

Engenharia de Requisitos

Pablo Dall'Oglio

Definição

“A Engenharia de Requisitos é um processo que envolve todas as atividades exigidas para criar e manter o documento de requisitos de um sistema.”

Roteiro

- Engenharia de Requisitos;
- Elicitação;
- Gerência;
- Documentação;
- Validação;
- Casos de Uso;

Introdução

- A Engenharia de Requisitos é uma área da Engenharia de Software que busca **propor métodos, técnicas e ferramentas** para **facilitar a definição de requisitos** de software;
- O **papel** da Engenharia de Requisitos é o de **descobrir o propósito de um sistema** por meio da identificação e documentação das **necessidades dos clientes**;
- O processo vai desde o **reconhecimento de um problema** a ser resolvido até sua **especificação detalhada**;
- Suas técnicas visam assegurar um **conjunto consistente** e significativo de **requisitos antes da construção do sistema**, para minimizar mudanças posteriores;
- Se mal realizada, **pode comprometer todo o projeto**.

Engenharia de Requisitos

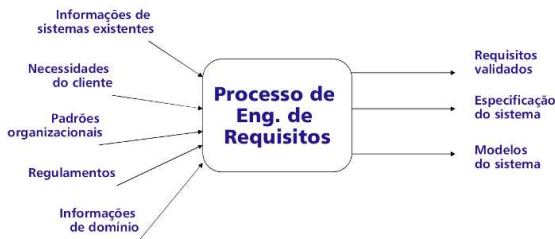
Atividades do processo

As **cinco atividades** do processo de Engenharia de Requisitos são:

- 1) Elicitação (obtenção);
- 2) Análise e negociação;
- 3) Documentação (especificação);
- 4) Validação;
- 5) Gerenciamento.

Entradas e saídas do processo

- A seguir, as entradas e saídas do processo:



Descrição das atividades

- Elicitação:** **Descoberta dos requisitos** através de consultas aos usuários;
- Análise e negociação:** Os requisitos são analisados em detalhes e as **divergências são negociadas** com os usuários;
- Documentação:** Os **requisitos são documentados** de forma que todos os envolvidos os compreendam;
- Validação:** Os **requisitos são validados** afim de evitar-se problemas futuros. Verifica-se nesta etapa se os requisitos estão completos e consistentes;
- Gerenciamento:** Em paralelo com as atividades anteriores existe o gerenciamento dos requisitos o qual tem como objetivo **gerenciar as mudanças**.

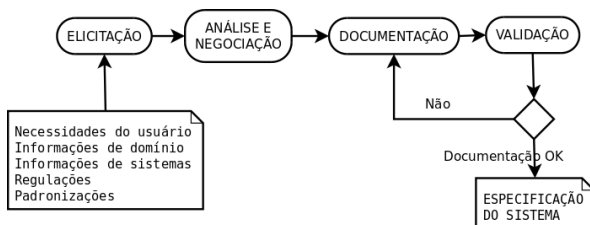
Processo de Engenharia de Requisitos

- O **processo** de Engenharia de Requisitos **pode variar muito**, desde um processo completamente **não estruturado até um processo sistemático**;
- Cada **empresa deve definir um processo genérico** de Engenharia de Requisitos o qual **deve ser adaptado para a realidade de cada projeto**;
- O processo é influenciado por:
 - Maturidade técnica;
 - Disciplina;
 - Cultura organizacional;
 - Domínio da aplicação.

Elicitação de Requisitos

Processo de Engenharia de Requisitos

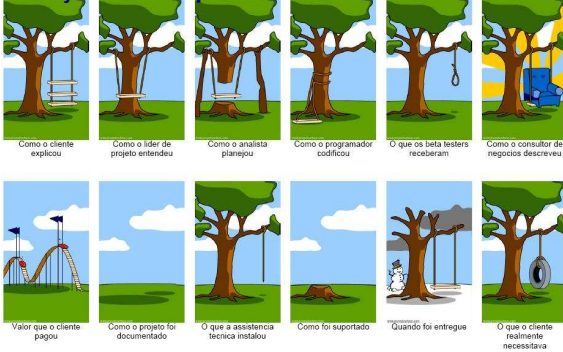
- A seguir, o fluxo de atividades de **um processo de Engenharia de Requisitos**;



Elicitação de Requisitos

- Elicitação de requisitos é um **conjunto de atividades** que envolvem a **descoberta dos requisitos de um sistema**;
- Também consiste em **identificar os usuários do sistema**, as **tarefas que desempenham** e como eles **gostariam de desempenhar** estas mesmas tarefas;
- Envolve a descoberta de **requisitos que o software deve possuir** para **satisfazer as diversas necessidades do cliente**;
- Analistas trabalham junto com clientes** e usuários finais para **entender o problema** a ser resolvido através do **detalhamento das funcionalidades** e restrições do futuro sistema;
- Não se trata apenas de questionar os clientes e usuários sobre o que eles precisam, mas sim de uma **análise da empresa, do domínio da aplicação e dos processos** de negócio envolvidos.

Elicitação de Requisitos



Entrevistas

- São utilizadas para **obter o conhecimento** por meio de **perguntas realizadas aos usuários** especialistas;
- Permite aos **analistas entenderem os processos** da organização, bem como **perceber as expectativas** dos usuários;
- Pode ser aplicado informalmente ou por **meio de questionários com perguntas pré-definidas**;
- Trazem um **conjunto rico em informações**, mas demandam grandes esforços de análise, por que **trazem visões conflitantes**;
- Como vantagem, **permitem um contato imediato com o usuário**;
- Como desvantagem, **não podem ser utilizadas como único meio** de obter os requisitos, pois os usuários utilizam termos muito específicos para explicar suas tarefas, e **acham desnecessário ou não conseguem explicar** o que é óbvio para o seu trabalho.



Dificuldades na Elicitação de Requisitos

- São **dificuldades** na Elicitação de Requisitos:
 - As **informações sobre o domínio da aplicação estão espalhadas** em textos, manuais, na cabeça das pessoas que trabalham na área, etc;
 - As **pessoas que entendem** do problema a ser resolvido **estão muito ocupadas** tentando resolvê-los e não tem tempo ou não desejam ajudar o analista;
 - Diferentes **interesses e políticas** internas;
 - Os **clientes geralmente não sabem** o que precisam ou sabem mas não conseguem explicar;
 - Mudanças no negócio** durante ou após a elicitación;
 - Mudanças no pessoal** envolvido durante o processo.

Leitura de documentos

- Como vantagem, tem-se o **fácil acesso e o volume de informações**;
- Como desvantagem, tem-se a **dispersão das informações**.



Questionários

- Podem ser utilizados quando **existe o conhecimento sobre um problema** e vários usuários envolvidos;
- Permite ter uma **visão estatística** de como alguns problemas são percebidos pelos usuários;
- Como vantagem, **permite padronização das perguntas e estatísticas** com as respostas;
- Como desvantagem, **limita o conjunto de respostas** e tem pouca interação.

Técnicas de Elicitação de Requisitos

- Várias técnicas podem ser utilizadas** na Elicitação de Requisitos:
 - Entrevistas;
 - Leitura de documentos;
 - Questionários;
 - Observação;
 - Reutilização de requisitos;
 - Cenários;
 - Casos de uso.

Observação (Etnografia)

- As pessoas geralmente acham **difícil descrever o que elas fazem** pois isto é muito natural para elas;
- As vezes, a **melhor forma é observá-las** no trabalho. A etnografia consiste em observar as pessoas em seu trabalho e **construir uma imagem de como o trabalho é realizado**;
- Os processos** reais de trabalho **geralmente diferem** daqueles **processos formais** descritos;
- Como vantagem, **permite uma visão mais completa** do processo. Além disso, as **pessoas preferem mostrar do que explicar** como fazem suas tarefas;
- Como desvantagem, **tem-se o elevado tempo gasto**, baixa sistematização do processo e **constrangimento** na observação;
- É uma **técnica complementar** e não deve ser utilizada sozinha.



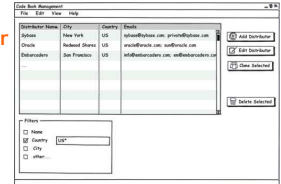
Reuso de requisitos

- Considera **requisitos que foram desenvolvidos para um sistema e usá-los em sistemas diferentes**;
- O reuso de requisitos **economiza tempo e esforço**, pois requisitos reutilizados **já foram analisados e validados** em outros sistemas;
- Atualmente o reuso de requisitos **é um processo informal**;
- O reuso **leva à uma maior consistência** entre aplicações;
- A capacidade de se aproveitar análises anteriores que **diferencia um analista experiente** de um inexperiente;
- Como vantagem, tem-se **maior produtividade e qualidade**, uma vez que os requisitos já foram validados;
- Como desvantagem, tem-se uma **dificuldade em se reutilizar um requisito sem que seja necessário sua modificação**.



Prototipação

- Um **protótipo** é uma **versão inicial de um sistema**;
- Seu objetivo é a **validação de requisitos ainda em estágios iniciais** de desenvolvimento;
- O protótipo de um sistema **pode ser descartado ou evoluir para uma versão final** do sistema;
- Os protótipos **minimizam a distância na comunicação entre cliente e analista** e constituem uma **forma concreta de mostrar para o cliente** os conceitos absorvidos.



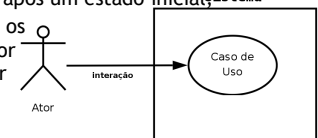
Cenários

- Um **cenário** é **uma história que explica como um sistema pode ser utilizado**;
- Um cenário **representa uma sessão** que descreve a interação entre o usuário e o sistema;
- Um cenário contém um **fluxo normal de eventos**, exceções, o estado inicial e final do sistema;
- Exemplo:**
 - 1) Aluno informa número de matrícula;
 - 2) Aluno seleciona o curso;
 - 3) Aluno seleciona as turmas a cursar;
 - 4) Aluno seleciona os horários disponíveis;
 - 5) O sistema salva a matrícula.



Casos de uso

- Técnica para obtenção de requisitos** criada em 1993 (Jacobson);
- Casos de Uso **descrevem as interações entre o sistema e seus atores**, trazendo à tona suas **relações e os diferentes papéis** desempenhados pelos usuários no sistema;
- Cenários provêm **descrição de um determinado conjunto de atividades** entre o sistema e o usuário, **simulando o fluxo de interações** entre os mesmos após um estado inicial; **sistema**
- No diagrama de casos de uso os agentes são representados por bonecos e cada interação por uma elipse com um nome.



Brainstorming

- É uma técnica que **envolve reunião em grupo**, e é **baseada na geração de idéias**;
- Envolve a **reunião de especialistas de negócio** e de sistema tendo **em vista a criação de idéias** para resolução de problemas;
- Envolve alguns **princípios como a não-crítica e o não-julgamento** das idéias geradas;
- A técnica de *Brainstorming* pode ser **aplicada no início da etapa de elicitação** de requisitos quando **o universo de requisitos conhecidos é pequeno**, sendo necessário gerar novas idéias;
- As **idéias são geradas de forma rápida**, coletadas e então discutidas e avaliadas pelo grande grupo.



Análise e Negociação

Análise e Negociação

- São atividades que tem como **objetivo descobrir problemas nos requisitos** e **obter um consenso** na solução junto aos clientes;
- Na análise, os **requisitos são analisados** para **detectar inconsistências** e identificar requisitos faltantes;
- Na negociação ocorre a **resolução de conflitos** e **priorização**;
- A análise é feita após a primeira versão do documento de requisitos ficar pronta e **envolve a revisão de todos os requisitos** com o objetivo de detectar problemas.
- Nesta etapa, os **usuários podem participar** desde que estes tenham sido treinados em técnicas de modelagem;
- Ocorrem em forma de espiral junto com a elicitação. Os **requisitos vão sendo descobertos, analisados e negociados**.



Documentação

Análise e Negociação

- **Etapas da Análise:**
 - **Análise de necessidade:** a real necessidade dos requisitos propostos é avaliada;
 - **Análise de consistência e completeza:** os requisitos são confrontados para evitar-se contradições e a omissões;
 - **Análise de viabilidade:** a viabilidade dos requisitos é verificada em relação a prazo e orçamento;
- **Etapas da Negociação:**
 - **Discussão:** os requisitos com algum tipo de problema são discutidos com os envolvidos;
 - **Priorização:** os requisitos são priorizados;
 - **Aceitação:** soluções para os requisitos problemáticos são acordadas.

Documentação

- Na fase de **documentação**, é realizada a **documentação dos requisitos**, que utilizada para comunicar os requisitos do sistema entre cliente e desenvolvedor;
- A forma pela qual os requisitos de um sistema são documentados **é de extrema importância**, para **garantir que os mesmos possam ser lidos e validados**;
- A **Especificação de Requisitos do Software** é a **declaração formal dos requisitos** do sistema e pode conter apenas os requisitos de usuário ou os de usuário e os de sistema;
- **É utilizada pelo cliente** (para validar), pelo **engenheiro de software** (para entender o que deve ser desenvolvido), pelo **engenheiro de testes** (para planejar os testes), dentre outros.



Análise e Negociação

- **Pode ser elaborado um checklist** para avaliar cada requisito:
 - ☒ Descrição incompleta;
 - ☒ Descrição ambígua;
 - ☒ Requisito desnecessário;
 - ☒ Não conformidade com os objetivos do negócio;
 - ☒ Não é viável tecnicamente;
 - ☒ O prazo não permite;
 - ☒ O custo não permite;
 - ☒ Não realista.



Documentação

- Padrão sugerido:
 - **Prefácio:** **Público-alvo**, histórico de versões;
 - **Sumário:** **Partes do documento** e numeração de páginas;
 - **Introdução:** **Funções** do sistema, **propósito** do documento, **escopo** do produto, referências, visão geral;
 - **Requisitos do usuário:** requisitos **do usuário** (RF e RNF);
 - **Requisitos do sistema:** requisitos **do sistema** (RF e RNF);
 - **Arquitetura do sistema:** **visão geral da arquitetura** e distribuição de módulos;
 - **Evolução do sistema:** **Mudanças previstas** devidas a necessidade de hardware, necessidades dos usuários, etc.

Validação

Revisão

- A revisão é a **técnica mais usada para validar** os requisitos;
- É realizada por um **grupo de pessoas que se reúnem** para discutir os problemas encontrados e definir soluções;
- **Etapas do processo:**
 - 1) **Planejamento:** Definição de datas, local e pessoas envolvidas;
 - 2) **Distribuição:** Documento é encaminhado para revisores;
 - 3) **Revisão:** Cada revisor revisa o documento de SRS;
 - 4) **Reunião:** Cada revisor expõe problemas encontrados;
 - 5) **Encaminhamento:** O coordenador encaminha as ações de correção;
 - 6) **Revisão:** O documento de SRS é revisado e corrigido. Um requisito pode ser: reescrito, complementado, corrigido ou excluído.

Validação

- A **validação de requisitos** tem como objetivo validar **consistência, completeza e precisão** dos requisitos a partir da documentação (SRS), a fim de descobrir problemas, omissões e ambiguidades;
- Na fase de validação, **analistas e usuários devem validar os requisitos e avaliar o protótipo**, o que muitas vezes leva à descoberta de novos requisitos;
- Este é um processo iterativo que **se repete até que todos os requisitos sejam validados** ou não existam mais requisitos a descobrir;
- Nesta fase é **verificada a consistência, completude e precisão** dos requisitos, para garantir que **não exista ambiguidade, inconsistências**, erros e omissões nos requisitos.

Revisão

- Para **facilitar a revisão**, um **checklist pode ser elaborado**:
 - Cada requisito possui **identificação única** ?
 - Todos os termos específicos **estão no glossário** ?
 - **É possível entender** os requisitos individualmente ?
 - Há algum **termo utilizado de diferentes** formas ?
 - Uma **função aparece em mais de um requisito** ?
 - Existe alguma **contradição** entre os requisitos ?
 - As **relações** entre os requisitos **estão claras** ?

Validação x Análise

- A **análise e a validação** de requisitos são **atividades próximas, mas diferentes**:
 - A **análise** avalia os **requisitos ainda incompletos** e não aprovados pelos clientes;
 - A **validação** inicia **somente após os requisitos estarem completos** e aprovados pelos clientes.

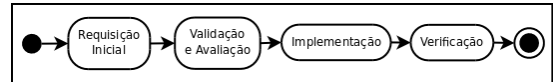
Gerenciamento

Gerenciamento

- Consiste em **gerenciar as mudanças nos requisitos** e garantir que elas **ocorram de uma maneira controlável**;
- Durante o ciclo de vida de um projeto, é necessário gerenciar as mudanças de requisitos;
- Os **requisitos evoluem ao longo do tempo** e sua própria definição **gera um feedback contínuo**, gerando novos requisitos e modificando os existentes.
- Dependências e **relações entre os requisitos** são controlados via **matriz de rastreabilidade**, que permite ao gerente conhecer **quais outros requisitos serão afetados** por uma mudança;
- Utiliza-se **versionamento de requisitos**, para que se mantenha o registro de **quem solicitou** a alteração do requisito, **quais foram** as alterações e **qual foi o impacto** decorrente.

Mudança de requisitos

- São fases da mudança de requisitos:
 - **Requisição inicial**: um membro da equipe **submete uma proposta de mudança**. O problema é identificado, analisado e descrito em termos da mudança proposta;
 - **Validação e Avaliação**: a **mudança é validada** em termos de **impactos** em cronograma e avaliada e revisada;
 - **Implementação**: a mudança aceita e **aprovada é implementada** e passa a ser parte integrante do sistema;
 - **Verificação**: é verificado **se a mudança foi implementada corretamente**.



Gerenciamento de Requisitos

- Para um efetivo gerenciamento de requisitos, cada requisito deve possuir algum tipo de **identificação única** (RF001);
- Quanto ao local, uma alternativa é armazenar todos os **requisitos em um único documento**;
- Entretanto, **tem-se algumas desvantagens**:
 - Dificuldades em **manter informações de dependência**;
 - A **pesquisa é limitada** às funções do processador de texto;
 - Dificuldade em **manter versionamento** dos requisitos;
 - Dificuldade na **navegação entre requisitos** relacionados.



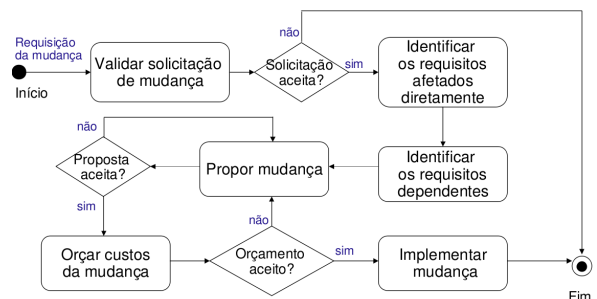
Análise de impactos

- A análise de impacto é a atividade que **identifica o que deve ser alterado** para que se **implemente determinada mudança** no sistema, identificando suas potenciais consequências;
- A análise de impacto **identifica as entidades que são possivelmente afetadas** por uma mudança proposta no sistema;
- Dentre **algumas formas** de realizar a análise de impacto, pode-se citar: a utilização de **listagens cruzadas**, relacionamentos de **rastreabilidade**, **consulta à especificação**, dentre outros;
- A análise de impacto é **tida como de sucesso** quando consegue **identificar corretamente o conjunto das partes afetadas** por uma mudança no sistema e é **tida como falha se os testes revelarem que outras partes** do sistema também necessitam ser alteradas.

Mudança de requisitos

- Alguns **motivos para a mudança de requisitos** são:
 - Mudanças no **negócio**;
 - Mudanças de **legislação**;
 - Mudanças no **processo**;
 - Evolução no **entendimento**;
 - Problemas ou **restrições técnicas**;
 - Restrições de **prazo e de orçamento**;
 - Mudanças nas **prioridades** dos cliente;
 - Erros, **conflitos** e inconsistências;

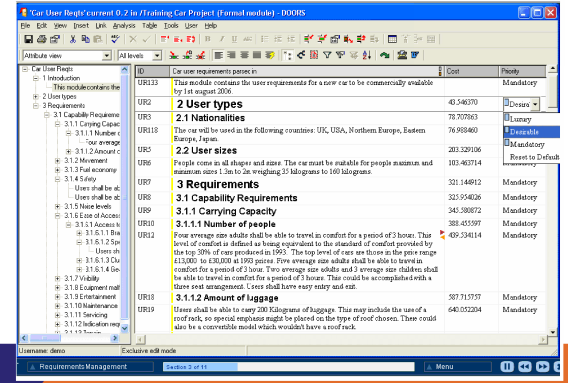
Processo de Gerência da Mudança



Rastreabilidade

- A rastreabilidade de requisitos consiste em **ligações entre as informações produzidas no desenvolvimento** de software;
- Estas ligações são **essenciais no desenvolvimento de sistemas**, por causa do grande o volume de informações produzidas;
- Quando uma **mudança ocorre** é necessário **estimar o custo e o escopo da mudança** e a rastreabilidade é chave para isto;
- Tradicionalmente, esta análise de impacto **tem sido conduzida de forma intuitiva** pelos profissionais de software, por meio da **análise de código-fonte e da documentação**;
- Esta abordagem pode ser **suficiente para projetos pequenos**, mas **não para projetos complexos**;
- Além disso, mesmo profissionais **experientes realizam estimativas erradas**;

Ferramentas



Rastreabilidade

- São **vantagens da rastreabilidade**:
 - Estimativas de custos** e prazos quando da inserção de uma nova funcionalidade;
 - Verificação de **alocação entre requisitos** e sua implementação;
 - Identificação das **origens de requisitos conflitantes**;
 - Verificação de requisitos para os quais **não foram previstos testes**;
 - Validação do sistema, verificando se o mesmo **atende ao conjunto de requisitos** proposto;
 - Identificação de **riscos que possam impactar** os requisitos;
 - Identificação de **ligações entre código e documentos** de análise.

Casos de Uso

Rastreabilidade

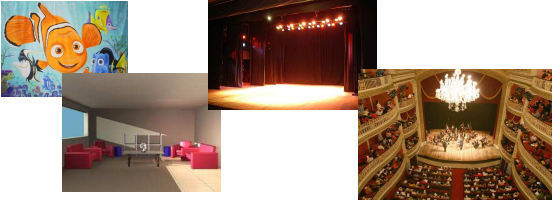
Rastreabilidade Casos de Uso -> Requisitos									
	RF1 Cadastro de Usuários	RF2 Níveis hierárquicos	RF3 Cadastro de Clientes	RF4 Controle de Notas Fiscais	RF5 Cadastro de Produtos	RF6 Cadastro de Condições de Pagamento	RF7 Cálculo de Comissão	RF14 Emissão relatórios de totais de vendas	RF15 Emissão boletim
UC1 Cadastro usuário	✖	✖	✖	✖	✖	✖	✖	✖	✖
UC2 Cadastro clientes	✖	✖	✖	✖	✖	✖	✖	✖	✖
UC3 Emissão Notas Fiscais	✖	✖	✖	✖	✖	✖	✖	✖	✖
UC4 Cálculo comissão	✖	✖	✖	✖	✖	✖	✖	✖	✖
UC5 Cadastro produtos	✖	✖	✖	✖	✖	✖	✖	✖	✖
UC7 Consultas e relatórios	✖	✖	✖	✖	✖	✖	✖	✖	✖

Introdução

- Casos de uso são utilizados para **captar e transmitir** os requisitos **funcionais** de um software por meio da descrição das **interações** que ocorrem entre o **sistema** e seus **usuários**;
- Os casos de uso possibilitam apresentar os principais **papéis** que utilizam o sistema e quais **tarefas** estes desempenham;
- A técnica de casos de uso está fortemente baseada na ideia de **cenários**;
- Um cenário trata-se de uma **narrativa** que descreve em **etapas** a sequência de **interações** entre um **usuário** e o **sistema**.

Um cenário

- Ex: O **leitor** vai até a **biblioteca**, onde seleciona os **livros** de seu interesse. Quando este finalizar, passa no balcão de empréstimos onde se **identifica**, é **autorizado** pelo **atendente**, que registra os **empréstimos** no sistema e lhe informa a data de devolução.



Casos de uso

- Um caso de uso deve ter **um nome único**, geralmente formado por um verbo e um objeto;
- Exemplo: **Realizar empréstimo**, Registrar devolução, Processar compra, Autenticar usuário;
- A escrita de um caso de uso geralmente inicia pelo **caso de sucesso**, ou seja, o caminho pelo qual a **seqüência de atividades** objetivadas pelo ator atinge o **êxito**;
- A partir do caso de sucesso, **são criadas extensões**, que podem indicar **falhas** (erros) ou caminhos **alternativos** de execução;
- Uma boa forma de identificar extensões é imaginar de que **outras maneiras** o processo poderia ser conduzido ou mesmo pensar quais **etapas podem falhar**.

Cenários e Casos de uso

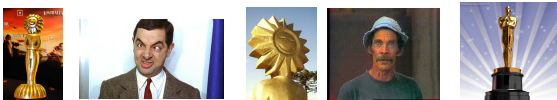
- Um cenário trata-se de uma descrição **subjettiva**. Ele é apenas uma das diversas **alternativas** que podem vir a ser realidades;
- Algumas etapas descritas em um cenário podem falhar e, neste caso, o cenário **não nos diz como o sistema deve se comportar**; Por exemplo, o leitor pode ter a senha rejeitada e neste caso, teríamos um novo cenário;
- Um caso de uso, de outra maneira, é uma **descrição** mais **completa e detalhada** de um cenário ou de vários cenários inter-relacionados;
- Dizemos que um caso de uso é um **conjunto de cenários** comum mesmo **objetivo em comum**.

Casos de uso

- Os casos de uso são utilizados principalmente para representar **os requisitos funcionais** que serão implementados pelo sistema;
- Mas também podem ser utilizados para servirem de **guias para os testes** que serão realizados posteriormente;
- A descrição textual de um caso de uso é uma fonte muito rica para a **execução de testes**, observando principalmente as **extensões**;
- Para Fowler, o **verdadeiro valor** de um caso de uso está em seu conteúdo, **sua descrição**, e não em seu diagrama. O **diagrama** é apenas uma ferramenta para facilitar sua **visualização**;
- Além disso, os casos de uso podem ser utilizados para a montagem dos **perfis de usuários** e **grupos** que utilizarão o sistema após implantado.

Atores (papéis)

- Os casos de uso são realizados por **atores**, usuários do **sistema**;
- Os atores são nomeados por **papéis** que desempenham no **sistema** e este papel tem relacionamento com a atividade desempenhada por ele;
- É comum encontrarmos atores chamados: atendente, cliente, administrador, etc. Um ator **não precisa ser** necessariamente **uma pessoa**, mas também um **outro sistema** da organização;
- É importante notar que um ator **nunca faz parte do sistema**, ele é externo ao sistema.



Exemplo de Casos de uso

- Caso de uso:** Empréstimo
- Fluxo principal:**
 1. Leitor seleciona exemplares para retirar por empréstimo;
 2. Leitor se direciona ao balcão de empréstimos;
 3. Leitor se identifica por meio de documento próprio fornecido pela biblioteca;
 4. Atendente realiza autenticação do leitor no sistema;
 5. Atendente realiza leitura dos exemplares pelo leitor ótico;
 6. Sistema registra retirada;
 7. Sistema emite comprovante;

Exemplo de Casos de uso

- **Extensões:**
 - 4a. A autenticação do leitor falha
 - 1.0 mesmo pode tentar digitar a senha novamente ou cancelar o empréstimo;
 - 2.0 leitor pode solicitar uma nova senha e reiniciar o processo.
 - 5a. Algum exemplar já esteja reservado para outro leitor
 - 1. O livro deverá ser devolvido ao acervo.

Diagrama de Casos de uso

- Um Caso de uso é representado por uma **elipse**;
- Um Caso de uso é geralmente escrito no **infinitivo + objeto**;
- Um Caso de uso **não é um módulo** do sistema (substantivo);
- Para descobrir os Casos de uso, **pergunte aos atores** o que eles irão fazer;
- O ator é algo ou alguém **externo ao sistema**;
- Um ator **interage** com o **sistema**;
- Para descobrir os atores: **“Quem interage com o sistema?”**
- Atores são representados por um **StickMan**;

Detalhes de um caso de uso

- Um caso de uso poderá estar **relacionado** à **outros** casos de uso.
- O diagrama de casos de uso apresenta uma **notação formal** para isto. Entretanto, **textualmente** não há uma forma **padronizada** para esta indicação;
- Apesar disto, muitas ferramentas utilizam o estilo **sublinhado** para **referenciar** um caso de uso externo.
- Além do fluxo de atividades principal e das extensões, um caso de uso pode ter **pré-condições**, que descrevem algumas situações que devem ser **garantidas antes** da execução do fluxo principal.
- Já uma **pós-condição** ou garantia descreve alguma situação ou estado que o sistema deve atingir ou **garantir ao final** da execução do fluxo principal no caso deste ser executado com êxito.

Diagrama de Casos de uso

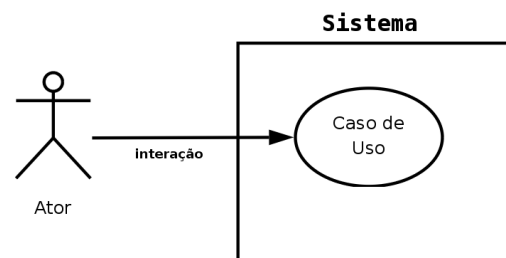


Diagrama de Casos de uso

- A UML apresenta o **diagrama de casos de uso** como uma notação gráfica que **exibe os atores e os casos de uso** que estes desempenham no sistema;
- É uma técnica que permite uma visão de alto nível sobre os papéis e suas ações no sistema;
- O diagrama de casos de uso não permite apenas apresentar as relações entre os atores e os casos de uso, mas também **relações entre os casos de uso** e permite conhecer os limites do sistema.

Diagrama de Casos de uso

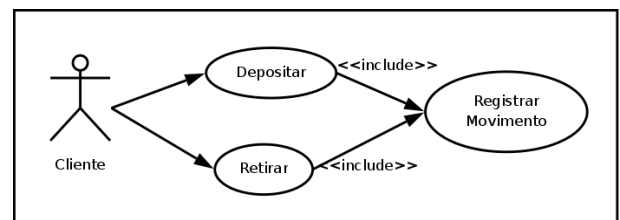
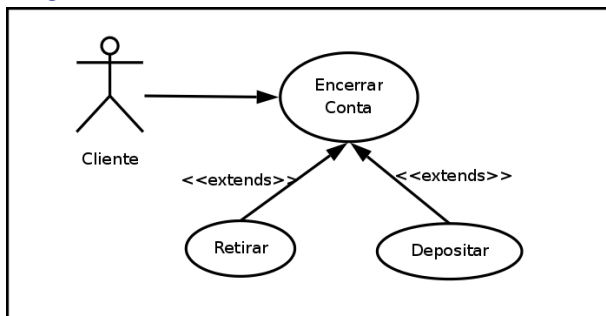


Diagrama de Casos de uso



Bibliografia

- DALL'OGGIO, Pablo. Uma Ferramenta para Gerenciamento de Requisitos em Projetos Baseados em Extreme Programming;
- DALL'OGGIO, Pablo. Um Sistema Multi-Agente Colaborativo para Gestão da Mudança de Requisitos de Software;
- SOMMERVILLE. Software Engineering;

Diagrama de Casos de uso

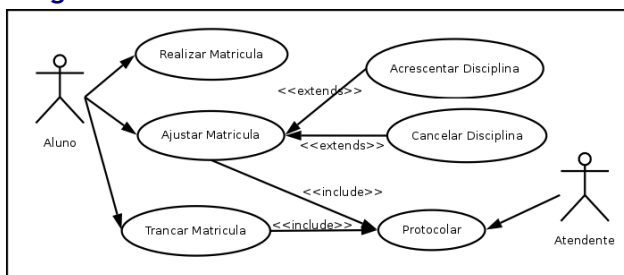


Diagrama de Casos de uso

