**Aseguramiento de Calidad de Proceso y Producto (PPQA)**

**Calidad**: todos los aspectos y características de un producto o servicio que se relacionan con su habilidad de alcanzar las necesidades manifiestas o implícitas. La calidad también se puede entender como la conformidad en base a una especificación detallada, aunque esta definición es aplicable a la industria de manufacturas.

Lo que nos interesa es la calidad del producto, la calidad del proceso es un medio para un fin.

Los requerimientos son la base de la calidad.

**Problemática**: la calidad es subjetiva, depende de la perspectiva de la persona que va a ser el consumidor de dicho producto. Distintos consumidores van a priorizar distintos aspectos para determinar la calidad de un producto. El equipo de gestión de calidad debe usar su propio juicio para decidir si se logró un nivel aceptable de calidad, y para ello debe ver si se ajusta o no a su propósito pretendido. Es decir, se podrá a prueba al sistema en contra de sus requerimientos. Sin embargo, la calidad del software no solo se trata de si una funcionalidad se implementó correctamente, sino que también depende de los atributos no funcionales del sistema.

**Cosas que ocurren frecuentemente en los proyectos de desarrollo de software**

Son elementos que complican la entrega de un producto de calidad:

* Atrasos en las entregas
* Costos excedidos
* Falta cumplimientos de los compromisos
* No están claros los requerimientos, tiene que ver con el producto, y es uno de los factores de mayor peso. El resto de las cosas que ocurren tienen que ver con los proyectos.
* El software no hace lo que tiene que hacer
* Trabajo fuera de hora
* Fenómeno 90-90
* ¿Dónde está el componente?: se soluciona con una buena implementación del SCM

Todo esto hace que fracasen los proyectos de desarrollo. El producto no solo tiene que hacer lo que se especificó (cumpliendo las expectativas del cliente), sino que tiene cumplir con las expectativas del usuario, las necesidades de la gerencia (cumplir con los plazos, costos, etc.), cumplir con las necesidades del equipo de desarrollo y mantenimiento, etc. Es mas amplio el concepto de calidad que se debe satisfacer.

**Principios de calidad**

* La calidad es una actitud, no se puede pensar que se va a dejar para el ultimo la calidad. La calidad se **construye** con cada decisión que tomamos, no se puede hacer al final. La calidad debe **embeberse** a lo largo de la construcción. La prevención es mucho mas conveniente que la corrección. El costo de retrabajo es altísimo por lo que es mejor invertir en actividades de prevención.
* La calidad es un **esfuerzo de todos**. Todos los integrantes del equipo deben trabajar juntos para lograrla.
* Las **personas** **son clave** para lograrla, el software se hace con gente. Si la misma no esta dispuesta a trabajar de cierta manera, no va a funcionar.
* Se necesita un **compromiso** a nivel **gerencial**, pero se puede empezar por uno. Es decir, se necesita el apoyo de la gerencia, pero nosotros debemos tratar de conseguirlo.
* Se debe liderar con el ejemplo, tiene que ver con “haz lo que predicas”. No se puede construir un producto de calidad si no realizamos buenas prácticas.
* No se puede controlar aquello que no se mide
* Simplicidad, empezar por lo básico
* El aseguramiento de calidad debe planificarse (tiene que ver con el primer principio, la calidad se construye, no se agrega al final)
* Más testing no implica mayor calidad, el testing llega tarde
* Debe ser razonable para mi negocio

**¿Calidad para quién? – Visiones de la calidad**

Debe cumplir con todas las visiones para tener calidad. Se la puede analizar desde varias perspectivas

* **Visión del producto**: si el producto hace lo que tiene que hacer
* **Visión del proceso (manufactura):** tiene que ver con el proceso que usamos, y si el mismo es el adecuado para generar valor en el producto que construimos
* **Visión del valor**: lograr un equilibrio entre lo que cuesta hacer el producto y el precio al que lo podemos vender. De nada sirve tener un software de excelente calidad si nos vamos a la bancarrota para hacerlo, o no conseguimos compradores.
* **Visión del usuario**: es la más difícil de todas porque está en la cabeza del usuario, es la perspectiva desde la cual el usuario juzgara el producto. Debemos acercarnos a las expectativas del usuario.
* **Visión trascendental**: es una utopía, es como el motor que nos impulsa a seguir trabajando y querer mejorar. Por ejemplo: software con 0 defectos, es utópico, pero me motiva a mejorar los procesos, la comunicación, etc. Es para lo que uno trabaja.

**Gestión de la Calidad**

Para contrastar si un producto tiene o no calidad, el mejor indicador o comparador son los requerimientos del cliente. Es decir, si el producto cumple las expectativas del cliente. Para poder hacer la comparación entre lo que el cliente quiere (requerimientos) y lo que construimos, los requerimientos deben estar expresados en términos objetivos y verificables. Cualquier actividad que tiene que ver con calidad implica un proceso de comparación, veo el nivel de adecuación que tiene con lo que yo quiero comparar, tengo mi producto y tengo un estándar.

Podemos decir que existen tres dimensiones de la calidad de un producto:

* **Calidad programada:** la que planificamos para el producto.
* **Calidad realizada**: la que efectivamente hicimos. Es la calidad que posee el producto funcionando.
* **Calidad necesaria:** la mínima para que el producto funcione y le agregue valor al cliente.

Diagrama

Descripción generada automáticamentePodemos entender a la **gestión de calidad** como el trabajo sistemático que se hace en algunas organizaciones para que el producto final tenga la calidad que uno espera que tenga.

Las acciones de la gestión de la calidad buscan o deben lograr un equilibrio o conjunción de las tres calidades.

El punto en común de las tres debe ser lo mas amplio posible, para lograr las expectativas de calidad, o la **calidad esperada**. Todo lo que queda fuera de dicha coincidencia o punto de común será desperdicio o insatisfacción.

**Calidad en el software**

La ingeniería de software se mueve en tres dimensiones, proceso, proyecto y producto. Al hacer aseguramiento de calidad de producto, en este es donde está puesto el foco.

**Aseguramiento de calidad del producto**

Tenemos dos recursos para hacer aseguramiento de calidad de producto, **auditorias** y **revisiones** **técnicas**. La diferencia es que la auditoria es un proceso objetivo e independiente que se tiene que llevar a cabo por una persona que no tiene vinculo con el equipo de trabajo. Dentro del proceso de desarrollo de software se hacen 3 auditorias, auditorias de configuración (físicas y funcionales) y auditoría de proyecto.

La otra actividad que ha trascendido frameworks y filosofías, no importa el enfoque tradicional, lean o ágil, son las **revisiones técnicas**. Son reconocidas como un medio de aseguramiento de calidad y evitar la propagación de defectos. El propósito de la revisión técnica es la detección temprana de defectos. Se mantienen para todos los frameworks ya que las hace un colega, no un jefe o superior, esto se conce como peer review (revisión de pares), no hay ningún tipo de vinculo jerárquico entre el que revisa y el que hizo el artefacto. Se pueden hacer revisiones técnicas a cualquier artefacto que se genera, a requerimientos, diseño, a código, al diseño de una BD, etc.

En los enfoques ágiles la auditoría suele estar mal vista, y existe cierta resistencia hacia las mismas. Sin embargo, esto ni significa que no puedan hacer actividades como SCM, solo que no podrán obtener una certificación, ya que SCM implica hacer auditorías. Adicionalmente, si les interesa mejorar el proceso de gestión de configuración, pueden transformar las auditorias de configuración en revisiones técnicas de configuración. En definitiva, si quieren ver cómo está funcionando la gestión de configuración, el mismo equipo hace la revisión y no trae a un auditor independiente. La acción se puede hacer igual si el equipo considera necesario hacerla.

La otra actividad que se puede realizar dentro del foco del producto es el control de calidad. Esta es una de las actividades que se hacen dentro del aseguramiento de calidad para controlar la diferencia entre lo que debía hacer y lo que realmente se hizo. Dentro de los recursos de control de calidad, tenemos el testing. El testing es control de calidad, llega tarde, cuando el producto ya está construido. Se pone en evidencia el nivel de calidad del producto. No importa cuánto testing se haga, no va a mejorar la calidad del producto.

**Modelos de calidad del producto**: Si me interesa medir la calidad del producto, aparecen en el mercado algunos modelos teóricos que me permiten hacerlo. Un ejemplo de estas es la **ISO 25000**, de requerimientos no funcionales, pero también tiene otros apartados como

* Calidad de Uso
* Calidad de producto
* Calidad de datos

No es el único modelo de calidad de producto que hay:

* **Modelo de Barbacci:** se toma performance, seguridad y confiabilidad. Yo quiero medir el producto para ver donde esta ubicado y lograr un equilibrio. Si logro este equilibrio, el producto tiene calidad.
* **Calidad de software de McCall**: habla de la calidad con un enfoque de perspectiva. Una cosa son las características de calidad del producto para que pueda usarlo, otra es lo que necesito del producto para transferirlo a otro entorno, ampliar sus capacidades, etc. Y otras características son las que necesito para hacer revisiones técnicas.

Independientemente de lo distintos que son los modelos de calidad de producto, lo que establecen es una definición teórica para comparar mi producto y ver que tanto se adecua a la definición. Ninguno de estos modelos es certificable y adecuable. No se puede acreditar debido a la variabilidad que tiene cada producto, no se puede estandarizar. Son modelos que sirven para medir, para implementar una visión trascendental.

**Calidad de proceso**

Ejemplos de modelos de calidad de proceso: ISO, CMMI sirven para la calidad de proceso.

El problema histórico al hacer calidad de proceso es que no hay dos empresas iguales, que les sirva exactamente el mismo proceso. Cuando apareció en el mercado el primer modelo de calidad “CMM” (Modelo de madurez y capacidad), surge a pedido del departamento de defensa de USA. Empezó a sacar en limpio prácticas que los proyectos exitosos hacían. Es un conjunto de buenas practicas asociadas a hacer software. Se puso a disposición libre para que todos lo vieran, y empezaron a hacer evaluaciones de la madurez de las organizaciones.

Iban a una organización, le preguntaban por el nivel que quieren (se acreditan los niveles del 2 al 5). Comenzaron a aparecer consultores, que venían de certificar normas ISO, y lo que hacían era bajar el proceso a las empresas, pero nunca tuvieron el criterio de si dicho proceso le generaba valor a la empresa. Lo que les sirve a unos no necesariamente les sirve a otros. Este modelo te dice que hay que hacer, pero no el cómo hacerlo, y el cómo es la clave del éxito.

La idea detrás de los modelos de calidad de los procesos es que la única cosa que yo puedo **controlar** al hacer software es el **proceso**, todo lo demás esta fuera de control (evolución de la tecnología, las personas, el producto y sus características). Este enfoque asume que yo puedo controlar el proceso, y es el único medio que tengo para controlar la calidad del producto.

Es decir, implican que puedo tener calidad de producto a través de la calidad del proceso. Si tengo un proceso de calidad, el producto resultante debe tener calidad también.

**Procesos sin definir**

Las organizaciones tienen una forma de hacer las cosas, eso se llama proceso. Si no está explicito, no está escrito, no está comunicado, estamos frente a un proceso sin definir. Lo que espera de mínima cualquier modelo de calidad, es que las organizaciones diseñen su proceso (que genera valor, y se considera que la empresa necesita). Dicho proceso es una definición teórica de lo que yo quiero hacer. Luego debo usarlo al proceso, y allí es donde aparecen los proyectos.

Tomamos el proceso y se hace una adaptación, la cual es fundamental (para evitar que la gente haga cosas que piensa que no sirven para nada). La adaptación se hace con una actitud ética y profesional. Adapto el proceso al proyecto, pero es el proyecto el que usa el proceso, el que lo instancia. Los modelos de calidad de proceso buscan que las organizaciones diseñen sus procesos de forma tal que le agreguen valor a la organización, pero si quieren acreditarlo, este proceso debe ser compatible con el modelo de referencia (CMM, ISO, SPICE). Es decir, miro los modelos de calidad de proceso para crear el propio, y dichos modelos no dicen cómo hacerlo, sino qué cosas debe tener/hace el proceso, y bajo estas consideraciones la organización y se definen prácticas concretas para el proceso.

Se debe definir un proceso compatible con el modelo para crearlo, pero el proceso se debe usar en un proyecto. Cualquier evaluación que mire solo el proceso no sirve, porque el proceso es la teoría y en dicha teoría el proceso funciona, por lo que lo importante es verlo en la práctica.

Yo evalúo los proyectos para ver el nivel de adecuación que tienen los procesos a los proyectos, y allí aparecen las auditorias. El aseguramiento de calidad del proceso tiene dos dimensiones, la teórica (definición de un proceso de calidad) y la practica (ver si el proceso se usa o no, y esto se hace en los proyectos) En la práctica es el único lugar donde lo puedo medir, allí se hacen las auditorias de proyecto que evalúa el nivel de cumplimiento que tiene el proyecto del proceso que dijo utilizar. La auditoria es la tercera que se usa en el software (las otras dos son las auditorias de configuración)

**Un proceso de desarrollo de software**

Es el procedimiento y métodos escritos, y también incluye gente que lleve adelante el proceso, y un soporte de herramientas que este lo mas automatizado posible. Cualquier pata que falte genera un desequilibrio (mesa de tres patas).

**Interfaz de usuario gráfica, Diagrama

Descripción generada automáticamenteDefinición de un proceso:** Se pueden definir con niveles de detalle distintos, dependiendo del nivel de madurez que se quiere alcanzar. Podemos definir las etapas, y dentro de las etapas las tareas. Podemos incluir quienes deben participar, que deben hacer, cuando lo deben hacer y como lo deben hacer.

Diagrama

Descripción generada automáticamentePara hacer software se hace siempre lo mismo, pero depende el nivel de detalle que le queramos introducir. Tenemos una parte de ingeniería la cual abarca la parte donde elicito requerimientos, los analizo y diseño la solución. Esto relacionado a las disciplinas técnicas. Luego se implementa, se prueba y despliega.

Transversalmente están las disciplinas de gestión y de soporte.

Si queremos definir un proceso de desarrollo de software debemos considera todo lo anterior, y debemos definir tareas y artefactos para cada una de las disciplinas.

**Administración de la calidad del software**

La gestión de calidad del software tiene tres intereses fundamentales:

* A nivel de organización, busca establecer un marco de procesos y estándares de organización que conducirán a software de mejor calidad. Es decir, se deben definir procesos de desarrollo, de software a usar.
* A nivel de proyecto, implica la aplicación de los procesos específicos de calidad y asegurarse de que dichos procesos sean respetados.
* Se ocupa de establecer un plan de calidad para cada proyecto

El aseguramiento de calidad implica la definición de procesos y estándares que deben conducir a la obtención de productos de software de alta calidad. Aunque si bien los estándares y los procesos son importantes para la calidad, los administradores de calidad deben enfocarse también en desarrollar una cultura de calidad, donde la calidad es vista como una responsabilidad de todos y cada uno de los integrantes del equipo. Se debe exhortar a los equipos a sumir la responsabilidad de la calidad de su trabajo. A pesar de que los estándares y procedimientos son la base de la gestión de calidad, los buenos administradores de calidad reconocen que existen aspectos intangibles a la calidad del SW que no pueden expresarse en estándares.

La administración de calidad de software implica definir procesos y mantenerlos. Si un proceso se usa va a demandar cambios permanentes debido a los ciclos de mejora continua. El proceso no se define una sola vez y se deja tal cual, sino que cambia con el tiempo a medida que aparecen oportunidades de mejora.

**Responsabilidades**: Las áreas de aseguramiento de calidad se responsabilizan por la calidad del proceso, y la misma sea un medio para ganar calidad de producto.

Si yo quiero poner un grupo de aseguramiento de calidad en una empresa (puede ser una sola persona), debe ser independiente del equipo de desarrollo, para tener una perspectiva objetiva del software. No alcanza con ser independiente, sino que el responsable de calidad no puede depender del responsable del proyecto. Debería ser una posición cercana a la gerencia de primer nivel (C-suite). El reporte debe ir a la primera línea, para tener la independencia y poder ser objetivo.

Un equipo de gestión de calidad independiente garantiza que las metas de calidad de la organización no estén comprometidas.

**Actividades de la administración de calidad de software**

Debe definir un proceso, realizar las auditorias a los proyectos (tanto auditoria de proyecto, como las de configuración), hacer revisiones (no técnicas), y después sacar métricas y reportar los resultados. Todo esto me permite obtener información para implementar la mejora continua del proceso. A esas áreas de aseguramiento de calidad también hay que auditarlas. Dichas auditorías se llaman auditorias de aseguramiento de calidad y suelen traer a alguien de afuera. Hay tres actividades que realiza la administración de calidad de software:

* **Aseguramiento de calidad**: Establece estándares y procedimientos organizacionales de calidad
* **Planificación de calidad**: Selecciona los procedimientos y estándares aplicables a un proyecto en particular, y los modifica de ser necesario. Debe establecer las cualidades deseadas de software y describir cómo se valorarán. Para hacer un plan de calidad se deben seguir ciertos pasos:
  1. Introducción del producto**:** descripcion del producto y las expectativas de calidad.
  2. Planes del producto: fechas de entrega, responsabilidades, planes de distribución.
  3. Descripciones de procesos: procesos y estándares de desarrollo que se usan para el diseño y gestión del producto.
  4. Metas de calidad: identificación de las metas y los planes de calidad.
  5. Riegos y gestión del riesgo: riesgos que pueden afectar la calidad del producto y acciones a tomar.
* **Control de calidad:** se verifica si lo que planificamos se cumple o no. Asegura que los procedimientos y estándares son respetados por el equipo de desarrollo.

Escala de tiempo

Descripción generada automáticamenteEl proceso de gestión de calidad verifica los entregables del proyecto para garantizar que sean consistentes con los estándares y metas de la organización. Implica la definición de los momentos de auditar, cada vez que ejecuto las actividades que planifique, voy generando reportes, tomando métricas y generando visibilidad.

**Funciones del aseguramiento de la calidad de software**

Practicas de Aseguramiento de Calidad: Desarrollo de herramientas adecuadas, técnicas, métodos y estándares que estén disponibles para realizar las revisiones de aseguramiento de calidad.

Evaluación de la planificación del proyecto de software, Evaluación de requerimientos, Evaluación del proceso de diseño, Evaluación de las prácticas de programación, Evaluación del proceso de integración y prueba de software. Adaptación de los procedimientos de aseguramiento de calidad para cada proyecto.

**Procesos basados en calidad**

Diagrama

Descripción generada automáticamenteEs el por qué la calidad de proceso garantiza la calidad del producto

Yo defino un proceso de calidad, basándome en alguno de los estándares disponibles, luego deberé adaptar dicho proceso a un proyecto, para desarrollar un producto. Una vez esté desarrollado, voy a revisar la calidad del producto (usando métricas, auditorías, testing, etc.). Si el juicio del área de gestión de calidad, determinó que posee un nivel adecuado de calidad el proceso será estandarizado. En caso de que la calidad sea mala, se debe mejorar y ajustar el proceso.