



Análise e projeto de sistemas

PROF. REGILAN SILVA

Roteiro da aula

- ▶ Levantamento de requisitos do sistema
- ▶ Tipos de requisitos
- ▶ Regras de negócio
- ▶ Casos de uso
- ▶ Descrição de casos de uso
- ▶ Documentos e modelos

Objetivos

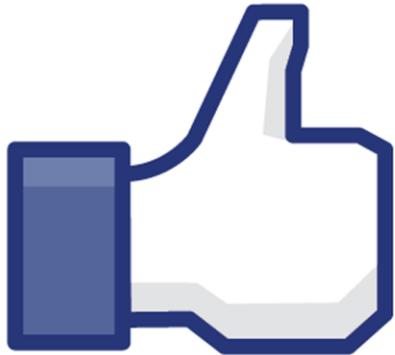
- ▶ Conhecer o que são os requisitos de um sistema.
- ▶ Conhecer técnicas para levantamento de requisitos.
- ▶ Usar o Diagrama de Caso de Uso para especificação das principais funções do sistema.



Os requisitos do sistema

- ▶ O início da construção de um sistema acontece na sua fase de concepção. Algumas etapas da fase de concepção são:
 - ▶ definir uma visão para o produto, ou seja, os objetivos atuais e futuros que os usuários pretendem alcançar com o seu uso;
 - ▶ produzir um plano de negócio e cronograma para o desenvolvimento, produção, implantação e teste do software;
 - ▶ definir do escopo do projeto; ou seja, seus limites de interface com os demais sistemas existentes na empresa;
 - ▶ estimar o custo total aproximado do projeto para que os usuários se capitalizem e se planejem para o projeto.

Os requisitos do sistema



É importante na **fase de concepção** termos certeza se o projeto é viável, do ponto de vista financeiro e computacional, e se ele trará algum benefício para a empresa.

Caso não consigamos estar certos dessas questões, é aconselhável que o projeto seja repensado.

Os requisitos do sistema

- ▶ Para desenvolver um sistema, precisamos descobrir o que o nosso cliente deseja.
- ▶ Nosso objetivo é escrever o **documento de requisitos de sistemas**, que deve conter de forma precisa e realista **tudo o que deve ser feito**, de modo a atender a uma ou mais necessidades do nosso cliente.



Os requisitos do sistema

- ▶ O documento de requisitos deve ser validado pelo cliente antes do desenvolvimento do sistema.
- ▶ Para encontrar os requisitos do sistema, precisaremos conversar com nosso cliente. **Esse cliente não é necessariamente uma única pessoa e pode representar uma empresa com milhares de funcionários!**



Os requisitos do sistema

- ▶ É muito útil a criação de um grupo de modelagem, que deve ser composto de:
 - a) **investidor:** cliente/financiador do sistema;
 - b) **usuários:** aqueles que irão utilizar diretamente o sistema ou qualquer pessoa que vá ser afetada pelo sistema;
 - c) **analistas ou desenvolvedores:** nós!
 - d) **facilitadores:** pessoas que conhecem bem a empresa e possuem habilidade para fazer reuniões; entendem o processo de modelagem dos requisitos e podem fazer perguntas válidas e inteligentes;
 - e) **escribas:** pessoas que devem escutar bem, possuir habilidade na comunicação oral e escrever bem. São eles que irão anotar os requisitos durante as reuniões e ajudar na escrita do documento de requisitos.

Técnicas de identificação de requisitos

- ▶ Para obtermos os requisitos de um sistema, utilizamos duas técnicas muito conhecidas:
 - a) **entrevistas**;
 - b) **brainstorming** (em português: “tempestade cerebral”): técnica em que um grupo de pessoas discute um assunto e diz qualquer coisa relativa ao assunto que lhe venha à mente sem a preocupação da relevância do que foi dito para o assunto em questão.

Técnicas de identificação de requisitos

- ▶ Recomendações para realizar entrevistas:
 - a) forneça uma agenda/cronograma de reuniões;
 - b) esclareça aos participantes quais os motivos do projeto;
 - c) faça as perguntas críticas em primeiro lugar;
 - d) não assuma que você tem conhecimento do assunto;
 - e) identifique o que é “necessário” e o que é “desejável”;
 - f) termine a entrevista com um sumário dos pontos abordados e redija ata para validação dos participantes



Técnicas de identificação de requisitos

- ▶ Recomendações para realizar **brainstorms**:
 - a) todas as ideias são boas: as ideias não são julgadas pelo grupo;
 - b) todas as ideias são do grupo, ninguém é “dono” de uma ideia;
 - c) todas as ideias são públicas, qualquer pessoa pode expandir ou modificar uma ideia;
 - d) sempre que surgir uma ideia, ela deve ser imediatamente escrita na lousa;
 - e) antes da sessão, forneça ao grupo uma cópia das regras de brainstorming.



O documento de requisitos

- ▶ Este documento deve conter a especificação do sistema, ou seja, uma descrição das necessidades ou dos requisitos do sistema a ser desenvolvido.
- ▶ Nele deve-se descrever:
 - a) uma visão geral do sistema;
 - b) quem são os usuários do sistema;
 - c) quais os objetivos;
 - d) quais as funções principais do sistema;

Tipos de requisitos

- ▶ Os tipos de requisitos são:
 - a) **funcionalidade**: descreve características, capacidades, segurança, etc., do sistema;
 - b) **usabilidade**: descreve características relativas a fatores humanos, ajuda *on-line*, documentação,
 - c) **implementação**: descreve limitações de recursos, linguagens e ferramentas, hardware, etc.;
 - d) **aspectos legais**: descrevem licenciamento, etc, performance: descreve necessidades relativas a tempos de respostas, precisão, disponibilidade, utilização de recursos, etc;
 - e) **suportabilidade**: descreve necessidades relativas à adaptabilidade, manutenibilidade, internacionalização, configurabilidade, etc;

Especificação de requisitos: template para projeto

- ▶ Para exemplificar um documento de requisitos tomamos como exemplo uma faculdade que precisa de uma aplicação para controlar alguns processos acadêmicos, como inscrições em disciplinas, lançamentos de notas, alocação de recursos para turmas, etc.
- ▶ [Ver exemplo](#)



Regras de negócio

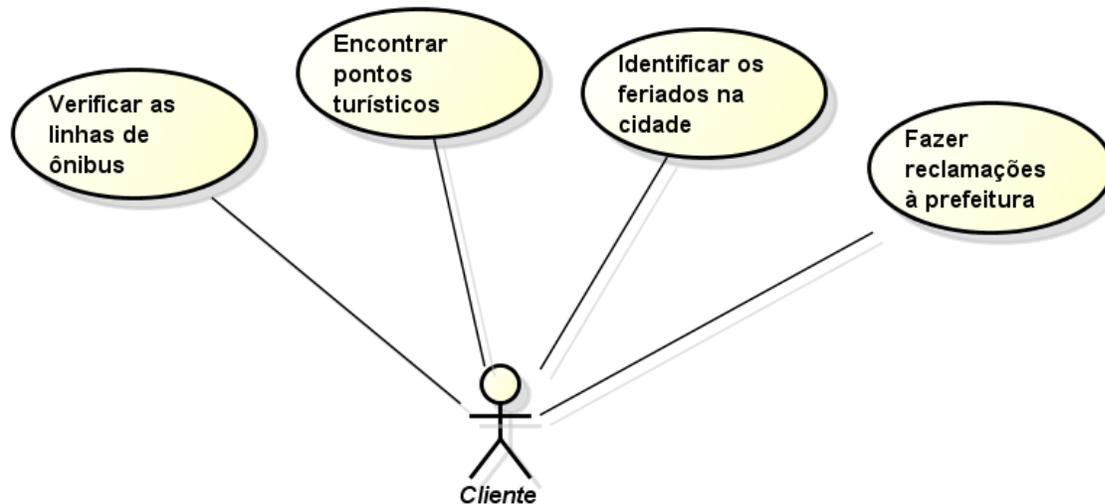
- ▶ Conjunto de instruções que os usuários já seguem e que o sistema a ser desenvolvido deve contemplar.
- ▶ Restrições, validações, condições e exceções do processo são exemplos clássicos de regras de negócio. Uma regra de negócio não necessariamente será refletida no sistema como uma funcionalidade, mas ela com certeza determinará o comportamento de uma ou mais funcionalidades do sistema

Regras de negócios: template para projeto

- ▶ Para exemplificar as regras de negócios, seguiremos no exemplo da faculdade que precisa de uma aplicação para controlar alguns processos acadêmicos, como inscrições em disciplinas, lançamentos de notas, alocação de recursos para turmas, etc.
- ▶ [Ver exemplo](#)

Casos de uso

- ▶ Um instrumento muito útil no desenvolvimento de sistemas é o **caso de uso**.
- ▶ Os **casos de uso** são escritos em um documento que serve para orientar a construção dos demais elementos do projeto.



Casos de uso

- ▶ Caso de uso é uma descrição de um conjunto de interações entre um usuário e o sistema, que serve para orientar a construção dos demais elementos do projeto.
- ▶ Para que um caso de uso seja executado, este deve está ligado a um ator, que é alguém ou alguma coisa que participa do caso de uso e é externo ao sistema.

Atores

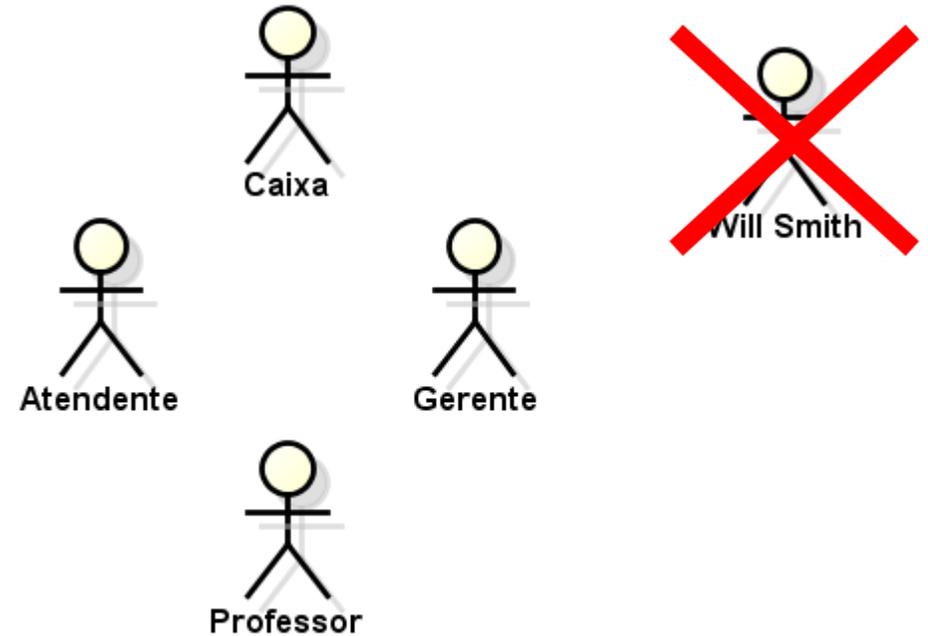
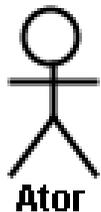
- ▶ Na definição de casos de uso encontramos o termo “usuário”. Nos casos de uso o “usuário” é chamado de ator.
- ▶ Temos dois tipos de atores:
 - a) **ator iniciador:** é ele que inicia o caso de uso, geralmente, sendo o seu ator principal. Por exemplo, se estamos fazendo um sistema bancário, o caso de uso “retirar dinheiro” terá como ator iniciador o ator cliente;
 - b) **ator participante:** é aquele que de alguma forma participa do caso de uso. Por exemplo, no caso de uso “comprar itens”, existe um ator atendente que “atende” o ator cliente.

Atores

- ▶ Um ator não precisa, necessariamente, ser uma pessoa. Um ator pode representar:
 - a) um grupo de pessoas. Por exemplo, clientes, estudantes, etc.;
 - b) uma função de uma pessoa. Por exemplo, estudante representante da turma;
 - c) outro sistema externo;
 - d) um conceito abstrato como hora, data, etc. Por exemplo, o ator data pode iniciar um caso de uso chamado “cancelar pedidos” para pedidos feitos há mais de seis meses.
 - e) um hardware.

Atores

- ▶ Um ator geralmente participa de mais de um caso de uso e representa papéis.
- ▶ Dê um nome ao ator de forma que fique claro seu papel no sistema.



Atores – Regras para identificar

- a) Qual o principal usuário do sistema?
- b) Quem fornece as informações para o sistema?
- c) Quem obtém as informações do sistema?
- d) Quem instala o sistema?
- e) Quem opera o sistema?
- f) Quem desativa o sistema?
- g) Que outros sistemas interagem com o sistema?
- h) Algum processo acontece automaticamente em horas predeterminadas?
- i) Quem fornecerá, utilizará ou removerá informações do sistema?

Identificação dos casos de uso

- ▶ Os casos de uso são encontrados e descritos por meio de entrevistas e **brainstorms**. O roteiro mais indicado para a aplicação dessas técnicas é:
 1. identificação de todos os atores;
 2. identificação de todos os casos de uso;
 3. descrição simples de cada caso de uso;
 4. desenho de um diagrama de casos de uso.

Diagrama de casos de uso

- ▶ Um diagrama de caso de uso é uma representação gráfica dos casos de uso e dos atores envolvidos no sistema em análise.
- ▶ O diagrama de casos de uso é um diagrama da UML cujo objetivo é representar um *requisito* do sistema que será automatizado.
- ▶ Considere como requisito uma necessidade/funcionalidade do sistema.

Diagrama de casos de uso

- ▶ Os símbolos que utilizamos são mostrados abaixo:



Ator

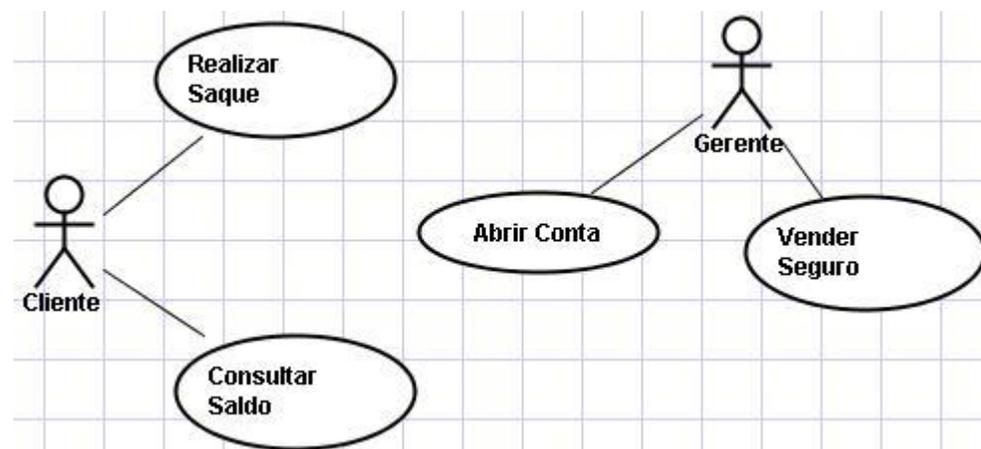
Utilizamos este símbolo
para representar um
ATOR



Utilizamos este símbolo
para representar um
CASO DE USO

- ▶ Usamos atores para representar as entidades que interagem com o sistema. Podem ser usuários, máquinas, sensores, etc... Um ator representa um papel no sistema, mas um papel pode ser representado por vários atores.

Diagrama de casos de uso



- ▶ O diagrama de casos de uso é muito interessante para dar uma visão geral do sistema, com seus atores, casos de uso e a relação entre eles.
- ▶ No exemplo acima, o ator **cliente** executará os casos de uso “realizar saque” e “consultar saldo”, enquanto o gerente poderá interagir com os casos de uso “abrir conta” e “vender seguro”.

Diagrama de casos de uso – tipos de interação

► Inclusão:

- Um caso de uso inclui outro caso de uso (**precisa de, é composto de**)
- Representada através de um arco pontilhado com o rótulo **<<include>>** ou **<<include>>**

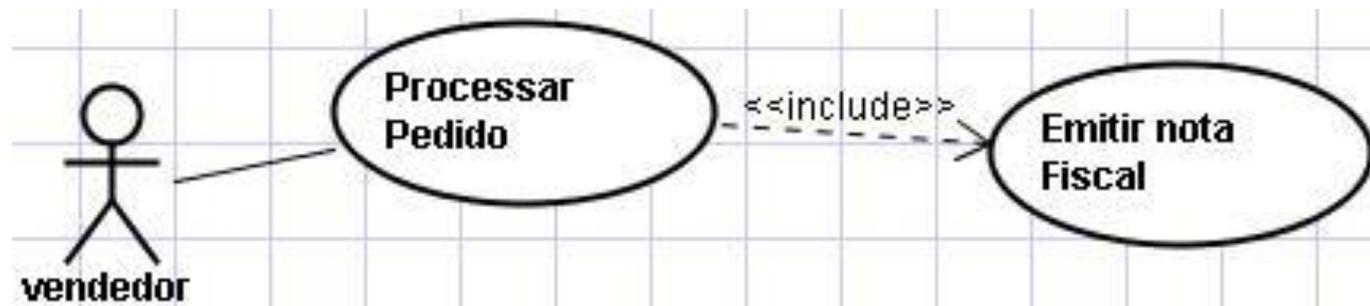


Diagrama de casos de uso – tipos de interação

► Extensão:

- Um caso de uso pode opcionalmente utilizar um outro

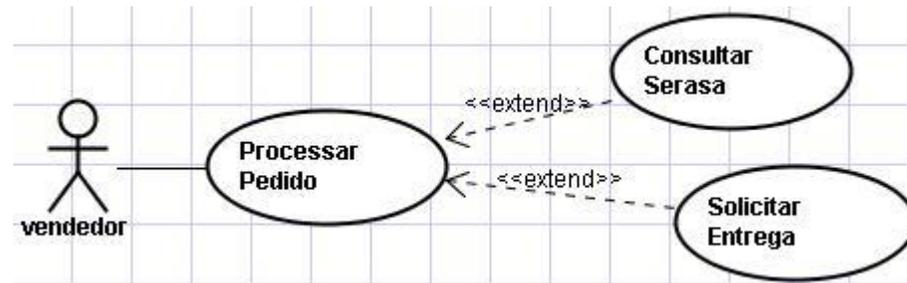
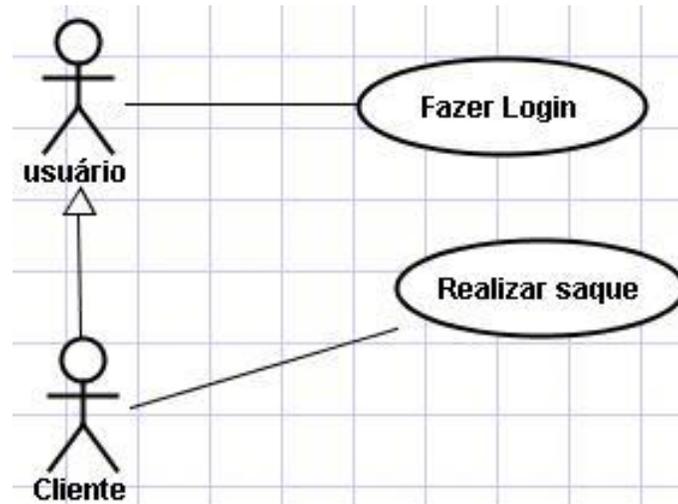


Diagrama de casos de uso – tipos de interação

- ▶ **Herança:** o ator pode herdar as funcionalidades (casos de uso) de outro ator.



Descrição de casos de uso

- ▶ Com relação a descrição dos casos de uso, podemos dividi-los em dois grupos
 - a) **caso de uso de alto nível:** descreve um processo de forma muito breve, usualmente, em duas ou três sentenças. Contém os dados: nome do caso de uso, atores, tipo, descrição.
 - b) **caso de uso expandido:** descreve um processo com mais detalhes que um caso de uso de alto nível.

Descrição de casos de uso: template para projeto

- ▶ Para exemplificar a documentação e descrição de casos de uso, seguiremos no exemplo de uma faculdade que precisa de uma aplicação para controlar alguns processos acadêmicos, como inscrições em disciplinas, lançamentos de notas, alocação de recursos para turmas, etc.
- ▶ [Ver exemplo](#)

Ferramenta case para modelagem

- ▶ Ferramentas CASE (do inglês Computer-Aided Software Engineering) é uma classificação que abrange todas ferramentas baseadas em computadores que auxiliam atividades de engenharia de software, desde análise de requisitos e modelagem até programação e testes.
- ▶ Durante a disciplina utilizaremos o software Astah Community e DBDesigner como ferramentas para elaboração de diagramas e modelo de dados.
- ▶ Astah community: modelagem UML
- ▶ Dbdesigner: modelagem de dados

Direitos e deveres dos clientes

- ▶ Os principais direitos dos clientes são:
 - a) receber do desenvolvedor as explicações sobre o processo de desenvolvimento de sistemas; receber do desenvolvedor um sistema que atenda às suas necessidades nos aspectos de funcionalidade e qualidade;
 - b) ouvir dos desenvolvedores todas as informações técnicas sem o uso de jargões da área de Informática, e sim numa linguagem simples e compreensível.

Direitos e deveres dos clientes

- ▶ Os principais deveres do cliente são:
 - a) explicar o funcionamento da empresa onde o sistema será implantado;
 - b) disponibilizar tempo para fornecer as informações necessárias;
 - c) ser preciso e claro na descrição dos requisitos; ser capaz de priorizar os requisitos;
 - d) ser capaz de tomar decisões relativas ao sistema e suas funcionalidades;
 - e) realizar, juntamente com o desenvolvedor, revisões dos requisitos.

Próxima aula...

- ▶ Exemplos e exercícios de modelagem de casos de uso