

Aseguramiento de la Calidad del Software

Universidad Autónoma de Coahuila

Facultad de Sistemas

Calidad y Pruebas del Software

Carlos Nassif Trejo García

“Cliente interno”

Definiciones

- Calidad de diseño: Características que los diseñadores especifican para el producto final
- Calidad de conformidad: Grado en la cual se siguen las especificaciones de diseño
- Calidad de control: Inspecciones, revisiones y pruebas
- Aseguramiento de Calidad (QA): Auditoria y reportes de procedimientos

Costos

Prevención

- Plan de calidad
- Revisiones técnicas formales
- Equipo de prueba
- Entrenamiento

Apreciación:

- Inspecciones
- Calibración y mantenimiento de equipo
- Pruebas

Falla

- Re trabajo, reparación, Análisis de fallas

Fallas externas

- Resolución de queja, devolución, Help line, garantías

Incluye

Proceso de ACS

Tareas específicas

Métodos y herramientas

Control de desarrollo y cambios

Cumplimiento de estándares

Medición y reportes

Elementos

Estándares

Revisiones y auditorías

Pruebas

Colección y análisis

Administración del cambio

Educación

Administración de los proveedores

Administración de la seguridad

Seguridad

Administración de riesgos

Estandares



IEEE

*Advancing Technology
for Humanity*





Revisiones y auditorias



Pruebas



Colección y análisis de los errores



Administración del cambio



Educación



Administración de los proveedores



Paquetes contenidos en un caja

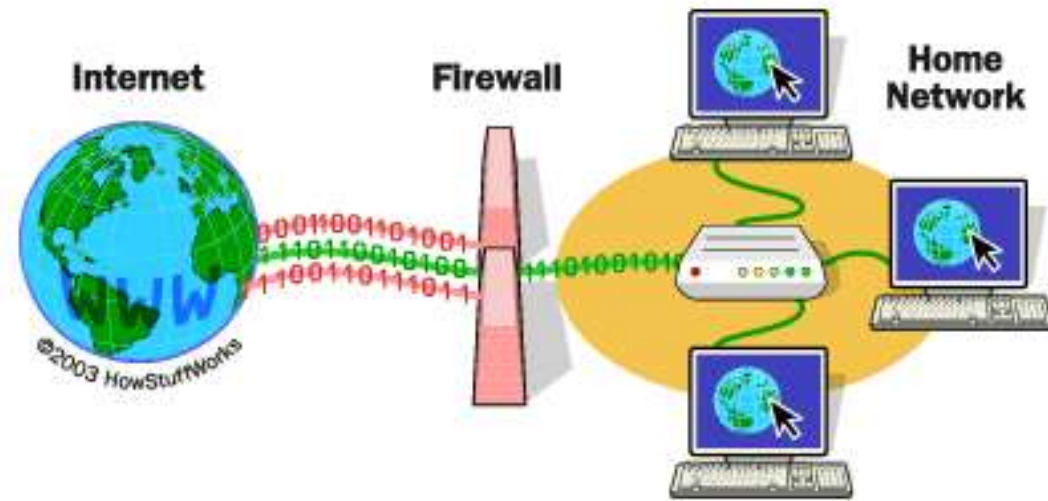


Shell personalizado



Software contratado

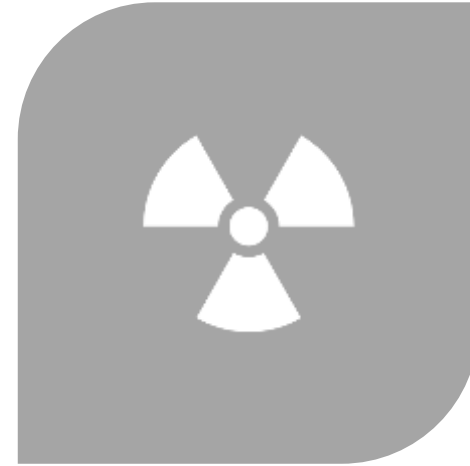
Administración de la seguridad



Seguridad



EFECTO DE FALLAS



DISMINUIR RIESGOS

Administración de riesgos



Asegura que se lleve acabo la
administración de riesgos



Planes de contingencia

Técnicas

Ingenieros de software

- Trabajo técnico

Grupo de ACS

- Planear
- Supervisar
- Registrar
- Analizar
- Reportar

Tareas del ACS

Preparar el plan de ACS para un proyecto

Participa en el desarrollo de la descripción del software del proyecto

Revisar actividades de los ingenieros de software

Audita productos de trabajo

Maneja desviaciones de trabajo

Registra y reporta fallas de cumplimiento

Meta	Atributo	Métrica
Calidad de los requerimientos	Ambigüedad	Número de modificadores ambiguos (por ejemplo, muchos, grande, amigable, etc.)
	Compleitud	Número de TBA y TBD
	Comprensibilidad	Número de secciones y subsecciones
	Volatilidad	Número de cambios por requerimiento
		Tiempo (por actividad) cuando se solicita un cambio
	Trazabilidad	Número de requerimientos no trazables hasta el diseño o código
	Claridad del modelo	Número de modelos UML
		Número de páginas descriptivas por modelo
		Número de errores de UML
Calidad del diseño	Integridad arquitectónica	Existencia del modelo arquitectónico
	Compleitud de componentes	Número de componentes que se siguen hasta el modelo arquitectónico
		Complejidad del diseño del procedimiento
	Complejidad de la interfaz	Número promedio de pasos para llegar a una función o contenido normal
		Distribución apropiada
Calidad del código	Patrones	Número de patrones utilizados
	Complejidad	Complejidad ciclomática
	Facilidad de mantenimiento	Factores de diseño (capítulo 8)
	Comprensibilidad	Porcentaje de comentarios internos
		Convenciones variables de nomenclatura
	Reusabilidad	Porcentaje de componentes reutilizados
	Documentación	Índice de legibilidad
	Asignación de recursos	Porcentaje de personal por hora y por actividad
	Tasa de finalización	Tiempo de terminación real versus lo planeado
	Eficacia de la revisión	Ver medición de la revisión (capítulo 14)
Eficacia del control de calidad	Eficacia de las pruebas	Número de errores de importancia crítica encontrados
		Esfuerzo requerido para corregir un error
		Origen del error

Aseguramiento estadístico

1. Se recaba y clasifica la información acerca de errores y defectos del software.
2. Se hace un intento por rastrear cada error y defecto hasta sus primeras causas (por ejemplo, no conformidad con las especificaciones, error de diseño, violación de los estándares, mala comunicación con el cliente, etc.).
3. Con el uso del Principio de Pareto (80 por ciento de los defectos se debe a 20 por ciento de todas las causas posibles), se identifica 20 por ciento de las causas de errores y defectos (las *pocas vitales*).
4. Una vez identificadas las pocas causas vitales, se corrigen los problemas que han dado origen a los errores y defectos.

Seis Sigma

- Definir los requerimientos y metas
- Determinar el desempeño actual de calidad (Medir)
- Analizar métricas de los defectos y sus causas

Mejorar

- Eliminar causas originales
- Controlar el proceso

Desarrollar:

- Diseñar el proceso
- Verificar

Confiabilidad

- “Probabilidad que tiene un programa de computo de operar sin fallas en un ambiente especifico por un tiempo especifico”
- Tiempo medio entre fallas = tiempo medio para la falla + tiempo medio para la reparación
- Fallas en el tiempo:
 - 1 FET = 1 falla cada mil millones de horas de operación

Disponibilidad

- Probabilidad de que un programa opere de acuerdo con los requerimientos en un momento determinado de tiempo
- Disponibilidad =
$$\left(\frac{\text{Tiempo medio para la falla}}{\text{tiempo medio para la falla} + \text{tiempo medio para la recuperación}} \right) * 100\%$$

Plan de ACS

IEEE, estructura:

- Propósito y alcance del plan
- Descripción de todos los productos
- Normas y practicas aplicables
- Acciones y tareas del ACS
- Herramientas y apoyos al ACS
- Procedimientos para la configuración
- Métodos para mantener los registros
- Roles y responsabilidades