

Introducción a la Ingeniería de Software

FAQ

1. Que es software?
2. ¿Cuáles son los atributos del buen software?
3. ¿Qué es ingeniería de software?
4. ¿Cuáles son las actividades fundamentales de la ingeniería de software?
5. ¿Cuáles son los principales retos que enfrenta la ingeniería de software?

Software (Pressman):

1. **Instrucciones** (programas de computadora) que cuando se ejecutan proporcionan la **función** y el **rendimiento** deseados.
2. **Estructuras de datos** que permiten a los programas manipular adecuadamente la información.
3. **Documentos** que describen la construcción y uso de programas

Características del Software (Pressman)

1. El software se desarrolla, no se fabrica

- a. Los costes se centran en ingeniería, no en fabricación
- b. Los proyectos software no se pueden gestionar como procesos de fabricación

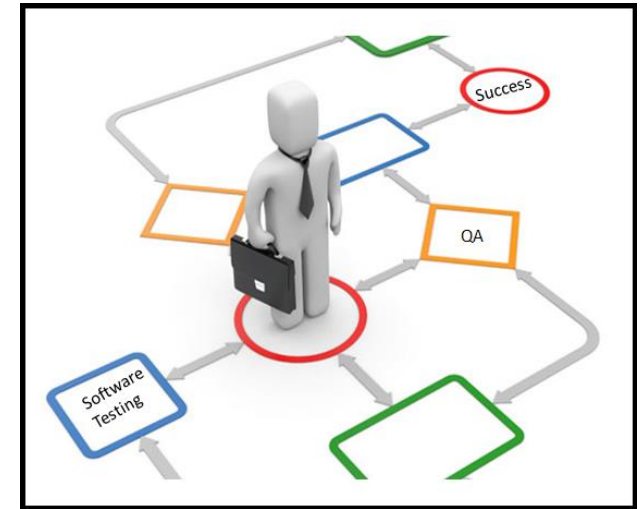
2. El software no se estropea: no es susceptible a males del entorno.

3. El software se deteriora. Durante su vida sufre cambios (mantenimiento), por lo que es posible de introducir nuevos defectos

La Crisis del Software

Los productos de software:

1. No cumplen con los requisitos del usuario.
2. Son susceptible frecuentemente a romperse.
3. Es caro.
4. Es difícil de modificar, depurar y mejorar.
5. A menudo es entregado tarde.
6. Tiene a utilizar recursos no óptimos.



Factores que contribuyen a la Crisis del Software

1. Problemas cada vez más grandes,
2. La falta de una formación adecuada en ingeniería de software,
3. El aumento de la falta de habilidades,
4. Baja productividad.



Programas vs Productos de Software

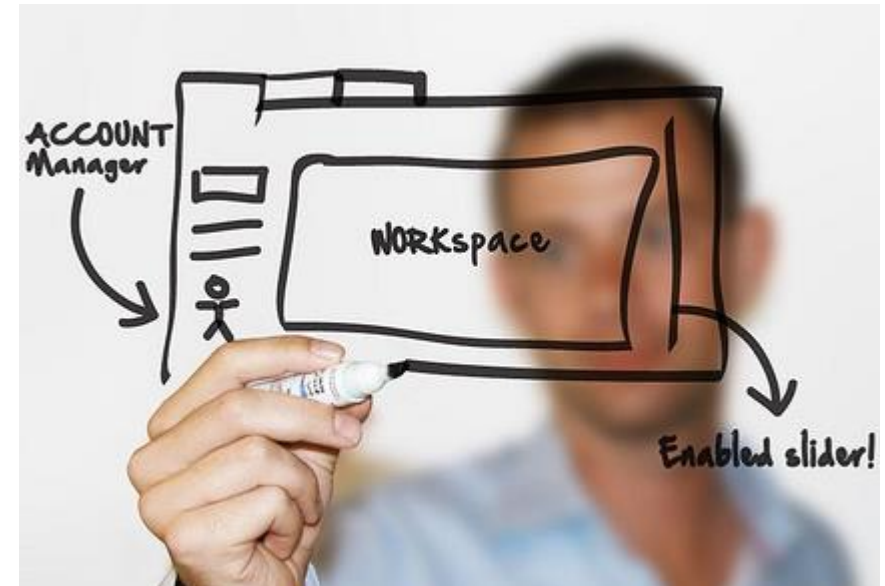
Programas	Productos de Software
Usualmente pequeños en tamaño	Tamaño grande
El autor es su único usuario	Gran número de usuarios
Un desarrollador	Equipo de desarrolladores
Carece de interfaz de usuario adecuada	Interfaz bien diseñada
Pérdida de la documentación	Buena documentación / Manuales de usuario
Desarrollo Ad hoc	Desarrollo sistemático

Ingeniería de Software: Una actividad de resolver problemas

Ingeniería de Software

Colección de **técnicas, metodologías y herramientas** para ayudar con la producción de:

1. Un software de alta calidad
2. Con un presupuesto dado
3. Antes del plazo determinado
4. Mientras el cambio se produce



Ingeniería de Software: Una actividad de resolver problemas

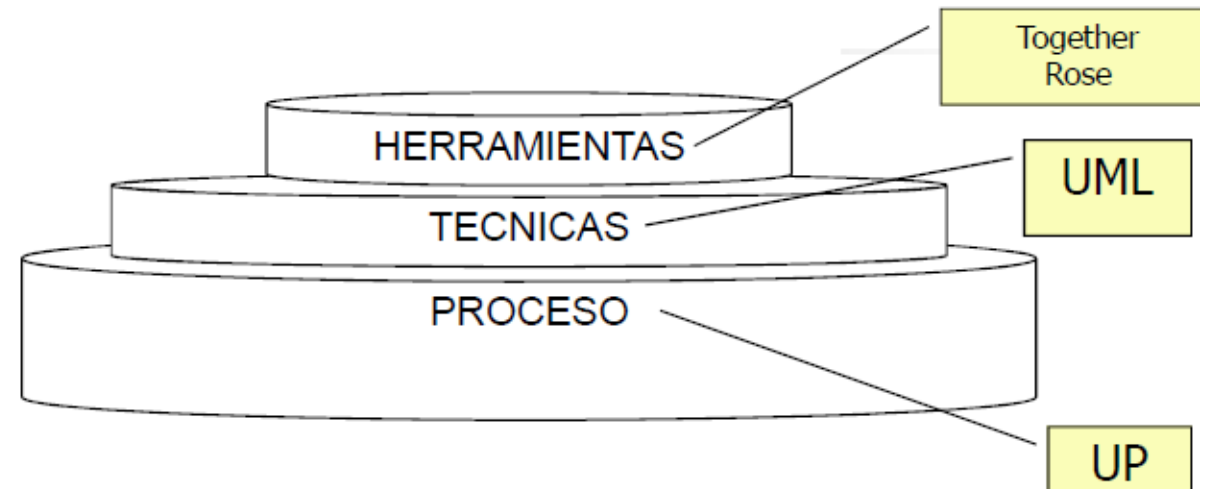
Para resolver un problema se usa:

- 1. Técnicas (métodos):** Procedimientos formales para producir resultados usando alguna notación bien definida (ej. técnicas de programación)
- 2. Metodologías (Proceso y Modelos):** Colección de técnicas aplicadas a través del desarrollo de software y unificado bajo un enfoque filosófico (Iterativo, incremental, RAD, RUP, XP, etc.)
- 3. Herramientas:** instrumentos o sistemas automatizados para llevar a cabo una técnica

Ingeniería de Software: Una actividad de resolver problemas



Capas de la Ingeniería de Software. Pressman



Cual es la diferencia entre Ingeniería de Software y un programador?

“Un Ingeniero de software sabe cómo programar, pero no todos los programadores son Ingenieros de Software. La Ingeniería de software es más que la programación. Involucra esfuerzos de grupo donde los integrantes tienen roles y responsabilidades diferentes que se complementan, donde hay que definir estrategias para resolver los problemas, utilizar tecnologías y procesos.

Los Ingenieros de software desarrollan software para cumplir con las especificaciones establecidas por clientes o para satisfacer las necesidades de un mercado. Los proyectos de Ingeniería de Software tienen plazos, fechas de lanzamiento, y una considerable interacción entre las personas responsables de los distintos componentes. Un Ingeniero de Software sabe aprender y utiliza efectivamente lenguajes y tecnologías que están en cambio permanente.”

¿Por qué estudiar Ingeniería de Software?

1. Adquirir **habilidades** para desarrollar programas grandes.
2. El crecimiento exponencial del nivel de **complejidad y dificultad** con el tamaño.
3. El **enfoque ad hoc se rompe** cuando el tamaño del software incrementa

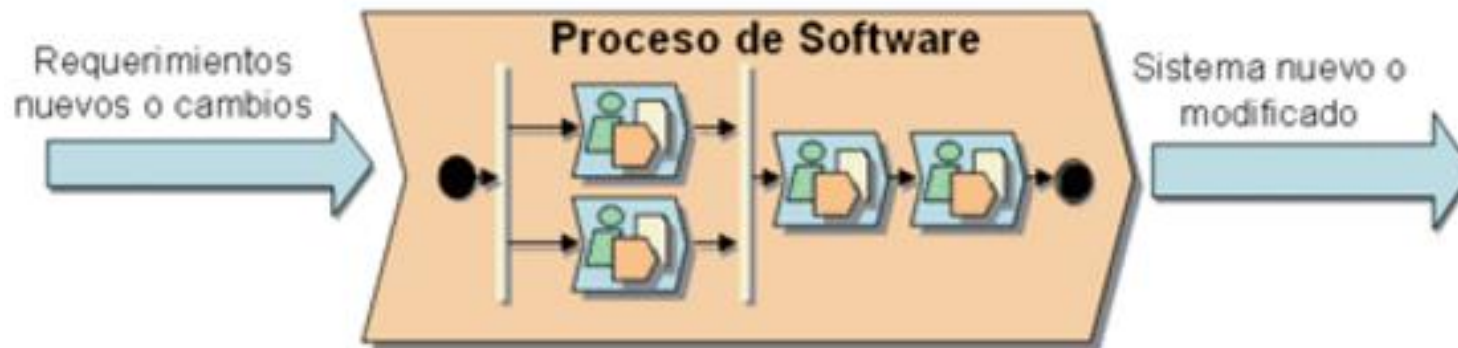


*Una espina de experiencia es mejor que todo
un desierto de advertencia.*



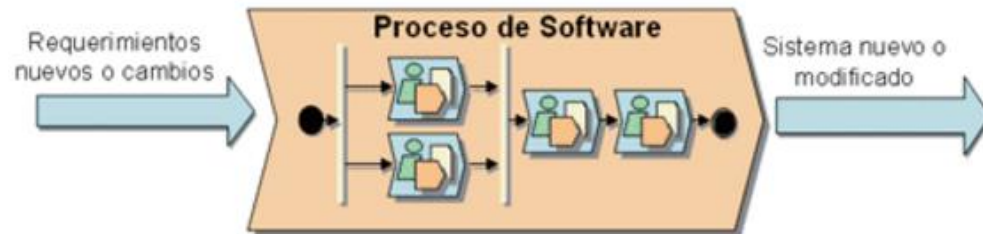
El proceso de software

Propósito: producción eficaz y eficiente de un producto software que reúna los requisitos del cliente.



Este proceso es intensamente intelectual, afectado por la creatividad y juicio de las personas involucradas.

El proceso de software



Elementos Típicos del Proceso de Software

Actividad

Definen las acciones que se llevan a cabo en un momento dado del desarrollo de software.

Flujo de Trabajo

Colección estructurada de actividades y elementos asociados (artefactos y roles), que producen un resultado de valor.

Rol

Son responsables por llevar a cabo las actividades del proceso, pueden ser personas o herramientas.

Producto o Artefacto

Son las entradas y salidas de las actividades, pueden ser de diferentes tipos, como documentos, modelos, componentes, planes, reportes, etc.

Disciplina

Conjunto integrado por actividades relativas a una rama particular de conocimiento. Ej. Análisis y diseño.

El proceso de software

El proceso de software no es único. No existe un proceso de software universal que sea efectivo para todos los contextos de proyectos de desarrollo.

Debido a esta diversidad, es difícil automatizar todo un proceso de desarrollo de software.

El proceso de software

A pesar de la variedad de propuestas de proceso de software, existe un conjunto de actividades fundamentales que se encuentran presentes en todos ellos:

1. **Especificación de software:** Se debe definir la funcionalidad y restricciones operacionales que debe cumplir el software.
2. **Diseño e Implementación:** Se diseña y construye el software de acuerdo a la especificación.
3. **Validación:** El software debe validarse, para asegurar que cumpla con lo que quiere el cliente.
4. **Evolución:** El software debe evolucionar, para adaptarse a las necesidades del cliente.

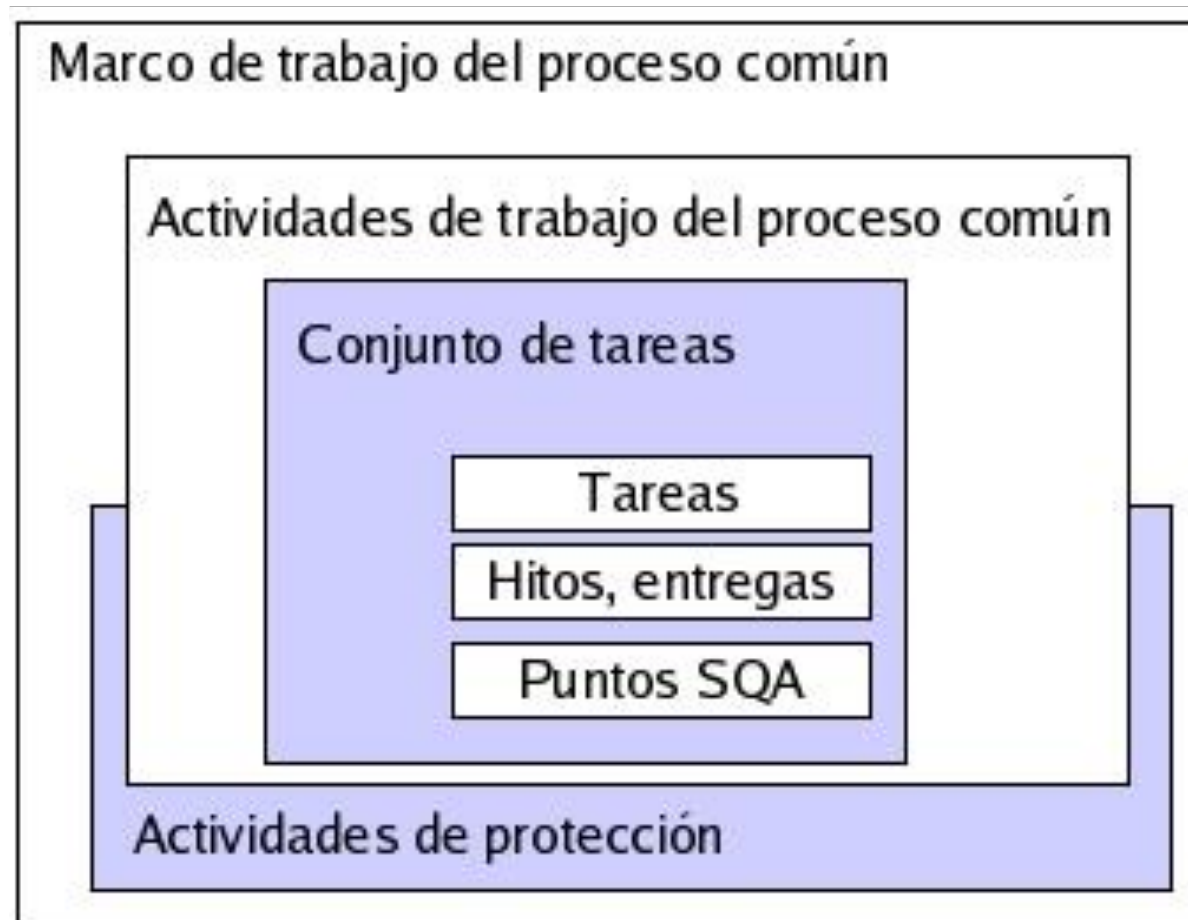
El proceso de software

Además de estas actividades fundamentales, Pressman menciona un conjunto de “actividades protectoras”, que se aplican a lo largo de todo el proceso del software. Ellas se señalan a continuación:

1. Seguimiento y control de proyecto de software.
2. Revisiones técnicas formales.
3. Garantía de calidad del software.
4. Gestión de configuración del software.
5. Preparación y producción de documentos.
6. Gestión de reutilización.
7. Mediciones.
8. Gestión de riesgos.

El proceso de software

Pressman caracteriza un proceso de desarrollo de software con los siguientes elementos:



El proceso de software

Pressman caracteriza un proceso de desarrollo de software con los siguientes elementos:

1. **Un marco común del proceso:** actividades que son aplicables a todos los proyectos de software, con independencia del tamaño o complejidad.
2. **Un conjunto de tareas:** colección de tareas de ingeniería del software, hitos de proyectos, entregas y productos de trabajo del software, y puntos de garantía de calidad, que permiten que las actividades del marco de trabajo se adapten a las características del proyecto de software y los requisitos del equipo del proyecto.
3. **Las actividades de protección,** tales como garantía de calidad del software, gestión de configuración del software y medición, abarcan el modelo del proceso. Las actividades de protección son independientes de cualquier actividad del marco de trabajo y aparecen durante todo el proceso.

Modelos de proceso software

Sommerville define el modelo de proceso de software como:

*“Una **representación simplificada de un proceso de software**, representada desde una perspectiva específica. Por su naturaleza los modelos son simplificados, por lo tanto un modelo de procesos del software **es una abstracción de un proceso real**.”*

Los modelos genéricos no son descripciones definitivas de procesos de software; sin embargo, son abstracciones útiles que pueden ser utilizadas para explicar diferentes enfoques del desarrollo de software.

Modelos de proceso software

Algunos modelos son:

1. Codificar y corregir
2. Modelo en cascada
3. Desarrollo evolutivo
4. Desarrollo formal de sistemas
5. Desarrollo basado en reutilización
6. Desarrollo incremental
7. Desarrollo en espiral

Modelos Genéricos

Abarcan todos los procesos relacionados con el desarrollo de software

CMM – modelo de madurez de capacidades – estándar de facto

CMMI – modelo integrado

ISO 9001-2000 – sistema para administración de la calidad - estándar

ISO/IEC 15504 – marco para evaluación de procesos de software – en vías de ser estándar, por ahora reporte técnico

MoProSoft – modelo de procesos para la industria de software en México – en vías de ser norma mexicana

- Nos dicen qué debemos hacer
- Se deben usar como referencia para definir procesos en una organización y para autoevaluación.
- Medio para evaluar que tan bien o mal esta la organización.

Modelos Específicos

Enfocados a la ingeniería de productos de software

UP – proceso de desarrollo

RUP – proceso de desarrollo

PSP – enfocado en individuos

TSP – enfocado en equipos (incluye PSP)

- Nos dicen el cómo debemos hacer las cosas
- Se usan como guía para ejecutar proyectos