

Modelo de Madurez para el Proceso de Aseguramiento de la Calidad del Software (SQA)

Patricio Rojas V. y Marcello Visconti Z. ()*

Departamento de Informática

Universidad Técnica Federico Santa María

Valparaíso, Chile

email: { projasv, visconti }@inf.utfsm.cl

Resumen

Un objetivo básico de la *Ingeniería de Software* es producir el mejor software posible. Sin embargo, el énfasis ha estado normalmente en la producción de código y no en las demás fases del desarrollo y mantenimiento de un producto de software (análisis, diseño, *testing*, revisiones, entre otros), las cuales son también importantes.

La experiencia ha mostrado la estrecha relación entre la calidad de dichos procesos y la calidad de los productos de software construidos, lo que motiva un cambio de enfoque para mejorar la calidad. La clave está en lograr una visión global de la forma en que el software es desarrollado y mantenido para establecer planes de mejoramiento. El principio se basa entonces, en mejorar la calidad del producto de software por medio de mejorar la calidad de los procesos de software que se utilizan.

En el presente artículo se aborda este tema a través del desarrollo de un modelo de madurez para el proceso de Aseguramiento de la Calidad del Software (*Software Quality Assurance*, SQA).

Palabras Claves: proceso de software, madurez de procesos, aseguramiento de la calidad del software.

(*) Trabajo de investigación financiado por el Fondo de Ciencia y Tecnología de CONICYT-CHILE, mediante proyecto FONDECYT N° 1950708.

1. Introducción

El proceso de Aseguramiento de la Calidad del Software es una disciplina de la Ingeniería de Software que se especializa en la aplicación de procesos de calidad a lo largo del proyecto de software. El proceso de SQA tiene a su cargo una gran cantidad de tareas [1], entre las que destacan: revisiones y auditorías, *testing*, estándares y procedimientos, administración del riesgo, entre otros. Estas actividades permiten asegurar que tanto los productos como los procesos de desarrollo y de control son correctamente ejecutados de acuerdo a procedimientos y estándares establecidos.

En el último tiempo el proceso de SQA ha tenido un gran auge dentro de la Ingeniería de Software, debido principalmente a tres factores importantes[2]:

- la realidad de un mercado global intensamente competitivo obliga a las empresas productoras de software a entregar productos de gran calidad
- las aplicaciones actuales son cada vez más complejas
- los clientes exigen un elevado nivel de calidad del producto que adquieren.

De acuerdo a estos factores es que el proceso de SQA emerge como una solución para lograr mejoras sustanciales en la calidad del producto y de los procesos involucrados en su desarrollo. Por tal motivo, el proceso de SQA debe estar inserto en el proceso de software de una organización. La misión que cumple el proceso de SQA no se limita solamente a actividades de verificación, sino que además asume un rol de liderazgo en la administración de la calidad durante el proceso de creación, diseño y mantención del producto de software. Es importante, por lo tanto, conocer qué tan eficazmente está siendo ejecutado, por la organización, el proceso de SQA.

Modelo de Madurez: Concepto de Madurez y *Capability Maturity Model (CMM)*

La Ingeniería de Software ha introducido últimamente la premisa que la calidad de un sistema de software está fundamentalmente determinado por los procesos utilizados para construir dicho producto [3].

Dentro de este marco de referencia, se define como una organización madura aquella que posee la potencialidad para desarrollar y administrar sus proyectos de software en forma consistente y proactiva, así como también la capacidad para mantener y mejorar sus procesos.

Una organización madura se caracteriza por mantener procesos consistentes a través de sus distintos proyectos. La aplicación de estos procesos debe resultar en una disminución de las crisis y de los costos anormales que afectan a una organización inmadura. La gerencia deja de estar ocupada casi exclusivamente enfrentando emergencias, de

manera que pueda ocuparse de monitorear la calidad del producto y la satisfacción del cliente. Los ingenieros de software pueden concentrarse en su trabajo sin ser distraídos en "apagar incendios" a cada momento. La organización que ha logrado un cierto nivel de madurez de procesos puede controlar sus proyectos razonablemente y ahorrar los costos inherentes al proceso caótico, aprovechando así los beneficios que brinda la calidad.

Las ideas y conceptos fundamentales de la madurez organizacional fueron planteados por el *Software Engineering Institute* (SEI) de la *Universidad de Carnegie Mellon*, EE.UU. a través de un Marco Conceptual [3] y más tarde a través del modelo de madurez de los procesos de software (*Capability Maturity Model*, CMM) [4].

El modelo de madurez describe un plan evolutivo de mejoramiento para pasar de un proceso ad-hoc o caótico a uno maduro. El modelo también establece ciertas características estándares contra las cuales se puede juzgar formalmente la madurez de una organización relativa a sus procesos de software. El modelo debe ser usado conjuntamente con un procedimiento de evaluación que permita mapear el estado de una organización a uno de los niveles de madurez y determinar, de esta forma, los desafíos y las potenciales áreas de mejoramiento para pasar al siguiente nivel de madurez.

Problema Bajo Estudio

Aplicar los conceptos de madurez al proceso de SQA es un gran paso para conocer qué tan eficazmente está siendo ejecutado este proceso en una organización. Sin embargo, aplicar el CMM al proceso de SQA no es conveniente ya que este modelo describe una potencialidad global en base a un proceso de desarrollo que involucra a todas las actividades responsables por la producción de un producto de software (análisis de requerimientos, diseño, codificación, *testing*, entre otros). De esta manera, la madurez corresponde a una visión global del "estado de la práctica" en que se encuentra dicha organización con respecto a su proceso de software. Una de las críticas principales que ha recibido el CMM es justamente esta situación; que un simple nivel de madurez no puede describir completamente el proceso de software de una organización [5]. Existe la posibilidad de que ciertas actividades dentro del proceso de software no alcancen el nivel de madurez que lo representa o viceversa.

La solución a este problema parte por desarrollar un modelo de madurez para el proceso de SQA.

El objetivo principal de este modelo de madurez es tener una visión particular del proceso de SQA inserto en el ciclo de vida de un producto de software, de manera de tener un mayor conocimiento de la potencialidad de dicho proceso.

2. Modelo de Madurez para el Proceso de Aseguramiento de la Calidad del Software

El modelo propuesto para el proceso de Aseguramiento de la Calidad del Software está compuesto por 5 niveles de madurez más un nivel base o de inmadurez. Estos niveles describen los distintos escenarios de madurez en que se desenvuelven las organizaciones que desarrollan o mantienen software con respecto a sus procesos de calidad.

Niveles de Madurez del Proceso de SQA

Nivel Cero: Inmaduro

En este nivel de madurez, las organizaciones que desarrollan o mantienen software no poseen una “cultura de calidad” en su ambiente organizacional. Por lo tanto, no existe un compromiso de la alta administración por aplicar los conceptos y prácticas de calidad en sus procesos de desarrollo y mantención del software. Resultado de ello es la carencia de procedimientos documentados que guíen la ejecución de prácticas de calidad tales como *testing*, inspecciones de código, revisiones de requerimientos, entre otros. Estas prácticas son realizadas en forma improvisada por los ingenieros de software.

No hay una adecuada definición de hitos de calidad (prácticas de calidad) a lo largo de un proyecto de software que permitan revisar y/o auditar las actividades y productos de cada fase del ciclo de vida del software. La carencia de apropiadas prácticas de calidad tales como revisiones e inspecciones a lo largo de un proyecto de software es un factor crítico de estas organizaciones inmaduras.

Las prácticas de calidad dominantes en este nivel corresponden a *testing* y *debugging*, siendo éstos los únicos motores del proceso de SQA en este nivel. Sin embargo, estas prácticas no están documentadas y no siguen un procedimiento establecido y corresponden a prácticas reactivas o “*por fuerza bruta*” que se utilizan solamente para resolver problemas inmediatos o para “*apagar incendios*” ante eventuales crisis que resultan de la carencia de una planificación adecuada y de objetivos de calidad claramente definidos.

Los resultados de eventuales pruebas al software no se registran (no hay captura del conocimiento), por lo tanto carecen de la capacidad de aprender de sus propios errores en la aplicación de los procesos de *testing*.

Desafíos para resolver estos problemas y avanzar al siguiente nivel de madurez:

- Establecimiento de hitos de calidad a lo largo del ciclo de vida de cada proyecto de software.
- Definición y documentación de procedimientos para las prácticas de calidad utilizadas en un proyecto de software.

Primer Nivel: Básico

Se comienza a tener una conciencia de la importancia o del rol que cumple la calidad en los procesos de software tanto para las organizaciones que desarrollan o mantienen software como para los clientes. Estos indicios de una "cultura de calidad" se manifiestan en la definición y planificación temprana de prácticas de calidad tales como *testing*, la incorporación de revisiones tanto para la planificación y requerimientos como para el diseño y codificación, y el registro de los resultados de las prácticas de calidad.

La existencia de los procesos de revisión o inspección a lo largo del ciclo de vida de un proyecto tiene por objeto minimizar, detectar y corregir tempranamente defectos del producto de software y revisar los requerimientos del cliente con la intención de definir en forma clara y completa sus necesidades. Las revisiones de requerimientos son realizadas conjuntamente entre el cliente y los desarrolladores del producto en etapas tempranas del ciclo de vida. Las necesidades del usuario son revisadas a lo largo del proceso de desarrollo con la intención de satisfacer correctamente los aspectos de funcionalidad y calidad del producto de software, tales como amistosidad, confiabilidad, usabilidad, entre otros.

Desafíos para este nivel:

- Desarrollo de un plan global de SQA que integre las prácticas de calidad.
- Estandarización de las prácticas de calidad.
- Implantación de un equipo independiente de SQA.

Segundo Nivel: Planificado

En este nivel de madurez, las organizaciones que desarrollan o mantienen software poseen una "cultura de calidad" que se traduce en la existencia de un plan de Aseguramiento de la Calidad y de un compromiso de la alta administración por orientar hacia una "cultura de la calidad" del software. El plan de Aseguramiento de la Calidad del Software define claramente las fases del desarrollo del producto con sus respectivas prácticas de calidad, estandariza la documentación de cada sub-producto de las fases del desarrollo, estandariza las prácticas de

revisión, inspecciones, auditorías, *testing* y "*capturas de conocimiento*" en forma de guías y *checklist* básicos (tipos de defectos).

Las organizaciones de este nivel realizan regularmente auditorías internas y/o externas con el objetivo de evaluar objetivamente el estado de las prácticas de calidad.

Existe un grupo de ingenieros de software dedicados exclusivamente a la gestión de la calidad. Este grupo es independiente de la gerencia del proyecto, a objeto de garantizar su objetividad y permitirle la libertad necesaria para señalar potenciales problemas tanto en los procesos como en el producto de software.

Las organizaciones que se encuentran en este nivel utilizan en sus proyectos de software tecnología CASE orientada a controlar y apoyar las metodologías utilizadas en el desarrollo del producto.

Desafíos para este nivel corresponden a:

- Definición explícita del proceso de SQA.
- Estandarización y documentación del proceso de SQA.
- Definición de prácticas de calidad como procesos (subprocesos).

Tercer Nivel: Estandarizado

Las organizaciones que se encuentran en este nivel definen explícitamente el proceso de Aseguramiento de la Calidad del Software. Este proceso de SQA se transforma en un sistema de calidad estándar para la organización, el cual es integrado a su proceso de software.

Se definen objetivos globales de calidad para el proceso de SQA. Además, se establecen claramente las responsabilidades y compromisos de la administración superior con respecto al proceso de SQA.

Las prácticas de calidad se definen y documentan de acuerdo a un enfoque de procesos con sus respectivos objetivos y compromisos.

Desafíos para este nivel:

- Establecimiento de un programa de medición.
- Definición de metas cuantitativas para los procesos de calidad.

Cuarto Nivel: Controlado

Se establece un plan de medición con el objetivo de controlar y comprender cuantitativamente el proceso de Aseguramiento de Calidad del Software. Este plan define métricas para el producto, procesos, estándares y procedimientos, describe cómo usarlas y explica procedimientos para recolectarlas

Las herramientas utilizadas por estas organizaciones están orientadas a monitorear y controlar la eficacia de los procesos de calidad del software.

Un programa de medición provee un esencial *feedback*, el cual permite guiar a las organizaciones en la mejora de sus procesos de calidad. Las organizaciones con esta capacidad se encuentran en el último nivel (*optimizado*) de este modelo propuesto.

Desafíos para este nivel:

- Mejoramiento de los procesos de calidad.
- Alto nivel de automatización a través de ambientes inteligentes.

Quinto Nivel: Optimizado

Las organizaciones que se encuentran en este nivel mejoran o perfeccionan en forma continua el proceso de SQA, con el objetivo de lograr procesos más efectivos para lograr la calidad del producto de software.

Existe un compromiso organizacional por orientar hacia un mejoramiento continuo de los procesos de calidad del software.

Existe una alta capacidad de automatización que involucra "*ambientes inteligentes*", que permiten perfeccionar y mejorar los procesos de software a través de la capacidad de aprender de sus propios errores.

Desafíos para este nivel:

- Mejoramiento continuo de los procesos de calidad.

Prácticas Claves del Modelo de Madurez del Proceso de SQA

Nivel 0: Inmaduro

Palabras Claves: prácticas improvisadas o *ad-hoc*, carencia cultura de calidad, *testing* reactivo.

- No hay prácticas claves de calidad en este nivel.

Nivel 1: Básico

Palabras Claves: hitos de calidad, procedimientos documentados, *testing* documentado, revisiones y/o inspecciones documentadas, registro de pruebas.

Objetivos:

- Monitoreo de la calidad de los productos a lo largo de un proyecto de software.
- Ejecución en forma consistente y pro-activa de las prácticas de calidad.

Prácticas:

- Definición y planificación de hitos de calidad a lo largo de un proyecto de software.
- Definición de procedimientos a seguir por las prácticas de calidad.
- Documentación de prácticas de calidad y de sus resultados.

Nivel 2: Planificado

Palabras Claves: integración de prácticas, plan de SQA, grupo independiente de SQA, auditorías internas y/o externas, cultura de calidad, compromiso organizacional, estándares y procedimientos, tecnología CASE.

Objetivos:

- Establecimiento de un compromiso organizacional con respecto a la calidad.
- Definición de un plan global que integre las prácticas de calidad.
- Implantación de un grupo independiente de SQA.

Nivel 5: Optimizado

Palabras Claves: retroalimentación, ambientes inteligentes, mejoramiento de procesos.

Objetivos:

- Mejoramiento continuo del proceso de SQA.
- Alta automatización a través de ambientes inteligentes.

Prácticas:

- Definición de un programa de mejoramiento continuo para el proceso de SQA.
- Utilización de ambientes inteligentes para apoyar el plan de mejoramiento.

3. Contraste entre CMM y Modelo de Madurez del Proceso de SQA

La siguiente figura muestra esquemáticamente el contraste entre estos dos modelos.

Nivel madurez modelo CMM	Capacidad proceso de software	Nivel madurez modelo proceso SQA	Capacidad proceso SQA
1	caótica	0	inmadura
2	disciplinada	1	básica
		2	planificada
3	estándar y consistente	3	estandarizada
		4	controlada
5	mejoramiento continuo	5	mejoramiento continuo

grado de aplicación de la capacidad del proceso de software al proceso de SQA.

La forma de realizar este contraste entre ambos modelos es el siguiente: primero, se define el marco global de un proceso de software de una organización, es decir "se fija" el nivel de madurez del proceso de software (CMM). Enseguida, teniendo en cuenta este marco, es posible visualizar hasta qué nivel de madurez puede alcanzar el proceso de SQA. La potencialidad del proceso de software puede ser transferida a su proceso de SQA en menor o

Prácticas:

- Establecimiento de compromisos y participación de toda la organización con respecto a la calidad.
- Definición de un plan de Aseguramiento de la Calidad del Software.
- Implantación de un grupo independiente de SQA.
- Utilización de herramientas CASE para apoyar a las actividades de SQA.

Nivel 3: Estandarizado

Palabras Claves: enfoque de procesos, proceso de SQA estandarizado y documentado.

Objetivos:

- Definición y estandarización explícita del proceso de SQA.

Prácticas:

- Establecimiento de compromisos y responsabilidades con respecto al proceso de SQA.
- Definición y estandarización del proceso de SQA.

Nivel 4: Controlado

Palabras Claves: programa de medición, control de procesos de calidad, herramientas orientadas a los procesos.

Objetivos:

- Monitoreo y control de los procesos de calidad mediante métricas y metas cuantitativas.

Prácticas:

- Definición de un plan de medición.
- Definición de metas cuantitativas para el proceso de SQA.
- Utilización de herramientas de apoyo a fin de monitorear la eficacia de los procesos de calidad.

mayor medida. Este grado de aplicación de la capacidad del proceso de software origina los desniveles que se muestran en el esquema anterior.

El contraste anterior entre CMM y el modelo de madurez del proceso de SQA, muestra que para conocer la potencialidad del proceso de SQA no basta con aplicar solamente el CMM a este proceso, ya que este modelo describe la madurez cuya potencialidad representa al proceso de software de una organización y no a un sub-proceso en particular inserto en el ciclo de vida del software. El CMM permite "rayar la cancha" y con ello conocer hasta qué nivel de capacidad puede alcanzar los sub-procesos involucrados en el proceso de software. El modelo de madurez para el proceso de SQA permite conocer, en forma más particular, la potencialidad del proceso de SQA y además permite conocer qué tan eficazmente está siendo ejecutado por la organización.

4. Propuesta de Herramienta de Evaluación del Modelo de Madurez para SQA

El propósito de una herramienta de evaluación es determinar en qué nivel de madurez se encuentra el proceso de Aseguramiento de la Calidad del Software de una organización. Para este modelo, se propone un cuestionario de evaluación como herramienta.

La filosofía de evaluación es la de determinar si un proceso de SQA cumple con los *objetivos* de un nivel de madurez del modelo. Cada objetivo se concreta en base a *prácticas*, las cuales corresponden a actividades que deben ser realizadas en conjunto para satisfacer dicho objetivo. Para cada práctica se pueden formular una o más *preguntas* que permiten establecer, mediante sus respuestas, si éstas han sido implantadas o institucionalizadas en el proceso de Aseguramiento de la Calidad del Software de una organización. Exceptuando el nivel 0, todos los niveles tienen sus respectivas preguntas.

Para avanzar hacia un nivel más alto en el modelo, el proceso de SQA no sólo debe cumplir con los objetivos del nivel en cuestión, sino que también debe cumplir con los objetivos de los niveles inferiores, es decir, debe haber estado en el nivel inmediatamente anterior al nivel que se desea llegar. Y esto se cumple para todos los niveles del modelo de madurez del proceso de SQA, exceptuando por cierto, el nivel 0 o de inmadurez.

Cumpliendo los desafíos de cada nivel se está en condiciones de moverse al siguiente nivel de madurez en la escala del modelo para el proceso de SQA.

En la actualidad, la herramienta de evaluación se encuentra en desarrollo y su versión preliminar consulta 54 preguntas.

5. Conclusiones y Trabajo Futuro Potencial

El modelo de madurez del proceso de SQA es una propuesta que presenta un marco conceptual para mejorar las actividades de calidad insertas en una organización dedicada al desarrollo y mantenimiento del software. Sin embargo, el modelo no garantiza el éxito de lograr productos de gran calidad o procesos de calidad más eficaces. Para lograr estas metas, debe existir un compromiso real de toda la organización con respecto a la calidad y se debe aunar esfuerzos para alcanzar las metas propuestas por la organización.

El modelo de madurez para el proceso de SQA es una versión preliminar, el cual se irá mejorando a través de la retroalimentación de la información obtenida de futuras evaluaciones a organizaciones.

CMM y el modelo de madurez del proceso de SQA se complementan: mientras uno permite tener un conocimiento global del proceso de software de una organización, el otro permite tener una visión más particular de un subproceso involucrado en él, en este caso, el proceso de Aseguramiento de la Calidad del Software.

La validación del modelo de madurez del proceso de SQA y su herramienta de evaluación es uno de los trabajos futuros potenciales. La validación persigue mostrar en forma objetiva que organizaciones con un alto nivel de madurez del proceso de SQA producen software de alta calidad. Otros trabajos potenciales futuros son el de desarrollar herramientas automatizadas para el modelo de madurez y procedimientos de evaluación, y desarrollar modelos de madurez para otras fases del proceso de software tales como análisis, diseño, implementación, *testing*, mantenimiento, entre otros.

Bibliografía

- [1] Sanders, J., Curran, E., "Software Quality, a Framework for Success in Software Development and Support", ADDISON-WESLEY, 1994, pp 109-130.
- [2] Basili, V., Rombach, H., "Implementing Quantitative SQA: A Practical Model", IEEE Software, Septiembre 1987, pp 6-9.
- [3] Humphrey Watts S., "Characterizing the Software Process: A Maturity Framework", IEEE Software, Marzo 1988, pp 73-79.
- [4] Paulk Mark C., Curtis Bill, y Chrissis Mary Beth, "Capability Maturity Model, Versión 1.1", IEEE Software, Julio 1993, pp 18-27.
- [5] Topper Andrew y Forgensen Paul, "More than One Way to Measure Process Maturity", IEEE Software, Noviembre 1991, pp 9-10.