

## ***Processo Unificado de desenvolvimento de software- RUP***

### ***Considerações***

O método iterativo: criado para superar as dificuldades impostas pelo modelo cascata. Já que o modelo cascata pode ser usado com sucesso em projetos pequenos, onde o domínio do problema é bem conhecido, a solução encontrada foi dividir grandes projetos em projetos menores. Dessa maneira, alguns requisitos e alguns riscos podem ser identificados, um projeto pode ser realizado, uma implementação pode ser construída para esse projeto, validade e testada, Esse processo se repete com outras partes do sistema até que o sistema inteiro seja terminado, isso é chamado de modo iterativo.

Em cada pequena parte do sistema é feita uma iteração. A iteração segue o modelo sequencial tradicional, com identificação das necessidades, análise, projeto, implementação e testes. A cada iteração o sistema é incrementado até que o ciclo de desenvolvimento do aplicativo termine. Nesse ponto, um novo modelo de desenvolvimento pode ser iniciado

A maneira de desenvolver projetos através de várias iterações que vão incrementando o projeto até que se chegue a um objetivo é chamada de modo iterativo e incremental. Atualmente esse , paradigma de desenvolvimento é bem aceito e vem sendo utilizado por várias metodologias de desenvolvimento de software.

### ***Definições***

- O RUP é uma maneira de desenvolvimento de software que é iterativa, centrada à arquitetura e guiada por casos de uso. É descrita em vários livros e artigos. Uma das maiores fontes de informação é o próprio produto IBM RUP [1], que contém guias detalhados, exemplos e modelos cobrindo todo o ciclo de vida do software.
- O RUP é um processo de engenharia de software bem definido e bem estruturado. O RUP define claramente quem é responsável pelo que, como as coisas devem ser feitas e quando fazê-las, O RUP também provê uma estrutura bem definida para o ciclo de vida de um projeto RUP, articulando claramente os marcos essenciais e pontos de decisão.
- O RUP é também um produto de processo que oferece uma estrutura de processo customizável para a engenharia de software. O produto IBM RUP suporta a customização e autoria de processos, e uma vasta variedade de processos, ou configuração de processos, podem ser montadas nele. Essas configurações do RUP podem ser criadas para suportar equipes grandes e pequenas, e técnicas de desenvolvimento disciplinadas ou menos formais, O produto IBM RUP contém várias configurações e visões de processos prontas que guiam analistas,

desenvolvedores, testadores, gerentes da projeto, gerentes de configuração, analistas de dados, e outros membros da equipe.

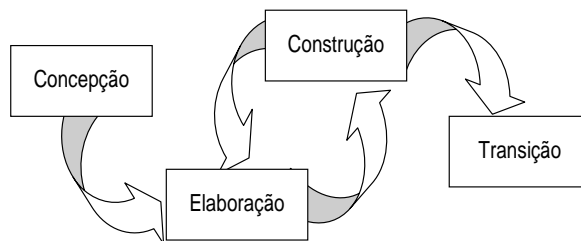
- O **Processo Unificado** (UP) de desenvolvimento de sistemas combina os ciclos iterativo e incremental para a construção de [softwares](#).
- O **processo unificado** de desenvolvimento de software é o conjunto de atividades necessárias para transformar *requisitos do usuário* em um *sistema de software*. Ele é *baseado em componentes*, o que significa o sistema ser construído a partir de componentes de software interconectados via interfaces muito bem definidas.
- *O Processo Unificado proposto pela Rational (Rational Unified Process – RUP) foi criado para apoiar o desenvolvimento orientado a objetos, fornecendo uma forma sistemática para se obter reais vantagens no uso da Linguagem de Modelagem Unificada (Unified Modeling Language – UML). De fato, ele não é exatamente um processo: é uma infraestrutura genérica de processo que pode ser especializada para uma ampla classe de sistemas de software, para diferentes áreas de aplicação, tipos de organização, níveis de competência e tamanhos de projetos.*

## ***Princípios básicos do RUP***

O RUP está fundamentado em três princípios básicos: orientação a casos de uso, arquitetura e iteração. Ele é dito dirigido a casos de uso, pois são os casos de uso que orientam todo o processo de desenvolvimento. Com base no modelo de casos de uso, são criados uma série de modelos de análise, projeto e implementação, que realizam estes casos de uso. É centrado em arquitetura, pois defende a definição de um esqueleto para a aplicação (a arquitetura), a ganhar corpo gradualmente ao longo do desenvolvimento. Finalmente, o RUP é iterativo e incremental, oferecendo uma abordagem para particionar o trabalho em porções menores ou mini-projetos. Esses três conceitos são igualmente importantes. A arquitetura provê a estrutura para guiar o desenvolvimento do sistema em iterações, enquanto os casos de uso definem as metas e conduzem o trabalho de cada iteração.

## ***Ciclo de vida do processo unificado – RUP***

O ciclo de vida adotado no RUP é tipicamente evolutivo. Contudo, uma forma de organização em fases é adotada para comportar os ciclos de desenvolvimento, permitindo uma gerência mais efetiva de projetos complexos. Ao contrário do tradicionalmente definido como fases na maioria dos modelos de ciclo de vida – planejamento, levantamento de requisitos, análise, projeto, implementação e testes, são definidas fases ortogonais a estas, a saber:



**Figura 1: fase do processo unificado**

- **Concepção:** nesta fase, é estabelecido o escopo do projeto e suas fronteiras, determinando os principais casos de uso do sistema. Esses casos de uso devem ser elaborados com a precisão necessária para se proceder estimativas de prazos e custos. As estimativas devem ser globais para o projeto como um todo e detalhadas para a fase seguinte. Assim, a ênfase nesta etapa recai sobre o planejamento e, por conseguinte, é necessário levantar requisitos do sistema e preliminarmente analisá-los. Ao término dessa fase, são examinados os objetivos do projeto para se decidir sobre a continuidade do desenvolvimento.
- **Elaboração e construção:** estas fases ocorrem dentro dos ciclos iterativos. A elaboração é constituída da análise e projeto, e a construção corresponde à implementação e aos testes. É nos ciclos iterativos propriamente ditos que acontece a análise detalhada do sistema (análise de requisitos e de domínios) e nos quais é feito o projeto do sistema usando padrões de projetos. Nos ciclos iterativos são também feitos as implementações dos códigos e os testes.
- **Transição:** Ocorre após o último ciclo iterativo. Nesta fase, o software é disponibilizado à comunidade usuária. Após o produto ter sido colocado em uso, naturalmente surgem novas considerações que vão demandar a construção de novas versões para permitir ajustes do sistema, corrigir problemas ou concluir algumas características que foram postergadas. É importante realçar que dentro de cada fase, um conjunto de iterações, envolvendo planejamento, levantamento de requisitos, análise, projeto e implementação e testes, é realizado. Contudo, de uma iteração para outra e de uma fase para a próxima, a ênfase sobre as várias atividades muda, como mostra a figura 1, em que a cor preta indica grande ênfase, enquanto a cor branca indica muito pouca ênfase. Na fase de concepção, o foco principal recai sobre o entendimento dos requisitos e a determinação do escopo do projeto (planejamento e levantamento de requisitos). Na fase de elaboração, o enfoque está

na captura e modelagem dos requisitos (levantamento de requisitos e análise), ainda que algum trabalho de projeto e implementação seja realizado para prototipar a arquitetura, evitando certos riscos técnicos. Na fase de construção, o enfoque concentra-se no projeto e na implementação, visando evoluir e recheiar o protótipo inicial, até obter o primeiro produto operacional. Finalmente, a fase de transição concentra-se nos testes, visando garantir que o sistema possui o nível adequado de qualidade. Além disso, usuários devem ser treinados, características ajustadas e elementos esquecidos adicionados.

	Levantamento de Requisitos	Análise	Projeto	Implementação	Testes
Concepção					
Elaboração					
Construção					
Transição					

**Figura 2: ênfase principal de cada uma das fases**

## ***Elementos RUP***

- Papeis: define o comportamento e as responsabilidades de um determinado indivíduo ou grupo de indivíduos trabalhando como uma equipe. Papéis não são indivíduos e nem títulos de trabalho. Um indivíduo pode assumir vários papéis. São exemplos de papéis:
  - Analista de sistema - O indivíduo que assume este papel coordena a obtenção dos requisitos e a modelagem dos casos de uso identificando funcionalidades do sistema e estabelecendo limites do sistema;
  - Projetista - Esse indivíduo define responsabilidades, operações, atributos, relacionamentos de uma ou mais classes e determina como elas devem ser ajustadas para serem implementadas no ambiente;
  - Projetista de testes - Responsável pelo planejamento, projeto, implantação e avaliação de testes, incluindo a geração de plano e modelo de teste,

implementando procedimentos de testes e avaliando a abrangência dos testes, resultados e a efetividade.

- Uma atividade é uma unidade de trabalho que um indivíduo executa quando está exercendo um determinado papel e produz um resultado importante para o contexto do projeto. Cada atividade pode ser dividida em passos. São exemplos de atividades: planejar uma iteração, encontrar casos de uso e atores, rever o projeto, executar um teste de performance.
- Um artefato é um pedaço de informação que é produzido, modificado ou utilizado em um processo. Os artefatos são os produtos de um projeto, são utilizados como entradas de atividades e são produzidos como saída. São exemplos:
  - Um modelo, como um modelo de caso de uso
  - Um elemento de um modelo, como uma classe
  - Um documento
  - Um código fonte
  - Executáveis
- Fluxos de trabalhos são seqüências de atividades que são executadas para a produção de um resultado valioso para o projeto. Podem ser representados por diagramas de seqüências, colaboração e atividades da linguagem UML
- Disciplina é um conjunto de atividades relacionadas que fazem parte de um contexto comum em um projeto. Proporcionam um melhor entendimento do projeto sob o ponto de vista tradicional do modelo cascata. O RUP possui nove disciplinas divididas em disciplinas de processo (modelagem de negócios, requisitos, análise e design, implementação, teste e distribuição) e suporte (configuração e gerenciamento de mudanças, gerenciamento de projeto e ambiente).

