pruebas. En esta iteración se ejecutarán los tipos de pruebas de sistema siguientes: <Especificar las pruebas que se realizarán en esta iteración.>	[DD/MM/AAAA]	[Nombre del coordinador de la evaluación]	[Probadores del LIPS y coordinador de la evaluación]	Se inicia la prueba con la revisión de los artefactos que hayan sido solicitados. Se aplican los Criterios de Criticidad.
17.	Notificación del fin de la iteración de prueba. Entrega de las NC detectadas al equipo de desarrollo.	[DD/MM/AAAA]	[Nombre del coordinador de la evaluación]	[Nombre del coordinador de la evaluación]	Vía correo electrónico.
18.	Solución de las NC en los artefactos en evaluación y actualización de los Casos de Prueba y artefactos de apoyo.	[DD/MM/AAAA]	[Líder de Proyecto]	[Equipo de desarrollo]	
19.	Conciliación de las NC que fueron declaradas como No Procede. Inicio de las Pruebas de Regresión.	[DD/MM/AAAA]	[Nombre del coordinador de la evaluación]	[Nombre del coordinador de la evaluación y especialistas del equipo de desarrollo]	Se aplican los Criterios de Criticidad.

20.	[Etapa 3: Pruebas de Integración]	[DD/MM/AAAA]	[Nombre del coordinador de la evaluación]	[Probadores del LIPS y coordinador de la evaluación]	[Se aplican los Criterios de Criticidad]
21.	[Notificación del fin de la iteración de prueba. Entrega de las NC detectadas al equipo de desarrollo]	[DD/MM/AAAA]	[Nombre del coordinador de la evaluación]	[Nombre del coordinador de la evaluación]	[Vía correo electrónico]
22.	[Solución de las NC en los artefactos en evaluación y actualización de los Casos de Prueba y artefactos de apoyo]	[DD/MM/AAAA]	[Líder de Proyecto]	[Equipo de desarrollo]	
23.	[Conciliación de las NC que fueron declaradas como No Procede. Inicio de las Pruebas de Regresión]	[DD/MM/AAAA]	[Nombre del coordinador de la evaluación]	[Nombre del coordinador de la evaluación y especialistas del equipo de desarrollo]	[Se aplican los Criterios de Criticidad]
24.	[Prueba Final]	[DD/MM/AAAA]	[Nombre del coordinador de la evaluación]	[Probadores del LIPS y coordinador de la evaluación]	
25.	Actualización del expediente de la evaluación.	[DD/MM/AAAA]	[Nombre del coordinador de la evaluación]	[Nombre del coordinador de la evaluación]	En este tiempo debe firmarse el Acta de Liberación.
26.	Realización de la Reunión de Cierre.	[DD/MM/AAAA]	[Nombre del coordinador de la evaluación]	[Involucrados en el proceso de evaluación]	Se recoge la Minuta de R/Cierre y se evalúa el proceso.

Anexo 10

Instrumento para la determinación de la categoría experto/a. Fuente: Elaboración propia.

Saludos cordiales.

Con la finalidad de determinar expertos/as en la gestión de procesos de pruebas de software, solicitamos su colaboración evaluando su conocimiento del tema y el grado de influencia de las fuentes de argumentación, según las indicaciones que se dan a continuación. Le agradecemos de antemano por su valiosa contribución.

Instrucciones.

- Según su criterio, marque con una X, en orden creciente, el grado de conocimiento que usted tiene sobre el tema.

1	2	3	4	5	6	7	8	9	10

- Entre las fuentes que le han posibilitado enriquecer su conocimiento sobre el tema, se someten a consideración algunas de ellas, para que las evalúe en las categorías de: Alto (A), Medio (M) y Bajo (B), colocando una X.

Fuentes de argumentación	Grado de influencia de cada una de las fuentes		
	Alto (A)	Medio (M)	Bajo (B)
Estudios teóricos realizados por usted.			
Experiencia adquirida durante su vida profesional.			
Conocimiento de investigaciones y/o publicaciones nacionales e internacionales.			
Conocimiento propio sobre el estado del tema de investigación.			
Actualización en cursos de postgrado, diplomados, maestrías, doctorado, etc.			
Intuición.			

- Datos del encuestado.

Nivel escolar	Técnico medio ____		Universitario ____	
Categoría docente	Instructor ____	Asistente ____	Auxiliar ____	Titular ____
Categoría científica	Especialista ____	Máster ____	Doctor ____	
Años de experiencia				

Anexo 11

Instrumento para la obtención de los criterios valorativos de los expertos. Fuente: Elaboración propia.

Saludos cordiales.

Con la finalidad de someter a su consideración como experto/a la estrategia para elevar la eficiencia del servicio de evaluación de productos en el Laboratorio Industrial de Pruebas de Software, solicitamos su valoración sobre diferentes aspectos que a continuación se presentan. De antemano, le agradecemos su valiosa contribución.

1. Marque con una X, según su opinión, respecto a los siguientes aspectos relativos a la estrategia para elevar la eficiencia del servicio de evaluación de productos en el Laboratorio Industrial de Pruebas de Software, atendiendo a las siguientes categorías:

- MA: Muy adecuado.
- BA: Bastante adecuado.
- A: Adecuado.
- PA: Poco adecuado.
- I: Inadecuado.

Aspectos	MA	BA	A	PA	I
¿Qué opinión le merece la estrategia considerando si con la misma se logrará gestionar el proceso de evaluación de productos en el Laboratorio Industrial de Pruebas de Software para elevar la eficiencia del servicio de evaluación de productos?					
Valore las etapas definidas para el desarrollo de la estrategia, en función de lograr la gestión del servicio de evaluación de productos en el LIPS.					
¿Cómo evaluaría el proceso definido teniendo en cuenta la gestión del proceso de evaluación de productos de software en el LIPS?					
¿Considera que los roles definidos en la primera etapa de la estrategia favorecen la gestión del proceso de evaluación de productos?					
¿Cómo valora usted la definición de las actividades de seguimiento y control para la gestión del proceso de evaluación de productos en el laboratorio?					