



Universidade Presbiteriana Mackenzie
Faculdade de Computação e Informática

Desenvolvimento de Sistemas II

Aula 5 – Especificação de requisitos

Gustavo Scalabrini Sampaio

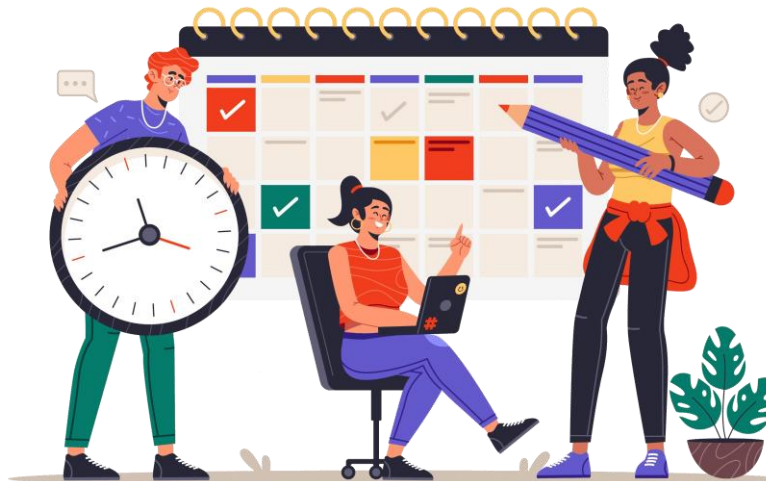
Requisitos

Definição de requisitos:

- Requisitos são **capacidades** e **condições** as quais o **sistema** (e em termos mais amplos, o projeto) **deve atender**.

O principal desafio da análise de requisitos é **encontrar, comunicar e lembrar** (registrar) o que é **realmente necessário**.

Os requisitos devem ser **expressos** de **forma clara** para o cliente e os desenvolvedores.



Requisitos

Como podemos chegar aos requisitos de forma racional?

Gerenciamento de requisitos: abordagem sistemática de encontrar, documentar, organizar e rastrear as mudanças de requisitos de um sistema.

Dentre as técnicas para encontrar requisitos temos:

- escrever casos de uso com os clientes;
- promover seminários de requisitos (desenvolvedores e clientes);
- demonstração dos resultados ao final de interação para solicitar realimentação.



Requisitos

Categorias de requisitos

No uso comum, os requisitos são categorizados como **funcionais** (comportamentais) ou **não funcionais** (os demais: qualidade, usabilidade, desempenho, etc).

É de grande valia **considerar outros tipos de categorização** para os requisitos, com o objetivo de garantir uma melhor cobertura. Um exemplo é o FURPS+:

Categoria	O que descreve
Funcional	Características, capacidades, segurança.
Usabilidade	Fatores humanos, recursos de ajuda, documentação.
Confiabilidade	Frequência de falhas, capacidade de recuperação, previsibilidade.
Desempenho	tempos de resposta, fluxo de vazão (throughput), precisão, disponibilidade, uso de recursos.
Facilidade de Suporte	facilidade de manutenção ou adaptação, internacionalização, configurabilidade.

Requisitos

Categoria +	O que descreve
Implementação	Limitação de recursos, linguagens e ferramentas, hardware.
Interface	Restrições impostas pelas interfaces com sistemas externos.
Operações	Gerenciamento do sistema no sistema operacional.
Empacotamento	Caixa física, por exemplo.
Questões legais	Licença de uso, por exemplo.

- Definir requisitos que se enquadrem nesse tipo de categorização **busca garantir** que os **principais requisitos** sejam **lembrados**. Note que todos os requisitos que não fazem parte do 'Funcional' são tipos de requisitos não funcionais...

Ficou melhor para identificar requisitos não funcionais?

O artefato '**Regras de negócio**', apesar de ser opcional, é **importante** para o **processo** de desenvolvimento. Vale a pena **reservar** um **tempo** para **entender** as **regras de negócio** e refinar os requisitos que serão implementados em cada iteração.

Artefatos que descrevem requisitos

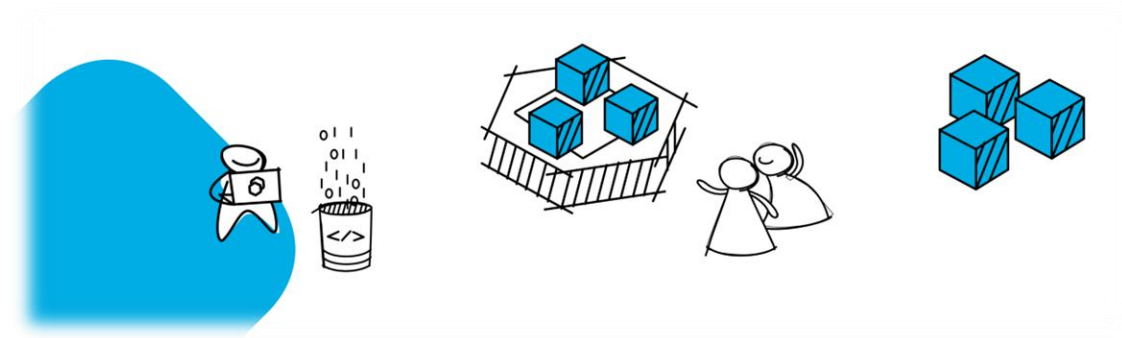
Ao contrário do que se possa imaginar os **requisitos** podem ser **descritos** em **outros artefatos**, além da famosa lista de requisitos.

Esses **artefatos** podem ser '**compilados**' para **gerar** uma **lista de requisitos**.

No entanto, tratar de forma formal ou informal o processo de levantamento de requisitos, dependendo do contexto, facilita a realização dessa tarefa.

Alguns **artefatos** que podem **descrever requisitos**:

- Lista de requisitos;
- Modelo de casos de uso;
- Histórias de usuários;
- Especificação suplementar;
- Glossário;
- Visão;
- Regras de negócio;
- Modelagem de Domínio.



Lista de requisitos

Se pensarmos em uma abordagem 'tradicional', os requisitos são definidos por meio da **engenharia de requisitos**.

- **Como resultado é gerada uma lista de requisitos.**

A lista de requisitos é uma lista padronizada indicando, pelo menos, o **tipo** de requisito (RF - Requisito Funcional; RNF - Requisito Não Funcional), a **descrição** do requisito e sua **prioridade** (depende de critério).

A **coleta colaborativa** de requisitos é uma das formas para **gerar** este artefato.



Engenharia de requisitos

Entendendo os requisitos

A **engenharia de requisitos constrói** uma **ponte entre** o **projeto** e a **construção**; e permite que examinemos:

- O **contexto** do trabalho de software a ser realizado;
- As **necessidades específicas** a que o projeto e a construção devem atender;
- As **prioridades** que orientam a ordem na qual o trabalho deve ser concluído;
- As **informações, funções e comportamentos** que terão um **impacto** profundo no **projeto** resultante.

A engenharia de requisitos **abrange sete tarefas** distintas:

- **Concepção**
- **Levantamento**
- **Elaboração**
- **Negociação**
- **Especificação**
- **Validação**
- **Gestão**

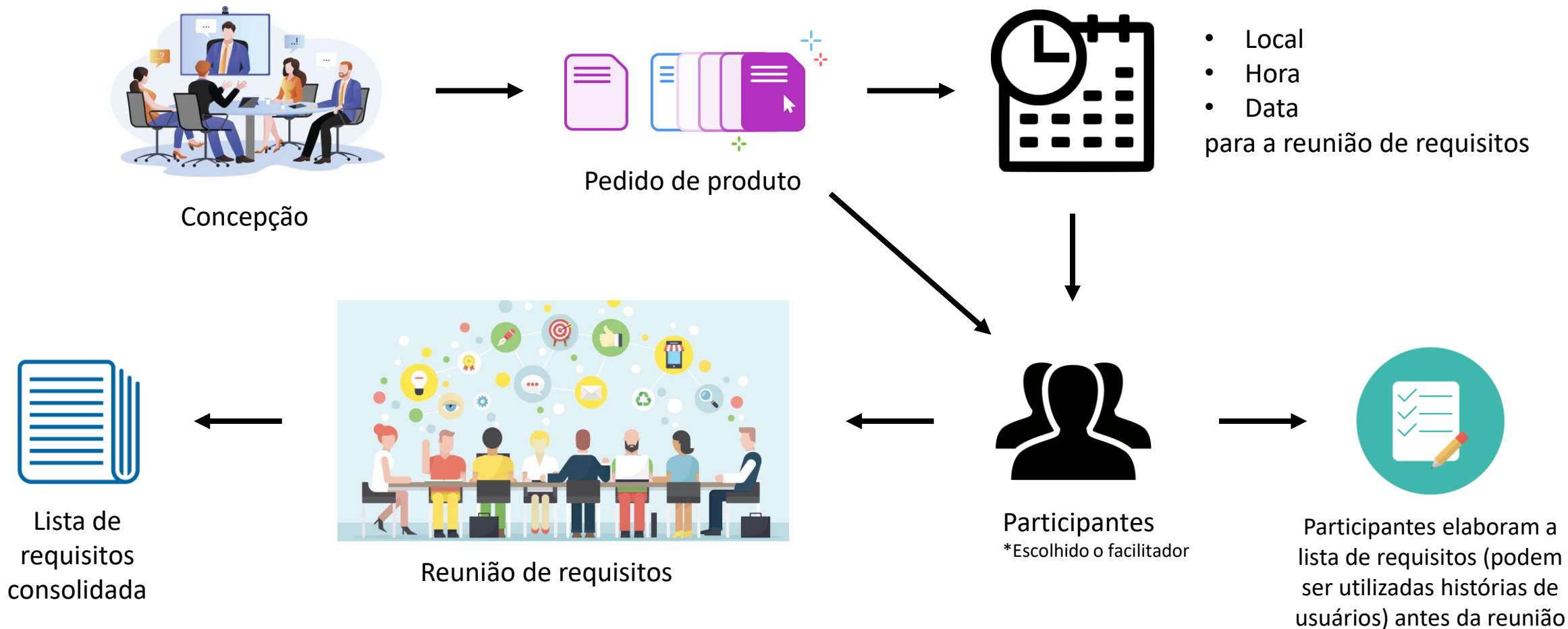


Algumas delas **ocorrem em paralelo** e **todas** são **adaptadas** às **necessidades do projeto**.

Engenharia de requisitos

Levantamento de requisitos

Processo



Engenharia de requisitos

Levantamento de requisitos

Exemplo: Biblioteca da Steam (Recurso da aplicação Steam).

Concepção: Dentro da aplicação Steam um recurso “Biblioteca” deve ser disponibilizado para que os usuários possam gerenciar seus jogos dentro da plataforma.

Requisitos iniciais (Individual):

1. O usuário pode visualizar a lista dos jogos que ele já adquiriu.
2. Notícias dos jogos adquiridos como atualizações devem ser exibidas para o usuário.
3. Uma lista dos últimos jogos adquiridos deve ser exibida em destaque.
4. O usuário deve ter uma opção de adicionar mais jogos à sua biblioteca (Redirecionar para a loja).

Requisitos consolidados e priorizados (Coletivo):

1. O usuário pode visualizar a lista dos jogos que ele já adquiriu.
2. O usuário pode gerenciar os jogos da biblioteca. Jogos podem ser adicionados, removidos ou iniciados.
3. O usuário pode filtrar a lista de jogos com opções como Gênero; quantidade de jogadores; suporte a hardware específico; etc.
4. O retorno da busca não pode demorar mais que 1 segundo.
5. O usuário pode criar uma coleção de jogos personalizada.
6. Sugestões de jogos de estilos similares já adquiridos pelo usuário devem ser exibidos na seção “Jogue também”.
7. Notícias dos jogos adquiridos como atualizações devem ser exibidas para o usuário.
8. O usuário pode configurar a seção de notícias.
9. Uma lista dos últimos jogos adquiridos deve ser exibida em destaque.
10. O usuário deve ter uma opção de adicionar mais jogos à sua biblioteca (Redirecionar para a loja).
11. Por questões de segurança o usuário deve realizar novamente o login ao ser redirecionado para a loja.

OBS: A negociação é um fator importante durante a elaboração e priorização dos requisitos.

Engenharia de requisitos

Levantamento de requisitos

Exemplo: Biblioteca da Steam (Recurso da aplicação Steam).

Concepção: Dentro da aplicação Steam um recurso “Biblioteca” deve ser disponibilizado para que os usuários possam gerenciar seus jogos dentro da plataforma.

Requisitos iniciais (Individual):

1. O usuário pode visualizar a lista dos jogos que ele já adquiriu.
2. Notícias dos jogos adquiridos como atualizações devem ser exibidas para o usuário.
3. Uma lista dos últimos jogos adquiridos deve ser exibida em destaque.
4. O usuário deve ter uma opção de adicionar mais jogos à sua biblioteca (Redirecionar para a loja).

Requisitos não funcionais

Requisitos consolidados e priorizados (Coletivo):

1. O usuário pode visualizar a lista dos jogos que ele já adquiriu.
2. O usuário pode gerenciar os jogos da biblioteca. Jogos podem ser adicionados, removidos ou iniciados.
3. O usuário pode filtrar a lista de jogos com opções como Gênero; quantidade de jogadores; suporte a hardware específico; etc.
4. O retorno da busca não pode demorar mais que 1 segundo.
5. O usuário pode criar uma coleção de jogos personalizada.
6. Sugestões de jogos de estilos similares já adquiridos pelo usuário devem ser exibidos na seção “Jogue também”.
7. Notícias dos jogos adquiridos como atualizações devem ser exibidas para o usuário.
8. O usuário pode configurar a seção de notícias.
9. Uma lista dos últimos jogos adquiridos deve ser exibida em destaque.
10. O usuário deve ter uma opção de adicionar mais jogos à sua biblioteca (Redirecionar para a loja).
11. Por questões de segurança o usuário deve realizar novamente o login ao ser redirecionado para a loja.

Lista de requisitos

Tipo de requisito	Requisito	Prioridade
RF	O usuário pode visualizar a lista dos jogos que ele já adquiriu.	Alta
RF	O usuário pode gerenciar os jogos da biblioteca. Jogos podem ser adicionados, removidos ou iniciados.	Alta
RF	O usuário pode filtrar a lista de jogos com opções como Gênero; quantidade de jogadores; suporte a hardware específico; etc.	Alta
RNF	O retorno da busca não pode demorar mais que 1 segundo.	Alta
RF	O usuário pode criar uma coleção de jogos personalizada.	Média
RF	Sugestões de jogos de estilos similares já adquiridos pelo usuário devem ser exibidos na seção “Jogue também”.	Média
RF	Notícias dos jogos adquiridos como atualizações devem ser exibidas para o usuário.	Média
RF	O usuário pode configurar a seção de notícias.	Baixa
RF	Uma lista dos últimos jogos adquiridos deve ser exibida em destaque.	Baixa
RF	O usuário deve ter uma opção de adicionar mais jogos à sua biblioteca (Redirecionar para a loja).	Baixa
RNF	Por questões de segurança o usuário deve realizar novamente o login ao ser redirecionado para a loja.	Baixa

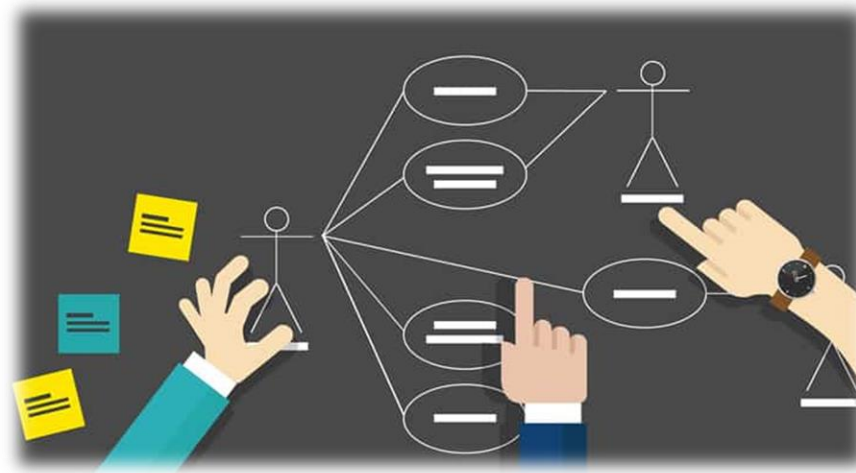
Caso de uso como artefato de requisitos

Como vimos, **casos de uso** são **amplamente utilizados** para **descobrir** e **registrar requisitos**.

Se fizermos uma análise de um caso de uso, podemos perceber que **casos de uso são requisitos!**

Veja, casos de uso **descrevem requisitos funcionais**, uma vez que são indicados os procedimentos que o sistema fará.

Na prática, a lista de requisitos 'tradicional' pode ter sua importância reduzida, já que **podemos redigir casos de uso** para **descrever requisitos funcionais**.



Caso de uso como artefato de requisitos

Veja um pequeno **exemplo comparativo** entre um caso de uso e uma lista de requisitos. **Contexto:** Sistema para aplicar filtros em imagens.

Caso de uso 'Aplicar filtro':

1. Usuário seleciona uma imagem a qual deseja aplicar um filtro.
2. Sistema carrega a imagem em memória, possibilitando realizar sua edição.
3. Usuário seleciona o filtro desejado.
4. Sistema aplica o algoritmo referente ao filtro desejado, gerando uma imagem temporária alterada.
5. Usuário visualiza o resultado da aplicação do filtro e solicita o salvamento da imagem alterada.
6. Sistema grava a imagem modificada no mesmo diretório da imagem original, adicionando ao final do nome do arquivo o pós-fixado '_alt'.
7. Caso de uso encerrado.

Lista de requisitos:

O sistema deve...

- Permitir que o usuário selecione uma imagem para aplicar um filtro.
- Possibilitar ao usuário a seleção do filtro desejado.
- Permitir que o usuário pré-visualize o filtro aplicado.
- Gravar a imagem modificada utilizando um pós-fixado ('_alt').

Note que o **caso de uso não é capaz** de **descrever todos** os requisitos (o que a lista de requisitos pretende). Entretanto, a leitura do **caso de uso** pode ser mais **rica e esclarecedora** do que capturar requisitos em uma lista que não expressa contexto.

Histórias de usuários

Em contraponto à abordagem clássica, temos alguns **artefatos** utilizados por **metodologias** de desenvolvimento **ágil**, que contribuem significativamente para a engenharia de requisitos.

As principais **contribuições** são a maior **velocidade** para levantar requisitos essenciais e a **inclusão** da **participação** de **clientes** e **usuários** nesse processo.

O **planejamento** e o **projeto** de uma **iteração** no desenvolvimento **ágil** podem ser **baseados** nas **histórias de usuários**.

Na prática, as **histórias de usuários** **não** necessariamente **precisam** ser **traduzidas** em itens de uma **lista de requisitos**.



Histórias de usuários

Planejamento

A atividade de **planejamento** (também chamada de o **jogo do planejamento**) se **inicia** com **ouvir**: uma atividade de **levantamento de requisitos** que capacita os membros técnicos da equipe a **entender** o **ambiente** de **negócios** do software e permite obter uma **percepção** ampla sobre os **resultados** solicitados, **fatores principais** e **funcionalidade**.

A **atividade** de **ouvir** conduz à **criação** de um conjunto de “**histórias**” (também denominadas **histórias de usuários**) que **descreve** o **resultado**, as **características** e a **funcionalidade** solicitados para o software a ser construído.

Cada **história** (similar aos casos de uso descritos) é **escrita pelo cliente** e é colocada em uma ficha. O **cliente atribui** um **valor** (uma **prioridade**) à história **baseando-se** no **valor de negócio** global do recurso ou função.

Os membros da **equipe** avaliam cada história e **atribuem** um **custo** a ela (medido em semanas de desenvolvimento). Se a história exigir, por estimativa, **mais do que três semanas** de desenvolvimento, é solicitado ao cliente que ele a **divida** em **histórias menores**, e a atribuição de valor e custo ocorre novamente. É importante notar que **podem ser escritas novas histórias a qualquer momento**.



Histórias de usuários

Planejamento

Clientes e desenvolvedores trabalham juntos para decidir como **agrupar histórias** para a **versão seguinte** (o próximo incremento de software) a ser desenvolvida pela equipe.

Conseguindo chegar a um **compromisso básico** (concordância sobre **quais histórias serão incluídas, data** de entrega e outras questões de **projeto**) para uma versão, a **equipe ordena** as histórias a serem desenvolvidas em uma de três formas:

1. **Todas** serão **implementadas** imediatamente (em um prazo de poucas semanas);
2. **As histórias de maior valor** serão deslocadas para **cima** no cronograma e implementadas primeiro
3. **As histórias de maior risco** serão deslocadas para **cima** no cronograma e implementadas primeiro.



Histórias de usuários

Planejamento

Depois de a **primeira versão** do projeto ter sido entregue, a **equipe calcula a velocidade do projeto**.

A **velocidade do projeto é o número de histórias de clientes implementadas durante a primeira versão**.

Assim, a velocidade do projeto pode ser utilizada para:

- **Ajudar a estimar as datas de entrega** e o cronograma para versões subsequentes
- **Determinar se foi assumido um compromisso exagerado** para todas as histórias ao longo de todo o projeto de desenvolvimento. **Se** ocorrer um **exagero**, o **conteúdo das versões é modificado** – ou as **datas finais de entrega são alteradas**.

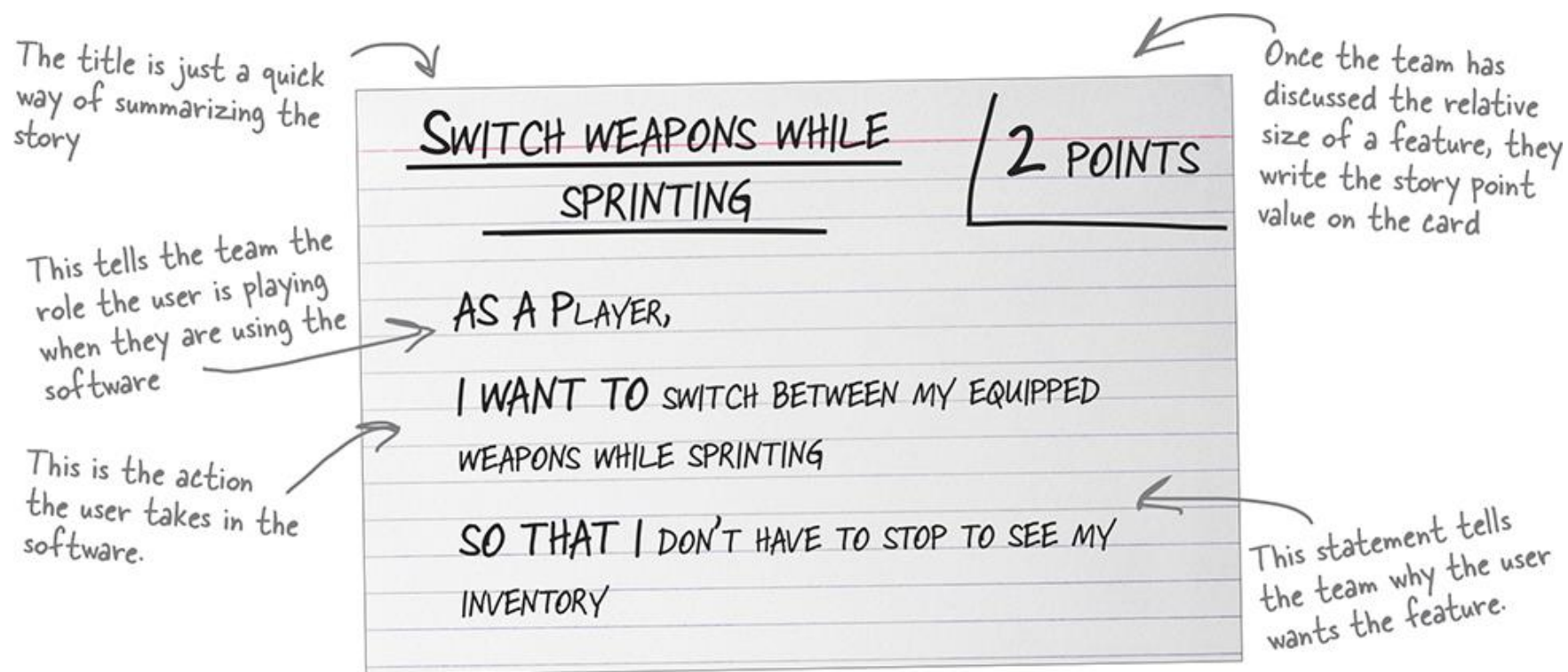
Conforme o trabalho de desenvolvimento prossegue, o **cliente** pode **acrescentar histórias, mudar o valor** de uma já existente, **dividir** algumas ou **eliminá-las**. Em seguida, a **equipe reconsidera** todas as versões remanescentes e **modifica** seus **planos** de forma correspondente.



Histórias de usuários

Planejamento

Exemplo de história de usuário



Histórias de usuários

Planejamento

Exemplo de história de usuário

Acompanhamento de atividades online

No papel de Professor

Eu gostaria que os alunos pudessem publicar suas tarefas online

Assim eu poderia verificar as atividades com maior agilidade

Critério de aceitação

O sistema deve permitir que os alunos possam realizar o upload das atividades

Histórias de usuários

Planejamento

Exemplo de história de usuário

Título

No papel de

Eu gostaria

Assim

Critério de aceitação

desc

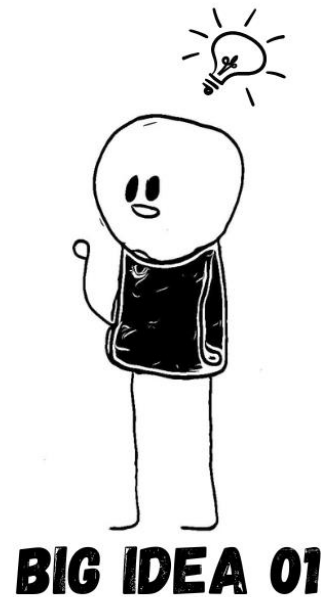
Outros requisitos

Casos de uso e histórias de usuários não são as únicas formas de **descrever requisitos** ou identificá-los.

Esses artefatos **descrevem requisitos funcionais**, sendo parte do problema do projeto.

A **coleta colaborativa** de requisitos **possibilita identificar requisitos não funcionais** e 'grandes ideias' do projeto.

Existem **outros artefatos** de requisitos que são úteis para descrever outros elementos do sistema, além de suas funcionalidades.



Outros requisitos

- **Especificação Suplementar:** capta e identifica outros tipos de requisitos. Por exemplo, relatórios, documentação, empacotamento, capacidade de suporte, licenciamento, etc.
- **Glossário:** contém termos e definições; também pode desempenhar o papel de um dicionário de dados.
- **Visão:** resume a visão do projeto; um resumo executivo. Serve para comunicar as grandes ideias.
- **Regras de negócio:** captam regras ou políticas de longo prazo e abrangentes, tais como leis fiscais que transcendem uma aplicação particular.



Especificação suplementar

A **especificação suplementar** é ainda um **documento de texto**. Basicamente são descritos **requisitos não funcionais**:

- Segurança;
- Usabilidade;
- Confiabilidade;
- Manutenibilidade;
- Restrições de implementação;
- Componentes comprados;
- Componentes livres com código-fonte aberto;
- Interfaces;
- Relatórios;
- Regras de negócio específicas da aplicação.
- entre outros.



Os **requisitos funcionais** também **podem estar presentes** nesse artefato, na seção 'Funcionalidade'.

Veja um exemplo, contendo requisitos de segurança, usabilidade e confiabilidade.

Exemplo básico de especificação suplementar: <ex_especificacao_suplementar.docx>

Regras de negócio

As **regras de negócio** são parte **importante** para a implementação de software.

Todo **sistema** tem um objetivo e, geralmente, esse **objetivo** é **atingido** seguindo algumas **regras**.

Sistemas de informação auxiliam a tomada de decisão. Para isso as **regras de negócio** devem estar **corretamente implementadas** para permitir a boa tomada de decisão.

Em termos de artefatos, as regras de negócio **podem estar dentro** das **especificações suplementares** em uma seção específica, principalmente quando essas regras forem específicas da aplicação.

Podem, ainda, ser **descritos** em um **documento separado**. Geralmente, quando existem muitas regras de negócio 'externas' que devem ser seguidas.

Para facilitar o entendimento, as regras de negócio **podem estar dispostas** em uma **tabela**. Veja um exemplo.



Regras de negócio

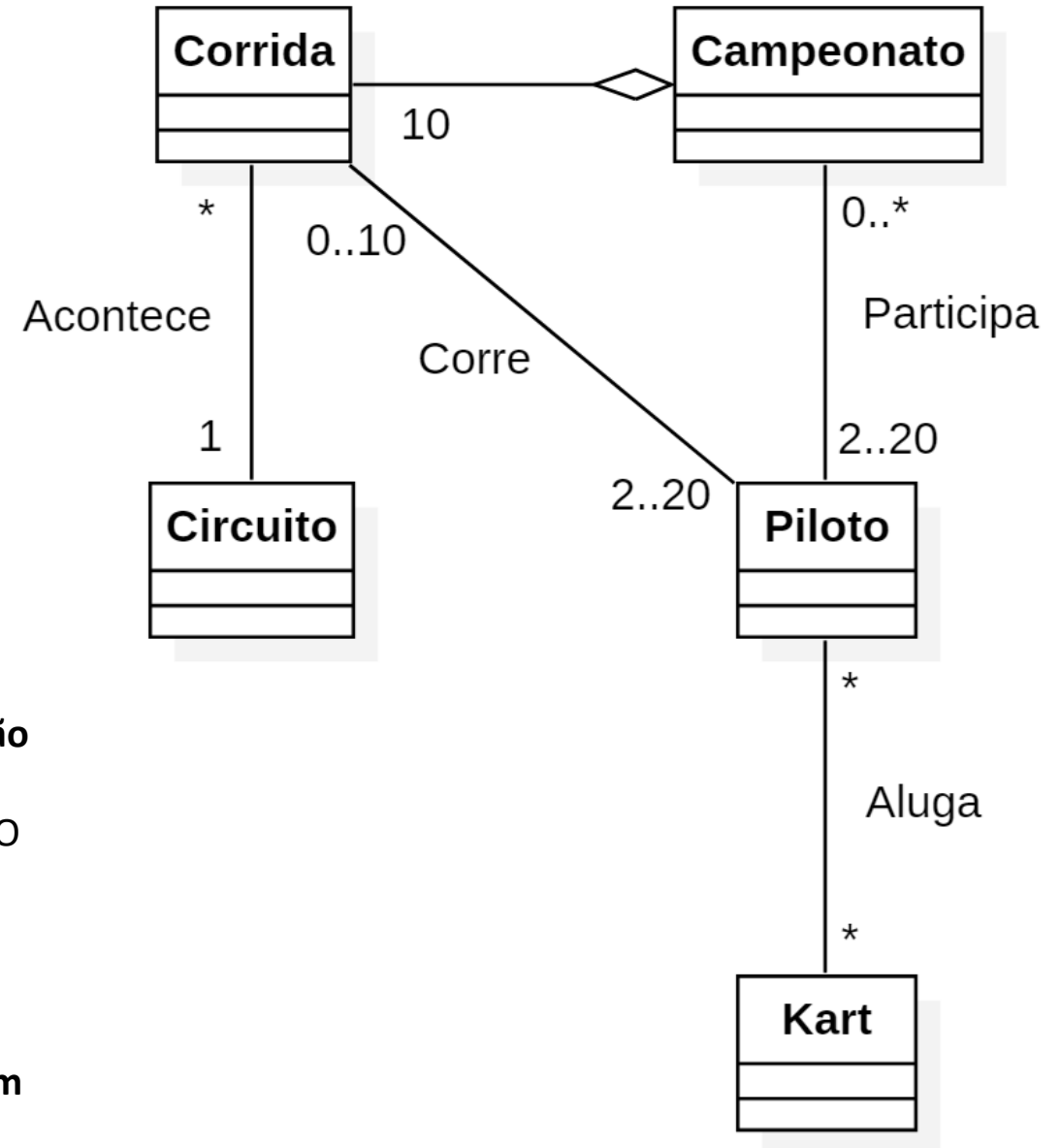
ID	Regra	Mutabilidade	Fonte
1	Regras de desconto para compradores: Empregado - 20% Cliente Preferencial - 10% Idoso - 15%	Alta. Cada varejista usa regras diferentes.	Política de varejo.
2	Regras de desconto de venda (em nível de transação). Aplica-se ao total antes do imposto. Exemplos: 10% se o total for maior do que R\$100,00. 5% toda segunda-feira. 10% para todas as vendas de hoje das 10:00h às 15:00h. Café: 50% hoje das 9:00h às 10:00h	Alta. Cada varejista usa regras diferentes e elas podem mudar diariamente ou de hora em hora.	Política de varejo.
3	Regras de desconto por produto (nível de linha de item). Exemplos: 10% de desconto para tratores nesta semana. Compre 2 hambúrgueres vegetarianos e leve um de graça.	Alta. Cada varejista usa regras diferentes e elas podem mudar diariamente ou de hora em hora.	Política de varejo.

Modelagem de Domínio

Uma vez que o **modelo de domínio representa** conceitos que envolvem **regras de negócio e associações entre objetos do domínio**, este artefato também **pode ser uma fonte de requisitos**.

Veja um exemplo de modelo de domínio no contexto de competições amadoras de kart. Quais requisitos poderiam ser identificados?

- Um piloto pode participar de campeonatos de kart. **O sistema deve permitir a inscrição de um piloto em um campeonato.**
- Um campeonato é composto por 10 corridas. **O sistema deve permitir que o administrador de um campeonato defina quais serão as corridas, as datas e o circuito.**
- O piloto deve alugar um kart para poder participar de uma corrida. O aluguel acontece por corrida; assim, o piloto pode alugar karts diferentes ao longo do campeonato. **O sistema deve permitir ao piloto solicitar um aluguel e realizar o pagamento antecipado.**
- O piloto participante de um campeonato pode correr em até 10 corridas. **O sistema deve considerar uma lista de pilotos que correm em uma corrida, registrando seus tempos por volta e as posições finais.**



Projeto

TG3 - Levantamento e especificação de requisitos

Levantamento de requisitos:

1. Utilize os artefatos produzidos até agora para realizar o levantamento dos principais requisitos do sistema proposto;
2. Vocês podem realizar uma coleta colaborativa entre os componentes do grupo. Experimente fazer uma rodada sozinhos, levem seus requisitos para a reunião e definam os requisitos de forma colaborativa.

Especificação de requisitos:

1. Os requisitos do sistema devem ser formalizados em uma especificação de requisitos;
2. O template dos requisitos suplementares pode ser utilizado para construir a especificação;
3. Crie uma seção para descrever os requisitos funcionais. Esses requisitos podem ser descritos utilizando casos de uso.
4. Crie seções para descrever os requisitos não funcionais;
5. Crie uma seção para descrever regras de negócio.

OBS: Todos os itens devem estar disponibilizados em uma única página da Wiki do projeto.

Referências

- LARMAN, C. Utilizando UML e padrões. 3ª ed. Porto Alegre: Bookman, 2007.
- GAMMA, E et al. Padrões de Projeto. Porto Alegre: Bookman, 2005.
- BEZERRA, E. Princípios de análise e projeto de sistemas com UML. 2ª ed. Rio de Janeiro: Elsevier Campus. 2007.
- Anotações e materiais da Professora Ana Claudia Rossi.