

CURSO DE POSGRADO

Técnicas y Gestión de las Pruebas de Software

Darío Macchi

DOCENTE (invitado)

Estrategia y plan de mejora de pruebas

Madurez

Un proceso de testing maduro:

- Satisface las necesidades de las partes interesadas
- Garantiza la entrega estable de un producto de calidad

Señales reveladoras de inmadurez:

- Quejas de los clientes
- Retrasos y problemas de calidad en proyectos

Madurez

¿Qué sucede si un cliente no está satisfecho?

- Medidas rápidas pueden no ser efectivas
- Un proceso maduro puede marcar la diferencia

¿Cómo?

- Proporciona patrones de comportamiento y actuación
- Ayuda a los equipos a resolver problemas de manera eficaz
- Reduce el tiempo y los efectos negativos

Modelos de madurez de la Gestión de la Calidad (GC)

Generalidades

Tipos de modelos

Dos tipos de modelos de madurez:

1. Modelos personalizados
2. Modelos reconocidos en el mundo

Ejemplos:

- Una empresa de desarrollo de software para el sector automotriz puede optar por un modelo personalizado.
- Un proveedor de pruebas suele elegir un modelo reconocido mundialmente, ya que puede aplicarse a una amplia gama de proyectos.

Modelos personalizados

Se adaptan a las particularidades empresariales, objetivos y clientes de una organización.

Principal ventaja: se ajustan perfectamente a las especificidades de la empresa

Principales inconvenientes:

1. Tiempo/esfuerzos adicionales para su uso en proyectos con otros proveedores o contratación de especialistas.
2. Aumenta el plazo de comercialización y los costes del proyecto.

Modelos reconocidos en el mundo

Los modelos reconocidos a nivel mundial permiten adaptarse a diferentes empresas y facilitan la cooperación con clientes, proveedores y empleados.

1. Adaptabilidad a una amplia gama de empresas.
2. Facilita la cooperación con clientes, proveedores y empleados.
3. Enfoque en las pruebas de software para garantizar un producto de calidad.

Test Maturity Model Integration

- TMMi es el modelo más común y utilizado en pruebas de software y control de calidad.
- es una adaptación del Modelo de Madurez de Capacidades (CMM) para pruebas de software
- Define criterios claros para cada nivel de madurez y requiere que se cumplan todas las áreas de proceso antes de avanzar al siguiente nivel.

Nivel 1 (Inicial)

En este nivel, las pruebas son ad hoc, no estructuradas y frenéticas.

1. Producto lento e inestable.
2. Las pruebas solo son auxiliares para la depuración.
3. No hay un proceso de pruebas documentado.

Nivel 2 (Repetible)

Equipos de pruebas tienen procedimientos establecidos y documentados, para repetir procesos y facilitar incorporación y resolución de problemas futuros.

1. Niveles de prueba (de humo, de módulos, de integración, de regresión, de aceptación del usuario)
2. Características del proceso:
 - Repetible y bien documentado
 - Las pruebas se realizan tarde en el SDLC

Nivel 3 (Definido)

Los equipos de pruebas de nivel 3 tienen un proceso documentado y comparten conocimientos y mejores prácticas.

1. Disponen de opciones de formación en el puesto de trabajo.
2. Las pruebas comienzan en la fase de recopilación de requisitos.
3. Se realizan sesiones de formación.
4. Se aplican enfoques de pruebas específicos del proyecto.

Nivel 4 (Medido)

En el nivel 4, los equipos de pruebas necesitan medidas cuantitativas de los procesos para identificar áreas problemáticas y mejorarlos.

1. Propiedades de calidad del producto adquieren equivalentes cuantitativos.
2. Examinar cómo los miembros del equipo siguen las prácticas adoptadas.
3. Las pruebas son un proceso medido.
4. Revisiones e inspecciones periódicas.
5. El equipo se esfuerza por hacer las pruebas más eficaces.

Nivel 5 (Optimización)

Los equipos de pruebas de nivel 5 tienen enfoques y prácticas bien afinados, pero buscan constantemente mejorar su proceso.

1. Enfoques y prácticas de pruebas bien afinados.
2. Mejora continua del proceso de pruebas para:
 - a. obtener pruebas más rápidas
 - b. plazos de comercialización más cortos
 - c. costes reducidos sin dañar la calidad del producto.
3. Logro estable de objetivos: encontrar errores y completar el proyecto a tiempo.

Test Process Improvement

El modelo TPI proporciona una visión clara de madurez del proceso de control de calidad y ayuda a mejorar pruebas. Para ello, se definen **20 áreas clave** que describen las **4 piedras angulares** del proceso de pruebas:

1. **Ciclo de vida (L)**: actividades de pruebas en conexión con el ciclo de desarrollo.
2. **Organización (O)**: importancia de una buena calidad en la organización.
3. **Infraestructura y herramientas (I)**: necesidad de contar con las herramientas adecuadas.
4. **Técnicas (T)**: aplicación fácil a las actividades de pruebas.

TPI - Áreas clave y madurez

El equipo evalúa cada área clave y les asigna **niveles de madurez** A - D (A más bajo; D más alto).

- No todas las áreas clave tienen los mismos niveles.

Ejemplo: el entorno Office solo tiene una implantación nivel A, mientras que Metrics tiene cuatro niveles.

Las áreas clave también se clasifican en catorce **escalas de madurez**:

- Escalas 1 - 5 - Controlada
- Escalas 6 - 10 - Eficaz
- Escalas 11 - 13 - Avanzada

TPI

ID	Key area	0	1	2	3	4	5	6	7	8	9	10	11	12	13
		Ad-hoc	Controlled				Efficient				Optimizing				
L	Test strategy		A					B				C		D	
L	Lifecycle model		A			B									
L	Pervasiveness			A				B				C		D	
T	Estimation and planning				A							B			
T	Test specification techniques		A		B										
T	Static test techniques					A		B							
T	Metrics						A			B			C		D
T	Test tools					A			B			C			
I	Test environment			A					B						C
I	Office environment			A											
I	Commitment and motivation		A				B						C		
O	Test functions and trainings				A			B			C				
O	Scope of methodology					A						B			C
O	Communication			A		B							C		
O	Reporting		A			B		C					D		
O	Defect management		A				B		C						
O	Testware management						A				C				D
O	Test process management		A										C		
A	Evaluation							A			B				
A	Low-level testing					A		B		C					

TPI - Puntos de control

Un nivel de madurez superior requiere cumplir todos los requisitos de los niveles inferiores.

1. Los puntos de control son requisitos obligatorios para un determinado nivel (de A a D).
2. Los puntos de control ayudan al equipo a evaluar la calidad de las pruebas en la organización.

Recordar:

No todas las áreas clave pueden tener niveles de madurez igualmente altos, ya que áreas claves se priorizan según especificidades del proyecto.

Example: Checkpoints for Test tools key area.

Planning and control tools (level A)

Checkpoints:

- Automated tools (other than standard word processing) are used for the defect administration and for at least two other activities of planning and control.

Execution and analysis tools (level B)

Checkpoints:

- At least two sorts of automated tools are used for test execution, such as Capture and Playback tools, test coverage tools, etc.
- The test team has a general insight into the cost/profit ratio of these tools.

Extensive automation of the test process (level C)

Checkpoints:

- Automated tools (other than standard word processing) are used for the planning phase (for the activities estimating, planning, progress monitoring, configuration management, and defect administration), preparation, specification, and execution (in total at least five sorts of tools should be used).
- The test team has an insight into the cost/profit ratio of these tools.

TPI - Sugerencias de mejora

- Sugerencias e ideas que ayudarán a alcanzar un determinado nivel de madurez de las pruebas.
- A diferencia del uso de puntos de control, el uso de sugerencias de mejora no es obligatorio.
- Sin embargo, cada nivel se suministra con varias sugerencias de mejora.

Example: Improvement suggestions.

Key area Test strategy, level A, Test strategy for single high level test.

Improvement suggestions:

- Involve the various interested parties such as end user, systems manager, and project manager in determining the test strategy
- Create awareness by indicating the risks of the current working method, or indicate how testing can be done cheaper and/or faster.

TPi - Beneficios

- Muestra el estado actual y necesario de las pruebas
- Permite visualizar el estado del proyecto y elaborar un plan de mejoras
- Ayuda a alcanzar un mayor nivel de madurez
- Permite a las pequeñas empresas realizar una evaluación legítima
- La evaluación puede realizarse siempre que sea necesario
- Fomenta la mejora constante del proceso de pruebas
- Impulsa a alcanzar niveles superiores en calidad y eficiencia

Estrategia de mejoras

Mejoras en varios planos...

1. Procesos de Desarrollo de Software
2. Gestión de los Proyectos
3. Calidad de los Productos

1- Procesos de desarrollo

Áreas prioritarias de mejoras

1. Gestión de Requerimientos
2. Aseguramiento de la calidad
3. Pruebas de Software

1a - Gestión de Requerimientos

Mejoras inmediatas

- Documentar mejor !!
- Definir responsable único
- Establecer método de validación y firma (responsabilidad)
- Organizarse para la gestión de cambios
 - Word, versionado de documentos, MS-Sharepoint ...

1a - Gestión de Requerimientos

Mejoras a mediano plazo

- Formalizar el ciclo de vida
- Formalizar lenguaje y representación de requerimientos
- Medida de impacto de los cambios en requerimientos
- Uso de herramientas específicas
- Trazabilidad sistemática

1b - Aseguramiento de la calidad

Mejoras inmediatas

- Identificar función de Responsable Calidad (diferente de Testing)
 - En un contexto Ágil puede ser el responsable del Producto
- Definir objetivos de calidad del proyecto y del producto
- Definir modelo de desarrollo y método de validación (Secuencial, Ágil, Evolutivo...)
- Introducir control cruzado entre pares

1b - Aseguramiento de la calidad

Mejoras a mediano plazo

- Plan de Aseguramiento de la Calidad
- Instancias de validación formal
- Objetivos cualitativos y cuantitativos para el proyecto y el producto
 - Atributos de calidad, errores, fiabilidad, ...
- Certificación ISO / CMM-I

2 - Gestión de proyectos

Mejoras inmediatas

- Aplicar la noción de proyecto como espacio autónomo
- Fijar objetivos por proyecto
- Asignación de recursos y responsabilidades
- Introducir noción de gestión de riesgos

2 - Gestión de proyectos

Mejoras a mediano plazo

- Adopción de estándares
 - ITIL, PMI, ...
- Optimizar objetivos de satisfacción al cliente y eficiencia del proyecto
- Incluir evaluación de resultados a mediano plazo (mantenimiento)

3 - Calidad de los Productos

Mejoras a corto plazo

- Arquitecturas orientadas a componentes
- Modelización para software orientado a procesos (BPM – Workflow)
- Software evolutivo gracias a bajo impacto en la programación

3 - Calidad de los Productos

Mejoras a mediano plazo

- Construcción de herramientas, bibliotecas y frameworks probados
- Utilizar motores de BPM - Workflow
- Software evolutivo gracias a cambios en metadatos y configuración
- Introducir lenguajes específicos del dominio de aplicación (DSL)