

U1: INGENIERÍA DE SOFTWARE EN CONTEXTO

Temas:

- ✓ Introducción a la Ingeniería del Software. ¿Qué es?
- ✓ Estado Actual y Antecedentes. La Crisis del Software.
- ✓ Disciplinas que conforman la Ingeniería de Software.
- ✓ Ejemplos de grandes proyectos de software fallidos y exitosos.
- ✓ Ciclos de vida (Modelos de Proceso) y su influencia en la Administración de Proyectos de Software.
- ✓ Procesos de Desarrollo Empíricos vs. Definidos.
- ✓ Ciclos de vida (Modelos de Proceso) y Procesos de Desarrollo de Software
- ✓ Ventajas y desventajas de c/u de los ciclos de vida. Criterios para elección de ciclos de vida en función de las necesidades del proyecto y las características del producto.
- ✓ Componentes de un Proyecto de Sistemas de Información.
- ✓ Vinculo proceso-proyecto-producto en la gestión de un proyecto de desarrollo de software.

COMPONENTES DE PROYECTO DE DESARROLLO DE SOFTWARE

Esto es una introducción a la gestión tradicional de desarrollo de Software (o Procesos Definidos).

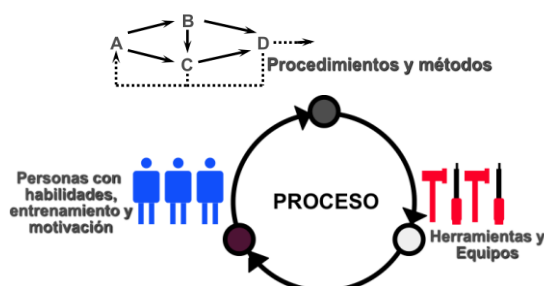
El Proceso de Software

Transforma ideas, necesidades (entradas) en un producto de software de calidad.

Cuando estructuramos el Desarrollo de Software utilizamos el concepto de Proceso; que agrupa un conjunto de Actividades, que tienen un objetivo (entregar producto de SW), utiliza recursos y suman personas para cumplir ese objetivo.

La industria del Software es "Humano Intensiva". Es decir, el SW lo hacen las personas y son la parte más importante de hacer software. La parte más costosa del desarrollo de SW es la mano de obra (o el costo del esfuerzo). Hoy en día es el 80% del costo total.

En los procesos están involucradas las Personas (son el componente principal); Procedimientos y Métodos; y Herramientas y Equipos (ayudan a automatizar el desarrollo). Es importante que haya un equilibrio entre estos componentes; que conforman el concepto de Proceso.



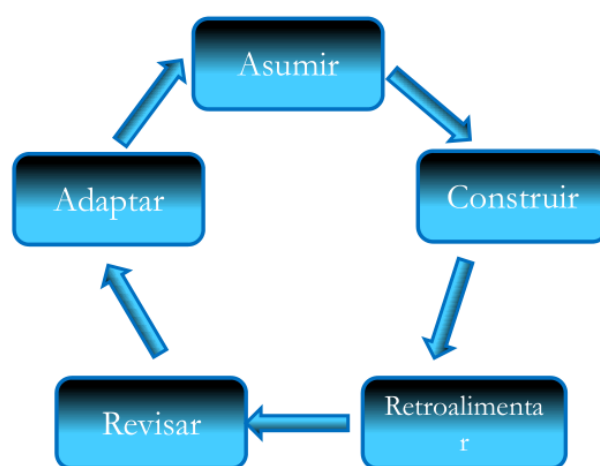
Proceso Definido vs Empírico

Los procesos Empíricos se basan en la experiencia. Esa experiencia se debe generar.

Los anteriores surgen en contraposición a los Procesos Definidos, vienen del taylorismo y de las líneas de producción. Plantea una sucesión de pasos bien definidos. El PUD es un ejemplo. Luego ese proceso se adapta a cada caso en particular. Su intención además es ser repetibles. Y esa repetibilidad se quiere para tener visibilidad y predecir lo que va a pasar en función de saber cuánto tiempo requiere hacer el trabajo. Se quiere tener una ilusión de control. La forma de encarar un proyecto de SW es siempre el mismo (Requerimientos, Diseño, Desarrollo, Testeo,

Despliegue...), pero la forma de encarar (el nivel de definición de) esas actividades son distintas. El PUD y el RUP son los referentes más utilizados de este tipo de procesos. Se intenta también de que sean Procesos Completos.

Los procesos Empíricos no están completos. Son en realidad lineamientos, heurísticas, guías o mejores prácticas que cubren algún aspecto del desarrollo. No cubren la totalidad de actividades a hacer y queda en manos del equipo definirlos al iniciar el trabajo. Lo decide en base a la experiencia. El empirismo se basa en ciclos de entrega cortos para generar realimentación que sirva como experiencia para evolucionar y seguir avanzando. Los procesos empíricos trabajan basados en una hipótesis: asumen que algo puede funcionar de cierta forma, hacen algo, obtiene realimentación, revisan y adaptan. Así es como el equipo va ganando experiencia. Otra diferencia es que los procesos empíricos dicen que la experiencia es aplicable al mismo equipo, a un proyecto y contexto específico.



Los procesos definidos esperan que haya repetibilidad en el sentido de que, si se cumplen las mismas condiciones de trabajo, el resultado debería ser el mismo o aunque sea debería llegar a una solución similar.

Estos dos son enfoques filosóficos.

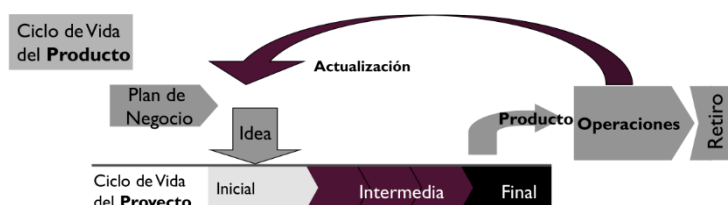
Ejemplos de Empirismo es el Lean y el Ágil. La gestión tradicional de proyectos basada en el PMI está basada en procesos definidos.

Ciclos de Vida

Son distintos a los procesos. Uno debe elegir un ciclo de vida para un proceso. El proceso elige el paso a paso para cumplir el objetivo. El ciclo de vida del proyecto dice cómo encarar las actividades, orden y cuanto de cada actividad hacer. Ejemplos: Cascada (hacer 100% de la

actividad para pasar al siguiente), iterativo e Incremental (hacer un poco de cada uno y después repite las actividades).

Los Productos tienen ciclos de vida distinto al de los Proyectos. Los de los productos son ciclos de vida más largos: inicia con la idea de hacer un producto y termina con la retirada del mercado. Mientras transcurre el ciclo de vida del producto puede haber "n" ciclos de vida de proyectos que van a producir otras versiones del producto.



La necesidad de cambio dispara (además de nuevos requerimientos) el comienzo de un nuevo proyecto. Así se vinculan el ciclo de vida del producto y del proyecto.

Un ciclo de vida de un proyecto software es una representación de un proceso. Grafica una descripción del proceso desde una perspectiva particular.

Hay tres tipos básicos de Ciclos de Vida para un proyecto de desarrollo de software:

- Secuencial (como el Cascada)
- Iterativo/Incremental (Iterativo Incremental del PUD)
- Recursivo (como el ciclo de vida en Espiral)

Cuando se trabaja con Procesos Definidos se puede elegir cualquier ciclo de vida. Los Empíricos demandan ciclos de vida Iterativos/Incrementales porque necesitan generar experiencia por retroalimentación.

Las cuatro P

Las personas son las involucradas en el Proyecto y autoras del Producto de SW que se quiere producir. El Proyecto organiza el trabajo (es la unidad organizativa) y tiene por resultado el Producto de SW. Es un medio por el cual se puede administrar los recursos que se necesitan y las Personas; para obtener como resultado un Producto o Servicio (de SW). El proyecto empieza decidiendo quiénes van a ser las personas que se van a involucrar en el trabajo y que van a asumir roles definidos

en los procesos. El proyecto necesita de personas y recursos, y de un método o indicación de cómo hacer el trabajo para cumplir con el objetivo. Se dice entonces que el Proyecto instancia al Proceso y lo adapta a sus necesidades. El proyecto arma las tareas a realizar.

El Proceso es la definición (teórica) de un conjunto de actividades organizadas por flujos de trabajo, que toman como entrada requerimientos y obtienen como salida un producto o servicio de software y las Herramientas ayudan a automatizar parte de ese desarrollo y esas actividades. Ese conjunto de actividades está estructurada y guiada por un objetivo.

El proyecto debe tomar de esa definición lo que parece que le haga falta con un criterio de necesidad, de corrección y de adecuación. Toma la definición y arma el conjunto de tareas que es necesario ayer y que conforman el alcance del proyecto a partir del proceso. El alcance se define como el trabajo que hay que hacer para cumplir con el objetivo.

El resultado es el Producto. El producto es el Software. Software se puede definir como conocimiento empaquetado a distintos niveles de abstracción (BD, Diseño de BD, Documentación de la Arquitectura, Diseño de Cus, Código ejecutable, etc.). Son los entregables de las tareas del proyecto. Cada Flujo de Trabajo (o tarea) tiene sus distintos tipos de entregables (artefactos) que también son Software. Cada una de estos Artefactos que se pueden gestionar y versionar se le puede llamar ítem de Configuración.

Estos son los ejes de la ingeniería de Software.

Nota: Scrum o XP son lineamientos (o FW) no procesos

PROYECTOS

Tienen ciertas características:

- Tienen un objetivo (lo que se quiere obtener), que actúa de guía. Debe cumplir con dos restricciones/condiciones: debe ser claro y alcanzable (que pueda lograr dedicando los recursos disponibles y necesarios).
- El Resultado que se obtiene es único (en contexto, versionado o en su naturaleza misma).

Único: distinto de los productos resultantes de otros proyectos. Aún siendo el mismo producto, el contexto en que se crea es distinto.

- No es de flujo constante. Es decir, tiene una fecha de inicio y una fecha de fin. Cuando termina, se liberan los recursos (incluido RRHH, que asumen roles en otros proyectos).
- Las tareas están interrelacionadas.

El objetivo del proyecto (no es el mismo que del producto) define el trabajo que el proyecto debe hacer para satisfacer al cliente (o stakeholders).

Interrelación de Tareas: Para alcanzar el objetivo (claro y alcanzable), teniendo los recursos disponibles, se debe dividir el trabajo en tareas. Esas tareas tienen una relación entre ellas. Esas tareas son sacadas del proceso (para no tener que invitarlas).

Administración de proyectos (Project Management en la Gestión tradicional)

"Lo importante es el acto de planificar, no el plan (resultado)"

No siempre planificar un proyecto hace que no fracase.

De planificar se aprende sobre el proyecto. Después se decide qué decisiones (conocimiento) deben quedar documentadas para compartir la misma visión.

Cuando se habla de administración se habla de que, a partir del planteo del Objetivo, se debe administrar todos los recursos disponibles, organizar el trabajo de la gente afectado, realizar seguimiento y, a partir de esas actividades es que se intenta guiar el proyecto hacia el cumplimiento del objetivo.

Otra forma de definirlo es: "Mas académicamente... administración de proyectos es la aplicación de conocimientos, habilidades, herramientas y técnicas a las actividades del proyecto para satisfacer los requerimientos del proyecto".

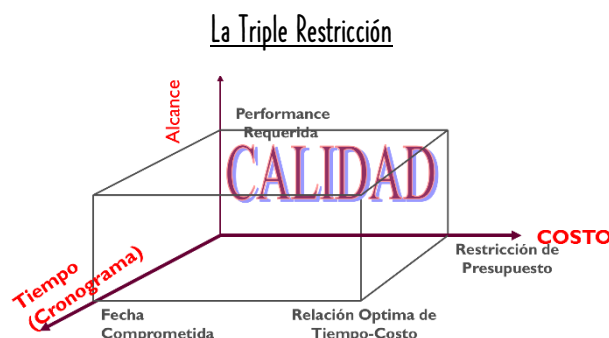
Teniendo gente, recursos y un objetivo no es suficiente para completar un proyecto. Es necesaria la administración.

La Planificación y el Seguimiento y Control son las dos tareas más importantes de la administración.

Administrar un proyecto incluye:

- Identificar los requerimientos
- Establecer objetivos claros y alcanzables
- Adaptar las especificaciones, planes y el enfoque a los diferentes intereses de los involucrados (stakeholders).

La gestión Ágil se define en contraposición con la gestión tradicional.



Para lograr entregar ese Producto o servicio se necesita Tiempo, Recursos (y personas) que generan Costos. El Alcance del producto es la dimensión que maneja el cliente. Son los requerimientos del producto. Idealmente se debe poder controlar dos de estas tres dimensiones. Si no hay margen de maniobra, el administrador no tiene nada qué hacer. Si una dimensión cambia, las otras dos también. La dimensión de Alcance no se cobra por sí sola, sino que se cobra luego de hacer el producto (definir los requerimientos no se cobra) y esta dimensión es la que condiciona las otras dos dimensiones.

Si existe desequilibrio entre las dimensiones, se sacrifica la calidad. Y si esto pasa, hay tareas que se recortan del proceso de desarrollo de SW.

(Gestión tradicional) Rol del Líder del Proyecto/Equipo

El Líder se relaciona con los Stakeholders y es el que dice qué, cuándo hay que hacer las tareas que hay que hacer. El equipo recibe las asignaciones de trabajo. Debe también informar el término del trabajo, cuánto tiempo le tomó, el porcentaje de avance, y si hay una desviación de los planes, debe actuarse en consecuencia a eso.

Como líderes se tiene la responsabilidad de: Estimar, Planificar, Asignar trabajo a la gente y hacer seguimiento. El trabajo del líder del proyecto es simplemente esto y no trabaja en lo técnico (idealmente).

Equipo de Proyecto es: un grupo de personas comprometidas en alcanzar un conjunto de objetivos de los cuales se sienten mutuamente responsables.

Tienen las siguientes características:

- Diversos conocimientos y habilidades

- Posibilidad de trabajar juntos efectivamente /desarrollar sinergia
- Usualmente es un grupo pequeño
- Tienen sentido de responsabilidad como una unidad

Plan de Proyecto

El líder de proyecto debe poder obtener un plan de proyecto. Ese Plan de Proyecto es un mapa que define el "viaje" o cómo va a realizarse el proyecto. Es el artefacto resultante del proceso de planificación.

Abarca las preguntas:

- ¿Qué es lo que hacemos? (Referido al Alcance)
- ¿Cuándo lo hacemos? (Referido a la Calendarización)
- ¿Cómo lo hacemos? (Referido a los Recursos)
- ¿Quién lo va a hacer? (Referido a la Asignación de Tareas)

Planificar un proyecto de Software implica:

- Definición del Alcance del Proyecto
- Definición de Proceso y Ciclo de Vida
- Estimación
- Gestión de Riesgos
- Asignación de Recursos
- Programación de Proyectos (Calendarización)
- Definición de Controles (Incluye Comunicación con el equipo y el cliente para ir revisando requerimientos y progreso del proyecto; y a su vez, qué informes se deben generar)
- Definición de Métricas (Para ir controlando el progreso del Proyecto)

Las métricas (son valores numéricos) hacen falta para dar una definición/valoración cuantitativa (u objetiva) del proyecto.

Estas actividades se deben hacer en proyectos basados en procesos definidos y con administración tradicional. Hay distintas formas de hacerlo.

Alcance del Producto (medida: requerimientos)

Son todas las características (requerimientos funcionales y no funcionales) que pueden incluirse en un producto o servicio. Son los Requerimientos definidos en el documento de Especificación de Requerimientos de Software.

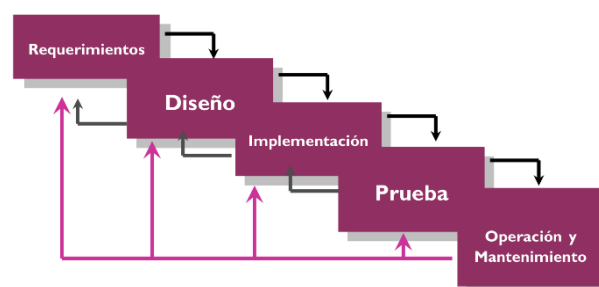
Alcance del Proyecto (medida: tareas)

Es todo el trabajo y solo el trabajo que debe hacerse para entregar el producto o servicio con todas las características y funciones especificadas. Son las tareas que se deben realizar para cumplir con el objetivo del Proyecto. Esas tareas se guardan en el Plan del Proyecto.

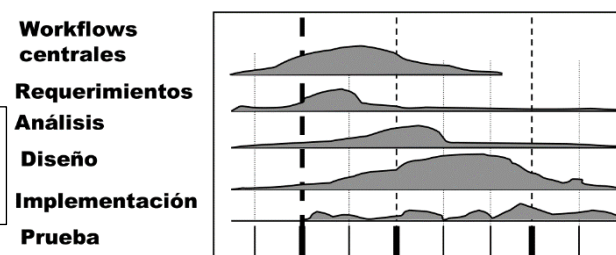
La relación entre ambos conceptos es que, por ejemplo, la cantidad de requerimientos que debe tener el producto es proporcional a la cantidad de tareas a realizar. Además, para definir el Alcance del Proyecto se debe definir primero el Alcance del Producto.

Definir un ciclo de Vida

Ciclo de vida en cascada:



Ciclo de vida del PUD:



El ciclo de vida define cómo se van a hacer las tareas y cuánto de cada tarea hacer. Al planificar el proyecto inicialmente se debe definir qué proceso usar y qué ciclo de vida se quiere usar (en la gestión ágil no se puede elegir el ciclo de vida).

Estimaciones de Software

- Tamaño
- Esfuerzo (Horas persona ideales)
- Calendario (¿Cuándo?)
- Costo
- Recursos Críticos (Son recursos que se necesitan para que el proyecto avance; ejemplo: dispositivos para los cuales desarrollo software que necesito físicamente como alarmas de incendio)

Esta secuencia de estimaciones debe cumplirse en la gestión tradicional.

Bibliografía:

- Steve McConnell - Rapid Development