

8. Procesos de Software

Introducción

Procesos y modelos de proceso

Especificación de un proceso

Enfoque ETVX ★

Características deseadas en el proceso de software

Componentes del proceso de software

0. Proceso de desarrollo de software ★

1. Proceso para la administración del proyecto ★

2. Proceso de inspección ★

3. Proceso de administración de configuración ★

4. Proceso de administración de cambios de requisitos ★

5. Proceso de administración de procesos

CMM - Capability Maturity Model ❌

Introducción

El concepto de proceso es fundamental en la ing. de software. En este contexto, con proceso nos referimos al método específico para desarrollar software, que involucra una serie de pasos u operaciones.

Un proceso de software es un conjunto de actividades con restricciones de orden entre ellas, tal que si se realizan correctamente y en conformidad con las restricciones, se logra tener un software de calidad alta a bajo costo.

En esta parte de la materia se busca comprender los procesos de software, sus componentes y algunos modelos propuestos para su aplicación.



Importante saber que proceso \neq producto.

Esta unidad capaz se entiende mejor si se piensa como anterior a la unidad 7, de modelos en el proceso de desarrollo, ya que el proceso de desarrollo es un componente del proceso de software.

Procesos y modelos de proceso

Los procesos guían un proyecto e influyen significativamente el resultado esperado del mismo. Éstos terminan siendo adaptaciones de lo que llamamos **modelo de procesos**:

Un modelo de proceso especifica un proceso general, usualmente en fases, en las que el proceso debe dividirse conjuntamente con otras restricciones y condiciones para la ejecución de dichas fases. Puede ser o no orientado al software.



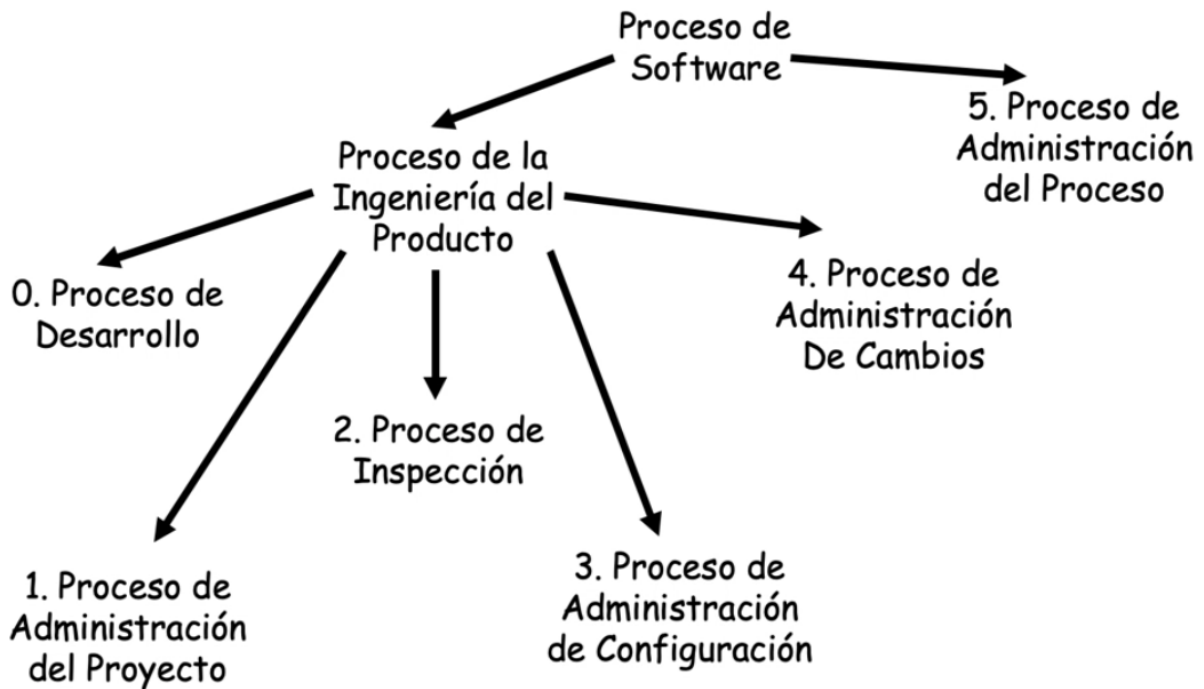
Un proceso adecuado ayuda a lograr los objetivos del proyecto con alta calidad y productividad.

Cuando un proceso es orientado al software, lo llamamos proceso de software. Éste está compuesto por procesos importantes, **dos** de ellos son:

1. Desarrollo: se enfoca en las actividades para el desarrollo y para garantizar la calidad necesarias para la ingeniería del SW. Éste es el corazón del proceso de software; los otros procesos giran alrededor del desarrollo.
 - Ejecutado por desarrolladores
2. Administración del proyecto: se enfoca en el planeamiento y control del proceso de desarrollo con el fin de cumplir objetivos.
 - Ejecutado por administradores

Los demás procesos que componen al proceso de software, giran entorno al proceso de desarrollo. Los veremos más adelante.

Paneo de todos los procesos que componen al proceso de software:



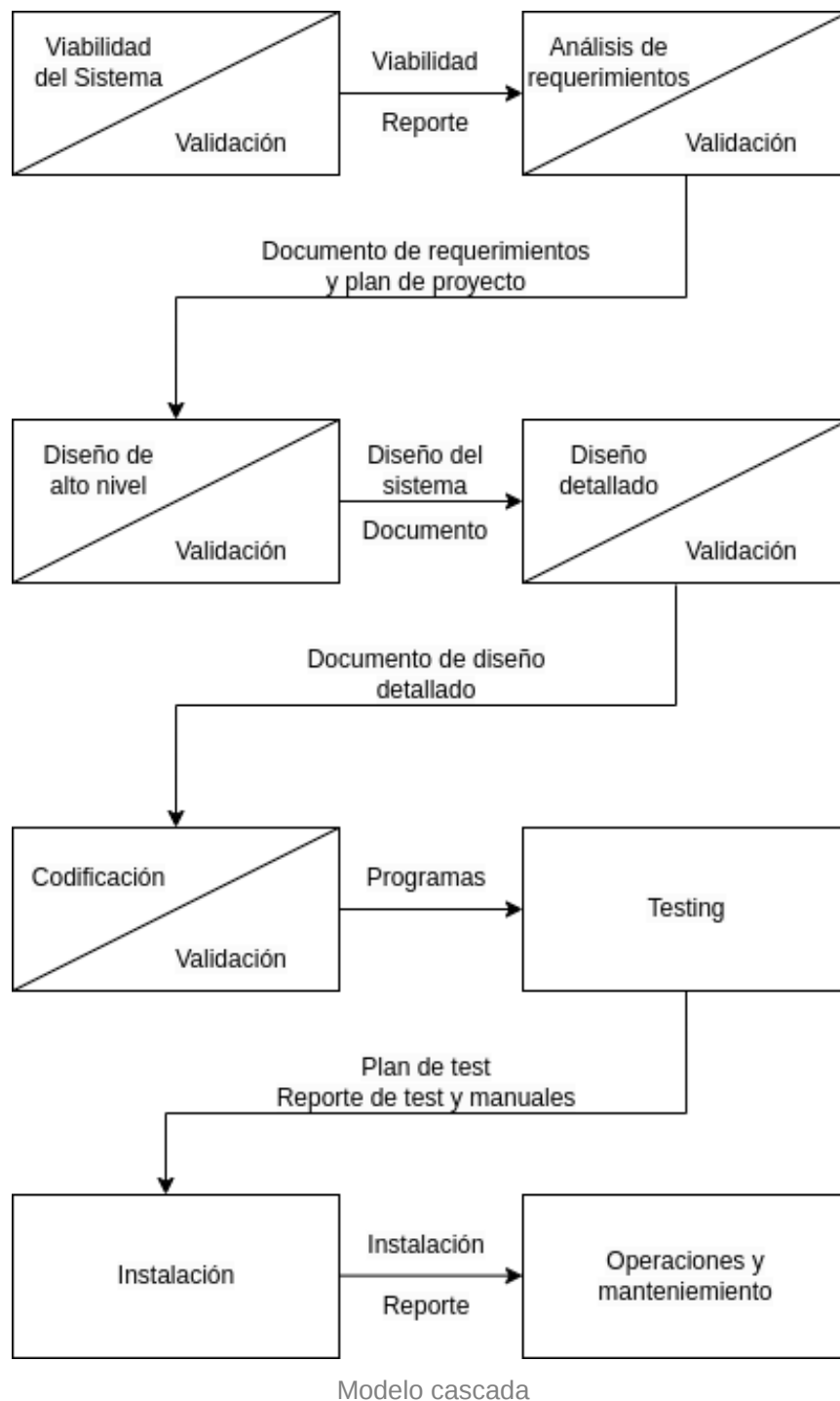
Vamos a ver esto detalladamente en la sección de **componentes de un proceso de software** (más abajo).

Especificación de un proceso

Veremos acá como es que se especifica un proceso de software en general. (es lo que vimos más o menos en la unidad 7)

Tenemos que un proceso de software es un **conjunto de fases**, donde cada fase realiza una **tarea** y produce una salida llamada **producto de trabajo** (entidad formal y tangible, capaz de ser verificada).

Un ejemplo de fases y salidas son los objetos que salen de la primera columna de cuadrados y van hacia la segunda columna de cuadrados, o viceversa en el modelo de cascada. (Reporte de viabilidad, plan de proyecto y documento de requerimientos, etc...)



Las fases que componen a un proceso pueden llevarse a cabo usando distintas metodologías (cascada, etc). De todas maneras, el enfoque que engloba estas metodologías es específicamente el **enfoque ETVX**.

Enfoque ETVX ★

Enfoque que siguen las fases de un proceso de software. Comprende:

- Criterio de entrada: qué condiciones deben cumplirse para iniciar la fase. Debe ser consistente al criterio de salida de la fase anterior.
- Tarea: lo que debe realizar la fase.
- Validación: inspecciones/controles/revisiones/verificaciones que deben realizarse a la salida de la fase.
- Criterio de salida: cuando puede considerarse que la fase fue realizada exitosamente.

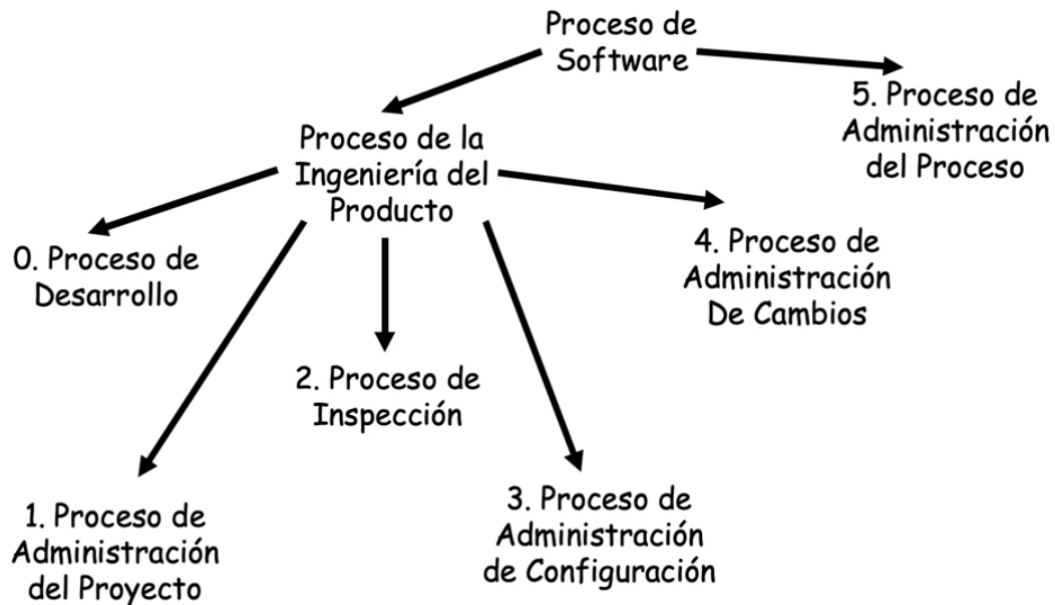
Características deseadas en el proceso de software

Lo que se busca siempre es proveer **alta calidad y productividad**. Se mencionan acá otras características que hablamos un montón ya en unidades pasadas:

- El SW sea mantenible y testeable.
- Se deben eliminar defectos en etapas tempranas. (por esto es importante la V en “ETVX”).
- El SW debe ser predecible y repetible.
- Debe soportar cambios y producir SW adaptable. Ésto se debe a que el SW puede cambiar de requerimientos.
- Hacer controles de calidad: para identificar y eliminar errores. Tener en cuenta que este control es limitado, no confiarse, prevenir errores.
- El resultado de utilizar un proceso debe poder predecirse, es decir, el desempeño de procesos similares debe repetirse. Esto estima costos y esfuerzos.
 - Se usa un control estadístico para saber si un proceso es predecible.

Componentes del proceso de software

Veamos detalladamente los componentes que mencionamos al inicio de la unidad.



0. Proceso de desarrollo de software ★

Como vimos, este es el corazón del proceso de software.

El objetivo de este proceso es construir sistemas de SW dentro de los costos y el tiempo planeado, y que posea la calidad apropiada satisfaciendo al cliente.

Para cada proyecto, el proceso de desarrollo a seguir se especifica durante el planeamiento.

Éste se compone de las siguientes actividades (que ya vimos un montón de veces):

1. Análisis de requerimientos y especificación: Comprender precisamente el problema y especificar SRS
2. Arqui y diseño: Se obtienen documentos correspondientes con los tipos de diseños detallados
3. Codificación: Implementar el diseño con código simple y fácil de comprender.
4. Testing: Se busca identificar la mayoría de los defectos.
5. Entrega e instalación.

Ahora, si bien el proceso de desarrollo es el principal, tenemos que los demás que rondan sobre éste son los que siguen a continuación

1. Proceso para la administración del proyecto ★

Para ejecutar las fases de un proceso de forma eficiente, debemos asignar, administrar recursos y tomar acciones correctivas.

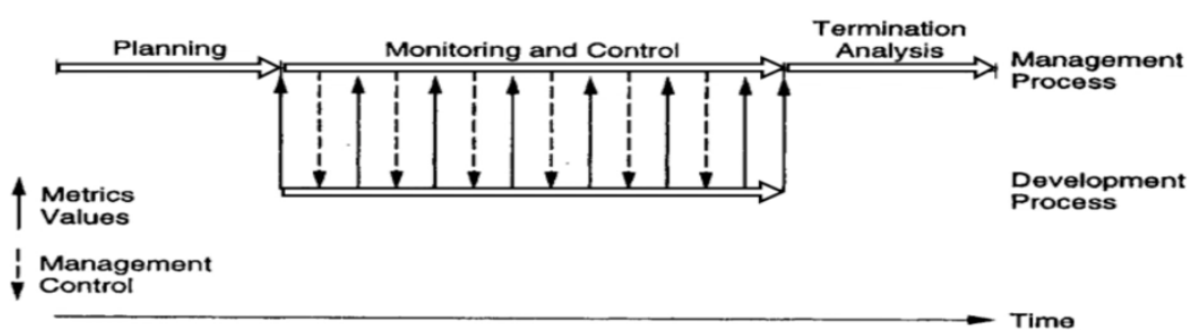
En esta etapa, tenemos 3 fases importantes:

- **Planeamiento:** antes de que el proceso comience, se deben asignar los recursos. Las tareas claves son:
 - Estimar costos y tiempos
 - Seleccionar personal
 - Planear seguimiento
 - Planear control de calidad
- **Seguimiento y control:** acompaña al proceso de desarrollo. Sus tareas son:
 - Seguir y observar parámetros como costo, tiempos, riesgo, etc.
 - Tomar acción correctiva si es necesario.

En esta fase se usan las métricas mencionadas en otras unidades.

- **Análisis de terminación:** se realiza al terminar el proceso de desarrollo y se centra en analizar el desempeño del proceso de desarrollo y llegar a conclusiones como:
 - Qué errores se cometieron
 - Qué lecciones se aprendieron

Esquema total



https://prod-files-secure.s3.us-west-2.amazonaws.com/206ee924-3842-491f-9926-b87d27a727f3/159593f0-09cc-4a11-8a2e-05aaefd197b1/WhatsApp_Ptt_2023-10-29_at_00.47.10.ogg

Explicación del esquema por la profe

2. Proceso de inspección ★

Tiene como objetivo principal detectar los defectos en los productos de trabajo. Se usa en todos los tipos de productos de trabajo y está reconocido como una de las mejores prácticas de la industria. **Mejora** tanto la **calidad** como la **productividad**.

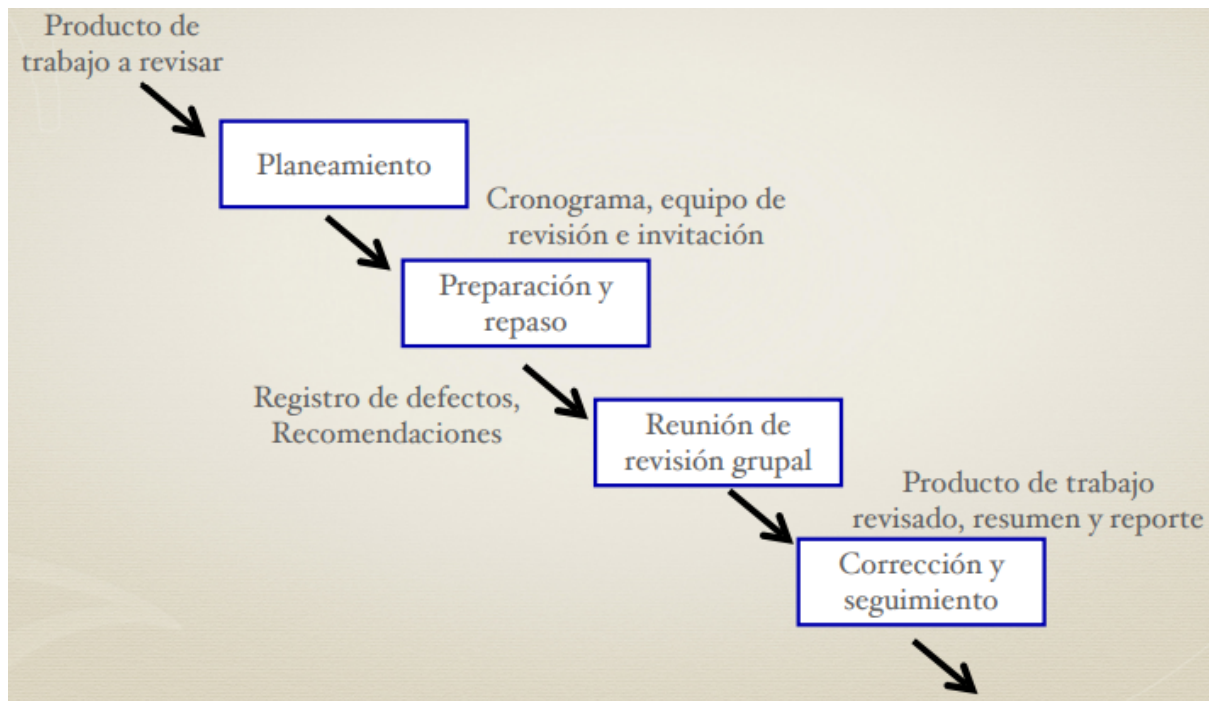
Se realizan inspecciones sobre cualquier documento, incluidos requerimientos, diseños y planificaciones.

En este proceso, se tienen roles que se respetan a lo largo de las **fases de revisión** que veremos. ★ Los roles para cada participante en la revisión son:

- **moderador**: tiene la responsabilidad general.
- **autor**: quién realizó el producto que se esta revisando.
- **revisor**: son los que buscan los errores.
- **lector**: lee línea a línea el producto a revisar para enfocar el progreso de la reunión.
- **escriba**: anota todo y registra observaciones indicadas.

Recordemos que el foco está en encontrar problemas y registrarlos para su monitoreo cuando se solucionen luego! En este proceso no las solucionamos!

Las fases de revisión son:



- **Planeamiento**:

- Se identifica a un moderador y el mismo planea cómo será revisado el producto en cuestión (se detalla cronograma, se asigna qué equipo va a revisarlo, etc).
- Se prepara un paquete para la distribución que tiene:
 - Producto a revisar y sus especificaciones.
 - Lista de control con items relevantes y estándares a seguir.
- **Preparación y repaso:** (se hace a veces)
 - Pequeña reunión entre moderador y revisores donde se entrega el paquete de Planeamiento y se explica lo que se tiene que revisar y para qué.
 - Se señalan áreas de cuidado en donde hubo típicos problemas.
 - Aquí todos los miembros del equipo de revisión deben revisar individualmente el producto. Se usan planillas de registro individual.

Project name and code :			
Work product name and ID:			
Reviewer name:			
Effort spent for preparation (hrs):			
Defect List:			
Sl	Location	Description	Criticality / Seriousness

Ejemplo de planilla de registro individual.

- **Reunión de revisión grupal:** (la importante)
 - Acá se define la lista final de defectos usando las revisiones individuales.
 - El lector lee línea a línea y se van haciendo observaciones.
 - El escriba registra decisiones y al final presenta la lista de defectos y observaciones surgidas (si hay poco defectos, se acepta el trabajo, si no, se rerevisa).

Project	XXXXXXXX
Work Product Type	Project Plan, V 1.0
Size of Product	14 pages
Review Team	P1, P2, P3, P4
Effort (Person Hours)	
Preparation	Total 10 person-hrs.
Group Review Meeting	10 person-hrs.
Total Effort	20 person-hrs.
Defects	
Number of Critical Defects	0
Number of Major Defects	3
Number of Minor Defects	16
Total Number of defects	19
Review Status	Accepted
Recommendations for next phase	
Comments	The plan has been well documented and presented

Ejemplo de un resumen que concluye de la reunión grupal

- El moderador debe estar a cargo de la reunión y asegurarse que el foco no se desvíe. Debe garantizar que la reunión sea ordenada y amigable. Además, usa el resumen para analizar la efectividad de la revisión.
- **Corrección y seguimiento:** (la que más se suelen olvidar en parciales)
 - Una vez se tiene la lista de defectos de la etapa anterior, el autor corrige los mismos. Una vez corregidos, el moderador debe dar un okay sobre los mismos o, si no, realizar otra revisión.
 - Una vez los defectos y observaciones fueron corregidos, la revisión se termina.

Ahora, el proceso de inspección se puede aplicar sobre varios productos. Depende del producto en el que se aplique que tendremos distintos enfoques y participantes además de los ya mencionados. Los siguientes cuadros resumen esto:

Prod. 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.

Prod. de trabajo	Enfoque de la inspección	Participantes
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 ★

El proceso de software produce muchos items como programas, documentos, datos, manuales, etc. Cualquiera de estos items puede cambiar, por lo que con el progreso y estado de cada uno de ellos, se controlará mediante la SCM. Ésta analizará los cambios y la evolución de los mismos (items) durante el tiempo. Entonces, la SCM:

- Controla sólo los productos del proceso de desarrollo
- Contiene los items producidos en el proceso de desarrollo.
- Si se produce un cambio sobre alguno de los items, se generan distintas versiones dentro de la SCM.
- Chequea que las versiones dentro de la SCM se combinen apropiadamente sin pérdidas.

Funcionalidades necesarias de las SCM

- Recolectar todas las fuentes, documentos, entre otros, del sistema.
- Evitar cambios o eliminaciones desautorizadas.
- Deshacer cambios o revertir a una versión especificada.
- Hacer disponible la última versión del programa.

Mecanismos de la SCM: técnicas para gestionar elementos a lo largo de su ciclo de vida.

- Control de acceso: gestión de quién tiene permisos para modificar, acceder o hacer cambios en los elementos.
- Control de versiones: llevar registro de las versiones y revisiones de los elementos a lo largo del tiempo. Ayuda preservar viejas versiones y deshacer cambios.
- Identificación de configuración: capacidad de etiquetar y asignar un identificador único a cada elemento y versión del software.
- entre otros

Items de la configuración: son los elementos o unidades de modificación, sus cambios se siguen rigurosamente y deben combinarse apropiadamente de manera periódica.

- Para facilitar la combinación, se usan baselines, que son una colección específica y apropiada de items en un estado dado. Establece un punto de referencia en el desarrollo, es como un “checkpoint” de un repositorio. Éstos forman la base estable a partir de la cuál se realizan cambios posteriores.

Para llevar a cabo este proceso, debemos definir las actividades que requieren un control sobre sus cambios. Para esto tenemos las siguientes fases:

- **Planeamiento:** identificar items, definir estructura del repositorio, el control de acceso, puntos de referencia, etc.
- **Ejecución:** realizar los procedimientos según lo planeado.
- **Auditoría:** para verificar que no se cometieron errores.

4. Proceso de administración de cambios de requisitos ★

Los requerimientos pueden cambiar en cualquier momento durante el desarrollo. Éstos cambios pueden producir impacto en los productos y en los distintos items de configuración. La idea es que éstos cambios se controlen a partir del siguiente proceso:

- Registrar cambios
- Hacer un análisis de impacto sobre los productos e items: o sea, ver qué cambios desencadenan.
- Estimar impacto en esfuerzo y cronograma.
- Analizar el impacto con las personas involucradas: para saber si efectuarlo o no.
- Reprocesar los productos de trabajo y los ítems.

5. Proceso de administración de procesos

Los procesos no son estáticos, deben cambiar para mejorar la calidad y productividad. Éste proceso se enfoca en **evaluar** y **mejorar** los procesos para lograr eso.

- Recolecta datos de los proyectos para comprender el desempeño del proceso en los proyectos
- Requiere que el proceso esté bien documentado
- Requiere que sea apropiadamente ejecutado en los proyectos.

Los cambios para mejorar los procesos se hacen de forma **incremental** y se deben hacer en **pasos pequeños** cuidadosamente seleccionados (los pasos).

Existen marcos que ayudan a proceder en la mejora del proceso, uno de ellos es CMM - Capability Maturity Model.

CMM - Capability Maturity Model 🚫