

Gestión de la calidad

Contenidos del capítulo

- 27.1 Calidad de proceso y producto
- 27.2 Garantía de la calidad y estándares
- 27.3 Planificación de la calidad
- 27.4 Control de la calidad
- 27.5 Medición y métricas del software

La **calidad del software** es un concepto complejo que no es directamente comparable con la calidad de la manufactura de productos. En la manufacturación, la noción de calidad viene dada por la similitud entre el producto desarrollado y su especificación. En un mundo ideal, esta definición debería aplicarse a todos los productos, pero, para sistemas de software, existen estos **problemas**:

1. La especificación se orienta hacia las características del producto que el consumidor quiere. Sin embargo, la organización desarrolladora también tiene requerimientos (como los de mantenimiento) que no se incluyen en la especificación.
2. No se sabe cómo especificar ciertas características de calidad (por ejemplo, mantenimiento) de una forma no ambigua.
3. Es muy difícil redactar especificaciones concretas de software. Por lo tanto, aunque un producto se ajuste a su especificación, los usuarios no lo consideran un producto de alta calidad debido a que no responde a sus expectativas.

Se deben reconocer estos problemas con la especificación del software y se tienen que diseñar procedimientos de calidad que no se basen en una especificación perfecta. En concreto, atributos del software como mantenibilidad, seguridad o eficiencia no pueden ser especificados explícitamente. Sin embargo, tienen un efecto importante en cómo es percibida la calidad del sistema.

Algunas personas piensan que la calidad puede lograrse definiendo **estándares y procedimientos organizacionales de calidad** que comprueban si estos estándares son seguidos por el equipo de desarrollo. Su argumento es que los estándares deben encapsular las buenas prácticas, las cuales nos llevan inevitablemente a productos de alta calidad. En la práctica, sin embargo, es más importante la **gestión de la calidad** que los estándares y la burocracia asociada para asegurar el seguimiento de estos estándares.

Los buenos gestores aspiran a desarrollar una «**cultura de la calidad**» donde todos seamos responsables de que el desarrollo del producto sea llevado a cabo obteniendo un alto nivel de calidad en éste. Mientras estándares y procedimientos son las bases de la gestión de la calidad, los gestores de calidad experimentados reconocen que hay aspectos intangibles en la calidad del software (elegancia, legibilidad, etc.) que no puede ser incorporada en los estándares. Ellos ayudan a la gente interesada en estos aspectos intangibles de calidad y fomentan comportamientos profesionales en todos los miembros del equipo.

La gestión formal de la calidad es particularmente importante para equipos que desarrollan **sistemas grandes y complejos**. La documentación de la calidad es un registro de que es hecho por cada subgrupo en el proyecto. Esto ayuda a la gente a ver qué tareas importantes no deben ser olvidadas o que una parte del equipo no haga suposiciones incorrectas acerca de lo que otros miembros han hecho. La documentación de calidad es también un medio de comunicación sobre el ciclo de vida de un sistema. Ésta permite al grupo responsabilizarse de la evolución del sistema para saber qué ha hecho el equipo de desarrollo.

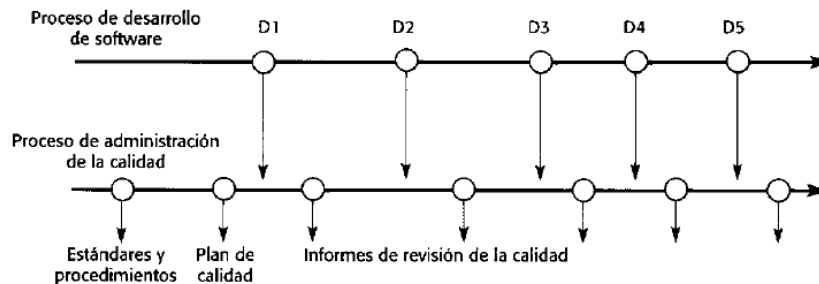
Para **sistemas pequeños**, la gestión de calidad es importante todavía, pero se debe adoptar una aproximación más informal. No son tan necesarios los documentos porque el grupo puede comunicarse informalmente. La clave de la calidad en el desarrollo de sistemas pequeños es el establecimiento de cultura de calidad y asegurarse de que todos los miembros del equipo hacen una aproximación positiva a la calidad del software.

La gestión de calidad del software se estructura en tres **actividades principales**:

1. Garantía de la calidad. El establecimiento de un marco de trabajo de procedimientos y estándares organizacionales que conduce a software de alta calidad.
2. Planificación de la calidad. La selección de procedimientos y estándares adecuados a partir de este marco de trabajo y la adaptación de éstos para un proyecto software específico.

3. Control de la calidad. La definición y fomento de los procesos que garantizan que los procedimientos y estándares para la calidad del proyecto son seguidos por el equipo de desarrollo de software.

La **gestión de la calidad** provee una comprobación independiente de los procesos de desarrollo software. Los procesos de gestión de la calidad comprueban las entregas del proyecto para asegurarse que concuerdan con los estándares y metas organizacionales. El **equipo de garantía de calidad** debe ser independiente del equipo de desarrollo para que puedan tener una visión objetiva del software. Ellos transmitirán los problemas y las dificultades al gestor principal de la organización.



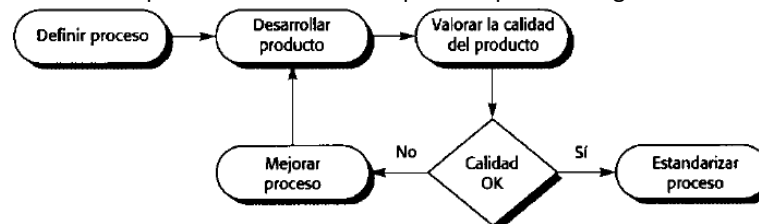
La gestión de la calidad y el desarrollo del software

Un equipo independiente de calidad garantiza que los objetivos organizacionales y la calidad no sean comprometidos por consideraciones de presupuesto o agenda.

27.1 Calidad de proceso y producto

Una suposición subyacente de la gestión de calidad es que la calidad del proceso de desarrollo afecta directamente a la calidad de los productos derivados.

La siguiente figura muestra una aproximación basada en proceso para conseguir la calidad del producto.



Calidad basada en procesos

Hay un vínculo claro entre la calidad del proceso y del producto en producción debido a que el proceso es relativamente fácil de estandarizar y monitorizar.

El software no se manufactura, sino que se diseña. El desarrollo de software es un proceso más creativo que mecánico. La calidad del producto, también se ve afectada por factores externos, como la novedad de una aplicación o la presión comercial para sacar un producto rápidamente.

En el desarrollo software, por lo tanto, la relación entre la calidad del proceso y la calidad del producto es muy compleja. Es difícil de medir los atributos de la calidad del software, en consecuencia, es difícil explicar cómo influyen las características del proceso en estos atributos. Además debido al papel del diseño y la creatividad en el proceso software, no podremos predecir la influencia de los cambios en el proceso en la calidad del producto.

La calidad del proceso tiene una influencia significativa en la calidad del software. La gestión y mejora de la calidad del proceso debe minimizar los defectos en el software entregado.

La gestión de la calidad del proceso implica:

1. Definir estándares de proceso.
2. Supervisar el proceso de desarrollo para asegurar que se sigan los estándares.
3. Hacer informes del proceso para el gestor del proyecto y para el comprador del software.

Un problema de la garantía de la calidad basada en el proceso es que el equipo de garantía de la calidad (QA) insista en unos estándares de proceso independientemente del tipo de software a desarrollar. El gestor principal debe intervenir para asegurar que el proceso de calidad ayude al desarrollo del producto en lugar de impedirlo.

27.2 Garantía de la calidad y estándares

La garantía de la calidad es el proceso que define cómo lograr la calidad del software y cómo la organización de desarrollo conoce el nivel de calidad requerido en el software. Como se indicó anteriormente, el proceso QA se ocupa ante todo de definir o seleccionar los estándares que deben de ser aplicados al proceso de desarrollo software o al producto software.

Podemos definir dos **tipos de estándares** como parte del proceso de garantía de calidad:

1. Estándares de producto. Se aplican sobre el producto software que se comienza a desarrollar. Incluyen estándares de documentación, como cabecera de comentarios estándar para definición de clases, y estándares de codificación.

2. Estándares de proceso. Definen los procesos que deben seguirse durante el desarrollo del software. Pueden incluir definiciones de procesos de especificación, diseño y validación, así como una descripción de los documentos que deben escribirse en el curso de estos procesos.

Existe una relación muy cercana entre los estándares de producto y los estándares de proceso. Los estándares de producto se aplican a las salidas del proceso software y, en muchos casos, los estándares de proceso incluyen actividades de proceso específicas que garantizan que se sigan los estándares de producto.

Los **estándares de software** son importantes por varias razones:

1. Están basadas en el conocimiento de la mejor o más apropiada práctica de la empresa, evita la repetición de errores anteriores.
2. Proveen un marco de trabajo alrededor del cual se implementa el proceso de garantía de la calidad. El control de la calidad sencillamente asegura que los estándares se siguen adecuadamente.
3. Ayudan a la continuidad cuando una persona continúa el trabajo que llevaba a cabo otra. Se reduce el esfuerzo de aprendizaje cuando se comienza un nuevo trabajo.

Utilizando estándares como punto de partida, el equipo de garantía de la calidad debe crear un «manual» de estándares. Éste define los estándares que son apropiados para la organización.

Algunas veces, los ingenieros de software consideran a los estándares como burocráticos e irrelevantes para las actividades técnicas de desarrollo de software.

Para evitar estos problemas, los gestores de la calidad que fijan los estándares necesitan estar informados y tomar en consideración los siguientes pasos:

1. Involucrar a los ingenieros de software en el desarrollo de los estándares del proyecto.
2. Revisar y modificar los estándares de forma regular con el fin de reflejar los cambios en la tecnología.
3. Proveer herramientas de software para apoyar los estándares donde sea necesario.

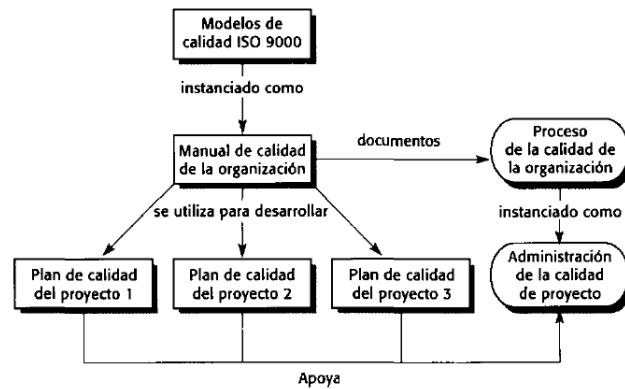
El gestor del proyecto y el gestor de calidad pueden evitarse los problemas de estándares inapropiados si planean cuidadosamente la calidad. Deben decidir cuáles son los estándares del manual que utilizarán sin cambio alguno, cuáles se modificarán y cuáles se dejarán de lado.

27.2.1 ISO 9000

Un conjunto de estándares internacionales que se puede utilizar en el desarrollo de un sistema de gestión de calidad en todas las industrias es ISO 9000. Los estándares ISO 9000 pueden aplicarse a un amplio abanico de organizaciones desde las de manufactura hasta las de servicios. ISO 9001 es el más general de estos estándares y se aplica en organizaciones interesadas en el proceso de calidad de diseño, desarrollo y mantenimiento de productos.

ISO 9001 no es un estándar específico para desarrollo de software, pero define principios generales que pueden aplicarse al software.

Aquí podemos observar las relaciones entre ISO 9000, el manual de calidad y los planes de calidad de proyectos particulares.



ISO 9000 y la gestión de la calidad

Los procesos de garantía de calidad en una organización se documentan en un manual de calidad que define el proceso de calidad.

El estándar ISO 9000 se refiere simplemente a la definición de los procedimientos que son utilizados en la compañía y la documentación asociada que muestre que los procesos han sido seguidos. Éste no se ocupa de asegurar que estos procesos sean la mejor práctica, ni asegura la calidad del producto.

27.2.2 Estándares de documentación

Los estándares de documentación en un proyecto de software son documentos muy importantes ya que son la única forma tangible de representar al software y su proceso. Los documentos estandarizados tienen una apariencia, estructura y calidad consistentes y, por lo tanto, son más fáciles de leer y de comprender.

Existen tres tipos de estándares de documentación:

1. Estándares del proceso de documentación. Definen el proceso a seguir para la producción del documento, esto implica definir los procedimientos involucrados en el desarrollo del documento y las herramientas de software utilizadas. También definen procedimientos de comprobación y refinamiento que aseguren que se produzcan documentos de alta calidad.
2. Estándares del documento. Gobiernan la estructura y presentación de los documentos.
3. Estándares para el intercambio de documentos. Aseguran que todas las copias electrónicas de los documentos sean compatibles.

Los estándares de calidad del **proceso de documentos** deben ser flexibles y les debe ser posible ajustarse a todos los tipos de documentos. Uno de los modelos posibles de proceso de documentación incluye: Crear un borrador, comprobarlo, revisarlo y rehacerlo es un proceso iterativo. Éste continúa hasta que se produce un documento de calidad aceptable. El nivel de calidad aceptable depende del tipo de documento y de los lectores potenciales de éste.

Aunque los estándares del documento se adapten a las necesidades de un proyecto específico, una buena práctica es que se utilice el mismo estilo en todos documentos producidos por una organización.

Algunos **ejemplos de estándares de documentos** a desarrollar son:

1. Estándares de identificación de documentos. Puesto que los proyectos de sistemas grandes producen cientos de documentos, cada documento debe identificarse de forma única. Para los documentos formales, este identificador es el identificador formal definido por el gestor de configuraciones. Para documentos informales, el estilo del identificador del documento es definido por el gestor del proyecto.
2. Estándares de la estructura del documento. Cada clase de documentos producida durante un proyecto de software debe seguir alguna estructura estándar.
3. Estándares de presentación de documentos. Estos estándares definen un «estilo propio» para los documentos y contribuyen notablemente a la consistencia de éstos.
4. Estándares para actualizar los documentos. Conforme el documento evoluciona y refleja los cambios en el sistema, se debe utilizar una forma consistente para indicar los cambios en el documento.

Los **estándares de intercambio** de documentos son importantes debido a que se pueden intercambiar copias electrónicas de los documentos. La utilización de estándares de intercambio permite que los documentos se transfieran electrónicamente y puedan reconstruirse en su forma original.

27.3 Planificación de la calidad

La planificación de la calidad es el proceso en el cual se desarrolla un plan de calidad para un proyecto. El plan de calidad define la calidad del software deseado y describe cómo valorar ésta. Por lo tanto, define lo que es software de «alta calidad». Sin esta definición, los diferentes ingenieros pueden trabajar en direcciones opuestas para optimizar los atributos de proyecto.

El plan de calidad selecciona los estándares organizacionales apropiados para un producto y un proceso de desarrollo particulares. Esta estructura comprende:

1. Introducción del producto. Descripción del producto, el mercado al que se dirige y las expectativas de calidad.
2. Planes de producto. Contiene las fechas de terminación del producto y las responsabilidades importantes junto con los planes para la distribución y el servicio.
3. Descripciones del proceso. Contiene los procesos de desarrollo y de servicio a utilizar para el desarrollo y administración del producto.
4. Metas de calidad. Contiene las metas y planes de calidad para el producto. Incluyendo la identificación y justificación de los atributos de calidad importantes del producto.
5. Riesgos y gestión de riesgos. Contiene los riesgos clave que podrían afectar a la calidad del producto y las acciones para abordar estos riesgos.

Los planes de calidad obviamente difieren dependiendo del tamaño y del tipo de sistema que se desarrolle. Existe una amplia variedad de **atributos de calidad del software** potenciales a considerar en el proceso de planificación de la calidad. Ver atributos de calidad en la figura siguiente.

Seguridad	Comprensión	Portabilidad
Protección	Experimentación	Usabilidad
Fiabilidad	Adaptabilidad	Reutilización
Flexibilidad	Modularidad	Eficacia
Robustez	Complejidad	Aprendizaje

Atributos de la calidad del software

Puede ser que la eficacia sea primordial, por lo que será necesario sacrificar otros factores para alcanzarla. Esto se establece en el plan, y los ingenieros que trabajan en el desarrollo deben cooperar para lograrlo. El plan también define el proceso de evaluación de la calidad.

27.4 Control de la calidad

El control de la calidad implica vigilar el proceso de desarrollo de software para asegurar que se siguen los procedimientos y los estándares de garantía de calidad. En el proceso de control de calidad del proyecto se comprueba que las entregas cumplan los estándares definidos.

Existen dos enfoques complementarios que se utilizan para **comprobar la calidad de las entregas** de un proyecto:

1. Revisiones de la calidad donde el software, su documentación y los procesos utilizados en su desarrollo son revisados por un **grupo de personas**. Se encargan de comprobar que se han seguido los estándares del proyecto y el software y los documentos concuerdan con estos estándares. Se toma nota de las desviaciones de los estándares y se comunican al gestor del proyecto.
2. Valoración automática del software en la que el software y los documentos producidos se procesan por algún **programa** y se comparan con los estándares que se aplican a ese proyecto de desarrollo en particular. Esta valoración automática comprende una medida cuantitativa de algunos atributos del software.

27.4.1 Revisiones de la calidad

Las revisiones son el método más utilizado para validar la calidad de un proceso o de un producto. Involucran a un grupo de personas que examinan todo o parte del proceso software, los sistemas o su documentación asociada para descubrir problemas potenciales. Las conclusiones de la revisión se registran formalmente y se pasan al autor o a quien sea responsable de corregir los problemas descubiertos.

En la siguiente figura se listan los **tipos de revisiones**.

Inspecciones de diseño o programas	Detectar errores finos en los requerimientos, el diseño o el código. La revisión es conducida por una lista de verificación de los posibles errores.
Revisiones del progreso	Proveer información del progreso del proyecto útil para su gestión. Ésta es una revisión tanto del proceso como del producto y se refiere a costes, duración y planificación.
Revisiones de la calidad	Llevar a cabo un análisis técnico de los componentes del producto o documentación para encontrar diferencias entre la especificación y el diseño del componente, código y documentación, y para asegurar que se siguen los estándares de calidad definidos.

Tipos de revisiones

El equipo de revisiones debe tener un núcleo de tres o cuatro personas como revisores principales.

Uno debe ser el diseñador principal, el cual tendrá la responsabilidad de tomar las decisiones técnicas. Los revisores principales pueden invitar a otros miembros del proyecto, para que colaboren en la revisión.

Los documentos a revisar deben distribuirse con anterioridad a la revisión para dar tiempo a los revisores a que los lean y los comprendan.

La revisión misma es relativamente corta (dos horas a lo más). Un miembro del equipo preside la revisión y otro registra formalmente todas las decisiones de la revisión. Este registro pasará a formar parte de la documentación formal del proyecto. El presidente es responsable de asegurar que se hagan todos los cambios requeridos. Si se requieren cambios importantes, habrá que hacer un seguimiento posterior de la revisión.

27.5 Medición y métricas del software

Sería posible acelerar el proceso de revisión utilizando herramientas que procesaran el diseño del software o el programa, e hiciesen valoraciones automáticas de la calidad del software. Estas valoraciones permiten comprobar que el software tiene el umbral de calidad requerido, y destacar las partes en las cuales no se ha alcanzado para revisarlas.

La **medición del software** se refiere a derivar un valor numérico desde algún atributo del software o del proceso software. Comparando estos valores entre sí y con los estándares aplicados en la organización, es posible sacar conclusiones de la calidad del software o de los procesos para desarrollarlo.

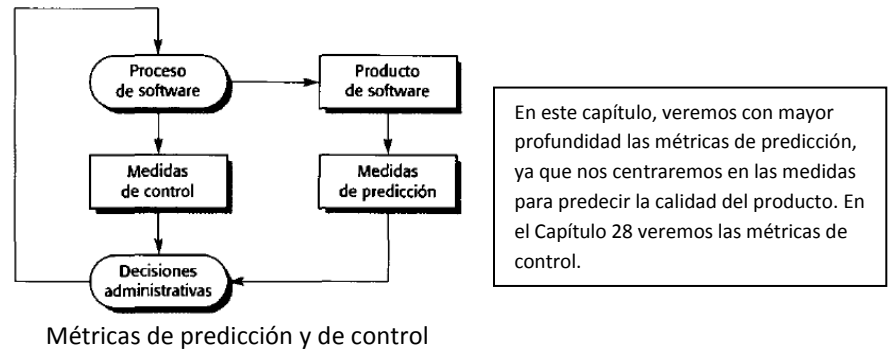
Las **mediciones del software pueden utilizarse para:**

1. Hacer predicciones generales acerca del sistema.
2. Identificar componentes anómalos.

Una **métrica de software** es cualquier tipo de medida relacionada con un sistema, proceso o documentación de software. Algunos ejemplos son las medidas que se utilizan para calcular el tamaño de un producto en líneas de código; el índice de Fog, que mide la claridad de un párrafo en un texto; el número de fallos encontrados en un producto software entregado; y el número de personas/día requeridas para desarrollar un componente del sistema.

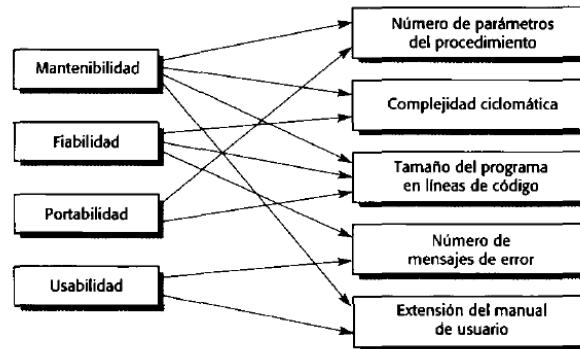
Las métricas son de control o de predicción. Las **métricas de control** suelen estar asociadas con los **procesos**, mientras que las **métricas de predicción** lo están a los **productos**. Ejemplos de las métricas de control o de procesos son el esfuerzo y el tiempo promedio requeridos para reparar los defectos encontrados. Ejemplos de métricas de predicción son la complejidad ciclomática de un módulo, la longitud media de los identificadores de un programa, y el número de atributos y operaciones asociadas con los objetos de un diseño.

Ambas pueden influir en la toma de decisiones de gestión como muestra la Figura siguiente.



Frecuentemente, es imposible medir los atributos de calidad del software directamente. Los atributos de calidad como la mantenibilidad, la comprensión y la usabilidad son atributos externos que nos dicen cómo ven el software los desarrolladores y los usuarios. Éstos se ven afectados por diversos factores y no existe un camino simple para medirlos. Más bien es necesario medir atributos internos del software (como su tamaño) y suponer que existe una relación entre lo que queremos medir y lo que queremos saber.

La siguiente figura muestra algunos atributos externos de la calidad que nos serán interesantes y los atributos internos relativos a ellos.



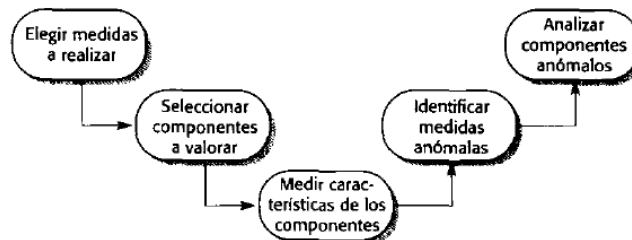
Relaciones entre los atributos internos y externos del software

Para que la medida del atributo interno sea un indicador útil de la característica externa, se deben cumplir tres **condiciones**:

1. El atributo interno debe medirse de forma precisa.
2. Debe existir una relación entre lo que se puede medir y el atributo de comportamiento externo.
3. Esta relación se comprende, ha sido validada y se puede expresar en términos de una fórmula o modelo.

27.5.1 El proceso de medición

Representación del proceso de medición del software dentro de un proceso de control de calidad:



Cada uno de los componentes del sistema se analiza por separado y los diversos valores de las métricas se comparan entre sí y, quizás, con los datos históricos de medición recogidos en los proyectos previos. Las medidas anómalas se utilizan para centrar el esfuerzo de garantía de calidad en los componentes que tienen problemas de calidad.

Los **pasos** clave en este proceso son:

1. Seleccionar las medidas a realizar. Se deben formular las preguntas que la medición intenta responder y definir las mediciones requeridas para resolver estas preguntas.
2. Seleccionar los componentes a evaluar. Se elige un conjunto representativo de componentes.
3. Medir las características de los componentes. Se miden los componentes seleccionados y se calculan los valores de las métricas.
4. Identificar las mediciones anómalas. Una vez que se obtienen las mediciones de los componentes, se comparan entre sí y con las mediciones previas registradas en una base de datos de mediciones.
5. Analizar los componentes anómalos. Una vez identificados los componentes con valores anómalos para, las métricas particulares, se examinan estos componentes para decidir si los valores de la métrica indican que la calidad del componente está en peligro.

27.5.2 Métricas de producto

Las métricas de producto se refieren a las características del software mismo. Desafortunadamente, las características del software que se miden fácilmente, como el tamaño y la complejidad ciclomática, no tienen una relación clara y consistente con los atributos de calidad como la compresión y la mantenibilidad. Las relaciones varían dependiendo de los procesos, la tecnología y el tipo de sistemas a desarrollar.

Las **métricas del producto se dividen en dos clases**:

1. Las métricas dinámicas, que son recogidas por las mediciones hechas en un programa en ejecución.
2. Las métricas estáticas, que son recogidas por las mediciones hechas en las representaciones del sistema como el diseño, el programa o la documentación.

Las **métricas dinámicas** ayudan a valorar la eficiencia y la fiabilidad de un programa y por lo general están relacionadas de forma cercana con los atributos de calidad del software. Las **métricas estáticas** ayudan a valorar la complejidad, la comprensión y la mantenibilidad de un sistema de software; por lo general están relacionadas de forma cercana con los atributos de calidad del software.

La siguiente figura describe varias métricas estáticas utilizadas para valorar los atributos de calidad.

Fan-in/Fan-out	Fan-in es una medida del número de funciones o métodos que llaman a otra función o método (por ejemplo, X). Fan-out es el número de funciones que son llamadas por una función X. Un valor alto de fan significa que X está fuertemente acoplada al resto del diseño y que los cambios en X tendrán muchos efectos importantes. Un valor alto de fan-out sugiere que la complejidad de X podría ser alta debido a la complejidad de la lógica de control necesaria para coordinar los componentes llamados.
Longitud del código	Ésta es una medida del tamaño del programa. Generalmente, cuanto más grande sea el tamaño del código de un componente, más complejo y susceptible de errores será el componente. La longitud del código ha mostrado ser la métrica más fiable para predecir errores en los componentes.
Complejidad ciclomática	Ésta es una medida de la complejidad del control de un programa. Esta complejidad del control está relacionada con la comprensión del programa. El cálculo de la complejidad ciclomática se trata en el Capítulo 22.
Longitud de los identificadores	Es una medida de la longitud promedio de los diferentes identificadores en un programa. Cuanto más grande sea la longitud de los identificadores, más probable será que tengan significado; por lo tanto, el programa será más comprensible.
Profundidad del anidamiento de las condicionales	Ésta es una medida de la profundidad de anidamiento de las instrucciones condicionales «if» en un programa. Muchas condiciones anidadas son difíciles de comprender y son potencialmente susceptibles de errores.
Índice de Fog	Ésta es una medida de la longitud promedio de las palabras y las frases en los documentos. Cuanto más grande sea el índice de Fog, el documento será más difícil de comprender.

Figura 27.12 Métricas estáticas de producto software.

Las métricas relevantes dependen del proyecto, las metas del equipo de gestión de calidad y el tipo de software a desarrollar.

27.5.3 Análisis de las mediciones

Uno de los problemas con la recogida de datos cuantitativos en el software y en los proyectos de software es comprender lo que significan realmente los datos. Es fácil malinterpretar los datos y hacer inferencias incorrectas. Las mediciones se deben analizar cuidadosamente para comprender lo que realmente significan.

Los procesos y productos para medir no están aislados de su entorno y los cambios en ese entorno invalidan las comparaciones de los datos. Los datos cuantitativos de las actividades humanas no siempre pueden tomarse como valores de entrada.

PU NTOS CLAVE

- La gestión de la calidad del software permite señalar si éste tiene un escaso número de defectos y si alcanza los estándares requeridos de mantenibilidad, fiabilidad, portabilidad, etcétera, las actividades de la gestión de la calidad comprenden la garantía de la calidad que establece los estándares para el desarrollo de software, la planificación de la calidad y el control de la calidad que comprueba el software con respecto a los estándares definidos.
- Un manual de calidad organizacional debe documentar un conjunto de procedimientos de garantía de la calidad. Éste puede basarse en los modelos genéricos sugeridos en los estándares ISO 9000.
- Los estándares de software son importantes para garantizar la calidad puesto que representan una identificación de las «mejores prácticas». El proceso de control de calidad implica comprobar que el proceso del software y el software a desarrollar concuerdan con estos estándares.
- Las revisiones de los productos a entregar por el proceso del software incumben a un equipo de personas los cuales comprobarán que se han seguido los estándares de calidad, las revisiones son la técnica más utilizada para valorar la calidad.
- Las mediciones de software se utilizan para recoger datos cuantitativos acerca del software y sus procesos, los valores de las métricas de software recogidas se utilizan para hacer inferencias de la calidad del producto y del proceso.
- Las métricas de calidad del producto son de gran valor para resaltar los componentes anómalos que tienen problemas de calidad. Estos componentes se deberán analizar con más detalle.
- No existen métricas de software estandarizadas y aplicables universalmente. las organizaciones deben seleccionar métricas y analizar mediciones basadas en el conocimiento y circunstancias locales.