# CAPITULO II

# FUNDAMENTOS TEÓRICOS

En el siguiente capítulo se presentan las bases teóricas que sustentan la presente investigación, haciendo un recorrido por conceptos de interés que van desde los antecedentes de la investigación, hasta llegar a los términos necesarios que soportan este trabajo.

## Antecedentes

Pasini, A., Esponda, S. Bertone, Rodolfo, A. y Pesado, P. en su trabajo llamado “Aseguramiento de Calidad en PYMES que desarrollan software. Una experiencia desde el proyecto COMPETISOFT.”, detallan los pasos y práctica de implementar su modelo COMPETISOFT en una empresa PYMES Argentina. Describen el modelo de procesos de COMPETISOFT basado en la Norma ISO 15504, el cual se divide en tres áreas: alta dirección que contiene el proceso de gestión de negocio, la gerencia que integra los procesos de gestión de procesos, de proyectos y recursos; y la última área es la de operaciones que integra los procesos administrativos de proyectos especiales y desarrollo y mantenimiento del software. Definen los cuatros niveles de capacidad de procesos de COMPETISOFT: Nivel 1 – Proceso realizado, Nivel 2 – Proceso Gestionado, Nivel 3 – Proceso Establecido, Nivel 4 – Proceso Predecible y por último el Nivel 5 – Proceso Optimizado.

Con base en PMCOMPETISOFT, desarrollan una herramienta de evaluación para el área de operaciones, en donde sólo implementa el modelo de mejora de procesos, que cuenta con 5 macro actividades: instalación de ciclos, Diagnóstico de los procesos, Formulación de ciclo de mejora, Mejora de proceso y Revisión del Programa. En la realización de esta investigación se completó cada una de las macro actividades de manera iterativa, se aplicó una encuesta para conocer las principales falencias de la empresa; luego realizó la formulación de propuesta de mejora, y establecen tiempo y los pasos a seguir en cada ciclo, y para finalizar realizaron una revisión de las mejoras, donde se compararon los cambios obtenidos en los ciclos y las fallas que aún presenta.

Los resultados obtenidos en este trabajo, validaron el modelo COMPETISOFT en Pymes, en donde se comprobó que la aplicación iterativa planteada permitió abordar el proceso de mejora eficientemente y sin problema, y al finalizar obtuvieron mejoras significativas en el desarrollo de software de la empresa.

Cuevas y Leal en 2014, estandarizaron los procesos asociados al desarrollo de proyectos informáticos en una empresa chilena, proporcionando una metodología estándar para el desarrollo de software. En primer lugar, analizaron los estándares existentes, seleccionando IEEE 730 e ISO/IEC 12007 por dos razones fundamentales, los Jefes de Proyectos y desarrolladores contaban con un conocimiento previo, y debido a que dichos modelos se pueden usar en desarrollo de software en cascada o secuencial. Posterior la selección de los estándares, definieron los tipos de proyectos informáticos que realizaba la empresa, para determinar las etapas qué debe cumplir cada desarrollo de proyecto y finalmente los productos generados en cada una de las etapas. A su vez, por cada producto definido, se establece un documento patrón, para crear estándares.

Identificado los tipos de proyectos, etapas y productos, realizan una matriz Tipo-Etapa-Producto, para unir las definiciones anteriores, de tal forma lograr seguir las pautas definidas y hacer las verificaciones correspondientes a medida que cada proyecto avanza en su desarrollo. Al ser implementado en un proyecto piloto, se logra medir el impacto, evaluar el uso por parte de los equipo de trabajo (Jefe de Proyecto y desarrolladores) y observar aspecto como: tiempos de finalización de etapas intermedias de los proyectos, controles de agenda de actividades y presupuesto.

Los resultados iniciales mostraron un avance significativo en la mejora de los procesos relacionados con el desarrollo software, obtuvieron mejoras de un 30% en cuanto a tiempo de entrega de producto, a pesar de incrementar el trabajo administrativo que conlleva, en especial por el registro de actividades que comúnmente no eran consideradas; y una disminución de 20% del presupuesto estimados en comparación con los reales. Su principal logro fue familiarizar a los integrantes de la empresa, con el concepto de calidad como una actividad transversal en el proceso de desarrollo de proyecto, y no como una actividad aislada que ocurría luego de codificar.

## Bases Teóricas

La presente investigación se sustenta en las siguientes teorías, conceptos, estándares que permitirán abordar de manera precisa el área de conocimiento en la que se desarrollará.

### Ingeniera de Software

Pressman (2005), señala que Fritz Bauer define la ingeniera de software como “el establecimiento y uso de principios solidos de la ingeniería para obtener económicamente un software confiable y que funcione de modelo eficiente en máquinas reales.”. Por otro lado, IEEE 610.12 (1990), elabora una definición más comprensible: “Ingeniera de software: 1) La aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento del software; es decir, la aplicación de la ingeniera al software. 2) El estudio en enfoques como en 1)”. Díaz (2008) hace referencia a Dennis indicando que el concepto de Ingeniera de Software “Cubre el desarrollo de sistemas totales, en los que pude o no incluir software. Es donde los ingenieros en sistema se concentran en transformar las necesidades y expectativas de los clientes en productos”.

### Calidad

El concepto de calidad ha sido definido por un elevado número de autores de forma no coincidente. Según Deming (1982) define a la calidad como el grado perceptible del cliente, donde mejorar la calidad implica una reducción de costos y una mejora en la productividad que permite a la empresa un incremento competitivo en el mercado. Sin embargo Juran (1989) indica que calidad es la adecuación al uso, y Albrecht (1995) considera la calidad como esa propiedad intangible que resulta de la diferencia entre el bien servicio que se espera y el que se recibe.

La norma ISO 8402 en 1994 define calidad como: “Totalidad de características de un producto o servicio que le confieren su aptitud para satisfacer unas necesidades expresadas o implícitas”. En el 2005, la norma ISO 9000 ofrece una definición de calidad muy genérica y de aplicación en distintos campos: “el grado en el que un conjunto de características inherente cumple con los requisitos”; dicha norma establece que un requisito es una “necesidad o expectativa establecida, generalmente implícita u obligatoria”, donde los requisitos son generados por las diferentes partes interesadas, por ejemplo: requisito de un producto, requisito de la gestión de calidad y requisito del cliente.

Por otro lado, La Sociedad Americana de Calidad (American Society for Quality - ASQ) es una comunidad global de personas dedicada a la calidad con entrenamiento y certificado profesional, definen calidad como: “un término subjetivo, para lo cual cada persona tiene su propia definición. En el uso técnico, la calidad puede tener dos significados: 1.- las características de un producto o servicio que les permite satisfacer una determinada necesidad. 2.- Un producto o servicio libre de deficiencias.”.

Concluyendo, calidad es la sumatoria de los valores agregados durante el proceso a los productos y/o servicios, donde los clientes obtienen la satisfacción de sus necesidades y/o expectativas. La calidad de un producto y/o servicio es la percepción que el cliente tiene del mismo, quien define los requisitos de lo que necesita y como lo necesita.

### Calidad de Software

El diccionario IEEE estándar de ingeniería del software (IEEE, 1990) define el software como “los programas de ordenador, los procedimientos y, posiblemente la documentación asociada y los datos relativos a la operación del sistema informática”. Así mismo, la IEEE Standard 610.12 (1990) define calidad de software como “el grado en el cual un componente, sistema o proceso satisface requisitos especificados y/o necesidades y expectativas del usuario/cliente”.

Pressman (2005) define la calidad del software como “la concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos con los estándares de desarrollo explícitamente documentados y con las características implícitas que se espera de todo software desarrollado profesionalmente”.

Por su parte la ISO 9126 (1991), establece las características de calidad para productos de software, y puede ser descrito en términos de una o más de seis características básicas, las cuales son: funcionabilidad, fiabilidad, mantenibilidad, portabilidad, usabilidad y eficiencia; cada una de las cuales se detalla a través de un conjunto de sub-características que permite profundizar en la evaluación de la calidad del producto software.

La calidad de software no es directamente comparable con la calidad en la manufactura de productos, Sommerville (2011) define al menos tres problemas propios de su contexto: “… atender requerimientos que no se establecen directamente en la especificación del sistema, existencia de ciertas características de calidad complejas de especificar en forma no ambigua y dificultades en desarrollar un sistema que satisfaga las expectativas del usuario…”.

Las anteriores definiciones sirve para concluir que los requisitos del software es la base fundamental de las medidas de calidad, existe falta de calidad si no hay concordancia con los requisitos; la evaluación de la calidad del software va a implicar la comparación entre los requisitos y el producto generado. La calidad tiene relación con la realización de actividades que permitan asegurarla durante el desarrollo del software, el cumplimiento de estándares y procedimientos definidos, ayuda a garantizar que el producto cumpla con las especificaciones definidas inicialmente con el cliente.

### Gestión de Calidad del Software

La calidad se logra a través de la gestión de la calidad, la cual, según ISO 9000 (2005), consiste en “actividades coordinadas para dirigir y controlar una organización en lo relativo a la calidad”, donde se incluye la planificación de la calidad, el control de la calidad, el aseguramiento de la calidad y la mejora de la calidad, donde se establecen los objetivos y directrices generales de calidad en función de las políticas de calidad expresada formalmente por la alta dirección de la organización. Su principal objetivo es comprender las expectativas del cliente en términos de calidad, y realizar acciones para satisfacer dichas expectativas. En ISO 9001 (2008) menciona que la gestión de calidad aspira a incrementar la satisfacción del cliente por medio de la aplicación eficaz del sistema, donde se incluye los procesos para la mejora continua del sistema y el aseguramiento de la conformidad con los requisitos del cliente o usuario.

Existe ochos principios de gestión de calidad que pueden ser usados por los directivos con el propósito de conducir a la organización hacia una mejora en el desempeño. (ISO 9000, 2005). Dichos principios ayudan a operar de forma exitosa una organización, donde se necesita que esta se dirija y controle de forma sistemática y transparente. Los principios se describen en la Tabla 1.

Tabla 1. Los ocho principios de la gestión de la calidad. (ISO 9000, 2005)

|  |  |
| --- | --- |
| Principios de Gestión de Calidad | |
| Enfoque al cliente. | Las organizaciones dependen de sus clientes y por lo tanto deberían comprender las necesidades actuales y futuras de los clientes, satisfacer los requisitos de los clientes y esforzarse en exceder las expectativas de los clientes. |
| Liderazgo. | Los líderes establecen la unidad de propósito y la orientación de la organización. Ellos deberían crear y mantener un ambiente interno, en el cual el personal pueda llegar a involucrarse totalmente en el logro de los objetivos de la organización. |
| Participación del personal. | El personal, a todos los niveles, es la esencia de una organización, y su total compromiso posibilita que sus habilidades sean usadas para el beneficio de la organización. |
| Enfoque basado en procesos. | Un resultado deseado se alcanza más eficientemente cuando las actividades y los recursos relacionados se gestionan como un proceso. |
| Enfoque de sistema para la gestión. | Identificar, entender y gestionar los procesos interrelacionados como un sistema, contribuye a la eficacia y eficiencia de una organización en el logro de sus objetivos. |
| Mejora continua. | La mejora continua del desempeño global de la organización debería ser un objetivo permanente de ésta. |
| Enfoque basado en hechos para la toma de decisión. | Las decisiones eficaces se basan en el análisis de los datos y la información. |
| Relaciones mutuamente beneficiosas con el proveedor. | Una organización y sus proveedores son interdependientes, y una relación mutuamente beneficiosa aumenta la capacidad de ambos para crear valor. |

Como se mencionó anteriormente, la gestión de calidad de software está formada por cuatros actividades, las cuales son: Planificación de Calidad de Software, Control de la Calidad de Software, Aseguramiento de la Calidad de Software y Mejora de Calidad de Software.

#### Planificación de la Calidad del Software

La norma ISO 9000 (2005), define planificación de calidad como la “parte de la gestión de calidad enfocada al establecimiento de los objetivos de la calidad y a la especificación de los procesos operativos necesarios y de los recursos relacionados para cumplir los objetivos de la calidad”, incluyendo el establecimiento de planes de la calidad. ISO 9001 (2008) menciona que le corresponde a la organización planificar y desarrollar los procesos necesarios para la elaboración del producto; por su parte la organización debe determinar los objetivos de la calidad y los requisitos para el producto, establecer procesos y documentos proporcionando recursos específicos para el producto, realizar las siguientes actividades: verificación, validación, seguimiento, medición, inspección y pruebas específicas para el producto; y por último la organización debe llevar un registro para proporcionar evidencia de que los procesos de realización y el producto final cumplen con los requisitos establecidos.

La Planificación de la Calidad del Software según la Norma ISO 90003 (2004) es:

“La planificación de la calidad facilita el modo de adaptar la planificación del sistema de gestión de la calidad a un proyecto específico, producto o contrato. La planificación de la calidad puede incluir referencias genéricas y/o proyecto / producto / contrato específico de procedimientos, como apropiados. La planificación de la calidad debería ser revisada de nuevo junto con el progreso del diseño y desarrollo, y los elementos, en cada fase, deberían ser completamente definidos al comienzo de dicha fase. La planificación de la calidad puede ser revisada y aprobada por todas las organizaciones interesadas en su aplicación, como apropiadas”.

Los procesos, actividades y tarea debe ser planificadas y ejecutadas usando modelos de ciclos de vida adecuada, de acuerdo a la naturaleza de proyecto de software, considerando tamaño, complejidad, riesgo e integridad. El diseño y la planificación pueden llevar a ser un proceso evolutivo y un procedimiento que podría necesitar cambio o actualizaciones a lo largo del progreso del proyecto, por lo que se debe mantener una relación entre los objetivos y recursos de la organización.

A su vez la planificación de la calidad de software a nivel de proyecto debe abordar los siguientes puntos:

* Inclusión o referencia a los planes de desarrollo.
* los requerimientos de calidad relacionados con el producto o/y proceso.
* Los sistemas de gestión de la calidad adaptando y/o identificando los procesos e instrucciones específicos, apropiados para el ámbito del manual de calidad y algunas exclusiones expuestas.
* Los procesos de proyectos-específicos e instrucciones, tales como, especificación de pruebas del software detallando los planes, diseños, casos de pruebas y procesos para la unidad, integración, sistemas y pruebas de aceptación.
* Los métodos, modelos, herramientas, convenios de lenguajes de programación, bibliotecas, marcos de trabajo y otros componentes reutilizables para ser usados en los proyectos.
* Los criterios para el comienzo y el final de cada fase o etapa del proyecto.
* Los tipos de análisis y otras verificaciones y actividades de validación para ser llevadas a cabo.
* Los procesos de gestión de la configuración para ser llevados a cabo.
* Las actividades de seguimiento y las medidas para ser llevadas a cabo.
* Las personas responsables de aprobar los procesos de salida para su uso posterior.
* La formación necesaria para el uso de herramientas y técnicas, y la organización de la formación previa a la habilidad necesaria.
* Los registros para ser mantenidos.
* La gestión de cambios, como por ejemplo, para recursos, escalas de tiempo y cambios de contrato.

El plan de Calidad es un documento que describe la planificación de calidad e incluye el plan de diseño y desarrollo, donde se especifica que procedimientos y recursos asociados deben aplicarse, quien le corresponde aplicarlos y cuando deben aplicarse a un proyecto, producto o procese, a su vez, se define los atributos de calidad más importantes del producto a ser desarrollado y el proceso de evaluación de la calidad.

#### Control de la calidad del Software

Es definida por ISO 9000 (2005) como la parte de la gestión de calidad orientada al cumplimiento de los requisitos de la calidad. Pressman (2005) menciona que el control de calidad de software son técnicas y actividades de carácter operativos y esenciales en cualquier negocio que elabore productos de consumo, para lograr evaluar la calidad de los productos de software desarrollado, centrados en mantener bajo control el proceso de desarrollo y garantizar que el producto final satisfaga los requisitos que se le han establecidos inicialmente. El control de la calidad de software es un ciclo de realimentación del proceso de creación del producto, para lograr minimizar los defectos producidos, ya que se elimina las causas de los defectos en las diferentes fases del ciclo de vida.

#### Aseguramiento de Calidad el Software

La norma ISO 9000 (2005) menciona como la “parte de la gestión de la calidad orientada a proporcionar confianza en que se cumplirán los requisitos e la calidad”. A su vez intenta hacer predecible la calidad del producto y certificar de que el bien fabricado o el servicio entregado responden a las necesidades o requerimientos del cliente (CEOE-CEPYME, 2004). El objetivo del aseguramiento de la calidad es dar confianza a la dirección y al cliente de que se cumplirán sus especificaciones y requisitos a través del uso de procesos controlados y definidos previamente. (Pérez, Velasco, 1994).

Tenemos que diferencia el término de control de la calidad con el aseguramiento de la calidad. El aseguramiento de la calidad aprovecha los resultados obtenidos del control de calidad para evaluar y mejorar los procesos con los que se desarrolla el producto. Por ello, el control de calidad se enfoca en productos, mientras que el aseguramiento de calidad lo hace en los procesos.

Las actividades de aseguramiento de calidad de software, mencionada por Pressman (2005) son:

1. Establecimiento de un plan de SQA para un proyecto.
2. Participación en el desarrollo de la descripción del proceso de software del proyecto.
3. Revisión de las actividades de Ingeniería del Software para verificar su ajuste al proceso de software definido.
4. Auditoria de los productos de software designados para verificar el ajuste con los definidos como parte del proceso del software.
5. Asegurar que las desviaciones del trabajo y los productos del software se documentan y se manejan de acuerdo con un procedimiento establecido.
6. Registrar lo que no se ajuste a los requisitos e informar a sus superiores.

#### Mejora de la Calidad del Software

La norma ISO 9000 (2005) describe la mejora de la calidad de software como la “parte de la gestión de la calidad orientada a aumentar la capacidad de cumplir con los requisitos de la calidad”, dichos requisitos pueden estar relacionados con cualquier aspecto de eficacia, eficiencia o trazabilidad.

La mejora de la calidad de software es la actividad de la gestión de calidad encargada de efectuar las mejoras en la calidad de software, por medio de las mediciones a los análisis de datos y auditorias. (Pressman, 2005)

### Factores que afectan la Calidad del Software

Presmman (2005) menciona dos grupos de factores que afectan la calidad de software: los que se miden directamente (por ejemplo, defecto descubierto durante la fase de prueba) y los que solo se miden indirectamente (por ejemplo, facilidad de uso o de portabilidad)

Existen diversos factores que afectan a la calidad del software y puede ser clasificado en diferentes maneras. Por ejemplo, los investigadores Mc Call, Richards y Walters (1977) en su libro titulado “Factors in Software Quality” proponen los siguientes factores en base a los aspecto que trata; estos factores son: sus características operativas, su capacidad al cambio y su capacidad de adaptarse a nuevos entornos. En la Tabla 2, se presentan los factores de calidad según Mc Call, Richards y Walter.

Tabla 2. Factores de Calidad de Software segun Mc Call, Richards y Walter (1977).

|  |  |  |
| --- | --- | --- |
| **Aspecto que trata** | **Factor de Calidad** | **Descripción** |
| Operación del producto | Corrección | El grado en que un programa satisface sus especificaciones y cumple con los objetivos establecido por el cliente. Responde a la pregunta: ¿Hace lo que quiero? |
| Fiabilidad | El grado en que se puede esperar que un programa cumpla con las funciones esperadas con la precisión requerida. Responde a la pregunta: ¿Lo hace de forma fiable todo el tiempo? |
| Eficiencia | Cantidad de recursos de computadora y de código requeridos por un programa para llevar a cabo sus funciones. Responde a la pregunta: ¿Se ejecutara en mi hardware lo mejor que se pueda? |
| Integridad | El grado en que se puede controlarse el acceso al software o el dato, por persona ajena al sistema. Responde a la pregunta: ¿Es seguro? |
| Facilidad de uso | Esfuerzo requerido para aprender un programa, trabajar con él, preparar su entrada e interpretar su salida. Responde a la pregunta: ¿Es sencillo de usar? |
| Revisión del producto | Facilidad de mantenimiento | Esfuerzo requerido para localizar y arreglar un error en un programa. Responde a la pregunta: ¿Es fácil de arreglar? |
| Flexibilidad | Esfuerzo requerido para modificar un programa operativo Responde a la pregunta: ¿Es fácil de realizar cambio? |
| Facilidad de prueba | Esfuerzo requerido para probar un programa de forma que se asegure que realiza la función requerida. Responde a la pregunta: ¿Es fácil de probar? |
| Transición del producto | Portabilidad | Esfuerzo requerido para transferir el programa desde un hardware y/o entorno de sistemas de software a otro. Responde a la pregunta: ¿Es fácil de usar en otro computador? |
| Reusabilidad | Grado en que un programa (o partes de un programa) puede reusarse en otras aplicaciones. Responde a la pregunta: ¿Se podrá usar alguna parte del software? |
| Facilidad de interoperación | Esfuerzo requerido para acoplar un sistema a otro. Responde a la pregunta: ¿Podre interactuar con otro programa? |

El estándar ISO 9126 (1991) fue creado para identificar los atributos de calidad para el software, identificando los atributos que se muestran en la siguiente tabla:

Tabla 3. Factores de Calidad de Software según ISO 9126 (1991)

|  |  |
| --- | --- |
| **Factor de Calidad** | **Descripción** |
| Funcionalidad | Satisface las necesidades de: idoneidad, exactitud, interoperabilidad, cumplimiento y seguridad. |
| Confiabilidad | Disponibilidad de uso según: madurez, tolerancia a fallas y facilidad de recuperación. |
| Facilidad de uso | Si satisface los siguientes requerimientos: facilidad de comprensión, facilidad de aprendizaje y operatividad. |
| Eficiencia | Grado en el que el software emplea en forma óptima los recursos del sistema para: comportamiento en el tiempo, comportamiento en los recursos. |
| Facilidad de mantenimiento | Capacidad de reparación con relación a: facilidad de análisis facilidad de cambio, estabilidad, y facilidad de pruebas. |
| Portabilidad | Capacidad de cambiar el entorno del software según su: adaptabilidad, facilidad de instalación, cumplimiento y facilidad de reemplazo. |
| Flexibilidad | Esfuerzo requerido para modificar un programa operativo Responde a la pregunta: ¿Es fácil de realizar cambio? |

Por otro lado, los investigadores Grady y Caswell (1987) identificaron los siguientes factores:

Tabla 4. Factores de Calidad de Software según Grady y Caswell (1987)

|  |  |
| --- | --- |
| **Factor de Calidad** | **Descripción** |
| Funcionalidad | Se evalúan las características y posibilidades del programa, la generalidad de las funciones y la seguridad. |
| Facilidad de uso | Se consideran factores humanos, estética global, consistencia y documentación. |
| Fiabilidad | Se mide la frecuencia de fallos y su importancia, eficacia de los resultados de salida, tiempo medio entre fallos, recuperación ante fallos y previsibilidad de programa. |
| Rendimiento | Se evalúa la velocidad del proceso, el tiempo de respuesta, consumo de recursos, rendimiento total de procesamiento y eficiencia. |
| Capacidad de soporte | Combina la posibilidad de ampliar el programa, la adaptabilidad y la utilidad, facilidad de prueba, compatibilidad, capacidad de configuración, facilidad de instalación y facilidad para localizar los problemas. |

## Estándares de Calidad del Software a Nivel Producto

### ISO/IEC 9126

EL ISO/IEC 9126 (2000) posee el título de Tecnología de información – Calidad de producto software; fue desarrollado para la evaluación del producto software indicando las características y métricas de calidad y las directrices para el uso de dichas características. Dichas características y métricas asociadas son útiles no solo para la evaluación de un producto software sino también para la definición de los requisitos de calidad. ISO/IEC 9126 está conformado por cuatros partes: calidad del modelo, métrica externa, métrica interna y calidad de uso.

Parte 1: Calidad de Modelo.

En esta parte se describe dos partes de calidad del producto software: a) calidad interna y externa y b) calidad de uso. La primera parte menciona seis características de calidad interna y externa, que a su vez se subdividen en subcaracteristicas. Estas subcaracteristicas son manifestadas externamente cuando el software es usado como una parte del sistema y son el resultado de un atributo interno de software. La segunda parte del modelo menciona cuatros característica de uso, es el efecto de combinar el uso de las seis características de calidad de uso. Dichas características y subcaracteristicas proporciona un marco de trabajo para la especificación de los requerimientos de calidad de software asociado en la adquisición, desarrollo, evaluación, soporte, mantenimiento, aseguramiento de calidad y auditoria de software. A su vez ayuda a validar la complejidad de los requerimientos definidos del software, identifica requerimientos de software, objetivos de diseño, objetivos de prueba, criterio de aseguramiento de calidad y criterio de aceptación del producto. Puede ser usada por desarrolladores, evaluadores independientes y grupos de aseguramiento de la calidad responsable de especificar y evaluar la calidad del software.

ACA