

## Teórico 01 – Modelos & Representações

### SISTEMAS:

- Conjunto integrado de componentes regularmente inter-relacionados e interdependentes, criados para realizar um objetivo definido, com relações definidas e mantidas entre seus componentes, cuja produção e operação como um todo é melhor que a simples soma de seus componentes.

### SUBSISTEMAS:

- Ambiente operativo único e predefinido através do qual o sistema coordena o fluxo de trabalho e o uso de recursos;

*O sistema pode conter vários subsistemas, todos a funcionar independentemente uns dos outros, num ambiente colaborativo de uns com os outros*

### UNIDADE:

- O menor componente possível de um subsistema que pode ser testado individualmente.

### MODELO:

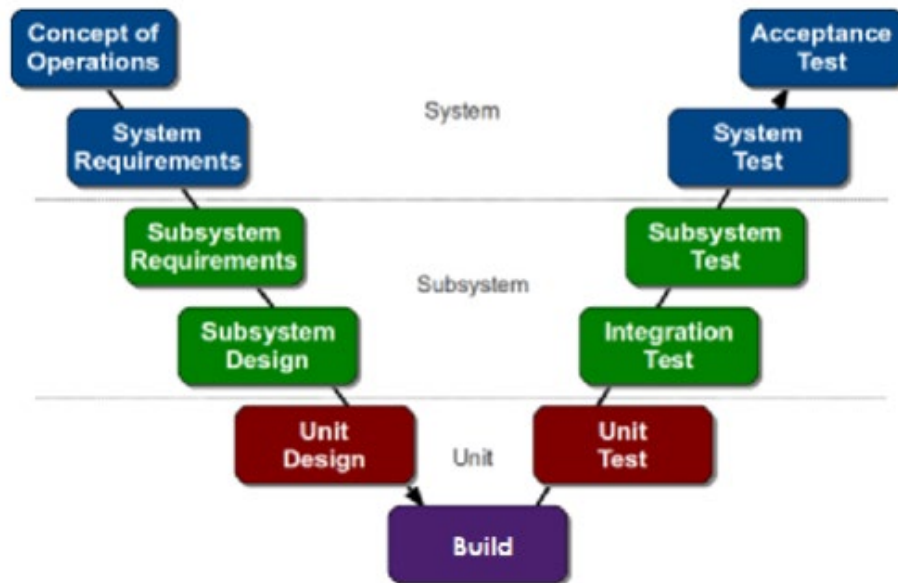
- Representação, através de desenho, diagrama, objeto em escala reduzida, imagem etc., de um objeto, produto, obra de arquitetura, programa etc. a ser reproduzido em dimensões normais e reais.

## MODELO V – DESENVOLVIMENTO DE SISTEMAS

- Processo predefinido para o desenvolvimento de um sistema, subdividido em etapas, as quais devem ser seguidas, para a obtenção do sistema final desejado.

Definição de modelo V para desenvolvimento de Sistemas:

- Um dos modelos estabelecidos, mais utilizados, para ser adotado e seguido durante o desenvolvimento de um sistema



## MODELAGEM

**Definição de Modelagem:** Processo de se construir modelos capazes de antever como os sistemas em estudo ficarão e se comportarão, antes de serem construídos

### Importância da Modelagem:

- 1- Ajuda a visualizar como o sistema será;
- 2- Permite especificar a estrutura e comportamento do sistema modelado;
- 3- Fornece um padrão a ser utilizado durante a construção do sistema;
- 4- Documenta as decisões tomadas durante o desenvolvimento do sistema.

### Finalidade da Construção de Modelos:

- Para entender melhor os sistemas que precisam ser construídos, bem como seus funcionamentos;
- Reflete como o problema deverá ser tratado;
- Expressa diferentes níveis de detalhes e precisão e devem ser conectados com a realidade;
- Descreve os aspectos do sistema que são relevantes ao seu propósito.

### Diagrama:

- Representações gráficas de uma coleção de elementos modelados;
- Demonstram graficamente um esquema simplificado ou um resumo sobre um assunto;
- Normalmente são formado por palavras-chaves ou conceitos, interligados por linhas, que definem o raciocínio para o entendimento do problema.

## UML:

- Sigla para Unified Modeling Language, ou Linguagem de Modelagem Unificada;
- Linguagem para especificação, construção, visualização e documentação de artefatos de um sistema;
- Baseia-se em diagramas que são modelados e classificados em visões de abstração do sistema.

## POO e MOO

- Um conceito que está relacionado com a ideia de classificar, organizar e abstrair coisas, como se fossem objetos;
- Organizar o mundo real como uma coleção de objetos que incorporam estrutura de dados e um conjunto de operações que manipulam estes dados.

### *Onde se aplica a orientação a objetos?*

- Na modelagem e no desenvolvimento de sistemas que utilizam esse paradigma, através das suas notações, regras, conceitos e representações gráficas.

## FIM TEORICO #01

## Téorico 02 - Fundamentos de Requisitos

### **Definição de Requisitos:**

- Os que são requisitados, requeridos;
- Condições para se alcançar determinado fim;
- Quesitos, condições necessárias e básicas para que algo, determinado objetivo ou propósito seja alcançado ou obtido.

### **Definição de Requisitos em Engenharia:**

- Definições documentadas de propriedades ou comportamentos que um produto ou serviço deve atender;
- Informações fundamentais para a fase de projeto de um produto ou serviço, especificando a propriedades e funções necessárias, ou desejáveis, a serem consideradas no desenvolvimento do projeto em questão;
- Definições de características, atributos, habilidades ou qualidades que um sistema, ou qualquer um de seus subsistemas, módulos e sub-rotinas, deve necessariamente prover para ser útil aos seus usuários

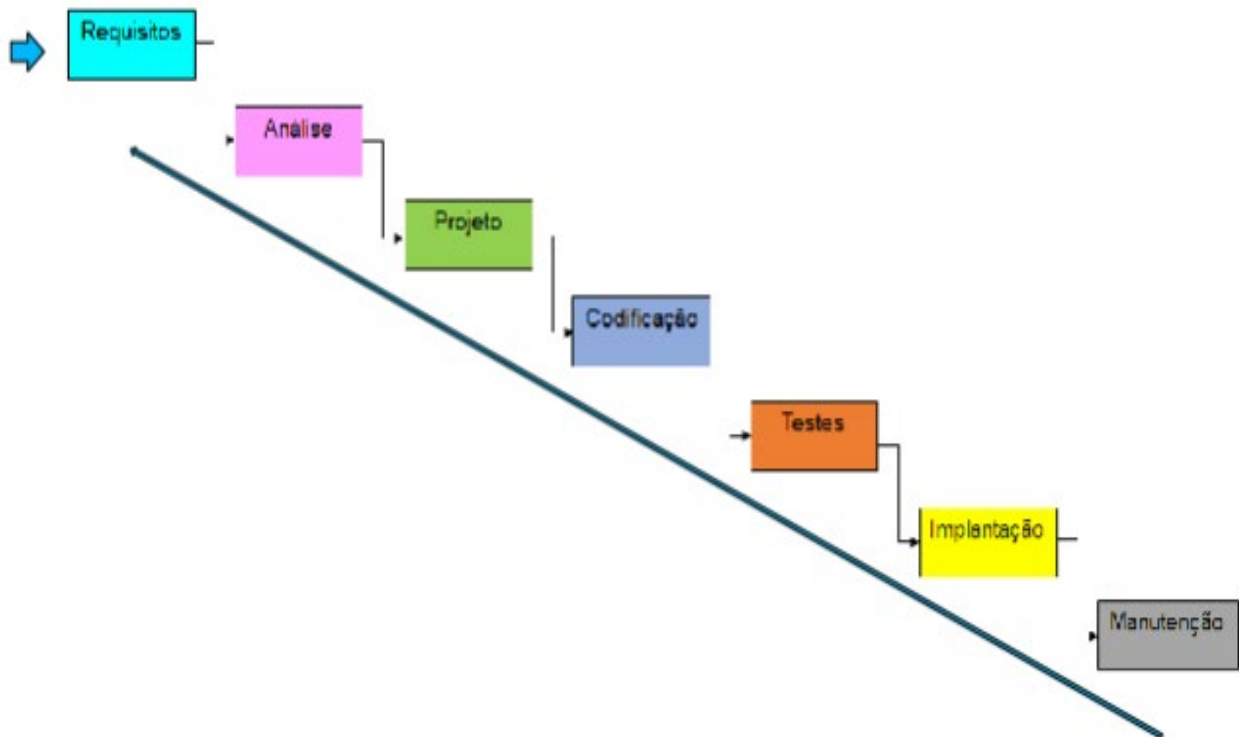
### **Definição de Requisitos de Software:**

- Ações que o software deve executar, possuindo características e condições próprias, de forma a automatizar uma tarefa de um processo de negócio;
- Propriedades que um software exhibe para solucionar problemas reais;
- EM CONJUNTO FORMAM O INDISPENSÁVEL PARA A SOLUÇÃO FUNCIONAR EM HARMONIA

**O que estabelece a engenharia de software:**

- Uma base sólida para o projeto e para a construção e, sem ela, o software resultante tem grande probabilidade de não atender às necessidades do cliente;
- As fases, suas sequências e seus respectivos artefatos para o desenvolvimento de software

## Fases do Desenvolvimento de Software

**Desafios de Engenharia de Software:**

1. Levantar quais são as necessidades dos clientes;
2. Descobrir o que os clientes realmente desejam;
3. Comunicar-se claramente com os clientes;
4. Verificar se os desejos dos clientes podem ser atendidos;
5. Informar e negociar as reais necessidades dos clientes

**Tipos de Requisitos de Software:**

## Tipos de Requisitos de Software

NÃO Funcionais(RNF)

FUNCIONAIS(RF)

### Requisitos NÃO FUNCIONAIS:

- Requisitos que se referem aos critérios que qualificam os requisitos funcionais ou de qualidade para o software;
- Por exemplo, requisitos de desempenho, usabilidade, confiabilidade, robustez, requisitos de entrega, implementação, rastreabilidade, de testes etc.

### Requisitos FUNCIONAIS:

- Requisitos que se referem ao que o sistema deve fazer, suas ações, funções, funcionalidades, informações e serviços;
- Requisitos relacionados às funções de negócios do dia a dia, que os usuários executam no sistema;

### Descrição de Requisitos NÃO FUNCIONAIS:

- Normalmente, na Forma Declarativa, textual

Identificador	Descrição	Categoria	Escopo	Prioridade	Requisitos Relacionados
RNF01	O sistema deve controlar o acesso às funcionalidades. Funcionalidades para controlar o acervo da locadora devem ser restritas a administradores. Funcionalidades de atendimento a clientes devem estar restritas a atendentes. Funcionalidades de consulta ao acervo devem estar disponíveis na Internet.	Segurança de Acesso	Sistema	Alta	
RNF02	A consulta ao acervo deve estar disponível pela Internet, a partir dos principais navegadores disponíveis no mercado.	Portabilidade	Funcionalidade	Média	
RNF03	Os itens devem ser identificados por um código de barras, sendo possível a leitura dos mesmos usando dispositivos de leitores de código de barras.	Facilidade de Operação	Funcionalidade	Alta	

## Taxonomia dos Requisitos Não-Funcionais – RNF?

- Ciência ou técnica de classificação dos Requisitos Não-Funcionais, segundo suas características

## Taxinomia dos Requisitos Não-Funcionais – RNF



## Descrição de Requisitos FUNCIONAIS

1. Em geral, por meio do Modelo de Casos de Uso, através do Diagrama de Casos de Uso, da UML, e da Especificação de Casos de Uso;
2. Na Forma Declarativa, textual;
3. Pelas User Stories, através de Cartões de Histórias.

RF = FORMA DECLARATIVA

Identificador	Descrição	Prioridade	Requisitos Relacionados
RF01	O sistema deve registrar locações, indicando o cliente e os itens locados, bem como a data e o valor da locação e a data de devolução prevista de cada item.	Alta	RF06,RF09,RN01,RN08,RN11,RN12,RN13, RN14,RN16,RNF03,RNF04
RF02	O sistema deve registrar devoluções, indicando os itens devolvidos e a data de devolução	Alta	RF01,RN02,RN08,RNF03,RNF04
RF03	O sistema deve registrar os pagamentos de locações	Alta	RF01,RF02,RN01,RN08,RN09,RNF04, RNF05,
RF04	O sistema deve registrar a reserva de filmes a clientes, permitindo indicar, ainda, o tipo de mídia desejado	Média	RF06,RF09,RF01,RF02,RN10,RN16,RN17, RNF04
RF05	O sistema deve permitir o cancelamento de uma reserva, tanto pelo usuário, quanto automaticamente pelo sistema, quando expirado o prazo para retirada do item, de acordo com política da empresa.	Média	RF05,RN03,RN15,RNF04

## User Stories

**User Story 85222:** Relatório analítico de processos por situação:

**Menu:** relatórios

**Solicitante:** Cliente

**Link para homologação:** sistema-desenv

**Perfil de acesso:**

1. todos os servidores com permissão ao sistema de relatórios
2. Perfil ambiente de produção "administrador"

Eu **como** gestor **gostaria** de gerar um relatório de processos, que fosse separado por situação **para** ter uma visão geral da minha lotação.

**Dado que** o gestor informe o período para o relatório

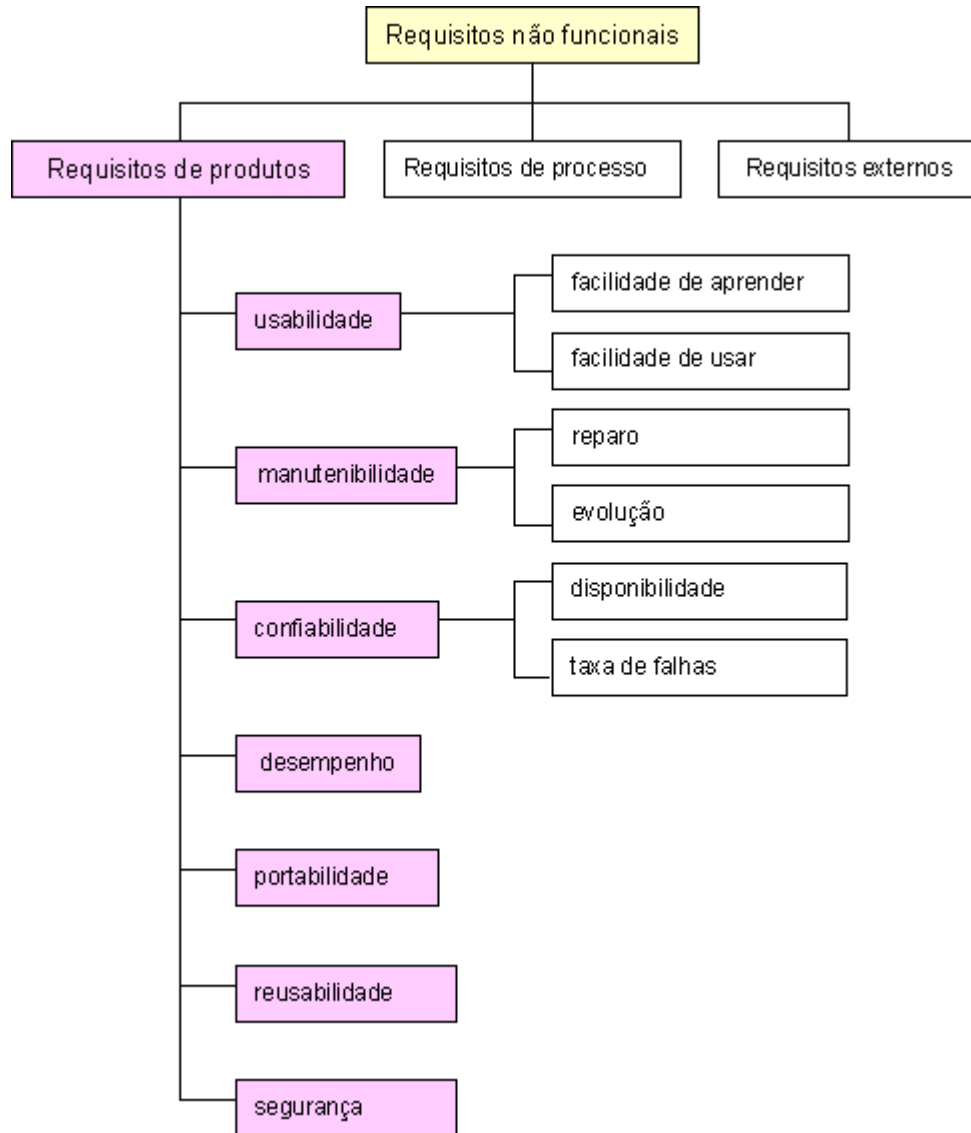
Quando clicar no botão "Gerar"

**Então** o relatório aparecerá em "grid" na tela e os processos devem ser agrupados por situação ( Pendente de recebimento, na lotação, elaborando parecer, concluído, e enviado para outra lotação), caso não existam processos em determinadas situações, será informado " Sem resultado"

**Resultado esperado:** Realizar a correção no relatório de consulta por situação

### COMPLEMENTO SOBRE REQUISITOS NAO FUNCIONAIS:

- Os requisitos não funcionais abordam aspectos de qualidade importantes em sistemas de software.



## Os 7 TIPOS de REQUISITOS NAO FUNCIONAIS:

1. USABILIDADE
2. MANTENIBILIDADE
3. CONFIABILIDADE
4. DESEMPENHO
5. PORTABILIDADE
6. REUSABILIDADE
7. SEGURANCA

### 1- USABILIDADE:

- A USABILIDADE e quando ocorre **interação entre o sistema e seres humanos**.
- O sistema projetado deveria ser fácil de aprender e fácil de usar, tornando assim **fácil e agradável a realização de qualquer tarefa**.
- Especificam nível de desempenho e satisfação do usuário no uso do sistema.



- FORMAS DE **EXPRESSAR A USABILIDADE**:
  - **Facilidade de aprender**: Associado ao tempo e esforço mínimo exigido para alcançar um determinado nível de desempenho no uso do sistema.
  - **Facilidade de uso**: Relacionado à velocidade de execução de tarefas e à redução de erros no uso do sistema.
- Coleta dos requisitos de Usabilidade se baseiam no uso de algumas das técnicas de elicitação de requisitos como entrevistas ou observação
- Requisitos de **USABILIDADE** podem ser expressos através de métricas de usabilidade, expressas em termos de medidas de desempenho.
- A figura abaixo mostra exemplos de requisitos de usabilidade não funcionais:

1. Tempo para realizar um tarefa.
2. Percentual de tarefa concluído.
3. Percentual de tarefa concluído por unidade de tempo.
4. Taxa de sucessos/falhas.
5. Tempo consumido com erros.
6. Percentual de erros.
7. Número de comandos utilizados.
8. Número de comandos disponíveis não utilizados.
9. Frequência de uso de *ajuda* (help) ou documentação.
10. Número de vezes que o usuário expressa satisfação ou frustração.

## 2- **MANTEABILIDADE**:

## 3- **CONFIABILIDADE**:

## 4- **PORTABILIDADE**:

## 5- **REUSABILIDADE**:

## 6- **SEGURANCA**:

# Teórico 03 - Levantamento de Requisitos

## Definição Elicitação de Requisitos:

- Definir, tornar explícito, obter o máximo de informação sobre extrair os requisitos;
- Processo de buscar, descobrir, adquirir e elaborar requisitos para sistemas baseados em computador;
- Primeira atividade no processo de engenharia de requisitos, na qual se busca entender quais são as necessidades do usuário que devem ser atendidas pelo software que será desenvolvido.

## Levantamento e Análises de Requisitos:

- Processo de compreensão e identificação das necessidades que o cliente espera ser solucionado pelo sistema que será desenvolvido, definindo o que o software vai fazer;
- Primeira etapa no ciclo de desenvolvimento de software, onde são definidas as funcionalidades e o escopo do projeto;
- Tarefa executada pelo Analista de Requisitos.

**Maior DIFICULDADE da Análises de Requisitos:**

- A comunicação entre o cliente e o analista de requisitos, por exemplo:
  - Um gerente ou diretor da empresa que não utiliza
  - Sistema diretamente, ao passar os requisitos para
  - Analista, faz isso do seu próprio ponto de vista, baseado apenas em observações que, às vezes, não traduz fielmente a forma que os usuários enxergam o mesmo cenário.

**Analista de Requisitos:**

- Enxergar todos os cenários possíveis, evitando omissões ou má compreensão do cenário, causando erros durante a criação das funcionalidades.

Contextualização de Elicitação de Requisitos:

<https://www.youtube.com/watch?v=kCJFBmAAvV4>

Contextualização de Especificação de Requisitos

<https://www.youtube.com/watch?v=kCJFBmAAvV4>

**Como deve agir o analista de requisitos:**

- Enxergar todos os cenários possíveis, evitando omissões ou má compreensão do cenário, causando erros durante a criação das funcionalidades.

**O que se realiza na fase de Levantamento de Requisitos?**

- Descobrir mais informações sobre o domínio da aplicação, quais serviços o sistema deve oferecer, o desempenho exigido etc.;
- Envolver diferentes tipos de pessoas da empresa;
- Definir os Stakeholders, ou seja, os envolvidos e/ou interessados pelo sistema, os quais terão influência direta ou indireta sobre os requisitos do sistema, por exemplo, usuários finais, área de TI, gerentes de negócio etc.

**O que são Técnicas de Extração de Requisitos?**

- **Procedimentos genéricos** e que devem fazer parte de qualquer processo de extração de requisitos:
  - **Perguntar:** identificar a pessoa apropriada e perguntar quais são os requisitos do sistema;
  - **Observar e inferir:** observar o comportamento dos usuários e inferir suas necessidades a partir dos seus comportamentos;
  - **Discutir e formular:** discutir com os usuários as reais necessidades e formular um entendimento comum dos requisitos do sistema.

- **Negociar a partir de um conjunto padrão:** começar com um conjunto padrão de requisitos e negociar com os usuários quais dessas características serão incluídas, excluídas ou modificadas;
- **Estudar e identificar problema:** investigar os problemas para identificar os requisitos que podem melhorar o produto;
- **Supor:** quando não existe acesso ao usuário, ou para a criação de um produto inédito, é preciso usar a intuição para identificar características ou funções que o usuário pode desejar

### Quais são Técnicas de Extração de Requisitos mais usadas?

1. Entrevistas;
2. Questionários;
3. Brainstorming;
4. JAD – Joint Application Development.
5. Prototipação

Técnica de Extração	Descrição
Entrevistas	<ul style="list-style-type: none"><li>• São séries de encontros com os stakeholders que explicam os seus trabalhos, ambientes em que atuam, necessidades etc.;</li><li>• Requerem desenvolvimento de habilidades sociais, como saber ouvir, inferir, dialogar etc., por parte da equipe de levantamento de requisitos.</li></ul>
Questionários	<ul style="list-style-type: none"><li>• Forma rápida de se obter dados de uma grande quantidade de usuários que podem estar em lugares geograficamente distintos;</li><li>• Tipos de dados que podem ser coletados:<ul style="list-style-type: none"><li>○ Utilização do sistema atual;</li><li>○ Problemas e dificuldades que os usuários enfrentam em seu trabalho;</li><li>○ Expectativa dos usuários em relação ao novo sistema.</li></ul></li><li>• Questões claras e objetivas;<ul style="list-style-type: none"><li>○ Preparar mais de uma questão para um tópico a fim de confirmar a resposta e deixá-la mais completa</li></ul></li></ul>
Brainstorm	<ul style="list-style-type: none"><li>• Técnica básica para geração de ideias;</li><li>• Uma ou várias reuniões que permitem às pessoas sugerirem e explorarem ideias sem que sejam criticadas ou julgadas;</li><li>• Existir um líder responsável por conduzir a reunião sem restringi-la</li><li>• Especialmente útil no começo do processo de extração de requisitos;</li></ul>

	<ul style="list-style-type: none"><li>• Desvantajoso por ser uma técnica não muito estruturada e não produzir a mesma qualidade ou nível de detalhe de outras técnicas.</li><li>• Dividida em 2 Etapas<ol style="list-style-type: none"><li>1. Geração das ideias: são as reuniões que tem como objetivo fornecer ideias, sem discussões sobre o mérito delas e existem 4 regras:<ol style="list-style-type: none"><li>a. É proibido criticar ideias;</li><li>b. Ideias não convencionais ou estranhas são encorajadas</li><li>c. Número de ideias geradas deve ser bem grande;</li><li>d. Os participantes devem ser encorajados a enriquecer ideias dos outros participantes.</li></ol></li></ol></li></ul>
	<b>2. Consolidação das ideias:</b> As ideias geradas são discutidas, revisadas, organizadas, avaliadas, consolidadas, descartadas e priorizadas.
<b>JAD – Joint Application Development?</b>	<ul style="list-style-type: none"><li>• Técnica para promover cooperação, entendimento e trabalho em grupo entre usuários e desenvolvedores</li><li>• Composta por 4 princípios:<ol style="list-style-type: none"><li>1. Dinâmica em grupo: utilização de sessões de grupo (encontros de usuários e desenvolvedores);</li><li>2. Uso de técnicas visuais: utilização de folhas de rascunho, Datashow, vídeos etc.;</li><li>3. Manutenção do processo organizado e racional: controle da sessão através de um líder de sessão que tem o objetivo final de identificar um conjunto preliminar de requisitos;</li><li>4. Utilização de documentação padrão: criação de um documento com os requisitos identificados.</li></ol></li><li>• Geralmente composta por 6 (seis) papéis, ou responsabilidades, divididos entre a equipe de desenvolvimento e os usuários:<ol style="list-style-type: none"><li>1- Líder da sessão;</li><li>2- Engenheiro de requisitos;</li><li>3- Executor;</li><li>4- Representantes dos usuários;</li><li>5- Representantes de produtos de software; e</li><li>6- Especialistas.</li></ol></li></ul>
<b>Prototipação</b>	<ul style="list-style-type: none"><li>• <b>Definindo Prototipação</b><ul style="list-style-type: none"><li>○ Técnica que tem como objetivo extrair, entender e validar requisitos;</li><li>○ Conceito do processo de desenvolvimento de software por prototipação.</li></ul></li></ul>

## Conclusões

- **Requisitos não são fáceis de serem levantados, nem de serem entendidos, tampouco de serem especificados!**
- **Termos vagos, ambíguos, de múltiplos requisitos que devem ser evitados:**

O usuário de internet rapidamente vê os dados da sua conta corrente.

X

O Sistema de Estoque deve ser fácil de usar e requerer um mínimo de treinamento exceto no modo de programação.

X

O módulo de Entrada de Pedidos deve consistir de: entrada de dados e comunicação, processamento de pedidos, resultado do processamento e apresentação de relatório. O módulo de Entrada de Pedidos deve ser integrado com o sistema de intranet e os resultados devem ser armazenados nos registros eletrônicos.

X

O Sistema de Pedidos deverá ser totalmente adaptável e configurável para todas as situações e não precisar de configurações frequentes por parte do usuário.

X

## LAB-4: Requisitos e Modelo de Casos de Uso

### Definição Engenharia de Requisitos:

- Termo usado para descrever as atividades relacionadas à produção (levantamento, registro, validação e verificação) e gerência (controle de mudanças, gerência de configuração, rastreabilidade, gerência de qualidade dos requisitos) de requisitos.



Definição

definicao Requisito:

- Característica do sistema ou a descrição de algo que o sistema é capaz de realizar para atingir os seus objetivos;
  - Descrição das funções e restrições do sistema;
  - Propriedade que o programa deve exibir para resolver algum problema no mundo real
  - Condição ou uma capacidade que deve ser alcançada ou estar presente em um sistema para satisfazer um contrato, padrão, especificação ou outro documento formalmente imposto;
  - Conjunto de necessidades explicitadas pelo cliente que deverão ser atendidas para solucionar um determinado problema do negócio no qual o cliente faz parte.

### Importância do Requisito

- Embora o requisito seja definido pelo cliente, nem sempre o que o cliente quer é o que o negócio precisa;
- Cabe à equipe de consultores identificar a real necessidade do negócio;
- Estabelece uma base de concordância entre o cliente e o fornecedor sobre o que o software fará;
- Fornece uma referência para a validação do produto;
- Reduz o custo de desenvolvimento, pois requisito mal definidos causam retrabalho.

### Porque a identificação e Especificação dos Requisitos é importante

- É complexa:
- Identifica as reais necessidades do cliente;
- Lida com clientes
- Formaliza as necessidades do cliente através da especificação de requisitos de forma que esta seja de fácil entendimento para o cliente e forneça as informações requeridas pela equipe de desenvolvimento;
- Lida com domínios desconhecidos. Entende-se por domínio o contexto para o qual o software está sendo desenvolvido. Por exemplo: contabilidade, medicina, controle de estoque etc.;
- Define as necessidades do usuário em termos de especificações



### Requisitos NÃO FUNCIONAIS (RNF)

- Requisitos que se referem aos critérios que qualificam os requisitos funcionais ou de qualidade para o software;
- Por exemplo, requisitos de desempenho, usabilidade, confiabilidade, robustez, requisitos de entrega, implementação, rastreabilidade, de testes etc.

**Descrição de RNF**

- Normalmente, na Forma Declarativa, textual;

Identificador	Descrição	Categoria	Escopo	Prioridade	Requisitos Relacionados
RNF01	O sistema deve controlar o acesso às funcionalidades. Funcionalidades para controlar o acervo da locadora devem ser restritas a administradores. Funcionalidades de atendimento a clientes devem estar restritas a atendentes. Funcionalidades de consulta ao acervo devem estar disponíveis na Internet.	Segurança de Acesso	Sistema	Alta	
RNF02	A consulta ao acervo deve estar disponível pela Internet, a partir dos principais navegadores disponíveis no mercado.	Portabilidade	Funcionalidade	Média	
RNF03	Os itens devem ser identificados por um código de barras, sendo possível a leitura dos mesmos usando dispositivos de leitores de código de barras.	Facilidade de Operação	Funcionalidade	Alta	

**Taxonomia dos RNF:**

- Ciência ou técnica de classificação dos RNF, segundo suas características.

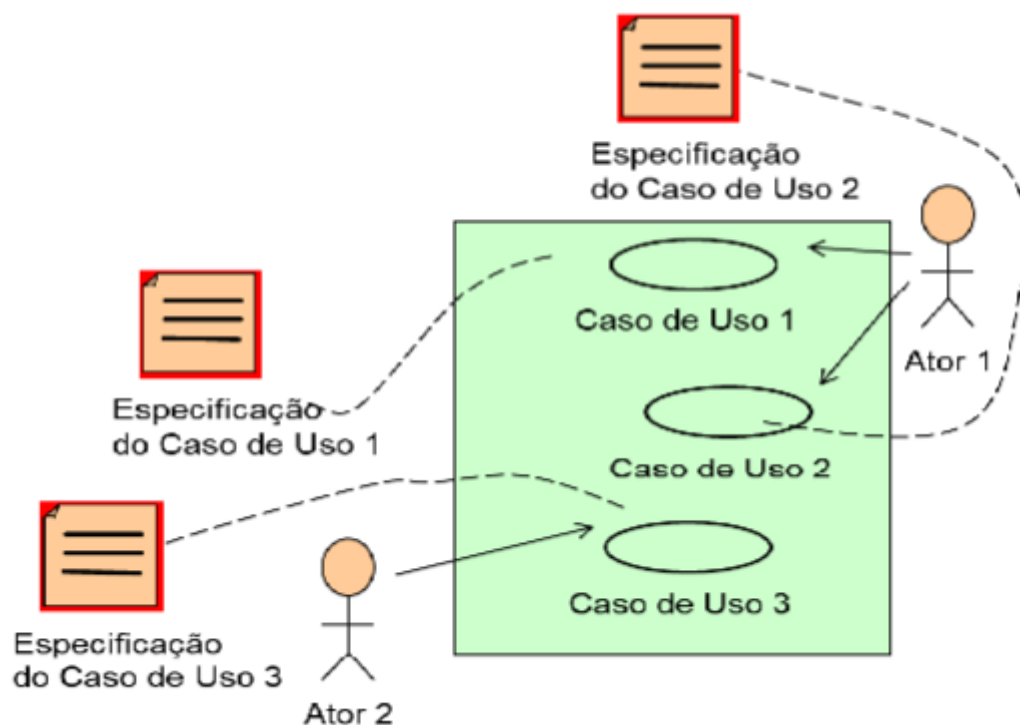
**Requisitos Funcionais:**

- Descrevem as funcionalidades – ações ou funções que o sistema de software deve realizar;
- Descrevem os serviços esperados que o sistema de software deve fazer;
- Normalmente estão relacionados às funções de negócios do dia a dia, que os usuários executa no sistema de software.

**Descrição de Requisitos Funcionais**

- Forma Declarativa – textual;
- Modelo de Casos de Uso, ou Use Cases – UCs

Identificador	Descrição	Prioridade	Requisitos Relacionados
RF01	O sistema deve registrar locações, indicando o cliente e os itens locados, bem como a data e o valor da locação e a data de devolução prevista de cada item.	Alta	RF06,RF09,RN01,RN08,RN11,RN12,RN13,RN14,RN16,RNF03,RNF04
RF02	O sistema deve registrar devoluções, indicando os itens devolvidos e a data de devolução	Alta	RF01,RN02,RN08,RNF03,RNF04
RF03	O sistema deve registrar os pagamentos de locações	Alta	RF01,RF02,RN01,RN08,RN09,RNF04,RNF05,
RF04	O sistema deve registrar a reserva de filmes a clientes, permitindo indicar, ainda, o tipo de mídia desejado	Média	RF06,RF09,RF01,RF02,RN10,RN16,RN17,RNF04
RF05	O sistema deve permitir o cancelamento de uma reserva, tanto pelo usuário, quanto automaticamente pelo sistema, quando expirado o prazo para retirada do item, de acordo com política da empresa.	Média	RF05,RN03,RN15,RNF04



### O que é modelo de caso de uso:

- Uma forma de descrever os requisitos, principalmente os Requisitos Funcionais – RF;
- Conjunto composto por: o Diagrama de Casos de Uso, da UML; e o Especificação de Casos de Uso.

### Diagramas de caso de uso da UML:

- Uma forma gráfica, definida pela UML, de descrever as funcionalidades e os requisitos, principalmente os Requisitos Funcionais – Rf

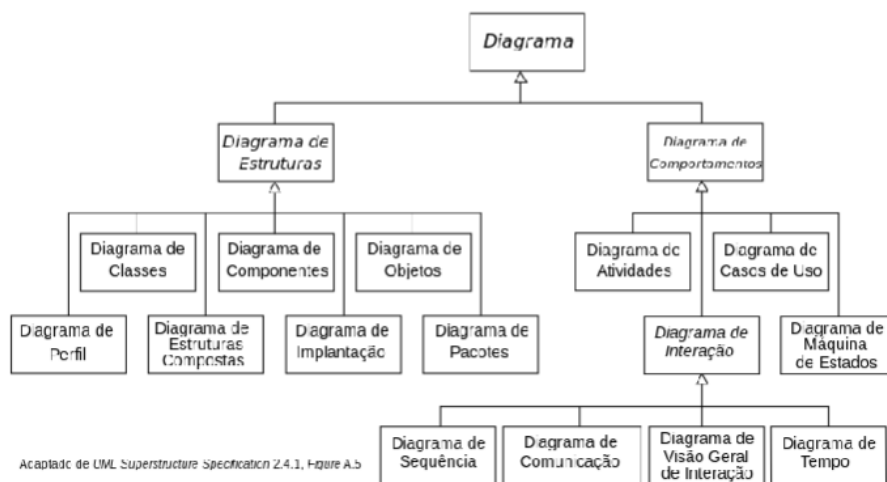
O que é UML:



- A UML, cuja sigla vem de Unified Modeling Language, significa Linguagem de Modelagem Unificada;
- É uma linguagem de notação utilizada para modelar e documentar as diversas fases do desenvolvimento de softwares orientados a objetos, auxiliando na construção, especificação e visualização de um programa;
- Utilizando uma série de elementos gráficos, como retângulos, setas e linhas, a UML consegue criar diagramas para representar as áreas de um software, suas construções, interações e comportamentos.

### Diagramas UML 2.5:

Diagramas Estruturais	Diagramas Comportamentais:
1- Diagrama de Classes; 2- Diagrama de Objetos; 3- Diagrama de Componentes; 4- Diagrama de Implementação ou Instalação; 5- Diagrama de Pacotes; 6- Diagrama de Estrutura Composta; 7- Diagrama de Perfil.	1. Diagrama de Casos de Uso; 2. Diagrama de Sequência; 3. Diagrama de Comunicação ou Colaboração; 4. Diagrama de Transição de Estados; 5. Diagrama de Atividades.

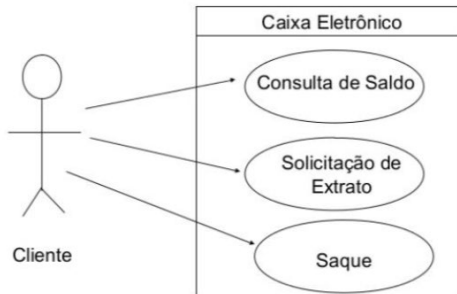


### Objetivo do Diagrama de Caso de Uso:

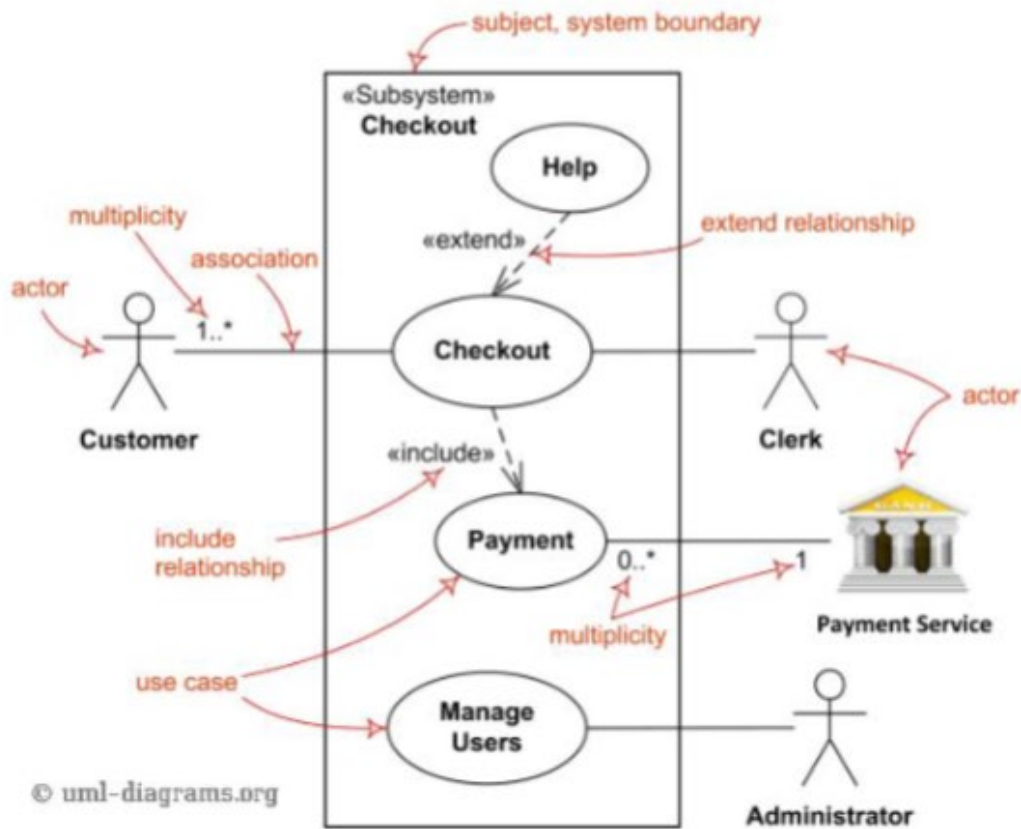
- Demonstrar graficamente as diferentes maneiras que o usuário pode interagir com um sistema;
- Resumir os detalhes dos usuários do seu sistema, conhecidos como atores, e as interações deles com o sistema;
- Ajuda a representar e discutir:
  - Cenários em que o sistema ou aplicativo interage com pessoas, organizações ou sistemas externos;

- Metas que o sistema ou aplicativo ajuda essas entidades, conhecidas como atores, a atingir;
- O escopo do sistema.

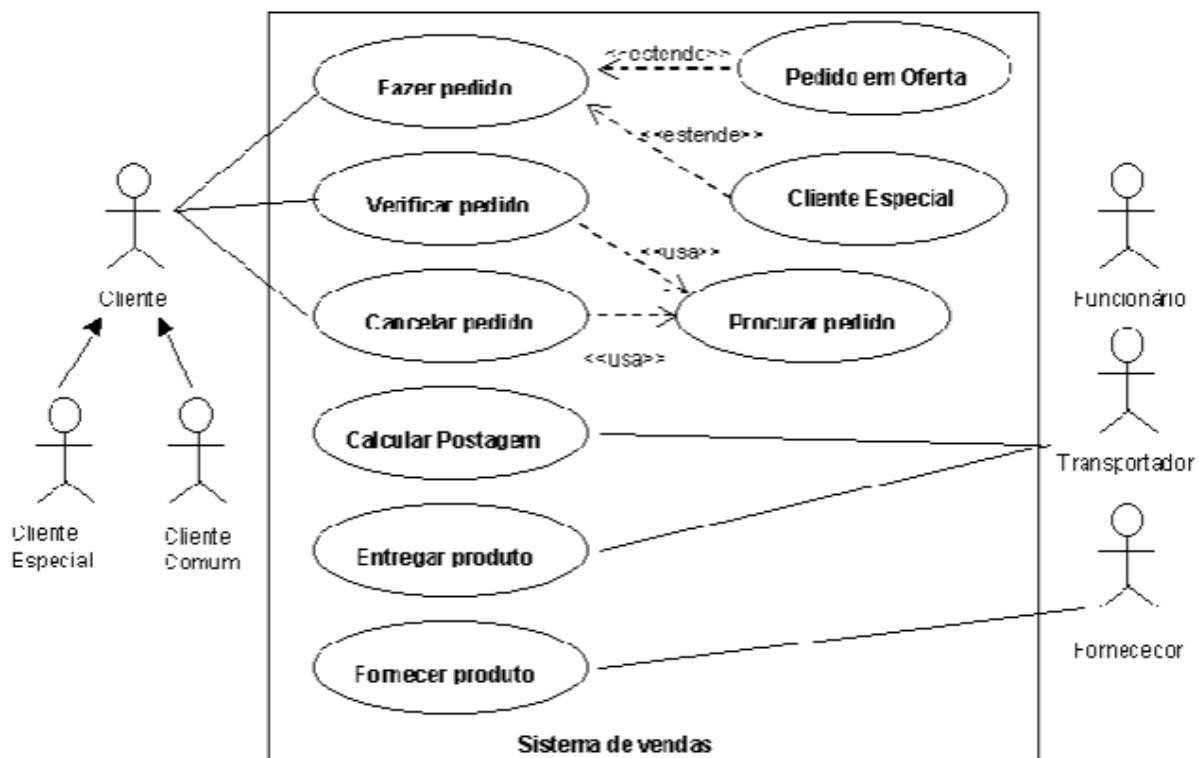
### Exemplo 1



### Exemplo 2



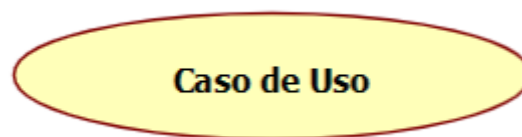
## Exemplo 3



### Composição do Diagrama de Caso de Uso

#### Caso de Uso

- Conjunto de ações que o sistema realiza para produzir um resultado observável por um Ator
- Um dos objetivos do sistema.



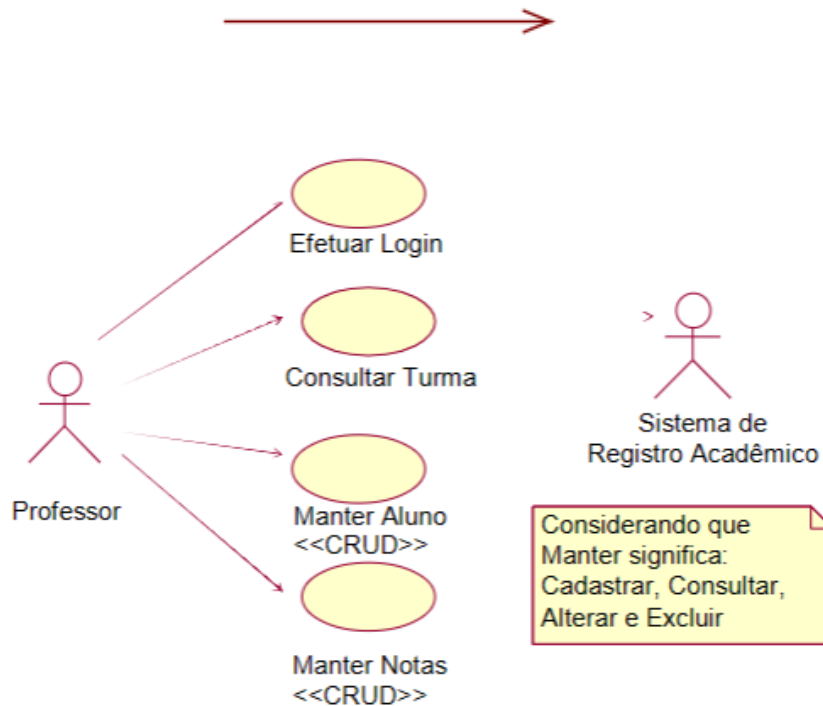
#### Ator:

- Elemento externo ao sistema, alguém ou alguma coisa, que interage com seus UCs;
- Papel executado por alguém ou alguma coisa.



### Associação:

- Relacionamento entre um ator e seus UCs.



### Especificação de Diagrama de Caso de Uso:

- Descrição textual de um cenário específico do sistema/programa a ser desenvolvido.

### Conteúdo das Especificações de Diagrama de Caso de Uso:

- Textos estruturados descrevendo, de forma simples e consistente, os modos como os UCs e os Atores
- interação;
  - Os objetivos dos UCs;
  - Como os UCs são iniciados;
  - Os fluxos das operações entre os Atores e os UCs;
  - Os fluxos alternativos e as condições de exceções;
  - Como os UCs terminam e o que produzem de benefícios aos Atores.

### As Especificações de UCs devem, adicionalmente:

- Concentrar-se nos comportamentos externos do sistema;
  - Ignorar como as tarefas são executadas internamente;
  - Ser claras, de modo que todos os participantes do projeto possam compreendê-las facilmente.

Objetivo:	Contém uma breve descrição do objetivo do caso de uso.
Requisitos:	Neste campo indicamos a qual requisito funcional o caso de uso em questão está associado.
Atores:	Neste campo definimos a lista de atores associados ao caso de uso. Ator é qualquer entidade externa que interage com o sistema (neste caso, com o caso de uso em questão).
Prioridade:	Informação identificada junto ao usuário que auxilia na definição dos casos de uso que serão contemplados em cada iteração do desenvolvimento do software.

Pré-condições:	Neste campo devemos informar as condições que devem ser atendidas para que o caso de uso possa ser executado.
Frequência de uso:	Informação identificada junto ao usuário que auxilia na definição dos casos de uso que serão contemplados em cada iteração do desenvolvimento do software.
Criticalidade:	Informação identificada junto ao usuário que auxilia na definição dos casos de uso que serão contemplados em cada iteração do desenvolvimento do software.
Condição de Entrada:	Neste campo definimos qual ação do ator dará início à interação com o caso de uso em questão.

Fluxo Principal:	Esta é uma das seções principais do caso de uso. É onde descrevemos os passos entre o ator e o sistema. O fluxo principal é o cenário que mais acontece no caso de uso e/ou o mais importante.
Fluxo Alternativo:	Fluxo alternativo é o caminho alternativo tomado pelo caso de uso a partir do fluxo principal, ou seja, dada uma condição de negócio o caso de uso seguirá por outro cenário que não o principal caso essa condição seja verdadeira.
Pós-condições:	Neste campo devemos informar o estado em que o sistema (ou entidade manipulada no caso de uso) estará depois que o caso de uso for executado.
Regras de negócio:	Nesta seção descrevemos todas as regras funcionais que o caso de uso deve cumprir durante sua execução.



- **UC1 – Manter Pessoa Física:**

Objetivo:	Permitir que funcionários adicionem, consultem, removam ou alterem dados de pessoas físicas.
Requisitos:	-
Atores:	Funcionário
Prioridade:	Baixa
Pré-condições:	

Frequência de uso:	Eventual
Criticalidade:	Baixa
Condição de Entrada:	O Funcionário seleciona a opção Manter Pessoa Física.

Fluxo Principal:	<p>1. O sistema apresenta formulário de busca de pessoas físicas contendo as informações:</p> <ul style="list-style-type: none"><li>- CPF (campo editável)</li><li>- Nome (campo editável)</li><li>- Estado (lista dos Estados brasileiros)</li><li>- As opções Buscar e Cancelar</li><li>- A opção Incluir Nova Pessoa Física</li></ul> <p>2. O ator escolhe a opção Incluir Nova Pessoa Física [A1][A2]</p>
------------------	---

	<p>3. O sistema apresenta formulário de cadastro de Pessoas Físicas contendo os seguintes campos a serem preenchidos:</p> <p>(Informações Gerais)</p> <ul style="list-style-type: none"><li>- CPF (campo editável)</li><li>- RG (campo editável)</li><li>- Nome (campo editável)</li></ul> <p>(Endereços)</p> <ul style="list-style-type: none"><li>- Tabela com os endereços cadastrados para a pessoa física que está sendo criada (cada endereço adicionado possui as seguintes informações: Logradouro, Bairro, Complemento, CEP, Município, Estado e País) e a opção de Excluir Endereço ...</li></ul>
--	---

#### níveis de detalhamento das Especificações de UCs

- **Breve:** Descrições concisas, de um único parágrafo, contendo os cenários de sucesso – fluxos principais, para especificação e análise dos requisitos nos estágios iniciais;
- **Casual:** Múltiplos parágrafos informais cobrindo vários cenários possíveis, para especificação e análise dos requisitos nos estágios iniciais;



- **Completo:** Todos os passos e variações registrados com detalhes, contendo pré e pós-condições, após serem identificados e descritos brevemente, para os mais significativos em termos arquiteturais do sistema.

<b>Seção do Caso de Uso</b>	<b>Comentários</b>
Nome	Deve começar com um verbo
Escopo	O sistema em questão
Nível	Primário ou secundário
Atores Primários	Utilizam os sistemas para atender seus objetivos
Interessados e interesses	Quem tem interesse no caso de uso e o que deseja
Precondições	O que deve ser verdadeiro ao iniciar o caso de uso
Garantia de Sucesso	O que deve ser verdadeiro ao término com sucesso do caso de uso
Fluxo principal	Um típico e incondicional cenário de sucesso na execução do caso de uso
Extensões	Cenários alternativos de sucesso ou de falha
Requisitos especiais	Requisitos não-funcionais relacionados ao caso de uso
Alternativas de tecnologia e de formato dos dados	Métodos alternativos para I/O e para o formato dos dados
Frequência	Influencia a investigação, os testes e o momento de implementação
Questões em aberto	Questões importantes para as quais ainda não existe uma solução