

**UNIVERSIDAD TECNOLOGICA DEL CENTRO DE VERACRUZ**

**LECTURA TECNICA:**

**EL PROCESO UNIFICADO DE DESARROLLO DE SOFTWARE**

**MATERIA:**

**INGENIERIA DE SOFTWARE**

**PROFESRO:**

**AGUILAR GARCIA HONORATO**

**ALUMNA:**

**RAMIREZ MEDINA ELIZABETH**

**MATRICULA: 6825**

**Índice**

[**CAPITULO 1** 2](#_Toc420613667)

[**El proceso unificado de desarrollo de software** 2](#_Toc420613668)

[**1.2 EL PROCESO UNIFICADO DE PALABRAS** 3](#_Toc420613669)

[**1.3. DIRIJIDO POR CASO DE USO** 3](#_Toc420613670)

[**1.4 EL PROCESO UNIFICADO ES ITERANTIVO E INCREMENTAL** 3](#_Toc420613671)

[**1.4.1 PRODUCTO** 4](#_Toc420613672)

[**1.4.2 FASES DE UN CICLO** 4](#_Toc420613673)

[Figura 1.4.2 5](#_Toc420613674)

[**1.4.3 Un proceso integrado** 5](#_Toc420613675)

[**CAPITULO II** 6](#_Toc420613676)

[**Las cuatro “P” en el desarrollo de software: Persona, Proyecto, Producto y Proceso** 6](#_Toc420613677)

[**Figura II** 6](#_Toc420613678)

[**2.1.1 los procesos de desarrollo afectan a las personas** 7](#_Toc420613679)

[Figura 2.1.1 8](#_Toc420613680)

[**2.2 LOS PROYECTOS CONSTRUYEN EL PRODUCTOS** 8](#_Toc420613681)

[**2.3 EL PRODUCTO ES CODIGO** 9](#_Toc420613682)

[**2.4 UN SISTEMA POSEE LA COLENCCION DE MODELOS** 9](#_Toc420613683)

[Figura 2.4 10](#_Toc420613684)

[Figura 2.4 10](#_Toc420613685)

[**2.5 PROCESO DIRIJE LOS PROYECTOS** 10](#_Toc420613686)

[**2.6 HERRAMIENTAS QUE DAN SOPORTE AL CICLO DE VIDA** 11](#_Toc420613687)

# **CAPITULO 1**

# **El proceso unificado de desarrollo de software**

La tendencia actual del software lleva a la construcción sistemas más grandes y más complejos .nuestro a petito de software aún más sofisticado crese a medida que vemos como pueden mejorar los productos de una versión a otra. Queremos un software que este mejor adaptado a nuestras necesidades, por eso a su vez, a ser al software más complejo .en breve queremos más.

El problema del software se reduce en la dificultad que afrontan los desarrolladores para coordinar las múltiples cadenas de trabajo de un gran proyecto de software.

Los cuatro procesos unificados del desarrollo de software

Proporcionar una guía para ordenar las actividades por equipo

Dirigir las tareas de cada desarrollador por separado y el equipo como un todo.

Especifiquen los artefactos que van a desarrollar

Ofrezcan criterios para el control y la medición de productos y actividades del proyecto

# **1.2 EL PROCESO UNIFICADO DE PALABRAS**

Es un proceso de desarrollo de software un conjunto de actividades necesarias para trasformar los requisitos de un usuario en un sistema de software, el proceso unificado está basado en componentes lo cual quiere decir que el sistema software en construcción esta formados por componentes software (apéndice a) interconectado atreves de interfaz (apéndice a) bien definidas.

Los verdaderos aspectos definitorios del proceso unificado se resumen en tres fases claves, dirigidas por caso de uso, centrado en la arquitectura, e interactivo e incrementales. Esto hace único al proceso unificado.

# **1.3. DIRIJIDO POR CASO DE USO**

Un caso de uso es un fragmento de funcionalidad del sistema que proporciona al usuario un resultado importante .el caso de uso representa los requerimientos funcionales. Los casos de usos no solo es una herramienta para especificar requisitos también guía su diseño, implementación y prueba. De este modo, los casos de usos no solo inician el proceso de desarrollo sino que le proporciona un hilo conductor.

Proceso unificado está basado en la arquitectura

Los conceptos de arquitectura de software incluyen los aspectos estáticos y dinámicos más significativos del sistema

# **1.4 EL PROCESO UNIFICADO ES ITERANTIVO E INCREMENTAL**

La iteración hace referencia a paso en el flujo de trabajo y los incrementos, al incremento del producto.

Los desarrolladores se basan la selección de los que se implementaran en una interacción en dos factores.

En cada iteración, los desarrolladores identifican y especifican los casos de unos relevantes, creando un diseño utilizando la arquitectura deseada como guía, implementar el diseño mediantes componentes y verifiquen que los componentes satisfacen los casos de usos. Cuando una iteración no cumple sus objetivos, los desarrolladores deben revisar sus decisiones previas y probar con un nuevo enfoque.

Estos conceptos de desarrollo dirigido por el caso de usos, centrado en la arquitectura, iterativo e incremental son de igual importancia .la eliminación de uno de las tres ideas reducirá drásticamente el valor del proceso unificado.

Ahora que hemos presentado los tres conceptos claves, es el momento de echarle un vistazo al proceso de su totalidad, su ciclo de vida, artefacto, flujo de trabajo, fases e iteración.

La 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 ciclo consta de cuatro fases inicio elaboración y transición. Cada fase se subdivide a su vez iteraciones. Los ciclos concluyen con una versión:

Nacimiento y muerte

Tiempo

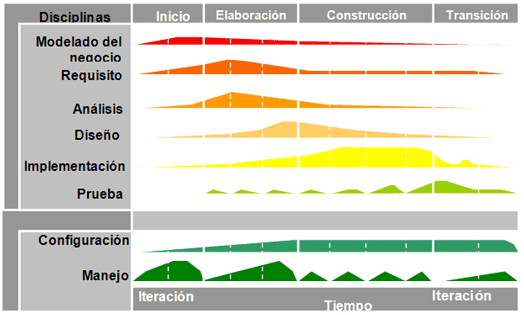
Un ciclo con sus fases iteración versión: inicio, elaboración, construcción y tracción las cuales generan un tiempo.

# **1.4.1 PRODUCTO**

Cada ciclo produce una versión del sistema, y cada versión es un producto preparado para la entrega. Consta de un cuerpo de código fuente incluido en componentes que puede compilarse y se, además los manuales y productos generados. El producto terminado incluye los requisitos, casos de usos, especificaciones no funcionales ya casos de usos. Incluye el modelo de la arquitectura y el modelo visual. Los requisitos cambien es una de las constante del desarrollo del software. Al final los desarrolladores deben a afrontar un nuevo ciclo, y los directores deben financiar. Para llevar a cabo siguiente ciclo de manera eficiente, los desarrolladores necesitan todas las representaciones del producto del software. Modelo de caso de uso, modelo de análisis, modelo de diseño, modelo de despliegue, modelo de implementación, modelo de prueba.

# **1.4.2 FASES DE UN CICLO**

Cada ciclo se desarrolla a lo largo del tiempo. Este tiempo a su vez, se divide en cuatro fases como se muestra en la figura .A través de una secuencia de modelos, los implicados visualizan lo que esta consiente en esta fase. Cada fase termina con hito .cada hito se determina por la disponibilidad de un conjunto de artefacto; es decir, cierto modelos o documentos han sido desarrollados hasta alcanzar un estado predefinido.



## Figura 1.4.2

Durante la fase de inicio, se desarrolla una descripción del producto final a partir de una buena idea y se representa el análisis de negocio para el producto.

Durante la fase de elaboración: el director del proyecto está en disposición de planificar las actividades y estimar los recursos necesarios para determinar el proyecto.

Durante la fase contracción: se crea el producto, se añade los músculos (software terminado) al esqueleto (la arquitectura).

La fase de transacción: cubre el periodo durante el cual el producto se convierte en versión beta .los desarrolladores corrigen los problemas e incorporan algunas de las sugerencias en una versión general dirigida a la totalidad de la comunidad de usuarios.

# **1.4.3 Un proceso integrado**

Es proceso unificado está basado en componentes, utiliza el nuevo estándar del modelo visual, el lenguaje unificado de modelo (UML), y se sostiene sobre tres ideas básicas casos de usos, arquitectura y desarrollo iterativo e incremental.

El propósito de este libro es describir el proceso unificado con énfasis particular en las facetas de ingeniería, en las tres ideas claves (caso de uso, arquitectura, y desarrollo iterativo e incremental).describiremos las cuatro fases y los diferentes flujos de trabajos, como planificar un proyecto, planificación de recursos, gestión de riesgo, control de configuración, recogida de métrica y control de cálida.

# **CAPITULO II**

# **Las cuatro “P” en el desarrollo de software: Persona, Proyecto, Producto y Proceso**

El resultado final de un Proyecto Software es un producto que toma forma durante su desarrollo gracias a la intervención de muchos tipos distintos de personas.

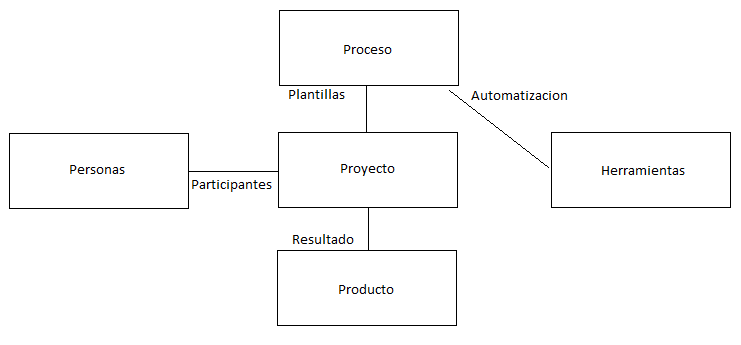
Personas: Los principales autores de un proyecto Software son los arquitectos, desarrolladores, ingenieros de prueba y el personal de gestión.

Proyecto: Elemento organizativo a través del cual se gestiona el desarrollo del software.

Producto: Artefactos que se crean durante la vida del proyecto.

Proceso: Un proceso de ingeniería de software es una definición del conjunto completo de actividades necesarias para transformar los requisitos de usuario en un producto.

Herramientas: Software que se utiliza para automatizar las actividades definidas en el proceso.



## **Figura II**

**2.1 LAS PERSONAS SON DESICIVAS**

Hay personas implicadas en el desarrollo de un producto de software durante todo el ciclo de vida. Financia el producto, lo planifican, lo desarrollan, lo gestiona, lo prueba, lo utiliza y se beneficia de él. El proceso de guía este desarrollo debe orientarse a la persona, es decir debe funcionar bien para las personas que lo utilizan.

# **2.1.1 los procesos de desarrollo afectan a las personas**

Conceptos como la viabilidad, la planeación del proyecto y la facilidad de compresión del proyecto tienen un papel importante:

Volatilidad del proyecto: la mayoría de la gente no disfruta trabajando, en proyectos que parecen imposible, los proyectos que no son viables pueden detenerse una fase temprana, aliviando así los problemas de moral.

Gestión de riesgo: cuando la gente siente que los riesgos no han sido analizados y reducidos se siente incómodo.

Estructura de los equipos: la gente trabaja de manera eficiente en grupos de seis u ocho miembros.

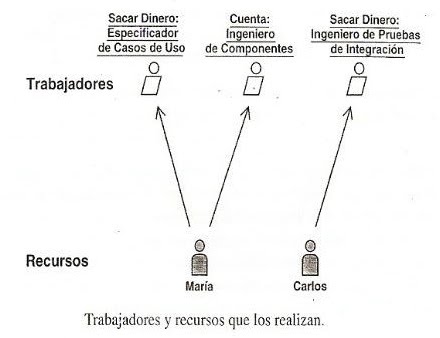
Clasificación del proyecto: la técnica que se utilizan en la fase de inicio y de elaboración permite a los desarrolladores tener una buena notación de cual debería ser el resultado del proyecto.

Facilidad de compresión del proyecto: tener una compresión global, la descripción de la arquitectura tiene una versión general para cualquier que se encuentre implicado en el proyecto.

Sanción de cumplimiento: un ciclo de vida iterativo, la retroalimentación frecuente y las conclusiones obtenidas aceleran el ritmo de trabajo.

Convertir un “recurso” en “trabajadores”

Una organización enfrenta una tarea esencial siempre que hace una persona pase de recurso "latente" a un puesto de "trabajador". La palabra "trabajador" la usamos para denominar a los puestos a los cuales se pueden asignar persona, el termino rol para hablar de los papeles que cumple un trabajador. Un trabajador puede asumir roles en relación con otros trabajadores en diferentes flujos de trabajos. Cada trabajador es responsable de un conjunto de actividades necesarias para el diseño de un subsistema.



# Figura 2.1.1

# **2.2 LOS PROYECTOS CONSTRUYEN EL PRODUCTOS**

Muchas veces cuando un cliente pide que le construyan una solución, siempre pregunta ¿cuánto me va a costar? Pues bien, todo producto requiere estimaciones cuantitativas y una adecuada planificación. Una adecuada recolección de información y un análisis detallado de los requerimientos proporciona la información necesaria para dar una estimación del costo del producto. Antes de planear un proyecto, se debe establecer los objetivos y el alcance que tendrá el proyecto, además de sus restricciones técnicas y de gestión. Con una buena planificación se puede estimar el tiempo que tomará desarrollar o construir el producto y redimensionar el valor cuantitativo del producto.

El desarrollador del software debe reunirse con el cliente las veces que sean necesarias para definir el dominio y los objetivos del producto. Esta actividad, comienza con la aplicación del proceso de ingeniería de requisitos; captura, análisis y, validación y verificación.

Definidos los objetivos y el dominio del producto se determina soluciones alternativas y viables, estas soluciones permitirán a los gestores del proyecto seleccionar las mejores opciones que convengan para cumplir con las restricciones que tenga la construcción del producto, sean estas de tiempo, presupuestarias, de personal, etc.

Para lograr rapidez en la construcción del producto, se debe dividir la carga de trabajo entre el equipo de desarrollo, es decir, dividir el problema. Esto, con el fin de desarrollar con mayor eficiencia y eficacia y en el tiempo acordado con el cliente, el producto.

# **2.3 EL PRODUCTO ES CODIGO**

En el contexto de proceso unificado, el producto que se obtiene es un sistema software. El termino producto aquí se hace referencia al sistema entero, y no solo al código que se entrega.  
  
Un sistema son todos los artefactos que se necesitan para representarlos en una forma comprensible para maquinas u hombres, para las maquinas, los trabajadores y los interesados.

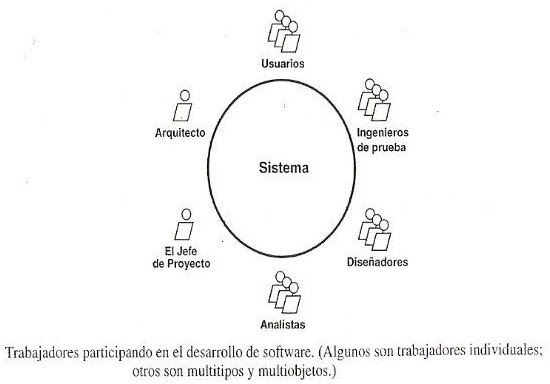
Artefactos

Es un término general para cualquier información creada, producida, cambiada o utilizada por los trabajadores en el desarrollo del sistema.

Básicamente, hay dos tipos de artefactos: artefactos de ingeniería y artefactos de gestión. Los de ingeniería creada durante las distintas fases de proceso (requisitos, análisis, diseño, implementación y prueba). Los de gestión tienen un tiempo de vida corto, lo que dura la vida del proyecto; A este conjunto pertenecen artefactos como el análisis de negocios, el plan de desarrollo (incluyendo el plan de versiones e interacciones).

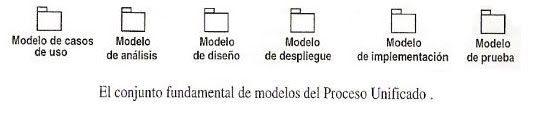
# **2.4 UN SISTEMA POSEE LA COLENCCION DE MODELOS**

Cada trabajador necesita una perspectiva diferente del sistema, las perspectivas recogidas de todos los trabajadores se encuentran en unidades más grandes, es decir, modelos de modo que un trabajador pueda pueda tomar una perspectiva concreta del conjunto de modelos.



## Figura 2.4

El Proceso Unificado proporciona un conjunto de modelos cuidadosamente seleccionando con cual empezar. Este conjunto de modelos hace claro el sistema para todos los trabajadores, incluyendo a los clientes, usuarios y jefes del proyecto.



## Figura 2.4

# **2.5 PROCESO DIRIJE LOS PROYECTOS**

LA PALABRA PROCESO ES UN TERMINO DEMASIADO UTILIZADO.EN ESTOS NEGOCIOS TAMBIEN EXISTEN OTRAS PROCESO, COMO EL PROCESO DE SOPORTE QUE INTERACTUA CON LOS USUARIOS DE LOS PRODUCTOS Y UN PROCESO DE VENTA.

EL PROCESO: UNA PLANTILLA

Proceso de desarrollo de software: conjunto complementos de actividades para convertir los requisitos de usuarios en un conjunto consiste de artefacto, que confirme un producto software.

La palabra requisito se utiliza con un sentido general, refiriéndole a “necesitad”

El resultado de valor añade del proceso es un conjunto consistente de artefacto, una línea base que conforma una aplicación o una familia de ellas que forman un producto de software.

Las actividades relacionadas conforman flujo de trabajo

* **CAPTURA DE REQUISITOS:**
  + IDENTIFICAR REQUISITOS DEL SISTEMA
  + CONSTRUIR UN MODELO DEL MISMO
    - MODELO DE CASOS DE USO
    - MODELO DEL DOMINIO (o NEGOCIO)
* **ANÁLISIS:**
  + ESPECIFICAR REQUISITOS
  + CONSTRUIR MODELO DEL ANÁLISIS
* **DISEÑO:**
* ENCONTRAR LA FORMA DEL SISTEMA (SOLUCIÓN)
* CONSTRUIR MODELO DEL DISEÑO
* **IMPLEMENTACIÓN:**
* CODIFICAR EL DISEÑO (SOLUCIÓN)
* CONSTRUIR MODELO DE IMPLEMENTACIÓN
* **PRUEBAS:**
* VERIFICAR LA IMPLEMENTACIÓN
* CONSTRUIR MODELO DE PRUEBAS

# **2.6 HERRAMIENTAS QUE DAN SOPORTE AL CICLO DE VIDA**

Hay herramientas que soportan todos los aspectos del ciclo de vida del software:

Gestión de requisitos: se utiliza para almacenar, examinar, revisar, hacer el seguimiento y navegación por los diferentes requisitos de un proyecto de software.

Modelo visual: se utiliza para automatizar el uso de UML.Para modelar y ensamblar una aplicación visual

Herramientas de programación: Proporciona una gama de herramientas, incluyendo editores, compiladores, depuradores, detectores de errores y analizadores de rendimiento.

Aseguramiento de calidad: Se utiliza para probar aplicación y componentes, para registrar y ejecutar casos de pruebas que dirigen la prueba de un IGU y de la interfaces de un componente.

Además de estas herramientas orientados a la funcionalidad, hay otras herramientas que abarcan todo ciclo de la vida estas herramientas incluyen las de control de versión, gestión de la configuración, seguimiento de defecto, documentación, gestión de proyecto y automatización de proceso.