



Calidad en el Ciclo de Vida y Actividades de Apoyo

Calidad en las Actividades del Ciclo vida
Calidad en las Actividades de Apoyo

Introducción

- ❑ Sanders y Curran proponen un marco conceptual con prácticas
 - ❑ basadas en la mejor práctica de la ingeniería de software
 - ❑ compatibles con ISO 9000
- ❑ Hay tres tipos de actividades
 - ❑ Actividades del ciclo de vida
 - ❑ Actividades de apoyo
 - ❑ Actividades de nivel organizacional
- ❑ Hay 3 tipos de prácticas
 - ❑ Prácticas esenciales que son obligatorias; se denota E
 - ❑ Prácticas importantes cuya omisión debe justificarse; se denota I
 - ❑ Prácticas útiles que son opcionales; se denota U



Control de documentos

- ❑ Muchos documentos necesitan un control de documentos estricto
 - ❑ Todas las versiones se mantienen
 - ❑ Sólo se entrega una versión después de una aprobación formal (revisión y firma)
 - ❑ Hay control de acceso para escritura concurrente
 - ❑ Hay notificación de cambios
 - ❑ Se denota C
- ❑ Muchos documentos deben ser verificados en forma independiente
 - ❑ Se denota V





Calidad en las actividades del Ciclo de Vida

- 1 Requisitos de usuarios
- 2 Requisitos de software
- 3 Diseño
- 4 Realización
- 5 Implantación
- 6 Evolución

Fases de un ciclo de vida

<i>Fase</i>	<i>Propósito</i>
1 Requisitos de usuarios	Definir el problema
2 Requisitos de software	Analizar el problema
3 Diseño	Plantear una solución
4 Realización	Hacer la solución
5 Implantación	Entregar e iniciar la operación
6 Evolución	Mantener la utilidad del software

- ❑ Descripción de las fases
 - ❑ Propósito
 - ❑ Entradas
 - ❑ Salidas
 - ❑ Actividades



1 Requisitos de los usuarios

PROPOSITO Definir el problema

- Refinar una idea sobre lo que hará el software
- Determinar los alcances del mismo
- Establecer requisitos de software

U El desarrollador debe ayudar a hacer el documento de requisitos de usuarios

E Incluir requisitos de soporte y mantenimiento

E Que el usuario retenga control del documento



1: Entradas y Salidas

ENTRADAS

- No [siempre] hay

- U Usar estudios de factibilidad, encuestas ...

SALIDAS

- EVC Documento de Requisitos de Usuarios

- U Plan detallado para Requisitos de Software

- U Plan de Pruebas de Aceptación

1: Actividades

- ❑ Capturar los requisitos de usuario
 - ❑ E Que sean tan completos como sea posible
 - ❑ I Lograr consenso entre las partes
 - ❑ U Comparar con software existente, con dibujos de pantallas y/o prototipos
- ❑ Determinar ambiente de operación
 - ❑ I Describir software, hardware, otros
 - ❑ IC Describir interfaces externas



1: Actividades (cont.)

- ❑ Especificar los requisitos de usuario
 - ❑ U Clasificar los requisitos
 - ❑ I Usar métodos cuantitativos de especificación
 - ❑ I Por cada atributo dar:
 - Nombre
 - Necesidad: por qué se necesita y cuán importante es
 - Estabilidad: cuán posible es que cambie
 - Prioridad, para entrega incremental
- ❑ Revisar salidas
 - ❑ E Revisar todas las salidas de la fase, con participación de usuarios, operadores, analistas y administradores



2 Requisitos de software

PROPOSITO Representar los requisitos del usuario en términos computacionales

ENTRADAS

- ❑ Documento de requisitos de usuario
- ❑ Plan detallado para Requisitos de Software
 - ❑ E Hacer las actividades de acuerdo al plan



2: Actividades

- ❑ E Construir un modelo lógico del sistema
 - ❑ I Usar el modelo como base para producir los requisitos
 - ❑ I Revisar un nivel antes de pasar al siguiente
 - ❑ I El modelo está hecho de componentes, cada una con su función
 - ❑ I Las interfaces entre componentes son mínimas
 - ❑ U Dividir el sistema en una jerarquía de subcomponentes
 - ❑ U Cada componente no tiene muchas subcomponentes
 - ❑ I El modelo omite descripción de implementación
 - ❑ I El modelo especifica atributos de rendimiento de las componentes



2: Actividades (cont.)

- ❑ E Especificación de requisitos de software
 - ❑ U Clasificar los requisitos
 - ❑ I Usar métodos cuantitativos de especificación
 - ❑ I Por cada atributo dar:
 - Nombre
 - Necesidad: por qué se necesita y cuán importante es
 - Estabilidad: cuán posible es que cambie
 - Prioridad, para entrega incremental
- ❑ E Hacer la Descripción de Servicios de Soporte
- ❑ E Revisar cada una de las salidas



2: Salidas

- ❑ ECV Documento de requisitos de software
 - ❑ U Poner una tabla que muestre la correspondencia entre requisitos de usuario y requisitos de software
 - ❑ I Incluir los resultados de los eventuales análisis
 - ❑ U Si se usaron métodos matemáticos poner explicaciones en castellano
- ❑ Descripción de Servicios de Soporte
- ❑ Plan detallado de la Fase de Diseño
- ❑ Plan de Pruebas de Sistema
- ❑ Otros documentos
- ❑ E Archivar toda la documentación



3 Diseño

PROPOSITO Diseñar la solución de software

- Poner en términos concretos el modelo lógico

ENTRADAS

- Documento de requisitos de software
- Descripción de Servicios de Soporte
- Plan de la Fase de Diseño
 - E Hacer las actividades de acuerdo al plan
 - I Adoptar un método de diseño y seguirlo



3: Actividades

- ❑ E Hacer un Documento de Diseño
 - ❑ Describe el software en términos de la implementación
 - ❑ I Derivar la descripción del modelo lógico
 - ❑ I Usar un método de descomposición
 - ❑ I Revisar cada componente en relación a los requisitos
 - ❑ U Considerar más de un posible diseño
 - ❑ I Documentar sólo un diseño como el 'oficial'
- ❑ E Especificar el diseño
 - ❑ Entradas y salidas, funciones, estructuras de datos usadas



3: Actividades (cont.)

□ E Diseñar el Soporte

- Especificación del soporte
- Entrega del soporte
- I Anticipar distintos niveles de necesidad de servicio y hacer planes de contingencia

E Revisar todas las salidas



3: Salidas

- ❑ Documento de Diseño
 - ❑ Documento de Especificación de Soporte
 - ❑ Plan detallado de la Fase de Realización
 - ❑ Plan de Pruebas de Integración
 - ❑ Otros documentos
- E Archivar toda la documentación



4 Realización

- ❑ PROPOSITO Programar y probar el software.

- ❑ ENTRADAS

- ❑ Documento de Diseño

- ❑ Documento de Especificación de Soporte

- ❑ Plan de Pruebas de Integración

- ❑ Plan detallado de la Fase de Realización

- E Hacer las actividades de acuerdo al plan

4: Actividades

- ❑ Diseño detallado

- ❑ En aquellas partes en donde falte detalle

- ❑ Codificación

- ❑ I Hacerlo de acuerdo a normas de programación

- ❑ Documentación

- ❑ I Concurrente con la codificación y antes de la prueba de unidad

- ❑ Integración

- ❑ E La integración debe hacerse con control de la configuración
 - ❑ I Integrar módulos en forma gradual



4: Actividades (cont.)

□ Pruebas

- E Realizar las pruebas de acuerdo a los planes
- Prueba de módulos
- E Prueba de integración
 - Todas las componentes del diseño deben ser probadas
 - Todas las interfaces deben ser probadas
- E Prueba de sistema
- Prueba de aceptación (en la fase de implantación)



4: Salidas

- ❑ Documentos de diseño detallado
 - ❑ Manual del Usuario
 - ❑ Código fuente y ejecutable
 - ❑ Procedimientos de instalación
 - ❑ Plan detallado de Implantación
 - ❑ Especificación de la Prueba de Aceptación
 - ❑ Otros documentos
- E Archivar toda la documentación



5 Implantación

PROPOSITO Instalar el software y demostrar que satisface los requisitos de usuario.

ENTRADAS

- ❑ Código
- ❑ Documentos de diseño detallado
- ❑ Manual del Usuario
- ❑ Especificación de la Prueba de Aceptación
- ❑ Plan detallado de Implantación

E Hacer las actividades de acuerdo al plan

5: Actividades

- ❑ Instalación
- ❑ Pruebas de aceptación
 - I El usuario esté bien representado
 - E Hacer la aceptación de acuerdo a los planes
 - E Documentar los resultados de las pruebas
- ❑ Aceptación provisional



5: Salidas

- ❑ Declaración de aceptación provisional
- ❑ Software aceptado provisionalmente
- ❑ Documento de transferencia de software
 - ❑ Enumera todos los documentos/componentes entregados con sus versiones y/o fechas



6 Evolución

PROPOSITO Mantener la satisfacción de las necesidades de software

ENTRADAS

- ❑ Software aceptado provisionalmente
- ❑ Documento de transferencia de software
- ❑ Documento de Especificación de Soporte



6: Actividades

- ❑ Aceptación final

- ❑ Se hace luego de un tiempo preestablecido de operación conforme

- ❑ Mantenimiento

- EC Realizar cambios de acuerdo a procedimientos establecidos
 - EC Mantener consistencia entre documentos y programas

- ❑ Servicios de soporte

- E Realizar servicio de acuerdo a lo acordado
 - I Registrar estadísticas de los servicios

- ❑ E Revisión de procedimientos y servicios



6: Salidas

- ❑ Software aceptado / mantenido
- ❑ Reportes de revisiones
- ❑ Documento de Especificación de Soporte actualizado

Esta fase se repite durante la vida del software

Las salidas de esta fase se retroalimentan en la misma fase



Calidad en las Actividades de Apoyo

Administración de proyectos
Gestión de la configuración
Revisión y control de calidad (SQA)
Actividades organizacionales

Actividades de apoyo

- ❑ Las actividades del ciclo de vida no sólo son aquellas relacionadas con desarrollar y mantener software:
 - ❑ Administración de proyectos
 - ❑ Gestión de la configuración
 - ❑ Revisión y control de calidad (SQA)
 - ❑ Actividades organizacionales
- ❑ Estas actividades deben hacerse en todas las fases y etapas
- ❑ E Hay que tener planes y recursos para hacer estas actividades
 - ❑ Parte de esto ya lo vimos: el desarrollo y uso de planes



Responsabilidades de la gerencia

- E Definir, documentar y comunicar la política de calidad
- E Asignar personal a las tareas de la calidad
- E Revisar periódicamente el sistema de calidad
- E Documentar esas revisiones



Administración de proyectos

PROPÓSITO Planificar, organizar, asignar personal, controlar y liderar un proyecto de software

ACTIVIDADES

- ❑ Organización del proyecto
- ❑ Análisis de riesgos
- ❑ Administración técnica
- ❑ Estimación y programación
- ❑ Seguimiento del proyecto



Organización el proyecto

Esta es una función clave de la gestión

E Definir las fases del desarrollo con

- Criterios de inicio
- Entradas y salidas
- Criterios de término

E Definir la estructura del equipo de trabajo

E Documentar los supuestos, dependencias y restricciones de estas decisiones



Análisis de riesgos

“Risk management is management for adults”

Tom DeMarco

E La administración debe

- identificar y evaluar los riesgos del proyecto
- priorizar los riesgos y hacer planes para reducir los más importantes
- re-evaluar riesgos durante el proyecto

I Tener planes de contingencia

Tipos de riesgos usuales

- ❑ Claridad y estabilidad de los requisitos de usuario
 - ❑ Claridad de los criterios de término
 - ❑ Cambios de requisitos
- ❑ Disponibilidad de recursos adecuados
 - ❑ Movilidad del personal
 - ❑ Exceso de carga de trabajo
 - ❑ Atrasos en otros proyectos no liberan recursos
 - ❑ Habilidades y experiencia del personal
- ❑ Riesgos técnicos
 - ❑ Novedad técnica del proyecto
 - ❑ Integración del sistema con sistemas existentes
- ❑ Falta de tiempo
 - ❑ Calendario para el desarrollo
 - ❑ Tiempo del cliente para las aprobaciones



Administración técnica

- ❑ La administración del proyecto debe elegir los métodos y herramientas, organizar el control de la configuración y actividades de verificación

E Estas elecciones deben ser consistentes con las necesidades del proyecto y las políticas organizacionales



Estimación y programación

E Definir la estructura de tareas y subtareas

- ❑ cubriendo todas las actividades de cada fase
- ❑ especificando fechas de inicio y término de las tareas
- ❑ con tareas detalladas y controlables
- ❑ con posibilidad de hacer trabajo independiente
- ❑ relacionando las fechas con las fechas del proyecto
- ❑ usando la descomposición modular del software
- ❑ En inglés, esto se llama Work Breakdown Structure.

IC Esta información se debe registrar en un programa de proyecto tipo carta Gantt

Seguimiento del proyecto

E Verificar el progreso del proyecto en forma continua y compararlo con los planes

I Refinar los planes durante el proyecto

I Hacer reportes regulares de avance

- Dan visibilidad al avance
- Sirven para informar del avance a otras personas

IC Hay que mantener actualizado el programa de proyecto en
caso de atrasos o cambios



Plan de proyecto

❑ Introducción

- ❑ Resumen
- ❑ Entregables
- ❑ Material de referencia
- ❑ Glosario

❑ Organización del proyecto

- ❑ Modelo de ciclo de vida
- ❑ Estructura organizacional
- ❑ Fronteras e interfaces organizacionales
- ❑ Responsabilidades

❑ Proceso de gestión

- ❑ Objetivos y prioridades de gestión
- ❑ Supuestos, dependencias y restricciones
- ❑ Administración de riesgos
- ❑ Mecanismos de seguimiento y control
- ❑ Asignación de personal

❑ Proceso técnico

- ❑ Métodos, técnicas y herramientas
- ❑ Documentación de software
- ❑ Funciones de apoyo

❑ Programación

- ❑ Paquetes de tareas (*work packages*)
- ❑ Requerimientos de recursos
- ❑ Presupuesto y asignación de recursos
- ❑ Calendario



Gestión de la configuración

PROPÓSITO Asegurar que el software es consistente, en sus distintas versiones

- ❑ Esta es una función técnica y de gestión
- ❑ Gestión de la configuración incluye
 - ❑ Identificar y definir ítemes de configuración
 - ❑ controlar la entrega y cambio de ítemes de configuración
 - ❑ registrar y reportar el estado de ítemes de configuración
 - ❑ verificar la completitud y corrección de ítemes de configuración
- E Tener un procedimiento para identificar, almacenar y modificar ítemes de configuración



Actividades de gestión de la configuración

- ❑ Identificación de ítemes
- ❑ Almacenamiento de ítemes
- ❑ Control de cambios
- ❑ Reporte de problemas y acciones correctivas
- ❑ Procedimiento de cambios en documentación
- ❑ Control del estado de la configuración
- ❑ Entrega



Aseguramiento de calidad de software (SQA)

PROPÓSITO Comprobar y registrar si los procedimientos y estándares producen calidad adecuada y si se han seguido en el proyecto

- En inglés se llama 'software quality assurance' (SQA)

E ACTIVIDADES

- Proveer una estructura adecuada para ACS
- Tener un plan de documentación
- Verificar el uso de estándares y prácticas
- Examinar el mecanismo de revisiones y auditorías



Actividades de SQA (cont.)

- ❑ Revisar la documentación de las pruebas
- ❑ Revisar procedimientos de errores y cambios
- ❑ Revisar procedimientos de control de configuración
- ❑ Exigir a los subcontratistas un plan de SQA verificable y compatible con el nuestro
- ❑ Asegurar que se guardan todos los documentos (ej., minutas de reuniones)
- ❑ Asegurar que el personal está entrenado
- ❑ Revisar que la administración de riesgos existe



Verificación & convalidación

PROPÓSITO Comprobar y registrar si los ítemes de configuración están o no de acuerdo a lo especificado

PROPÓSITO Comprobar y registrar si el software satisface o no las necesidades

Esto se hace a lo largo del ciclo de vida

ACTIVIDADES

- ❑ Revisiones, inspecciones y caminatas (walkthroughs)
- ❑ Rastreo (trace) de requisitos y otros
- ❑ Pruebas
- ❑ Auditorías
- ❑ Demostraciones formales



Actividades a nivel organizacional

- ❑ Administración del Proceso de Software
 - ❑ Definición del proceso estándar
 - ❑ Medición y registro del proceso
 - ❑ Prevención de defectos
 - ❑ Innovación tecnológica
 - ❑ Mejora del proceso estándar
- ❑ Subcontratación
- ❑ Entrenamiento
- ❑ Responsabilidades de la gerencia



Definición del proceso estándar

- E La organización debe tener un proceso de software estándar documentado
 - I El personal debe participar en definirlo
 - I La definición es un proyecto con un plan, recursos y aprobación de la gerencia
 - I El proceso debe ser adaptado en cada proyecto de acuerdo a una pauta



Medición y registro del proceso

E Registrar información del resultado de cada proyecto

- ❑ lecciones aprendidas
- ❑ costo, tiempo y tamaño del proyecto

E Medir el proceso estándar en cada proyecto

- ❑ Analizar esas medidas
- ❑ Usar las medidas para mejorar el proceso

I Las medidas deben

- ❑ ser consistentes entre los proyectos
- ❑ derivarse de los objetivos
- ❑ estar relacionadas con los requisitos de usuarios
- ❑ incluir todo el ciclo de vida

Prevención de defectos

PROPÓSITO Detectar y eliminar fuentes de defectos y defectos recurrentes

E Prevenir defectos a nivel organizacional

E Prevenir defectos en cada proyecto

U Tener reuniones para identificar defectos

U Clasificar y registrar todos los defectos

- Descripción y tipo de la causa
- Origen y detección (etapas del ciclo de vida)
- Gravedad y urgencia de corrección
- Acción y/u omisión



Innovación tecnológica

E Monitorear nueva tecnología e implementar aquella que pudiera ser útil

- ❑ Para hacerlo hay que destinar recursos, tener planes, etc.
- ❑ Es decir esto no se hace solo, aún cuando tengamos ingenieros capaces e interesados
- ❑ Organizaciones pequeñas pueden externalizar en parte esta función



Mejora del proceso estándar

EC Mejorar el proceso de software e informar de los cambios

E Definir responsables de hacerlo

I Debe haber procedimientos de sugerencias

- Todas las sugerencias se registran y se decide sobre ellas
- El resultado se comunica a quien propone
- Las 'buenas' ideas se hacen parte del proceso
- Las 'malas' ideas se tratan con respeto

U Probar cambios en proyectos pilotos



Subcontratación

- E Desarrollar y mantener procedimientos en relación a la compra y subcontratación de software
- E Elegir proveedores que demuestran calidad
- E Hacer un contrato que incluya
 - objetivos técnicos
 - alcance del trabajo
 - estándares y procedimientos a usar
 - términos y condiciones
 - costos, plazos y otras restricciones
- E Verificar todo lo recibido (también del cliente)



Entrenamiento

- E Desarrollar y mantener programas de entrenamiento para asegurar que el personal tiene las habilidades técnicas y gerenciales adecuadas
- E Cada proyecto debe
 - definir sus necesidades en cuanto a habilidades
 - hacer entrenamiento de ser necesario
- E Mantener registros de cursos y alumnos



Referencia

- ❑ Joc Sanders y Eugene Curran. Software Quality: A framework for success in software development and support, Addison-Wesley, 1994.

