

8 - Proceso de Software

Conceptos importantes

Ownership of code: Cuanto menos claro está definido el dueño de una porción de código, más probable es que esa porción de código tenga errores.

Intercontinental partners: Cuanto más inclusivo es el proceso, mejor es el producto final, porque se cuenta con una mirada más amplia.

Procesos y modelos de proceso

El proceso es distinto del producto: el producto es el resultado de ejecutar un proceso. La IS se enfoca en el proceso, siguiendo la premisa de que un **proceso adecuado ayuda a lograr los objetivos del proyecto con alta C&P**.

Un proyecto exitoso es el que satisface las expectativas en costo, tiempo, y calidad. Al planear y ejecutar un proyecto de sw, las decisiones se toman con el fin de reducir costos y tiempos e incrementar la calidad.

Un **modelo de proceso** especifica un **proceso general**, usualmente con fases en las que el proceso debe dividirse, conjuntamente con otras restricciones y condiciones para la ejecución de dichas fases. Es importante notar que un modelo de proceso no se traduce directamente al proceso: en general el proceso real es una adaptación del modelo del proceso de acuerdo a las características del proyecto (**especificación del proceso**). Así, es el **proceso** el que guía un proyecto, e influye significativamente en el resultado esperado del proyecto.

Componentes del proceso de software

Dos procesos fundamentales:

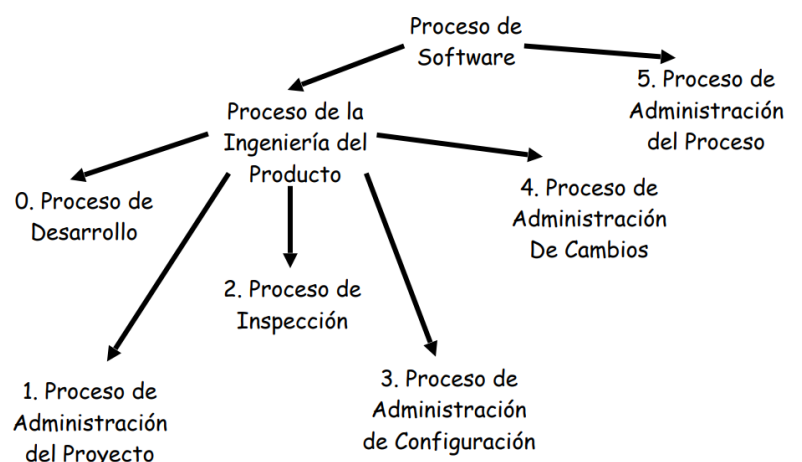
- **Desarrollo:** se enfoca en las actividades para el desarrollo y para garantizar la calidad necesaria para la ingeniería del sw.
- **Administración del proyecto:** se enfoca en el planeamiento y control del proceso de desarrollo con el fin de cumplir los objetivos.

El proceso de desarrollo es el “corazón” del proceso de software; los otros procesos giran alrededor de él. Cada proceso se ejecuta por gente diferente:

- Los **desarrolladores** ejecutan el proceso de desarrollo.
- Los **administradores** del proyecto ejecutan el proceso de administración.

Además, podemos distinguir:

- Proceso de administración del proyecto
- Proceso de inspección
- Proceso de administración de configuración
- Proceso de administración de cambios



Especificación del proyecto

El proceso generalmente es un conjunto de fases, cada fase realiza una tarea bien definida y produce una salida. Tal salida intermedia se llama producto de trabajo. Cada producto de trabajo es una entidad formal y tangible, capaz de ser verificada. Cada fase puede ser llevada a cabo usando distintas metodologías.

ETVX

En esta etapa suele utilizarse un enfoque llamado ETVX (Entry-Task-Verification-Exit), que especifica para cada etapa:

- **Criterio de entrada:** qué condiciones deben cumplirse para iniciar la fase. **El criterio de entrada de una fase debe ser consistente con el criterio de salida de la anterior.**
- **Tarea:** Lo que debe realizar esa fase.
- **Verificación:** Las inspecciones/controles/revisiones/verificaciones que deben realizarse a la salida de la fase (i.e. al producto de trabajo).
- **Criterio de salida:** Cuando se considera que la fase fue realizada exitosamente.

Además, cada fase produce información para la administración del proceso.

Características deseadas del proceso de software

Siempre buscamos **calidad** y **productividad** alta. Para esto, el proceso:

- Debe producir **sw testeable**: testing es la tarea más cara dentro del proceso de desarrollo; entre 30 y 50% del esfuerzo total de desarrollo.
- Debe producir **sw mantenible**: el mantenimiento puede ser más caro que el desarrollo; hasta 80% del costo total durante la vida del sw.
- Debe **eliminar defectos en etapas tempranas**.
 - El costo de eliminar un defecto se incrementa a medida que perdura en el proceso de desarrollo. El crecimiento es exponencial, no lineal.
 - Los errores deberían ser encontrados en la etapa en que se introdujeron (La 'V' en ETVX).
 - El control de calidad es la actividad cuyo propósito principal es identificar y eliminar errores. Pero es limitada, por lo tanto el proceso debe lograr que se introduzcan la menor cantidad posible de defectos.
- Debe ser **predecible y repetible**.
 - Los procesos deben conseguir repetir el buen desempeño cuando se utilizan en distintos proyectos. Es decir, el resultado de utilizar un proceso debe poder predecirse.
 - Sirve para estimar costos, esfuerzo y calidad.
 - Un proceso predecible se dice que está bajo un **control estadístico**. (Define un rango entre el tiempo estimado y el precio).
- Debe **soportar cambios y producir sw que se adapte a cambios**
 - Los **cambios de requerimientos** son esperables y no pueden tratarse como algo "malo".
 - Todo **proceso** de desarrollo de software debe dejar lugar para cambios y tratarlos apropiadamente.

Fases

Dijimos que el proceso de desarrollo de software es un conjunto de fases. Cada fase es a su vez una secuencia de pasos que definen la metodología de la fase.

¿Por qué utilizar fases?

- Dividir y conquistar.
- Cada fase ataca distintas partes del problema.
- Ayuda a validar continuamente el proyecto.

Usualmente está compuesto por las siguientes actividades:

1. Análisis de requerimientos y especificación:
2. Arquitectura y Diseño
3. Codificación:
4. Testing
5. Entrega e instalación

Los distintos modelos de proceso realizan estas actividades de distinta manera.

Otros procesos del software

El proceso de desarrollo es el proceso principal sobre el cual rondan los otros procesos, influye sobre los métodos para los otros procesos. Los otros procesos son:

1. Proceso para la **administración del proyecto**.
2. Proceso de **inspección**.
3. Proceso para la **administración de configuración**.
4. Proceso para la **administración de cambios**.
5. Proceso para la **administración del proceso**.

1 Proceso para la administración del proyecto

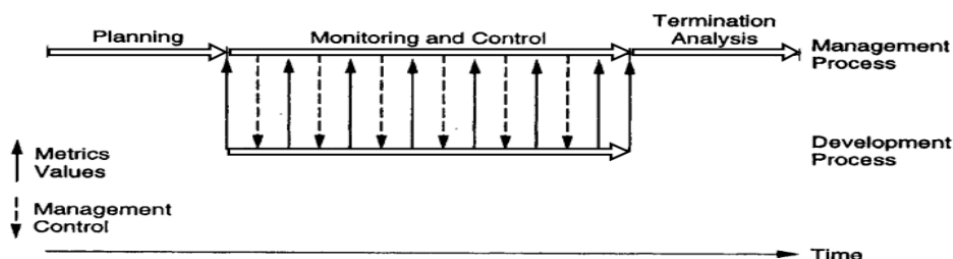
El proceso de desarrollo divide el desarrollo en fases y actividades. Para ejecutarlas eficientemente, se deben **asignar recursos, administrarlos, observar el progreso, tomar acciones correctivas, etc.** Todo esto es parte del proceso para la administración del proyecto, que se vuelve parte esencial de la ejecución del proyecto.

Fases:

1. **Planeamiento:** Es la actividad principal que produce un plan el cual forma la base del seguimiento. Se realiza antes de comenzar el proyecto. Tareas claves:
 - Estimación de costos y tiempos.
 - Seleccionar el personal.
 - Planear el seguimiento.
 - Planear el control de calidad.
2. **Seguimiento y control:** Acompaña al proceso de desarrollo. Tareas:
 - Seguir y observar parámetros claves como costo, tiempos, riesgo, así como los factores que los afectan.
 - Tomar acción correctiva si es necesario.

Las métricas proveen la información del proceso de desarrollo necesaria para el seguimiento.

3. **Análisis de terminación:** Se realiza al finalizar el proceso de desarrollo. El propósito fundamental es analizar el desempeño del proceso e identificar las lecciones aprendidas. En procesos iterativos el análisis de terminación se realiza al finalizar cada iteración y se usa para mejorar en iteraciones siguientes.



Este gráfico muestra la relación temporal entre el proceso de administración y el proceso de desarrollo. Vemos que la planificación se realiza antes de que empiece el proceso de desarrollo, y el análisis de terminación se realiza una vez que finalizó el desarrollo. Durante el desarrollo, desde las distintas fases del proceso de desarrollo, se envía información cuantitativa a la fase de monitoreo y control del proceso de administración, que utiliza esta información para controlar el proceso de desarrollo.

2 Proceso de inspección

Objetivo principal: detectar los defectos en los productos de trabajo.

Inicialmente utilizado para el código, actualmente usado en todos los tipos de productos de trabajo. Está reconocido como una de las mejores prácticas de la industria. Mejora tanto la calidad como la productividad. Los defectos pueden introducirse en el SW en cualquier etapa, y deben eliminarse en cada etapa. Las inspecciones pueden realizarse sobre cualquier documento, incluidos requerimientos, diseños y planificaciones.

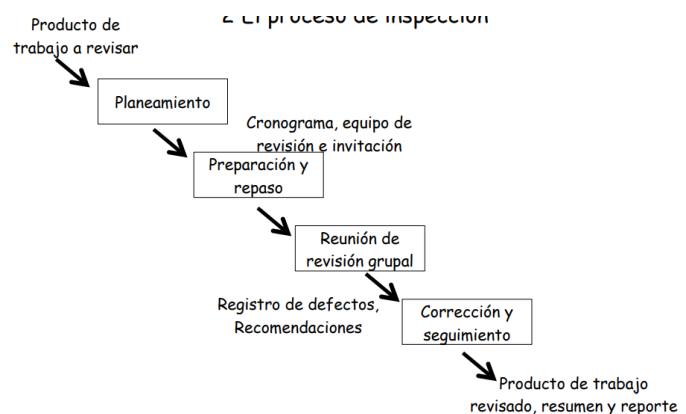
Es realizado por personal técnico para personal técnico, es una **revisión/crítica (review) hecha por pares**. Es un proceso estructurado con roles definidos para cada participante (moderador, autor, revisor, lector, y escriba). **El foco está en encontrar problemas, no en resolverlos.** La información recolectada en la revisión es registrada y utilizada para monitorear la efectividad de la solución.

Roles y responsabilidades:

- **Moderador:** tiene la responsabilidad general.
- **Autor:** quien realizó el producto de trabajo.
- **Revisor:** quien identifica los defectos.
- **Lector:** lee línea a línea el producto de trabajo para enfocar el progreso de la reunión.
- **Escriba:** registra las observaciones indicadas.

Planeamiento:

- Seleccionar el equipo de revisión.
- Identificar al moderador (tiene la responsabilidad principal en la inspección).
- Preparar el paquete para la distribución, compuesto por:
 - el producto de trabajo a revisar;
 - las especificaciones del producto de trabajo;
 - lista de control con ítems relevantes;
 - estándares.



Preparación y repaso previo (overview):

- Breve reunión (opcional) (el moderador decide si se hace):
 - se entrega el paquete;
 - se explica el propósito de la revisión;
 - se da una breve introducción señalando áreas de cuidado.
- Todos los miembros del equipo revisan individualmente el producto de trabajo:
 - identificar defectos potenciales en registro individual; se usan listas de control (checklists), pautas, y estándares.
 - Idealmente no debería durar más de 2 horas y debería hacerse de corrido.
- Se utiliza una planilla de registro de la revisión individual:

Project name and code :

Work product name and ID:

Reviewer name:

Effort spent for preparation (hrs):

Defect List:

Sl	Location	Description	Criticality / Seriousness

Reunión de revisión grupal:

- Propósito: **definir la lista final de defectos.**
- Criterio de entrada: cada miembro debe haber hecho apropiadamente la revisión individual (el moderador revisa los registros individuales).
- La reunión:
 - El lector lee línea a línea el producto de trabajo (o cualquier otra pequeña unidad).
 - En cualquier línea, cualquier observación que hubiere (preparada o nueva) es efectuada.
 - Se sigue una discusión para identificar el defecto.
 - La decisión es registrada por el escriba.
- El moderador está a cargo de la reunión y juega un rol central:
 - Asegura que el foco permanece sobre la identificación de defectos y debe evitar que se prolonguen o se discutan soluciones.
 - Lo que se está revisando es el producto de trabajo y no el autor de éste.
 - Debe garantizar que la reunión se ejecute ordenada y amigablemente.
 - Utiliza el resumen para analizar la efectividad de la revisión.
- A final de la reunión:
 - El escriba presenta la lista de defectos/observaciones.
 - Si hay pocos defectos el producto de trabajo se acepta; si no, se puede requerir otra revisión.
 - El grupo no propone soluciones, aunque podrían registrarse sugerencias.
 - Se prepara un resumen de las inspecciones - se usa para evaluar la efectividad de la revisión.

Corrección y seguimiento:

- Los defectos en la lista de defectos son posteriormente corregidos por el autor.
- Una vez corregidos, el autor obtiene el visto bueno del moderador o el producto de trabajo se somete a una nueva revisión.
- Una vez que los defectos/observaciones fueron satisfactoriamente procesados, la revisión finaliza.
- Pautas para la revisión de los productos de trabajo:

Producto de trabajo	Enfoque de la inspección	Participantes
Especificación de requerimientos	Cumple con las necesidades del cliente. Es implementable. Omisiones, inconsistencias, ambigüedades.	Cliente, Analista, Diseñador, Desarrollador
Diseño de alto nivel	El diseño implementa los requerimientos. El diseño es implementable. Omisiones, calidad del diseño.	Autor de SRS, Diseñador a nivel detalle, Desarrollador.
Código	El código implementa el diseño. El código es correcto y completo. Defectos en el código. Otras características de calidad	Diseñador, Tester, Desarrollador.
Casos de Tests	Conjunto de casos de tests verifica los requerimientos en la SRS. Los casos de tests son ejecutables.	Autor de SRS, Tester, Líder del proyecto.
Plan de administración de proyecto	El plan es completo y especifica todos los componentes. Es implementable. Omisiones y ambigüedades.	Líder del proyecto, otros administradores de proyecto.

3 Proceso de administración de configuración

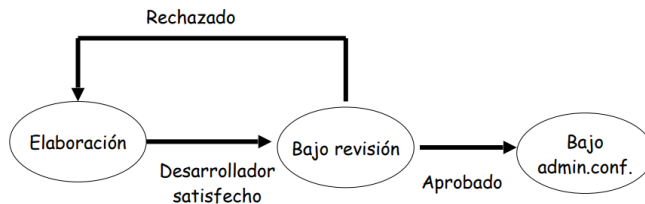
Un proyecto de software produce muchos ítems: programas, documentos, datos, manuales, etc. Cualquiera de ellos puede cambiar fácilmente, por lo que es necesario saber el progreso del estado de cada ítem. La administración de configuración del software (SCM): controla sistemáticamente los cambios producidos (Se enfoca en los cambios durante la evolución; los cambios de requerimientos se manejan aparte).

Funcionalidades necesarias:

- Recolectar todos las fuentes, documentos y otra información del sistema actual.
- Evitar cambios o eliminaciones desautorizadas.
- Deshacer cambios o revertir a una versión especificada.
- Hacer disponible la última versión del programa.

Ítems de configuración: Cada ítem es una unidad de modificación y se modifican intensamente, los cambios a estos ítems se siguen rigurosamente. Las distintas versiones de ítems deben combinarse apropiadamente de manera periódica: baseline es un arreglo apropiado de ítems de configuración.

La **baseline** establece puntos de referencia a lo largo del desarrollo del sistema. Captura el estado lógico del sistema y forma la base de cambios posteriores.



Control de acceso:

4 Proceso de administración de cambios de requisitos

Los requerimientos pueden cambiar en cualquier momento durante el desarrollo. Los cambios producen impactos en los productos de trabajo y en los distintos ítems de configuración. Los cambios deben permitirse, pero siempre de manera controlada, pues no controlados pueden impactar adversamente en el proyecto, tanto en costo como en tiempo.

El proceso:

- Registrar los cambios.
- Realizar análisis de impacto sobre los productos de trabajo y los ítems.
- Estimar el impacto en esfuerzo y en cronograma.
- Analizar el impacto con las personas involucradas.
- Reprocesar los productos de trabajo y los ítems.

Los cambios se inician a través de un **requerimiento de cambio**. Existe un **registro de requerimientos de cambio**. El análisis de impacto para un requerimiento de cambio incluye identificar los cambios necesarios en los distintos ítems y la naturaleza del cambio. El impacto del cambio en el proyecto es analizado para decidir si hacerlo efectivo o no. Los **cambios acumulativos** también se registran.

5 Proceso de administración de procesos

Un proceso no es una entidad estática: éste debe cambiar para mejorar la calidad y productividad. La administración de procesos se enfoca en la **evaluación y mejora del proceso**. Se enfoca en el **proceso**.

Para mejorar el proceso, una organización debe comprender el proceso actual: requiere que el proceso esté bien documentado, que sea apropiadamente ejecutado en los proyectos, y recolecta datos de los proyectos para comprender el desempeño del proceso en los proyectos.

Es mejor que los cambios realizados al proceso se hagan incrementalmente, de a pasos pequeños, y deben ser cuidadosamente seleccionados: decidir qué cambios hacer y cuándo. Existen marcos que sugieren formas de proceder en la mejora del proceso. Ej.: CMM - Capability Maturity Model.

CMM:

Tiene 5 niveles para el proceso de software (el 1 es ad-hoc). En cada nivel, el proceso tiene ciertas capacidades y establece las bases para pasar al siguiente nivel. Para moverse de un nivel a otro, CMM especifica áreas en las cuales enfocarse. Se utiliza ampliamente en la industria del software.