

**Estudiante:** Ángel Gabriel Vargas Varela

**Grupo:** 20

**Curso:** POO

**Profesor:** Martin Flores

**Tema:** Resumen capítulo 2 Libro Pressman

**Año:** 2022



## 2.1 Un modelo general de proceso:

Una estructura general para la ingeniería de software define cinco actividades estructurales: **comunicación, planeación, modelado, construcción y despliegue**. Además, a lo largo de todo el proceso se aplica un conjunto de actividades sombilla: seguimiento y control del proyecto, administración de riesgos, aseguramiento de la calidad, administración de la configuración, revisiones técnicas, entre otras.

Un flujo de proceso lineal ejecuta cada una de las cinco actividades estructurales en secuencia, comenzando por la comunicación y terminando con el despliegue. Un flujo de proceso iterativo repite una o más de las actividades antes de pasar a la siguiente. Un flujo de proceso evolutivo realiza las actividades en forma “circular”.

## 2.2 Evaluación y mejora del proceso:

La existencia de un proceso del software no es garantía de que el software se entregue a tiempo, que satisfaga las necesidades de los consumidores o que tenga las características técnicas que conducirán a características de calidad de largo plazo. Los patrones de proceso deben acoplarse con una práctica sólida de ingeniería de software.

En las últimas décadas se han propuesto numerosos enfoques para la evaluación y mejora de un proceso del software:

- Método de evaluación del estándar CMMI para el proceso de mejora
- Evaluación basada en CMM para la mejora del proceso interno
- SPICE
- ISO9001:2000 para software

## 2.3 Modelos de proceso prescriptivo:

Los modelos de proceso prescriptivo fueron propuestos originalmente para poner orden en el caos del desarrollo de software. El autor los llama “prescriptivos” porque prescriben un conjunto de elementos del proceso: actividades estructurales, acciones de ingeniería de software, tareas, productos del trabajo, aseguramiento de la calidad y mecanismos de control del cambio para cada proyecto. Cada modelo del proceso también prescribe un flujo del proceso, también llamado flujo de trabajo, es decir, la manera en la que los elementos del proceso se relacionan entre sí.

Cuando el trabajo desde la comunicación hasta el despliegue fluye en forma razonablemente lineal, se encuentra en ocasiones cuando deben hacerse adaptaciones o mejoras bien definidas a un sistema ya existente, por ejemplo, una adaptación para software de contabilidad que es obligatorio hacer debido a cambios en las regulaciones gubernamentales. También ocurre en cierto número limitado de nuevos esfuerzos de desarrollo, pero sólo cuando los requerimientos están bien definidos y tienen una estabilidad razonable.

## 2.4 Modelos de proceso especializado:

Los modelos de proceso especializado tienen muchas de las características de uno o más de los modelos tradicionales que se han ido presentando. Sin embargo, dichos modelos tienden a

aplicarse cuando se elige un enfoque de ingeniería de software especializado o definido muy específicamente.

El modelo del desarrollo basado en componentes lleva a la reutilización del software, y eso da a los ingenieros de software varios beneficios en cuanto a la mensurabilidad. Si la reutilización de componentes se vuelve parte de la cultura, el equipo de ingeniería de software tiene la posibilidad tanto de reducir el ciclo de tiempo del desarrollo como el costo del proyecto

## **2.5 El proceso unificado:**

En cierto modo, el proceso unificado es un intento por obtener los mejores rasgos y características de los modelos tradicionales del proceso del software, pero en forma que implemente muchos de los mejores principios del desarrollo ágil de software. El proceso unificado reconoce la importancia de la comunicación con el cliente y los métodos directos para describir su punto de vista respecto de un sistema. Hace énfasis en la importancia de la arquitectura del software y “ayuda a que el arquitecto se centre en las metas correctas, tales como que sea comprensible, permita cambios futuros y la reutilización”

- La fase de concepción del PU agrupa actividades tanto de comunicación con el cliente como de planeación
- La fase de elaboración incluye las actividades de comunicación y modelado del modelo general del proceso
- La fase de construcción del PU es idéntica a la actividad de construcción definida para el proceso general del software

## **2.6 Modelos del proceso personal y del equipo:**

El mejor proceso del software es el que está cerca de las personas que harán el trabajo. Si un modelo del proceso del software se ha desarrollado en un nivel corporativo, será eficaz sólo si acepta una adaptación significativa para que cubra las necesidades del equipo de proyecto que en realidad hace el trabajo de ingeniería de software. En la situación ideal se crearía un proceso que se ajusta del mejor modo a los requerimientos, y al mismo tiempo cubriera las más amplias necesidades del equipo y de la organización.

## **2.7 Tecnología del proceso:**

El equipo del software debe adaptar uno o más de los modelos del proceso estudiados en las secciones precedentes. Para ello, se han desarrollado herramientas de tecnología del proceso que ayudan a las organizaciones de software a analizar su proceso actual, organizar las tareas de trabajo, controlar y vigilar el avance, y administrar la calidad técnica. El modelo, que normalmente se representa como una red, se analiza para determinar el flujo de trabajo normal y se examinan estructuras alternativas del proceso que podrían llevar a disminuir el tiempo o costo del desarrollo

## **2.8 Producto y proceso:**

Si el proceso es deficiente, no cabe duda de que el producto final sufrirá. Pero también es peligrosa la dependencia excesiva del proceso. La gente obtiene tanta satisfacción del proceso creativo como del producto final. Un artista disfruta las pinceladas tanto como el resultado que enmarca. Un escritor goza de la búsqueda de la metáfora apropiada tanto como del libro

terminado. Como profesional creativo del software, usted también debe obtener tanta satisfacción del proceso como del producto final. La dualidad de producto y proceso es un elemento importante para hacer que personas creativas se involucren conforme la ingeniería de software evoluciona.

## **2.9 Resumen:**

- Un modelo general del proceso para la ingeniería de software incluye un conjunto de actividades estructurales y sombrilla, acciones y tareas de trabajo.
- Los modelos de proceso prescriptivo se han aplicado durante muchos años en un esfuerzo por introducir orden y estructura al desarrollo de software.
- Los modelos de proceso secuencial, como el de la cascada y en V, son los paradigmas más antiguos del software.
- Los modelos de proceso incremental son de naturaleza iterativa y producen con mucha rapidez versiones funcionales del software.
- El modelo de proceso concurrente permite que un equipo de software represente los elementos iterativos y concurrentes de cualquier modelo de proceso.

## **Preguntas:**

### **1. ¿Cuál es el problema que plantea el capítulo?**

Los problemas o conflictos que se dan a la hora de crear o planear un proyecto, debido a que, durante la planeación, muchos programadores no tienen un rumbo fijo, por lo que el capítulo formula diferentes formas o métodos para organizarse en caso de que haya conflictos durante la realización del proyecto.

### **2. ¿Por qué el problema es interesante o importante?**

Porque es un problema muy común en el mundo laboral, siempre hay conflictos o todo es un caos cuando las cosas no salen como se tenían previsto. Por eso es sumamente importante tener un proceso bien planteado y tener previsto los futuros problemas que pueden ocasionarse.

### **3. ¿Cuál es la solución propuesta por el autor?**

Crear diferentes tipos de modelos para el procedimiento de un proyecto, creando planes para diferentes tipos de situaciones y amoldarse en situaciones específicas, creando así los modelos general, prescriptivo, unificado, personal y de equipo y muchos otros. Siendo estos muy importantes para cuando haya problemas durante la creación de algún proyecto de software

### **4. ¿Qué tan exitosa es esta solución?**

Es muy utilizada, debido a que prevé futuros conflictos o situaciones que puedan ralentizar el proceso del software, a nivel laboral podríamos decir que son practicas muy utilizadas, gracias a que son moldeables para distintas situaciones a futuro.