

# UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

## *Campus* CAMPO MOURÃO

### Resenha Capítulo 2 e 9 - APOO

#### Processo de Desenvolvimento e Casos de Uso

Martin Fowler

Estudante: Reginaldo Gregório de Souza Neto

RA: 2252813

O Processo de Desenvolvimento das metodologias de APOO foram desenvolvidos através da mistura da linguagem gráfica de modelagem e o processo de descrição de como o software deveria funcionar. Após a criação da UML, foi-se decidido que ela seria restringida à uma linguagem de modelagem somente depois que o processo fosse decidido.

Processo Iterativo e em Cascata, são dois modelos de divisão e organização do trabalho dentro de uma empresa. Sendo que o estilo em cascata subdivide um projeto com base nas atividades (Análise -> Projeto -> Codificação -> Teste). Já o modelo iterativo por sua vez se trata da divisão através de funcionalidades, ou seja, se trata de realizar TODOS os passos citados anteriormente no modelo cascata em um período de tempo menor. Entretanto, esses passos seriam concretizados para uma parcela menor do projeto de cada vez. Para a maioria dos especialistas em APOO o modelo cascata não é recomendável, pois dificulta a avaliação do andamento do projeto, se há de fato algum progresso.

Por conta disso, Fowler diz: *“Somente por esse motivo, recomendo veementemente que os projetos não utilizem uma estratégia em cascata pura. Você deve usar pelo menos a entrega por etapas, se não usar uma técnica iterativa mais pura.”*

Uma estratégia eficiente para o lançamento de um projeto é a utilização do estilo iterado com prazos fixos de entregas, onde caso não seja possível a entrega completa, que se retire algumas funcionalidades, porém não se altere o tempo de entrega.

Testes de regressão automatizados ajudam, permitindo que você detecte rapidamente quaisquer defeitos que possam ter sido introduzidos na alteração de algo. Já o refactoring é uma técnica de alteração de um software pré-existente.

Planejamento Preditivo e Adaptativo, a estratégia preditiva procura fazer o trabalho antecipadamente no projeto, a fim de gerar um maior entendimento do que precisa ser feito posteriormente. Já a adaptativa como o próprio nome sugere, condiz com a maneira de tratar os problemas conforme eles vão surgindo.

Os Casos de Uso são uma técnica para representar as funcionalidades do software através de textos ou diagramas para facilitar o entendimento do mesmo. Os casos de uso são frequentemente utilizados para auxiliar no desenvolvimento do software tendo em vista que através deles é possível representar o desejo do cliente de uma maneira simples e prática.

Deste modo podemos definir quais serão as interações dos usuários (atores) com o sistema. Se o sistema for criado para trabalhar em conjunto com outra máquina, essa, pode ser considerada um ator também.

Na descrição dos casos de uso via texto, cada passo é um caso de uso da interação entre um ator e o sistema. Exemplo:

**Compra de um Produto**  
Nível do Objetivo: Nível do Mar  
Cenário Principal de Sucesso:  

1. O cliente navega pelo catálogo e seleciona itens para comprar
2. O cliente vai para o caixa
3. O cliente preenche o formulário da remessa (endereço de entrega; opção de entrega imediata ou em três dias)
4. O sistema apresenta a informação completa do faturamento, incluindo a remessa
5. O cliente preenche a informação de cartão de crédito
6. O sistema autoriza a compra
7. O sistema confirma imediatamente a venda
8. O sistema envia uma confirmação para o cliente por *e-mail*

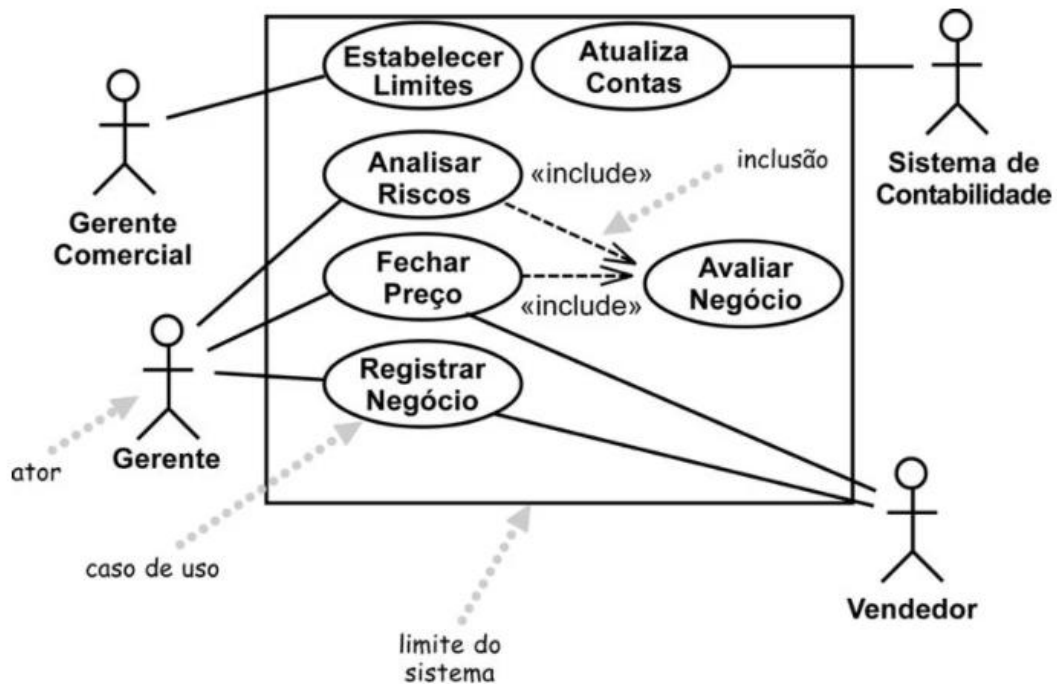
  
Extensões:  
3a: Cliente regular  

- .1: O sistema mostra a informação atual da remessa, a informação de preço e a informação de cobrança
- .2: O cliente pode aceitar ou escrever por cima desses padrões, retornando ao CPS, no passo 6

  
6a. O sistema falha na autorização da compra a crédito  

- .1: O cliente pode inserir novamente a informação do cartão de crédito ou cancelar

Já no diagrama de casos de uso, a representação dos atores é feita por bonecos-palito e os casos de uso são inseridos em elipses. Exemplo:



Devido à sua grande abstração, os casos de uso normalmente são divididos em duas partes, sendo uma o caso de uso de sistema que é uma interação com o *software*, enquanto um caso de uso de negócio examina como a aplicação responde ao cliente ou a um evento.

As funcionalidades podem ser explicitadas de maneira mais prática utilizando-se de “histórias” que narram como será uma interação do usuário com o sistema, para que o desenvolvedor possa prever as situações que o sistemas poderá ser submetido.