UNIVERSIDAD TECNOLÓGICA DE PEREIRA

FUNDAMENTOS DE LA METODOLOGIA RUP

RATIONAL UNIFIED PROCESS

JUAN PABLO GOMEZ GALLEGO

ING JORGE GALVES

16/09/2007

SOBRE EL PROCESO RACIONAL UNIFICADO

RUP es un proceso para el desarrollo de un proyecto de un software que define claramente *quien, cómo, cuándo y qué debe hacerse en el proyecto .* Como 3 características esenciales está dirigido por los Casos de Uso: **que orientan el proyecto a la importancia para el usuario y lo que este quiere** , está centrado en la arquitectura: **que Relaciona la toma de decisiones que indican cómo tiene que ser construido el sistema y en qué orden**, y es iterativo e incremental: **donde divide el proyecto en miniproyectos donde los casos de uso y la arquitectura cumplen sus objetivos de manera más depurada**

Como filosofía RUP maneja 6 principios clave:

Adatpación del proceso

El proceso deberá adaptarse a las características propias de la organización. El tamaño del mismo, así como las regulaciones que lo condicionen, influirán en su diseño específico. Tambien se deberá tener en cuenta el alcance del proyecto.

Balancear prioridades

Los requerimientos de los diversos inversores pueden ser diferentes, contradictorios o disputarse recursos limitados. *Debe encontrarse un balance que satisfaga los deseos de todos*.

Colaboración entre equipos

El desarrollo de software no lo hace una única persona sino múltiples equipos. Debe haber una comunicación fluida para coordinar requerimientos, desarrollo, evaluaciones, planes, resultados,etc.

Demostrar valor iterativamente

Los proyectos se entregan, aunque sea de un modo interno, en **etapas iteradas**. En cada iteración se analiza la opinión de los inversores, la estabilidad y calidad del producto, y se refina la dirección del proyecto asi como tambien los riesgos involucrados

Elevar el nivel de abstracción

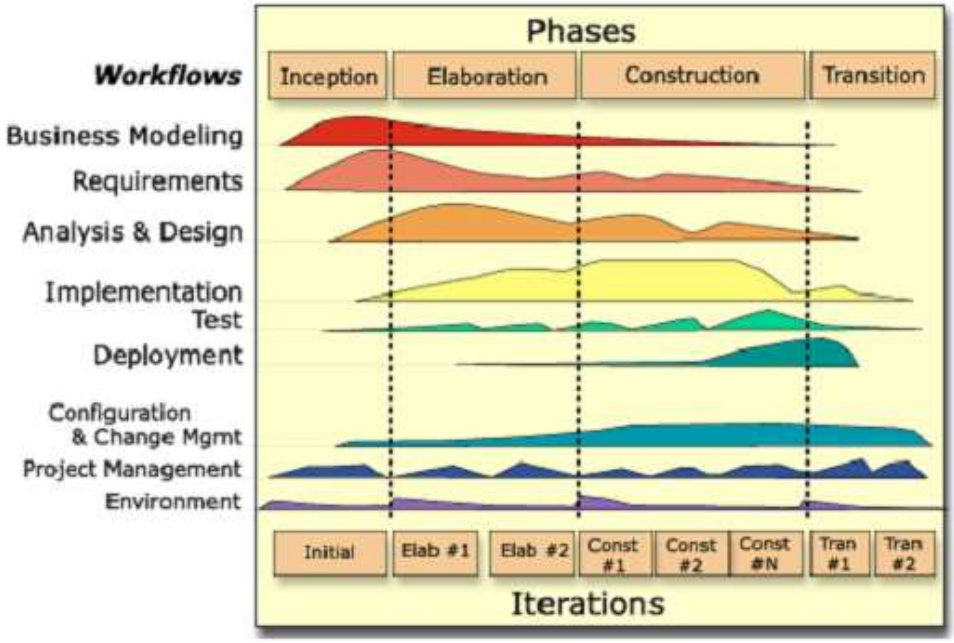
Este principio dominante motiva el uso de conceptos reutilizables tales como patrón del software, lenguajes 4GL o esquemas (frameworks) por nombrar algunos. Éstos se pueden acompañar por las representaciones visuales de la arquitectura, por ejemplo con UML.

Enfocarse en la calidad

El control de calidad no debe realizarse al final de cada iteración, sino en **todos** los aspectos de la producción

El ciclo de vida de RUP

RUP divide el proceso en 4 fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace un mayor o menor hincapié en los distintas actividades.



En las iteraciones de cada fase se hacen diferentes esfuerzos en diferentes actividades

- **Inicio**: Se hace un plan de fases, se identifican los principales casos de uso y se identifican los riesgos. Se define el alcance del proyecto
- **Elaboración**: se hace un plan de proyecto, se completan los casos de uso y se eliminan los riesgos
- **Construcción**: se concentra en la elaboración de un producto totalmente operativo y eficiente y el manual de usuario
- **Transición**: se Instala el producto en el cliente y se entrena a los usuarios. Como consecuencia de esto suelen surgir nuevos requisitos a ser analizados.

DESCRIPCIÓN DE LAS ACTIVIDADES

Dependiendo de las iteración del proceso el equipo de desarrollo puede realizar 7 tipos de actividades en este:

FASE DE INICIO

Durante la fase de inicio las iteraciones hacen ponen mayor énfasis en actividades modelado del negocio y de requisitos.

Modelado del negocio

En esta fase el equipo se familiarizará más al funcionamiento de la empresa, sobre conocer sus procesos.

- Entender la estructura y la dinámica de la organización para la cual el sistema va ser desarrollado (
- Entender el problema actual en la organización objetivo e identificar potenciales mejoras.
- Asegurar que clientes, usuarios finales y desarrolladores tengan un entendimiento común de la organización objetivo.

Requisitos

En esta línea los requisitos son el contrato que se debe cumplir, de modo que los usuarios finales tienen que comprender y aceptar los requisitos que especifiquemos.

- Establecer y mantener un acuerdo entre clientes y otros *stakeholders* sobre lo que el sistema podría hacer.
- Proveer a los desarrolladores un mejor entendimiento de los requisitos del sistema.
- Definir el ámbito del sistema.
- Proveer una base para estimar costos y tiempo de desarrollo del sistema.
- Definir una interfaz de usuarios para el sistema, enfocada a las necesidades y metas del usuario.

FASE DE ELABORACIÓN

En la fase de elaboración, las iteraciones se orientan al desarrollo de la *baseline* de la arquitectura, abarcan más los flujos de trabajo de requerimientos, modelo de negocios (refinamiento), análisis, diseño y una parte de implementación orientado a la *baseline* de la arquitectura.

Análisis y Diseño

En esta actividad se especifican los requerimientos y se describen sobre como se van a implementar en el sistemas

- Transformar los requisitos al diseño del sistema.
- Desarrollar una arquitectura para el sistema.
- Adaptar el diseño para que sea consistente con el entorno de implementación

FASE DE CONSTRUCCIÓN

Implementación

Se implementan las clases y objetos en ficheros fuente, binarios, ejecutables y demás. El resultado final es un sistema ejecutable.

- Planificar qué subsistemas deben ser implementados y en que orden deben ser integrados, formando el Plan de Integración.
- Cada implementador decide en que orden implementa los elementos del subsistema.
- Si encuentra errores de diseño, los notifica.
- Se integra el sistema siguiendo el plan.

Pruebas

Este flujo de trabajo es el encargado de evaluar la calidad del producto que estamos desarrollando, pero no para aceptar o rechazar el producto al final del proceso de desarrollo, sino que debe ir integrado en todo el ciclo de vida.

- Encontrar y documentar defectos en la calidad del software.
- Generalmente asesora sobre la calidad del software percibida.
- Provee la validación de los supuestos realizados en el diseño y especificación de requisitos por medio de demostraciones concretas.
- Verificar las funciones del producto de software según lo diseñado.
- Verificar que los requisitos tengan su apropiada implementación.

Despliegue

Esta actividad tiene como objetivo producir con éxito distribuciones del producto y distribuirlo a los usuarios. Las actividades implicadas incluyen:

- Probar el producto en su entorno de ejecución final.
- Empaquetar el software para su distribución.
- Distribuir el software.
- Instalar el software.
- Proveer asistencia y ayuda a los usuarios.
- Formar a los usuarios y al cuerpo de ventas.
- Migrar el software existente o convertir bases de datos.

DURANTE TODO EL PROYECTO

Gestión del proyecto

Se vigila el cumplimiento de los objetivos, gestión de riesgos y restricciones para desarrollar un producto que sea acorde a los requisitos de los clientes y los usuarios.

- Proveer un marco de trabajo para la gestión de proyectos de software intensivos.
- Proveer guías prácticas realizar planeación, contratar personal, ejecutar y monitorear el proyecto.
- Proveer un marco de trabajo para gestionar riesgos.

Configuración y control de cambios

El control de cambios permite mantener la integridad de todos los artefactos que se crean en el proceso, así como de mantener información del proceso evolutivo que han seguido.

Entorno

La finalidad de esta actividad es dar soporte al proyecto con las adecuadas herramientas, procesos y métodos. Brinda una especificación de las herramientas que se van a necesitar en cada momento, así como definir la instancia concreta del proceso que se va a seguir.

En concreto las responsabilidades de este flujo de trabajo incluyen:

- Selección y adquisición de herramientas
- Establecer y configurar las herramientas para que se ajusten a la organización.
- Configuración del proceso.
- Mejora del proceso.
- Servicios técnicos.

ROLES EN RUP

Analistas:

- Analista de procesos de negocio.
- Diseñador del negocio.
- Analista de sistema.
- Especificador de requisitos.

Desarrolladores:

- Arquitecto de software.
- Diseñador
- Diseñador de interfaz de usuario
- Diseñador de cápsulas.
- Diseñador de base de datos.
- Implementador.
- Integrador.

Gestores:

- Jefe de proyecto
- Jefe de control de cambios.
- Jefe de configuración.
- Jefe de pruebas
- Jefe de despliegue
- Ingeniero de procesos
- Revisor de gestión del proyecto
- Gestor de pruebas.

Apoyo:

- Documentador técnico
- Administrador de sistema
- Especialista en herramientas
- Desarrollador de cursos
- Artista gráfico

Especialista en pruebas:

- Especialista en Pruebas (*tester*)
- Analista de pruebas
- Diseñador de pruebas

Otros roles:

- *Stakeholders*.
- Revisor
- Coordinación de revisiones
- Revisor técnico
- Cualquier rol

Notas:

- Para grandes organizaciones con un números equipos de ingenieros y la comunicación entre cada equipo es crítica por lo tanto es necesario que los artefactos sean completos y bastante comprensivos

- En tanto que para pequeños proyectos no es recomendable presentarse tanto rigor en las preparaciones de los artefactos, la eficiencia del proceso depende más de las habilidades de cada trabajador