

El Proceso Unificado

1. Introducción

El proceso de desarrollo de software define qué debe hacerse, quién debe hacerlo, cuándo y cómo debe ser hecho.

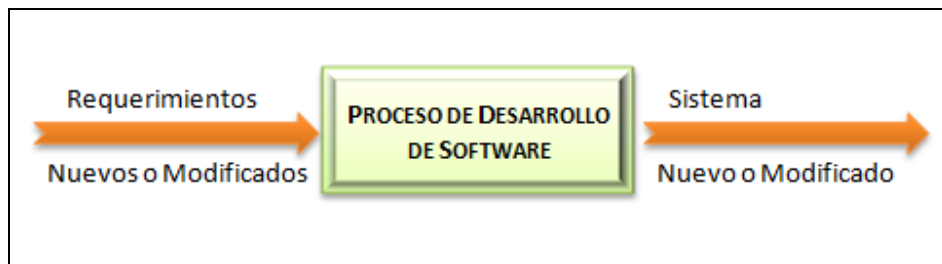


Figura 1. Modelo Entrada-Proceso-Salida (IPO)

No existe un proceso universal, las características de cada proyecto (equipo de desarrollo, recursos, etc.) exigen que el proceso sea “parametrizable” o adaptable a las necesidades del mismo, intentando cubrir estos requerimientos surge el Proceso Unificado.

2. Proceso Unificado

El Proceso Unificado es más que simplemente un proceso, es un marco de desarrollo de software (framework) que se puede adaptar fácilmente a una gran diversidad de sistemas, para diferentes áreas de aplicación, organizaciones, y tamaños de proyecto.

Un proceso común que guíe a los equipos de desarrollo trae implícitos, entre otros, los siguientes beneficios:

- ❖ Permite trabajar de manera coordinada
- ❖ Integra las múltiples facetas del desarrollo
- ❖ Proporciona un elemento común de comunicación
- ❖ Brinda una guía para ordenar las actividades del equipo
- ❖ Especifica lo que se debe producir en cada etapa

Así y todo, el proceso unificado no resuelve por sí solo las dificultades que conlleva un proyecto, la gente es quien los resuelve, pero un buen proceso ayuda a la gente a sobresalir como equipo. Un proceso bien definido y gestionado marca una diferencia notable entre los proyectos exitosos y aquellos que fracasan.

Características del Proceso Unificado

El UP se caracteriza principalmente por ser:

- ❖ Dirigido por casos de uso
- ❖ Centrado en la arquitectura
- ❖ Iterativo e incremental

Dirigido por casos de uso:

Cada iteración toma un conjunto de casos de uso o escenarios que definen la funcionalidad del sistema. Los casos de uso no solo limitan la especificación de los requisitos del sistema, sino que además guían el diseño, la implementación y las pruebas, es decir, guían el proceso de desarrollo, proporcionan un hilo conductor para el mismo.

Centrado en la arquitectura:

En el Proceso Unificado los casos de uso no se piensan en forma aislada, sino que se deben desarrollar en forma paralela a la arquitectura, ambos maduran a medida que avanza el ciclo de desarrollo (I. Jacobson, G. Boosh, & Rumbaugh, 2000). La arquitectura de software incluye los aspectos estáticos y dinámicos más significativos del sistema. Surge de una necesidad y se refleja en los casos de uso, se ve influenciada por muchos factores tales como la plataforma (sistema operativo, sistema de gestión de base de datos, protocolos de comunicación de red, etc.) componentes reutilizables, sistemas heredados, requisitos no funcionales, etc.

La arquitectura abarca decisiones importantes sobre:

- La organización del sistema.
- Los elementos estructurales que compondrán el sistema y sus interfaces.
- El estilo de la arquitectura que guía esta organización: los elementos y sus interfaces, sus colaboraciones y su composición.

Los casos de uso corresponden a la “función” mientras que la arquitectura da la “forma”, ninguna es suficiente por sí misma y deben evolucionar paralelamente.

Iterativo e incremental:

Se divide el trabajo en iteraciones o mini proyectos que, producen incrementos del producto o dicho de otro modo, resultados.

Las iteraciones hacen referencia a un paso por el flujo de trabajo y los incrementos al crecimiento del producto. En cada iteración, se identifica y especifican los casos de uso relevantes, se diseña utilizando la arquitectura seleccionada como guía, se implementa y se prueba esta versión funcional pero reducida del proyecto. Si la iteración cumple sus objetivos, se continúa con la próxima. Sino deben revisarse las decisiones previas y probar un nuevo enfoque.

Beneficios del enfoque iterativo e incremental:

- ❖ Como se van creando mini proyectos, es más sencillo soportar los posibles costos derivados de los mismos.
- ❖ Gracias a que en primer lugar se atiende a los riesgos más importantes, se reduce el riesgo del proyecto en general y, por lo tanto, es más factible cumplir los plazos establecidos.
- ❖ Los mini proyectos son proyectos más factibles de cumplir a corto plazo con lo que los trabajadores se sentirán mejor al poder alcanzar los objetivos y al ver una parte del proyecto funcionando.
- ❖ Permite obtener una visión de los requisitos globales del proyecto y poco a poco ir depurándolo, esto es muy importante debido a la dificultad que presenta el conocer todos los requisitos al inicio del desarrollo del sistema.
- ❖ Gracias al tamaño reducido del problema y los requisitos que se abordan en cada iteración, se consigue una retroalimentación rápida y se facilita el control de cambios.

3. El Ciclo de Vida del Proceso Unificado

El Proceso Unificado se repite a lo largo de una serie de ciclos que constituyen la vida de un sistema, cada uno de ellos constituye una versión del sistema.

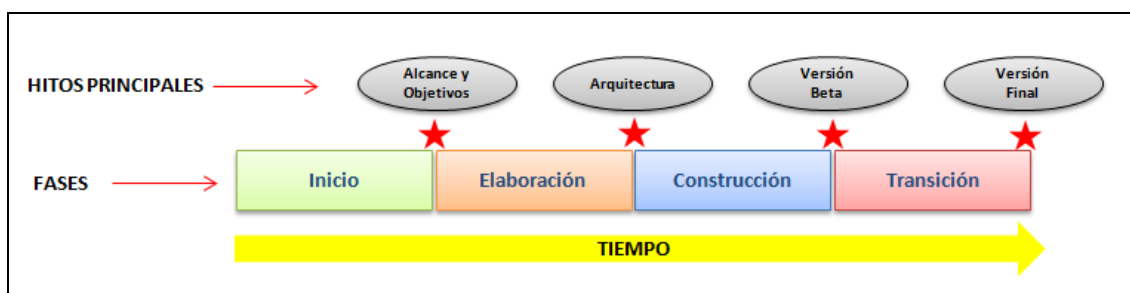


Figura 2. Fases e Hitos del Proceso Unificado

Fases: Cada ciclo, constas de cuatro fases: inicio, elaboración, construcción y transición, a su vez, cada fase se subdivide en *iteraciones*.

Disciplinas: En cada iteración se desarrolla en secuencia un conjunto de disciplinas, cada una de ellas es un conjunto de actividades relacionadas, o flujos de trabajo¹ (workflows) vinculados a un área específica dentro del proyecto. Las disciplinas más importantes son: *Requerimientos, Análisis, Diseño, Codificación, y Prueba*.

¹ Un flujo de trabajo o workflow describe la secuencia en que se realizan las actividades en una disciplina, quienes la realizan (trabajadores) y qué artefactos producen.

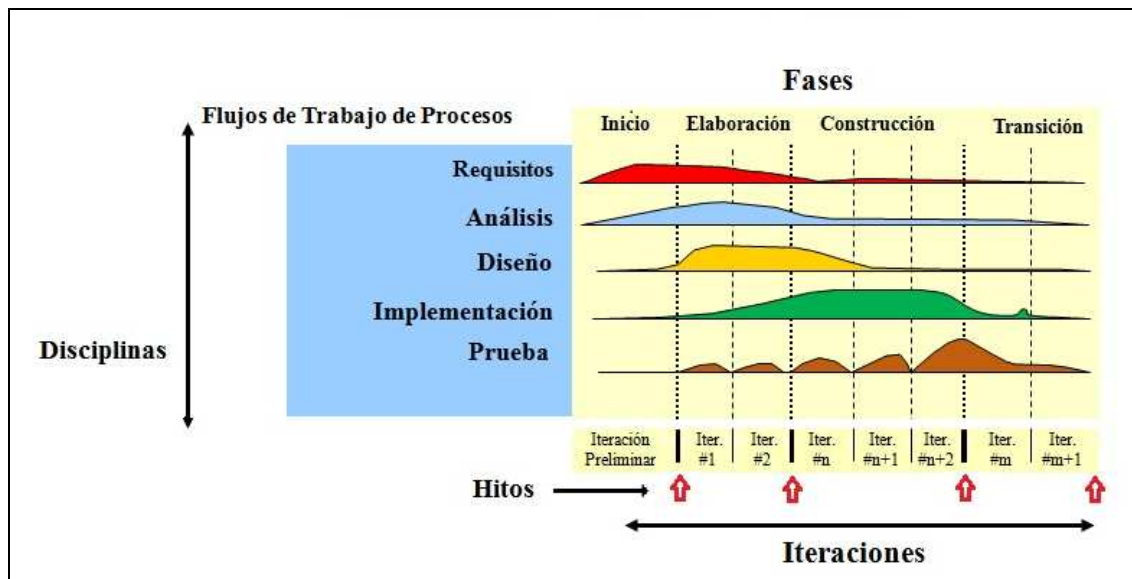


Figura 3. Estructura del Proceso Unificado

Cada disciplina está asociada con un conjunto de modelos que se desarrollan, compuestos por artefactos². Los artefactos más importantes son los modelos que cada disciplina realiza, en la figura 4 por ejemplo, se indican las dependencias entre el modelo de casos de uso y los demás modelos.

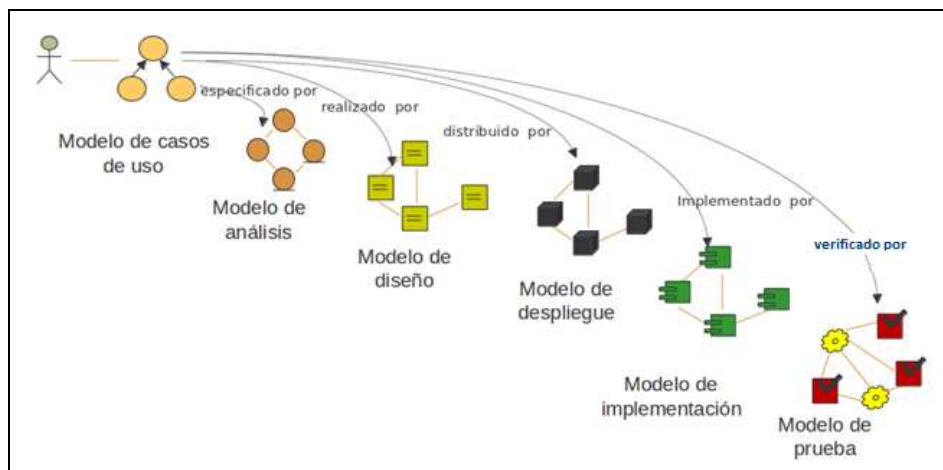


Figura 4. Modelos del Proceso Unificado. Fuente: I. Jacobson y otros

² Los artefactos (constructos) son un conjunto de elementos que se producen en cada actividad, pueden ser documentos, modelos, diagramas u otros elementos útiles para poder clarificar el desarrollo del proyecto.

4. Fases del Proceso Unificado

Cada ciclo transcurre a lo largo del tiempo, y dicho tiempo se divide en fases tal como se muestra en la figura 3. A su vez, cada fase del proceso unificado finaliza con un hito.

Cada hito se define por un conjunto de documentos o artefactos que permiten que los responsables de la dirección de un proyecto puedan saber si se está realizando el trabajo de forma adecuada y en la dirección adecuada. Los hitos además de tener objetivos como el expuesto anteriormente, tienen un objetivo crítico que es permitir la toma de decisiones antes de pasar a la siguiente fase del proyecto.

Como se explicó anteriormente el Proceso Unificado propone las fases: inicio, elaboración, construcción y transición. A continuación se da una breve descripción de cada una de estas:

Inicio³:

El objetivo de esta fase es ayudar al equipo de proyecto a decidir cuáles son los verdaderos objetivos del sistema a desarrollar. Las iteraciones exploran diferentes soluciones, arquitecturas posibles y se identifican y priorizan los riesgos más importantes.

Se responden las siguientes preguntas(I.Jacobson, G.Boosh, & Rumbaugh, 2000):

- ¿Cuáles son las principales funciones del sistema para los usuarios más importantes?
- ¿Cómo podría ser la arquitectura del sistema?
- ¿Cuál es el plan del proyecto y cuanto costará desarrollar el producto?

Los artefactos que suelen producirse en esta fase son:

- Un enunciado de los mayores requerimientos planteados como casos de uso.
- Un boceto inicial de la arquitectura.
- Una descripción de los objetivos del proyecto.
- Una versión muy preliminar del plan del proyecto.
- Un modelo del negocio.

La fase de inicio finaliza con el hito de Alcances y Objetivos, el cual se alcanza cuando el equipo de proyectos y las personas involucradas (stakeholders) llegan a un acuerdo sobre cuál es el conjunto de necesidades del negocio, y qué funciones satisfacen estas necesidades.

Elaboración

En esta fase se obtiene una visión más concreta del proyecto, se especifican en detalle la mayoría de los casos de uso, se diseña la arquitectura, se establece un plan detallado para las siguientes iteraciones y se controlan los riesgos críticos.

³ La fase de inicio también es conocida como “Fase de Concepción”

En esta fase se construyen generalmente los siguientes artefactos:

- El cuerpo básico del software en un prototipo de la arquitectura.
- Los casos de prueba.
- La mayoría de los casos de uso que describen la funcionalidad del sistema.
- Un plan detallado para las siguientes iteraciones.

La fase de elaboración finaliza con el hito de la Arquitectura, el que se alcanza cuando el equipo de desarrollo y los stakeholders llegan a un acuerdo sobre los casos de uso que describen la funcionalidad del sistema, la línea base de la arquitectura, los mayores riesgos han sido controlados y se elaboró un plan del proyecto.

Construcción

Durante esta fase se construye el producto, a partir del modelo de arquitectura de la fase anterior se desarrollan o adquieren los componentes que harán que cada caso de uso sea operativo para los usuarios finales. Al final de esta fase el producto contiene todos los casos de uso acordados para esa versión. Sin embargo aún pueden existir errores, muchos defectos se descubrirán y solucionarán en la siguiente fase.

Los artefactos producidos son:

- El sistema software
- Los casos de prueba
- Los manuales de usuario

La fase de construcción finaliza con el hito de la obtención de la Versión Beta o de capacidad operativa inicial, el cual se alcanza cuando el producto es estable para ser usado, provee alguna funcionalidad de valor y las partes están listas para comenzar la transición.

Transición

Durante esta fase se realizan pruebas sobre el sistema, centrándose en corregir posibles errores y en incorporar algunas mejoras. Incluye actividades como la fabricación, capacitación del cliente corrección de defectos detectados tras la entrega. Cuando el producto se haya finalizado, la versión beta de los usuarios será intercambiada por la versión final. Para esta fase habrá tantas iteraciones como características se le añadan al producto.

Los artefactos construidos en esta fase son los mismos que en la fase de construcción. El equipo se encuentra ocupado fundamentalmente en corregir y extender la funcionalidad del sistema desarrollado en la fase anterior.

La fase finaliza con el hito de lanzamiento del producto u obtención de la versión final, se alcanza cuando el equipo de desarrollo y los stakeholders llegan a un acuerdo sobre el cumplimiento de los objetivos fijados en la fase de Inicio.

Bibliografía

I.Jacobson, G.Boosh, G., & Rumbaugh, J. (2000). *El Proceso Unificado de Desarrollo de Software*. Madrid: Pearson Educación S.A.