

Metodología RUP



RATIONAL UNIFIED PROCESS

Forma disciplinada de asignar tareas y responsabilidades en una empresa de desarrollo (quién hace qué, cuándo y cómo).

RUP está estructurado en 2 dimensiones:

FASES: son las 4 etapas por las que atraviesa el proyecto.

DISCIPLINAS: representan la serie de actividades lógicas que surgen a través del proyecto.

Fases RUP :

Inicio (Inception)



- Se define el proyecto
- Se estiman los costos y el calendario de trabajo
- Se definen los riesgos
- Se desarrolla el proyecto a trabajar
- Se prepara el ambiente del proyecto
- Se identifica la arquitectura

Elaboración (Elaboration)



- Se especifican los requerimientos a gran detalle
- Se valida la arquitectura
- Evoluciona el ambiente del proyecto
- Se establece el personal que trabajará en el proyecto

Construcción (Construction)



- Se moldea, construye y se prueba el sistema
- Se desarrolla el soporte de la documentación

Transición (Transition)



- Se prueba el sistema
- El usuario prueba el sistema
- Se identifican fallas y se hacen mejoras
- Se implementa el sistema

1



Inicio (Inception)

El objetivo general de esta fase es establecer un acuerdo entre todos los interesados, se necesita comenzar a documentar el requerimiento, que será la base para empezar el desarrollo del proyecto, se definirá el alcance del mismo y sus objetivos, así como sus hitos correspondientes.

Al término de esta fase los involucrados en el proyecto deberán estar de acuerdo en:

El alcance del proyecto

Los requerimientos iniciales deben estar bien identificados y documentados

El plan de desarrollo del proyecto es realista

Los riesgos han sido identificados y manejados de manera apropiada

2



Elaboración (Elaboration)

El objetivo en esta fase es establecer la arquitectura base del proyecto para proveer bases estables para el desarrollo.

El requerimiento debe de estar lo más detallado posible para entender los riesgos en la arquitectura del proyecto, y para asegurarse que lo establecido en el mismo sea lo que en realidad se estará desarrollando.

Al término de esta fase los involucrados en el proyecto deberán estar de acuerdo en:

La visión del proyecto se ha establecido y es realista

Los requerimientos del proyecto están en orden

La arquitectura es estable y es suficiente para satisfacer los requerimientos

Los riesgos continúan y han sido manejados

Los gastos ocurrentes son razonables y aceptables sobre el estimado

El equipo del proyecto tienen grandes posibilidades de éxito

El plan de iteraciones para la siguiente construcción de iteraciones es correcta y a lugar

3



Construcción (Construction)

El objetivo de la fase es desarrollar el proyecto al punto que esté listo para su implementación; se moldea, construye y se prueba el sistema (proyecto). Se priorizan los requerimientos y se completan de una manera específica para tener soporte de la documentación.

En esta fase se puede obtener retroalimentación del proyecto.

Al término de esta fase los involucrados en el proyecto deberán estar de acuerdo en:

El proyecto y la documentación es aceptable para su desarrollo

Los involucrados en el proyecto están de acuerdo con la implementación

Los riesgos continúan y se manejan de manera efectiva

Los gastos ocurrentes son razonables y aceptables sobre el estimado

El plan de iteraciones para la siguiente construcción de iteraciones es correcta y a lugar

4



Transición (Transition)

Esta fase se enfoca en la entrega a producción del proyecto, El área de producción se encargará de las pruebas a los usuarios y su entrenamiento.

En este punto, la retroalimentación de los usuarios se centra en depurar el producto, configuraciones, instalación y aspectos sobre utilización.

Al término de esta fase los hitos deben de ser revisados por los involucrados en el proyecto y deberán estar de acuerdo en:

El proyecto y la documentación y el entrenamiento es aceptable para su desarrollo

Los gastos ocurrentes son razonables y aceptables, realizando un estimado de futuros costos

El proyecto puede operar una vez que está en producción

El proyecto puede ser soportado correctamente una vez que está en producción

RUP está dividido en una o más **iteraciones**, cada iteración tiene un plan con un objetivo en específico.

Una **iteración** es un ciclo completo de tareas dando como resultado una entregable u objetivo propuesto.

Durante cada iteración se puede alternar el trabajo de atrás a adelante entre las actividades de las disciplinas, es decir durante una iteración una porción de los requerimientos son seleccionados, analizados, designados, codificados, probados e integrados con los productos de iteraciones anteriores.

RUP define **nueve** disciplinas a realizar en cada fase del proyecto.

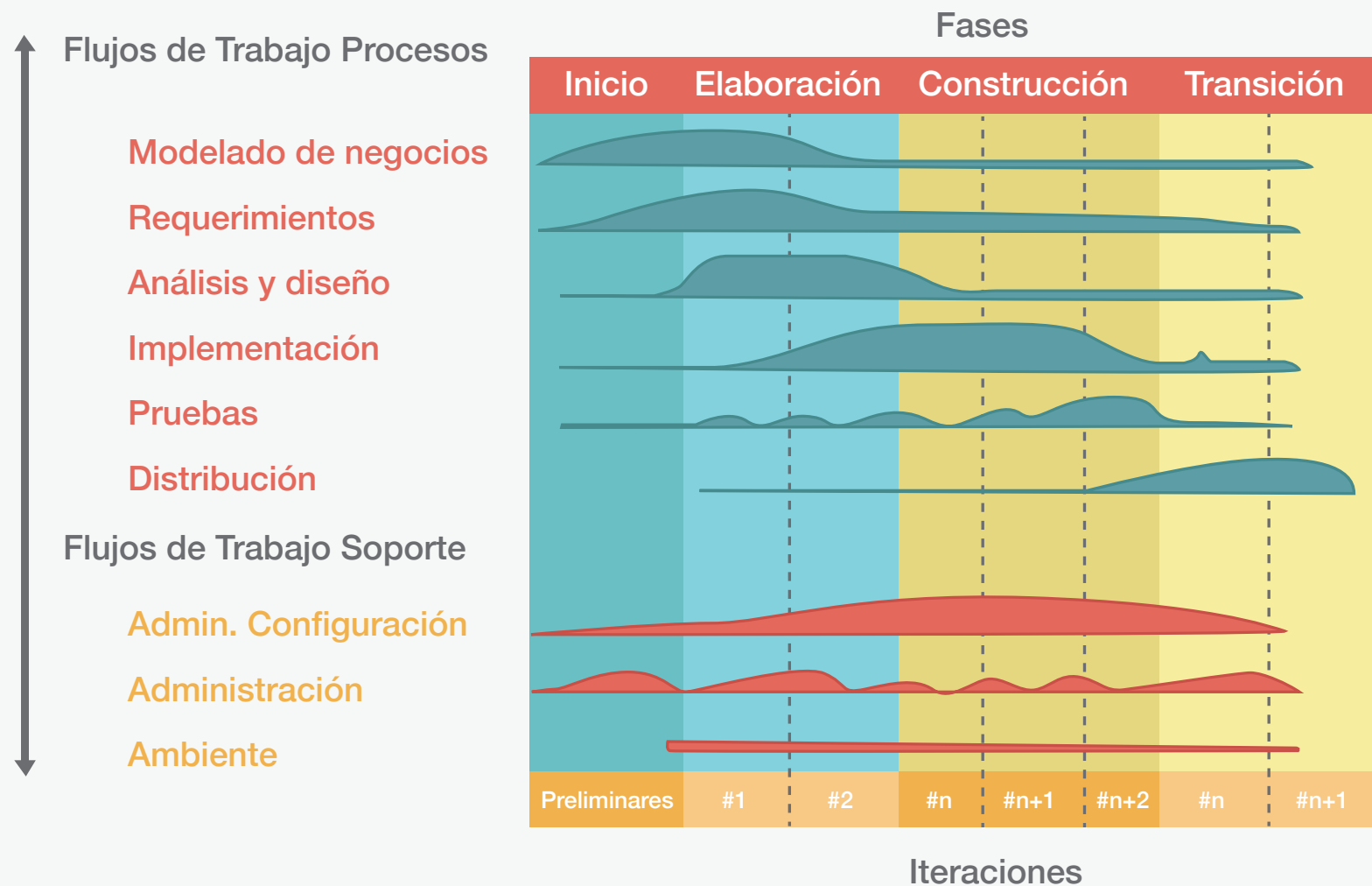
Una **disciplina** es la serie de actividades lógicas que surgen a través del proyecto.

El grupo de actividades que se encuentran dentro de una disciplina principalmente son una ayuda para entender el proyecto.

Las disciplinas son:

- **Modelado de Negocios** (Business Modeling)
- **Requerimientos** (Requirements)
- **Análisis y Diseño** (Analysis & Design)
- **Implementación** (Implementation)
- **Pruebas** (Test)
- **Distribución** (Deployment)
- **Configuración y Administración del Cambio** (Configuration & Change Mgmt)
- **Administración de Proyectos** (Project Management)
- **Ambiente** (Environment)





Ciclo de vida



• Modelado de Negocios (Business Modeling)

Los propósitos que tiene el Modelado de Negocios son:

- Entender los problemas que la organización desea solucionar e identificar mejoras potenciales.
- Medir el impacto del cambio organizacional.
- Asegurar que clientes, usuarios finales, desarrolladores y los otros participantes tengan un entendimiento compartido del problema.
- Derivar los requerimientos del sistema de software, necesarios para dar soporte a los objetivos de la organización.
- Entender como el sistema a ser desarrollado se incorpora a la organización.

• Requerimientos (Requirements)

Esta disciplina tiene el propósito de:

- Establecer y mantener un acuerdo con los clientes y los otros interesados acerca de que debe hacer el sistema.
- Proveer a los desarrolladores del sistema un mejor entendimiento de los requerimientos del sistema.
- Definir los límites del sistema.
- Proveer una base para la planeación de los contenidos técnicos de las iteraciones.
- Proveer una base para la estimación de costo y tiempo necesarios para desarrollar el sistema.
- Definir una interfaz de usuario para el sistema, enfocada en las necesidades y objetivos del usuario.

• **Análisis y Diseño** (Analysis & Design)

El propósito del Análisis y Diseño es:

- Transformar los requerimientos a diseños del sistema.
- Desarrollar una arquitectura robusta para el sistema.
- Adaptar el diseño para hacerlo corresponder con el ambiente de implementación y ajustarla para un desempeño esperado.

• **Implementación** (Implementation)

El propósito de la implementación es:

- Definir la organización del código, en términos de la implementación de los subsistemas organizados en capas.
- Probar los componentes desarrollados como unidades.
- Integrar los resultados de los implementadores individuales en un sistema ejecutable.
- La disciplina de implementación limita su alcance a como las clases individuales serán probadas. Las pruebas del sistema son descritas en futuras disciplinas.



• Pruebas (Test)

Esta disciplina actúa como un proveedor de servicios a las otras disciplinas en muchos aspectos.

Pruebas se enfoca principalmente en la evaluación y aseguramiento de la calidad del producto, desarrollado a través de las siguientes prácticas:

- Encontrar fallas de calidad en el software y documentarlas.
- Recomendar sobre la calidad percibida en el software.
- Validar y probar las suposiciones hechas durante el diseño y la especificación de requerimientos de forma concreta.
- Validar que el software trabaja como fue diseñado.
- Validar que los requerimientos son implementados apropiadamente.

• Distribución (Deployment)

Esta disciplina describe las actividades asociadas con el aseguramiento de la entrega y disponibilidad del producto de software hacia el usuario final.

Existe un énfasis en probar el software en el sitio de desarrollo, realización de pruebas beta del sistema antes de su entrega final al cliente.



• Configuración y Administración del Cambio (Configuration & Change Mgmt)

Consiste en controlar los cambios y mantener la integridad de los productos que incluye el proyecto.

- Identificar los elementos configurables.
- Restringir los cambios en los elementos configurables.
- Auditar los cambios hechos a estos elementos.
- Definir y mantener las configuraciones de estos elementos.
- Los métodos, procesos y herramientas usadas para proveer la administración y configuración del cambio pueden ser considerados como el sistema de administración de la configuración.

• Administración de Proyectos (Project Management)

El propósito de la Administración de Proyectos es:

- Proveer un marco de trabajo para administrar los proyectos intensivos de software.
- Proveer guías prácticas para la planeación, soporte, ejecución y monitoreo de proyectos.
- Proveer un marco de trabajo para la administración del riesgo.

• Ambiente (Environment)

- Se enfoca en las actividades necesarias para configurar el proceso al proyecto.
- Describe las actividades requeridas para desarrollar las líneas guías de apoyo al proyecto.
- El propósito de las actividades de ambiente es proveer a las organizaciones de desarrollo de software del ambiente necesario (herramientas y procesos) que den soporte al equipo de desarrollo.

RUP en cada una de sus fases realiza una serie de **artefactos** que sirven para comprender mejor tanto el análisis como el diseño del sistema.

Los **artefactos** son todos los elementos generados para poder desarrollar un proyecto:

Inicio



Documento Visión

Especificación de requisitos

Elaboración



Diagramas de casos de uso

Construcción



Documento Arquitectura que trabaja con las siguientes vistas:

Vista lógica

Diagrama de clases

Modelo E-R (Si el sistema así lo requiere)

Vista de implementación

Diagrama de secuencia

Diagrama de estados

Diagrama de colaboración

Vista conceptual

Modelo de dominio

Vista física

Mapa de comportamiento a nivel de hardware

4 + 1 Vistas



El **Lenguaje de Modelado Unificado** (UML:Unified Modeling Language) es un lenguaje para especificar la construcción,documentación y visualización de artefactos en específico de cada proyecto.

UML nos proporciona 3 beneficios:

Visualización

Validación

Clara comunicación



Puedes usar UML con todo el proceso a través del desarrollo del ciclo de vida de cada proyecto.

Los artefactos de UML se especifican en forma de diagramas, éstos, junto con la documentación sobre el proyecto constituyen los artefactos principales que el usuario (modelador) puede observar.

Se necesita más de un punto de vista para llegar a representar un sistema (proyecto). UML utiliza los diagramas gráficos para obtener estos distintos puntos de vista. ejem:

Diagramas de implementación

Diagramas de comportamiento o interacción

Diagramas de casos de uso

Diagramas de clases

UML no define un proceso concreto que determine las fases de desarrollo de un sistema, es un método independiente del proceso. Los procesos de desarrollo deben ser definidos dentro del contexto donde se van a implementar los sistemas (proyectos).



Los miembros involucrados en una metodología RUP son:



Conclusiones

RUP realiza un levantamiento exhaustivo de requerimientos.

Busca detectar defectos en las fases iniciales.

Intenta reducir el número de cambios tanto como sea posible.

Realiza el Análisis y Diseño, tan completo como sea posible.

Diseño genérico, intenta anticiparse a futuras necesidades.

Existe un contrato prefijado con los clientes.

El cliente interactúa con el equipo de desarrollo mediante reuniones.

El verdadero corazón de RUP se encuentra en las disciplinas no en las fases.

