

CPRE

Certified Professional for Requirements Engineering

Handbook

Requirements Elicitation

Practitioner | Specialist

Anja Brand, Dominik Häußer
Kim Lauenroth, Hans van Loenhoud,
Patrick Steiger

Termos de uso

Este Handbook, incluindo todas as suas partes, é protegido pela lei de direitos autorais. Com o consentimento dos proprietários dos direitos autorais e seguindo a lei de direitos autorais, o uso do Handbook é permitido — a menos que seja explicitamente mencionado que não é permitido. Isto se aplica em particular às reproduções, adaptações, traduções, microfilmagem, armazenamento e processamento em sistemas eletrônicos, e divulgação pública.

Os provedores de treinamento podem utilizar este Handbook como base para seminários e treinamentos, desde que o detentor dos direitos autorais seja reconhecido e que a fonte e o proprietário dos direitos autorais sejam mencionados. Além disso, com o consentimento prévio da IREB, este Handbook pode ser utilizado para fins publicitários.

Qualquer indivíduo ou grupo de indivíduos pode usar este Handbook como base para estudo, artigos, livros ou outras publicações derivadas desde que o detentor dos direitos autorais seja reconhecido e a fonte e o proprietário dos direitos autorais sejam mencionados.

Agradecimentos

Este handbook foi escrito por Dominik Häußler, Kim Lauenroth, Hans van Loenhoud, Anja Brand e Patrick Steiger.

Resenha de Birgit Penzenstadler. Tradução na Língua Portuguesa por: Ana Moreira, Carlos Silva, George Fialkovitz Jr. e Guilherme Siqueira Simões.

Aprovado para publicação em 8 de outubro de 2018 pelo Conselho do IREB por recomendação de Thorsten Weyer.

Agradecemos a todos pelo seu envolvimento.

Copyright © 2017–2022 para este Handbook pertence aos autores listados anteriormente. Os direitos foram transferidos para o IREB International Requirements Engineering Board e.V.

A compilação deste handbook contou com o apoio de



Prefácio

Este *Handbook* complementa o syllabus do módulo CPRE Requirements Elicitation.

Este Handbook destina-se aos provedores de treinamento que desejam oferecer seminários ou treinamentos sobre o CPRE Requirements Elicitation Practitioner e/ou Specialist de acordo com o padrão IREB. Também se destina ao treinamento de participantes e interessados que desejam obter uma visão detalhada sobre o conteúdo deste módulo.

Este Handbook não substitui o treinamento sobre o tema. O Handbook representa uma ligação entre o Syllabus (que lista e explica os objetivos de aprendizagem do módulo) e a ampla gama de literatura que foi publicada sobre o tema.

O conteúdo deste Handbook, juntamente com referências a literatura mais detalhada, apoia os prestadores de treinamento na preparação dos participantes do treinamento para o

exame de certificação. Este Handbook oferece aos participantes do treinamento e aos interessados uma oportunidade de aprofundar seus conhecimentos de Engenharia de Requisitos em um ambiente Ágil e de complementar o conteúdo detalhado com base nas recomendações da literatura. Além disso, este Handbook pode ser usado para atualizar o conhecimento existente sobre os vários tópicos da elicitação de requisitos, por exemplo, depois de ter recebido o certificado de Requirements Elicitation Practitioner ou Specialist.

Sugestões de melhorias e correções são sempre bem-vindas!

Contato por e-mail: info@ireb.org

Esperamos que você goste de estudar este Handbook e que seja aprovado no exame de certificação do IREB CPRE Requirements Elicitation Practitioner ou Specialist.

Mais informações sobre o IREB CPRE Requirements Elicitation podem ser encontradas em: <http://www.ireb.org>.

Histórico das Versões

Versão	Data	Comentário	Autor
2.1.0	11 de dezembro de 2025	Versão inicial	Ana Moreira, Carlos Silva George Fialkovitz Jr. Guilherme Siqueira Simões

Sumário

1	Um framework para estruturar e gerenciar a elicitação de requisitos	8
1.1	O escopo da elicitação na Engenharia de Requisitos	8
1.2	Fatores relevantes para a abordagem do planejamento da elicitação ..	9
1.3	Planejamento e execução da elicitação de requisitos	10
1.3.1	Atividade de elicitação	11
1.3.2	Atividade de resolução de conflitos	16
1.3.3	Diretrizes para a parte de elicitação de um projeto	18
1.4	Padrões de processo	26
1.4.1	Estrutura e benefícios dos padrões de processo para elicitação de requisitos	26
1.4.2	Waterfall	28
1.4.3	Design centrado no ser humano	34
1.4.4	Design Thinking	39
2	Fontes de Requisitos	45
2.1	Fundamentos de fontes de requisitos	45
2.2	Identificar, classificar e gerenciar os stakeholders	48
2.2.1	Identificação e seleção de stakeholders como fontes de requisitos	49
2.2.2	Gerenciamento do relacionamento com os stakeholders	52
2.2.3	Esquema de documentação para os stakeholders envolvidas	55
2.2.4	O usuário como um grupo especial de stakeholders	56
2.3	Identificar, classificar e gerenciar documentos	57
2.3.1	Identificação e seleção de documentos como fontes de requisitos	57
2.3.2	Esquema de documentação para documentos	61
2.4	Identificar, classificar e gerenciar sistemas	61
2.4.1	Identificação e seleção de sistemas como fontes de requisitos	62
2.4.2	Esquema de documentação para sistemas	64
3	Elicitação	66
3.1	Técnicas de coleta	68

3.1.1	Técnicas de questionamento	68
3.1.2	Técnicas de observação	79
3.1.3	Técnicas de colaboração	90
3.1.4	Técnicas baseadas em artefatos	98
3.2	Técnicas de design e de geração de ideias (L2)	106
3.2.1	Brainstorming	108
3.2.2	Técnica de analogia	110
3.2.3	Protótipos	111
3.2.4	Cenários e Storyboards	116
3.3	Ferramentas de pensamento	119
3.3.1	Pensar em níveis de abstração	119
3.3.2	Pensar em termos de problemas e metas	121
3.3.3	Evitar efeitos de transformação	124
3.3.4	Pensando em termos de modelos	130
3.3.5	Mapeamento mental	132
3.4	Descrição das técnicas de elicitação por atributos	134
4	Resolver conflitos	143
4.1	Identificar conflitos	144
4.2	Analisar conflitos	145
4.2.1	As características de um conflito de requisitos	146
4.2.2	Os tipos de conflito de Moore	147
4.3	Resolver conflitos	152
4.3.1	Acordo	152
4.3.2	Compromisso	153
4.3.3	Votação	155
4.3.4	Análise de alternativas	156
4.3.5	Imposição	158
4.3.6	Técnicas Auxiliares	159
4.3.7	Encontrar uma técnica adequada de resolução de conflitos	160
4.4	Documentar a resolução de conflitos	164
5	Habilidades do Engenheiro de Requisitos	165
5.1	Habilidades necessárias nas áreas de elicitação	165

5.2	Teoria da comunicação e modelos de comunicação	166
5.3	Autorreflexão sobre as habilidades pessoais em elicitação de requisitos.....	170
5.4	Oportunidades de desenvolvimento pessoal	173
5.5	Aprender com a experiência anterior - aprendizado contínuo	174
6	Referências e leituras adicionais	176

1 Um framework para estruturar e gerenciar a elicitação de requisitos

Nos últimos anos, os sistemas de TI se tornaram essenciais para o funcionamento das empresas, do governo e da própria sociedade. Portanto, a alta qualidade desses sistemas é essencial. Os profissionais de TI aprenderam que a qualidade de um sistema de TI é determinada principalmente por seus requisitos.

A partir dessa percepção, surgiu uma profissão de TI totalmente nova: Engenharia de Requisitos. A ideia principal é compartilhar informações. A Engenharia de Requisitos, como disciplina, está preocupada em obter, documentar, validar, negociar e gerenciar todas as informações de que os desenvolvedores e operadores de sistemas precisam para criar, operar e manter sistemas bem-sucedidos.

A Engenharia de Requisitos ajuda todas as partes a entender que tipo de sistema é realmente necessário. Em certos contextos, a Engenharia de Requisitos é realizada por uma função dedicada: "Engenheiro de Requisitos". Em outros contextos, a Engenharia de Requisitos faz parte de uma definição de função mais ampla, por exemplo: Engenheiro de Sistemas [Walten et al.2015] ou Designer Digital [Bitkom2017]. Por motivos de simplicidade, este handbook usará o termo Engenheiro de Requisitos.

1.1 O escopo da elicitação na Engenharia de Requisitos

De acordo com a definição de Engenharia de Requisitos apresentada em [PoRu2015], o objetivo da elicitação de requisitos e da resolução de conflitos é "conhecer os requisitos relevantes", "obter um consenso entre os stakeholders sobre esses requisitos" e "entender [...] os desejos e as necessidades dos stakeholders".

Na elicitação, é tarefa do Engenheiro de Requisitos compreender os desejos e necessidades dos stakeholders, garantindo ao mesmo tempo que os requisitos de todas as fontes de requisitos relevantes foram coletadas. Isso inclui a identificação dessas fontes, a compreensão da natureza e da importância dos diferentes tipos de requisitos e a aplicação de técnicas apropriadas para obtê-los. Um ponto importante na elicitação é transformar demandas, desejos e expectativas implícitas em requisitos explícitos [ISO29148].

Durante a elicitação, é comum encontrar requisitos conflitantes de diferentes fontes. Esses conflitos precisam ser resolvidos para criar um conjunto único, consistente e acordado que possa servir de insumo para o desenvolvimento, a manutenção e a operação eficientes de um sistema eficaz.

Este documento descreve a *elicitação* e a *resolução de conflitos* para o Requirements Elicitation Practitioner e Specialist. Este primeiro capítulo serve como uma introdução ao assunto e como um guia para sua aplicação prática. No **Capítulo 2**, são descritas as **Fontes de Requisitos**. Determinar quais fontes são relevantes é o ponto de partida para todo esforço de elicitação. O **Capítulo 3 Elicitação** apresenta uma visão geral das técnicas que podem ser usadas para elicitar requisitos, bem como orientações sobre como usá-las.

O **Capítulo 4 a Resolução de Conflitos** trata de maneiras de resolver situações em que os requisitos são conflitantes entre si. O Handbook termina com o **Capítulo 5 Habilidades do Engenheiro de Requisitos**, que inclui pontos focais para profissionais que desejam atuar nesse domínio.

De acordo com o syllabus IREB CPRE Foundation Level [IREB2020], a Engenharia de Requisitos tem três outras atividades principais: *documentação*, *validação* e *gerenciamento*. A documentação diz respeito a maneiras de capturar os resultados do processo de elicitação como um meio de comunicação posterior. Um Handbook separado de Requirements Modeling, de acordo com o padrão IREB [CHQW2016], fornece mais informações sobre uma parte desse tópico. O gerenciamento consiste em manter um conjunto de requisitos coletados, documentados e consolidados em bom estado durante todo o seu ciclo de vida. Essa atividade é descrita em mais detalhes no Handbook do Requirements Management de acordo com o padrão IREB [BuHe2019].

1.2 Fatores relevantes para a abordagem do planejamento da elicitação

A literatura sobre estimativa de projetos de software [McCo2006] e os resultados da prática industrial atribuem à disciplina de Engenharia de Requisitos uma grande responsabilidade pelo cumprimento das expectativas gerais do projeto. Do ponto de vista da Engenharia de Requisitos, uma parte significativa dessa responsabilidade deve ser atribuída à elicitação de requisitos. Isso requer uma abordagem de planejamento específica pelos seguintes motivos:

1. A elicitação de requisitos não pode ser planejada apenas com base no tamanho esperado do resultado. Não é possível afirmar que queremos obter 107 requisitos e que precisaremos de uma média de 1,25 horas para cada requisito. A razão para isso é simples: não sabemos o tamanho e a forma dos resultados da elicitação. Temos que obter os requisitos porque não os conhecemos.
2. Embora os conflitos de requisitos em um projeto não possam ser planejados ou previstos, em todos os projetos haverá tais conflitos. Quando um conflito de requisitos ocorre e é detectado, o Engenheiro de Requisitos precisa reagir a ele.

Ambos podem levar ao mal-entendido de que não é possível programar e controlar a elicitação de requisitos por meio de técnicas de gerenciamento de projetos. É verdade que nem sempre é aconselhável definir um plano detalhado e antecipado para a elicitação de requisitos (incluindo as técnicas de elicitação selecionadas, um orçamento detalhado e um cronograma). A ênfase está na palavra-chave "upfront", porque um plano detalhado e antecipado deve se basear em suposições (lembre-se de que fazemos elicitações porque não sabemos tudo), e essas suposições geralmente são invalidadas logo após o início do projeto. Portanto, um plano detalhado só é aconselhável se houver conhecimento prévio suficiente (p. ex., sobre a estrutura do sistema pretendido) ou confiança suficiente nas suposições subjacentes (p. ex., quais aspectos do novo sistema serão importantes).

O planejamento e a execução das atividades de elicitação são muito semelhantes ao planejamento e à execução de um projeto de pesquisa. Um projeto de pesquisa normalmente começa com uma ou mais perguntas de pesquisa (ou definições de problemas) e define uma sequência de atividades para responder e detalhar as perguntas de pesquisa definidas. A semelhança entre uma atividade de elicitação e um projeto de pesquisa é que o início de ambas as atividades é caracterizado por incertezas e suposições (ou uma hipótese). Portanto, o projeto de pesquisa não pode ser planejado completamente do início ao fim. Em vez disso, um projeto de pesquisa define atividades que abordam questões de pesquisa selecionadas, para esclarecer incertezas ou para validar (ou falsificar) as suposições (ou hipóteses) no decorrer do projeto. Isso significa, em particular, que o plano de pesquisa e as perguntas de pesquisa devem ser revisados, refinados e atualizados continuamente, com base nas novas descobertas.

O mesmo se aplica à elicitação de requisitos. As atividades de elicitação geralmente seguem uma abordagem exploratória. No início, o plano de elicitação define determinados objetivos ("perguntas de pesquisa" a serem respondidas), incluindo um cronograma geral e critérios de saída de alto nível. Em uma série de iterações consecutivas, as respostas a essas perguntas são encontradas e refinadas, e as próximas etapas são tomadas com base nessas respostas até que os stakeholders aceitem os requisitos resultantes como adequados. Os processos modernos de desenvolvimento de software e as metodologias ágeis apoiam uma abordagem iterativa que consiste em ciclos curtos nos quais são produzidas soluções variantes e o feedback é incorporado.

Leituras recomendadas

[Beveridge 1957] fornece uma boa introdução à definição de projetos de pesquisa. O texto completo desse livro está disponível em www.archive.org.

1.3 Planejamento e execução da elicitação de requisitos

Como vimos acima, a elicitação de requisitos exige uma abordagem de planejamento específica. Nesta seção, descrevemos uma estrutura para planejar e executar atividades de elicitação. O principal objetivo dessa estrutura é didático: apoiar o ensino e o treinamento. A estrutura torna explícitas as etapas e as informações necessárias para o planejamento e a execução da elicitação. A estrutura não deve ser entendida como um kit de ferramentas de processo que possa ser aplicado diretamente; como ocorre com outras estruturas de processo, a aplicação prática exige uma adaptação da estrutura à situação em questão.

Por fim, o planejamento e a execução de um esforço de elicitação não podem ser tratados isoladamente de outras atividades em um projeto de desenvolvimento de sistema. A adaptação da abordagem a um projeto específico requer uma compreensão profunda do contexto do projeto, levando em consideração produtos de trabalho como a declaração de visão do produto, o resumo do projeto e o caso de negócios e, portanto, não é considerada neste handbook. Em vez disso, a discussão a seguir se concentra nos conceitos centrais da elicitação.

Para a definição de nossa estrutura, presumimos que todo projeto que inclui atividades de elicitação usa algum tipo de plano para estruturar a abordagem ou as tarefas. Isso pode ser um plano de projeto sofisticado, incluindo marcos, ou um backlog ágil.

We define two activities that can be included in any kind of plan:

- Atividades de elicitação para identificar as fontes de requisitos e capturar seus requisitos
- Atividades de resolução para solucionar quaisquer conflitos de requisitos que possam ocorrer

Além disso, ambas as atividades podem fornecer informações de gerenciamento de projetos relacionadas a prazos e recursos. Os detalhes sobre a definição dessas atividades de elicitação são fornecidos nas subseções a seguir.

1.3.1 Atividade de elicitação

Uma atividade de elicitação é usada para planejar a elicitação de requisitos ou a identificação de fontes de requisitos. O conteúdo de uma atividade de elicitação é descrito por cinco elementos: objetivo da elicitação, qualidade do resultado, fonte de requisitos, técnica de elicitação e informações de gerenciamento do projeto.

Usamos o exemplo de uma atividade de elicitação em um projeto em que os engenheiros de serviço de uma empresa mundial de serviços de motores de navios devem receber suporte de dispositivos móveis:

ID	RS_EA_13
Objetivo da elicitação	Determinar a diversidade (em qualquer aspecto relevante) entre o grupo de usuários "Engenheiro de Serviços"
Qualidade do resultado	Personas válidas (uma se forem homogêneas, várias se forem heterogêneas) para o grupo de usuários "Service Engineer"
Fontes de requisitos	Engenheiros de serviço em 5 estações de serviço selecionadas em todo o mundo: Hamburgo, Cidade do Cabo, Buenos Aires, Dubai, Osaka
Técnica de elicitação	Pesquisa contextual

Figura 1: Exemplo de uma atividade de elicitação (ID é a informação de gerenciamento do projeto)

Esses cinco elementos serão descritos na subseção a seguir. Esses elementos têm fortes relações entre si. A compreensão dessas relações ajuda a definir boas atividades de elicitação. A descrição dos relacionamentos é apresentada no final desta subseção.

1.3.1.1 Objetivo da elicitação

O *objetivo da elicitação* é o conceito central da abordagem de planejamento, pois orienta todos os outros elementos. O mesmo objetivo de elicitação pode ocorrer várias vezes em um plano de projeto de elicitação, por exemplo, se quisermos responder a uma determinada pergunta com técnicas diferentes ou analisando várias fontes de requisitos.

O objetivo da elicitação deve ser formulado da forma mais precisa possível. Ele serve para os seguintes propósitos:

1. Ele caracteriza o que queremos aprender ou entender com essa atividade de elicitação específica, ou seja, as fontes de requisitos a serem identificadas ou os requisitos a serem elicitados.
2. Ele apoia a identificação de fontes de requisitos apropriadas (consulte a Seção 2.2).
3. Ele apóia a seleção de uma técnica de elicitação.
4. Ele pode ser usado para medir o sucesso da atividade (o objetivo foi alcançado ao final da atividade?).
5. A qualidade do resultado do objetivo da elicitação é um indicador importante do nível de compreensão do sistema a ser desenvolvido (consulte a Seção 1.3.2.2).

Há várias maneiras diferentes de formular objetivos de elicitação, por exemplo:

1. Formule perguntas reais sobre os requisitos do sistema¹.

Exemplo: Quais são as principais etapas do processo de negócios que devem ser suportadas pelo novo sistema de CRM (gerenciamento de relacionamento com o cliente) e como é esse suporte?

2. Formule objetivos de elicitação como hipóteses que você confirmará ou rejeitará.

Exemplo: O aplicativo para smartphone do sistema CRM deve oferecer funcionalidades para adicionar novos clientes. Não é suficiente ter essa funcionalidade apenas na parte do desktop.

3. Formule objetivos de elicitação com base no Modelo Kano, usando as categorias de básicos, de performance e atrativos (consulte [IREB2020]).

Exemplo: Entenda os fatores básicos do sistema CRM existente.

¹ [Mill2009] fornece uma longa lista de perguntas relacionadas a requisitos não funcionais. [Withall2007] fornece uma lista de padrões de requisitos que podem ser usados para derivar perguntas.

4. Utilize padrões ou modelos de especificações em seu domínio para definir os objetivos da elicitação.

Exemplo: Identifique os requisitos de qualidade de acordo com [ISO25010], como requisitos de usabilidade, confiabilidade ou segurança.

1.3.1.2 Qualidade do resultado

A *qualidade do resultado* descreve a qualidade pretendida do resultado da atividade em termos de nível de certeza, integridade e concordância.

Certeza refere-se ao grau de evidência que pode ser dado para a exatidão do resultado (ou seja, fontes de requisitos ou requisitos).

Com completude, queremos dizer a cobertura do resultado com relação à quantidade "teoricamente possível" de informações que poderiam ter sido obtidas no nível de detalhe desejado.

Acordo refere-se ao grau em que o resultado deve ser aceito pelos stakeholders. Também abrange os stakeholders que precisam concordar. Lembre-se de que chegar a um acordo entre os stakeholders pode exigir muita comunicação e, portanto, pode gerar um esforço significativo.

O Engenheiro de Requisitos deve sempre buscar um equilíbrio entre o custo de atividades adicionais de elicitação e os benefícios para o projeto atual. O nível certo de certeza e integridade resulta de uma interação contínua entre a ER e os stakeholders relevantes.

1.3.1.3 Fontes de requisitos

A *fonte* caracteriza a fonte ou fontes das quais os requisitos devem ser extraídos ou as fontes que são usadas para identificar outras fontes de requisitos. Estipulamos que uma atividade de elicitação se concentre em exatamente um tipo de fonte.

1.3.1.4 Técnica de elicitação

A *técnica de elicitação* é a técnica específica usada para elicitar os requisitos da fonte.

As informações relevantes para uma técnica de elicitação incluem:

- **Esforço estimado para preparação/execução/pós-processamento:** Uma técnica de elicitação pode exigir um esforço significativo de preparação, execução e pós-processamento para o Engenheiro de Requisitos e/ou stakeholders. Por exemplo, uma entrevista exige a preparação das perguntas. O aprendizado pode levar vários dias, tanto para o Engenheiro de Requisitos quanto para os stakeholders afetados, dependendo da complexidade do projeto. Um workshop pode exigir a análise de seus resultados. Esse esforço deve ser estimado e documentado para melhorar o planejamento das técnicas de elicitação.

- **Esforço despendido na preparação/execução/pós-processamento:** Além do esforço estimado, o esforço real gasto em uma técnica de elicitacão deve ser documentado. O desvio significativo entre o esforço estimado e o esforço gasto deve ser analisado para entender os motivos do desvio.
- **Cronograma de preparação/execução/pós-processamento:** Além do esforço necessário, o cronograma da atividade de elicitacão é importante para os Engenheiros de Requisitos e os stakeholders, especialmente para as atividades que exigem um esforço significativo. O cronograma também pode fornecer um prazo.
- **Referência ao material de preparacão:** Se uma técnica exigir material de preparacão (p. ex., um workshop que discute um modelo de interface do usuário), o material de preparacão deve ser referenciado.

A tarefa aqui é selecionar a(s) técnica(s) ideal(is) para o objetivo de elicitacão em questão. Toda técnica de elicitacão tem suas vantagens e desvantagens. Assim como ocorre com as fontes de requisitos, você não deve confiar em uma única técnica de elicitacão. Embora as entrevistas possam ajudá-lo a obter informacões detalhadas e específicas de participantes falantes, elas provavelmente não o ajudarão com pessoas introvertidas ou com alguém que tenha medo de dar a resposta "errada".

A seçã 3.4 fornece orientacões sobre como chegar a uma seleçã adequada de técnicas de elicitacão.

Lembre-se: defina atividades de elicitacão separadas para cada técnica!

1.3.1.5 Informacões sobre gerenciamento de projetos

Uma atividade de elicitacão pode ser caracterizada por vários atributos de gerenciamento de projetos: o conjunto exato de atributos depende do contexto e do método específicos do projeto.

A lista a seguir fornece exemplos úteis de atributos de gerenciamento de projetos:

- **Autor:** A pessoa que definiu a atividade de elicitacão específica
- **Engenheiro(s) de requisitos responsável(is):** o(s) membro(s) do projeto responsável(is) pela execuçã da atividade de elicitacão
- **Prioridade:** A importãncia de uma atividade de elicitacão para o projeto geral
- **Dependências de outras atividades de elicitacão:** Uma atividade de elicitacão pode depender do resultado de outras atividades de elicitacão. Por exemplo, um workshop de requisitos pode precisar da contribuicão de entrevistas com os stakeholders.
- **Referências aos requisitos documentados:** Após a conclusã de uma atividade, os requisitos resultantes podem conter referências para manter a rastreabilidade entre os requisitos e a atividade de elicitacão.

1.3.1.6 Utilizar as relações entre os elementos de uma atividade de elicitação

A técnica de elicitação é a ferramenta para realizar a elicitação. A seleção da técnica adequada é fundamental para o sucesso de uma atividade de elicitação. No entanto, a decisão por uma técnica específica deve ser a última etapa na definição de uma atividade de elicitação.

A principal razão para isso é que cada aspecto de uma atividade de elicitação tem uma relação com os outros aspectos que podem ser usados para validar e aprimorar a atividade geral de elicitação. Figura 2 mostra os quatro aspectos e as seis relações entre eles:

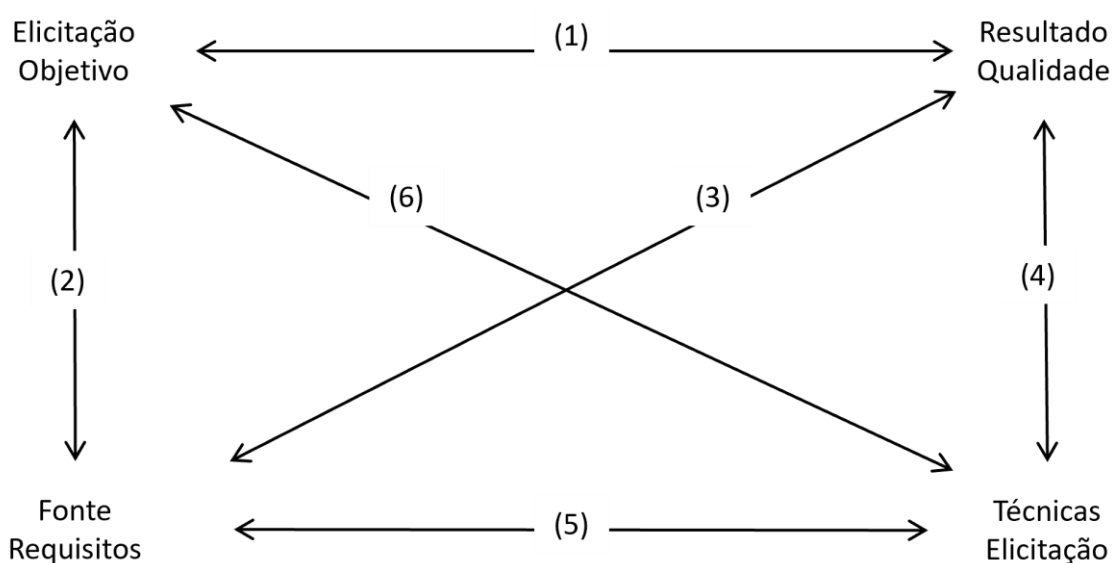


Figura 2: Relações entre os elementos das atividades de elicitação

1. **Objetivo da elicitação – Qualidade do resultado:** O objetivo da elicitação deve ser definido de forma que seja possível dar uma definição precisa (suficiente) para a certeza e a integridade desejadas do resultado.
2. **Objetivo da elicitação – Fonte de requisitos:** A fonte selecionada é útil para atingir o objetivo da elicitação? (Um documento de processo antigo pode não ser adequado para entender o processo de negócio real de uma empresa.)
3. **Qualidade do resultado – Fonte de requisitos:** A fonte selecionada pode fornecer a qualidade de resultado desejada? (É possível esclarecer determinados requisitos de apenas um stakeholder?)
4. **Qualidade do resultado – Técnica de elicitação:** A técnica selecionada pode fornecer a qualidade desejada? (É possível esclarecer certos requisitos em uma sessão de brainstorming?)
5. **Fonte de requisitos – Técnica de elicitação:** A técnica e a fonte devem ser compatíveis (não faz sentido entrevistar um documento!), e as fontes devem ser abordáveis pela técnica. (É possível reunir todos os seis vice-presidentes de uma empresa em um workshop de cinco dias?)

6. **Objetivo da elicitação – Técnica de elicitação:** A técnica selecionada é adequada para atingir o objetivo definido? (É possível entender um processo de negócio específico com uma técnica de criatividade?)

Essas seis relações mostram que uma técnica de elicitação adequada só pode ser selecionada se os outros três aspectos estiverem bem definidos.

1.3.2 Atividade de resolução de conflitos

Esse tipo de atividade é usado para resolver conflitos entre requisitos. Os requisitos são considerados conflitantes se não puderem ser implementados no mesmo sistema ao mesmo tempo. Ver Capítulo 4 para detalhes.

O conteúdo de uma atividade de resolução é descrito por quatro elementos: descrição dos requisitos envolvidos, fontes dos requisitos envolvidos, técnica de resolução e resultado alcançado (após a resolução do conflito).

A atividade de resolução de conflitos é descrita assim que o conflito é identificado. Normalmente, nesse momento, nem todos os aspectos da atividade de resolução de conflitos podem ser descritos. Durante a resolução, os aspectos restantes são documentados.

ID	RS_CRA_3
Requisitos envolvidos	RS_REQ_37 e RS_REQ_221 não são compatíveis. A RS_REQ_37 exige conexão sem fio, enquanto a REQ_221 exige conexão com fio. Tipo de conflito: ainda não está claro
Fontes de requisitos envolvidas	A contribuição da RS_REQ_37 foi da Sra. Highmore, O RS_REQ_221 tem origem na arqueologia do sistema legado.
Resolução técnica	Técnica preferida: acordo Planejar a resolução: Reunir-se com a Sra. Highmore e o Sr. Strong (system owner do sistema legado) e encontrar uma solução adequada.
Resultado da resolução alcançada	TBD (será documentado quando a solução for encontrada)

Figura 3: Exemplo de uma atividade de resolução de conflitos (ID é a informação de gerenciamento do projeto) no estado de identificação

1.3.2.1 Descrição dos requisitos envolvidos

A descrição nomeia os requisitos conflitantes e especifica porque esses requisitos são conflitantes. Ele deve descrever a necessidade de resolução de forma clara e com detalhes adequados, incluindo uma referência aos requisitos afetados. A descrição deve mencionar ainda o tipo de conflito (consulte a Seção 4.2).

1.3.2.2 Fontes de requisitos envolvidas

As fontes de requisitos envolvidas (p. ex., stakeholders) são os participantes que devem ser incluídos no processo de resolução.

Por exemplo, um requisito relacionado ao monitoramento da atividade do usuário por motivos de segurança está em conflito com um requisito de proteção de dados pessoais. Um especialista em segurança mencionou que o requisito de monitoramento e o requisito de proteção de dados são originários de uma lei específica do domínio. O especialista em segurança é um stakeholder nesse conflito, e o stakeholder para a lei específica do domínio pode ser o diretor de proteção de dados da empresa.

A identificação dos stakeholders envolvidos é uma parte importante da identificação de conflitos (consulte a Seção 4.1).

1.3.2.3 Resolução técnica

A abordagem de resolução define a abordagem pretendida para resolver o conflito. A abordagem de resolução geralmente consiste em uma descrição da técnica de resolução selecionada e em atividades de preparação adicionais.

Detalhes sobre a seleção da abordagem de resolução adequada são apresentados na Seção 0.

1.3.2.4 Resultado da resolução alcançada

Após a resolução, uma breve descrição do resultado alcançado é documentada como a última etapa da atividade de resolução. Esta breve descrição tem o objetivo de ser um resumo da resolução de conflitos para as pessoas que analisarem essas atividades em uma etapa posterior do projeto.

1.3.2.5 Informações sobre gerenciamento de projetos

Uma atividade de resolução pode ser caracterizada por vários atributos de gerenciamento de projetos. O conjunto exato de informações depende do contexto específico do projeto. A lista a seguir fornece itens úteis de informações de gerenciamento de projetos:

- **Autor:** A pessoa que definiu a atividade de resolução específica
- **Engenheiro(s) de requisitos responsável(is):** o(s) membro(s) do projeto responsável(is) pela execução da atividade de resolução

- **Prioridade:** A importância de uma atividade de resolução para o projeto geral
- **Referência à documentação detalhada do conflito:** Se um conflito estiver relacionado a questões complexas, deve-se fazer referência à documentação adicional que explica os detalhes por trás do conflito.
- **Referência ao material de preparação:** Se a resolução de conflitos exigir material de preparação (p. ex., uma descrição detalhada para os stakeholders), esse material deve ser mencionado.
- **Dependências de outras atividades:** Uma atividade de resolução pode depender do resultado de outras atividades de resolução ou elicitación de conflitos. Por exemplo, uma proposta de resolução de conflitos pode exigir informações de entrevistas com stakeholders.
- **Esforço estimado para preparação/execução/pós-processamento:** Uma técnica de resolução pode exigir um esforço significativo de preparação, execução ou pós-processamento para o Engenheiro de Requisitos e/ou os stakeholders.
- **Último momento em que o conflito deve ser resolvido:** Talvez não seja necessário resolver todos os conflitos imediatamente após sua identificação. Se puder ser adiada, o último ponto até o qual essa resolução pode ser deixada sob responsabilidade deve ser documentado.
- **Cronograma de preparação/execução/pós-processamento:** Além do esforço, o cronograma de uma atividade de resolução é importante para os Engenheiros de Requisitos e os stakeholders, especialmente para atividades que exigem esforço significativo.

1.3.3 Diretrizes para a parte de elicitación de um projeto

Na subseção anterior, definimos as atividades de elicitación; nesta subseção, apresentamos diretrizes para a aplicação das atividades de elicitación, desde o planejamento até a execução. Essas diretrizes são independentes de qualquer abordagem específica de gerenciamento de projetos (p. ex., Waterfall ou Ágil). O termo geral projeto pode se aplicar, por exemplo, a um projeto de ER puro, um projeto de análise de negócios ou um projeto de software.

Dica 1.3.1:

Se você estiver usando um quadro Kanban (físico ou digital) para o gerenciamento de tarefas, as atividades de elicitación são itens perfeitos para o quadro de tarefas. O mesmo se aplica aos gráficos de Gantt.

1.3.3.1 Distinguir diferentes conjuntos de atividades de elicitação

Recomendamos distinguir entre três conjuntos diferentes de atividades de elicitação:

- *Conjunto 1 – Atividades de elicitação executadas:* Esse conjunto contém todas as atividades de elicitação que foram executadas durante o projeto até o momento. Ele descreve o histórico da perspectiva de elicitação de seu projeto e serve como uma memória do projeto. No início, esse conjunto estará vazio, é claro.
- *Conjunto 2 – Atividades de elicitação de curto prazo:* Esse conjunto contém todas as atividades de elicitação que estão planejadas para execução em breve. As atividades de elicitação nesse conjunto precisam ser planejadas detalhadamente e devem ser programadas e preparadas para execução. Você pode considerar esse conjunto como a lista de tarefas para o futuro próximo de seu projeto.
- *Conjunto 3 – Atividades de elicitação de longo prazo:* Esse conjunto contém todas as atividades de elicitação que são consideradas importantes, mas que ainda não foram planejadas e programadas em detalhes. A razão para definir algumas atividades como de longo prazo é que o status atual do projeto (especialmente o conhecimento e as suposições existentes) nem sempre permite um planejamento detalhado. Você pode considerar esse conjunto como uma lista de pendências para os objetivos que ainda precisam ser mais elaborados.

À medida que o projeto avança, o conjunto de atividades executadas aumenta, pois as atividades de curto prazo são executadas. As atividades de longo prazo serão detalhadas e se tornarão atividades de curto prazo, ou serão refinadas por várias atividades de curto prazo, ou poderão ser completamente abandonadas se não fizerem mais sentido para o projeto.

Recomendamos distinguir entre as fases de configuração e execução da elicitação. Nas subseções a seguir, fornecemos diretrizes para ambas as fases.

Além da fase de configuração e execução, é possível adicionar uma *fase de conclusão* que se concentre no aprimoramento das habilidades de elicitação dos participantes do projeto. Nessa fase, o histórico de versões do plano de elicitação é revisado em relação aos resultados (os requisitos elicitados e os conflitos resolvidos) para aprender com os sucessos e as falhas no projeto finalizado. Em projetos iterativos, isso pode ocorrer após cada iteração.

1.3.3.2 Diretrizes para a fase de configuração

O conjunto inicial de atividades de elicitação é definido na *fase de configuração*. Esse conjunto inicial descreve a abordagem pretendida para a elicitação de requisitos. Ele baseia-se nas características específicas do projeto (p. ex., a abordagem de desenvolvimento) e no conhecimento e nas suposições existentes dos participantes do projeto. Normalmente, as atividades de resolução não são definidas na fase de configuração, pois os conflitos de requisitos são desconhecidos nesse estágio. No entanto, se houver conflitos de requisitos conhecidos (p. ex., de projetos anteriores) ou indicadores

de possíveis conflitos, um plano para lidar com eles deve ser incorporado o mais rápido possível.

O termo *fase de configuração* não significa que você tenha que gastar vários dias ou até semanas para desenvolver o conjunto inicial de atividades. No entanto, a fase de configuração é de grande importância para um início de projeto limpo e para permitir a elicitación eficaz de requisitos. Lembre-se de que a elicitación é semelhante a um projeto de pesquisa, e um bom projeto de pesquisa exige a definição de metas de pesquisa adequadas para ser eficaz.

Obter uma visão geral da situação do projeto e do caso de negócios

Cada projeto é único. O que pode ter funcionado para um projeto de desenvolvimento pode ser completamente errado para outro projeto. É importante analisar cada novo projeto para ter uma visão clara de quais atividades de elicitación são adequadas. Primeiro, você precisa entender a natureza e o contexto do projeto. Você pode ser completamente novo no domínio e/ou na organização ou pode estar trabalhando lá há muitos anos. Se for o último caso, alguns dos problemas discutidos na seção a seguir podem não ser menos importantes devido à sua experiência.

Um projeto de desenvolvimento também é fortemente influenciado por seu **domínio**. Você sabe o suficiente sobre esse domínio para entender como ele influencia o projeto e para saber o que é importante? Se você for novo em um domínio, terá que ler a literatura (pelo menos introdutória) e/ou os sites para entender o contexto e a terminologia do domínio. Um objetivo comum de elicitación para desenvolver uma compreensão do domínio é a criação de um glossário (consulte também [IREB2020]). Além disso, você deve procurar pessoas que possam ajudá-lo a entender o domínio e pedir a elas que expliquem o que você precisa saber no contexto do seu projeto.

É importante entender o histórico de um projeto. Nenhum projeto surge do nada; mesmo um projeto novo em folha tem uma história. Você precisa conhecer esse histórico para entender os objetivos do projeto e evitar armadilhas. Algumas perguntas que você deve fazer nesse contexto são: Por que esse projeto foi iniciado? Quem o iniciou? Já houve tentativas fracassadas de atingir a meta do projeto? Se sim, quem estava envolvido nessas abordagens? Por que essas abordagens falharam? Foi realizado um pré-estudo? Em caso afirmativo, quem estava envolvido e quais foram os resultados (possíveis fontes de requisitos!)? Compreender o histórico de um projeto não só o ajudará a definir o objetivo da elicitación e a identificar as fontes de requisitos corretas, mas também durante as atividades de resolução (identificar um conflito, compreender um conflito, conhecer as raízes de um conflito).

Obviamente, você também deve conversar com os iniciadores do projeto, líderes do projeto, membros do projeto e qualquer pessoa que possa lhe fornecer informações sobre o projeto (para obter mais detalhes sobre a análise dos stakeholders, consulte o Capítulo 2). Quem são as pessoas "importantes" em seu projeto? Quem sabe o que está acontecendo com o projeto, com as tecnologias usadas, com o domínio etc.? Quais são os objetivos do projeto? Quais são os cronogramas? Como o projeto é organizado?

Quem é responsável pelo quê? Essas são apenas algumas perguntas que o ajudarão a entender o projeto de desenvolvimento e seu contexto. Você também deve descobrir se há **restrições específicas do projeto** que possam influenciar a abordagem de elicitação.

Você também deve tentar obter uma visão clara da **complexidade** do projeto ou de como ele pode se tornar. O que influencia a complexidade do projeto (fatores de complexidade)? Há alguma indicação de que o projeto pode ser (muito) mais complexo do que se supõe atualmente? Há alguma regulamentação referente à organização do projeto ou aos produtos de trabalho que precisam ser cumpridos (p. ex., Automotive SPICE, GxP²)? Quantas pessoas ou empresas estão envolvidas no projeto? A equipe do projeto está localizada no mesmo local ou está distribuída (globalmente)? E quanto aos possíveis stakeholders? Já está claro que alguns deles são difíceis de alcançar?

Determinar o objetivo da elicitação

Recomendamos iniciar a parte de elicitação de um projeto com a definição dos objetivos da elicitação. O objetivo da elicitação é o elemento central da atividade de elicitação, pois orienta o que queremos aprender sobre os requisitos ou suas fontes, bem como a técnica de elicitação que devemos aplicar para atingir esse objetivo.

Primeiro, deve ser criada uma lista inicial que mostre o nível de compreensão do projeto por você (ou pela equipe). Se os participantes do projeto concordarem com uma lista de perguntas detalhadas, isso mostra que eles sabem o que querem aprender no projeto. Os objetivos detalhados de elicitação podem ser transformados em atividades de elicitação de curto prazo e programados de acordo com as prioridades do projeto.

Uma lista de objetivos de elicitação abstratos ou vagos pode ser um indicador de um entendimento fraco do resultado do projeto visado. Essa situação não é incomum. Selecione dois ou três objetivos que pareçam ser os mais importantes e planeje uma atividade de elicitação de curto prazo para obter uma melhor compreensão desses objetivos. Em seguida, refine o objetivo. Os objetivos vagos de elicitação com alta importância para o projeto, devem ser detalhados por uma ou mais atividades de elicitação de curto prazo. Objetivos vagos de elicitação com menos importância devem ser transformados em atividades de elicitação de longo prazo para consideração posterior no projeto.

Planejar a análise sistemática do contexto do sistema

Desde o nível básico, sabemos que o contexto do sistema é essencial para a identificação das fontes de requisitos e para a compreensão dos requisitos (consulte [IREB2020]).

Um objetivo de elicitação altamente recomendado, portanto, é **entender o contexto do sistema**. Os objetivos de elicitação recomendados para a aquisição de uma compreensão do contexto do sistema são:

- Identificar pessoas (stakeholders ou grupos de stakeholders) relacionadas ao sistema

² GxP é uma abreviação geral para diretrizes e normas de qualidade de "boas práticas". O "x" representa o campo de aplicação, por exemplo, GAP para "Good Agricultural Practice" (Boas Práticas Agrícolas).

- Identificar os sistemas em operação (outros sistemas técnicos ou hardware) relacionados ao sistema
- Identificar documentos
- Identificar os processos (processos técnicos ou físicos, processos de negócio) nos quais o sistema está envolvido
- Identificar eventos

Planejar a identificação sistemática e pragmática de [vários tipos de] fontes de requisitos

As fontes de requisitos corretas são um recurso fundamental para o sucesso da elicitação de requisitos. Compreender o domínio e a organização é essencial para a identificação sistemática e pragmática das fontes de requisitos. Observe que a precisão com que uma fonte é definida pode variar para atividades de elicitação de curto e longo prazo (veja acima):

- As fontes de requisitos para atividades de elicitação de curto prazo devem ser nomeadas (p. ex., nomear os stakeholders específicas para uma entrevista ou nomear o padrão que deve ser analisado). Isso é necessário para tornar a atividade de elicitação executável.
- As fontes de requisitos para atividades de elicitação de longo prazo também podem ser definidas como tipos ou categorias de fontes de requisitos.

Entender a **organização** e sua cultura também é importante. Mesmo dentro do mesmo domínio, duas empresas podem ter uma cultura e um modo de fazer as coisas totalmente diferentes. Dependendo da organização, também pode haver grandes diferenças culturais entre as subsidiárias ou até mesmo entre os departamentos dessa organização. Certifique-se de que você conhece as regras não escritas da organização. Isso inclui coisas simples, como a forma como as pessoas são tratadas (p. ex., é comum tratar alguém pelo nome ou sobrenome?). Você também deve saber qual é o papel da hierarquia nessa organização. Até que nível você pode convidar diretamente pessoas em cargos de liderança e quando é necessário marcar uma reunião com o escritório delas? Além da cultura da organização, você também deve saber como ela é estruturada. Quais são os departamentos existentes? Como os departamentos estão conectados uns aos outros? E, é claro, você deve saber como a organização cria o fluxo de caixa (ou seja, você deve saber o que ela produz/vende – mesmo que isso não tenha nada a ver com o projeto de desenvolvimento para o qual você está solicitando requisitos!).

Considere os padrões de processo relevantes para definir as atividades

À primeira vista, a estrutura descrita nesta seção pode parecer esmagadora e complicada. Essa não é, obviamente, a intenção. A estrutura descrita aqui é uma abstração dos melhores pontos de várias abordagens encontradas na literatura.

A literatura fornece vários métodos sofisticados que apóiam a elicitação de requisitos. Os métodos populares são, por exemplo, o design centrado no ser humano e o Design Thinking.

Usando nossa estrutura, esses métodos podem ser considerados como uma sequência de várias técnicas de elicitación e exigem uma quantidade significativa de planejamento e esforço. Ao mesmo tempo, a descrição do método na literatura fornece várias dicas para a definição dos objetivos da elicitación e das fontes de requisitos.

É altamente recomendável planejar mais de uma atividade de elicitación ao usar um método. Atividades diferentes podem, por exemplo, corresponder a técnicas diferentes pertencentes ao mesmo método.

Para apoiá-lo na aplicação dos métodos apresentados na literatura, desenvolvemos o conceito de padrões de processo na elicitación de requisitos. A seção 1.4 apresenta esse conceito como um tópico separado.

Permitir tempo e orçamento para atividades de resolução

Embora possa não haver conflitos no início de um projeto, você deve planejar tempo para atividades de resolução. Nem todos os conflitos podem ser um obstáculo, mas haverá conflitos que você terá de resolver:

- Dê tempo para encontrar conflitos ativamente, pois quanto mais cedo você fizer isso, maior será a possibilidade de encontrar tempo para resolvê-los.
- Dê tempo para resolver novos conflitos antes de ter evidências concretas de sua existência.
- Programe a análise de conflitos o mais cedo possível, pois é possível obter muito conhecimento útil e evitar ter de lidar com stakeholders que não cooperam mais tarde.

Busque vitórias rápidas. Se puder evitar o surgimento de um conflito ou encontrar uma solução rápida (mesmo que não tenha sido explicitamente planejada), aproveite a oportunidade. Talvez você não tenha todos os participantes em seu workshop tão cedo.

1.3.3.3 Diretrizes para a fase de execução

A *fase de execução* concentra-se na execução das atividades de elicitación. Nessa fase, as atividades de elicitación são realizadas de acordo com o plano existente.

Considere as atividades de elicitación como atividades com prazo determinado

Um sério risco na elicitación de requisitos é o desperdício de tempo e recursos em atividades ineficazes que não alcançam os resultados esperados. Portanto, recomendamos que as atividades de elicitación sejam consideradas como *time-boxed*, em que cada *time-box* adiciona informações de forma iterativa e incremental ao conjunto de requisitos.

Se uma atividade não atender às expectativas de esforço e/ou cronograma, interrompa a atividade e examine os resultados alcançados para entender o motivo do fracasso. Pode haver vários motivos para o possível fracasso de uma técnica de elicitación: por exemplo, a técnica de resolução de conflitos não se encaixa na situação de conflito ou as fontes de requisitos abordadas não são capazes de fornecer informações adequadas.

Os resultados desse exame podem ser usados para planejar novas atividades ou para refinar as atividades existentes.

Questionar o plano após cada atividade (e revisar, se necessário)

O plano de elicitação existente não deve ser considerado fixo em pedra. O plano foi criado com as informações disponíveis na época, incluindo as suposições. A aquisição de novas informações, a ocorrência de conflitos de requisitos e a refutação de suposições existentes são atividades cotidianas na elicitação.

Ajuste seu plano continuamente, usando o conhecimento obtido sobre a organização, os stakeholders e a complexidade do projeto durante as atividades em andamento. Não se concentre apenas nos requisitos obtidos, mas também em todas as outras informações. As perguntas a seguir podem servir como uma lista de verificação para revisar seu plano após cada atividade de elicitação:

- Os resultados de uma atividade têm um impacto sobre as atividades de curto prazo definidas?
- Os objetivos da atividade relacionada ainda são válidos?
- Os resultados de uma atividade podem ser usados para refinar objetivos vagos ou abstratos existentes?
- É possível revisar os objetivos de longo prazo existentes?
- Os resultados de uma atividade levam a novos objetivos?
- Os resultados indicam um novo conflito de requisitos?
- Os resultados são úteis para resolver algum conflito de requisitos existente?

Faça uma programação defensiva e use atividades de elicitação de curto e longo prazo

Além da definição precisa das atividades de elicitação, a sequência de execução adequada é um fator importante para o sucesso da elicitação de requisitos. Especialmente em situações de projeto com um alto nível de incerteza, recomendamos uma abordagem defensiva para a programação das atividades de elicitação a fim de evitar o desperdício de recursos.

A programação das atividades de elicitação é orientada pelos seguintes fatores:

- **Disponibilidade de stakeholders/fontes de requisitos:** Uma atividade só pode ser realizada se os participantes necessários (para atividades de elicitação) ou as fontes de requisitos (para atividades de elicitação) estiverem disponíveis.
- **Disponibilidade de Engenheiros de Requisitos:** Uma atividade só pode ser realizada se Engenheiros de Requisitos qualificados estiverem disponíveis para executá-la.
- **Valor para o projeto:** As atividades com alto valor para o projeto devem ser executadas com prioridade. A definição de valor depende da situação específica. Algumas definições de valor são, por exemplo:
 - Importância dos requisitos que uma atividade de elicitação fornece (definida pelo objetivo da elicitação)
 - Importância do conflito a ser resolvido

- Importância das informações fornecidas para outras atividades de elicitação (p. ex., uma atividade de elicitação foi definida para detalhar um objetivo de elicitação vago, mas importante)

Incorporar folga para deixar tempo para criatividade, resolução de conflitos e eventos inesperados

Esteja ciente de que não é possível planejar tudo antecipadamente, pois é preciso lidar com um alto nível de incerteza. Não se surpreenda se, em seu caminho, surgirem novas atividades urgentes. Se você não previu uma folga para lidar com isso, suas outras atividades planejadas serão prejudicadas e não haverá espaço para a criatividade. Em uma fase inicial de um projeto de elicitação, 25% de folga não é incomum. Posteriormente, essa porcentagem pode ser reduzida, mas nunca será zero.

Paralelizar atividades independentes

A paralelização de atividades independentes pode aumentar a eficiência das atividades de elicitação. Duas ou mais atividades de elicitação são consideradas independentes entre si se os objetivos da elicitação forem independentes entre si e as fontes de requisitos não forem idênticas. Não é aconselhável paralelizar atividades que tenham objetivos de elicitação dependentes (ou mesmo iguais), porque o resultado de uma atividade pode ter impacto sobre a outra.

Duas ou mais atividades de resolução são consideradas independentes umas das outras se os requisitos conflitantes forem independentes uns dos outros e os participantes do conflito forem diferentes. Não é aconselhável paralelizar atividades que tenham conflitos interdependentes, pois a resolução de um conflito pode ter impacto sobre o outro.

No entanto, certifique-se de coordenar regularmente todas as atividades paralelas. Caso contrário, você pode perder dependências ocultas ou descobertas que podem ajudar em outras atividades.

Combinar atividades de elicitação que abordam a mesma fonte de requisitos

As atividades de elicitação que abordam a mesma fonte de requisitos (p. ex., a mesma stakeholder ou grupo de stakeholders) podem ser combinadas para aumentar a eficiência.

Por exemplo, se quisermos abordar três objetivos diferentes de elicitação com entrevistas e o interessado nesses objetivos for idêntico, é possível agendar uma entrevista e trabalhar em todos os três objetivos nessa reunião.

Procurar conflitos e reagir a eles de acordo com uma estratégia acordada

A busca ativa de conflitos é uma tarefa diária. Não importa se você realiza atividades de elicitação, revisões de especificações, garantia de qualidade em um modelo de requisitos ou qualquer outra tarefa, você deve desenvolver boas habilidades para verificar a consistência em todos os momentos.

Documente informações sobre possíveis conflitos, por exemplo, indicadores de conflito em sua lista de stakeholders, para revisão posterior.

Para conflitos sociais ou mistos, esteja ciente dos indicadores. Nem todo indicador pode levar a um conflito real, mas todo conflito tem indicadores. É sempre útil saber onde observar antes que seja tarde demais para reagir.

Mantenha-se sempre neutro em um conflito e tente encontrar uma solução sustentável para todo o projeto e todas os stakeholders. Também é útil ser percebido como neutro ao assumir a função de moderador na situação de conflito ou no workshop.

Exemplos de como encontrar conflitos na fase de execução são:

- Procure termos inconsistentes em seu glossário ou modelo de terminologia.
- Contar: Se sua GUI tem cinco campos e você definiu seis rótulos, deve haver um erro!
- Se o participante A quiser um botão vermelho, o mesmo botão não poderá ser verde para o participante B.
- Se você definiu um comportamento para uma condição, também definiu um comportamento alternativo quando a condição não se aplica?

1.4 Padrões de processo

Sabemos, pela prática do setor, que cada projeto é um evento único. Portanto, é um desafio fornecer orientações concretas para a realização da elicitação de requisitos, uma vez que há vários fatores que podem influenciar a melhor abordagem. Alguns desses fatores são o tempo e o orçamento disponíveis, o tipo de sistema a ser desenvolvido, a disponibilidade dos stakeholders e a experiência dos Engenheiros de Requisitos.

No entanto, há certos métodos da literatura e da prática do setor que comprovaram sua utilidade em várias situações. Além de descrever alguns desses métodos, apresentamos a ideia de padrões de processo. O conceito de padrões foi originalmente desenvolvido em um contexto arquitetônico [Allis1977] para documentar o conhecimento reutilizável para a criação de arquiteturas e, posteriormente, foi transferido para o desenvolvimento de software com a introdução de padrões de design para software [GHJV1994].

Em um contexto de elicitação, os padrões de processo fornecem um formulário para documentar maneiras úteis e comprovadas de realizar a elicitação de requisitos. O principal objetivo desta seção é ensinar a ideia subjacente dos padrões de processo como um kit de ferramentas, em vez de ensinar abordagens específicas. Primeiro, é descrita a estrutura dos padrões de processo para elicitação de requisitos. Em seguida, são apresentados alguns exemplos de padrões de processo.

1.4.1 Estrutura e benefícios dos padrões de processo para elicitação de requisitos

Um padrão de processo consiste nos seguintes elementos:

- *Escopo*: Esta seção descreve a situação ou as situações do projeto que são adequadas para a aplicação de um padrão. Essa descrição também pode incluir situações específicas em que a aplicação do padrão não é aconselhável.

- *Esforço/recursos necessários*: Esta seção descreve o esforço e os recursos necessários para aplicar um determinado padrão. Normalmente, o esforço é descrito em termos de recursos de tempo e pessoal. Os recursos adicionais podem incluir, por exemplo, material de workshop, software especial ou locais especiais.
- *Elementos do padrão*: Esta seção explica os detalhes desse padrão. A descrição inclui referências aos métodos aplicados e descrições concretas das atividades de elicitação para que o leitor possa entender o padrão em termos de elicitação de requisitos.
- *Instanciação*: Esta seção fornece detalhes específicos sobre como começar. A descrição inclui atividades para preparar o padrão e um plano inicial com atividades de elicitação (consulte a Seção 1.3.1.1) que pode ser usado como diretriz para começar.
- *Leitura adicional*: Esta seção fornece referências adicionais à literatura útil com mais detalhes sobre o padrão.

Dependendo da origem das ideias subjacentes ao padrão, os detalhes descritos não estão necessariamente restritos à elicitação de requisitos e podem incluir outras atividades, como design ou teste. Essas partes não relacionadas à solicitação devem ser curtas. Sempre que possível, a descrição descreverá os resultados de tais atividades de não elicitação para uso posterior na elicitação de requisitos.

O Engenheiro de Requisitos deve procurar ativamente por padrões que sejam relevantes para sua própria situação. Não se esqueça disso:

- Os padrões de processo são boas práticas da literatura e da prática, fornecendo um ponto de partida para definir as atividades de elicitação em situações comparáveis.
- Os padrões de processo fornecem uma meta-perspectiva sobre a maneira de trabalhar. Em vez de aplicar a mesma abordagem em todas as situações do projeto, os padrões de processo permitem o desenvolvimento de uma caixa de ferramentas além de técnicas específicas (p. ex., entrevistas).
- Normalmente, as informações fornecidas não são suficientes para a execução imediata do processo. A análise das semelhanças e diferenças entre o escopo do padrão e a situação real ajuda a identificar uma abordagem adequada e a selecionar as técnicas apropriadas.
- The list of patterns covered in this chapter is neither complete nor exhaustive. Além disso, os padrões podem, e muitas vezes devem, ser combinados de várias maneiras.
- Os engenheiros de requisitos experientes são incentivados a desenvolver e compartilhar seus próprios padrões.

Esta seção apresenta três exemplos de padrões: Waterfall, design centrado no ser humano e Design Thinking. A Waterfall foi selecionada como padrão porque o modelo de processo da Waterfall é frequentemente confundido com a Engenharia de Requisitos como um método. A descrição aqui deve deixar claro que o método Waterfall é apenas uma forma de realizar a engenharia/elicitação de requisitos.

O design centrado no ser humano e o Design Thinking são apresentados porque são duas abordagens muito populares para a obtenção de requisitos e combinam várias técnicas (p. ex., workshops e técnicas de criatividade) com um modelo de processo que permite uma aplicação fácil em várias situações.

O desenvolvimento ágil não foi intencionalmente incluído aqui como um padrão porque os detalhes sobre a Engenharia de Requisitos em projetos ágeis são abordados no IREB CPRE RE@Agile.

1.4.2 Waterfall

O modelo Waterfall descreve um modelo de ciclo de vida de desenvolvimento de software linear e sequencial, no qual o desenvolvimento de um sistema passa por uma sequência de diferentes fases. Royce [Royce70] descreveu pela primeira vez esse modelo com as fases de Requisitos, Análise, Projeto, Codificação, Testes e Operações. Outras definições de fase foram publicadas, por exemplo, o padrão DOD-STD-2167A do Departamento de Defesa dos Estados Unidos [DoD88]. O modelo em V [FoMo91] ou o modelo em espiral de Boehm [Boehm88] são variações do ciclo de vida no Waterfall.

1.4.2.1 Escopo

Esse tipo de desenvolvimento é visto principalmente em projetos grandes e de longa duração, desenvolvendo sistemas técnicos (incorporados), sistemas complexos com interfaces para outros sistemas (legados) e seguindo uma estratégia big-bang para implementação.

As características mais óbvias desse tipo de projeto são:

- Uma meta de negócio claramente definida;
- Compreensão precisa do sistema a ser desenvolvido;
- Compreensão precisa do domínio;
- Compreensão precisa da tecnologia que se pretende usar;
- Diretrizes rígidas de um único proprietário do negócio ou de um comitê de direção;
- Uma abordagem rigorosa, definida em um plano formal com orçamentos e cronogramas desde o início, e controlada por um sólido gerenciamento de projetos;
- Equipes de diferentes profissionais trabalhando nas fases individuais;
- Os resultados de cada fase servem de entrada para a fase seguinte, transmitindo informações por meio de documentação formal;
- Portas de qualidade entre as fases, com decisões formais da gerência para prosseguir;
- Depois de passar por uma porta de qualidade, os requisitos resultantes são "congelados" como especificações e, se forem necessárias alterações posteriores, eles seguem um procedimento formal de alteração.

1.4.2.2 Esforço / recursos

Os projetos em Waterfall geralmente têm uma longa duração (de meses a anos) e exigem um grande esforço (geralmente várias centenas de homens-mês). No início do projeto, um plano detalhado é definido e acordado para o restante do projeto. Esse plano inclui uma fase inicial de requisitos que pode durar de algumas semanas a vários meses e pode envolver vários profissionais.

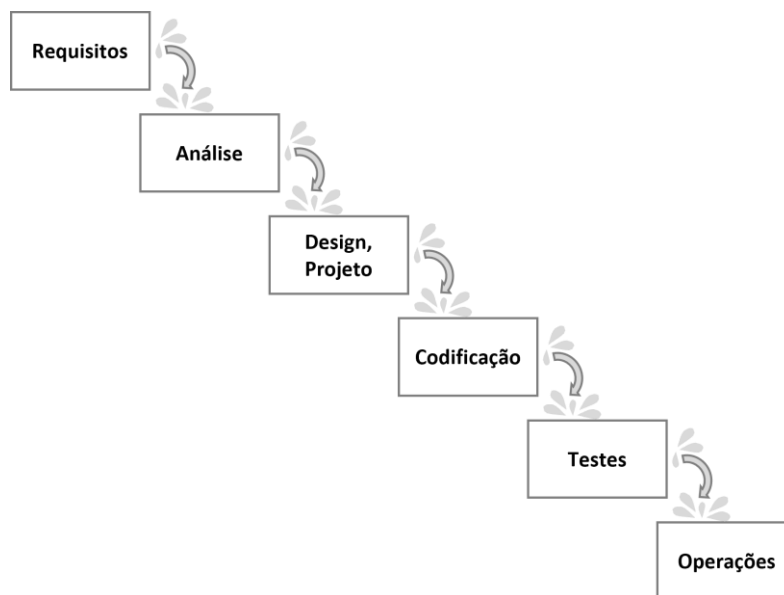


Figura 4: Processo em Waterfall

1.4.2.3 Elementos de padrão

Existem diferentes definições de fase para o modelo Waterfall, geralmente variantes da definição original de Royce (consulte Figura 4). Do ponto de vista da Engenharia de Requisitos, a fase de requisitos pode ser considerada a mais importante. Nessa fase, os principais requisitos são determinados, definindo a direção geral para o restante do projeto. Embora menos proeminente, a Engenharia de Requisitos continua a desempenhar um papel em todas as fases subsequentes.

Os parágrafos a seguir descrevem cada fase em mais detalhes, incluindo diretrizes para atividades de elicitação.

Requisitos

O objetivo dessa fase é traduzir o objetivo de negócio de alto nível formulado pelo proprietário do negócio em um conjunto validado de requisitos principais (negócio). Além disso, são identificadas as principais restrições que definem o espaço de solução para o sistema e o projeto.

Os Engenheiros de Requisitos normalmente obtêm os requisitos por meio de entrevistas qualitativas com um grupo (geralmente pequeno) de executivos e gerentes, resumindo os resultados em relatórios e processando o feedback deles.

Pode ser difícil conseguir tempo suficiente dos entrevistados, devido às suas agendas lotadas. Além das entrevistas, a clareza sobre os requisitos também pode ser obtida por meio da análise de documentos.

Os resultados são consolidados em um relatório final detalhado que passa por uma série de versões preliminares, que são revisadas repetidamente. Os conflitos são resolvidos por meio do processamento do feedback até que se chegue a um consenso ou por meio da superação pela gerência superior. No final da fase, uma inspeção formal no último rascunho serve como um portão de qualidade, após o qual o relatório final é liberado para entrada na próxima fase.

Análise

A fase de análise concentra-se na elaboração da arquitetura de informações, da interface do usuário e das interfaces com outros sistemas em um conjunto de requisitos do sistema.

Uma das principais tarefas é extrair as restrições decorrentes do cenário de TI circundante, principalmente por meio de análise de documentos e entrevistas com administradores de sistemas e outros profissionais de TI. No lado do usuário, o foco muda para os usuários finais diretos. Tanto as entrevistas quanto os questionários podem ser usados para a elicitación, dependendo do tamanho e da complexidade da comunidade de usuários. Workshops com grupos de usuários também podem ser relevantes. Se a comunidade de usuários for diversificada, as personas podem ser úteis. Uma armadilha pode ser "esquecer" os usuários externos e outros clientes indiretamente afetados, presumindo que os usuários internos representam suficientemente sua opinião.

Mais uma vez, essa fase resulta em um relatório final consolidado e validado, que passa por um controle de qualidade antes da liberação.

Design, Projeto

Na fase de projeto, os requisitos do sistema são elaborados em requisitos de software como um projeto para o sistema a ser construído.

Nessa fase, os profissionais de TI estão na liderança: designers funcionais e técnicos, administradores de banco de dados e de sistemas, operadores técnicos. As interfaces são definidas com mais detalhes, e o feedback dos representantes dos usuários é coletado e processado. Os protótipos de baixa fidelidade podem ser úteis. Técnicas de observação podem ser aplicadas para alinhar o sistema projetado com os processos operacionais dos usuários finais. Uma tarefa importante pode ser conciliar os requisitos de qualidade com as restrições técnicas.

Ao final dessa fase, o conjunto completo de requisitos comerciais, de sistema e de software terá se transformado em uma coleção grande e complexa que deve ser gerenciada ativamente (gerenciamento de alterações e configurações, linhas de base, controle de versão, priorização, rastreabilidade). Manter a consistência entre todos os requisitos é uma grande preocupação.

A fase de design resulta em um ou mais documentos de design, que também passarão por um controle de qualidade antes do lançamento. Quando a terceirização é usada, os documentos formais do projeto formam a base do contrato com a parte terceirizada. Nesses casos, a qualidade dos requisitos nos documentos de design é fundamental para o sucesso do projeto.

Codificação

O objetivo dessa fase é criar uma solução técnica que atenda aos requisitos estabelecidos durante as fases anteriores. Os desenvolvedores de TI, como designers técnicos, especialistas em bancos de dados e programadores, estão envolvidos.

Em teoria, não são desenvolvidos novos requisitos. Na prática, os desenvolvedores podem detectar inconsistências entre os requisitos, conflitos com restrições de baixo nível negligenciadas ou interpretações e cálculos incorretos de fases anteriores. Às vezes, protótipos de alta fidelidade são usados no início dessa fase para obter feedback dos usuários finais.

Os testes de componentes e de integração servem como portas de qualidade para a próxima fase.

Testes

Na fase de testes, testadores independentes, usuários finais, administradores e operadores de sistemas tentam verificar se o sistema atende aos requisitos definidos e validar se o sistema dará suporte aos processos de negócios pretendidos sem grandes riscos. Mais uma vez, não são esperados novos requisitos, mas a resolução de defeitos pode exigir a reconsideração e o ajuste de soluções anteriores.

A fase de testes é, por si só, o portão de qualidade final, fornecendo as informações necessárias para que a gerência sênior tome uma decisão de ativação.

Operações

Na fase de operações, o sistema é usado em uma situação de negócio real.

Podem ocorrer incidentes no sistema operacional e mudanças no ambiente de negócios, o que leva a requisitos novos ou atualizados. A análise de impacto é aplicada para decidir sobre solicitações de mudança. É importante incorporar todas as alterações aceitas e garantir que um conjunto único, consistente e atualizado de requisitos seja mantido durante toda a vida útil do sistema.

1.4.2.4 Instanciação desse padrão

No padrão Waterfall, a demanda por qualidade de resultado é muito alta. A baixa qualidade nas fases iniciais inevitavelmente leva a projetos com excesso de tempo e custos e a sistemas que não atendem às expectativas dos clientes.

Os portões de qualidade no final de cada fase servem como proteção contra a baixa qualidade. Nas fases iniciais, as revisões e inspeções são as principais técnicas para manter a qualidade em um alto nível. É uma boa prática envolver os participantes não apenas da fase atual, mas também das fases anterior e posterior. Depois de passar pelo controle de qualidade, os requisitos são "congelados" como especificações para as fases seguintes. Isso significa que, se alguém quiser alterar um requisito posteriormente, será necessário seguir um procedimento formal de alteração, no qual o impacto da alteração será analisado e avaliado. Na prática, isso geralmente significa que as mudanças são desencorajadas. Dessa forma, as especificações das fases anteriores funcionam como restrições para as fases seguintes. Isso é tanto um ponto forte – traz estabilidade – quanto um ponto fraco – causa inflexibilidade – do padrão Waterfall. Portanto, esse modelo é mais bem utilizado em ambientes comerciais estáveis, nos quais uma meta clara deve ser alcançada de forma altamente controlada e auditável.

Ao iniciar a Engenharia de Requisitos em um projeto em Waterfall, você se concentrará na fase de requisitos. Embora pareça contrário à natureza de um projeto como esse, considere uma abordagem iterativa e com prazo determinado, na qual a principal meta de negócio é refinada gradualmente em um conjunto de requisitos de negócio. Os prazos devem se encaixar no planejamento geral do projeto para produzir um relatório final a tempo para o início da fase de análise.

A tabela a seguir mostra um exemplo de lista de objetivos.

Rua	Objetivo da Elicitação	Técnica de Elicitação	Fonte de Requisitos	Qualidade do Resultado
1	Identificar os stakeholders e a documentação relevantes	Entrevista	Proprietário do Negócio	Média
2	Prepare-se para as entrevistas	Análise de documentos	Documentos relevantes de domínio e TI	Média
3	Entenda as necessidades do negócio	Entrevista	Executivos relevantes	Média
4	Obter feedback	Distribuição para comentários	Executivos relevantes	Médio – Alto
Repita as etapas 2 a 4 até que o feedback comprove que o nível de qualidade é satisfatório				
5	Consolidar o quadro geral	Oficina	Proprietário do negócio, entrevistados, outros stakeholders	Alta

Rua	Objetivo da Elicitação	Técnica de Elicitação	Fonte de Requisitos	Qualidade do Resultado
6	Validar resultados	Inspeção	Stakeholders relevantes e representantes da próxima fase	Alta
7	Obter o aceite	Relatório final	Proprietário do negócio e stakeholders relevantes	Alta

Nas próximas fases, o Engenheiro de Requisitos pode não estar em uma função de liderança. Os requisitos detalhados do sistema e do software serão desenvolvidos e refinados em conjunto com outros profissionais. A principal preocupação do Engenheiro de Requisitos será o gerenciamento do crescente conjunto de requisitos. O Engenheiro de Requisitos geralmente atua como consultor e instrutor em questões de requisitos, resolvendo conflitos e mantendo a consistência.

Certifique-se de que haja capacidade suficiente de Engenharia de Requisitos disponível durante essas fases, especialmente durante os testes (de aceite), nos quais os Engenheiros de Requisitos devem ter uma função ativa para garantir que o sistema realmente se comporte conforme especificado no conjunto de requisitos.

1.4.2.5 Leitura adicional

- [Royce70] W. Royce: Managing the Development of Large Software Systems. In Proceedings of IEEE WESCON 26 (August): 1–9, 1970
- [DoD88] United States Department of Defense: DOD–STD–2167A, MILITARY STANDARD: DEFENSE SYSTEM SOFTWARE DEVELOPMENT, 1988
- [FoMo91] K. Forsberg, H. Mooz: The Relationship of System Engineering to the Project Cycle. In: Proceedings of the First Annual Symposium of National Council on System Engineering, October 1991: 57–65.
- [Boehm88] B.W. Boehm: A Spiral Model of Software Development. In Tutorial: Software Engineering Project Management. Editado por R.H. Thayer, IEEE Computer Society Press, Washington D.C., 1988, pp. 128–142.

1.4.3 Design centrado no ser humano

O design centrado no ser humano (HCD), também conhecido como desenvolvimento orientado pelo usuário (UDD), é uma estrutura de processos (não restrita a interfaces ou tecnologias) na qual o contexto de uso, as metas de usabilidade, as características do usuário, o ambiente, as tarefas e o fluxo de trabalho de um produto, serviço ou processo recebem atenção especial em cada estágio do processo de design [RiFi2014].

A usabilidade de um sistema é o grau em que o sistema pode ser usado por usuários específicos para atingir objetivos específicos com eficácia, eficiência e satisfação em um contexto de uso específico [ISO9241.11].

Considerando a definição mencionada acima, fica claro que a "usabilidade" não é algo que possa ser simplesmente mencionado em uma entrevista e depois definido como um requisito não funcional, por exemplo, "o sistema deve ser utilizável". A usabilidade de um sistema depende das tarefas de seus usuários e do contexto de uso. O design centrado no ser humano é uma abordagem para identificar esses elementos interdependentes:

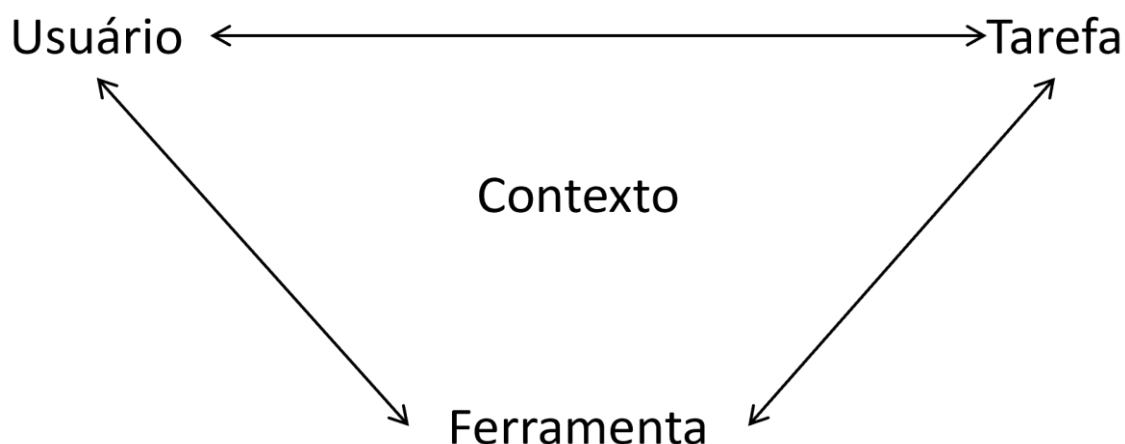


Figura 5: Elementos de dependência mútua que precisam ser considerados no HCD [Shackel1991]

A experiência do usuário (UX) refere-se ao conceito de contatos multicanais entre um cliente e um fornecedor. O cliente vivencia uma jornada desde seu primeiro contato com o fornecedor (provavelmente por meio de um anúncio ou representante de vendas), passando por várias interações com pontos de contato on-line ou off-line (p. ex., pesquisar um produto na loja on-line, comprá-lo, lidar com o atendimento ao cliente e pagar por ele), até encerrar o relacionamento com o cliente. Portanto, o trabalho do fornecedor não é otimizar a usabilidade de um canal específico, mas sim otimizar toda a experiência do usuário/cliente. No modelo de Shackel (consulte Figura 5), esse é um dos aspectos importantes do contexto de uma ferramenta: a ferramenta é apenas um dos muitos pontos de contato com o usuário.

1.4.3.1 Escopo

O padrão HCD se aplica a projetos em que há um alto nível de interação com os usuários. Se as interações forem principalmente com outros sistemas, esse pode não ser um padrão adequado. No entanto, a maioria dos sistemas acaba servindo a seres humanos e, portanto, vale a pena descobrir quem são eles (grupos de usuários), o que fazem (tarefas) e em que contexto operarão o novo sistema.

1.4.3.2 Esforço / recursos

Se um sistema grande e novo for desenvolvido e a equipe do projeto for nova no domínio e/ou recém-formada, deve ser planejada uma fase de reflexão inicial significativa (até 3 meses). Isso dá à equipe a oportunidade de aprender sobre o domínio, pesquisar os usuários, suas tarefas e seu contexto, além de identificar e resolver os principais riscos. A configuração inicial do desenvolvimento, por exemplo, o estabelecimento de ambientes de desenvolvimento, a integração contínua, uma primeira versão da arquitetura da solução etc., também deve ocorrer. São necessárias habilidades de UX, com mais profissionais necessários se houver vários grupos de usuários a serem investigados.

Se o sistema for menos complicado, a equipe de desenvolvimento estiver bem estabelecida no domínio e tiver um longo histórico de trabalho conjunto bem-sucedido, ou se o projeto for apenas uma extensão ou correção de manutenção de um sistema desenvolvido anteriormente de acordo com o HCD, o período de reflexão inicial poderá ser reduzido para algumas semanas ou até dias. São necessárias menos habilidades especializadas em UX, embora seja uma boa prática que todos os membros da equipe do projeto estejam cientes da importância da UX e tenham uma mentalidade centrada no ser humano.

1.4.3.3 Elementos de padrão

O HCD pode ser executado como um processo que abrange todo o desenvolvimento do sistema – consulte a ISO 9241 210, o Processo de Design Centrado no Ser Humano:

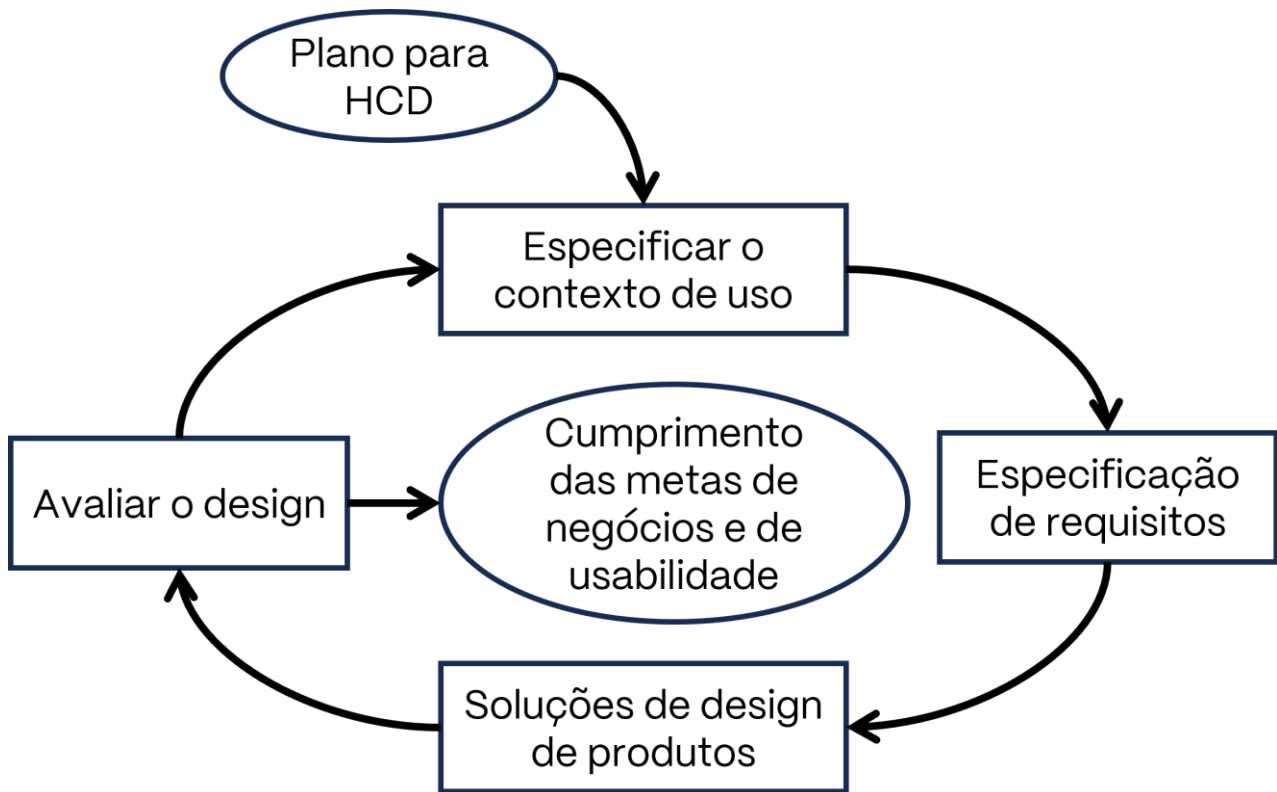


Figura 6: Design iterativo e centrado no ser humano (HCD) para sistemas interativos
[IS09241.210]

Entretanto, o HCD é uma abordagem ou até mesmo uma atitude, e não um modelo de processo. Muitas vezes, ele é integrado aos modelos de processos de engenharia (p. ex., abordagens ágeis ou em Waterfall, consulte a Seção 1.4.2) ou até mesmo é parte integrante dos processos de design (p. ex., Design Thinking, consulte a Seção 1.4.4) para abranger os aspectos que tratam do stakeholder mais importante – o usuário.

1. Plano para HCD

Planeje as atividades de HCD necessárias e garanta que elas sejam estabelecidas no plano geral do projeto (consulte também a próxima seção "Instanciação desse padrão"): As etapas são executadas iterativamente até que as metas comerciais e de usabilidade sejam atingidas.

2. Pesquisa com usuários (especifique o contexto de uso)

Primeiro, o Engenheiro de Requisitos identifica e analisa os usuários do sistema a ser desenvolvido. Essa análise do usuário deve capturar o contexto, as características, as metas, as atitudes e as tarefas de um número adequado de usuários reais (não de seus superiores!), agregá-los em grupos de usuários e documentá-los, por exemplo, na forma de uma persona por grupo de usuários.

Uma vez conhecidos os grupos de usuários, pode-se tomar uma decisão consciente sobre qual deles deve ser considerado como o principal grupo de usuários a ser apoiado pela solução. O sistema, e especialmente a sua interface de usuário, será otimizado para esses usuários.

Em seguida, é possível decidir se os outros grupos de usuários devem ser abordados por meio da mesma interface de usuário ou se devem ser fornecidas interfaces de usuário separadas. Pode até ser decidido que alguns grupos de usuários não serão atendidos pelo sistema.

3. Especificação de requisitos

O Engenheiro de Requisitos seleciona técnicas de elicitação que ajudarão a coletar informações sobre as tarefas realizadas pelos usuários e os produtos de trabalho que eles usam. Considerando os usuários, as tarefas e o contexto, os requisitos do sistema podem ser especificados.

4. Produzir soluções de design

Ao validar os requisitos com usuários reais, evite modelos complicados e documentos de especificação complicados que eles não entendam. É mais eficaz mostrar protótipos ou permitir que eles percorram cenários ou storyboards.

5. Avaliar o design

Aplice a avaliação de usabilidade de qualquer forma: por exemplo, o teste informal de corredor (uma breve apresentação do seu protótipo com alguns membros da equipe que passam pelo corredor e podem dedicar de 5 a 10 minutos do seu tempo) é uma técnica rápida, barata e informal ou, no outro extremo da escala, os testes formais de usabilidade baseados em laboratório com um número maior de usuários recrutados externamente.

1.4.3.4 Instanciação

Rua	Objetivo da Elicitação	Técnica de Elicitação	Fonte de Requisitos	Qualidade do Resultado
1	Para saber mais sobre as tarefas dos usuários (para se preparar para as consultas contextuais)	Entrevista	Especialistas específicos que sabem o que os usuários fazem (ou deveriam fazer): Gerentes, instrutores, engenheiros de processo, representantes de usuários, etc.)	Alta certeza / baixa completude

Rua	Objetivo da Elicitação	Técnica de Elicitação	Fonte de Requisitos	Qualidade do Resultado
2	Para saber mais sobre as tarefas dos usuários (para se preparar para as consultas contextuais)	Leitura baseada em perspectiva Estudar o material de treinamento, participar de cursos de treinamento Explore os sistemas do usuário (sistemas de treinamento, sistemas de teste)	Materiais de treinamento Sistema de treinamento Sistemas de teste	Alta certeza / baixa completude
3	Para identificar grupos de usuários no contexto	Investigação contextual (CI - contextual inquiry) Vários CIs	Alguns usuários específicos	Certeza média / baixa completude
4	Aprender sobre os usuários no contexto público (p. ex., usar um sistema de informações em um local público)	Observação de campo	Muitos usuários casuais	Certeza média / baixa completude
5	Para coletar dados quantitativos sobre questões específicas	Pesquisa com questionário	Muitos usuários em locais diferentes	Alta certeza
6	Para entender um negócio específico	Aprendizagem por alguns dias	Um ou dois usuários específicos	Alta certeza / baixa completude
7	Para validar os requisitos e encontrar novos requisitos	Walkthrough do usuário baseado em: - Prototipagem - Cenários de Uso - Storyboards	Usuários	Alta certeza / alta completude

Hint 1.4.1: Uma das primeiras lições de HCD para os membros do projeto de desenvolvimento é reconhecer que "eu não sou o usuário". Mesmo que o aplicativo seja direcionado a usuários como eu, os outros usuários são tão diferentes em termos de atitude, experiência, objetivos, tarefas etc., que não há como substituir o feedback direto deles. Até mesmo um usuário representativo que se junta à equipe do projeto fica, após um curto período de tempo, "contaminado" pelas discussões que ocorrem na equipe de desenvolvimento (p. ex., ele conhece o conceito que levou ao design de uma tela complicada), de modo que ele não é mais realmente representativo.

1.4.3.5 Leitura adicional

[RiFi2014] fornecem uma visão geral breve, mas muito prática, da engenharia centrada no ser humano, ou seja, a criação de produtos para seres humanos.

[HaPy2012] fornecem uma visão geral abrangente sobre UX.

[BeHo1998] são os inventores da pesquisa contextual.

1.4.4 Design Thinking

O Design Thinking é um processo formalizado para a resolução prática e criativa de problemas e para a criação de soluções, com a intenção de melhorar o resultado futuro. É uma forma de pensamento baseado em soluções ou focado em soluções, que começa com uma meta (uma situação futura melhor) em vez de resolver um problema específico. Existem várias abordagens de Design Thinking (consulte [A4qu2018]). Uma abordagem bastante difundida é a d.school do Hasso Plattner Institute of Design da Universidade de Stanford (consulte [Dsch2012]).

1.4.4.1 Escopo

O Design Thinking é útil para criar ideias de soluções alternativas e possivelmente requisitos, seja para a meta geral do projeto ou para algum aspecto específico (p. ex., determinados recursos do sistema). Portanto, o Design Thinking é aplicável logo no início de um projeto (p. ex., para desenvolver ideias inovadoras para uma loja on-line) ou durante o projeto, ao elaborar algum aspecto específico (p. ex., uma tela de visão geral do cliente em um sistema de administração de apólices). Uma alternativa a esse padrão é o padrão de design centrado no ser humano, também descrito neste handbook.

1.4.4.2 Esforço / recursos

O design thinking é um método escalável. Todo um processo de Design Thinking pode ser realizado em algumas horas, dias ou semanas, dependendo dos recursos disponíveis. A literatura (p. ex., [LiOg2011]) recomenda que a equipe de Design Thinking trabalhe em uma sala dedicada, pois a equipe produz vários resultados e, dependendo das técnicas aplicadas, precisa de espaço de trabalho (p. ex., para desenvolver storyboards ou outros tipos de técnicas de canvas).

1.4.4.3 Elementos de padrão

Na abordagem da d.school, o processo de Design Thinking consiste em cinco fases iterativas. Os participantes são os stakeholders no projeto e os Engenheiros de Requisitos. Um Engenheiro de Requisitos também é responsável por moderar e conduzir o processo. O Design Thinking enfatiza uma configuração multidisciplinar para os participantes, por exemplo: usuários de diferentes idades, representantes da empresa e de equipes técnicas, pessoas de outros grupos relacionados ao projeto (p. ex., ONGs). Do ponto de vista da Engenharia de Requisitos, recomendamos a realização de atividades de elicitação que analisem sistematicamente o contexto do projeto para identificar um grupo ideal de participantes para o processo de Design Thinking.

Embora as fases sejam apresentadas sequencialmente, o processo de Design Thinking enfatiza a importância das iterações. Se os resultados de uma fase levarem a uma melhor compreensão ou modificação dos resultados já desenvolvidos, o processo deverá voltar à fase correspondente e modificar os resultados. Por exemplo, se a fase de protótipo criou resultados que permitem uma melhor definição do objetivo do projeto, o processo deve voltar à fase de definição.

A seguir, descreveremos cada fase da abordagem d.school em mais detalhes, incluindo diretrizes para a definição das atividades de elicitação.

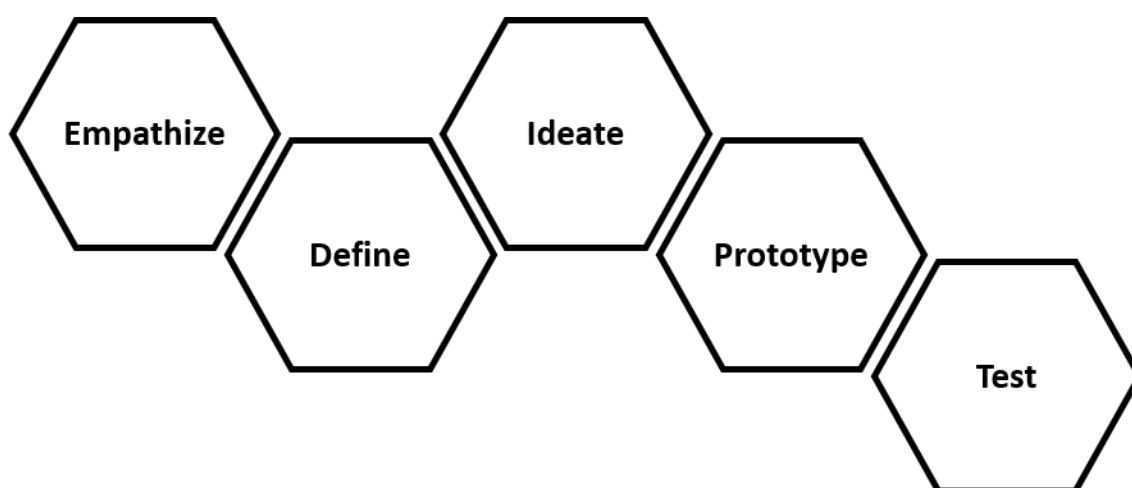


Figura 7: abordagem do Design Thinking da d.school [Dsch2012]

Empatia

O objetivo dessa fase é entender as pessoas que são afetadas pelo resultado do projeto (p. ex., os usuários do software), entender sua maneira de trabalhar, suas necessidades físicas ou emocionais ou o ambiente em que trabalham.

Do ponto de vista da ER, essa fase se concentra na definição do contexto do sistema e na identificação dos stakeholders (e outras fontes de requisitos). Para realizar essa fase, você precisa definir atividades de elicitacão com objetivos de elicitacão voltados para a compreensão das pessoas no contexto do seu sistema. As fontes de requisitos são stakeholders que estão fora da equipe de Design Thinking. A qualidade do resultado depende do objetivo geral do projeto: no entanto, todas as informações que contribuam para a compreensão dos stakeholders são bem-vindas.

As técnicas possíveis para essa fase são entrevistas qualitativas (consulte a Seção 3.1.1.1), workshops de requisitos (consulte a Seção 3.1.3.1) e técnicas de observação (consulte a Seção 0). As técnicas baseadas em artefatos (consulte a Seção 3.1.3) também são úteis caso você precise entender melhor o domínio ou a tecnologia. Uma maneira adequada de documentar suas descobertas seria por meio de personas (consulte a Seção 2.2.4)

Definir

Processe e sintetize as descobertas do seu trabalho de empatia para formar um ponto de vista do usuário que será abordado no seu projeto.

Do ponto de vista da Engenharia de Requisitos, essa fase é direcionada para a documentação dos resultados obtidos. Um resultado importante é uma definição concreta do objetivo do projeto dentro da sua equipe de Design Thinking. Você pode definir atividades de elicitacão com objetivos de elicitacão voltados para a compreensão de relacionamentos e dependências. A qualidade do resultado deve ser muito alta, pois os resultados devem refletir o entendimento comum da sua equipe de projeto. As fontes de requisitos para essas atividades são os participantes de seu processo de Design Thinking. Técnicas úteis incluem, por exemplo, o mapeamento mental (consulte a Seção 3.3.5) e storyboards (consulte a Seção 3.2.4), que ajudam a reestruturar e visualizar os resultados.

Idealizar

Explore uma ampla variedade de soluções possíveis por meio da geração de uma grande quantidade de soluções diversas, permitindo que você vá além do óbvio e explore uma série de ideias.

Do ponto de vista da Engenharia de Requisitos, essa fase é uma fase de criatividade na qual você define atividades de elicitacão com objetivos de elicitacão que se concentram na geração de um grande número de ideias diferentes.

As fontes de requisitos para essas atividades são os participantes da sua equipe de Design Thinking. O Design Thinking enfatiza a importância de gerar várias ideias diferentes para resolver o problema em questão. A qualidade do resultado é flexível e depende dos recursos disponíveis e dos limites de tempo. As técnicas úteis incluem, por exemplo, o brainstorming (consulte a Seção 0).

Protótipo

Transforme suas ideias mais promissoras em forma física para que você possa experimentar e interagir com elas e, no processo, aprender e desenvolver mais empatia.

Do ponto de vista da Engenharia de Requisitos, essa fase trata de aprender com o desenvolvimento de protótipos. Você define atividades de elicitacão com objetivos de elicitacão que se concentram na elaboracão de ideias da fase de ideação. As fontes de requisitos para essas atividades são os participantes da sua equipe de Design Thinking. A qualidade do resultado depende do projeto e, principalmente, do tempo e dos recursos disponíveis. Se o seu projeto tiver um orçamento baixo e restrições de tempo, você deve desenvolver protótipos simples e baratos. O Design Thinking fornece uma regra prática útil para protótipos: Não gaste muito esforço em um protótipo, caso contrário, você pode se apegar demais a uma determinada ideia e introduzir preconceitos na avaliação. Técnicas úteis são todos os tipos de prototipagem (consulte 3.2.3), por exemplo, protótipos de papel e lápis, mock-ups ou wireframes.

Teste

Experimente protótipos de alta fidelidade (consulte a Seção 3.2.3) e use observações e feedback para aperfeiçoar os protótipos, aprender mais sobre o usuário e refinar seu ponto de vista original.

Do ponto de vista da Engenharia de Requisitos, essa fase é uma mistura de validação e elicitacão de requisitos. Você define as atividades de elicitacão com objetivos de elicitacão que se concentram em obter feedback dos stakeholders. As fontes de requisitos para essas atividades são stakeholders fora da sua equipe de Design Thinking. Esse feedback neutro é importante, pois os participantes da sua equipe estavam envolvidos no desenvolvimento das ideias e poderiam ser tendenciosos.

A qualidade dos resultados deve ser alta, pois o feedback dos stakeholders é muito importante para aprimorar ainda mais as ideias desenvolvidas. As técnicas úteis incluem, por exemplo, entrevistas quantitativas (consulte 3.1.1.1) e workshops (consulte 3.1.3.1) nos quais os protótipos são apresentados.

1.4.4.4 Instanciação

Para iniciar um processo de Design Thinking, primeiro é preciso identificar sua equipe de Design Thinking e planejar detalhadamente a fase de empatia. No início, seu plano poderia ser assim:

Rua	Objetivo da Elicitação	Técnica de Elicitação	Fonte de Requisitos	Qualidade do Resultado
Atividades de curto prazo				
1	Identificar os membros do Grupo de Design Thinking	Entrevista	patrocinador do projeto / chefe do departamento técnico / ...	Alta certeza / alta completude
2	Compreender as pessoas no contexto do sistema: Usuário (fase de empatia)	Entrevista	Usuário do sistema	Alta certeza
3	Compreender as pessoas no contexto do sistema: usuário	Observação	Usuário do sistema	Alta certeza
4	Entenda as pessoas no contexto do sistema: superiores dos usuários	Oficina	Superior de usuários	Alta certeza
Atividades de longo prazo				
5	Definir fase	Workshops	Equipe de Design Thinking	TBD
6	Fase de idealização	Técnicas de criatividade	Equipe de Design Thinking	TBD
7	Fase de protótipo	Protótipos	Equipe de Design Thinking	TBD
8	Fase de teste	TBD*	TBD*	TBD*

*Os TBDs nesta tabela significam que você não pode ou não deve decidir esses aspectos no início do processo de Design Thinking porque eles dependem muito dos resultados das outras fases. Tenha em mente também a natureza iterativa do processo de Design Thinking e que as atividades de longo prazo são definidas de forma abstrata e precisam ser detalhadas no decorrer do processo.

1.4.4.5 Leitura adicional

[Brown2009] oferece uma visão geral das ideias por trás do Design Thinking e seu potencial.

[Design Council 2007] explica o modelo Double Diamond que foi publicado em 2005 pelo British Design Council.

[KnZK2016] apresentam uma versão compacta do Design Thinking em cinco dias. O livro é estruturado ao longo dos cinco dias e fornece orientações metodológicas concretas sobre o que fazer em cada dia.

[LeLL2018] fornece uma coleção útil de dicas, exemplos e técnicas para profissionais de Design Thinking.

2 Fontes de Requisitos

Este capítulo aborda os três tipos de fontes de requisitos e como eles podem ser identificados, classificados e gerenciados. A seção 2.1 explica a abordagem pragmática e sistemática para a identificação das fontes de requisitos. A seção 2.2 abrange a identificação, a classificação e o gerenciamento dos stakeholders, e as seções 2.3 e 2.4 fazem o mesmo para documentos e sistemas.

2.1 Fundamentos de fontes de requisitos

A qualidade e a completude dos requisitos dependem muito das fontes de requisitos envolvidas. A falta de uma fonte relevante levará a uma compreensão incompleta dos requisitos e aumentará o risco de sua empreitada. Por exemplo, a falta de envolvimento do usuário aumenta o risco de baixa aceitação do novo sistema, negligenciar uma stakeholder importante pode fazer com que o projeto seja bloqueado em uma fase crítica e ignorar a funcionalidade existente de um sistema legado pode fazer com que os requisitos básicos sejam ignorados. Portanto, no decorrer do desenvolvimento, o Engenheiro de Requisitos precisa identificar e consultar todas as fontes de requisitos relevantes.

Os três tipos mais importantes são [IREB2020]:

- Stakeholders
- Documentos
- Sistemas

Como nenhum requisito vem sem uma fonte, é naturalmente uma das primeiras atividades na elicitação de requisitos identificar as possíveis fontes de requisitos. Esse é um processo iterativo: não basta identificar essas fontes apenas no início do desenvolvimento de um projeto ou produto, mas sim um processo que deve ser repetido várias vezes. Com cada nova atividade de elicitação e um conhecimento cada vez maior do produto a ser construído, novas fontes potenciais de requisitos podem ser identificadas, levando, por sua vez, a outras atividades de elicitação e à identificação de fontes em um processo recursivo.

Exemplo

Você está entrevistando um usuário em potencial (viajante do tempo) da máquina do tempo que está prestes a desenvolver. Durante a entrevista, a viajante do tempo menciona que seu colega geralmente executa a ação sobre a qual você acabou de perguntar e que ela teria que procurar os detalhes na documentação do processo de viagem no tempo.

O viajante do tempo acabou de revelar duas possíveis fontes de requisitos que você deve levar em consideração.

Então, você pergunta o nome da colega e pede detalhes sobre a documentação do processo de viagem no tempo que ela acabou de mencionar.

Após a entrevista, você verifica em sua documentação de fontes de requisitos (consulte 2.3.2) se o colega do entrevistado e a documentação do processo de viagem no tempo já foram identificados ou se são novas fontes de requisitos em potencial.

Ao interagir com uma fonte de requisitos previamente identificada, você identificou outra. Isso o torna um processo recursivo.

Esteja preparado para identificar novas fontes de requisitos mesmo durante os estágios finais do projeto, quando, normalmente, uma sensação de "nós sabemos quem eles são e o que eles querem" está pairando sobre os participantes do projeto.

Você deve revisar constantemente sua documentação de fontes de requisitos e reavaliar se ela ainda está atualizada, se novas fontes de requisitos podem ser relevantes ou se algumas fontes identificadas anteriormente se tornaram obsoletas.

Identificação pragmática e sistemática das fontes de requisitos

Distinguimos duas abordagens básicas para a identificação de fontes de requisitos:

- Identificação pragmática
- Identificação sistemática

Identificação pragmática de fontes de requisitos

"Identificação pragmática" é um termo sofisticado para usar sua intuição e experiência. Se você já esteve envolvido em um projeto dentro do mesmo departamento antes, ou em um projeto semelhante em um contexto de negócio diferente, poderá citar uma lista de possíveis participantes, documentos e sistemas sem pensar muito.

Esse é um importante meio de identificação de possíveis fontes de requisitos; com um pequeno investimento de tempo, você obtém uma lista de possíveis fontes de requisitos para começar. Posteriormente, no desenvolvimento do produto, você poderá encontrar outras fontes de requisitos potenciais.

No entanto, é perigoso confiar na identificação pragmática como a única abordagem, pois você pode perder fontes de requisitos cruciais. Você deve sempre fazer o backup e acrescentar aos seus resultados da identificação pragmática usando a identificação sistemática.

Identificação sistemática das fontes de requisitos

A identificação sistemática dos requisitos consiste em duas etapas:

1. Determinar critérios que possam caracterizar as fontes de requisitos e, assim, contribuir para sua identificação.
2. Pesquisa sistemática com base nos critérios (aplique o princípio da bola de neve³)

Para a identificação sistemática, é necessário definir atividades de elicitação (consulte 1.3) com objetivos de elicitação focados na identificação de fontes de requisitos. Dessa forma, você divide o objetivo abstrato de encontrar todas as fontes de requisitos relevantes em tarefas específicas e acionáveis.

³ Quando você identificar um novo interessado, peça a ele que forneça outras fontes de requisitos.

Figura 8 os sites Figura 10 mostram exemplos de atividades de elicitação para identificar os stakeholders na construção de uma máquina do tempo.

ID	RS_EA_01
Objetivo da elicitação	Encontre pelo menos 10 possíveis viajantes do tempo (usuários)
Qualidade do resultado	Nome do usuário, função e dados de contato
Fontes de requisitos	Organograma
Técnica de elicitação	Leitura baseada em perspectiva

Figura 8: Exemplo de uma atividade de elicitação para a identificação de usuários como stakeholders

ID	RS_EA_02
Objetivo da elicitação	Encontre pelo menos 5 documentos legais potencialmente relevantes para uma máquina do tempo
Qualidade do resultado	Nome do documento, versão atual, local onde pode ser encontrado
Fontes de requisitos	Evelyn Hall, Bob Miller e o consultor jurídico/advogado da empresa em desenvolvimento
Técnica de elicitação	Entrevista

Figura 9: Exemplo de uma atividade de elicitação para a identificação de documentos legais

ID	RS_EA_03
Objetivo da elicitação	Descubra: quem é o consultor jurídico/advogado da empresa em desenvolvimento
Qualidade do resultado	Nome, dados de contato, disponibilidade
Fontes de requisitos	Departamento jurídico
Técnica de elicitação	Entrevista por telefone

Figura 10: Exemplo de uma atividade de elicitação para a identificação de um consultor jurídico ou advogado

Neste exemplo, ao planejar a atividade de elicitação (EA) RS_EA_02 de fontes de requisitos (RS), a engenheira de requisitos percebeu que ainda não havia identificado quem é o consultor jurídico da empresa para o projeto de desenvolvimento. Portanto, ela acrescentou outra atividade de elicitação (RS_EA_03) para abordar essa questão.

Você deve sempre aplicar tanto a identificação pragmática quanto a sistemática. A identificação pragmática economiza tempo e recursos, mas apresenta o risco de viés subjetivo. Na maioria dos casos, a identificação pragmática não será suficiente por si só. A identificação sistemática, por outro lado, exige mais tempo e recursos do que a identificação pragmática, mas também leva a uma lista de fontes de requisitos mais objetiva e potencialmente mais completa.

Sugerimos começar com a identificação pragmática e, em seguida, fazer o backup e corrigir seus resultados usando a identificação sistemática.

2.2 Identificar, classificar e gerenciar os stakeholders

Esta seção se concentra na principal fonte de requisitos para a maioria dos projetos: os stakeholders. Os Stakeholders são seres humanos ou pessoas jurídicas no contexto de seu projeto que influenciam o projeto e que são afetadas por ele. Como Engenheiro de Requisitos, você pode conhecê-los, observá-los, conversar com eles e fazer perguntas. As pessoas jurídicas são representadas por pessoas físicas (p. ex., a empresa WeAreTheBest Ltd é representada por seu CEO ou por um porta-voz).

Obviamente, os stakeholders podem ser agregados em grupos ou funções, como usuários, equipe de manutenção, testadores, etc. Esse conceito de grupos ou funções de stakeholders é muito útil para muitas tarefas da Engenharia de Requisitos. Lembre-se, no entanto, de que você não pode interagir com funções abstratas, mas, em última análise, sempre precisa de um ser humano real para se comunicar.

E você não costuma ter problemas com grupos abstratos, mas com indivíduos (p. ex., o Sr. Fudge, chefe do departamento de autorização de viagens, que não foi convidado a dar sua opinião sobre os requisitos do sistema de viagem no tempo).

Hint 2.2.1:

É uma boa prática nomear stakeholders individuais para cada grupo de stakeholders, especialmente para grupos internos importantes, como o departamento jurídico, agentes de segurança, gerentes, representantes de departamentos relacionados, etc.

A seção 2.2.1 explica como os stakeholders podem ser identificados de forma sistemática e pragmática; a seção 2.2.2 trata do gerenciamento do relacionamento com os stakeholders (incluindo a classificação);

A seção 2.2.3 abrange a documentação dos stakeholders e

A seção 2.2.4 concentra-se no usuário como uma função específica de stakeholder.

2.2.1 Identificação e seleção de stakeholders como fontes de requisitos

os stakeholders desempenham um papel fundamental na maioria dos projetos de desenvolvimento. Portanto, identificá-los e gerenciá-los é uma tarefa importante durante a elicitacão de requisitos.

2.2.1.1 Abordagem pragmática para a identificação dos stakeholders

É comum identificar os stakeholders de forma pragmática. Com base em sua experiência no contexto do projeto (p. ex., em projetos anteriores na mesma organização) e pela reutilização de listas de stakeholders existentes, o Engenheiro de Requisitos monta rapidamente uma lista inicial de grupos e funções de stakeholders (geralmente já inclui representantes individuais).

2.2.1.2 Abordagem sistemática para a identificação de grupos e funções de stakeholders

Embora a identificação pragmática seja uma maneira útil de iniciar a identificação dos stakeholders, ela deve sempre ser apoiada por uma identificação sistemática dos stakeholders.

Fontes e técnicas úteis para a identificação sistemática dos stakeholders são:

- **Listas de verificação de grupos e papéis típicos de stakeholders**
Listas como o mapa metafórico do país dos stakeholders de [Rupp et al.2014] ou a lista simples a seguir podem ser usadas para qualquer projeto como uma diretriz:
 - Usuários diretos do sistema,

- Gerentes de negócios/processos,
- Clientes e clientes individuais ou organizações que representam clientes,
- Oponentes e concorrentes,
- Equipe de TI,
- Instituições governamentais e regulatórias.

Examine cada função de participante da lista e esclareça se é ou não uma função relevante para o seu projeto. Além disso, faz sentido criar sua própria lista de verificação ao longo do tempo, principalmente se estiver trabalhando em um campo específico (p. ex., no setor de seguros), em que as funções dos participantes necessárias para cada projeto podem ser muito semelhantes.

- **Estruturas organizacionais**

(p. ex., organogramas da empresa que usará o sistema a ser construído)

A maioria das empresas tem organogramas. Eles podem ser úteis para encontrar os diferentes departamentos de uma empresa e os stakeholders nesses departamentos.

- **Documentação do processo de negócio**

Embora possam não estar totalmente atualizados, muitas empresas documentaram seus processos de negócios de alguma forma. Isso pode ser um modelo de processo de negócios (p. ex., modelo BPMN) ou uma descrição em linguagem natural de como o processo funciona. Com essa documentação, você poderá descobrir qual função é responsável por uma tarefa específica. Essa documentação é, por si só, uma fonte potencial de requisitos (consulte 2.3).

- **Esquemas de categorização de stakeholders**

Os esquemas de categorização de participantes existentes também são fontes valiosas para a identificação de funções de participantes potencialmente relevantes. Eles citam categorias e funções típicas dos participantes.

Os exemplos são: O modelo cebola de Alexander [Alexander2005] – consulte a seção subsequente para obter mais detalhes – ou o mapa genérico de stakeholders de Robertson [RoRo2013].

- **Foco nas informações**

Tente identificar as informações ou os campos de conhecimento relevantes e os grupos de stakeholders que possam fornecer essas informações ou que sejam especialistas nesses campos.

Exemplo:

ID	RS_EA_04
Objetivo da elicitação	Encontre pelo menos dois stakeholders que possam me fornecer informações sobre as restrições físicas da viagem no tempo.
Qualidade do resultado	Nome, função e dados de contato dos stakeholders
Fontes de requisitos	Organograma
Técnica de elicitação	Leitura baseada em perspectiva

Figura 11: Exemplo de uma atividade de elicitação com foco em informações

▪ **Análise do ciclo de vida do produto**

Faça um passeio virtual pelo ciclo de vida do produto a ser desenvolvido. Quem interagirá com o produto ou sua documentação de alguma forma, desde os requisitos, passando pelo desenvolvimento e uso, até a desinstalação ou desconstrução?

Isso revelará outros departamentos (p. ex., produção e manutenção) ou organizações (p. ex., usuários de segunda mão, fornecedores ou empresas de reciclagem) como possíveis interessados.

Hint 2.2.2:

Tem se mostrado útil realizar um workshop para identificar os stakeholders, reunindo todas as funções, nomes e campos de conhecimento dos stakeholders em potencial que vêm à mente dos participantes. Na próxima etapa, são identificadas as triplas de nome, função(ões) dos participantes e campo(s) de conhecimento, bem como as possíveis lacunas.

2.2.1.3 Abordagem sistemática para a identificação de stakeholders individuais (pessoas)

Conforme observado anteriormente, é importante identificar indivíduos tangíveis e passíveis de contato. Esses stakeholders têm nomes, dados de contato e datas de aniversário. Esse último pode não ser de interesse primordial para as atividades de Engenharia de Requisitos, mas é uma informação útil quando se trata de gerenciamento do relacionamento com os stakeholders.

Durante a identificação sistemática dos stakeholders individuais, o Engenheiro de Requisitos define as atividades de elicitação correspondentes.

Exemplo:

ID	RS_EA_05
Objetivo da elicitação	Encontre pelo menos uma pessoa para a função de participante "Treinamento do viajante do tempo"
Qualidade do resultado	Nome e dados de contato dos stakeholders
Fontes de requisitos	Dr. Emmet Brown (viajante do tempo experiente, ele pode conhecer candidatos adequados)
Técnica de elicitação	Entrevista

Figura 12: Exemplo de uma atividade de elicitação para identificar stakeholders individuais

2.2.2 Gerenciamento do relacionamento com os stakeholders

Os problemas com os stakeholders surgem normalmente se os direitos e obrigações de deles, no que diz respeito ao sistema proposto ou ao projeto atual, não forem claros ou se as necessidades dos stakeholders não forem suficientemente atendidas. A gestão do relacionamento com os stakeholders é uma forma eficaz de combater os problemas com os stakeholders. Para envolver os stakeholders no processo de elicitação, precisamos garantir que elas saibam do que se trata o projeto e qual é a função delas no projeto.

O círculo de stakeholders [Bourne2015] é uma estrutura útil para o gerenciamento bem-sucedido do relacionamento com os stakeholders. Ele consiste nas cinco etapas a seguir:

1. Identificar todos stakeholders
2. Priorizar para determinar quem é importante
3. Visualizar para entender a comunidade geral de stakeholders
4. Engajar por meio de comunicação eficaz
5. Monitorar o efeito da comunicação

A estrutura de Bourne é voltada para o gerenciamento dos stakeholders do projeto. Modificamos ligeiramente os detalhes das etapas individuais para nos adequarmos ao contexto do gerenciamento de requisitos dos stakeholders e aos programas de estudos do IREB.

Etapa 1: Identificação de todos os stakeholders

O círculo de stakeholders começa com a identificação de todas os stakeholders. Essa etapa é descrita na Seção 2.2.1.

Etapa 2: Priorização para determinar quem é importante

Nem todos os stakeholders são iguais. Alguns são mais importantes do que outros e alguns são mais importantes no início de um projeto do que no final.

Como temos recursos limitados disponíveis para o gerenciamento dos stakeholders e a elicitación de requisitos, é fundamental priorizar os stakeholders identificadas. Dependendo do projeto, diferentes esquemas de priorização podem fazer sentido. Antes de escolher um esquema de priorização, pergunte a si mesmo:

- Qual é a vantagem desse esquema?
- Como as informações fornecidas são úteis para o projeto?
- Qual é a vantagem desse esquema em relação aos outros esquemas que poderíamos usar?
- Como isso ajuda a identificar quem é realmente importante para o nosso projeto?

A primeira etapa antes da priorização é a classificação. Quando todos os participantes são classificados de acordo com um determinado esquema, essas classes podem ser priorizadas umas em relação às outras. Muitas vezes, também faz sentido priorizar ainda mais em uma classe identificada.

Um exemplo de esquema de classificação é o Onion Model de Ian Alexander [Alexander2005]. Ele usa três classes:

- **Stakeholders do sistema:** Esses stakeholders são diretamente afetados pelo sistema novo ou modificado. Exemplos típicos dessa classe são os usuários, a equipe de manutenção e os administradores de sistemas.
- **Stakeholders do sistema de contenção:** Esses stakeholders são indiretamente afetados pelo sistema novo ou modificado. Exemplos típicos dessa classe são os gerentes de usuários, project owners ou patrocinadores.
- **Stakeholders do ambiente mais amplo:** Esses stakeholders têm uma relação indireta com o sistema novo ou modificado. Exemplos típicos dessa classe são legisladores, órgãos de definição de padrões, organizações não governamentais (ONGs, por exemplo, sindicatos ou associações de proteção ambiental), concorrentes e também os membros do projeto, que estão envolvidos no desenvolvimento do sistema, mas não serão afetados pelo sistema em uso produtivo.

os stakeholders também podem ser classificados de acordo com sua influência no projeto (alta ou baixa influência) e sua motivação em relação ao projeto (forte ou fraca motivação). os stakeholders com grande influência podem, por exemplo, obstruir o projeto ou promovê-lo. os stakeholders com maior motivação são, por exemplo, valiosos para o projeto, porque elas próprias têm interesse em avançar no projeto [Rupp et al.2014].

Outros exemplos de atributos de classificação dos participantes são:

- Proximidade
- Disponibilidade
- Interesse
- Potência
- Experiência em projetos semelhantes

- Habilidades de comunicação

Etapa 3: Visualização para entender a comunidade geral de stakeholders

Todas as informações coletadas sobre os stakeholders devem ser documentadas (ou seja, visualizadas) e essa documentação deve ser mantida atualizada. Essa representação visual dos stakeholders ajuda a entender a comunidade geral de stakeholders e a garantir que nenhuma stakeholder importante seja ignorada.

Consulte a Seção 2.2.3 para obter detalhes sobre a visualização/documentação dos stakeholders.

Etapa 4: Engajamento por meio de comunicação eficaz

Agora é importante entender o que cada stakeholder espera em termos de comunicação e qual é sua atitude em relação ao projeto. Os seguintes elementos devem ser considerados [Bourne2015]:

- Cultura (organizacional, de equipe ou individual)
- Identificação com a atividade (de desenvolvimento) e seus resultados,
- Importância percebida da atividade e de seus resultados
- Atributos pessoais, como personalidade e função.

Depois de avaliar as atitudes dos stakeholders, você precisa definir uma "atitude-alvo realista" para cada uma delas. Essa atitude-alvo deve servir tanto para a stakeholder quanto para o projeto de desenvolvimento.

Existe uma lacuna entre a atitude atual e a atitude desejada? Em caso afirmativo, você deve definir atividades destinadas a fechar essa lacuna (plano de comunicação).

As perguntas a seguir podem ser úteis para a etapa de engajamento:

- Com que frequência devo entrar em contato com stakeholders específicas para informá-las sobre o status em relação à elicitação de requisitos?
- Quais informações são relevantes para cada stakeholder?
- Qual é a melhor maneira de manter um acionista específico atualizado? (p. ex., ligação telefônica, e-mail, boletim informativo, almoço)
- Qual é a melhor maneira de entrar em contato com essa stakeholder se eu precisar de informações?
- Como posso saber quando entrei em contato com esse interessado pela última vez?
- Há algum fator cultural relevante para a comunicação?

Hint 2.2.3:

O gerenciamento dos stakeholders nos requisitos e o gerenciamento dos stakeholders no projeto tendem a se sobrepor. Certifique-se de coordenar com a pessoa responsável pelo gerenciamento dos stakeholders do projeto como você lidará com essa sobreposição.

O gerenciamento do relacionamento dos requisitos com os stakeholders concentra-se no gerenciamento do relacionamento para obter os requisitos necessários de uma stakeholder. O gerenciamento do relacionamento com os stakeholders do projeto geralmente tem uma visão mais ampla do gerenciamento do relacionamento com os stakeholders.

Etapa 5: Monitoramento do efeito da comunicação

Repita a avaliação da etapa 4 regularmente para identificar onde são necessárias mais atividades de gerenciamento do relacionamento com os stakeholders e para verificar se as atividades realizadas foram eficazes. Ajuste o plano de comunicação conforme necessário para continuar fechando a lacuna entre as atitudes atuais e as atitudes dos acionistas-alvo.

2.2.3 Esquema de documentação para os stakeholders envolvidas

Todas as informações coletadas durante a identificação e o gerenciamento do relacionamento com os stakeholders devem ser suficientemente documentadas. Essa documentação deve incluir pelo menos o seguinte para cada stakeholder:

- Nome
- Função (cargo)
- Dados de contato
- Disponibilidade (espacial e temporal)
- Relevância
- Área e tempo de experiência
- Metas e interesses com relação ao projeto

Além disso, as informações de classificação (p. ex., de acordo com Ian Alexander) e a prioridade da etapa 2, bem como as informações obtidas na etapa 4 (incluindo o plano de comunicação), devem ser documentadas.

Dependendo do projeto, informações adicionais podem ser relevantes. Os fatores de influência podem ser:

- *Relevância pública*: Em um contexto com maior relevância pública, pode ser útil documentar o quanto um stakeholder sabe ou pode influenciar a opinião pública.
- *Criticidade de tempo*: Em um contexto com um prazo muito restrito, a disponibilidade ou o tempo de resposta de um participante pode ser uma informação muito importante quando decisões críticas precisam ser tomadas.

Certifique-se de usar uma forma de documentação que atenda às necessidades de documentação e de gerenciamento do relacionamento com os stakeholders da configuração do seu projeto.

As formas de documentação comumente usadas são:

- Tabela de stakeholders
- Banco de dados de stakeholders (geralmente incorporado à ferramenta de gerenciamento de requisitos)

- Mapa mental dos stakeholders

Além disso, diagramas ou outras representações gráficas podem ser usados para rastrear mudanças nas atitudes ou prioridades dos participantes.

A documentação dos stakeholders deve ser mantida atualizada enquanto houver necessidade de informações sobre os stakeholders (ou seja, geralmente pelo menos até o final do projeto de desenvolvimento ou até o final do ciclo de vida do produto).

2.2.4 O usuário como um grupo especial de stakeholders

Para sistemas interativos com uma interface de usuário, todos os usuários diretos do sistema são de interesse primordial para o Engenheiro de Requisitos.

Os usuários internos (dentro da empresa, conhecidos e envolvidos individualmente) são significativamente diferentes dos usuários externos (p. ex., compradores de produtos de consumo; fora da empresa, geralmente não conhecidos individualmente e não envolvidos diretamente).

Normalmente, o número de usuários potenciais não permite envolver todos os indivíduos no processo de elicitación. Por esse motivo, os usuários reais podem ser agregados em grupos de usuários, com base na análise do usuário ou no conhecimento de domínio de outros stakeholders.

Uma forma comum de representar grupos de usuários é o uso de personas [Cooper2004]. Personas são indivíduos fictícios, que representam grupos de usuários típicos do sistema com necessidades, metas, comportamentos ou atitudes semelhantes. As personas são modeladas a partir de dados coletados sobre usuários reais por meio de pesquisas com usuários [BaCC2015]. Se nenhum dado de pesquisa de usuário relevante estiver (ainda) disponível, personas provisórias, também chamadas de personas ad-hoc [CRCN2014] ou proto-personas [Gothelf2013], pode ser criado.

As personas também podem ser criadas com base em dados brutos coletados por investigação contextual, entrevistas, pesquisas ou aprendizado (consulte a descrição desses métodos na Seção 2.2.2), conforme descrito em [Goodwin2009]. O conceito central da criação de personas é a identificação de variáveis bipolares que caracterizam e diferenciam as personas.

Os grupos de usuários ou personas resultantes devem ser priorizados como primários ou secundários. O sistema, especialmente sua interface de usuário, será otimizado para o grupo de usuários principal. Os grupos de usuários secundários são suportados apenas na medida em que o atendimento às suas necessidades não comprometa a experiência do usuário dos grupos primários.

Se os usuários forem os principais interessados, aplique o padrão de design centrado no ser humano para atender às expectativas deles (consulte a Seção 0).

Hint 2.2.4: A abordagem ágil do desenvolvimento de software concentra-se no valor que a solução deve oferecer aos usuários ("história do usuário"). Portanto, o usuário de um sistema interativo é considerado a principal stakeholder. No entanto, existe o risco, principalmente em grandes organizações, de que as equipes ágeis ou os product owners não estejam em contato com os usuários reais. As personas podem ter sido inventadas por alguém que finge conhecer os usuários! Nesse contexto, os stakeholders da organização (gerentes, departamentos comerciais, departamento jurídico, marketing, etc.) estão muito mais presentes e dominantes do que o usuário final externo. Esteja ciente dessa armadilha: insista no acesso direto aos usuários finais para realizar uma pesquisa de usuário adequada e na coleta de feedback direto após a entrega do sprint para realmente conhecer seus usuários e suas necessidades.

2.3 Identificar, classificar e gerenciar documentos

Os documentos são usados para transferir informações entre humanos ao longo do tempo e à distância. Muitas vezes, os requisitos podem ser derivados de documentos, o que os torna uma fonte valiosa.

Na Seção 2.3.1, discutimos a identificação e a seleção de tais documentos e, na Seção 2.3.2, abordamos a documentação de documentos como fontes de requisitos.

2.3.1 Identificação e seleção de documentos como fontes de requisitos

Os requisitos podem ser derivados de muitos tipos diferentes de documentos. Dependendo do caráter específico do projeto de desenvolvimento, os documentos podem ter alta, média ou baixa importância como fontes de requisitos. Normalmente, os projetos técnicos de engenharia de sistemas (p. ex., transmissão de dupla embreagem) têm muitos documentos como fontes de requisitos (p. ex., padrões técnicos, patentes), enquanto os projetos centrados no ser humano (p. ex., um aplicativo móvel de compras) têm menos.

Os possíveis tipos de documentos que podem ser usados como fontes de requisitos são (não exaustivos):

- **Normas técnicas, legislação, regulamentos internos**

Todos os projetos de desenvolvimento têm restrições legais de acordo com a legislação civil e criminal (nacional e, possivelmente, também internacional).

Normalmente, as leis de privacidade de dados são relevantes para muitos projetos de desenvolvimento. Nos setores médico e alimentício, a U. S. As regulamentações da Food and Drug Administration (FDA) são de grande importância, enquanto no setor automotivo aplicam-se normas internacionais específicas, como a ISO 26262 sobre segurança funcional. Esses são apenas alguns exemplos; cada setor tem seu próprio conjunto de normas e leis aplicáveis. Além disso, cada empresa pode ter regulamentos internos que podem conter requisitos para o projeto de desenvolvimento (p. ex., guia de estilo).

- **Documentos de requisitos**

Nem todos os sistemas são desenvolvidos como projetos de campo verde. Muitas vezes, um ou mais sistemas predecessores devem ser substituídos ou uma variante de um sistema existente é desenvolvida (p. ex., um novo sistema de gerenciamento de contratos para uma seguradora que se fundiu recentemente com um concorrente).

Nesses casos, os documentos de requisitos de sistemas anteriores podem ser fontes valiosas para novos requisitos, por exemplo, regras comerciais, glossário, modelos de objetos comerciais, casos de uso, especificações de fluxo de trabalho, documentos de autenticação e autorização etc.

Os documentos de requisitos dos sistemas de interface também podem ser relevantes.

- **Manuais do usuário**

Se houver sistemas predecessores ou concorrentes, os manuais do usuário podem estar disponíveis. Eles fornecem uma boa visão geral da funcionalidade do sistema e podem estar mais atualizados⁴ do que a documentação de requisitos desses sistemas.

- **Documentos de estratégia**

Se o sistema a ser desenvolvido fizer parte de uma estratégia geral da empresa, é possível que existam documentos ou apresentações estratégicas que possam incluir requisitos relevantes para o sistema.

- **Documentação do objetivo**

Antes de iniciar o desenvolvimento de um sistema, geralmente já existem vários documentos que avaliam a necessidade e os possíveis benefícios do projeto. Essa documentação geralmente inclui as metas do projeto, que são uma fonte essencial de requisitos.

- **Documentação do processo de negócio**

Se o sistema a ser desenvolvido estiver relacionado a uma ou mais etapas de um processo de negócio, qualquer documentação do processo de negócio será uma fonte valiosa de requisitos. Essa documentação pode ser encontrada em diferentes formas: por exemplo, documentos de texto simples, apresentações de slides, páginas de intranet, sites wiki, BPMN ou outros modelos.

- **Interface de documentação**

A maioria dos sistemas interage com outros sistemas. Qualquer documentação de interface para esses sistemas de interface pode conter informações relevantes para o sistema a ser desenvolvido.

⁴ Um bom gerenciamento de requisitos deve evitar esse cenário. Infelizmente, nem todas as empresas e projetos atingiram ainda esse nível de maturidade do gerenciamento de requisitos.

- **Documentos gerados pelo processo de negócios**
Em muitos processos de negócio, são gerados documentos como contratos de seguro, contas ou listas de clientes. Esses documentos geralmente contêm informações valiosas para qualquer sistema desenvolvido para dar suporte a esse processo.
- **Documentos gerados na análise técnica**
O desenvolvimento de sistemas, incluindo componentes mecânicos ou de hardware, geralmente envolve a análise dos resultados de simulações, avaliações de segurança ou métodos estabelecidos, como Quality Function Deployment (QFD), Failure Mode and Effects Analysis (FMEA).

Assim como no caso dos stakeholders, os documentos podem ser identificados de forma pragmática e sistemática.

Ao *identificar* documentos de *forma pragmática*, os engenheiros de requisitos usam seu conhecimento e experiência atuais do contexto (p. ex., domínio) para nomear documentos e tipos de documentos relevantes.

Na *identificação sistemática de documentos*, o Engenheiro de Requisitos pode:

- Pesquisar representantes de categorias típicas de documentos
(por exemplo: quais padrões técnicos se aplicam a uma máquina do tempo?)
- Pesquisar referências em documentos previamente identificados para outros documentos relevantes
(por exemplo: A ISO26262 cita outras normas que podem se aplicar à nossa máquina do tempo?)
- Solicitar documentação relevante aos stakeholders previamente identificados
(p. ex., pergunte ao Dr. Emmet Brown se há alguma documentação sobre o Compensador de Fluxo)
- Pesquise a documentação sobre sistemas previamente identificados como relevantes (consulte 2.4)
(p. ex., existe um manual do usuário para a TARDIS ou para a máquina do tempo DeLorean?)

Durante a *identificação sistemática de documentos*, o Engenheiro de Requisitos define as atividades de elicitacão com foco na identificação de documentos. Dois tipos diferentes de objetivos de elicitacão devem ser considerados:

- *Focado em informações*: Encontrar documentos para determinadas informações necessárias
- *Focado em documentos*: Encontrar documentos de determinados tipos de documentos, considerados relevantes para o desenvolvimento

ID	RS_EA_06
Objetivo da elicitação	Encontre pelo menos um documento sobre a física da viagem no tempo.
Qualidade do resultado	Nome do documento, autor, data de publicação, local para encontrá-lo
Fontes de requisitos	Biblioteca, Internet
Técnica de elicitação	Leitura baseada em perspectiva

Figura 13: Exemplo de uma atividade de elicitação com foco em informações

ID	RS_EA_07
Objetivo da elicitação	Descubra quais documentos legais afetam o tempo de viagem
Qualidade do resultado	Nome do documento, autoridade de publicação, data de publicação, região em que se aplica, onde encontrá-lo
Fontes de requisitos	Dr. Who, consultor jurídico/advogado da empresa em desenvolvimento
Técnica de elicitação	Entrevista

Figura 14: Exemplo de uma atividade de elicitação com foco em documentos

O Engenheiro de Requisitos precisa decidir quais dos documentos coletados têm valor potencial como fontes de requisitos. Assim, os documentos precisam ser digitalizados e avaliados quanto à sua utilidade potencial.

Dependendo do contexto, diferentes critérios podem ser relevantes. Além do conteúdo, outros aspectos podem influenciar o valor (e, portanto, a prioridade) de um documento como fonte de requisitos:

- **Disponibilidade:** Os documentos podem ser confidenciais e requerem uma certa autorização de segurança para serem acessados.
- **Tamanho ou, melhor ainda, proporção estimada de tamanho/conteúdo:** Se um projeto tiver um prazo apertado, documentos pequenos contendo informações importantes podem ser mais valiosos do que documentos longos com muitos detalhes irrelevantes.
- **Idade:** quanto mais antigo for um documento, maior será a probabilidade de seu conteúdo estar desatualizado.
- **Relevância:** O documento de requisitos do sistema predecessor pode ser mais relevante do que o documento de requisitos de um sistema de interface.

2.3.2 Esquema de documentação para documentos

Todas as informações coletadas nos documentos devem ser suficientemente documentadas. Essa documentação deve incluir, no mínimo:

- Título
- Local onde ele é mantido (p. ex., nome da pasta física, link para o documento digital)
- Versão do documento
- Breve descrição (que tipo de informação o documento pode fornecer)
- Relevância

Dependendo do contexto, informações adicionais podem ser relevantes. São exemplos:

- Pessoa responsável pelo documento (proprietário do documento)
- Pessoa que adicionou o documento à lista (relevante se mais de uma pessoa estiver atualizando a lista)
- Data em que o documento foi adicionado à lista
- Data em que o documento foi revisado pela última vez (para requisitos ou se uma nova versão está disponível)
- Tamanho do documento

Os documentos sempre têm certos relacionamentos com os stakeholders, que também devem ser registrados. Portanto, pode fazer sentido vincular a documentação dos documentos à documentação dos stakeholders.

A seguir, exemplos de relacionamentos entre documentos e stakeholders:

- Stakeholders que mencionaram a relevância do documento
- Autor, organização emissora
- Organizações que utilizam o documento em seus processos
- Organizações envolvidas na verificação da adesão ao documento

O Engenheiro de Requisitos deve manter atualizadas as informações sobre os documentos. Isso inclui reconsiderar se outros documentos se tornaram relevantes ou se os documentos identificados anteriormente perderam a relevância. Deve-se dar atenção especial a alterações, atualizações e numeração de versões.

2.4 Identificar, classificar e gerenciar sistemas

No contexto do sistema a ser desenvolvido, podem existir outros sistemas que representem fontes de requisitos. Assim que um sistema é identificado como uma possível fonte de requisitos, ele se torna parte do contexto do sistema (consulte [IREB2020]).

Na Seção 2.4.1, discutimos a identificação e a seleção de tais sistemas e, na Seção 2.4.2, abordamos a documentação dos sistemas como fontes de requisitos.

2.4.1 Identificação e seleção de sistemas como fontes de requisitos

Os requisitos podem ser derivados de muitos tipos diferentes de sistemas. Dependendo do caráter do projeto de desenvolvimento, outros sistemas podem ter uma importância alta, média ou baixa como fontes de requisitos. Para sistemas com muitas interfaces, é provável que haja muitos sistemas relevantes como fontes de requisitos. Se um novo sistema for desenvolvido para substituir um ou mais sistemas existentes, esses sistemas terão grande importância como fontes de requisitos.

Mesmo para novos sistemas, os sistemas existentes de outros campos de aplicação podem ser relevantes (consulte também a Seção 3.2.2 analogy technique).

Os tipos de sistemas que podem ser relevantes como fontes de requisitos são (não exaustivos):

- **Sistemas de interface, incluindo sistemas legados**
Para que o novo sistema possa interagir com seus sistemas de interface, é essencial conhecer esses sistemas e, em particular, suas especificações/requisitos de interface.
- **Sistemas que compartilham uma plataforma/ambiente/ecossistema**
Se o sistema a ser desenvolvido tiver que ser integrado a uma plataforma, ambiente ou ecossistema existente, haverá requisitos (principalmente restrições) resultantes dessa plataforma, ambiente ou ecossistema, por exemplo: um conceito de usabilidade ou requisitos técnicos.
- **Sistemas da concorrência**
Os concorrentes são motores constantes da inovação. Portanto, é muito importante conhecer os produtos da concorrência, tanto os produtos OTF (off the shelf) quanto aqueles usados ou que estão sendo desenvolvidos dentro da empresa, por exemplo, por outros departamentos ou escritórios. No caso do último, devemos estar atentos tanto às possíveis sinergias quanto às soluções que podem se tornar concorrentes internos.
- **Sistemas com dados, funcionalidade ou interfaces de usuário semelhantes**
Quase nenhum sistema é único. De fato, talvez não seja desejável desenvolver um sistema exclusivo.
Os sistemas que são semelhantes ao seu sistema em desenvolvimento, seja porque processam dados semelhantes (p. ex., máquina de venda automática vs. caixa registradora vs. loja on-line), têm funcionalidade semelhante (p. ex., máquina de venda automática de passagens de trem vs. máquina de venda automática de bebidas vs. caixa eletrônico) ou compartilham uma interface de usuário semelhante (p. ex., controle remoto vs. calculadora) podem ser fontes valiosas de requisitos.

- **Sistema(s) predecessor(es) a ser(em) substituído(s)**
Se um sistema predecessor tiver que ser substituído, o sistema antigo geralmente é uma das fontes de requisitos mais importantes. No entanto, é importante não se deixar influenciar demais pelo sistema antigo e estar aberto a melhorias e emendas.
- **Sistemas futuros (em construção)**
É importante estar atento aos sistemas futuros no contexto do sistema a ser desenvolvido. Pode haver sistemas já em desenvolvimento. Esses sistemas futuros podem mudar o ambiente do sistema a ser desenvolvido e, portanto, seus requisitos.

Assim como ocorre com os stakeholders e os documentos, os sistemas podem ser identificados de forma pragmática e sistemática.

Ao *identificar* sistemas de *forma pragmática*, os engenheiros de requisitos usam seu conhecimento e experiência atuais do projeto e seu contexto (p. ex., domínio) para nomear sistemas e tipos de sistemas relevantes.

Na *identificação sistemática*, o Engenheiro de Requisitos pode:

- Usar a documentação do contexto do sistema
- Pergunte às stakeholders previamente identificadas sobre os sistemas relevantes (p. ex., quais sistemas são planejados ou estão sendo desenvolvidos que podem ser relevantes para o sistema a ser desenvolvido?)
- Pesquisar documentos previamente identificados para obter informações sobre sistemas relevantes
(p. ex., pesquisar a documentação da arquitetura de sistemas de um sistema de interface para obter informações sobre suas interfaces com outros sistemas)
- Usar técnicas de geração de ideias para identificar sistemas potencialmente análogos
(p. ex., brainstorming)
- Realizar pesquisas de mercado para identificar sistemas concorrentes
(ou seja, que outros sistemas estão disponíveis ou sendo desenvolvidos para o mesmo propósito?)
- Considerar os sistemas legados
(p. ex., quais sistemas legados tinham uma finalidade semelhante?)

Durante a *identificação sistemática do sistema*, o Engenheiro de Requisitos define as atividades de elicitação voltadas para a identificação dos sistemas. Dois tipos diferentes de objetivos de elicitação devem ser considerados:

- *Focado em informações*: Encontrar sistemas que contenham determinadas informações necessárias
- *Focado no sistema*: Encontrar sistemas de determinados tipos que sejam considerados relevantes para o projeto de desenvolvimento

Figura 15 e Figura 16 dão exemplos de atividades de elicitação focadas em informações e focadas no sistema.

ID	RS_EA_08
Objetivo da elicitação	Encontre pelo menos cinco máquinas de viagem no tempo concorrentes.
Qualidade do resultado	Nome do sistema, criador, local onde (ou quando) encontrá-lo. O ideal é que as máquinas do tempo estejam funcionando, mas uma máquina do tempo que não esteja funcionando também seria aceitável.
Fontes de requisitos	Biblioteca, Internet, conferência de viagem no tempo
Técnica de elicitação	Leitura baseada em perspectiva

Figura 15: Exemplo de uma atividade de elicitação com foco no sistema

ID	RS_EA_09
Objetivo da elicitação	Encontre pelo menos cinco sistemas que ofereçam espaço para dois adultos.
Qualidade do resultado	Nome do sistema, criador, local onde pode ser encontrado
Fontes de requisitos	Todos os desenvolvedores
Técnica de elicitação	Oficina

Figura 16: Exemplo de uma atividade de elicitação com foco em informações

2.4.2 Esquema de documentação para sistemas

Todas as informações coletadas sobre os sistemas devem ser suficientemente documentadas. Isso deve incluir pelo menos o sistema de cada um:

- Nome
- Tipo (p. ex., sistema concorrente, sistema predecessor, sistema de interface, ...)
- Dados, funcionalidade, processos, grupos de usuários (breve descrição)
- Versão
- Relevância

Dependendo do contexto, informações adicionais podem ser relevantes. São exemplos:

- Pessoa responsável pelo sistema (system owner)
- Pessoa que adicionou o sistema à lista (relevante se mais de uma pessoa estiver atualizando a lista)
- Data em que o sistema foi adicionado à lista

- Data em que o sistema foi revisado pela última vez (para requisitos ou se uma nova versão está disponível)
- Número de usuários

Deve-se dar atenção especial aos sistemas com interface direta. Eles podem ser categorizados como:

- Fontes de dados: fornecimento de dados
- Sumidouros de dados: usando dados
- Sistemas de suporte, como um sistema operacional (SO) ou um sistema de gerenciamento de banco de dados (DBMS)

Os sistemas sempre têm certos relacionamentos com os stakeholders, que também devem ser registrados. Portanto, pode fazer sentido vincular a documentação dos sistemas à documentação dos stakeholders.

A seguir, exemplos de relacionamentos entre sistemas e stakeholders:

- Stakeholders/Organizações que usam o sistema de forma direta ou indireta em seus processos
- Stakeholders/Organizações que operam o sistema
- Stakeholders/Organizações que projetam, desenvolvem ou comercializam o sistema
- Stakeholders/Organizações que mantêm o sistema, oferecem suporte ou treinamento
- Organizações que observam o sistema (p. ex., governos, ONGs)
- stakeholders, mencionando a relevância do sistema
- System Owner
- Organizações envolvidas na verificação da aderência do sistema à aplicação de leis e padrões

O Engenheiro de Requisitos precisa manter as informações sobre os sistemas atualizadas. Isso inclui reconsiderar se outros sistemas se tornaram relevantes ou se os sistemas identificados anteriormente podem ter perdido a relevância. Fique de olho nas alterações, atualizações e numeração de versões.

As informações sobre os sistemas geralmente estão presentes em documentos. Esses documentos devem ser gerenciados separadamente como fontes de requisitos (consulte 2.3). A relação entre o documento e o sistema também deve ser documentada.

3 Elicitação

Uma infinidade de técnicas foi desenvolvida para a elicitação de requisitos. Para o syllabus Requirements Elicitation e, portanto, para este handbook, selecionamos algumas dessas técnicas e as apresentamos aqui de forma estruturada.

Como adicionamos técnicas ao módulo de Elicitação de Requisitos, ao mesmo tempo adaptamos a categorização usada no Nível Básico. A categorização torna mais compreensível a apresentação de nossas 19 técnicas de elicitação e ferramentas de raciocínio. Essa diferenciação é, obviamente, artificial: na prática, não há uma separação clara entre as técnicas. Entretanto, para fins de apresentação e ensino, a diferenciação é importante para fornecer uma estrutura e descrever o foco principal de cada técnica.

Todas as técnicas de elicitação descritas neste capítulo são estruturadas da seguinte forma:

- **O que é isso?**
descreve resumidamente os principais fatores da técnica de elicitação.
- **Função dos participantes (se aplicável)**
descreve quais funções estão envolvidas na aplicação dessa técnica.
- **Preparação**
descreve quais ações devem ser tomadas para preparar a aplicação da técnica de elicitação.
- **Aplicação**
descreve como a técnica de elicitação é aplicada e o que deve ser levado em consideração durante sua aplicação.
- **Processamento de resultados**
descreve quais ações são necessárias para processar os resultados (veja a observação abaixo).
- **Produtos de trabalho típicos**
cite os produtos de trabalho normalmente usados ou produzidos durante a aplicação da técnica de elicitação (veja a observação abaixo).
- **Oportunidades**
descreve as possíveis vantagens da técnica de elicitação.
- **Desafios**
descreve as possíveis armadilhas ou desvantagens da técnica de elicitação.
- **Variantes (se aplicável)**
descreve as variantes da técnica de elicitação e fornece uma breve descrição.

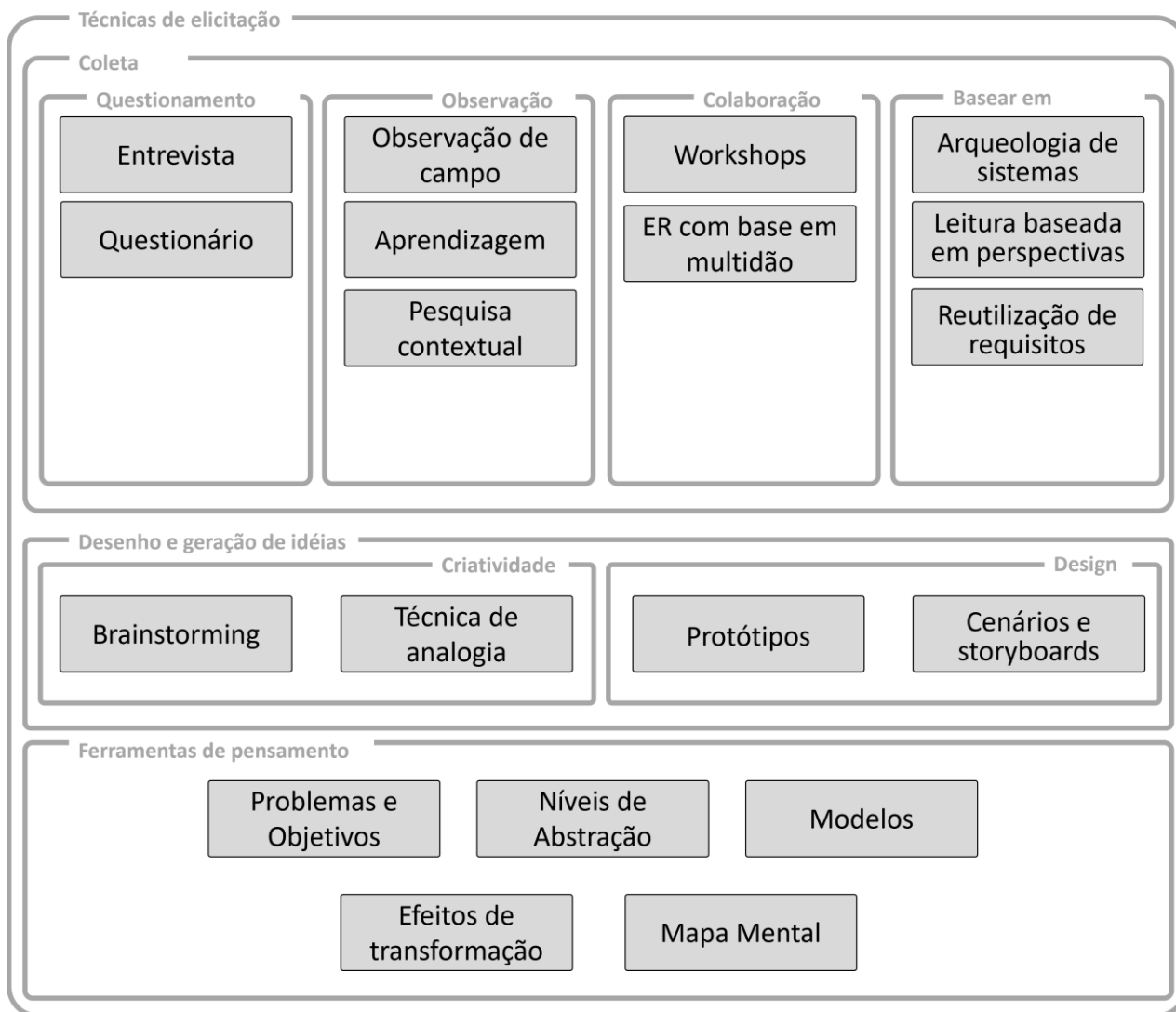


Figura 17: Visão geral das técnicas de elicitação

Observação:

Os resultados obtidos com a execução de uma técnica de elicitação são informações brutas que precisam ser processadas e formuladas em requisitos. Esse último faz parte da documentação de requisitos e, portanto, não é abordado neste Handbook.

Após cada atividade de elicitação, a documentação deve ser atualizada com as informações recebidas, por exemplo, documentação de requisitos, documentação de contexto do sistema, documentação de recursos de requisitos. Também deve ser decidido se são necessárias outras atividades de elicitação.

Seção 3.4 fornece características típicas de identificação de técnicas de elicitação. Elas podem ser usadas para descrever novas técnicas e fornecer diretrizes gerais sobre quais características são úteis em uma determinada situação de projeto.

3.1 Técnicas de coleta

As técnicas de coleta são técnicas estabelecidas para a elicitación de requisitos. Eles ajudam a obter satisfação e insatisfação. Usamos esse termo para agrupar técnicas de questionamento (entrevista e questionário), técnicas de observação (observação de campo, aprendizado, investigação contextual), técnicas de colaboração (workshop de requisitos e Engenharia de Requisitos baseada em multidões) e técnicas baseadas em artefatos (leitura baseada em perspectiva, arqueologia de sistemas e reutilização de requisitos), consulte Figura 17: Visão geral das técnicas de elicitación.

3.1.1 Técnicas de questionamento

As técnicas de questionamento têm como objetivo fazer perguntas apropriadas aos participantes para receber respostas das quais os requisitos podem ser derivados.

Perguntas abertas e fechadas

Para todas as técnicas de perguntas abordadas nesta Seção, é importante saber a distinção entre perguntas abertas e fechadas.

Para perguntas fechadas, as respostas são dadas ou definidas, seja pela própria pergunta ou por referência explícita às respostas disponíveis (p. ex., How old are you? Você gosta de peixe?). As perguntas fechadas geram dados quantitativos, ou seja, dados que podem ser processados estatisticamente sem adaptações adicionais.

As perguntas abertas, por outro lado, permitem uma resposta livre, a fim de consultar o conhecimento narrativo ou argumentativo e, portanto, levam a dados qualitativos (p. ex., Qual é o seu prato favorito? Como os aviões voam?). Os dados qualitativos só podem ser avaliados por meio de análise cognitiva.

No contexto da elicitación de requisitos, são necessários dados predominantemente qualitativos. Portanto, são necessárias principalmente perguntas abertas. No entanto, também há situações que exigem dados quantitativos (p. ex., avaliar se a declaração de um usuário representativo é confirmada por outros usuários). Nesse caso, devem ser usadas perguntas fechadas.

Ao usar técnicas de questionamento em uma atividade de elicitación, a definição da qualidade do resultado deve incluir se os dados qualitativos ou quantitativos devem ser elicitados.

A seguir, serão descritas as duas técnicas de questionamento mais relevantes: entrevista e questionário.

3.1.1.1 Entrevista

Devido à sua flexibilidade, as entrevistas são provavelmente uma das técnicas de elicitación usadas com mais frequência. Eles não exigem uma grande configuração ou ferramentas específicas e podem ser usados para obter requisitos de alto nível, bem como requisitos muito específicos.

Normalmente, os requisitos elicitados em uma entrevista são fatores esperados, já que o entrevistado expressa informações conscientes. No entanto, ao usar as técnicas corretas de questionamento e observar as reações do usuário, também é possível identificar quem está insatisfeito ou quem está encantado.

Embora uma entrevista não seja muito complicada e a maioria das pessoas tenha uma boa compreensão do que se trata, ela exige objetivos claros e uma preparação detalhada para aproveitar bem o tempo da entrevista e obter resultados valiosos e sustentáveis.

O que é uma entrevista?

Em uma entrevista, o Engenheiro de Requisitos faz perguntas a uma ou mais stakeholders a fim de obter novos requisitos ou refinar os existentes (objetivo de obtenção). Os stakeholders respondem a essas perguntas e o Engenheiro de Requisitos (ou um assistente) registra as informações que recebe. Uma entrevista difere de uma simples conversa devido, principalmente, a três fatores: o papel dos participantes, a preparação e o processamento dos resultados.

Papel dos participantes

Em uma entrevista, três funções são obrigatórias: *entrevistador*, *entrevistado* e *anotador*.

O Engenheiro de Requisitos assume o papel de *entrevistador*. É sua obrigação se preparar (veja Preparação) e conduzir (veja Seção Processamento de **resultados**) a entrevista e para processar as informações coletadas (consulte a Seção Processamento de **resultados**). Um entrevistador deve ser: *conhecedor*, *estruturador*, *claro*, *gentil*, *sensível*, *aberto*, *direcionador*, *crítico*, *recordador* e *intérprete* (com base em [Kvale2008], explicado em detalhes na Seção Aplicação).

A stakeholder é o *entrevistado*. Ele responde às perguntas feitas pelo entrevistador, expressando assim seus requisitos.

É função do *anotador* registrar todas as informações relevantes fornecidas pelo entrevistado. Ele precisa saber de antemão qual é o objetivo da entrevista e deve conhecer e entender o roteiro da entrevista. Somente se o anotador conhecer suficientemente o contexto da entrevista e o tópico (inclusive a terminologia), ele poderá acompanhar a entrevista e decidir quais informações anotar e quais não.

O entrevistador também pode assumir a função de anotador. No entanto, se possível, isso deve ser evitado, pois exige que o entrevistador alterne entre dois modos cognitivos diferentes, exigindo energia adicional e geralmente afetando negativamente o desempenho em ambas as funções. A função de anotador pode ser substituída por um dispositivo de gravação de som ou vídeo. As vantagens e desvantagens são discutidas na Seção Preparação.

Preparação

Defina o(s) objetivo(s) da elicitación e a qualidade do resultado exigido como parte da definição da atividade de elicitación (consulte 1.3.1).

Selecione o(s) interessado(s) adequado(s) para a entrevista, que você espera que seja(m) capaz(es) de responder às suas perguntas para atingir o(s) objetivo(s) da elicitación e, ao mesmo tempo, alcançar a qualidade de resultado exigida. Você nunca saberá com antecedência se selecionou os stakeholders certos. No entanto, com um bom gerenciamento dos stakeholders e uma documentação dos stakeholders devidamente pesquisada, você aumentará significativamente a chance de fazer uma boa escolha.

Prepare um guia de entrevista. Há muitas maneiras diferentes de fazer isso: algumas pessoas preferem uma lista de tópicos, mostrando a ordem e a hierarquia pretendidas das perguntas; outras preferem um guia no estilo de mapa mental, em que as perguntas são ordenadas no sentido horário em torno do objetivo da entrevista. A última forma pode facilitar o acompanhamento da trilha de pensamento do entrevistado quando ele estiver pulando de um tópico para outro enquanto responde à pergunta originalmente feita.

O entrevistador desenvolve uma estrutura geral para a entrevista, refletida no guia de entrevista. Depois de decidir quais perguntas fazer e quais perguntas têm prioridade, deve-se pensar em como fazer as apresentações, o objetivo da entrevista, qual deve ser a pergunta inicial, a ordem das perguntas e como a entrevista deve terminar. O guia não precisa conter o texto exato das perguntas a serem feitas (pelo menos em uma entrevista qualitativa). No entanto, para serem *claros* em seus questionamentos, os entrevistadores devem pensar em como formularão suas perguntas.

Outra questão organizacional é a de tempo e lugar. O entrevistador deve definir um horário adequado que se encaixe na agenda do entrevistado. É aconselhável adicionar mais 15 minutos ao tempo planejado para a entrevista para permitir o tempo de aquecimento e como um amortecedor geral. A entrevista em si não deve exceder 60 minutos. Para a maioria dos entrevistadores e entrevistados, 20 a 40 minutos é um bom período de tempo.

A entrevista deve ser realizada em uma sala separada, permitindo a concentração e a confidencialidade. O ideal é que o local seja próximo e conveniente para o entrevistado.

O entrevistador deve decidir antecipadamente como a entrevista deve ser gravada. Há três opções: o próprio entrevistador faz anotações, um anotador faz anotações ou uma gravação de áudio ou vídeo.

Essas são as vantagens e desvantagens das três opções:

	O entrevistador faz anotações	O anotador faz anotações	Gravação de áudio/vídeo
Vantagens	<p>Não é necessária nenhuma pessoa ou dispositivo adicional.</p> <p>Enquanto fazem anotações, o entrevistador e o entrevistado têm tempo para pensar.</p>	<p>O entrevistador pode se concentrar totalmente em fazer perguntas e ouvir as respostas/observar o entrevistado.</p> <p>O entrevistador e o tomador de notas têm duas perspectivas sobre o mesmo evento (possibilidade de descobrir vieses cognitivos).</p>	<p>O entrevistador pode se concentrar totalmente em fazer perguntas e ouvir as respostas/observar o entrevistado.</p> <p>Todas as informações estão disponíveis sem filtros após a entrevista.</p>
Desvantagens	<p>Algumas informações podem ser perdidas ou distorcidas.</p> <p>O entrevistador precisa alternar entre dois modos cognitivos (requer energia).</p> <p>Durante a tomada de notas, podem surgir longas pausas incômodas</p> <p>As anotações geralmente são muito curtas e podem não ser mais compreendidas após a entrevista.</p>	<p>Algumas informações podem ser perdidas ou distorcidas.</p> <p>Se o anotador não estiver totalmente envolvido no tópico da entrevista, ele poderá interpretar mal as informações e anotá-las de forma errada ou incompleta.</p>	<p>A filtragem de informações começa somente após a entrevista; a entrevista inteira precisa ser ouvida novamente.</p> <p>Restrições legais podem ser aplicadas, dependendo da legislação nacional ou dos regulamentos da empresa.</p> <p>O entrevistado pode proibir a gravação.</p>

Hint 3.1.1:

Muitas vezes, as opções 1 e 3 são combinadas, ou seja, a entrevista é gravada e o entrevistador ainda faz algumas anotações (mas pode se concentrar em fazer as perguntas e ouvir as respostas).

Se a gravação da entrevista for planejada, é aconselhável obter a aprovação do entrevistado e de sua organização com antecedência.

Aplicação

Durante a entrevista, o entrevistador conduz o entrevistado fazendo perguntas. O entrevistador coloca em prática as seguintes qualificações [Kvale2008]:

- *Conhecedor*: tem amplo conhecimento sobre o tópico; sabe quais questões são importantes para se defender
- *Estruturação*: prepara e segue uma estrutura para a entrevista. Comunica a estrutura ao entrevistado durante a entrevista
- *Clara*: usa linguagem simples e faz perguntas claras, simples, fáceis e curtas
- *Gentil*: deixa as pessoas terminarem; dá a elas tempo para pensar; tolera pausas
- *Sensível*: ouve atentamente o que é dito e como é dito; é empático ao lidar com o entrevistado
- *Aberto*: responde ao que é importante para o entrevistado; está aberto a novos aspectos introduzidos pelo entrevistado e os acompanha
- *Direção*: sabe o que quer descobrir. O entrevistador controla o curso da entrevista e não tem receio de interromper digressões.
- *Crítico*: está preparado para contestar o que é dito, por exemplo, lidando com inconsistências nas respostas dos entrevistados
- *Lembrança*: consegue se lembrar de declarações anteriores e relaciona o que é dito com o que foi dito anteriormente na entrevista
- *Interpretação*: esclarece e amplia os significados das declarações do entrevistado; fornece interpretações do que foi dito, que podem ser confirmadas ou negadas pelo entrevistado

O(s) interessado(s) (entrevistado) responde(m) a essas perguntas e o Engenheiro de Requisitos (entrevistador) ouve atentamente, verificando vários aspectos, por exemplo:

- Se o stakeholder entendeu a pergunta e está fornecendo as informações desejadas
- Se o Engenheiro de Requisitos entende o que o interessado está dizendo
- Se a pergunta foi totalmente respondida
- Se o stakeholder está enviando informações não verbais relevantes
- Se o anotador está anotando as informações necessárias

O anotador ouve o entrevistador e o entrevistado, filtra as informações relevantes da conversa e as registra. Ele observa atentamente o entrevistador para captar sinais não verbais ou verbais sobre o que deve ser anotado.

[Port2013] e [BaCC2015] fornecem mais informações sobre a arte de entrevistar, com foco em entrevistas com usuários.

Após a entrevista, o *entrevistador* e/ou *anotador* prepara as anotações da entrevista e as envia ao(s) entrevistado(s) para revisão. Isso serve a dois propósitos: Em primeiro lugar, você se certifica de que entendeu corretamente todas as informações da entrevista e não se esqueceu de nenhum aspecto importante. Em segundo lugar, você demonstra apreço pelo tempo e pela contribuição do entrevistado.

Hint 3.1.2:

Defina uma data para solicitar o feedback do entrevistado.

Também é aconselhável enviar um lembrete breve e amigável alguns dias antes do término do prazo.

Processamento de resultados

Após as entrevistas, os dados coletados (brutos) precisam ser analisados e agregados em informações úteis. Podem ser requisitos, necessidades, metas, problemas, grupos de usuários, cenários, processos, produtos de trabalho, etc.

Um diagrama de afinidade [BaCC2015] pode ajudar no processamento dos dados coletados. Veja o exemplo a seguir, que ilustra os dois principais aspectos de um diagrama de afinidade:

1. Extraia insights dos dados coletados e escreva cada insight em um cartão (verde).
2. Agrupamento: Agrupe seus insights/cartões em grupos e rotule cada grupo (cartões amarelos).



Figura 18: Diagrama de afinidade

Artefatos típicos

Durante a preparação, a execução e o pós-processamento de uma entrevista, geralmente são criados ou atualizados os seguintes produtos de trabalho:

- Guia de entrevista
- Anotações e/ou gravações de áudio/vídeo feitas durante a entrevista
- Anotações finais da entrevista (a serem enviadas a todos os participantes).

Oportunidades

A grande vantagem de uma entrevista é o feedback direto – não apenas verbalmente, mas também não verbalmente. Ao observar atentamente o entrevistado, você pode obter muito mais informações do que apenas o que ele está dizendo. Você também pode reagir imediatamente a qualquer informação que receber.

Além disso, uma entrevista pode ser realizada em um prazo bastante curto. Embora nunca deva ser despreparado, um entrevistador experiente pode precisar apenas de cerca de uma hora de antecedência para preparar suficientemente a entrevista.

Desafios

A realização e, principalmente, a avaliação de entrevistas consomem muito tempo. Se o Engenheiro de Requisitos não tiver algumas das qualificações mencionadas, a entrevista pode facilmente sair do controle e não produzir os resultados pretendidos. Fazer as perguntas erradas, ou as perguntas certas da maneira errada, pode reduzir drasticamente a utilidade da entrevista.

Variáveis

Há diversas variantes de entrevistas diferentes, por exemplo:

- **Entrevista aberta/qualitativa:**

A entrevista não segue uma estrutura rígida. O entrevistador pode se desviar de seus tópicos ou perguntas inicialmente preparadas durante a entrevista. Se várias entrevistas forem realizadas sobre o mesmo tópico, cada entrevista terá seu curso individual.

Essa variante é bastante comum na elicitación de requisitos.

- **Entrevista parcialmente padronizada:**

Algumas perguntas são padronizadas e precisam ser feitas exatamente da mesma forma em todas as entrevistas. O restante da entrevista é uma entrevista aberta.

A parte padronizada é usada para coletar dados quantitativos.

- **Entrevista totalmente padronizada/quantitativa:**

A ordem das perguntas, seu texto exato e suas possíveis respostas são fornecidos. Seu objetivo é fornecer resultados comparáveis. Eles podem ser processados diretamente com métodos estatísticos.

Na Engenharia de Requisitos, essa variante não é muito comum. Ele é usado principalmente em pesquisas de mercado.

- **Entrevista em grupo:**

Vários entrevistados são questionados por um entrevistador em uma única sessão de entrevista.

Essa variante tem a vantagem de que as diferenças de opinião entre os entrevistados podem ser tratadas imediatamente. No entanto, esse tipo de entrevista não é aconselhável se houver diferenças pessoais entre os entrevistados ou se um deles ocupar um cargo mais alto que o outro. É provável que ambos os fatores influenciem negativamente o resultado da entrevista.

3.1.1.2 Questionário

O que é um questionário?

Com um questionário, várias pessoas são solicitadas a responder, por escrito, o mesmo conjunto de perguntas, que são apresentadas de forma estruturada. Há dois tipos principais de questionários: *quantitativo* e *qualitativo*.

Os *questionários quantitativos* são usados para confirmar hipóteses ou requisitos previamente obtidos. Eles podem ser avaliados rapidamente e fornecem informações estatísticas.

Os questionários quantitativos usam perguntas fechadas (consulte 0).

Os *questionários qualitativos* são adequados para a obtenção de novos requisitos. Eles tendem a fornecer resultados complexos e, portanto, geralmente consomem muito mais tempo para serem preparados e avaliados.

Os questionários qualitativos usam perguntas abertas (consulte 3.1.1).

Os questionários quantitativos também podem incluir algumas perguntas abertas e os questionários qualitativos podem incluir algumas perguntas fechadas.

Papel dos participantes

Em uma pesquisa por questionários⁵, são necessárias duas funções: o autor do questionário e os respondentes.

O Engenheiro de Requisitos assume a função de *redator do questionário*. É sua obrigação elaborar e distribuir o questionário e avaliar os resultados.

Os *respondentes*, que são os stakeholders, recebem o questionário e são solicitados a respondê-lo em uma data específica.

Dependendo de como o questionário é emitido, outras funções podem estar envolvidas, como um processador de dados para inserir dados de questionários impressos em um sistema de processamento de dados ou para processá-los manualmente.

Preparação

Como uma pesquisa por questionários não pode ser corrigida depois que os questionários são distribuídos, você deve investir tempo e pensamentos suficientes na fase de preparação.

Em primeiro lugar, você deve esclarecer se há alguma restrição legal ou específica da empresa aplicável ao seu projeto com relação à forma como a pesquisa é realizada. Em algumas empresas, as pesquisas precisam ser aprovadas por um conselho específico.

⁵ O termo "questionário" descreve a ferramenta que é usada em uma pesquisa por meio de questionários.

Defina o(s) objetivo(s) da elicitação e a qualidade do resultado exigido como parte da definição da atividade de elicitação (consulte 1.3.1).

Selecione os stakeholders adequadas para a pesquisa. Quantos participantes são necessários para obter um resultado valioso ou representativo? Um questionário quantitativo precisa fornecer resultados representativos. No caso de um questionário qualitativo, os resultados representativos podem não ser o foco, mas sim os valiosos, que fornecem novos requisitos.

Ao selecionar os participantes, você deve consultar a lista de stakeholders. Se os participantes não estiverem na lista ou se faltarem informações de contato, atualize a lista de stakeholders.

Defina o tamanho máximo do questionário: quanto mais longo for o questionário, menor será o número de respostas. Quantas perguntas são aceitáveis para seus participantes?

Selecione a forma de apresentação. Você pode usar uma ferramenta de pesquisa ou precisa usar papel?

Formule as perguntas com relação ao objetivo da elicitação e às stakeholders alvo. Use perguntas abertas para obter novos aspectos e requisitos; use perguntas fechadas para verificar os requisitos que você já obteve ou para confirmar ou rejeitar uma hipótese com base nos requisitos obtidos até o momento.

Se estiver usando perguntas fechadas, decida sobre um tipo adequado de escala. Certifique-se também de que não esteja usando muitos tipos diferentes de escalas em seu questionário. Verifique, para todas as perguntas fechadas, se é necessário fornecer uma opção "não aplicável". Se essa opção estiver ausente – especialmente quando as perguntas forem obrigatórias – isso pode falsificar o resultado, pois o participante é forçado a responder à pergunta mesmo que não saiba respondê-la.

Defina a ordem das perguntas e certifique-se de que essa ordem seja lógica. Tome muito cuidado para que as respostas a algumas perguntas não excluam outras perguntas (p. ex., "Você bebe vinho?") e "Você prefere vinho tinto ou vinho branco?" Se o participante responder à primeira pergunta com "não", a última pergunta não fará sentido).

Planeje o cronograma da sua pesquisa. Quando vocês enviarão o questionário? Quanto tempo você deve dar para responder? Quando você deve enviar um lembrete amigável? Quanto tempo você precisará para a avaliação dos resultados?

Teste seu questionário antes de distribuí-lo. Faça com que ele seja revisado por outras pessoas e que seja respondido por alguns candidatos do teste para identificar questões pouco claras. Assim que o questionário for distribuído, você não poderá mais corrigir as falhas. Além disso, faça uma avaliação experimental dos dados fornecidos pelos questionários de teste devolvidos. Muitas vezes, você só descobre, ao fazer a avaliação, que deixou passar um aspecto importante ou que as perguntas não fornecem os dados exatos que você queria sondar. Além disso, o fato de ter feito a avaliação uma vez ajuda você a fazê-la novamente de forma eficiente com os dados do questionário real.

Aplicação

Distribua o questionário para os participantes. Ao enviar o questionário por e-mail, leve em consideração se os destinatários devem ser divulgados ou não. Normalmente, são usados destinatários não divulgados.

Em seu anúncio com a distribuição, comunique o contexto do questionário: Quem é você e por que está enviando um questionário? Qual é o objetivo da pesquisa? Quanto tempo levará para responder ao questionário? O que acontecerá com os resultados?

Os participantes serão informados sobre os resultados? Se sim, até quando? Para aumentar o número de possíveis respostas, pode ser uma boa ideia anunciar um incentivo.

Processamento de resultados

No caso de questionários qualitativos, o processamento dos resultados geralmente leva algum tempo. Quanto mais respostas você receber, mais tempo será necessário para a avaliação. Nesse caso, você geralmente terá que lidar com declarações conflitantes, ou potencialmente conflitantes, e diferentes níveis de requisitos.

Hint 3.1.3:

Um diagrama de afinidade [BaCC2015] pode ajudar no processamento dos dados coletados (consulte o exemplo de um diagrama de afinidade no capítulo 3.1.1.1 sobre o processamento de dados coletados em entrevistas).

Para questionários quantitativos, o processamento de resultados pode ser feito automaticamente (se a pesquisa for realizada eletronicamente). Usando uma ferramenta de pesquisa baseada na Web, o número de respostas não influencia a quantidade de tempo necessária para agregar os dados. Entretanto, a interpretação intelectual das estatísticas ainda pode levar algum tempo.

Produtos de trabalho típicos

Durante a preparação, a execução e o processamento de resultados de uma pesquisa por questionários, os seguintes produtos de trabalho geralmente são criados ou atualizados:

- Questionário
- Resultados processados (p. ex., estatísticas, diagramas, relatórios)

Oportunidades

Uma pesquisa por questionário é uma técnica de elicitación assíncrona, o que significa que o Engenheiro de Requisitos e o interessado não precisam estar no mesmo local e ao mesmo tempo para usar essa técnica.

Com um questionário, é possível envolver muitos stakeholders de uma só vez, seja para coletar novos requisitos de muitos stakeholders ao mesmo tempo ou para avaliar os requisitos existentes com um grande conjunto de stakeholders e, dessa forma, confirmar ou refutar uma hipótese (p. ex., "Os usuários estão satisfeitos com o sistema como está").

Desafios

O uso de uma pesquisa por questionário pode consumir muito tempo: "Um dos maiores equívocos sobre uma pesquisa é a velocidade com que você pode preparar, coletar e analisar os resultados. Uma pesquisa pode ser um método extremamente valioso, mas é preciso tempo para fazê-la corretamente." [BaCC2015]

Especialmente no caso de questionários qualitativos, a avaliação pode se tornar extremamente demorada.

Uma grande desvantagem dos questionários é a possibilidade limitada (ou inexistente!) de feedback em caso de dúvidas. Isso se aplica tanto aos entrevistados quanto ao Engenheiro de Requisitos. Assim, tanto as perguntas quanto as respostas podem ser interpretadas incorretamente.

[BaCC2015] descreva também os seguintes pontos que devem ser observados:

- *Viés de seleção*: muitas vezes, por conveniência, são selecionados apenas os respondentes que são fáceis de encontrar (p. ex., o próprio departamento, amigos e familiares). Esse grupo não é representativo, o que pode resultar em dados imprecisos.
- *Viés de não resposta*: mesmo que os participantes aos quais o questionário é distribuído sejam selecionados de forma representativa, pode ocorrer um desequilíbrio, pois alguns deles responderão e outros não. A taxa de resposta pode variar entre 20% e 60%.
- *Satisfação*: Se o questionário exigir muito esforço cognitivo, eles podem aplicar a satisfação, uma estratégia para alcançar resultados satisfatórios, mas não ideais. Por exemplo, os entrevistados podem selecionar a mesma opção para todas as perguntas.

Variáveis

Além das variantes discutidas anteriormente de questionários qualitativos e quantitativos, há também as seguintes variantes:

- **Questionário em papel e lápis:**
O questionário é respondido em papel. Isso resulta em um grande esforço de processamento, pois os dados precisam ser computadorizados antes da avaliação. Para esse tipo de questionário, o esforço de avaliação aumenta com o número de respostas, tanto para questionários qualitativos quanto quantitativos.
- **Questionário computadorizado/baseado na Web:**
O questionário é respondido on-line. Para questionários quantitativos, as respostas podem ser agregadas automaticamente.

3.1.2 Técnicas de observação

O objetivo das técnicas de observação é extrair requisitos da observação de, por exemplo, processos, usuários ou situações típicas de uso.

Deve-se dar atenção especial ao viés de simplificação dos pesquisadores [BaCC2015]: Observadores inexperientes (ou seja, novatos no domínio) têm a tendência de simplificar demais as estratégias de solução de problemas do usuário especialista enquanto o observam. Portanto, é altamente recomendável adquirir conhecimento sobre o tópico (p. ex., conversar com um especialista no assunto) antes de usar técnicas de observação para minimizar esse viés. Também é aconselhável permitir que os especialistas no assunto revisem as notas de observação posteriormente.

Na prática, as três técnicas apresentadas neste handbook: observação de campo, aprendizado e investigação contextual, podem se sobrepor. Portanto, é importante fazer a transição de uma técnica para outra ou criar uma combinação consciente de técnicas. O que deve ser evitado é passar de uma técnica para outra, por exemplo, de uma pesquisa contextual para uma observação de campo pura, ou vice-versa. Se houver motivos para misturar as técnicas, misture-as, mas tenha em mente o que é importante para cada elemento.

3.1.2.1 Observação de campo

O Engenheiro de Requisitos observa os stakeholders durante o trabalho em seu ambiente habitual, sem interferir. As observações feitas são usadas para derivar requisitos que devem ser confirmados por meio de revisão ou de outras técnicas de elicitación. A observação de campo também é conhecida como acompanhamento de trabalho, acompanhamento de pessoas, observação ou observação pura.

O que é observação de campo?

Durante a observação de campo, o Engenheiro de Requisitos observa o interessado (geralmente o usuário final) em seu ambiente enquanto ele realiza as tarefas para as quais o sistema está planejado para ser desenvolvido ou aprimorado. A diferença importante entre a observação de campo e o aprendizado ou a pesquisa contextual é que não há interação entre o observador e o(s) sujeito(s) observado(s).

A observação de campo é normalmente usada em situações em que a interação com os usuários não é possível (p. ex., seria uma distração) ou interferiria no próprio processo e poderia falsificar os resultados. Ele também pode ser aplicado em locais públicos, mesmo sem informar os sujeitos observados (p. ex., sentar-se com outros pacientes em um consultório médico e observá-los durante o tempo de espera).

Papel dos participantes

O Engenheiro de Requisitos assume o papel de *observador*. A stakeholder é *osujeito observado*. Pode haver mais de um observador, bem como mais de um sujeito observado.

Preparação

Defina o(s) objetivo(s) da elicitação e a qualidade do resultado exigido como parte da definição da atividade de elicitação e selecione uma ou mais stakeholders adequadas para a observação.

Aqui, o objetivo da elicitação é especialmente importante, pois ajuda você a se concentrar nos aspectos relevantes durante a observação. Como nosso cérebro não é capaz de observar tudo o tempo todo, é de extrema importância saber exatamente o que você está procurando.

Para a observação de campo, o tempo e o local têm uma grande influência no resultado. Pode ser necessário fazer uma pesquisa prévia para descobrir a hora e o local certos.

Alguns fatores que influenciam o tempo e o local podem ser:

- O processo a ser observado ocorre apenas em determinados momentos (p. ex., pessoas iniciando o dia de trabalho, distribuição de alimentos em uma cantina).
- O processo ocorre em diferentes locais e em diferentes momentos (p. ex., pessoas entrando ou saindo do transporte público).
- Partes do processo geral a ser observado ocorrem em momentos e/ou locais diferentes (p. ex., uma peça é produzida em uma linha de produção e, em seguida, entregue a outra linha de produção, onde não é processada imediatamente).

Decida antecipadamente se os sujeitos observados serão informados de que estão sendo observados. Em qualquer situação em que seja óbvio que você está observando (p. ex., você observa alguém que normalmente trabalha sozinho em uma sala), você deve informar o sujeito observado com antecedência. Em outros casos, talvez nem seja possível informar os sujeitos da observação (p. ex., observação em um local público como um supermercado).

Aplicação

Quando o sujeito observado for informado sobre a observação, pode ser uma boa ideia informá-lo sobre o motivo pelo qual você está fazendo isso e que ele deve tentar esquecer a sua presença e fazer o trabalho normalmente.

Durante a observação de campo, o observador usa todos os seus sentidos (ver, ouvir, cheirar, sentir) para coletar informações sobre a(s) situação(ões)/processo(s) observada(s) e faz anotações sobre todas as informações que possam ser relevantes em relação ao objetivo da elicitação. Dependendo da situação, pode ser possível tirar fotos e/ou fazer vídeos.

Conforme mencionado na seção de preparação, é muito importante ter em mente o objetivo da elicitação durante a observação. No entanto, há também o risco de ser muito focado e, portanto, tendencioso durante a observação. Figurativamente falando, o objetivo da elicitação é o seu farol: ele o ajuda a navegar pela observação e a decidir se algo é relevante ou não para atingir o objetivo da elicitação.

Não participe nem interfira na(s) situação(ões) que está(ão) sendo observada(s).

Em alguns casos, pode ser possível realizar a observação de campo por meio de uma câmera (p. ex., câmeras de segurança previamente instaladas), evitando assim a interferência da presença física no processo.

[BaCC2015] sugere considerar os seguintes pontos durante a observação de campo:

- Que linguagem e terminologia as pessoas usam?
- Se você estiver observando o uso de um sistema existente, quanto do sistema/software/funcionalidades os usuários realmente usam?
- Quais são as barreiras ou pontos de parada que as pessoas encontram?
- Se o seu interesse for focado em tarefas:
 - Quanto tempo as pessoas dedicam à realização de uma tarefa?
 - Que perguntas as pessoas precisam fazer para realizar uma tarefa?
 - Com quais ferramentas os usuários interagem quando estão tentando realizar uma tarefa?

Processamento de resultados

Observação: o processamento de resultados em geral é idêntico para observação de campo e pesquisa contextual.

Cada sessão de observação de campo resulta em uma grande quantidade de dados brutos coletados. Isso precisa ser analisado e agregado. As possíveis técnicas de suporte para o processamento de resultados de uma observação de campo podem ser:

- Análise usuário-tarefa-sistema-contexto-problema, ou seja, aplicar a leitura baseada em perspectiva para extrair as informações sobre esses cinco aspectos dos dados brutos coletados. O resultado é um produto de trabalho intermediário no qual as análises subsequentes podem ser feitas. Essas análises devem ser realizadas na equipe, de preferência envolvendo os stakeholders e os membros da equipe técnica.
- Como alternativa, o material coletado pode ser processado com um diagrama de afinidade. [BaCC2015]
- A partir dos dados brutos coletados durante a observação de campo ou com base nos produtos de trabalho intermediários criados de acordo com os dois pontos anteriores, identifique e documente:
 - Grupos de usuários ou personas reais
 - Processos e atritos nesses processos
 - Modelo cultural, modelo físico e modelo de fluxo [BeHo1998]
 - Lista de produtos de trabalho importantes usados durante a conclusão da tarefa
 - Lista de problemas
 - Metas, problemas, necessidades e requisitos

Produtos de trabalho típicos

Durante a preparação, a execução e o pós-processamento de uma atividade de observação de campo, geralmente são criados ou atualizados os seguintes produtos de trabalho:

- Anotações da observação (p. ex., texto, desenhos)
- Produtos de trabalho coletados durante a observação (p. ex., fotos tiradas, vídeos)
- Resultados processados, por exemplo, conforme proposto por [BeHo1998]:
 - Esboços de conexões factuais interessantes
 - Modelo físico que mostra o contexto do trabalho
 - Modelo cultural que mostra a cooperação entre os usuários participantes
 - Modelo de fluxo mostrando as interrupções do processo

Oportunidades

A observação de campo é uma técnica muito valiosa se a interação com os usuários não for possível ou não for pretendida [BaCC2015]. Ela ajuda a coletar informações que não podem ser expressas verbalmente pelos stakeholders (p. ex., conhecimento implícito) e ajuda a entender a situação da stakeholder e, assim, ser mais empático nas interações subsequentes.

Uma grande vantagem da observação de campo é que ela não exige recursos adicionais dos stakeholders: elas realizam suas tarefas normalmente.

De acordo com [Koelsch2016], a observação de campo também pode ajudar a

- identificar fluxos de trabalho ou de processos,
- identificar o que incomoda o usuário,
- observar quaisquer passos estranhos que encontrarem, e
- identificar qualquer espaço para aprimoramento.

Desafios

A tendência do observador influencia o que ele realmente vê. Ele/ela pode achar que entende do que se trata uma determinada ação, mas pode estar totalmente errado. A observação de campo nunca deve ser usada isoladamente, mas sempre deve ser seguida de técnicas adicionais de elicitación (p. ex., entrevista).

Dependendo do que está sendo observado e de onde está sendo observado, a observação de campo pode consumir muito tempo, especialmente quando há longos períodos em que nada de relevante acontece.

Variáveis

[BaCC2015] também descrevem o "deep hanging out" como uma forma mais estruturada de observação. O observador se torna um usuário (p. ex., usando transporte público) e usa uma estrutura formal para organizar o processo de observação, por exemplo, concentrando-se nestes dez pontos focais: família e crianças, alimentos e bebidas, ambiente construído, posses, consumo de mídia, ferramentas e tecnologia, demografia, tráfego, acesso a informações e comunicação e experiência geral.

3.1.2.2 Aprendizagem

Fazemos aqui uma diferenciação rigorosa entre observação de campo e aprendizado: a observação de campo é não participativa, enquanto o aprendizado é participativo. Na literatura, há definições diferentes em que ambos os conceitos, ou mesmo todos os três conceitos, incluindo a investigação contextual, são chamados de aprendizado (consulte [RoRo2013]).

O que é aprendizado?

O aprendizado segue a ideia de mestres e aprendizes [RoRo2013]. O Engenheiro de Requisitos faz um breve estágio no ambiente em que o sistema a ser desenvolvido/melhorado posteriormente será usado (ou já está em uso).

Especialistas experientes no assunto ("mestres") ensinam o Engenheiro de Requisitos ("aprendiz") a fim de capacitá-lo a entender melhor o domínio e, portanto, a extrair melhor os requisitos. "Seja qual for o trabalho, parece sensato ter um entendimento justo dele antes de tentar fazer qualquer alteração" [RoRo2013].

A duração ideal do estágio depende de muitos fatores diferentes, por exemplo, complexidade do processo, alta repetitividade versus baixa repetitividade, disponibilidade de tempo do mestre e do aprendiz.

A diferença importante entre o aprendizado e a investigação contextual é que, no aprendizado, o Engenheiro de Requisitos realmente pratica o trabalho que está sendo investigado e aprende sobre o contexto e as tarefas não apenas observando e perguntando, mas principalmente fazendo – o que, por natureza, leva mais tempo do que a realização de uma investigação contextual. A última consiste apenas na observação do especialista realizando uma tarefa em seu contexto e na análise cognitiva verbal da tarefa junto com o especialista (sem execução prática pelo Engenheiro de Requisitos).

Papel dos participantes

O Engenheiro de Requisitos se torna o aprendiz. Sua tarefa é aprender com a stakeholder, que se torna o mestre.

Preparação

Defina o(s) objetivo(s) da elicitação e a qualidade do resultado exigido como parte da definição da atividade de elicitação e selecione uma ou mais stakeholders adequadas como mestres para o aprendizado (veja abaixo).

Assim como na observação de campo, o objetivo da elicitação é especialmente importante para o aprendizado, pois ajuda você a se concentrar nos aspectos relevantes durante o aprendizado.

O Engenheiro de Requisitos deve coletar antecipadamente todas as informações que possam ajudá-lo a se tornar um aluno melhor na situação de aprendizado, evitando assim a necessidade de fazer perguntas básicas durante o aprendizado. Dessa forma, o tempo gasto pelos stakeholders é usado de forma eficiente.

Do ponto de vista organizacional, é preciso encontrar um mestre adequado e ele deve estar disposto a participar do aprendizado. Se o mestre não estiver motivado para fazer o aprendizado, é melhor mudar para uma técnica de elicitación diferente. Ele/ela tem uma participação ativa e essencial nessa técnica. Sugerimos uma reunião preparatória com o mestre para esclarecer como e quando o aprendizado será realizado.

É uma boa ideia instruir o mestre a desconfiar se o aprendiz permanecer quieto por um longo período. Ele/ela deve então fazer perguntas como "no que você está pensando?" ou "tell me what's on your mind" (diga-me o que está pensando).

Aplicação

O mestre ensina ao aprendiz os processos e/ou ações relevantes. O principal objetivo do aprendizado é experimentar o contexto no qual a solução posterior será aplicada. Portanto, o Engenheiro de Requisitos deve executar pelo menos algumas das etapas do processo geral. O Engenheiro de Requisitos evita fazer suposições, mas expressa tudo o que conclui, permitindo assim que o mestre afirme ou corrija essas conclusões.

É importante fazer muitas perguntas durante o aprendizado para obter o máximo de informações possível e evitar falsas suposições.

O aprendiz faz anotações durante o aprendizado.

Em algumas situações, pode ser possível que o aprendiz realize a tarefa ou tarefas por conta própria, se o mestre considerar apropriado.

Processamento de resultados

Logo após o aprendizado, o Engenheiro de Requisitos deve anotar o maior número possível de aspectos do aprendizado, para preservar o conhecimento adquirido.

Em seguida, os dados coletados do aprendizado são analisados.

As perguntas a seguir podem ser usadas como orientação:

- Quais foram as dificuldades que tive?
- O que poderia ser facilitado?
- O que eu considerava complicado?
- Qual seria o maior benefício?

Produtos de trabalho típicos

Durante a preparação, a execução e o pós-processamento de uma atividade de aprendizado, os seguintes produtos de trabalho geralmente são criados ou atualizados:

- Notas sobre o aprendizado (geradas durante e depois)
- Produtos de trabalho coletados durante o aprendizado (p. ex., capturas de tela feitas, material de treinamento)
- Resultados processados (devem ser revisados pelo mestre!)

Oportunidades

O aprendizado ajuda a entender o ambiente do sistema e o domínio do aplicativo. O Engenheiro de Requisitos vivencia as dificuldades e as soluções alternativas usadas, em vez de apenas aprender sobre elas. A interação com os stakeholders durante o aprendizado geralmente melhora o relacionamento com elas, o que posteriormente melhora a comunicação com esses stakeholders. Além disso, como o Engenheiro de Requisitos já experimentou o ambiente de trabalho dos stakeholders, ele pode ser considerado parte do grupo e menos estranho. Isso também ajuda o Engenheiro de Requisitos a entender a situação do interessado e, assim, ser mais empático nas interações subsequentes.

Com o aprendizado, o conhecimento implícito pode ser extraído e transformado em conhecimento explícito.

Desafios

A qualidade de um aprendizado depende da motivação e das habilidades didáticas dos stakeholders. Portanto, é importante esclarecer esses aspectos antes do aprendizado. É necessário um investimento considerável por parte dos stakeholders: durante o aprendizado, elas não conseguem realizar seu trabalho, ou pelo menos não no ritmo habitual.

Variáveis

Se o Engenheiro de Requisitos não realizar ativamente as tarefas, mas apenas acompanhar o interessado por um longo período, perguntar e receber explicações do interessado, isso geralmente é chamado de "acompanhamento de trabalho".

Observação:

O acompanhamento de um trabalho por um curto período de tempo é uma variante da pesquisa contextual.

3.1.2.3 Pesquisa contextual

O que é pesquisa contextual?

A investigação contextual é uma técnica iterativa de coleta de dados de campo, em que o Engenheiro de Requisitos estuda em profundidade alguns usuários cuidadosamente selecionados para chegar a uma compreensão mais completa da prática de trabalho em toda a base de usuários [BeHo1998]. É uma combinação rica de observação e discussão. O Engenheiro de Requisitos (e outros membros da equipe de acompanhamento) observa os usuários em seu próprio contexto de trabalho. Além disso, o Engenheiro de Requisitos discute com os usuários suas tarefas e se envolve com eles para descobrir aspectos não articulados de seu trabalho. O objetivo é descobrir estruturas e padrões e descobrir como o usuário organiza seu trabalho.

A investigação contextual se baseia em quatro princípios:

- *Contexto*: Vá para o contexto do próprio usuário para observá-lo realizando suas tarefas.
- *Parceria*: Tente entrar em um relacionamento que seja uma parceria: Juntos, tentem entender a prática de trabalho do usuário.
- *Interpretação*: Desenvolver um entendimento compartilhado com o usuário sobre os aspectos do trabalho que são importantes.
- *Foco*: na preparação da investigação contextual, defina os objetivos da elicitación e direcione sua investigação para coletar os dados relevantes a fim de atingir os objetivos.

As principais diferenças entre o aprendizado e a investigação contextual são que o aprendizado geralmente leva muito mais tempo (geralmente de 1 a 3 dias) e que o Engenheiro de Requisitos no papel de aprendiz está realmente praticando as tarefas relevantes, enquanto uma sessão de investigação contextual geralmente leva de 60 a 90 minutos e o Engenheiro de Requisitos apenas observa e conversa com o stakeholder que está executando a tarefa.

Papel dos participantes

O usuário é o especialista no domínio. O Engenheiro de Requisitos é o especialista que reconhece as estruturas de trabalho, os padrões e as diferenças na organização do trabalho dos indivíduos. Nessa sessão de investigação, ambos são parceiros, tentando revelar juntos aspectos relevantes do trabalho do usuário.

Outros participantes assumem o papel de observadores. Cada um deles pode se concentrar em diferentes aspectos, por exemplo, produtos de trabalho usados, contexto físico, comunicação, processo, cultura etc.

Preparação

A preparação de uma pesquisa contextual depende do grupo-alvo. Quando o acesso aos usuários é limitado ou potencialmente caro (p. ex., operadores da bolsa de valores, gerentes de alto nível, usuários no exterior etc.), o esforço de preparação pode ser muitas vezes maior do que a execução real da consulta. Nesses casos, o entendimento do Engenheiro de Requisitos sobre o que ele verá deve ser estabelecido com antecedência (p. ex., entrevistando uma pessoa aposentada do grupo-alvo ou perguntando à equipe de suporte no contexto, participando do treinamento de usuários, assistindo a vídeos e estudando tutoriais ou outros materiais escritos). Se o grupo-alvo for facilmente acessível, o entendimento básico pode ser obtido por meio de uma única entrevista com alguém que conheça os usuários e, em seguida, a primeira investigação contextual pode se concentrar em realmente conhecer as tarefas e o contexto do usuário.

Essas são as tarefas gerais para a preparação de uma pesquisa contextual:

- Dependendo do objetivo da elicitação, determine o número, o tipo e o local dos usuários a serem visitados.
- Marque reuniões com os usuários selecionados e peça a eles que preparem tarefas típicas – ou específicas – nas quais trabalharão durante a investigação contextual.
- Se você planeja fazer uma série de investigações contextuais, prepare iterativamente cada sessão individual de investigação:
 - Quem vai participar (se possível, envolva os stakeholders e/ou outros membros da equipe para que eles também tenham a experiência dos usuários no campo)?
 - Dependendo do objetivo da elicitação: Qual deve ser o foco da pesquisa? Quem está se concentrando em que aspecto?
 - Como registrar as descobertas (prepare listas de verificação para suas anotações; verifique se é permitido fazer gravações de áudio ou vídeo, fotos, etc.)?

Aplicação

Ao chegar ao local, apresente as pessoas participantes e seu objetivo de elicitação. Explique a configuração da parceria: o usuário é o especialista para a tarefa em questão; o Engenheiro de Requisitos é o especialista para reconhecer as estruturas de trabalho, os padrões e as diferenças na organização do trabalho das pessoas. Em seguida, peça ao usuário que execute as tarefas preparadas e explique o que ele está fazendo e por quê. Se você tiver permissão, não se esqueça de iniciar a gravação.

Observe o usuário durante a investigação contextual, tire fotos (se permitido), colete produtos de trabalho com os quais o usuário trabalha, faça anotações e pergunte ao usuário regularmente sobre aspectos de como ele executa as tarefas. Dessa forma, o usuário é forçado a tornar explícitas as coisas que permitem que o Engenheiro de Requisitos descubra as atitudes, as metas, os problemas e as necessidades do usuário. Além disso, a equipe pode obter informações sobre os processos reais e os atritos no fluxo de trabalho, os aspectos culturais do usuário e de seus colegas de trabalho, o fluxo de informações, o ambiente físico, etc.

Após a investigação contextual, os Engenheiros de Requisitos às vezes conduzem uma entrevista com o usuário para analisar uma lista de perguntas preparadas. No entanto, isso seria uma inserção de outra técnica de elicitação. Conclua a pesquisa contextual com um resumo.

Processamento de resultados

Observação: o processamento de resultados em geral é semelhante para a observação de campo e a pesquisa contextual.

Cada sessão de investigação contextual leva a uma grande quantidade de dados brutos coletados. Isso precisa ser analisado e agregado. As possíveis técnicas de suporte para o processamento de resultados de uma consulta contextual podem ser:

- Análise usuário–tarefa–sistema–contexto–problema, ou seja, aplicar a leitura baseada em perspectiva para extrair as informações sobre esses cinco aspectos dos dados brutos coletados. O resultado é um produto de trabalho intermediário no qual as análises subsequentes podem se basear. Essas análises devem ser realizadas na equipe, de preferência envolvendo os stakeholders e os membros da equipe técnica.
- Como alternativa, o material coletado pode ser processado em um diagrama de afinidade. [BaCC2015]
- A partir dos dados brutos coletados durante a sessão de investigação, ou com base nos produtos de trabalho intermediários criados de acordo com os dois pontos anteriores, identifique e documente:
 - Grupos de usuários ou personas reais
 - Processos e atritos nesses processos
 - Modelo cultural, modelo físico e modelo de fluxo [BeHo1998]
 - Lista de produtos de trabalho importantes usados durante a conclusão da tarefa
 - Lista de problemas
 - Metas, problemas, necessidades e requisitos

Repita essas etapas para cada investigação contextual que realizar, enriquecendo assim sistematicamente sua compreensão dos aspectos mencionados anteriormente (usuários, tarefas, sistemas, contexto e problemas).

Produtos de trabalho típicos

Durante a preparação, a execução e o pós–processamento de uma consulta contextual, geralmente são criados ou atualizados os seguintes produtos de trabalho:

- Anotações e gravações de áudio da pesquisa contextual
- Produtos de trabalho coletados durante a investigação contextual (p. ex., fotos tiradas, vídeos, formulários preenchidos, anotações, listas, ferramentas, etc.)
- Resultados processados, por exemplo, conforme proposto por [BeHo1998]:
 - Esboços de conexões factuais interessantes
 - Modelo físico que mostra o contexto do trabalho
 - Modelo cultural que mostra a cooperação entre os usuários participantes
 - Modelo de fluxo mostrando as interrupções do processo

Oportunidades

Embora um desafio da observação de campo seja evitar interpretações errôneas do comportamento do usuário observado, é um recurso fundamental da investigação contextual abordar e verificar essas observações imediatamente no contexto. Além disso, os usuários não precisam pensar em como explicar adequadamente sua maneira de trabalhar em uma apresentação na sala de reuniões. Em vez disso, eles apenas realizam seu trabalho (tarefas que normalmente fazem todos os dias) e explicam ao Engenheiro de Requisitos o que estão fazendo no momento, refletindo assim seu próprio curso de ação e tornando explícito seu conhecimento especializado. O Engenheiro de Requisitos e o usuário discutem, com base em uma tarefa de trabalho recém-executada, os problemas, as relações factuais, as motivações e as oportunidades de melhoria.

A investigação contextual é a disciplina suprema para a elicitación de requisitos centrada no ser humano [RiFi2014]. Em um curto período de tempo, o Engenheiro de Requisitos pode efetivamente descobrir muitas informações sobre os usuários, seus objetivos e tarefas, os sistemas usados, o contexto relevante e os problemas e questões reais.

Desafios

Dependendo de fatores como o conhecimento existente do Engenheiro de Requisitos sobre o domínio, a disponibilidade de usuários adequados para realizar a consulta contextual, a localização geográfica do contexto a ser visitado e o número de grupos de usuários em potencial, o número de consultas contextuais a serem planejadas pode ser alto e o esforço de preparação e execução correspondente pode ser considerável.

Certifique-se de selecionar os usuários corretos. Como a investigação contextual é uma técnica de elicitación qualitativa, a distribuição dos usuários selecionados não precisa ser necessariamente representativa em um sentido estatístico, mas deve representar os grupos de usuários esperados.

Embora o Engenheiro de Requisitos possa orientar a investigação contextual e motivar o usuário a participar, uma investigação contextual só é possível se o usuário realmente participar dela. Com um usuário desmotivado, a consulta contextual não levará a resultados satisfatórios.

A pesquisa contextual não é apenas uma simples combinação de observação e entrevista. A preparação e a implementação de uma investigação contextual são completamente diferentes e exigem treinamento e experiência adequados.

Variáveis

A entrevista etnográfica condensada [DeDe2011] é caracterizada como uma abordagem "de cima para baixo" – em contraste com a abordagem "de baixo para cima" da investigação contextual – porque você começa com uma entrevista semiestruturada, seguida de uma observação de como os usuários realizam suas tarefas no contexto real, concentrando-se em processos e ferramentas. Os produtos de trabalho também são coletados e discutidos.

3.1.3 Técnicas de colaboração

As técnicas de colaboração visam à eliciação de requisitos, concentrando-se na colaboração entre os stakeholders. Nesse processo de cocriação, os stakeholders com diferentes perspectivas, geralmente usuários do sistema, estão diretamente envolvidas na coleta, evolução ou refinamento dos requisitos. Esses tipos de técnicas oferecem uma plataforma para discussão e permitem o feedback instantâneo dos stakeholders. Exemplos de técnicas de colaboração são os workshops de requisitos e a Engenharia de Requisitos baseada em multidões.

3.1.3.1 Workshops de requisitos

Workshop é um termo genérico para técnicas orientadas para grupos. Eles podem ser conduzidos de maneiras muito diferentes, incluir outras técnicas de eliciação ou até mesmo padrões de processo (p. ex., um workshop de Design Thinking em um desenvolvimento ágil). Os formatos dos workshops variam de pequenas reuniões informais a eventos organizados com várias dezenas ou até centenas de participantes.

O que é um workshop de requisitos?

Neste handbook, nos concentramos no workshop de requisitos, conforme definido em [Gottesdiener2002]:

"Um workshop de requisitos é uma reunião estruturada na qual um grupo cuidadosamente selecionado de stakeholders e especialistas em conteúdo trabalham juntos para definir, criar, refinar e chegar a um fechamento sobre os resultados (como modelos e documentos) que representam os requisitos do usuário."

Um workshop de requisitos difere de qualquer outra reunião em dois aspectos principais:

- É uma reunião facilitada (ou seja, planejada, estruturada e moderada).
- Seu foco é a eliciação⁶ de requisitos.

Papel dos participantes

Em um workshop de requisitos, há quatro funções distintas: *facilitador*, *participante*, *registrador* e *patrocinador do workshop*.

O Engenheiro de Requisitos assume a função de *facilitador*. É sua obrigação preparar (consulte a seção Preparação) e facilitar o workshop de requisitos (consulte a seção Aplicação), garantir que todas as informações relevantes do workshop sejam registradas (consulte a seção Aplicação) e derivar os requisitos das informações coletadas durante o workshop de requisitos (consulte a seção Processamento de resultados).

⁶ Esses workshops geralmente também abordam questões de resolução de conflitos. No entanto, esse não é o foco deste capítulo.

O facilitador deve ser neutro em relação ao resultado e não precisa ser um especialista em conteúdo. Isso pode até ser um obstáculo se o facilitador for um especialista em conteúdo, pois há o risco de que ele tenha uma participação (oculta) no resultado.

Os *participantes* são stakeholders. É tarefa deles criar os produtos do workshop, com base no objetivo do workshop e orientados pelo facilitador.

O *gravador* registra o trabalho do grupo. Isso pode ser feito por escrito e/ou por meio de fotos ou vídeos. Dependendo da complexidade do workshop de requisitos, essa função também pode ser desempenhada pelo Engenheiro de Requisitos, juntamente com a função de facilitador. O gravador também não deve ser um especialista em conteúdo. Embora ele/ela precise entender o suficiente sobre o tópico discutido para poder registrar as informações corretamente.

O *patrocinador do workshop* autoriza e legitima o workshop, incluindo os objetivos do workshop e a qualidade dos resultados. Ele/ela é responsável por garantir que todos os recursos necessários (ou seja, pessoas e dinheiro) estejam disponíveis.

Preparação



Escrito pelos autores.

Figura 19: Interdependências típicas das atividades de preparação do workshop

O *projeto de um workshop* influencia e é influenciado por vários fatores (consulte Figura 19). Como alguns desses fatores influenciam e são influenciados, a criação de um projeto de workshop é um processo iterativo.

Defina os *objetivos do workshop* (objetivos de elicitação) e a qualidade do resultado exigido como parte da definição da atividade de elicitação. Ambos os tópicos devem ser esclarecidos com o patrocinador do workshop.

A configuração da sala depende do *projeto do workshop* e, inversamente, o projeto do workshop depende da *configuração da sala*. Já a configuração da sala e as *características da sala* influenciam umas às outras.

Em um mundo ideal, o projeto da oficina seria independente das características da sala. Na maioria dos casos, entretanto, o quarto é escolhido com antecedência ou a seleção do quarto é restrita pela distância e/ou pelo orçamento. Portanto, na maioria dos casos, o workshop deve ser preparado de acordo com as características da sala, o que resulta na configuração da sala: Que material (p. ex., flipcharts, quadros de avisos, quadros brancos, pôsteres) será colocado em que lugar da sala? São necessárias tabelas? Como elas devem ser colocadas? São necessárias cadeiras? Como elas devem ser colocadas? Onde os resultados do workshop serão armazenados?

As características da sala e o *número de participantes* também são interdependentes: quanto mais participantes, maior a sala, ou quanto menor a sala, menos participantes. O projeto do workshop também depende do número de participantes. Alguns modelos de workshop não são tão escalonáveis quanto outros. O número ideal de participantes para um workshop de requisitos é de 7 a 12 participantes. O número máximo de participantes deve ser 16, a menos que haja mais de um facilitador [Gottesdiener2002].

Assim como as características da sala, *as ferramentas e os materiais* não deveriam ser fatores que influenciam o projeto da oficina, mas, em muitos casos, de fato são. Em geral, elas também são influenciadas pelas características da sala: por exemplo, se a sala não tiver um quadro branco enorme para trabalhar, ele não poderá ser usado como ferramenta durante o workshop e outras ferramentas deverão ser usadas, como post-its.

Quando essas interdependências se estabilizarem, os participantes poderão ser convidados. O facilitador precisa esclarecer com o patrocinador do workshop quem será convidado para o workshop. Esclareça também com o patrocinador se ele estará presente. Em geral, o patrocinador do workshop deve estar presente na abertura e, possivelmente, também no encerramento do workshop. Durante o workshop, a presença do patrocinador do workshop geralmente não é necessária e, em alguns casos, também não é recomendada (p. ex., se a posição hierárquica do patrocinador puder interferir nos resultados do workshop). Comunique o horário e o local, bem como os objetivos do workshop no convite. O ideal é que você já possa fornecer uma *agenda inicial do workshop*.

A *agenda do workshop* é o resultado do planejamento do workshop. Essa é a diretriz para o workshop. A agenda de um workshop deve permitir intervalos suficientes para que os horários possam ser ajustados durante o workshop. Algumas atividades do workshop podem precisar de muito mais tempo do que o planejado, enquanto outras podem ser mais rápidas do que o previsto.

Aplicação

Como a aplicação detalhada depende muito do projeto da oficina, nesta seção são fornecidas apenas orientações gerais.

Antes da chegada dos participantes, a sala deve ser preparada (p. ex., colocar pôsteres, mover mesas e cadeiras, etc.). Certifique-se também de que haja refrescos disponíveis.

Dê as boas-vindas aos participantes quando eles chegarem.

Abra o workshop e explique os objetivos do workshop e o resultado esperado (isso também pode ser feito pelo patrocinador do workshop).

Estabeleça regras básicas para o workshop, de preferência em cooperação com os participantes.

Conduza o grupo pelo workshop. Certifique-se de que todos os participantes estejam envolvidos e que todas as informações fornecidas pelos participantes sejam registradas. Continue verificando se algum dos seguintes itens precisa ser ajustado: atividades do workshop, cronograma, ferramentas usadas, clima da sala.

Encerre o workshop: Agradeça aos participantes por suas contribuições. Esclarecer as próximas etapas e responsabilidades. Informe-os sobre o que acontecerá com os resultados do workshop.

Processamento de resultados

Após o workshop, o facilitador ou registrador processa a documentação do workshop e a disponibiliza para os participantes.

Como os resultados do workshop, assim como as configurações do workshop, são altamente diversificados, não é possível fornecer diretrizes gerais para o processamento de resultados.

Produtos de trabalho típicos

Os produtos de trabalho dependem do tipo de workshop (consulte a seção Variantes), do projeto do workshop e do tipo de registro.

Normalmente, são geradas variantes dos seguintes produtos de trabalho:

- Agenda do workshop
- Documentação do workshop (p. ex., fotos, flipcharts, anotações digitais, atas)

Oportunidades

Os workshops de requisitos apoiam a comunicação e a tomada de decisões da equipe. Eles reúnem diferentes stakeholders para obter um melhor entendimento mútuo do projeto ou produto. Conflitos e mal-entendidos em potencial podem ser identificados e resolvidos durante os workshops de requisitos, o que pode não se tornar óbvio com outras técnicas de elicitación.

Desafios

Como os workshops de requisitos exigem muitos recursos (pessoas, tempo e dinheiro), eles dificilmente podem ser realizados sem um patrocinador comprometido. O tempo necessário para a preparação do workshop e o processamento dos resultados é frequentemente subestimado. Os resultados do workshop dependem muito do projeto do workshop. Facilitadores inexperientes tendem a subestimar o esforço necessário e muitas vezes não têm a flexibilidade e o conjunto de ferramentas para preparar e realizar um workshop eficaz, bem como para processar os resultados de forma eficiente.

Hint 3.14:

Se for novo na função de facilitador, você deve trabalhar em conjunto com um facilitador mais experiente em todas as fases do workshop.

Variáveis

Há muitas variantes diferentes de workshops de requisitos. Eles podem ser combinados com ou assumir a forma de outros modelos de oficina bem conhecidos, como:

- **World Café** [BrIs2005]:
Uma configuração de várias mesas. Cada mesa é dedicada a um tópico e tem um moderador. O grupo em uma mesa (4–6 pessoas) discute o tópico por 15–30 minutos. Em seguida, as equipes são misturadas e ocupam seus lugares em outra mesa.
- **Tecnologia de espaço aberto** [Owen2008]:
Os participantes sugerem tópicos para discussão. Vários tópicos são discutidos ao mesmo tempo em diferentes locais (salas). O proprietário do tópico é responsável pela documentação da discussão. Os participantes podem entrar e sair livremente de uma discussão (lei dos dois pés). No final do espaço aberto, os grupos se reúnem e os responsáveis pelos tópicos apresentam os resultados para todo o grupo.
- **Design Thinking**:
Consulte a descrição do padrão na Seção 1.4.4.

3.1.3.2 Engenharia de Requisitos baseada em multidão

O que é Engenharia de Requisitos baseada em multidões?

A Engenharia de Requisitos baseada em multidões (CrowdRE) também é conhecida como Engenharia de Requisitos centrada em multidões (CCRE).

Definimos CrowdRE de acordo com [Groen et al.2017] como "um termo abrangente para abordagens automatizadas ou semiautomatizadas para coletar e analisar informações de uma multidão para derivar requisitos de usuário validados".

Essas informações podem ser comentários fornecidos em um fórum, classificações e avaliações em lojas de aplicativos, dados de uso e outros. A mineração de texto e de uso são técnicas típicas usadas no CrowdRE.

Um dos principais benefícios do CrowdRE, no entanto, é o feedback coletado do público para apoiar a eliciação de requisitos. O CrowdRE conta com a colaboração das pessoas que fazem parte da multidão e dos Engenheiros de Requisitos.

Papel dos participantes

O principal participante do CrowdRE é "a multidão", geralmente composta por usuários atuais ou futuros do produto a ser desenvolvido. No entanto, isso não se restringe apenas aos usuários [LiFi2012].

Essa multidão pode já existir ou ser "criada" no decorrer do CrowdRE, por exemplo, criando uma experiência de gamificação que forma a multidão [Snijders et al.2015].

O Engenheiro de Requisitos não tem uma função dominante nessa técnica de elicitação. Ele ou ela reúne ou inicia as técnicas de coleta de dados da multidão e avalia os resultados. Em algumas formas de CrowdRE, os membros da multidão podem nem mesmo perceber que estão participando de um processo de elicitação de requisitos (p. ex., avaliações em lojas de aplicativos).

Como a aplicação do CrowdRE pode exigir um software especificamente desenvolvido (p. ex., plataforma de elicitação gamificada, ferramenta de análise de dados de uso, módulo de feedback dentro do produto a ser desenvolvido), o CrowdRE também pode exigir uma equipe de desenvolvimento para criar esse software.

Preparação

Como o CrowdRE pode assumir formas muito diferentes, não há uma forma padrão de preparação.

Em geral, pode-se dizer que é preciso definir o público certo para o produto a ser desenvolvido, bem como uma maneira apropriada de envolver o público e avaliar os resultados da elicitação, que geralmente são enormes quantidades de dados brutos (big data). [Snijders et al.2015] e [LiFi2012] sugerem métodos/ferramentas concretos.

Aplicação

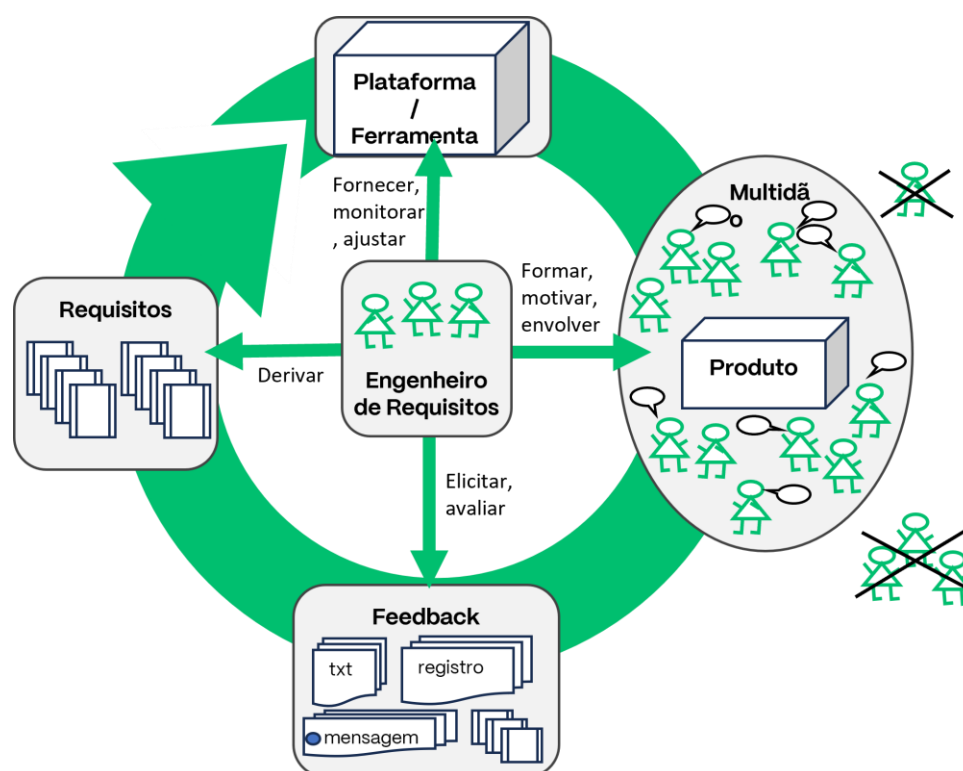


Figura 20: As relações entre os aspectos da Engenharia de Requisitos baseada em multidões de acordo com [Groen et al.2017]

De acordo com [Groen et al.2017], o Engenheiro de Requisitos precisa motivar os membros da multidão a fornecer feedback do usuário sobre o produto. Esse feedback é obtido e analisado pelo Engenheiro de Requisitos (p. ex., usando ferramentas de mineração de texto). Além disso, ele analisa os dados contextuais e de uso. Os requisitos derivados são implementados pela equipe de desenvolvimento. A nova versão do produto é devolvida ao público para que ele forneça feedback e valide os requisitos obtidos.

Processamento de resultados

Assim como na preparação, não é possível descrever um processamento típico de resultados. Normalmente, é necessário aplicar ferramentas e métodos para lidar com big data (p. ex., mineração de texto).

Produtos de trabalho típicos

Não aplicável.

Oportunidades

O CrowdRE oferece a oportunidade de coletar dados representativos de usuários reais. Com as ferramentas certas e uma configuração adequada, o CrowdRE tem o potencial de reduzir o custo geral da elicitación ao longo do tempo. O vínculo com o cliente pode ser aprimorado e sua criatividade pode ser usada diretamente, envolvendo-o ativamente. Além disso, os requisitos não podem ser apenas obtidos, mas também priorizados e validados pelo público.

Desafios

Geralmente, a configuração do CrowdRE é bastante demorada. Mesmo que sejam usados dados que já estejam "lá" (p. ex., classificações em uma loja de aplicativos), é necessário encontrar e aplicar maneiras adequadas de avaliar os dados.

Se forem fornecidas plataformas específicas para incentivar a CrowdRE, elas terão de ser desenvolvidas ou compradas e administradas. Se nenhuma multidão puder ser motivada a participar do CrowdRE, os resultados poderão ser inúteis. Por outro lado, se houver uma multidão muito ativa, são criadas enormes quantidades de dados que precisam ser avaliados para derivar os requisitos.

Outra possível armadilha do CrowdRE é a criação de uma visão tendenciosa, pois somente o feedback das pessoas que participam da comunicação em massa é obtido. Além disso, o CrowdRE é vulnerável a manipulações e pegadinhas (p. ex., em uma competição de crowdsourcing da Henkel para encontrar um novo design de embalagem para um detergente para lava-louças, uma capa com o slogan "Schmeckt lecker nach Hähnchen" (tem gosto de frango) obteve o maior número de votos [Katie2017]).

Variáveis

O Crowd Testing é um campo relacionado no qual a multidão é instrumentalizada para fornecer feedback valioso. Ao identificar um defeito (de usabilidade), os participantes de uma atividade de teste coletivo geralmente são convidados a dar sugestões para uma solução melhor. Nesse caso, pode ser útil cooperar com os testadores para coletar defeitos e requisitos com base na mesma multidão.

3.1.4 Técnicas baseadas em artefatos

Os artefatos são produtos do trabalho humano, como sistemas de TI, documentos, imagens, arquivos de áudio e vídeo, etc. Alguns tipos desses produtos de trabalho são relevantes como fontes de requisitos. Normalmente, é uma tarefa demorada examinar detalhadamente os produtos de trabalho, especialmente se esses produtos de trabalho contiverem muitas informações (irrelevantes) ou forem mal estruturados. No entanto, as técnicas baseadas em artefatos são muito potentes e oferecem benefícios significativos, principalmente quando os stakeholders não estão prontamente disponíveis.

Nesta seção, abordamos três técnicas baseadas em artefatos: leitura baseada em perspectiva, arqueologia do sistema e reutilização de requisitos. A leitura baseada em perspectiva é a principal técnica e uma habilidade básica para as outras duas técnicas. Há também alguma sobreposição entre as técnicas. Portanto, é importante entender a essência de cada técnica.

3.1.4.1 Leitura baseada em perspectiva

O que é leitura baseada em perspectiva?

A leitura baseada em perspectiva foi investigada pela primeira vez como uma técnica de revisão por [BaGL1996]. Eles descobriram que os revisores encontraram mais defeitos em menos tempo usando a leitura baseada em perspectiva, em comparação com a maneira usual de revisar documentos de requisitos. Como uma técnica de revisão, cada usuário assume a perspectiva de um usuário específico do documento de requisitos (p. ex., desenvolvedor, testador ou arquiteto).

Desde então, essa técnica também se mostrou valiosa como uma técnica de eliciação de requisitos ao lidar com documentos como fontes de requisitos.

Papel dos participantes

A única função necessária para a leitura baseada em perspectiva é a do leitor.

Preparação

Defina o(s) objetivo(s) da eliciação e a qualidade do resultado como parte da definição da atividade de eliciação (consulte 1.3.1).

Defina o documento de origem e selecione a(s) perspectiva(s) apropriada(s) para o documento de origem.

Quais perspectivas são adequadas dependem do documento de origem e do objetivo da eliciação? As perspectivas adequadas podem ser: requisitos para funcionalidade específica, requisitos de usabilidade, requisitos funcionais, requisitos não funcionais, etc.

Esclareça quantos leitores (Engenheiros de Requisitos) participarão da leitura baseada em perspectiva.

Aplicação

Um leitor só pode aplicar uma perspectiva em uma sessão de leitura. Se mais de uma perspectiva for relevante para um documento de origem, o Engenheiro de Requisitos deverá considerar essas perspectivas uma após a outra.

Se houver mais leitores, cada leitor terá uma perspectiva diferente. Os leitores podem ler os documentos em paralelo (se for um documento digital ou se houver várias cópias analógicas disponíveis), até mesmo na mesma sala, ou de forma assíncrona. Se for o caso, uma data-alvo deve ser acordada.

Ao ler o documento de origem, cada leitor examina o documento em busca de conteúdo relevante para sua perspectiva atual e, em seguida, lê com mais detalhes qualquer seção potencialmente interessante. Assim que encontram informações relevantes do ponto de vista deles, registram-nas de forma adequada.

Exemplo:

O documento de origem é o manual do usuário de um sistema concorrente. Um leitor adota a perspectiva de "requisitos de qualidade", enquanto outro leitor adota a perspectiva de "recursos impressionantes". Ambos examinam o manual do usuário, procurando informações relevantes do ponto de vista deles. Quando encontram uma seção potencialmente interessante no manual, eles a leem detalhadamente, extraíndo os requisitos. O primeiro leitor pode parar em qualquer frase que contenha números (indicadores de possíveis requisitos não funcionais); o segundo leitor pode parar em determinados capítulos em que a funcionalidade do sistema é descrita.

Processamento de resultados

As passagens identificadas como potencialmente relevantes para o sistema a ser desenvolvido devem ser verificadas pela aplicação de outras técnicas de elicitação ou por uma revisão dos requisitos. Sugerimos fortemente a combinação da leitura baseada em perspectiva com técnicas focadas nos stakeholders (p. ex., entrevista, investigação contextual).

Produtos de trabalho típicos

Durante a preparação, a execução e o pós-processamento da leitura baseada em perspectiva, geralmente são criados ou atualizados os seguintes produtos de trabalho:

- Notas da investigação original para documentos adequados
- Potencialmente: Cópias físicas ou digitais do(s) documento(s) com marcações e/ou anotações das seções relevantes
- Documentação das informações extraídas (notas de leitura)
- Resultados processados

Oportunidades

Quando os stakeholders não estão prontamente disponíveis, a leitura baseada em perspectiva é uma boa maneira de obter possíveis requisitos. Além disso, o conhecimento já documentado é reutilizado e não se perde.

Desafios

Muitas vezes é difícil saber se um documento ainda está atualizado e é válido. Ao usar documentos obsoletos para leitura baseada em perspectiva, os resultados úteis podem ser poucos para o tempo investido.

Como em todas as técnicas baseadas em artefatos, o Engenheiro de Requisitos nunca sabe quais requisitos extraídos da documentação permanecem relevantes para o sistema a ser desenvolvido. São necessárias outras técnicas.

Variáveis

Não aplicável.

3.1.4.2 Arqueologia de sistemas

O que é arqueologia sistêmica?

Na arqueologia de sistemas, os requisitos são extraídos de sistemas existentes: sistemas legados, bem como sistemas concorrentes ou até mesmo sistemas análogos (sistemas em um contexto diferente com funcionalidade semelhante). Na discussão a seguir, chamaremos esses *sistemas de origem*.

Essa técnica é usada principalmente se um sistema existente tiver sido usado (e possivelmente alterado) durante muitos anos e agora tiver que ser substituído por um novo sistema, porque, por exemplo, a tecnologia do sistema legado não é mais compatível com os sistemas vizinhos, seu desempenho não atende mais aos requisitos ou é necessária uma nova funcionalidade.

Papel dos participantes

A arqueologia sistêmica não envolve funções específicas.

Preparação

Defina o(s) objetivo(s) da elicitação e a qualidade do resultado como parte da definição da atividade de elicitação (consulte a Seção 1.3.1). O objetivo da elicitação também pode lhe dar uma dica de onde procurar: se, por exemplo, o objetivo da elicitação for elicitar os requisitos para os dados do sistema, o banco de dados do sistema pode ser um bom lugar para começar.

Selecione o sistema de origem do qual você gostaria de derivar os requisitos. Na maioria dos casos, esse seria um sistema em uso (sistema legado) que está prestes a ser substituído pelo sistema a ser desenvolvido. Entretanto, os sistemas concorrentes ou análogos ao sistema a ser desenvolvido também podem ser fontes para a arqueologia do sistema.

Em seguida, colete todos os documentos potencialmente relevantes para o sistema de origem, por exemplo, manual do usuário, casos de teste, documentação de arquitetura e cartas de projeto. Se for possível obter a documentação de requisitos, use a técnica de reutilização de requisitos (Seção 3.1.4.3).

Pode haver casos em que nenhuma documentação atual e útil para o sistema de origem esteja disponível, ou em que a documentação por si só seja insuficiente para a arqueologia do sistema. Nesses casos, você pode precisar do sistema de origem executável em um ambiente adequado (geralmente um ambiente de teste, no qual os dados podem ser manipulados sem o uso de dados reais) e/ou do código-fonte. Se você precisar extrair informações do código-fonte e não tiver treinamento na linguagem de programação usada, precisará de alguém para ajudá-lo.

Aplicação

O aplicativo difere dependendo se a documentação é a fonte para a arqueologia do sistema ou se é o código-fonte ou o próprio sistema.

Documentação

Se a sua fonte for documentação, como manuais de usuário ou casos de teste, aplique a leitura baseada em perspectiva (Seção 3.1.4.1).

Código-fonte

Dependendo da linguagem de programação usada, aplicam-se estratégias de leitura diferentes para o código-fonte (p. ex., linguagem de programação orientada a objetos ou assembler). Além disso, um código bem documentado será muito mais fácil de ler do que um código não documentado.

Em geral, é útil descobrir como o código é estruturado e quais convenções de nomenclatura são usadas, bem como familiarizar-se com o estilo de programação usado, antes de entrar em detalhes.

Normalmente, a análise do código-fonte só é necessária para descobrir detalhes específicos da implementação.

Exemplo:

O sistema de origem é executável em um ambiente de teste e você conseguiu extrair todos os principais recursos e funcionalidades. No entanto, há um recurso em que não é possível simular um comportamento específico no ambiente de teste, pois você não tem dados de teste suficientes. Portanto, você procura esse recurso específico no código-fonte e analisa as regras implementadas para ele.

Sistema de origem executável (análise da interface do usuário)

Se o sistema de origem estiver disponível em um formato executável, o sistema poderá ser analisado de forma estruturada. As boas práticas são:

- **Seguir os processos de negócio implementados no sistema:**
 - Siga um caso de negócios após o outro.
 - Procure exceções e caminhos alternativos.
 - Frequentemente útil: Documentar o conhecimento adquirido usando modelos (p. ex., usando BPMN, UML).
- **Siga a estrutura da IU:**
 - Analise uma tela após a outra, começando pela tela inicial.
 - Analise elemento por elemento, seguindo a direção de escrita do idioma usado (p. ex., da esquerda para a direita e de cima para baixo, no caso do inglês).
 - Documente o fluxo da tela e a funcionalidade dos botões com capturas de tela e/ou storyboards ou outros métodos adequados.

Processamento de resultados

Os requisitos identificados como potencialmente relevantes para o sistema a ser desenvolvido devem ser verificados pela aplicação de outras técnicas de elicitação ou pela revisão dos requisitos. Sugerimos enfaticamente a combinação da arqueologia sistêmica com técnicas voltadas para os stakeholders (p. ex., entrevista, investigação contextual).

Produtos de trabalho típicos

Durante a preparação, a execução e o pós-processamento da arqueologia do sistema, geralmente são criados ou atualizados os seguintes produtos de trabalho:

- Notas da investigação original para sistemas de origem adequados e sua documentação
- Potencialmente: Cópias físicas ou digitais do(s) documento(s) com marcações e/ou anotações nas seções relevantes
- Documentação das informações extraídas (notas de leitura, incluindo modelos ou desenhos)
- Resultados processados (p. ex., documentação baseada em modelos de processos identificados)

Oportunidades

A arqueologia do sistema garante que nenhum requisito implementado no sistema de origem seja perdido. É especialmente útil se nenhuma outra documentação atual estiver disponível para descobrir o que o sistema *realmente* faz. Além disso, não depende da disponibilidade dos stakeholders.

Desafios

A arqueologia do sistema pode consumir muito tempo, pois muitos documentos ou linhas de código precisam ser verificados quanto à relevância e lidos. Uma armadilha específica é que essa técnica ajuda apenas a descobrir a funcionalidade existente e não informa se essa funcionalidade ainda é necessária ou se está correta. Na pior das hipóteses, os defeitos do sistema antigo podem ser implementados novamente no novo sistema!

Variáveis

Não aplicável.

3.1.4.3 Reutilização de requisitos

O que é reutilização de requisitos?

Embora cada projeto seja, por definição, único e, como tal, resulte em um produto único, há uma boa chance de que em cada projeto haja potencial para reutilização, inclusive de requisitos (p. ex., funções e perfis ou procedimentos de login).

A reutilização consciente evita reinventar a roda várias vezes.

Há diferentes formas de reutilização de requisitos. No caso de linhas de produtos, uma grande porcentagem dos requisitos pode ser reutilizada, seja como está ou com alguma modificação.

Exemplo:

Os compensadores de fluxo da máquina do tempo DeLorean versão 1 e da máquina do tempo DeLorean versão 2 serão diferentes em alguns recursos específicos, mas a funcionalidade geral será a mesma.

Mesmo que dois sistemas pareçam, à primeira vista, totalmente diferentes, eles ainda podem ter requisitos semelhantes.

Exemplo:

O dispositivo de definição de hora e local e o sistema de aquecimento da mesma máquina do tempo têm funcionalidades completamente diferentes. Mas como ambos fazem parte do mesmo sistema (a máquina do tempo), eles compartilham uma alta porcentagem de requisitos não funcionais (p. ex., aceleração máxima).

Mesmo os sistemas que não têm interface entre si e não fazem parte de um supersistema comum ainda podem ter requisitos semelhantes, pois os processos que eles abrangem têm etapas semelhantes.

Exemplo:

Um avião e uma máquina do tempo têm algo em comum, pois ambos transportam pessoas. Alguns requisitos relativos a sistemas de suporte à vida (p. ex., fornecimento de oxigênio) ou requisitos de segurança provavelmente serão os mesmos para ambos.

A reutilização de requisitos tem três aspectos: o aspecto da elicitacão, o aspecto da documentacão e o aspecto do gerenciamento de requisitos. O aspecto de elicitacão abrange o princípio geral de reutilizacão e sua funçao na coleta de requisitos para um novo projeto. O aspecto da documentacão aborda a questao de como documentar os requisitos para dar suporte à reutilizacão posterior. O aspecto de gerenciamento de requisitos aborda a questao de onde e como organizar os requisitos para dar suporte à reutilizacão posterior. Neste handbook, abordamos apenas o primeiro aspecto.

Hint 3.1.5:

Se o seu produto estiver disponível em diversas variantes ou se houver várias versões, invista tempo e reflexao na configuracão da estrutura de documentacão e no uso de uma ferramenta de gerenciamento de requisitos para facilitar a reutilizacão dos requisitos obtidos.

Papel dos participantes

A reutilizacão de requisitos não envolve funçoes específicas.

Preparacão

Defina o(s) objetivo(s) da elicitacão e a qualidade do resultado como parte da definicão da atividade de elicitacão (consulte a Seçao 1.3.1).

Selecione os documentos que podem conter requisitos para reutilizacão. Os documentos típicos são especificaçoes de requisitos de sistemas de interface ou de versões anteriores do sistema a ser desenvolvido. Embora a reutilizacão de requisitos funcionais normalmente só seja possível entre sistemas semelhantes, os requisitos não funcionais também podem ser reutilizados entre sistemas que, à primeira vista, não parecem ter muitas semelhanças. Isso torna os requisitos não funcionais especialmente interessantes para a reutilizacão de requisitos.

[RoRo2013] sugerem a busca de documentos com conteúdo potencialmente reutilizável, percorrendo seu modelo *Volere*, [Robles2012], Capítulo 15:

1. **O objetivo do projeto:** Existem outros projetos na organizacão que sejam compatíveis ou que abranjam substancialmente os mesmos domínios ou áreas de trabalho?
2. **O cliente, o cliente e outros stakeholders:** Você pode reutilizar uma lista existente de stakeholders, um mapa de stakeholders ou uma planilha de análise de stakeholders? Usuários do produto: Outros produtos envolvem os mesmos usuários e, portanto, têm requisitos de usabilidade semelhantes?

3. **Restrições obrigatórias:** Suas restrições já foram especificadas para outro projeto? Há alguma restrição em toda a organização que também se aplique ao seu projeto?
4. **Convenções de nomenclatura e definições:** É quase certo que você pode usar partes de um glossário existente.
5. **Fatos e suposições relevantes:** Preste atenção aos fatos relevantes de projetos recentes. As premissas de outros projetos se aplicam ao seu projeto?
6. **O escopo do trabalho:** seu projeto tem grandes chances de ser um sistema adjacente a outros projetos que estão em andamento na sua organização. Usar as interfaces estabelecidas por outros modelos de contexto de trabalho. Considere o escopo de seu trabalho e pergunte se outros projetos já definiram eventos de negócios semelhantes.
7. **Modelo de dados comerciais e dicionário de dados:** Existem modelos de dados comerciais de projetos sobrepostos ou conectados que você poderia usar como ponto de partida?

Aplicação

Pesquise nos documentos identificados os requisitos que podem ser relevantes para o produto atual a ser desenvolvido. Essa é uma aplicação da leitura baseada em perspectiva (compare com a Seção 3.1.4.3). [RoRo2013] sugerem que se concentre nos verbos para encontrar requisitos reutilizáveis, pois eles representam processos.

Somente alguns requisitos reutilizáveis se apresentarão em uma bandeja de prata. Para encontrar os outros, você terá que interpretar e abstrair o que está lendo.

Processamento de resultados

Os requisitos identificados como potencialmente adequados para reutilização devem ser verificados pela aplicação de outras técnicas de elicitação ou pela revisão dos requisitos. Sugerimos enfaticamente a combinação da reutilização de requisitos com técnicas focadas nos stakeholders (p. ex., entrevista, investigação contextual).

Produtos de trabalho típicos

Durante a preparação, a execução e o pós-processamento da reutilização de requisitos, geralmente são criados ou atualizados os seguintes produtos de trabalho:

- Notas da investigação para documentos adequados
- Potencialmente: Cópias físicas ou digitais dos documentos de requisitos originais com marcações e/ou anotações em seções reutilizáveis
- Resultados processados, por exemplo, modelos abstratos de especificações existentes (consulte [RoRo2013])

Oportunidades

A reutilização de requisitos evita criar de novo o que já existe. Além disso, ajuda a não esquecer requisitos importantes. Se as especificações de requisitos usadas forem bem estruturadas e bem escritas, muitos requisitos poderão ser extraídos em um curto espaço de tempo.

Desafios

Se as especificações de requisitos usadas não forem bem estruturadas, bem escritas ou atualizadas, a reutilização de requisitos pode se tornar uma tarefa muito demorada e pode não produzir o resultado desejado.

Outro perigo em potencial é copiar requisitos errados que não se aplicam ao novo sistema ou que, de fato, já estavam errados no documento original. Além disso, ao copiar as soluções existentes, a nova inovação pode ser suprimida.

Variáveis

Não aplicável.

3.2 Técnicas de design e de geração de ideias (L2)

A tarefa tradicional da Engenharia de Requisitos era reunir e documentar os requisitos necessários de todas os stakeholders relevantes para um processo de desenvolvimento subsequente (cf. [Boehm2006]). Essa tarefa levou à aplicação das técnicas de coleta já apresentadas acima (consulte a Seção 3.1). A influência cada vez maior do software como um impulsionador da inovação em muitas empresas acabou levando a um novo entendimento da Engenharia de Requisitos como uma atividade criativa e de solução de problemas (cf. [Couger1996] ou [Maiden et al.2010]).

Fora da comunidade de software e Engenharia de Requisitos, surgiu o termo mais amplo, técnicas de design.

As técnicas de design compreendem técnicas de criatividade para a geração de ideias e fornecem técnicas adicionais ou combinadas para elaborar ideias e obter mais insights sobre uma determinada ideia [Kumar2013]. As técnicas populares de design incluem prototipagem (p. ex., maquetes), storyboarding e cenários.

Na discussão a seguir, apresentamos duas técnicas de criatividade (brainstorming e técnica de analogia) e duas técnicas de design (prototipagem e cenários/ storyboards) como exemplos de técnicas aplicáveis em vários contextos. No entanto, muitas outras técnicas podem ser encontradas na literatura (cf., por exemplo, [Kumar2013]). A disciplina de design também oferece vários princípios que apoiam, por um lado, o processo de desenvolvimento de sistemas (p. ex., conceitos de processo) e, por outro lado, o desenvolvimento do próprio sistema (p. ex., forma, formato ou funcionalidade do sistema). A apresentação e a discussão desses princípios estão além do escopo deste handbook. Uma boa visão geral dos princípios de design é apresentada, por exemplo, em [LiHB2003].

Condições prévias para a criatividade

Embora existam muitas técnicas que visam a gerar resultados criativos, nenhuma delas garante o sucesso. Vários mecanismos em nosso cérebro precisam se unir para permitir ideias criativas. Simplificando, as quatro condições prévias a seguir precisam ser atendidas para que a criatividade surja:

- **Chance** – e, portanto, tempo – para o surgimento de uma ideia:
Uma ideia é uma nova conexão entre os neurônios. O fato de essa conexão existir ou não é principalmente uma questão de sorte. No entanto, ela pode ser aprimorada influenciando positivamente as outras três pré-condições para a criatividade e, principalmente, dando-lhe tempo.
- **Conhecimento** do assunto, o que aumenta as chances de uma ideia que faça a diferença:
O conhecimento é o solo do qual as ideias criativas podem brotar. Ao adquirir conhecimento sobre um determinado tópico, as respectivas regiões do cérebro são ativadas, aumentando a chance de novas conexões neuronais. Isso significa que, se quisermos ter ideias novas e inovadoras sobre um tópico específico, precisamos primeiro adquirir conhecimento sobre esse tópico e depois pensar sobre ele para ativar as respectivas regiões do cérebro. Isso não significa que somente os especialistas no assunto podem apresentar inovações em seu campo de conhecimento; pessoas de fora podem apresentar ideias novas e não convencionais – **se** aprenderem o suficiente sobre o tópico em foco (p. ex., uma etapa específica em um processo que deve ser aprimorada ou a funcionalidade e as restrições de um produto existente que precisa de uma reforma).
- **Motivação**, pois nosso cérebro só pode ser criativo se houver um benefício direto para seu proprietário:
Ser criativo é uma tarefa que consome energia do nosso cérebro. A principal diretriz de nosso cérebro é: economizar energia. Dessa forma, ele só investirá a energia necessária se antecipar um benefício. Isso pode ser um aumento no status (ou seja, melhor chance de sobreviver) ou simplesmente se divertir.
- **Proteção e segurança**, como ideias inúteis, não devem ter consequências negativas:
Se temermos que uma ideia inútil ameace nosso status ou segurança de alguma forma, nosso cérebro desligará qualquer criatividade e "ficará em branco". Ele entra no modo de segurança e adere às maneiras conhecidas e seguras de fazer as coisas.

Com a aplicação de qualquer técnica de design ou de geração de ideias, é necessário garantir que os quatro pré-requisitos para a criatividade sejam atendidos pelos participantes.

3.2.1 Brainstorming

O que é brainstorming?

O Brainstorming foi desenvolvido por Alex F. Osborn [Osbo1948] como uma técnica de criatividade em grupo para apoiar o desenvolvimento de novas ideias para uma determinada questão ou problema. Como na maioria das técnicas de criatividade, o ponto crucial do brainstorming é adiar o julgamento de ideias, separando a ideia apresentada da sua análise. Uma boa visão geral do brainstorming em ER é fornecida por [MaGi2001] e [Pohl2010].

Papel dos participantes

Os participantes de uma sessão de brainstorming desenvolvem as ideias. Um moderador cuida das regras do brainstorming. Um anotador pode ser usado para ajudar o moderador a visualizar as ideias desenvolvidas.

Preparação

Antes de realizar o brainstorming, é necessário definir a pergunta principal ou o problema. O objetivo da elicitação (consulte o syllabus EU1) pode ser usado como ponto de partida ou até mesmo diretamente como uma pergunta principal. Em uma segunda etapa, os participantes da sessão de brainstorming devem ser identificados. Como regra geral, os participantes devem vir de várias áreas. Não é necessário que todos os participantes da sessão tenham um entendimento profundo da pergunta. Ter alguns participantes de diferentes domínios pode ser uma vantagem, pois eles podem oferecer perspectivas completamente diferentes sobre a questão.

Aplicação

No início de uma sessão de brainstorming, o moderador explica a pergunta principal e as regras do brainstorming a todos os participantes. [Pohl2010] descreve (com base em Osborn) estas sete regras para brainstorming na Engenharia de Requisitos:

- Quantidade em vez de qualidade
- A livre associação e o pensamento visionário são explicitamente encorajados e promovidos.
- Considerar e combinar ideias apresentadas é permitido e desejado.
- É proibido criticar as ideias de outros participantes, mesmo que uma ideia pareça absurda.
- Perguntas para esclarecimento são permitidas.
- Mesmo em deadlocks mais duradouros, não aborte imediatamente – supere pelo menos dois deadlocks mais duradouros.
- Espere até que o brainstorming chegue a um fim natural.

O moderador inicia a sessão e todos os participantes podem expressar suas ideias para a pergunta principal. O moderador (ou um anotador adicional) visualiza as ideias para todos os participantes. Durante uma sessão de brainstorming, o fluxo de ideias geralmente para depois de um certo tempo. Essas pausas não indicam o fim do brainstorming.

Uma nova declaração de um participante pode reiniciar o fluxo de ideias imediatamente. O grupo deve superar pelo menos dois desses intervalos antes que a sessão chegue a um fim natural (veja as regras acima).

Processamento de resultados

Após a realização da sessão de brainstorming, a lista de ideias deve ser analisada. Em geral, cada ideia consiste em uma breve declaração. Portanto, é preciso fazer um esforço adicional para detalhar as melhores ideias, por exemplo, realizando um workshop adicional com alguns dos participantes do brainstorming. Normalmente, as técnicas de priorização são aplicadas durante o processamento dos resultados do brainstorming.

Produtos de trabalho típicos

O resultado de uma sessão de brainstorming é a lista de ideias produzidas. Se todos os participantes do brainstorming concordarem, uma gravação de áudio ou vídeo do brainstorming pode ser um resultado adicional para análise posterior.

Chances

O brainstorming é útil para desenvolver um grande número de ideias em um curto período de tempo. As ideias que podem ser consideradas absurdas no início podem inspirar o grupo a desenvolver outras ideias inesperadas e inovadoras. O apoio a esse efeito de dinâmica de grupo (ou seja, ideias que se inspiram mutuamente) é um dos principais benefícios de uma boa sessão de brainstorming.

Desafios

Os resultados do brainstorming são apenas esboços de ideias. O trabalho de Engenharia de Requisitos começa após a sessão de brainstorming. As ideias desenvolvidas devem ser avaliadas, priorizadas e detalhadas.

Os efeitos da dinâmica de grupo são o principal fator para o desenvolvimento de boas ideias. Iniciar e manter uma atmosfera criativa na sessão de brainstorming é, portanto, o principal desafio.

Variáveis

Muitas variantes diferentes de brainstorming evoluíram ao longo do tempo, por exemplo:

- O brainstorming paradoxal segue o mesmo procedimento de um brainstorming normal. A diferença está no tópico do brainstorming, que é o oposto do tópico normal do brainstorming. O objetivo é explorar riscos e perigos em relação a um tópico (p. ex., "Quais fatores ajudam a irritar os possíveis compradores e fazem com que eles abandonem nossa loja on-line?").

3.2.2 Técnica de analogia

O que é a técnica de analogia?

A técnica de analogia (cf., por exemplo, [Robertson2001]) é uma técnica que ajuda a ter ideias para tópicos críticos e também complexos. Ele usa analogias para apoiar o pensamento e gerar ideias. Seu sucesso ou fracasso é influenciado principalmente pela qualidade da analogia. Boas fontes de analogias são os sistemas relacionados (consulte o Capítulo 2).

Papel dos participantes

O Engenheiro de Requisitos assume o papel de *moderador*, que desenvolve a analogia e a apresenta ao grupo.

Os *participantes* da técnica de analogia elaboram a analogia fornecida e transferem as ideias desenvolvidas para o problema original.

Preparação

A preparação para a técnica de analogia consiste em duas etapas.

Selecione uma analogia adequada para o problema dado. A analogia selecionada pode ser próxima (p. ex., o mesmo problema em outro domínio) ou distante do problema original. Por exemplo, o problema original é melhorar o processo de compra de produtos de seguro de vida pela Internet. Uma analogia próxima poderia ser um processo de compra de um produto diferente pela Internet (p. ex., comprar um livro em vez de uma apólice de seguro em uma loja on-line). Uma analogia distante poderia ser uma situação em que um garçom aconselha o cliente sobre a seleção do cardápio (o cardápio é a analogia para a apólice de seguro).

Selecione os participantes para a técnica de analogia. Os participantes devem ser especialistas no problema original. Se possível, devem ser convidados especialistas (ou pelo menos pessoas com conhecimento) na analogia selecionada. Não é necessário que um participante seja especialista tanto no problema original quanto na analogia. Também é possível selecionar a analogia junto com os participantes.

Aplicação

A aplicação da técnica de analogia consiste em duas etapas:

- Elabore a analogia selecionada. Os participantes elaboram os aspectos positivos e negativos da analogia em detalhes (p. ex., usando quadros brancos ou um flipchart). Essa etapa deve ser executada sem fazer referência ao problema original.
- Transferir para o problema original. Depois que a analogia é elaborada, os participantes examinam todos os aspectos da analogia e os transferem para o problema original.

Processamento de resultados

Os participantes criam uma lista de afirmações para o problema original que foram derivadas da analogia. Para chegar a um resultado final, os participantes devem examinar cada afirmação e discutir o valor da afirmação para o problema em questão. Durante essa discussão, os participantes podem elaborar ainda mais a declaração transferida para melhorar seu valor para o problema em questão.

Assim como na técnica de brainstorming, o resultado da técnica de analogia requer mais esforço para analisar e priorizar os resultados.

Produtos de trabalho típicos

O resultado da técnica de analogia é uma descrição detalhada da analogia, a lista de declarações transferidas para o problema original e a lista final de declarações avaliadas e detalhadas para o problema original.

Oportunidades

Uma analogia bem selecionada permite o desenvolvimento de ideias inovadoras para um determinado problema. Especialmente se o problema em questão estiver oculto para os participantes, a técnica de analogia pode gerar ideias inesperadas e não convencionais.

Se o problema em questão for difícil e/ou delicado, a técnica da analogia pode ajudar a criar um ambiente de trabalho aberto e construtivo. A parte desafiadora é, obviamente, a transferência das descobertas para o problema original.

Desafios

O principal fator de sucesso da técnica de analogia é a seleção de uma analogia adequada. Se os participantes estiverem cientes do problema original, é importante manter o problema dado fora da discussão da analogia.

Variáveis

Bisociation [Koes1964], derivado dos termos "bi" (duas coisas completamente diferentes) e "association" (associação), permite que os participantes de um workshop de criatividade associem ideias para um determinado enunciado de problema com algo que parece não ter nada em comum com o problema (p. ex., 5 fotos de pintores famosos, alguns objetos físicos especiais, vídeos curtos de animais interessantes).

3.2.3 Protótipos

O que é prototipagem?

Um protótipo é um termo genérico para qualquer produto de trabalho intermediário criado para investigar determinadas características ou soluções alternativas para um sistema a ser desenvolvido por meio de algum tipo de experiência tangível. Na maioria dos casos, está relacionado a características que não podem ser facilmente compreendidas ou definidas antecipadamente em modelos ou descritas na documentação.

Os protótipos podem variar de esboços em papel muito simples ou simulações de interface de usuário clicáveis a instâncias físicas de um dispositivo ou implementações iniciais de software; eles possuem características especificamente escolhidas – mas não todas – do futuro sistema e permitem a investigação de outras características ainda não claras.

É o uso do produto do trabalho para investigação, elaboração, esclarecimento, projeto, teste, validação, etc. (consulte [LiHB2003]) que o define como um protótipo. Em suma, o objetivo da prototipagem é experimentar (certas partes de) sistemas "... antes que eles sejam reais, mesmo antes de chegarmos ao seu projeto final, muito menos à sua implementação" [Buxton2007].

T.Z. Warfel [Warfel2009] descreve oito princípios orientadores para o uso da prototipagem:

- Entender seu público e sua intenção
- Planejar um pouco – prototipar o resto
- Definir expectativas
- Você pode esboçar
- É um protótipo, não a Mona Lisa
- Se você não pode fazer isso, finja
- Prototipar apenas o que você precisa
- Reduzir os riscos – com antecedência e frequência

Papel dos participantes

Na Engenharia de Requisitos, a prototipagem é usada tanto para elicitacão quanto para validação. Na elicitacão, os participantes geralmente têm uma função interna, como Engenheiros de Requisitos, designers digitais, designers de UX ou outros membros de uma equipe de desenvolvimento.

Seu principal objetivo é entender o problema, encontrar soluções adequadas (em um processo criativo) e, por fim, criar novos requisitos ou refinar os existentes. Na validação, os participantes geralmente têm uma função externa: eles podem ser usuários ou outros stakeholders fora da equipe de desenvolvimento que avaliam a implementação de determinados requisitos. Isso também pode levar a novos requisitos e ao refinamento dos existentes.

Em ambas as situações, é possível distinguir três funções principais:

- *Moderador*: Geralmente é o Engenheiro de Requisitos que decide usar a prototipagem como técnica de elicitacão, planeja e gerencia a prototipagem, instrui os outros participantes, analisa os resultados e tira as conclusões.
- *Desenvolvedor*: Um protótipo precisa ser projetado, construído e testado antes de poder ser usado. Para protótipos simples, como esboços, essa função pode ser desempenhada pelo Engenheiro de Requisitos, mas protótipos mais sofisticados podem exigir o apoio de designers, desenvolvedores ou especialistas em ferramentas.

- *Investigador*: A investigação do protótipo geralmente é um trabalho de equipe, no qual, dependendo do tamanho e da complexidade, podem estar envolvidos Engenheiros de Requisitos, designers, programadores, testadores, usuários-chave e outros stakeholders.
Para que a investigação se concentre nas características desejadas, é importante seguir as diretrizes ou os roteiros do moderador, mas também dar algum tempo para a exploração livre.

Preparação

Como os protótipos podem ser de natureza muito diferente, todos eles exigem uma abordagem diferente em sua preparação [McEl2017]. Para um simples esboço, lápis e papel podem ser suficientes; para um protótipo físico, uma impressora 3D e um software extenso podem ser necessários.

A única – e mais importante – coisa em comum é um plano: o Engenheiro de Requisitos deve ter uma boa ideia de quais características investigar e como fazê-lo. Também deve ficar claro quais outras características já estabelecidas (não sujeitas a investigação) devem ser implementadas e quais aspectos devem ser deliberadamente omitidos. Por exemplo, se você quiser criar um protótipo de uma loja on-line para analisar o desempenho, a funcionalidade em si deve estar funcionando.

É uma boa prática testar o protótipo para as características estabelecidas, pois os defeitos nessas áreas podem ter um impacto negativo na investigação das características que estão em foco. Além disso, é preciso comunicar claramente quais partes ou funcionalidades do sistema estão fora do escopo (ou seja, não estão disponíveis ou não funcionam no protótipo), especialmente quando há participantes externos envolvidos.

Dependendo do tamanho e da complexidade do protótipo, a preparação pode incluir o desenvolvimento de diretrizes, instruções, scripts, procedimentos e modelos para os pesquisadores (consulte, por exemplo, a preparação para um teste de usabilidade [UXQB2017]).

Aplicação

Os aplicativos mais comuns para prototipagem incluem [Warfel2009]:

- **Comunicação compartilhada**
Isso significa usar a prototipagem como uma ferramenta colaborativa, uma língua franca (linguagem comum) entre todos os participantes. O objetivo é criar um entendimento compartilhado dos requisitos entre a empresa, os Engenheiros de Requisitos, os designers, os desenvolvedores e os usuários. Em geral, envolve um simples esboço para determinar o contorno do futuro sistema e pode apresentar diversas variantes para serem escolhidas para desenvolvimento posterior.
- **Trabalhando em um projeto**
Os protótipos são uma ótima maneira de realmente trabalhar em um projeto, testá-lo, ver quais alternativas funcionarão e detalhar os detalhes. Está relacionado principalmente à elaboração e ao refinamento de requisitos anteriores de alto nível e ajuda a manter a integridade e a consistência.

- **Vender sua ideia internamente**
"Mostrar é melhor do que dizer." Ao criar um protótipo rápido de diferentes opções de design, é mais fácil convencer os stakeholders da empresa sobre os benefícios do seu design, fazer a escolha certa e obter apoio para ele.
- **Teste de usabilidade**
A usabilidade e outras características de qualidade são notoriamente difíceis de serem capturadas em especificações e modelos, mas fáceis de serem experimentadas em um protótipo, mesmo no início de um projeto.
- **Avaliação da viabilidade técnica e do valor**
Para cada projeto de um novo sistema, as perguntas finais permanecem: "Ele pode ser construído?" e "Isso agregará valor?". A construção e a operação de um protótipo podem gerar confiança ao fornecer respostas positivas a essas perguntas ou, se forem negativas, evitar falhas dispendiosas no projeto em um estágio inicial.

Processamento de resultados

Colete todas as descobertas da prototipagem e decida o que fazer com elas na equipe junto com as principais stakeholders (p. ex., em um passo a passo). Alguns comentários precisam ser analisados com um grupo maior de stakeholders; algumas ideias de melhoria precisam ser verificadas com a equipe técnica.

As atividades subsequentes dependem da situação do projeto: por exemplo, o desenvolvimento de um novo protótipo para explorar outras características do sistema, a escolha de uma variante preferida para desenvolvimento posterior ou a especificação detalhada dos requisitos descobertos para um projeto escolhido.

Produtos de trabalho típicos

- O protótipo
- Diretrizes, instruções, etc.
- Análise e documentação dos resultados e conclusões, como ideias, novos requisitos, esclarecimento de requisitos existentes para referência futura

Oportunidades

A prototipagem é uma ferramenta versátil que pode ser aplicada em vários estágios da Engenharia de Requisitos. Os estágios iniciais se beneficiam da exploração de ideias, enquanto os estágios posteriores se beneficiam da elaboração e do refinamento dos requisitos:

- A prototipagem permite que os Engenheiros de Requisitos obtenham feedback qualitativo dos stakeholders no início do projeto e, assim, evitem defeitos dispendiosos posteriormente.
- A prototipagem ajuda os Engenheiros de Requisitos a pensarem sobre o problema, a supervisionar e a reduzir a complexidade e, portanto, a se concentrar na entrega de valor ao cliente.

- A prototipagem pode ser usada como uma abordagem de design iterativa, participativa e conjunta que permite que a equipe de desenvolvimento e os stakeholders elaborem soluções em conjunto.
- A prototipagem se encaixa perfeitamente nas abordagens modernas de desenvolvimento, como Agile, DevOps, Lean e Design Thinking.

Desafios

Dependendo do tipo de protótipo selecionado, seu desenvolvimento pode exigir um esforço significativo. Por exemplo, um modelo funcional realista e clicável de uma interface de usuário (protótipo horizontal) ou um protótipo físico pode levar vários dias para ser projetado, desenvolvido e testado. O desafio é equilibrar o esforço de criação do protótipo com os benefícios esperados dos insights obtidos. Lembre-se do ditado "Um protótipo não é a Mona Lisa" e resista à tentação de tornar o protótipo um pouco mais perfeito.

Outro desafio (especialmente para protótipos sofisticados) é que o protótipo pode ser considerado o produto final por alguns stakeholders. Essa concepção errônea pode levar a expectativas irrealistas sobre o cronograma do projeto, pois os stakeholders geralmente subestimam o tempo necessário para transformar um protótipo em um software pronto para produção.

Variáveis

Embora os protótipos possam variar amplamente em termos de intenção e funcionalidade, três características principais são geralmente discerníveis: fidelidade, ciclo de vida e noção.

A fidelidade diz respeito à congruência do protótipo com o produto-alvo:

- Um *protótipo de baixa fidelidade (LoFi)* assemelha-se ao sistema futuro apenas o suficiente para permitir alguma experiência relevante com ele no que diz respeito às características pretendidas, por exemplo, esboços de tela em um storyboard.
- Um *protótipo de alta fidelidade (HiFi)* imita as interfaces externas do futuro sistema em um grau elevado, de modo que, à primeira vista, dificilmente se percebe a diferença (mais uma vez, no que diz respeito às características pretendidas). Internamente, "embaixo d'água", o sistema pode estar longe de estar completo.

O ciclo de vida é a relação de um protótipo com o produto-alvo:

- Um *protótipo exploratório* é criado apenas para fins de investigação e avaliação. Ele também é chamado de *protótipo "descartável"*, pois será descartado após o uso. Esse tipo de protótipo é comum na produção em massa.
- Um *protótipo evolutivo* será continuamente elaborado, ampliado, aprimorado e refatorado, até que se torne o produto final. Durante seu desenvolvimento iterativo, ele é usado primeiro para investigação e, em seguida, refatorado com base nas descobertas. Esse tipo de prototipagem é frequentemente encontrado em uma produção única. Pode-se argumentar que o desenvolvimento Agile/Scrum é geralmente um tipo de prototipagem evolutiva.

Noção, ou seja, os protótipos podem ser vistos de diferentes pontos de vista durante o desenvolvimento:

- *O feedback* pode ser o objetivo principal, por exemplo, quando um Engenheiro de Requisitos deseja avaliar um determinado aspecto da solução, como a interface do usuário, com um grupo mais amplo de stakeholders.
- *O design* pode vir em primeiro lugar. Esse é o caso quando um protótipo é criado para explorar e comparar diferentes soluções para um determinado problema, como geralmente é feito em projetos de design industrial.

Para obter uma compreensão mais aprofundada dos protótipos, não apenas no contexto da Elicitação de Requisitos, mas também no contexto do Design Digital, consulte [DDP2021].

3.2.4 Cenários e Storyboards

O que é um cenário?

A palavra cenário é derivada da palavra latina "scaenarium", que significa "local para montagem de palcos". Atualmente, a palavra cenário é usada para se referir a um esboço ou uma sinopse de uma peça⁷. No mesmo sentido, os cenários podem ser usados para criar um esboço do uso de um sistema [Carrol2003].

Os cenários podem ser documentados de forma escrita ou visual. A forma visual de um cenário é chamada de storyboard. Normalmente, um storyboard é uma série de painéis que mostram esboços de uma cena ou ação em uma série de tomadas (como em um filme, programa de TV ou comercial)⁸. Novamente, no mesmo sentido, um storyboard pode ser usado para descrever um fluxo de ações para um sistema.

Preparação

A preparação para a técnica de cenários consiste em uma decisão sobre a história ou histórias do cenário e na coleta de material e informações para o desenvolvimento dos cenários. A história do cenário refere-se à ação ou ações concretas que devem ser descritas. Lembre-se de que os cenários vêm do teatro e uma peça sempre tem uma história. A história pode ser derivada das metas ou dos problemas que precisam ser resolvidos. Esteja ciente de que definir o escopo da história é um desafio.

Se a história for muito grande (p. ex., um processo de negócio de ponta a ponta em uma loja on-line), o cenário poderá se tornar muito longo. Cenários longos são como livros muito longos: não são realmente legíveis.

Por outro lado, tornar o escopo muito pequeno também é um risco, pois o cenário se torna muito refinado e a pessoa pode se perder nos detalhes (p. ex., adicionar um produto a um carrinho de compras pode ser muito restrito para um cenário).

⁷ Veja, por exemplo, <https://www.merriam-webster.com/dictionary/scenario>

⁸ Veja, por exemplo, <https://www.merriam-webster.com/dictionary/storyboard>

Como regra geral, deve-se tentar identificar um escopo que descreva uma parte fechada de toda a história com um resultado visível (p. ex., a pesquisa de produtos em uma loja online). Em geral, um cenário deve se concentrar em apenas um escopo específico. Caso se queira descrever o mesmo cenário para diferentes escopos, cada combinação de cenário e escopo deve ser tratada como atividades de elicitação independentes (consulte o Capítulo 0).

As informações normalmente coletadas durante a preparação incluem: informações sobre usuários típicos, local típico em que o sistema é usado e eventos ou situações importantes que devem ser considerados no cenário.

Além disso, o formato da documentação do cenário deve ser selecionado (consulte os produtos de trabalho típicos para obter uma lista de formatos).

Aplicação

A aplicação da técnica de cenários é simples: comece a desenvolver o cenário. O desenvolvimento pode ocorrer em um ambiente de grupo ou como trabalho individual. O trabalho em grupo é aconselhável se o contexto do cenário exigir várias competências (p. ex., um processo de negócio complicado).

Durante o desenvolvimento de um cenário, deve-se sempre criar alternativas diferentes. Isso permite a exploração de versões alternativas, aumentando as chances de resultados finais ideais. Se o cenário for usado como um resultado intermediário (consulte processamento de resultados), não será necessário consolidar as versões desenvolvidas em uma versão final. A apresentação de versões alternativas de um cenário às stakeholders promove a discussão e também permite que a opinião delas sobre as alternativas seja incorporada ao processo de desenvolvimento.

Processamento de resultados

Os cenários podem se tornar parte de uma especificação de requisitos ou um resultado intermediário. Se for o caso, o cenário deve ser documentado de acordo com os padrões do projeto. Se for o último caso, serão necessárias outras atividades para derivar os requisitos do cenário. Essa etapa é uma técnica subsequente e não faz parte da técnica de cenário.

Produtos de trabalho típicos

Normalmente, os cenários são descritos por escrito. A forma mais simples é um texto em prosa:

John Doe espera na estação de ônibus "Market Street". Está frio e chuvoso e ele quer ir para casa rapidamente. Infelizmente, o ônibus parece estar atrasado. John quer verificar o horário de chegada do ônibus. Ele pega seu smartphone, abre o aplicativo de transporte público e seleciona a função "partidas da minha estação". O aplicativo usa as funcionalidades do smartphone (p. ex., GPS) para identificar a estação "Market Street" como a localização atual de John e apresenta o horário atual, incluindo o atraso de 11 minutos.

Uma abordagem mais estruturada é uma descrição de cenários com etapas específicas:

1. John Doe espera na estação de ônibus "Market Street". Está frio e chuvoso e ele quer ir para casa rapidamente. Infelizmente, o ônibus parece estar atrasado. John quer verificar o horário de chegada do ônibus.
2. O usuário pega seu smartphone e abre o aplicativo de transporte público.
3. O aplicativo apresenta o menu ao usuário e ele seleciona a função "partidas da minha estação".
4. O aplicativo usa a funcionalidade do smartphone (p. ex., GPS) para identificar a estação "Market Street" como a localização atual do John.
5. O aplicativo apresenta o horário atual, incluindo o atraso de 11 minutos.

A diferença entre o cenário contínuo e o cenário estruturado é a divisão em etapas individuais. Cada etapa normalmente se refere a uma interação entre o usuário e o sistema ou a uma atividade do usuário no contexto do sistema (etapa 1 no exemplo). Os cenários são sempre específicos em relação a pessoas (John Doe), quantidades (11 minutos), local (estação Market Street) e contexto relevante (frio e chuva). Isso ajuda a dar vida ao cenário: ser uma história real em um contexto real. Isso é importante para que os participantes entendam do que se trata a solução e os ajuda a falar sobre questões e tópicos muito específicos.

Observação:

Não confunda cenários com casos de uso [Cockburn2001]. Uma especificação de caso de uso é uma técnica para documentar a interação genérica entre um ator e o sistema. Os cenários, por outro lado, são mais parecidos com casos de teste (na verdade, eles podem formar a base para casos de teste). Eles podem ser vistos como uma instância específica de movimentação em um caso de uso.

Os storyboards já foram mencionados na introdução e são uma forma de representação visual de cenários. Os storyboards são úteis se o cenário tiver muita ação e se o contexto puder ser mais facilmente visualizado do que descrito.

Oportunidades

Os cenários são uma técnica boa e leve para a elaboração inicial de ideias em termos de processos e atividades. Eles podem ser usados para discutir e explorar formas alternativas de realizar um processo em um sistema. Devido à sua estrutura leve, eles são fáceis de desenvolver e podem mudar rapidamente. Como regra geral, todo projeto de desenvolvimento deve ter um entendimento comum – e explícito – dos cenários que o sistema em desenvolvimento deve suportar. A técnica de cenário é, portanto, uma boa candidata para a elicitación antecipada.

Desafios

A criação de bons cenários requer boas habilidades de criação (p. ex., estruturar a história ou usar palavras expressivas para descrevê-la). Sem essas habilidades, os cenários geralmente se tornam enfadonhos e boas ideias podem se perder por causa de uma apresentação fraca. Um escritor habilidoso também evita cenários excessivamente detalhados, nos quais muitas informações desnecessárias são apresentadas ao leitor.

Além das habilidades de criação, a aplicação de storyboards requer ainda a capacidade básica de desenho para criar imagens reconhecíveis para cada etapa. Esse nível básico não é difícil de alcançar: um storyboard simples consiste em bonecos de palito e esboços simples. O desafio mais difícil pode ser superar as inibições ao mostrar seus esboços para outras pessoas!

Variáveis

Normalmente, os cenários são descritos em um sentido positivo, ou seja, o cenário termina com um resultado positivo e o usuário alcança o que deseja. Uma boa variante para cenários é concentrar-se no resultado negativo e descrever o que pode acontecer se o cenário não terminar de forma positiva. Por exemplo, no cenário da estação de ônibus acima, o GPS pode não funcionar. O que o aplicativo apresentará ao usuário? Essa maneira de analisar um cenário normalmente oferece muitos insights adicionais e novas ideias para o sistema em desenvolvimento.

Uma terceira variante de cenários são os casos de uso indevido [SiOp2005]. Um caso de uso indevido se concentra em um uso indevido intencional das funcionalidades do sistema para causar danos ao usuário ou a outros stakeholders. Eles são especialmente úteis para obter requisitos relacionados à segurança do sistema.

Em vez de desenhar storyboards, a equipe poderia representar cenas e tirar fotos que, em seguida, poderiam ser deliberadamente transformadas usando filtros em um programa de pintura.

3.3 Ferramentas de pensamento

As seções anteriores explicam várias técnicas comuns que se mostraram bem-sucedidas na elicitación de requisitos. Eles descrevem maneiras de coletar informações e produzir produtos de trabalho para a documentação e a comunicação de requisitos.

Esta seção se concentra em determinadas técnicas de suporte. Elas não são usadas isoladamente, mas em conjunto com outras técnicas de elicitación. Nós as chamamos de *ferramentas de pensamento*, porque elas pretendem estimular uma forma de pensar ou criar uma mentalidade que contribua para o sucesso da própria elicitación.

3.3.1 Pensar em níveis de abstração

Os níveis de abstração são uma poderosa ferramenta de raciocínio na elicitación de requisitos [GoWo2005], [Laue2014]. Eles são frequentemente usados como um tipo de modelo de processo para estruturar o trabalho de elicitación, ou seja, primeiro elicitamos os requisitos apenas no nível mais alto e continuamos com os níveis mais baixos posteriormente.

Para entender os níveis de abstração como uma ferramenta de pensamento, é necessário entender o termo "abstração". De acordo com o Cambridge Dictionary⁹, abstração é "a situação em que um assunto é muito geral e não se baseia em situações reais".

Níveis de Abstração

alta	↑	<i>abstrato, irreal, geral</i>
		Exemplos:
		Objetivo de alto nível: Queremos fornecer todos os livros sobre a 2ª Guerra Mundial
		Todo o sistema: Aplicativo da Web que fornece livros sobre a 2ª Guerra Mundial
		Subsistema: Mecanismo de pesquisa, site, banco de dados, sistema de pagamento, etc.
		Épico: Como usuário, quero pesquisar livros e obter uma lista de resultados
		Diagrama de Casos de Uso: Aplicativo da Web que fornece os seguintes casos de uso: Pesquisar livro, Obter resultados, Ver detalhes do livro, ...
		Modelo de dados de alto nível: Ilustrando as classes principais e seus relacionamentos
		Caso de uso essencial: Caso de uso que especifica os objetivos do usuário (em vez de suas ações específicas)
médio		
		Característica, recurso, funcionalidade: Pesquisa ampliada, Fornecer uma resenha/crítica sobre um livro específico, Mostrar livros relacionados, ...
		Protótipo de baixa fidelidade: GUIs de wireframe em contexto umas com as outras, para que o fluxo de navegação possa ser explorado
		Histórias de Usuário: Como usuário, quero pesquisar livros de um autor específico.
		Modelo de dados detalhado: Ilustração de todas as classes, seus atributos e relacionamentos entre elas com informações específicas
		Protótipo de alta fidelidade: Representação visual muito concreta do programa final, ainda sem lógica em segundo plano
		Especificação do caso de uso: Especificação detalhada de cada ação do usuário (ator) e do sistema
		Cenário/Storyboard: Simon quer aprender sobre as experiências de seu avô em Dünkirchen, em 1940, quando lutava...
		Especificação da Interface: Para cada atributo, especifique os campos no sistema de envio/recebimento e as regras de transformação
Baixa	↓	<i>concreto, real, particular</i>
		Código: Se... então... faça... repita... -> Programa real que especifica em particular o que o computador deve fazer

Figura 21: Representação do conceito de níveis de abstração, ilustrada com exemplos esclarecedores

Na ciência da computação, a abstração é obtida por meio da ocultação de informações [CoSh2007]. As linguagens de programação e as APIs são bons exemplos disso. Cada instrução em uma linguagem de programação e cada método em uma API fornecem uma funcionalidade específica que está oculta por trás do nome da instrução ou do método. Caso alguém queira saber os detalhes, é possível descobri-los lendo o código da função ou do método. O objetivo da abstração é reduzir a complexidade, ocultando informações detalhadas por trás de uma construção única e mais simples. A seguir, descreveremos dois exemplos de aplicação de camadas de abstração na Engenharia de Requisitos.

Três camadas de abstração essenciais na Engenharia de Requisitos

Na Engenharia de Requisitos, temos uma fronteira intelectual natural que pode ser usada para reduzir a complexidade: A fronteira entre o sistema e o contexto (limite do sistema, consulte [IREB2020]). O contexto trata de tudo o que é visível e pode ser experimentado pelo usuário ou por outro sistema. Isso inclui a interface do usuário e as interfaces técnicas. Quando discutimos os requisitos de um sistema no nível de abstração do contexto do sistema, não é necessário falar sobre os detalhes técnicos que realizam esses requisitos.

⁹ <https://dictionary.cambridge.org/dictionary/english/abstraction>, validado em 2.7.2019

Os Engenheiros de Requisitos, por exemplo, fazem isso com um diagrama de casos de uso ou um mapa de histórias de usuários, visualizando a funcionalidade fornecida pelo sistema aos atores no contexto e omitindo todos os detalhes de implementação.

O próprio sistema pode ser subdividido em um sistema lógico e um sistema técnico. O sistema lógico é uma descrição idealizada e inclui estruturas de dados lógicas, funcionalidades e descrições comportamentais de determinados elementos no nível do contexto do sistema (p. ex., a especificação dos casos de uso mencionados acima). O sistema técnico refere-se à realização em termos de hardware, estruturas de dados técnicos e componentes de software (p. ex., bancos de dados e estruturas). Como exemplo, considere o sistema operacional de um computador: essa é a API para os detalhes técnicos do hardware, por exemplo, como elementos específicos da memória são endereçados ou como o processador executa operações individuais, de forma simultânea ou não.

A separação entre o contexto do sistema, o sistema lógico e o sistema técnico pode ser usada para:

- Estruture as especificações de requisitos para tornar o conteúdo complicado mais acessível.

Por exemplo, uma interação entre um usuário e um sistema é descrita no nível do contexto, enquanto a estrutura de dados detalhada e os requisitos funcionais são especificados no nível do sistema em uma parte posterior da especificação. O leitor de uma especificação pode, então, decidir se deseja ler os detalhes (p. ex., de um modelo de dados) ou ignorar os detalhes em favor de um entendimento geral.

- Estruturar o processo de desenvolvimento de uma especificação.

Por exemplo, um dos extremos seria seguir uma abordagem ampla, entendendo primeiro o máximo possível no nível do contexto antes de detalhar o nível do sistema. Uma abordagem alternativa extrema seria uma abordagem profunda, que elabora um aspecto no nível do contexto com todos os detalhes necessários no nível do sistema antes de continuar com o próximo elemento no nível do contexto. Para fins práticos, é recomendável uma combinação de ambas as abordagens.

Por exemplo, os aspectos críticos para o sucesso de um sistema devem ser elaborados com uma abordagem que prioriza a profundidade, ao passo que os aspectos simples ou adequadamente compreendidos podem ser elaborados com base na amplitude, pois os detalhes são claros de qualquer forma e o risco de omitir detalhes importantes é baixo.

3.3.2 Pensar em termos de problemas e metas

O desenvolvimento de um sistema geralmente se baseia em um determinado problema vivenciado por um cliente. Em outras ocasiões, o desenvolvimento é acionado pelo desejo de um cliente de atingir uma determinada meta. Mas, na maioria dos casos, os clientes não são muito claros, nem certos, nem abertos sobre seus verdadeiros problemas e objetivos.

Às vezes, um cliente propõe uma solução sem nem mesmo conseguir explicar que problema ela resolveria ou que objetivo ela deveria alcançar!

Portanto, é essencial que o Engenheiro de Requisitos não considere como certo que um único problema ou objetivo seja o ponto de partida para o desenvolvimento. É necessária uma análise completa da situação em questão para revelar todo o panorama de problemas e objetivos inter-relacionados antes que uma solução viável possa ser projetada (consulte [LoSL2017]). Primeiro, precisamos esclarecer as definições.

Um *problema* é um aspecto no contexto do stakeholder que atualmente é vivenciado de forma negativa. Uma experiência negativa prevista no contexto do stakeholder no futuro é chamada de *risco* (ou seja, um problema futuro em potencial). Muitas vezes, um determinado estado no contexto é percebido como um problema porque *inibe* o interessado de fazer algo desejável ou de atingir uma meta.

Uma *meta* é um aspecto positivo previsto no contexto do stakeholder no futuro. Muitas vezes, um determinado estado futuro no contexto é percebido como uma meta se ele *permitir que* o interessado faça coisas desejáveis.

É importante perceber que os problemas e as metas são *construções mentais*: eles não existem no mundo real, mas apenas na mente dos stakeholders. Portanto, eles só podem ser encontrados por meio de elicitación. Como consequência, um determinado estado de contexto (futuro) pode até mesmo ser um objetivo para uma stakeholder, enquanto constitui um problema em potencial para outra.

Os problemas e as metas estão sempre conectados uns aos outros. Um problema implica o objetivo de mudar o estado negativo em uma direção positiva. Uma meta só é reconhecida como tal se algo no estado atual impedir que ela já esteja acontecendo.

Os problemas e as metas também estão conectados uns aos outros por outra construção mental: a *solução*.

Uma solução é o roteiro para uma intervenção no contexto da stakeholder: ela descreve uma maneira pela qual o estado negativo atual pode ser transformado em um estado desejado no futuro. As soluções são desenvolvidas por meio de um processo de design criativo que começa com os problemas e as metas levantados.

Problemas e metas explícitos podem ser ocorrências de problemas ou metas implícitos de nível superior ou inferior. Eles nunca vêm sozinhos: fazem parte de uma grande família de pais e filhos. Os irmãos dessa família devem ser descobertos para se ter uma visão completa.

Os pais de um determinado problema podem ser encontrados por meio da busca de causas. *O que causa esse problema?*

Os pais de uma meta podem ser revelados pela análise do comportamento que é ativado quando a meta é atingida. *Por que* o interessado tem essa meta?

As crianças da família (os problemas e metas de nível inferior) podem ser encontradas por meio de soluções. Cada ação de uma solução viável define uma nova meta (e um novo

problema) em um nível inferior para alguém que seja responsável por implementá-la e que tenha o desafio de como fazer isso de forma eficaz.

O modelo de classe a seguir mostra a relação entre esses conceitos. Um problema inibe uma meta e é causado por problemas de nível superior. Uma meta possibilita metas de nível superior. Um problema pode ser resolvido por uma solução que pode atingir o objetivo desejado. Uma solução define determinadas ações a serem executadas. Cada ação define uma ou mais metas de nível inferior para sua implementação.

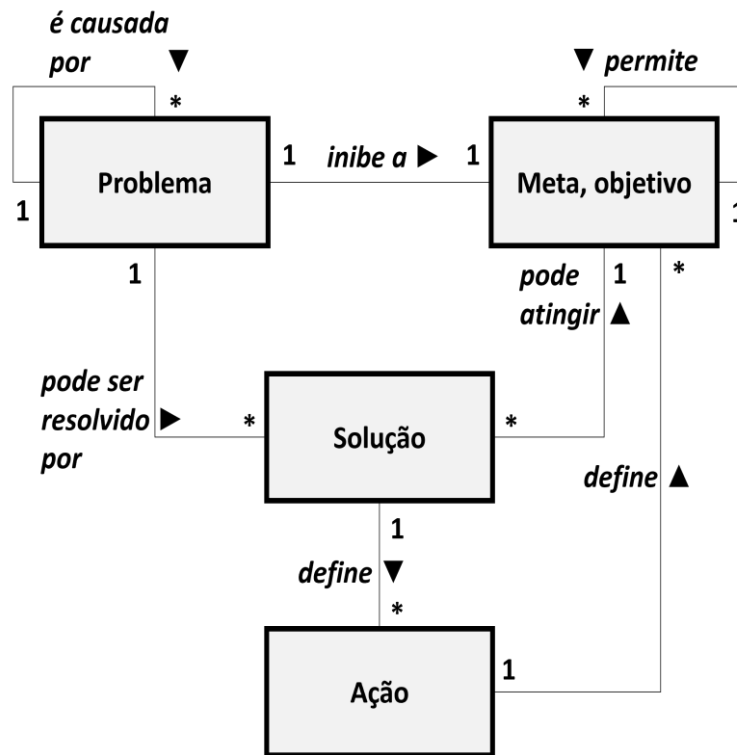


Figura 22: A relação entre problemas, metas e soluções

Pensar em termos de problemas e metas não é uma técnica própria. Trata-se de uma mentalidade que incentiva o Engenheiro de Requisitos a se aprofundar no contexto dos stakeholders antes de adotar uma solução aparentemente óbvia, mas talvez inadequada. Essa mentalidade é relevante para todos os tipos de técnicas de elicitação, sempre que um problema ou uma meta for encontrado.

Para um problema, tente descobrir:

- É um problema no contexto atual (de qual stakeholder) ou é uma expectativa de um problema futuro (então pense em maneiras de evitar que esse futuro se torne realidade)?
- Que situações no presente ou no passado causaram esse problema (isso pode lhe dar ideias sobre possíveis soluções)?
- Quais ações futuras são inibidas pelo problema (a meta conectada)? O que cada stakeholder (poderia) fazer se o problema não existisse?

Para um objetivo, pense em:

- Que situações no presente (ou expectativas sobre o futuro) impedem que a meta seja alcançada (o problema conectado)? O que acontecerá se nenhuma ação for tomada?
- Que ações futuras serão possíveis quando a meta for alcançada? Quem obterá qual benefício (o valor)?

Para soluções:

- Tenha cuidado quando um cliente apresentar diretamente uma determinada solução. Certifique-se de que você realmente entenda todos os problemas e objetivos relacionados. Verifique se a solução sugerida resolve o problema e atinge a meta.
- Considere o valor: o equilíbrio entre os benefícios esperados de uma solução, os custos (totais) para implementá-la e o risco de fracasso.

A análise de todo o cenário de problemas e metas conectados ajuda a encontrar uma solução que traga o maior valor geral. Uma solução para um problema ou meta mencionada por uma stakeholder pode trazer valor para essa pessoa, mas pode ser um risco para a empresa como um todo. Como a meta de uma stakeholder pode causar um problema para outra, pensar em termos de problemas e metas também pode ajudá-lo a identificar e resolver conflitos de requisitos.

os stakeholders geralmente contam apenas parte de toda a história quando discutem problemas e metas. Como os problemas e as metas são construções mentais, presentes apenas na mente de determinados participantes, eles são, por natureza, subjetivos, o que é difícil de revelar. Se você, como Engenheiro de Requisitos, tiver dificuldades para entender a natureza de um problema ou o valor de uma meta, os componentes subjetivos podem estar ocultos. As perguntas "por que" podem ajudar a esclarecê-las.

3.3.3 Evitar efeitos de transformação

Devido aos diferentes níveis de conhecimento, às diferentes origens culturais e sociais ou à experiência profissional, os stakeholders podem perceber as mesmas informações de maneiras diferentes. A realidade ou uma funcionalidade solicitada de um sistema será filtrada pela percepção pessoal de um indivíduo, transformada em conhecimento pessoal e, posteriormente, expressa em uma declaração mais ou menos bem formada. A declaração representa o conhecimento e, idealmente, uma grande quantidade de informações relevantes sobre a realidade. É possível distinguir dois tipos diferentes de transformação [BaGr2005] [Rupp et al.2014]:

As transformações de percepção ocorrem porque cada pessoa percebe a realidade de uma maneira diferente e cria uma imagem individual dela.

As transformações de representação ocorrem devido a uma conversão que acontece assim que uma pessoa expressa seu conhecimento em linguagem natural.

Esses processos de transformação podem levar à perda e à distorção de informações, que o Engenheiro de Requisitos precisa revelar para documentar um conjunto completo de requisitos de alta qualidade. Figura 23 descreve os processos de transformação.

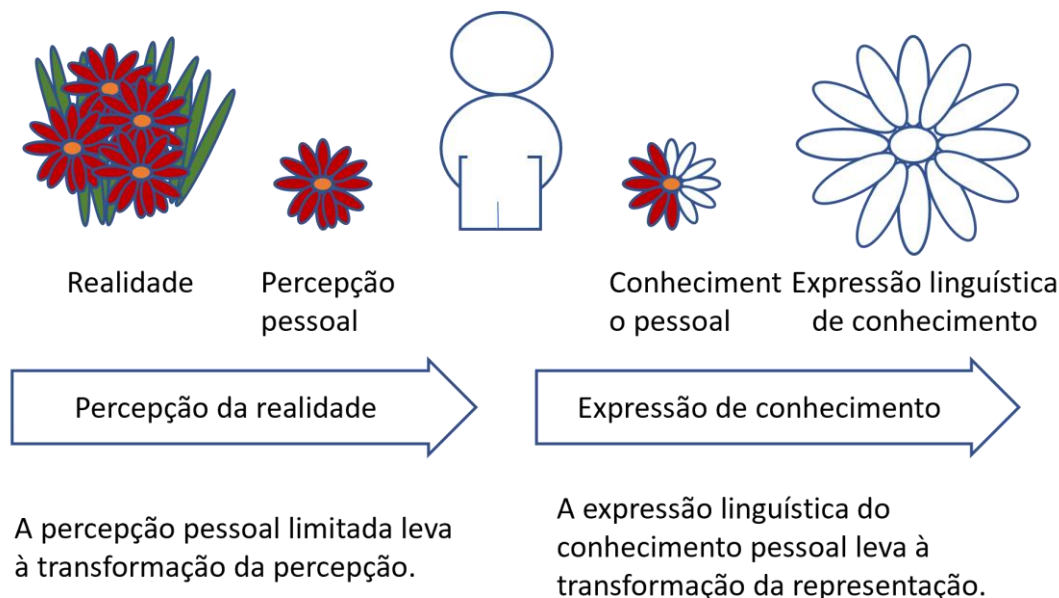


Figura 23: Processos de transformação

Para descobrir quais informações foram perdidas ou distorcidas, o Engenheiro de Requisitos precisa conhecer as diferentes categorias de processos de transformação.

A exclusão é um processo pelo qual prestamos atenção seletivamente a determinadas dimensões de nossa experiência e excluimos outras. A exclusão reduz o mundo a proporções com as quais nos sentimos capazes de lidar [BaGr2005]. A exclusão é um indicador de que o Engenheiro de Requisitos precisa obter informações ausentes.

Exemplo:

Um desenvolvedor de software pode considerar principalmente as informações lógicas (dados do cliente) de uma interface e pode não estar ciente de todas as informações físicas (material do plugue) necessárias.

A generalização é o processo pelo qual os elementos do modelo de uma pessoa se desvinculam de sua experiência original e passam a representar toda a categoria da qual a experiência é um exemplo [BaGr2005]. A generalização é um indicador de que o Engenheiro de Requisitos precisa obter as informações que faltam para decidir quais informações podem ser generalizadas e quais são, de fato, válidas apenas em circunstâncias especiais.

Exemplo:

Para uma dos stakeholders, o endereço de um cliente sempre contém o nome e o número da rua, mas em outro país os números das ruas não são comuns.

A distorção nos permite encaixar um evento ou ocorrência em uma estrutura de conhecimento pré-existente. Ela muda nossa interpretação dos eventos para se adequar ao nosso entendimento existente [GrBa2017]. As pessoas substituem partes da realidade por seu sistema de crenças pessoais. A distorção é um indicador de que o Engenheiro de Requisitos precisa obter os fatos que faltam para decidir quais das declarações representam a realidade.

Exemplo:

Uma stakeholder pode estar convencida de que dois sistemas operam usando o mesmo banco de dados, pois observa que os dados atualizados estão instantaneamente disponíveis em ambos os sistemas. Na verdade, há dois bancos de dados separados que são sincronizados muito rapidamente.

Cada uma das categorias de transformação mencionadas acima faz parte de nossa comunicação e é usada todos os dias. Em um contexto familiar, ou trabalhando com stakeholders conhecidas, os processos de transformação podem ser menos prováveis, mas, em geral, um novo sistema ou uma nova funcionalidade envolverá novos conhecimentos.

Para evitar efeitos resultantes puramente da comunicação, o Engenheiro de Requisitos deve aplicar algumas medidas básicas nas comunicações escritas e verbais [Rupp et al.2014]:

- Sempre expresse as declarações em frases completas
- Sempre use a voz ativa em suas frases
- Use os termos definidos no glossário
- Use terminologia consistente e evite sinônimos ou homônimos
- Expressar processos por meio de verbos principais

Para obter mais informações sobre comunicação na Engenharia de Requisitos, consulte a Seção 5.2.

Hint 3.3.1:

Inclua a busca de definições adequadas em suas atividades de elicitação. Em particular, os verbos de processo acordados podem ajudar a desenvolver uma comunicação mais unificada. Os modelos de termos (p. ex., com diagramas de classe UML) podem ajudar a abordar o conjunto de dados correto em um requisito.

Hint 3.3.2:

O uso de modelos de requisitos reduzirá os defeitos de transformação, pois um modelo exige que as informações básicas sejam preenchidas no requisito.

Os Engenheiros de Requisitos podem usar o conjunto de *normas*SOPHIST, conforme descrito em [Rupp et al.2014], que ajuda a analisar declarações (ou requisitos) para revelar defeitos com base nas categorias de transformação (exclusão, generalização, distorção).

Uma regra (de dezoito) abrange um ou mais efeitos transformacionais. O Engenheiro de Requisitos procurará determinadas palavras-sinal (etapa 1: Identificar) e derivará perguntas para o interessado a partir da palavra-sinal (etapa 2: Analisar). Depois que o interessado tiver respondido às perguntas, o Engenheiro de Requisitos corrigirá ou completará o requisito (etapa 3: Resolver). As regras a seguir são exemplos de efeitos transformacionais.

Hint 3.3.3:

Pode ser útil incluir uma determinada regra em seu plano de desenvolvimento pessoal e tentar implementá-la na comunicação, na documentação e nas revisões de requisitos. Após quatro semanas, você pode tentar aplicar a próxima regra.

Resolver nominalizações

As nominalizações podem obscurecer o processo (ou as etapas do processo) abordado no requisito, de modo que não fica claro como o processo deve ser realizado. As informações estão distorcidas (Distorção).

Exemplo de declaração: "O sistema da biblioteca deve oferecer arquivamento."

Etapa 1: Identificar

Archiving é uma nominalização (também a palavra sinal) do verbo "to archive" (arquivar). Arquivar dados é uma funcionalidade com etapas definidas. As etapas ainda não estão claras, nem os detalhes do processo (p. ex., quando e como o processo começa).

Etapa 2: Analisar

A pergunta a ser respondida pela stakeholder pode ser: "Quais são os detalhes do processo de 'arquivamento'?"

Etapa 3: Resolver

Neste exemplo, a stakeholder (o bibliotecário) executará as etapas do processo "escolher os dados do cliente" e "arquivar os dados do cliente escolhidos" manualmente.

"O sistema de biblioteca deve fornecer ao bibliotecário a capacidade de **escolher os dados do cliente** para arquivamento."

"Se um cliente não tiver itens emprestados no momento, o sistema da biblioteca deverá fornecer ao bibliotecário a capacidade de **arquivar os dados do cliente escolhido**."

Neste exemplo, a nominalização foi liberada para que as duas etapas do processo fossem documentadas em dois requisitos separados, mas interdependentes.

Substantivos sem ponto de referência

Se uma declaração de participante contiver substantivos sem um índice de referência, não ficará claro a quais objetos ou atores se refere. A formulação é muito geral para ser implementada (generalização).

Exemplo de declaração: "O sistema deve exibir os dados para o usuário."

Etapa 1: Identificar

Neste exemplo, não está claro quais **dados** devem ser exibidos com qual **usuário**.

Etapa 2: Analisar

A pergunta a ser respondida pela stakeholder pode ser:

- "Quem é o usuário que deve ler os dados?"
- Resposta: "O Bibliotecário."
- "Quais dados devem ser exibidos para o bibliotecário?"
- Resposta: "todos os dados estatisticamente calculados dos itens da biblioteca."

Etapa 3: Resolver

"O sistema da biblioteca deve exibir **todos os dados calculados estatisticamente dos itens da biblioteca** para o **bibliotecário**".

Hint 3.3.4:

Para obter o índice de referência, você pode revisar as funções definidas na lista de participantes ou na definição de função do sistema. Para encontrar o conjunto de dados correto, você pode revisar o modelo de dados do sistema.

Hint 3.3.5:

Às vezes, pode ser útil definir um termo mais geral para um superconjunto de dados. Por exemplo, dados de registro e detalhes de pagamento podem ser generalizados para dados do cliente, que são definidos no glossário.

Quantificadores universais

Se uma declaração de participante contiver quantificadores universais, a quantidade de objetos poderá ser muito geral ou muito específica, e deverá ser adaptada.

Exemplo de declaração: "O sistema da biblioteca deve fornecer a cada cliente a capacidade de alterar todos os dados do cliente."

Etapa 1: Identificar

Dependendo da definição dos dados do cliente, o requisito pode ser formulado corretamente. Mas, nesse caso, o conjunto de dados e a referência ao cliente são questionados.

Etapa 2: Analisar

A pergunta a ser respondida pela stakeholder pode ser:

- Então, os clientes podem alterar todos os dados já salvos no sistema da biblioteca?
Resposta: "Não, claro que não. Apenas a senha e os dados de registro do perfil dele."
- "Isso significa que cada cliente pode alterar os dados de registro de todos os clientes?"
Resposta: "Não, claro que não! Um cliente só pode alterar **seus próprios dados de registro.**"

Etapa 3: Resolver

"O sistema de bibliotecas deve fornecer a **cada** cliente a capacidade de **alterar seus dados de registro de cliente.**"

No final, **cada** cliente foi identificado como o quantificador correto. A quantidade de dados de clientes foi limitada aos **dados de registro de clientes** e a referência que um cliente poderia alterar além de **seus próprios dados de registro de clientes** foi corrigida.

Estruturas condicionais incompletas

Se um requisito contiver condições, elas podem parecer ter mais de um aspecto. Para completar o requisito, todas as condições relevantes devem ser exploradas. Caso contrário, alguns aspectos permanecerão excluídos (Exclusão).

Exemplo de declaração: "Se um item não estiver danificado e não for reservado, o sistema da biblioteca deverá fornecer ao bibliotecário a capacidade de continuar o processo de empréstimo."

Etapa 1: Identificar

A palavra de sinal "if" indica uma condição. A palavra de sinal "e" indica que a condição tem mais de uma parte.

Etapa 2: Analisar

O que acontecerá se o item for danificado e/ou reservado?

Como o sistema se comporta?

Etapa 3: Resolver

Se o item **não** estiver **danificado e for reservado**, o sistema da biblioteca exibirá uma mensagem de erro para o bibliotecário.

Se o item estiver **danificado e não for reservado**, o sistema da biblioteca deverá fornecer ao bibliotecário a capacidade de interromper o processo de empréstimo.

Se o item estiver danificado e for reservado, ...

No final, todas as combinações de condições devem ser reveladas e o Engenheiro de Requisitos deve obter o comportamento do sistema para cada caminho.

A análise detalhada dos requisitos minimizará o risco de que a compreensão da realidade possa estar errada, mas o Engenheiro de Requisitos também deve considerar o custo da atividade de elicitação. Dependendo do nível de abstração (consulte 3.3.1), do andamento do projeto e das características dos stakeholders, pode ser aceitável tolerar alguma imprecisão.

Hint 3.3.6:

Se a comunicação parecer difícil ou os mecanismos parecerem complicados, peça a pelo menos dois stakeholders diferentes que forneçam informações sobre o assunto em questão.

3.3.4 Pensando em termos de modelos

O syllabus de estudos do IREB CPRE Foundation Level [IREB2020] apresenta vários tipos de modelos (p. ex., diagramas de fluxo de dados, diagramas de atividades) para documentar os requisitos. Os modelos permitem focar em uma perspectiva específica de um sistema: dados, função, comportamento. Esse foco não se aplica apenas à documentação de requisitos, mas também pode servir como uma ferramenta de reflexão durante a elicitação de requisitos. O Engenheiro de Requisitos pode selecionar um modelo específico e se concentrar na perspectiva fornecida por esse modelo; o modelo pode ser uma ferramenta de pensamento explícita ou implícita.

Quando um modelo é usado como uma *ferramenta de pensamento explícito*, o Engenheiro de Requisitos desenvolve o modelo junto com os stakeholders. O Engenheiro de Requisitos deve ter em mente que os modelos só serão úteis nessas situações se a linguagem de modelagem for bem compreendida por todas os stakeholders envolvidas.

Por exemplo, um Engenheiro de Requisitos deseja elaborar um processo de negócio específico para ser suportado por um sistema. Essa atividade de elaboração pode ser realizada em um workshop usando diagramas de atividades. O diagrama de atividades que representa o processo de negócio é desenvolvido em conjunto com os stakeholders, por exemplo, desenhando o diagrama de atividades em um quadro branco ou em grandes flip charts. Em tal situação, a notação do diagrama de atividades serve como um kit de ferramentas para o que pode ser expresso e documentado durante o workshop. O Engenheiro de Requisitos deve prestar atenção ao conteúdo desenvolvido durante o workshop, pois os stakeholders normalmente não estão muito familiarizadas com a modelagem. Se os stakeholders não aplicarem corretamente a notação selecionada, o Engenheiro de Requisitos deverá dar suporte a elas para que criem um modelo adequado. Esses erros geralmente ocorrem porque os participantes querem expressar uma informação importante que não se encaixa no modelo.

Hint 3.3.7:

As informações que não se encaixam em uma notação de modelagem selecionada não devem ser desconsideradas. Por exemplo, se um grupo de stakeholders estiver desenvolvendo um diagrama de atividades, geralmente surgem requisitos relacionados a estruturas de dados. Esses requisitos não podem ser documentados adequadamente em um diagrama de atividades. Para não perder esses requisitos, eles devem ser salvos, por exemplo, no protocolo do workshop, para permitir uma análise posterior.

Os modelos também podem ser usados como uma *ferramenta de pensamento implícito*. Nessa situação, o Engenheiro de Requisitos usa uma linguagem de modelagem específica para estruturar seus próprios pensamentos durante a elicitación de requisitos. O modelo não se torna uma parte explícita da atividade de elicitación e não é discutido com os stakeholders envolvidas.

Em vez disso, o Engenheiro de Requisitos usa as informações obtidas durante a atividade de elicitación para desenvolver o modelo e usa o modelo criado como um ponto de reflexão para seu próprio pensamento e como uma sugestão para fazer mais perguntas ou buscar informações adicionais.

Por exemplo, um Engenheiro de Requisitos que está solicitando requisitos para um sistema on-line de venda de seguro contra acidentes quer entender os dados necessários para a aplicação. Ele/ela decide entrevistar especialistas em seguros de acidentes e analisar os formulários de solicitação em papel existentes.

Ele/ela pode começar com os formulários de inscrição e derivar um modelo de dados simples do conteúdo do formulário de inscrição. O resultado dessa análise é um modelo de dados inicial que contém entidades que não estão totalmente claras para o Engenheiro de Requisitos, que não tem relacionamentos entre as entidades e que tem um conjunto incompleto de atributos. O Engenheiro de Requisitos agora pode usar o modelo incompleto para se preparar para as entrevistas com os especialistas em seguros. Em vez de mostrar o modelo durante a entrevista, o Engenheiro de Requisitos usa o modelo como diretriz para a entrevista. As respostas dadas pelos especialistas agora podem ser mapeadas em elementos existentes e conhecidos no modelo, permitindo que o Engenheiro de Requisitos identifique mais facilmente as lacunas e faça perguntas adicionais para esclarecer essas áreas.

Hint 3.3.8:

Os modelos criados como uma ferramenta de raciocínio não devem ser confundidos com requisitos documentados. A ferramenta de raciocínio é um resultado intermediário e será usada para desenvolver requisitos detalhados e documentados em uma atividade subsequente. Portanto, é aconselhável jogar fora os modelos de ferramentas de raciocínio (ou colocá-los em um arquivo) assim que os insights do modelo forem incorporados a um documento de requisitos. Caso contrário, há o risco de que o modelo da ferramenta de raciocínio cause confusão no projeto, pois esses modelos normalmente não são mantidos e logo ficam desatualizados ou se tornam inconsistentes com os documentos de requisitos.

3.3.5 Mapeamento mental

O mapeamento mental é uma atividade na qual um conceito é visualizado em um chamado mapa mental. Em outras palavras, um mapa mental é uma ferramenta de pensamento gráfico [BuBu2005]. Ao colocar um tópico principal no centro e distribuir as ideias em ramificações, os pensamentos e as ideias podem ser classificados e reestruturados. Texto e imagens devem ser usados, assim como cores. representações "chatas" (linhas retas, apenas uma cor) devem ser evitadas para tornar a representação mais "estimulante" para o cérebro.

A ideia é baseada em estudos sobre o funcionamento do cérebro humano. Em vez de uma representação linear ou lateral, como em livros ou listas, o cérebro organiza o conhecimento de uma forma multidimensional, também chamada de "radiante". Um mapa mental é uma expressão de pensamento radiante que apoia o processo natural de pensamento [BuBu2005]. As características essenciais de um mapa mental são:

1. O objeto de atenção é cristalizado em uma **imagem central** de cada mapa mental.
2. Os principais temas do assunto "irradiam" da imagem central como **ramificações**.
3. As ramificações compreendem **imagens ou palavras-chave** e são refinadas por mais ramificações que representam **subtemas**.

Durante o mapeamento mental, é criado um mapa mental estruturado com **hierarquias, visualizações e associações** entre ramificações, como em Figura 24.

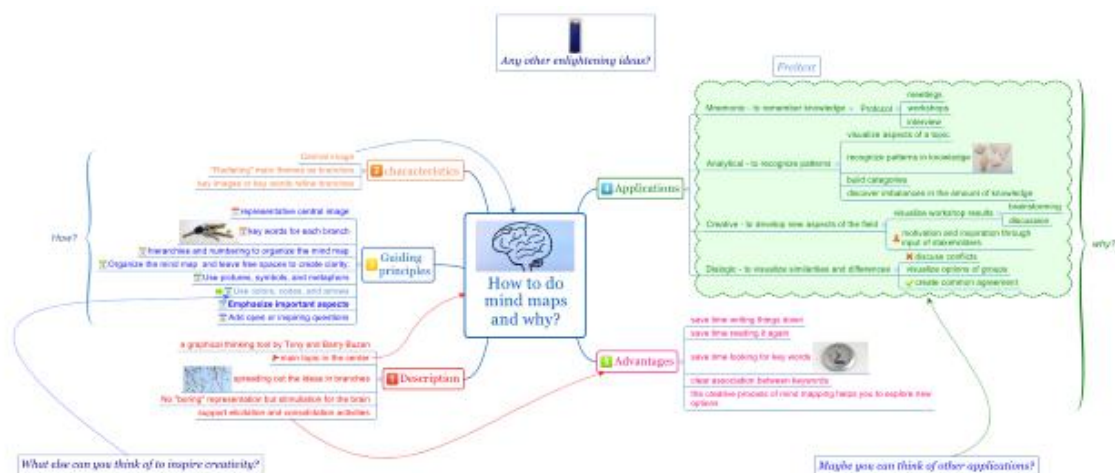


Figura 24: Exemplo de um mapa mental com o conteúdo deste subcapítulo

Há várias aplicações de um mapa mental que também podem apoiar atividades de elicitación e resolução de conflitos.

Mnemônico – para lembrar o conhecimento

- Um mapa mental pode ser usado como uma visualização de um protocolo em uma reunião ou workshop. Ele pode até ser criado simultaneamente durante a entrevista,

pois abrange apenas as palavras-chave mais importantes. Ele também cria uma visão geral do tópico principal que deve ser abordado em uma reunião.

Analítico – para reconhecer padrões

- Isso pode ser muito útil para visualizar diferentes aspectos de um tópico e reconhecer padrões no conhecimento. Dessa forma, o Engenheiro de Requisitos pode criar categorias ou descobrir desequilíbrios na quantidade de conhecimento.

Criativo – para desenvolver novos aspectos do campo

- O mapa mental também pode ser usado em workshops para visualizar os resultados de um brainstorming ou de uma discussão. O mapa mental é uma maneira rápida e motivadora de envolver os stakeholders e incentiva a inspiração por meio da contribuição de outras pessoas.

Dialógico – para visualizar semelhanças e diferenças entre conceitos e mentes

- Se os stakeholders tiverem opiniões diferentes, pode ser útil visualizar os pontos principais de diferentes indivíduos e discuti-los em um grupo. Depois disso, os stakeholders podem criar um mapa mental comum para documentar os acordos.

Para tirar o máximo proveito do mapa mental como ferramenta de raciocínio, o autor deve seguir algumas diretrizes.

Princípios orientadores do mapeamento mental:

- Use uma imagem central representativa do tópico no centro de cada mapa;
- Use palavras-chave para cada ramo ou tópico;
- Use hierarquias e numeração para organizar o mapa mental;
- Organize o mapa mental para manter a visão geral e deixe espaços livres para criar clareza;
- Use imagens, símbolos e metáforas para abordar todos os sentidos humanos;
- Use cores, códigos e setas para visualizar associações entre subtópicos e ramificações;
- Enfatize aspectos importantes por meio da variação da fonte, dos estilos de linha, da cor e do tamanho da visualização;
- Use várias dimensões na estrutura do seu mapa;
- Use cores e nuvens para enquadrar os ramos que devem estar juntos;
- Adicione perguntas abertas ou inspiradoras ao seu mapa mental.

Para obter mais orientações, consulte [BuBu2005].

O Engenheiro de Requisitos pode usar todos os tipos de ferramentas (p. ex., telas ou quadros) para criar mapas mentais. Uma variação do mapeamento mental é usar uma ferramenta baseada em computador para documentar os resultados digitalmente e apresentá-los em um projetor. Isso facilita muito a reorganização do mapa mental. De fato, com essas ferramentas, a edição simultânea do mesmo mapa mental de qualquer lugar do mundo pode ser possível, aumentando as oportunidades de cooperação.

As ramificações do mapa mental geralmente contêm termos importantes para o glossário ou modelo de termos, e as associações entre ramificações podem ser documentadas como "Dependências" entre objetos.

3.4 Descrição das técnicas de elicitação por atributos

Os Engenheiros de Requisitos devem selecionar cuidadosamente as técnicas de elicitação a serem usadas com base no contexto específico e nas necessidades da situação atual. Muitos pesquisadores propõem modelos e estruturas para mapear as técnicas de elicitação para as circunstâncias específicas. Por exemplo, Carrizo et al. fornecem uma extensa pesquisa sobre a seleção sistemática de técnicas de elicitação [CRCN2014], enquanto Tiwari e Singh propõem uma metodologia para a seleção de técnicas de elicitação de requisitos e resumem muito bem as pesquisas relacionadas a esse tópico [TiSi2017].

Nas abordagens mencionadas acima, os autores geralmente se concentram na variabilidade do problema em questão (projeto, usuários, contexto etc.), mas fornecem um mapeamento direto para técnicas de elicitação adequadas.

Com o objetivo de aprender as técnicas de elicitação de requisitos sem precisar lembrar de cor todos os prós e contras de cada uma delas em detalhes (ou seja, não como sugerido por [YoAs2015]), introduzimos *atributos de identificação* que ajudam a classificar as técnicas de elicitação. Com essa camada adicional entre o espaço do problema (como elicitar requisitos em uma determinada circunstância) e o espaço da solução (centenas de técnicas de elicitação disponíveis), o aluno pode se concentrar nos principais conceitos de solução de problemas na elicitação de requisitos: ou seja, quais atributos de identificação uma técnica de elicitação deve ter para atingir o objetivo da elicitação (consulte a Seção 1.3).

Tabela 3.4.3-1 abaixo fornece uma lista de atributos de identificação. Definimos um atributo como *identificador* se ele for uma propriedade essencial de uma técnica de elicitação.

As características de cada técnica de elicitação podem ser descritas por uma combinação desses atributos. Por exemplo, a técnica de *entrevista* é caracterizada pelos atributos de *conversação* e *questionamento*. Uma entrevista também pode ser *observacional*, no caso de o Engenheiro de Requisitos conduzir a entrevista no local do usuário final. Entretanto, a *observação* não é um atributo essencial (ou seja, identificador) das entrevistas, pois elas também podem ser realizadas por telefone ou em outros locais onde a observação não é possível.

A classificação de uma longa lista de técnicas disponíveis por atributos relevantes pode ajudar a selecionar as técnicas certas em uma determinada situação. A Tabela 3.4.3 2 fornece essa classificação para um subconjunto de técnicas.

"*Há boas práticas no contexto, mas não há práticas recomendadas*" [KaBP2002]: cada situação exige uma combinação específica de técnicas de elicitação identificadas pelos atributos relevantes para essa situação.

Ou, em outras palavras, dado o objetivo da elicitação, a situação do projeto, os stakeholders, etc., os atributos relevantes devem ser determinados e a(s) técnica(s) de elicitação que corresponde(m) exatamente a esses atributos de identificação deve(m) ser selecionada(s). Um exemplo desse mapeamento entre contexto e atributos é fornecido em Tabela 3.4.3-3.

Tabela 3.4.3-1: Atributos para classificar as técnicas de elicitação

Atributo	Descrição curta	Visando os seguintes objetivos	Adequado para as seguintes situações
Conversar	Um diálogo entre o Engenheiro de Requisitos e os stakeholders	Para entender o contexto do sistema; para obter metas e uma visão geral dos satisfatórios (Kano)	Quando os stakeholders (relevantes) estão disponíveis para troca de informações orais
Questionamento	Fazer perguntas preparadas aos stakeholders (pelo menos parcialmente) para conhecer os fatos ou a opinião deles	Para obter metas e satisfações; para verificar insatisfações; para obter a opinião dos stakeholders ou informações adicionais sobre requisitos previamente obtidos; para obter informações detalhadas; para esclarecer requisitos específicos	Se perguntas relevantes puderem ser formuladas antecipadamente; se for possível alguma forma de comunicação com os stakeholders; se o assunto for complicado
Observacional	Observar o comportamento dos stakeholders em uma situação real, geralmente operando um sistema existente ou executando tarefas específicas	Para coletar informações sobre o comportamento real dos stakeholders; para obter os atributos de performance; para analisar os requisitos de usabilidade	Se os stakeholders não puderem ser abordados diretamente ou se não puderem declarar suas necessidades e ações (suficientemente detalhadas); em caso de dúvida sobre a congruência entre a situação real e a declarada; para melhorar a compreensão das necessidades dos usuários; para melhorar a compreensão do projeto (p. ex., em preparação para outras técnicas de elicitação)

Atributo	Descrição curta	Visando os seguintes objetivos	Adequado para as seguintes situações
Provocar acordos (ou desfazê-los)	Demonstrar aspectos relevantes de uma solução para obter feedback afirmativo ou contraditório dos stakeholders	Para tornar os requisitos tangíveis para os stakeholders; para avaliar os requisitos obtidos anteriormente; para obter feedback sobre as variantes de uma solução	Se os stakeholders tiverem dificuldade para imaginar coisas; se o Engenheiro de Requisitos puder explicar ou mostrar aspectos da solução proposta para os stakeholders (ou até mesmo deixá-las usá-la); se os stakeholders tiverem dificuldade para explicar o que precisam
Basear em artefatos	Coleta e análise de produtos de trabalho existentes (p. ex., documentos, modelos, produtos ou sistemas em uso)	Derivar requisitos de produtos de trabalho existentes; obter (des)satisfações, especialmente restrições	Quando os produtos de trabalho relevantes estiverem disponíveis e acessíveis; para melhorar a compreensão do projeto e do domínio (p. ex., na preparação para outras técnicas de elicitação); se os stakeholders não estiverem diretamente disponíveis
Estimular a criatividade	Incentivar a criatividade e a inovação	Para provocar surpresas; para criar abordagens inovadoras	Se a inovação for necessária; se não houver uma direção predeterminada; quando outras abordagens falharem
Vivenciar	Experimentar o ambiente e o espaço do problema em que o sistema a ser desenvolvido será usado	Obter requisitos a partir de circunstâncias reais; entender o problema a ser resolvido pelos usuários em seu contexto de trabalho; obter empatia	Se os usuários e a usabilidade forem os principais aspectos do projeto; quando for possível acessar o ambiente onde o uso realmente ocorre

Tabela 3.4.3-2: Subconjunto de técnicas de elicitação descritas por seus atributos de identificação

Técnicas	Atributos						
	Conversacional	Questionamento	Observacional	Provocar acordos (ou desfazê-los)	Basear em artefatos	Estimular a criatividade	Vivenciar
Entrevista							
Questionário							
Workshop de requisitos (p. ex., grupo focal)							
Observação de campo							
Aprendizagem							
Pesquisa contextual							
Técnicas de criatividade (Brainstorming, ...)							
Arqueologia do sistema							
Leitura baseada em perspectiva							
Reutilização de requisitos							
Protótipos							

Técnicas	Atributos						
	Conversacional	Questionamento	Observacional	Provocar acordos (ou desfazê-los)	Basear em artefatos	Estimular a criatividade	Vivenciar
Cenários							
Storyboards							
Guia do usuário							
Teste de usabilidade							
Adivinhação de requisitos							
Elaboração de histórias de usuários (Cartão, Conversação, Confirmação)							
Estudo de diário							
Classificação de cartões							
...							

Legenda:

- I **Atributo de identificação, ou seja, é uma propriedade fundamental da técnica – sem esse atributo, é uma técnica diferente.**
- <vazio> O atributo não é uma propriedade essencial (ou seja, **nenhum atributo de identificação**), embora possa ser possível que a técnica se encaixe nesse atributo em determinadas circunstâncias.

Tabela 3.4.3-3: Objetivos de elicitação, restrições e situações de projeto que determinam a identificação dos atributos que as técnicas de elicitação adequadas devem conter:

Questão de pesquisa / Restrição / Situação do projeto	Identificar os atributos das técnicas que você pode aplicar
Se você precisar obter o conhecimento implícito dos stakeholders	Observacional
Se não for possível interromper os usuários que estão trabalhando em uma tarefa	Observacional
Se os aspectos relevantes de um sistema anterior não estiverem claros ou desempenharem um papel importante	Observacional e/ou de conversação
Se você já tiver (partes de) um sistema existente, sistemas anteriores ou sistemas semelhantes ou se já tiver produzido possíveis soluções ou modelos, por exemplo, por meio de prototipagem	Provocar acordos (ou desfazê-los)
Se o ambiente no qual o futuro sistema será usado for uma fonte importante de requisitos (ruído, pressão de tempo, condições físicas etc.)	Vivenciar
Se você precisar obter requisitos importantes de usabilidade	Vivenciar
Se o seu projeto estiver operando em território desconhecido	Provocar (des) acordo Estimular a criatividade
Se nenhuma stakeholder puder especificar os requisitos em um nível específico de detalhes	Provocar acordos (ou desfazê-los)
Se você desenvolver um sistema complexo que será usado por usuários casuais	Provocar acordos (ou desfazê-los)
Se você identificou fontes de requisitos não humanos	Basear em artefatos
Se você quiser descobrir onde estão as falhas do processo	Com base em artefatos, observação e experiência
Se você quiser descobrir como e com quais produtos de trabalho seus futuros usuários realmente trabalham	Com base em experiências e artefatos
Se você precisar de material para se preparar para atividades de elicitação com os stakeholders	Basear em artefatos
Se você precisar de um catalisador para que os stakeholders apresentem requisitos que, de outra forma, não teriam lembrado ou conhecido	Basear em artefatos

Questão de pesquisa / Restrição / Situação do projeto	Identificar os atributos das técnicas que você pode aplicar
Se você desenvolver um sistema novo e inovador que precise oferecer novos recursos ou novas formas de interação (ou seja, se estiver procurando por pessoas que causem atratividade, de acordo com o modelo Kano)	Estimular a criatividade
Se a solução para o seu problema não for óbvia e você e sua equipe precisarem criar uma nova abordagem	Estimular a criatividade
Se o Engenheiro de Requisitos precisar especificar a solução em vez de apenas ouvir os stakeholders e escrever os requisitos, ele será informado	Vivenciar
Se as ERs continuarem a manter a solução depois de concluída a implementação inicial (ou seja, o aumento e a fundamentação do conhecimento adquirido pela ER durante suas atividades de elicitação são, por si só, um ativo valioso para o patrocinador)	Vivenciar
Se, em configurações específicas de projeto, os stakeholders relevantes raramente estiverem disponíveis e, portanto, for mais fácil e/ou mais rápido estudar o domínio sob investigação e o Engenheiro de Requisitos criar os próprios requisitos (e apenas validá-los com os stakeholders)	Vivenciar
Se você precisar esclarecer os requisitos relativos à situação atual, às necessidades e às possíveis soluções	Conversar
Se você quiser ter certeza de que fala a mesma língua com os stakeholders, especialmente em domínios especializados	Conversar
Se o grau de interação do usuário for alto	Experiência Provocando (des) acordo
Se o grau de integração técnica for alto	Basear em artefatos
Se o nível de inovação (=> Kano) for alto	Estimular a criatividade Provocar (des) acordo

4 Resolver conflitos

Durante a elicitación, o Engenheiro de Requisitos reúne uma ampla coleção de requisitos, geralmente de diferentes fontes (consulte o Capítulo 2), com diferentes técnicas e em diferentes níveis de abstração e detalhe (consulte o Capítulo 3). As técnicas de elicitación por si só não garantem que essa coleção como um todo atenda a todos os critérios de qualidade dos requisitos (consulte [IREB2020]). Se os critérios de qualidade não forem atendidos, podem ser necessárias atividades adicionais de elicitación para melhorar a qualidade.

São necessárias atividades adicionais para transformar essa coleção em um conjunto único e consistente de requisitos que capture a essência do sistema. Muitas vezes, verifica-se que alguns requisitos são conflitantes: inconsistentes, incompatíveis, contraditórios.

Normalmente, isso é causado por discordância entre determinados stakeholders. Como consequência, o "acordo" é um critério de qualidade muito importante que deve ser sempre verificado: todas os stakeholders precisam entender e concordar com todos os requisitos que são relevantes para elas. Se alguns stakeholders não concordarem, essa situação deve ser reconhecida como um conflito de requisitos a ser resolvido adequadamente.

Um dos objetivos da Engenharia de Requisitos é chegar a um consenso entre os stakeholders sobre os requisitos. Uma tarefa importante nesse sentido é o tratamento de conflitos de requisitos.

A resolução de conflitos no sentido amplo consiste em quatro tarefas:

- Identificar conflitos
- Análisar conflitos
- Resolver conflitos
- Documentar a resolução de conflitos

A identificação e análise de conflitos é uma atividade contínua na Engenharia de Requisitos e é um pré-requisito para a resolução de qualquer conflito. Uma vez identificado um conflito de requisitos, o Engenheiro de Requisitos deve iniciar as atividades de resolução de conflitos para selecionar uma técnica de resolução adequada e documentar seu resultado.

Em nossa vida cotidiana, estamos frequentemente envolvidos em conflitos. Como, em geral, não é agradável lidar com eles, uma estratégia comum é simplesmente fugir deles. Como consequência, as pessoas não conversam entre si, procuram áreas de trabalho diferentes ou até mesmo mudam de projeto ou trabalho para não se envolverem mais com o conflito.

Lidar com conflitos de requisitos pode ser estressante e demorado, principalmente se houver questões pessoais. Mas é essencial considerar os seguintes aspectos:

1. A solução de problemas pessoais não faz parte da descrição do cargo e deve ser escalada por meio de diferentes atividades de gerenciamento. Veja a seção 4.2 – Análise de conflitos.

2. Escapar dos conflitos de requisitos não é uma opção, pois os conflitos de requisitos não resolvidos resultam em documentos de requisitos de baixa qualidade e em stakeholders frustradas.

A conscientização constante dos conflitos e a aplicação regular de análises [IREB2020] ajudarão a descobrir indicadores de conflito e a coletar dados para a resolução de conflitos.

4.1 Identificar conflitos

Os conflitos em geral são um tema das ciências sociais e, normalmente, são chamados de "conflito social" para indicar que um conflito surge entre pessoas. Um conflito social pode ser definido da seguinte forma: [Glasl2004] ¹⁰ define um conflito social como "uma interação entre agentes (indivíduos, grupos, organizações etc.), em que pelo menos um agente percebe incompatibilidades entre seu pensamento/ideias/percepções e/ou sentimentos e/ou vontade e o do outro agente (ou agentes), e se sente restringido pela ação do outro".

Um conflito de requisitos pode ser interpretado como um tipo especial de conflito social e é definido da seguinte forma: "Um conflito na Engenharia de Requisitos (conflito de requisitos) é uma incompatibilidade de requisitos, com base em uma percepção contraditória de dois ou mais stakeholders." [Rupp et al.2014] Há vários indicadores pelos quais os conflitos podem ser detectados. Os indicadores podem ser observados na comunicação e na documentação.

Os indicadores comumente encontrados na comunicação são:

- **Negação:** Um stakeholder nega suas próprias declarações ou declarações de outros stakeholders ou não aceita acordos ou declarações feitas no passado.
- **Indiferença:** Um stakeholder não (quer) contribuir em uma discussão ou aprova sem questionamento crítico.
- **Pedantismo:** Uma stakeholder sempre encontra falhas em todas as proposições e sempre encontra um pequeno problema que usa para argumentar contra todas as sugestões.
- **Questões de detalhes:** Um participante usa sua posição ou seu conhecimento para questionar todas as declarações de forma muito crítica. Pode parecer que o interessado quer garantir que os requisitos importantes não sejam esquecidos, mas, na verdade, isso atrasa o processo de elicitação.
- **Interpretação incorreta:** Um participante interpreta erroneamente os fatos de propósito para confundir ou retardar o processo de elicitação.
- **Ocultação:** Um participante oculta as informações de forma consciente ou subconsciente e só as compartilha quando solicitado.
- **Delegação:** Uma stakeholder se compromete com declarações apenas vagamente com a exigência de que outros as declarem com mais detalhes.

¹⁰ Citação traduzida do alemão com base em [Glasl1999]

Os indicadores comumente encontrados na documentação são (com exemplos):

- **Declarações contraditórias dos participantes:** Durante os workshops, os participantes concordam com um requisito que não é consistente com um requisito derivado de um protocolo de entrevista de uma atividade de elicitação anterior.
- **Resultados conflitantes da análise de documentos ou sistemas:** A especificação da interface de um sistema contém um endereço temporário para um cliente, mas no sistema só é possível inserir um segundo endereço principal.
- **Requisitos inconsistentes em detalhes:** O sistema identifica conjuntos de dados de clientes duplicados por meio de nome, dia de nascimento e endereço, mas há alguns clientes que não têm endereço.
- **Uso inconsistente de termos na especificação:** os stakeholders usam os termos cliente, usuário e funcionário com significados diferentes, mas não aplicam as definições do glossário.

A maioria dos conflitos tende a ser oculta e só pode ser detectada por meio do monitoramento cuidadoso desses indicadores. Se um dos indicadores ocorrer, isso não significa que haja um conflito de requisitos. No entanto, o Engenheiro de Requisitos deve estar sempre atento. Durante a maior parte das atividades de elicitação de requisitos, ele/ela está incentivando os stakeholders a expor suas posições claramente, revelando assim problemas inesperados ou conflitos existentes.

4.2 Análisar conflitos

Se não houver absolutamente nenhum sinal de conflito de requisitos, o Engenheiro de Requisitos deve desconfiar e planejar algumas revisões. Assim que houver suspeita de um conflito, ele/ela poderá incluir a coleta de dados para um conflito nas atividades de elicitação de requisitos.

Depois que um conflito é identificado, o Engenheiro de Requisitos precisa esclarecer se o conflito identificado é ou não um conflito de *requisitos*. Essa distinção é importante, pois a resolução de um conflito de requisitos é a principal responsabilidade do Engenheiro de Requisitos, enquanto outros conflitos precisam ser resolvidos por outros participantes (p. ex., um gerente de projeto).

A análise das características de um conflito de requisitos ajuda o Engenheiro de Requisitos a entender sua natureza. Durante a análise de conflitos, há vários indicadores de conflitos. Pode ser útil coletar indicadores primeiro e revisá-los depois. Com mais informações, é muito mais fácil encontrar uma solução adequada para o conflito.

As seguintes características [Rupp et al.2014] de um conflito pode ajudar a entender sua natureza e a encontrar uma solução adequada.

4.2.1 As características de um conflito de requisitos

O *tipo de conflito* determina suas características e o grau de envolvimento pessoal dos stakeholders. É um dos indicadores mais importantes com base no qual as técnicas de resolução de conflitos devem ser excluídas ou aplicadas (consulte a Seção 0 – Resolução de conflitos). Em alguns casos, pode ser difícil determinar o tipo de conflito. Se esse for o caso, vários tipos de conflito devem ser considerados.

O *tema do assunto* é o problema por trás do conflito. Descobrir qual é a verdadeira questão por trás da discussão pode ser muito difícil, dependendo do tipo de conflito e de sua história. Mas também é muito valioso para uma resolução adequada. O Engenheiro de Requisitos deve permanecer neutro com relação às opções disponíveis e pode facilitar ativamente a análise, espelhando as declarações para os participantes.

Os *requisitos afetados* são as declarações representativas do conflito. Eles podem ser usados para análise e visualização de detalhes. Uma vez encontrada uma solução para o conflito, a documentação dos requisitos relevantes deve ser fácil.

Hint 4.2.1:

O Engenheiro de Requisitos pode pedir às stakeholders que formulem suas preocupações como requisitos e permitir que outras partes em conflito confirmem ou rejeitem as declarações. Isso favorecerá uma comunicação precisa e reflexiva.

Os *stakeholders envolvidas* podem ser os autores ou outros responsáveis, de alguma forma, pelos requisitos afetados. Eles são as fontes das informações para análise e podem, eles próprios, fazer parte do conflito. Às vezes, pode ser útil envolver mais stakeholders para fornecer conhecimento especializado e moderar ou até mesmo resolver o conflito entre as partes por meio de sua autoridade.

As *opiniões* são as declarações dos stakeholders ou um resumo verbal dos conceitos que elas têm em mente. Como Engenheiro de Requisitos, é possível reformular (ou deixar que os stakeholders reformulem) na frente das outras partes envolvidas. Aspectos vagos ou pouco claros podem ser explicados para ajudar as partes conflitantes a entenderem a questão subjacente.

A *causa* é o motivo pelo qual os stakeholders não podem continuar trabalhando de forma independente. Quando isso for esclarecido, poderá indicar o caminho para uma técnica de resolução adequada ou até mesmo a própria resolução.

O *histórico* do conflito pode ajudar as novas partes a entenderem as abordagens ou argumentos anteriores contra requisitos ou opções afetados. É provável que nem todas as partes tenham o mesmo nível de conhecimento, o que pode, de fato, ser o principal motivo do conflito.

As *consequências* são os custos associados aos requisitos relevantes implementados. Esses custos também podem não ser claros para algumas das partes e devem ser estimados a fim de contribuir para a resolução do conflito.

Os *riscos resultantes* podem ser muito importantes para decidir quando (ou seja, em que fase da análise) resolver o conflito.

As *restrições do projeto* são de natureza pessoal, organizacional, de conteúdo específico ou de domínio específico. Elas estão relacionadas ao tipo de conflito e influenciam a escolha de uma técnica adequada de resolução de conflitos. Em uma situação específica do projeto, por exemplo, pode não haver tempo suficiente (restrição organizacional do projeto) para resolver o conflito com o acordo técnico (consulte a Seção 4.3 – Resolução de conflitos).

4.2.2 Os tipos de conflito de Moore

O tipo de conflito é importante para decidir se um determinado conflito é um conflito de requisitos ou não. [IREB2020] estendeu os cinco tipos de conflito originais da Moore [Moor2014] pelo conflito de assunto, já que esse conflito ocorre com frequência em ER:

- Conflito de temas
- Conflito de dados
- Conflito de interesses
- Conflito de valor
- Conflito estrutural
- Conflito de relacionamento

A maioria dos conflitos de requisitos pode ser categorizada como conflitos de dados, conflitos de interesses ou conflitos de valores. Os conflitos de assunto, se houver, geralmente são revelados nas fases iniciais do projeto. Os conflitos estruturais e de relacionamento geralmente não estão relacionados aos requisitos e, portanto, devem ser resolvidos por outros participantes. Entretanto, a maioria dos conflitos apresenta características de mais de um tipo, pois diferentes causas interagem. Portanto, os Engenheiros de Requisitos devem prestar atenção a todos os tipos de conflito, mesmo que a solução não seja de sua responsabilidade.

4.2.2.1 Conflito de temas

Um conflito de conteúdo ocorre quando as partes em conflito realmente têm diferentes necessidades factuais, causadas principalmente pelo uso pretendido do sistema em ambientes diferentes. Às vezes, a existência de um conflito de assunto está clara para determinados stakeholders desde o início de um projeto, mas é ignorada para evitar atrasos no início do projeto. Mais frequentemente, ele é detectado quando o círculo de stakeholders é ampliado após a fase inicial do projeto.

Exemplos de conflitos de assunto:

- Um sistema que deve ser usado em diferentes países, cada um com sua própria legislação.
- os stakeholders de um departamento de pessoal solicitam dados consolidados no final do dia de trabalho, enquanto os stakeholders da central de serviços precisam de dados em tempo real.

Pode ser difícil resolver tal conflito porque os fatos subjacentes não podem ser alterados.

Estratégias de prevenção e resolução:

- Analise e documente os fatos em detalhes.
- os stakeholders devem concordar com a natureza exata do conflito.
- os stakeholders devem indicar seus riscos, custos ou danos, caso o conflito não seja resolvido a contento.

4.2.2.2 Conflito de dados

Um conflito de dados se baseia na falta ou na distribuição desigual de conhecimento, ou em uma interpretação diferente dos dados disponíveis para as partes em conflito.

Exemplos de conflitos de dados:

- os stakeholders podem argumentar sobre a existência ou a interpretação de uma regra de negócio descrita em um requisito.
- os stakeholders divergem quanto ao motivo de um requisito.
- os stakeholders têm uma compreensão diferente dos termos e de suas definições contidas nos requisitos.

Se um conflito de dados for o motivo por trás de uma discussão, pode-se observar que a comunicação entre as partes em conflito se concentra nas "informações factuais" do modelo de "quatro lados" de Schulz von Thun [Schulz von Thun 1981]. os stakeholders trocam informações e compartilham fatos e números importantes.

Estratégias de prevenção e resolução:

- Os dados devem ser fornecidos às stakeholders por meio da troca ou coleta de informações adicionais.
- As informações relevantes para o conflito de dados específico devem ser identificadas.
- os stakeholders devem chegar a um acordo sobre um processo comum de coleta de dados com critérios de avaliação.
- os stakeholders devem chegar a um acordo sobre os especialistas que fornecerão informações relevantes.

4.2.2.3 Conflito de interesses

Um conflito de interesses se baseia nas diferentes motivações das partes em conflito. As motivações podem ser formadas por metas pessoais, metas relacionadas a um grupo ou metas relacionadas a uma função. Como um conflito de interesses não se baseia na posse de informações (como o conflito de dados), é importante entender as preocupações e as necessidades dos stakeholders para resolver esse tipo de conflito. No caso de interesses pessoais, os stakeholders geralmente não revelam seus verdadeiros motivos, mas encontram argumentos artificiais.

Exemplos de conflitos de interesses:

- Uma stakeholder do departamento de segurança pode solicitar padrões mais altos de criptografia que exigem mais tempo, enquanto o usuário do sistema enfatiza o desempenho do sistema em seu trabalho diário.
- Uma stakeholder quer que seu departamento seja responsável pela implementação de uma função por causa do prestígio de seus funcionários, enquanto o arquiteto do sistema defende outro componente para melhorar a estabilidade da arquitetura do sistema.
- Uma stakeholder exige que uma função para seu trabalho seja implementada na próxima versão, mas o patrocinador do sistema acredita que outras funções são mais importantes para a maioria dos usuários.

Se um conflito de interesses for o motivo de uma discussão, é possível observar que a comunicação entre as partes em conflito se concentra no "apelo" do modelo de "quatro lados" de Schulz von Thun [Schulz von Thun 1981]. As partes em conflito tentam convencer os outros a seguir seus argumentos e a entender as necessidades da função ou do grupo.

Estratégias de prevenção e resolução:

- Os interesses e as consequências devem ser revelados e considerados em um contexto diferente ou por uma instância diferente que possa ser mais objetiva.
- A partir do conjunto de interesses, os fatos relevantes para o conflito podem ser extraídos para que uma resolução baseada em fatos possa ser apoiada.

4.2.2.4 Conflito de valor

Um conflito de valores se baseia em valores e princípios diferentes. Está relacionado ao conflito de interesses, mas é mais individual e envolve perspectivas globais ou de longo prazo. Se uma pessoa mudar de função, os interesses podem mudar, mas os valores são mais estáveis e raramente mudam no curto prazo.

Exemplos de conflitos de valores:

- os stakeholders podem evitar produtos com muito plástico ou que não sejam recicláveis, enquanto outras preferem produtos de baixo custo. O valor da proteção do meio ambiente se opõe ao valor da produção barata.
- os stakeholders podem considerar a representação gráfica no software menos importante do que os aplicativos baseados em comandos.
- os stakeholders podem considerar a discriminação de preços (= um fornecedor vende o mesmo produto a preços diferentes para clientes diferentes) injusta para os usuários de um site de e-business.

Se um conflito de valores for o motivo de uma discussão, pode-se observar que a comunicação entre as partes em conflito se concentra na "autorrevelação" do modelo dos "quatro lados" de Schulz von Thun [Schulz von Thun 1981].

As partes em conflito enfatizam porque seus argumentos são importantes do ponto de vista delas e apresentam vários argumentos que revelam seus valores e princípios internos. Eles tendem a insistir em seus argumentos e parecem ter experiência em defender a respectiva questão.

Estratégias de prevenção e resolução:

- os stakeholders devem permitir que as partes em conflito concordem ou discordem dos argumentos, sem julgamento, mas com tolerância.
- As partes em conflito devem se concentrar em seus pontos comuns, onde seus valores estão alinhados.
- As partes em conflito devem se concentrar em um objetivo global comum e no contexto mais amplo, e não em suas diferenças e nos detalhes.

4.2.2.5 Conflito estrutural

Um conflito estrutural se baseia na desigualdade de poder, na competição por recursos limitados e nas dependências estruturais que influenciam as partes em conflito. O desequilíbrio percebido causa problemas na comunicação, na obtenção de requisitos e na tomada de decisões. Outro motivo para um conflito estrutural pode ser restrições rígidas de recursos ou dependências de produtos de trabalho a serem entregues por outras partes.

Exemplos de conflitos estruturais:

- os stakeholders podem suprimir requisitos porque acreditam que uma parte em conflito com maior autoridade pode argumentar contra eles.
- os stakeholders com maior influência na organização podem tentar alterar a prioridade dos requisitos.
- os stakeholders que desejam mais transparência formulam um requisito de acesso a um aplicativo específico para que não possam ser surpreendidas por outro departamento que sempre entrega informações tarde demais.

Se um conflito estrutural for o motivo por trás de uma discussão, pode-se observar que a comunicação entre as partes em conflito se concentra no "relacionamento" do modelo dos "quatro lados" de Schulz von Thun [Schulz von Thun 1981]. As partes em conflito podem usar a discussão sobre os requisitos para alterar ou preservar o status quo. Dependendo do ponto de vista, as partes em conflito enfatizam sua discordância ou desejo de mudar (para a parte detentora do poder: preservar) a estrutura ou os relacionamentos atuais.

Estratégias de prevenção e resolução:

- As responsabilidades (p. ex., para fornecer requisitos) e os recursos devem ser redistribuídos por uma parte em uma posição sênior.
- As dependências devem ser dissolvidas por uma parte em uma posição superior.
- Outro processo de decisão deve ser implementado.
- A pressão externa deve ser redefinida de forma que não influencie o trabalho ou os requisitos da parte em conflito.

Como a maioria das estratégias de resolução descritas aqui pode não estar dentro das responsabilidades do Engenheiro de Requisitos, ele só pode escalar conflitos estruturais e deixar que outros stakeholders intervenham.

4.2.2.6 Conflito de relacionamento

Um conflito de relacionamento pode se basear em experiências negativas com as partes em conflito no passado ou em outras experiências negativas de situações comparáveis com pessoas semelhantes. Muitas vezes, ele está ligado a emoções e falhas de comunicação, o que o torna muito mais difícil de resolver.

Exemplos de conflitos de relacionamento:

- os stakeholders atacam verbalmente a outra parte em conflito sem um bom motivo ou de forma altamente emocional. Os fatos e a discussão justa parecem ser irrelevantes.
- Em uma discussão, uma stakeholder do departamento de software não aceita um requisito discutido com o arquiteto do sistema porque percebe que os arquitetos não têm ideia da realidade do trabalho dos programadores.
- Um participante não aceita o convite do Engenheiro de Requisitos para o workshop porque seu arqui-inimigo participará da reunião.

Se um conflito de relacionamento for o motivo de uma discussão, pode-se observar que a comunicação entre as partes em conflito se concentra no "relacionamento" do modelo dos "quatro lados" de Schulz von Thun [Schulz von Thun 1981]. As partes em conflito podem usar a discussão de requisitos para expressar sua discordância com o comportamento da outra parte em conflito. Os fatos e a discussão justa parecem não ter importância para as partes em conflito.

Estratégias de prevenção e resolução:

- os stakeholders devem concordar com as regras e os procedimentos da reunião quando surgirem emoções.
- O Engenheiro de Requisitos deve obter requisitos das partes envolvidas no conflito separadamente e sem revelar as fontes dos requisitos. Dessa forma, a atividade de elicitação permanece mais objetiva.
- Impedir o comportamento negativo das partes em conflito.
- A revisão e o acordo devem ser realizados sem o envolvimento direto das partes em conflito ou por um terceiro neutro.

Como a maioria das estratégias de resolução descritas aqui não é de responsabilidade do Engenheiro de Requisitos, ele só pode escalar os conflitos de relacionamento e deixar que outros stakeholders intervenham.

4.3 Resolver conflitos

Um pré-requisito para a seleção de uma técnica de resolução adequada é um entendimento profundo da natureza do conflito de requisitos.

Com base nessa análise e nas restrições do projeto, o Engenheiro de Requisitos pode selecionar uma técnica de resolução adequada. Todas as técnicas descritas nesta seção são estruturadas de forma semelhante às técnicas de elicitación no Capítulo 3.

As seguintes técnicas gerais de resolução podem ser distinguidas (consulte [IREB2020]):

- Acordo
- Compromisso
- Votação
- Análise de alternativas
- Imposição

4.3.1 Acordo

O que é isso?

O acordo é o resultado de uma discussão na qual um conjunto de requisitos existente ou uma resolução de conflito é selecionado sem alterações entre várias opções disponíveis. Para chegar a essa resolução, é essencial que haja tempo suficiente para entender completamente as posições de todos os stakeholders e convencê-las de que a opção selecionada também é a preferida delas.

Papel dos participantes

O moderador pode assumir a função de lembrar às stakeholders que devem conversar de forma construtiva e eficiente. Ele deve ser neutro em relação às opções e ser aceito nessa função pelo grupo. Os stakeholders envolvidas devem se concentrar nos fatos e permitir que outros façam perguntas.

Preparação

Para um grupo muito grande, é aconselhável nomear um representante por grupo. Os stakeholders devem ser convidadas, informadas sobre a agenda e sua contribuição esperada para o processo.

Aplicação

- O moderador definirá o tópico e estabelecerá uma agenda com um cronograma.
- O moderador deve apresentar os dados sobre o conflito e explicar as possíveis técnicas de resolução.
- Os stakeholders devem apresentar seus argumentos sem interrupções, após o que as outras partes podem fazer perguntas.

O moderador acompanha a discussão e sugere uma resolução adequada. Se isso parecer impossível devido ao tipo de assunto, ao tipo de conflito ou ao surgimento de novos conflitos, o moderador deve encerrar a discussão e sugerir outra técnica de resolução.

Processamento de resultados

Um participante pode resumir o resultado para todos os participantes. O moderador deve facilitar um acordo informal de todas as partes imediatamente. Se as opções forem muito complexas, o resultado e os requisitos afetados serão concluídos posteriormente e revisados individualmente.

Produtos de trabalho típicos

Argumentos a favor e contra as diferentes alternativas; a solução selecionada (incluindo os requisitos acordados pelas partes em conflito).

Chances

Com o acordo, as partes em conflito têm a oportunidade de entender melhor umas às outras e suas necessidades. Um resultado positivo pode proporcionar motivação adicional para o grupo e o resultado tem uma boa chance de ser duradouro.

Desafios

Os desafios estão em apresentar a essência do conflito para que todos saibam o que será discutido. Para o moderador, pode ser um desafio evitar que a cultura da discussão influencie negativamente o resultado e que não se chegue a um acordo, embora um acordo pareça provável. É um desafio manter os stakeholders em um nível factual, dividir igualmente o tempo de discussão e manter-se dentro do prazo.

Variáveis

Não aplicável.

4.3.2 Compromisso

O que é isso?

O compromisso é o resultado de uma discussão em que aspectos de um conjunto existente de requisitos ou resoluções de conflitos, bem como novos aspectos, são combinados para criar novas opções. Para chegar a essa resolução, é essencial que haja tempo suficiente para chegar a um bom entendimento das respectivas posições e de todos os aspectos do problema, a fim de negociar uma resolução que atenda às necessidades de todos.

Papel dos participantes

O moderador pode assumir a função de lembrar aos participantes que devem seguir as regras acordadas e continuar a discussão de forma construtiva e eficiente. Ele deve ser neutro em relação às opções e ser aceito nessa função pelo grupo. Os stakeholders envolvidas devem se concentrar nos fatos e permitir que outros façam perguntas para entender completamente todos os aspectos importantes para as partes em conflito.

Preparação

O acordo deveria ter sido considerado e descartado como uma opção imediata. Observe que, por definição, um acordo não é possível em um conflito de assunto. Nesse tipo de conflito, o compromisso é a primeira opção a ser considerada.

Para um grupo muito grande, é aconselhável nomear um representante por grupo. Os stakeholders devem ser convidadas, informadas sobre a agenda e sua contribuição esperada para o processo.

Aplicação

- O moderador definirá o tópico e estabelecerá uma agenda com um cronograma.
- O moderador deve apresentar os dados sobre o conflito e explicar as possíveis técnicas de resolução.
- Os stakeholders devem apresentar seus argumentos sem interrupções, após o que as outras partes podem fazer perguntas.
- O moderador deve aconselhar as partes em conflito a negociar primeiro seus aspectos mais importantes.
- Assim que as partes em conflito chegarem a um acordo sobre um aspecto específico, o moderador deve garantir que esse acordo seja documentado.
- O moderador acompanha a discussão e incentiva a aplicação das regras de comunicação.

Processamento de resultados

Um participante pode resumir o resultado para todos os participantes. O moderador deve facilitar um acordo informal de todas as partes imediatamente. Se as opções forem muito complexas, o resultado (e os requisitos afetados) poderá ser concluído posteriormente e revisado individualmente.

Produtos de trabalho típicos

Argumentação sobre o fato de um acordo não ser uma técnica de resolução adequada. Diferentes aspectos discutidos, incluindo os requisitos com os quais as partes em conflito concordaram.

Chances

Com um compromisso, há uma oportunidade para que as partes em conflito entendam melhor umas às outras e suas exigências, sem a necessidade de um acordo completo.

Novos aspectos podem ser integrados, e mais stakeholders contribuem para a resolução criada. Um resultado positivo pode proporcionar motivação adicional para o grupo e o resultado tem uma boa chance de ser duradouro.

Desafios

Os desafios estão em apresentar a essência do conflito para que todos saibam o que deve ser discutido. Para o moderador, pode ser um desafio impedir que a cultura da discussão influencie negativamente o resultado, como, por exemplo, quando a parte dominante obtém um resultado melhor para si mesma, mas não para todo o projeto. O moderador deve, portanto, garantir que os stakeholders permaneçam no nível factual, tenham uma parcela igual do tempo de discussão e respeitem o prazo especificado.

Variáveis

Não aplicável.

4.3.3 Votação

O que é isso?

A votação é o resultado de uma seleção de um conjunto existente de requisitos. É fundamental que as opções possíveis, ou o conjunto de requisitos, sejam bem compreendidos pelos tomadores de decisão. Para que as dependências ou o desequilíbrio de poder não influenciem o resultado, a votação de uma opção deve ser secreta.

Papel dos participantes

Os tomadores de decisão devem entender as consequências de suas escolhas e as opções disponíveis. Um moderador pode conduzi-los pelo processo e comunicar as etapas do procedimento de votação, as opções disponíveis e o resultado da votação.

Preparação

os stakeholders com autoridade para decidir sobre o assunto devem transferir o poder de decisão para os stakeholders selecionadas e concordar em aceitar o resultado. Para deixar claro em que os stakeholders devem votar, devem ser criadas propostas para cada opção. Um comitê de votação neutro deve ser selecionado e folhas de votação com as opções apropriadas devem ser preparadas.

Aplicação

- Uma vez acordada a solução de um conflito por meio da técnica de resolução "Votação", cada parte envolvida no conflito deve preparar uma proposta que descreva sua posição e as possíveis consequências da solução proposta para os tomadores de decisão. Depois que as propostas forem preparadas e distribuídas, a reunião de votação poderá ser realizada.
- Na reunião de votação, o moderador deve explicar o procedimento de votação e as respectivas posições aos tomadores de decisão.

- Opcionalmente, cada parte pode apresentar sua proposta e responder às perguntas dos tomadores de decisão.
- As folhas de votação são coletadas e avaliadas.

Processamento de resultados

Os resultados da votação devem ser documentados.

Produtos de trabalho típicos

Propostas para as opções sugeridas, incluindo prós e contras, bem como as folhas de votação preenchidas.

Chances

A votação pode acomodar o envolvimento de muitos stakeholders, o que, por sua vez, pode gerar maior satisfação dos stakeholders.

Desafios

Pode ser difícil explicar opções complexas de forma que elas possam ser compreendidas por todos os tomadores de decisão. As tendências políticas podem influenciar a decisão.

Variáveis

Para decisões simples, a votação pode ser realizada espontaneamente, sem qualquer preparação, e ser concluída rapidamente. Várias ferramentas apoiam a tomada de decisões e o envolvimento dos stakeholders localizadas em qualquer lugar do mundo. Essas ferramentas também podem oferecer suporte a um procedimento de votação que não seja síncrono.

4.3.4 Análise de alternativas

O que é isso?

A definição de variantes é o resultado da integração dos requisitos relevantes e diferentes em uma solução na qual o sistema pode ser configurado para suportar qualquer uma das opções. O configurador ou o usuário do sistema pode então selecionar um recurso que represente o conjunto desejado de requisitos durante a configuração ou no tempo de execução. Muitas vezes, há um conjunto adicional de requisitos que precisam ser implementados para dar suporte à alternância entre as diferentes opções. Como todas as opções, bem como a alternância entre elas, precisam ser obtidas e mantidas, a definição de variantes pode se tornar dispendiosa e resultar em sistemas complicados com recursos raramente usados.

Hint 4.3.1:

O Engenheiro de Requisitos precisa avaliar se a definição de variantes não é, de fato, uma fuga de um processo adequado de resolução de conflitos e se vale a pena o esforço adicional.

A configuração do sistema deve ser implementada de modo que as variantes possam ser usadas de forma independente e o conflito seja realmente resolvido. A definição de variantes geralmente é a única solução viável para um conflito de assunto.

Papel dos participantes

O Engenheiro de Requisitos deve destacar as diferenças entre as variantes para que fique claro porque não foi possível chegar a uma única opção de solução.

Preparação

Técnicas como acordo, compromisso e votação deveriam ter sido consideradas antes. Todas as partes devem concordar que a criação de uma variante que cause esforço adicional no futuro é a única solução possível.

Aplicação

- Poderiam ser organizadas reuniões separadas com cada parte para obter os requisitos afetados para as diferentes opções, de modo que não interfiram entre si.
- Depois que as opções separadas forem especificadas, as partes deverão revisar as opções para maior alinhamento. Se houver um alinhamento maior, as partes comuns das várias opções devem ser combinadas.
- Quando não for possível um alinhamento adicional, os requisitos para a troca entre as variantes devem ser especificados.

Processamento de resultados

Os resultados da definição de variantes devem ser documentados.

Produtos de trabalho típicos

O ideal é que os produtos de trabalho sejam os requisitos afetados para cada opção e os requisitos para o mecanismo de seleção das opções.

Chances

Com a definição de variantes, a integração das necessidades específicas de diferentes stakeholders pode ser alcançada. Isso pode melhorar o nível de envolvimento e aumentar a satisfação dos stakeholders.

Desafios

Pode ser difícil criar opções que sejam realmente necessárias e que minimizem o esforço adicional no processo de desenvolvimento.

Variáveis

A definição do ponto de desvio pode ser muito cedo ou muito tarde no ciclo de vida do produto. Isso pode variar desde a configuração durante o tempo de execução até a configuração durante a instalação de produtos ou famílias de produtos completamente diferentes.

É importante encontrar o ponto de desvio correto para encontrar um bom equilíbrio entre o esforço (principalmente os custos) e as necessidades individuais dos stakeholders.

4.3.5 Imposição

O que é isso?

A anulação é o resultado da seleção de um conjunto de requisitos existente. É fundamental que as opções possíveis, ou o conjunto de requisitos, sejam bem compreendidos pelo tomador de decisões. Como o tomador de decisões pode se sobrepor a todas as partes envolvidas no conflito, o desequilíbrio de poder não influencia o resultado.

Papel dos participantes

O tomador de decisões deve entender as consequências de suas escolhas e as opções disponíveis. Ele também deve justificar a decisão para que ela possa ser aceita por todas as partes envolvidas no conflito. Especialistas podem ser consultados para coletar argumentos para as diferentes opções e para apoiar o tomador de decisão com o conhecimento necessário.

Preparação

Deve-se identificar um tomador de decisões que seja aceito por todas as partes envolvidas no conflito. As partes em conflito ou os especialistas preparam propostas contendo uma explicação de todas as opções sujeitas à anulação.

Aplicação

- O tomador de decisões precisa ler as propostas e pode fazer perguntas.
- Opcionalmente, cada parte pode apresentar sua proposta pessoalmente e responder às perguntas do tomador de decisões.
- O tomador de decisões anuncia a decisão.

Processamento de resultados

Os resultados, incluindo os motivos, devem ser documentados.

Produtos de trabalho típicos

Um documento contendo as propostas e a opção selecionada, bem como a justificativa para a decisão.

Chances

A hierarquia na organização pode ser usada para encontrar uma solução se não houver outra maneira de encontrá-la.

Desafios

Pode ser um desafio criar uma proposta que contenha todos os fatos, necessidades e consequências das diferentes opções.

Variáveis

A anulação pode ser realizada por um comitê de tomadores de decisão. Em vez do gerente das partes em conflito, um especialista neutro que seja aceito por todas as partes pode tomar a decisão.

4.3.6 Técnicas Auxiliares

Além disso, existem várias técnicas auxiliares, por exemplo:

- Comunicação não violenta [Rosenberg2015]
- Técnicas de negociação [FiUP2012]
- Considere todos os fatos [DeBono2006]
- Mais-menos-interessante [DeBono2006]
- Matriz de decisão [BiAB2006] [IsNe2013]

4.3.7 Encontrar uma técnica adequada de resolução de conflitos

Com base nas características de um conflito, devem ser selecionadas técnicas de resolução adequadas.

	Acordo	Compromisso	Definição de variantes (configurabilidade)	Votação	Imposição	Legenda:
						+ recomendado 0 aplicável, decidir com base em outras restrições se deve ser usado - não recomendado
1	-	-	0	+	+	Ouvir todos os argumentos poderia levar muito tempo e seria difícil de conseguir.
2	+	-	-	-	0	As decisões devem ser bem pensadas e aceitas por todas os stakeholders.
3	-	-	+	+	+	Veja 1. Pode ser inteligente criar variantes individuais para atender a diferentes necessidades (locais). A votação e a anulação podem ser feitas virtualmente.
4	-	-	-	+	+	A definição de variantes levará muito tempo, pois duas variantes e o mecanismo de configuração precisam ser definidos.

¹¹ Significa que o resultado errado da resolução leva a um alto risco para a implementação.

	Acordo	Compromisso	Definição de variantes (configurabilidade)	Votação	Imposição	Legenda:	
						+ recomendado 0 aplicável, decidir com base em outras restrições se deve ser usado - não recomendado	
5	A clareza do resultado é importante	+	+	-	0	+	Ao definir uma variante, não é fácil explicar o que o produto contém.
6	Baixa competência social dos stakeholders ¹²	-	-	+	+	+	O acordo e o compromisso exigem competência social para a troca de opiniões.
7	Situação complicada ¹³	-	+	-	-	0	Em situações complicadas, uma parte em conflito precisa se concentrar nos aspectos que consegue entender. A combinação de competências e a aceitação de diferentes aspectos levam a compromissos. Caso um especialista com a competência necessária seja o tomador da decisão, é possível anular a decisão.
8	Longa vida útil dos resultados	+	-	+	-	-	Compromissos, votações e anulações ruins podem levar a partes em conflito que não aceitam o resultado e querem mudá-lo ao longo do tempo.

¹² os stakeholders não podem ou não estão dispostos a ouvir umas às outras e não são capazes de aceitar outras opiniões.

¹³ O assunto é tão complicado que nem todas os stakeholders conseguem entender as consequências.

	Acordo	Compromisso	Definição de variantes (configurabilidade)	Votação	Imposição	Legenda:	
						+ recomendado 0 aplicável, decidir com base em outras restrições se deve ser usado - não recomendado	
9	Baixa motivação dos stakeholders (para participar ativamente)	-	-	+	+	+	O acordo e o comprometimento requerem algum tempo para discussões e envolvimento, portanto, é necessária uma alta motivação.
10	Pouca disponibilidade de tempo dos stakeholders	-	-	0	+	+	O acordo e o compromisso exigem algum tempo para discussões e participação. Dependendo da natureza da variante, pode ser menos demorado definir apenas sua própria variante em vez de todo o sistema.
11	Necessidades factuais conflitantes	-	0	+	-	-	Um compromisso, se possível, pode ser mais fácil de desenvolver, mas, na maioria dos casos, implicará em mais custos operacionais. Durante a vida útil do sistema, as variantes podem ser a melhor solução.
12	Conflito em relação aos dados	+	+	0	+	-	Dependendo do conteúdo, as variantes podem ser uma técnica adequada.
13	Conflito de interesses	-	0	+	0	+	A votação não resolve de fato o conflito, pois a maioria sempre vence.
14	Conflito em relação ao valor	-	-	+	-	0	Como os valores não podem ser alterados no curto prazo, a criação de variantes é a única maneira de satisfazer todos os valores. Se a criação de variantes não for uma técnica adequada, a única opção é a anulação.

	Acordo	Compromisso	Definição de variantes (configurabilidade)	Votação	Imposição		
						Legenda: + recomendado 0 aplicável, decidir com base em outras restrições se deve ser usado - não recomendado	
15	Conflito relativo à estrutura	-	0	-	+	+	<p>Os efeitos negativos do conflito estrutural impedem um acordo honesto e uma discussão justa, o que torna improvável um acordo ou compromisso. Entretanto, o compromisso pode ser a única técnica de resolução restante.</p> <p>Como a votação é anônima e a anulação envolve uma instância acima de ambas as partes em conflito, a distribuição de poder é minimizada.</p> <p>Se o conflito na distribuição do voto majoritário não for uma técnica adequada.</p>
16	Conflito relativo ao relacionamento	-	-	-	-	-	<p>Não existe uma técnica adequada ou a responsabilidade de resolver esse tipo de conflito não é da Engenharia de Requisitos.</p>

4.4 Documentar a resolução de conflitos

Após sua resolução, o conflito deve ser devidamente documentado. Além das características do conflito mencionadas na Seção 4.2, Isso deve incluir, em particular:

- As suposições sobre o conflito e sua resolução
- Restrições que influenciam a escolha da técnica e/ou resolução de conflitos
- As possíveis alternativas consideradas
- Resolução de conflitos, incluindo as razões para a resolução escolhida
- Quais foram os tomadores de decisão e outros colaboradores relevantes

Se não forem documentadas, os stakeholders podem simplesmente esquecer ou ignorar as decisões que foram tomadas, ou tentar alterá-las posteriormente. Isso geralmente ocorre em situações em que o conflito de requisitos em si é resolvido, mas um conflito social subjacente continua sem solução.

5 Habilidades do Engenheiro de Requisitos

Este capítulo apresenta uma visão geral das habilidades importantes para o Engenheiro de Requisitos. Como a comunicação é um elemento central da Engenharia de Requisitos, serão apresentadas teorias importantes de comunicação para proporcionar uma compreensão mais profunda da mecânica que impulsiona a comunicação. Por fim, como o desenvolvimento pessoal é um processo contínuo, as três últimas seções deste capítulo tratam da autorreflexão, das oportunidades de desenvolvimento pessoal e da aprendizagem contínua.

5.1 Habilidades necessárias nas áreas de elicitação

As abordagens e técnicas explicadas nos capítulos anteriores formam o conjunto de habilidades básicas de cada Engenheiro de Requisitos. O domínio dessas habilidades (difíceis) é um pré-requisito para profissionais e especialistas. Mas isso não é suficiente: para ser bem-sucedido, o Engenheiro de Requisitos também deve possuir várias habilidades sociais (consulte, por exemplo, [GleA2020], [Klaus2007]), como habilidades de comunicação, pensamento analítico, empatia, habilidades de resolução de conflitos, habilidades de moderação, autoconfiança e capacidade de convencer.

Marcel Robles [Robles2012] apresenta uma visão geral das dez habilidades interpessoais mais importantes:

- **Comunicação** – oral, capacidade de falar, escrita, apresentação, audição
- **Cortesia** – boas maneiras, etiqueta, etiqueta de negócio, gracioso, diz "por favor" e "obrigado", respeitoso
- **Flexibilidade** – adaptabilidade, disposição para mudar, aprendiz ao longo da vida, aceita coisas novas, ajusta-se, pode ser ensinado
- **Integridade** – honesto, ético, moral elevada, tem valores pessoais, faz o que é certo
- **Habilidades interpessoais** – simpático, pessoal, senso de humor, amigável, carinhoso, empático, tem autocontrole, paciente, sociabilidade, cordialidade, habilidades sociais
- **Atitude positiva** – otimista, entusiasmado, encorajador, feliz, confiante
- **Profissionalismo** – profissional, bem-vestido, aparência, equilíbrio
- **Responsabilidade** – responsável, confiável, faz o trabalho, engenhoso, autodisciplinado, quer fazer bem-feito, consciente, senso comum
- **Trabalho em equipe** – cooperativo, se dá bem com os outros, agradável, solidário, prestativo, colaborativo
- **Ética de trabalho** – esforçado, disposto a trabalhar, leal, iniciativa, automotivado, pontual, bom atendimento

Embora todos os itens acima sejam relevantes para todos os profissionais, as características a seguir também são particularmente importantes em relação à elicitação de requisitos:

- **Consciência contextual** – conhecer o contexto em que você está operando e ajustar sua abordagem de acordo com ele

- **Consciência ética** – qualquer tecnologia pode ter um impacto negativo sobre as pessoas, a sociedade e o meio ambiente. Se o Engenheiro de Requisitos detectar esse impacto negativo, ele deve torná-lo explícito para incentivar a discussão para atenuar o impacto.
- **Competência intercultural** – capaz de trabalhar em e com diferentes culturas (de negócio, de domínio, regional etc.)
- **Liderança** – capaz de conduzir os stakeholders a um determinado objetivo
- **Natureza motivadora** – capaz de inspirar os stakeholders para um determinado objetivo
- **Neutralidade** – capaz de atender a todas os stakeholders (relevantes) e seus interesses igualmente, sem interesses pessoais
- **Reflexão** – capaz de receber feedback e avaliar a situação e o próprio comportamento
- **Autoconhecimento** – conhecer sua própria posição em relação às stakeholders e ajustar sua abordagem de acordo com isso

Dependendo do negócio, do contexto do projeto e dos stakeholders em questão, algumas habilidades podem precisar de mais atenção do que outras. No entanto, todos são importantes: de fato, a consciência contextual refere-se à capacidade de aplicar as habilidades certas no momento certo, quando uma determinada situação assim o exige.

De todas as habilidades interpessoais, as habilidades de comunicação são o principal fator de sucesso para o Engenheiro de Requisitos – não é por acaso que Robles menciona essa habilidade em primeiro lugar.

Toda interação entre o Engenheiro de Requisitos e os stakeholders (que são as principais fontes de requisitos) é uma forma de comunicação e a maioria, se não todas, as habilidades mencionadas acima desempenham um papel. Algumas habilidades, por exemplo, integridade e neutralidade, são importantes para a própria comunicação; outras, como liderança e trabalho em equipe, só podem ser realizadas por meio da comunicação.

5.2 Teoria da comunicação e modelos de comunicação

A comunicação pode ser vista como uma forma de os indivíduos trocarem mensagens e criarem significado. Inclui qualquer comportamento de uma pessoa com o objetivo de deixar algo claro para outra pessoa que o percebe e interpreta. Além disso, a comunicação é considerada um processo complexo e interpessoal que usa qualquer combinação de fala, escrita e outros sinais como base para a troca de conceitos, pensamentos, opiniões e informações.

A comunicação é eficaz se a troca resultar em congruência entre o significado pretendido e o percebido. A comunicação eficaz não é necessariamente eficiente no sentido de que o resultado desejado é alcançado com um mínimo de esforço, tempo, complexidade e investimento de recursos. De fato, como a boa comunicação depende muito da redundância, mais eficiência pode facilmente levar a menos eficácia. O Engenheiro de Requisitos deve se certificar de que a comunicação com os stakeholders seja eficaz dentro de limites razoáveis de eficiência.

A comunicação geralmente é direcionada a um objetivo: o remetente tem a intenção de invocar determinado comportamento no receptor. Na elicitación e na resolução de conflitos, o Engenheiro de Requisitos está tentando motivar os stakeholders a se comportarem de forma aberta e colaborativa e a revelarem livremente todas as informações relevantes.

Em grande parte, a elicitación e a resolução de conflitos bem-sucedidas dependem da compreensão adequada dos "elementos básicos" da comunicação. A seguir, vários modelos do campo da teoria da comunicação são apresentados com sua relevância para a elicitación de requisitos.

O modelo Shannon-Weaver [ShWe1971] é geralmente considerado a "mãe de todos os modelos de comunicação". Ele se concentra na *codificação* de uma *mensagem* de um *remetente* para um *receptor*, que a *decodifica* após a *transmissão* por meio de um determinado *canal*, com o risco de o *ruído* perturbar essa mensagem (consulte Figura 25: O modelo de comunicação Shannon-Weaver).

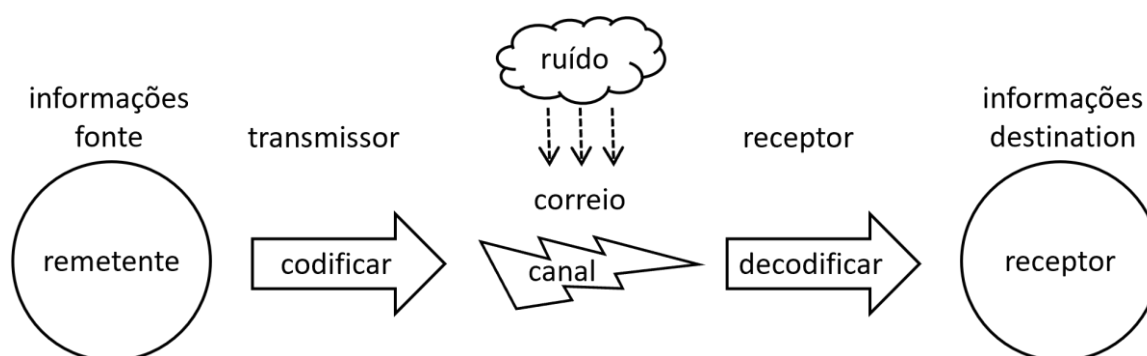


Figura 25: O modelo de comunicação Shannon-Weaver

Na Engenharia de Requisitos, muitas vezes é o ruído "virtual" que precisa de atenção, pois ele pode levar a requisitos incorretos e conflitos: preconceitos, fofocas, boatos, agendas ocultas. O Engenheiro de Requisitos deve ser capaz de reconhecer esse tipo de ruído, verificar a compatibilidade da codificação e da decodificação ("Entendemos um ao outro?") e verificar a compatibilidade da codificação e da decodificação.) e selecionar os canais adequados (p. ex., reuniões, apresentações, publicações) para a comunicação com os stakeholders.

Schramm [Schramm1971] contribuiu com dois modelos para a teoria da comunicação.

A comunicação é vista como uma *interação social* entre um emissor e um receptor. O primeiro modelo de Schramm (consulte Figura 26) deixa claro que uma mensagem só pode ser comunicada com sucesso se ela se encaixar em uma área de experiência compartilhada. Schramm também indicou que devemos examinar o impacto (desejado e indesejado) que uma mensagem tem sobre o alvo da mensagem.

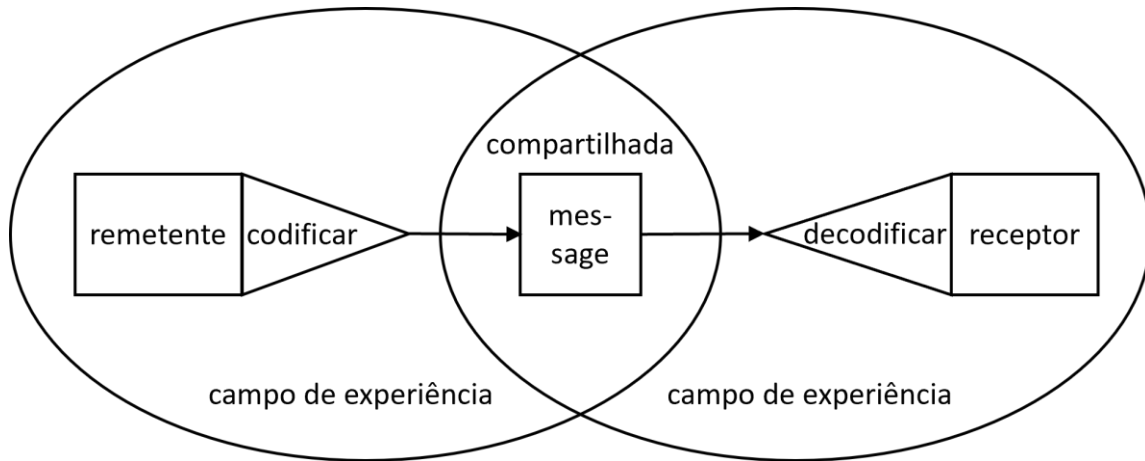


Figura 26: Modelo de comunicação de Schramm sobre áreas compartilhadas de experiência

O Engenheiro de Requisitos deve verificar se existe o compartilhamento necessário no relacionamento com os stakeholders. A falta dela pode levar ao fracasso. O Engenheiro de Requisitos pode ampliar a área de experiência compartilhada, por exemplo, adquirindo experiência no domínio por meio de estudo autônomo ou ministrando treinamento sobre questões de requisitos para os stakeholders.

Schramm também desenvolveu o *modelo circular* de comunicação (consulte Figura 27). Nesse modelo, o remetente codifica uma mensagem que é decodificada e interpretada pelo receptor, que responde codificando outra mensagem e transmitindo-a. Na elicitação, esse padrão pode ser observado, por exemplo, quando o Engenheiro de Requisitos faz perguntas a um interessado, ouve as respostas e as resume em seguida. A ideia de feedback se encaixa nesse modelo.

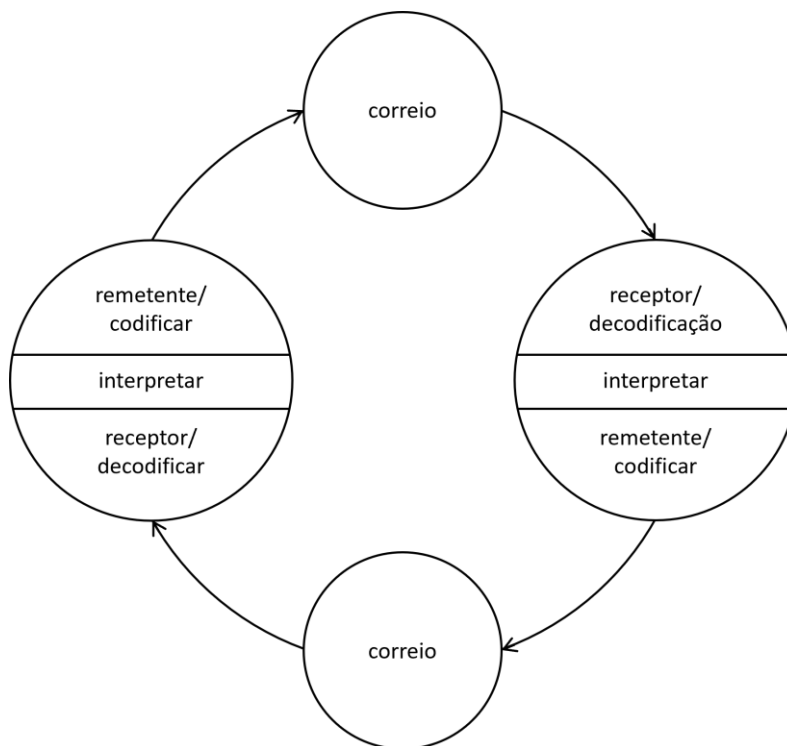


Figura 27: Modelo de comunicação circular de Schramm

O modelo de comunicação de 4 lados de Schulz von Thun [ScTh1981] coloca a *mensagem* no centro e descreve *quatro aspectos* a serem considerados (consulte Figura 28).

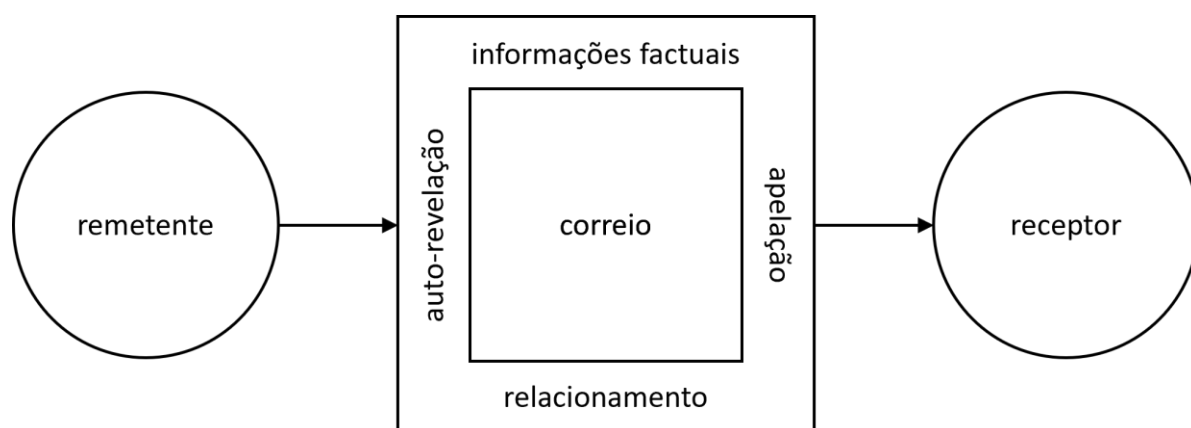


Figura 28: Modelo de comunicação de Schulz von Thun

O Engenheiro de Requisitos deve estar ciente de todos esses aspectos:

- *Informações factuais*: o conteúdo factual de uma mensagem é a base para a obtenção e a consolidação de requisitos.
- *Autorrevelação*: diz respeito ao grau de comprometimento de uma stakeholder com um determinado requisito, enquanto o Engenheiro de Requisitos deve manter uma neutralidade estrita.
- *Relacionamento*: o Engenheiro de Requisitos deve buscar um relacionamento aberto e construtivo com os stakeholders e deve verificar a perspectiva delas.
- *Apelo*: o aspecto do apelo pode dar pistas sobre o que os stakeholders esperam do Engenheiro de Requisitos e fornece feedback sobre o conteúdo factual comunicado.

Todos esses modelos contribuem para a compreensão da comunicação entre o Engenheiro de Requisitos e os stakeholders. Uma ideia comum é que a comunicação consiste em *compartilhar conceitos significativos entre indivíduos*. O ponto principal é que, durante essa comunicação, as informações podem ser perdidas, adicionadas, distorcidas ou mal interpretadas. Esses chamados *efeitos transformacionais* (consulte 3.3.3) resultam em conceitos percebidos de forma diferente entre os indivíduos envolvidos.

O Engenheiro de Requisitos deve garantir a qualidade de sua comunicação atenuando esses efeitos. É essencial prestar a devida atenção à *codificação* e *decodificação* das mensagens, escolhendo os *canais* corretos, evitando *ruídos* e verificando a interpretação correta por meio de *feedback*.

A *redundância* é a principal solução para problemas de comunicação. Normalmente, isso significa que a mesma informação é transmitida várias vezes, muitas vezes por meio de canais diferentes. Exemplos são um diagrama acompanhado de uma explicação ou uma apresentação apoiada por gestos não verbais. Se, e somente se, as mensagens redundantes forem consistentes, elas apoiarão a qualidade da comunicação.

Hint 5.2.1:

Use redundância em toda a sua comunicação.

Sempre peça feedback.

Fique atento aos efeitos transformacionais.

Tente inspirar o interessado a reformular as declarações suspeitas com outras palavras, exemplos concretos ou por meio de outras mídias, por exemplo, desenhos, metáforas.

5.3 Autorreflexão sobre as habilidades pessoais em elicitación de requisitos

"Eu sei que não sei nada" (Sócrates)

"Não sei de nada, sou de Barcelona" (Manuel)

O syllabus, este handbook e qualquer treinamento complementar estabelecem a base para a aplicação bem-sucedida das técnicas apresentadas. Entretanto, o desenvolvimento e o aprimoramento de habilidades pessoais para a obtenção de requisitos é um processo de aprendizado de longo prazo.

A base para o aprimoramento é a autorreflexão. Há três tipos de reflexão:

1. Pensar sobre o que você planejou fazer e repensar criticamente esse plano: Isso poderia ser feito de forma diferente? O que posso esperar das atividades planejadas? O que eu quero aprender? Posso melhorar dessa forma? (reflexão prospectiva)
2. Pensar em como estou fazendo meu trabalho atualmente: Estou me comportando profissionalmente? Eu aplico o que aprendi? Quais são meus medos e esperanças atuais em relação às atividades de elicitación e resolução de conflitos em que estou envolvido? (reflexão complementar)
3. Pensar sobre o que você fez e como executou suas tarefas de elicitación e resolução de conflitos no passado (reflexão retrospectiva).



Figura 29: Três tipos de reflexão

Mesmo que a Engenharia de Requisitos durante um projeto de desenvolvimento seja considerada um sucesso, normalmente há várias oportunidades de melhoria.

Hint 5.3.1:

Aqui está uma pequena lista de verificação de perguntas para ajudá-lo a iniciar seu processo de autorreflexão:

- Uma técnica produziu os resultados esperados/contribuiu para o desenvolvimento? Em caso afirmativo, qual foi o principal fator de sucesso? Em caso negativo, qual foi o motivo da falha? O que eu poderia ter feito de diferente? O que devo melhorar na próxima vez?
- Os stakeholders aceitaram as técnicas de elicitação/resolução de conflitos aplicadas? Em caso afirmativo, qual foi o motivo da colaboração? Em caso negativo, qual foi o problema? O que eu poderia ter feito de diferente? O que devo melhorar na próxima vez?
- O esforço para uma técnica foi justificável com relação à contribuição para o desenvolvimento? Se não, por quê? Foi um problema da técnica em nosso contexto (ou seja, eu deveria ter selecionado uma técnica mais apropriada) ou como falhei ao aplicar a técnica adequadamente?
- Qual técnica poderia ter permitido a obtenção de requisitos que surgiram no final do desenvolvimento em um momento anterior? Qual foi o motivo para não identificá-los antes?
- Quais técnicas alternativas também poderiam ter sido aplicadas? Seria vantajoso considerar técnicas específicas de elicitação ou de resolução de conflitos para um projeto futuro em uma situação semelhante? É necessário adquirir mais conhecimento e experiência sobre eles (consulte a Seção 0)?

Como Engenheiro de Requisitos avançado, depois de ter feito muitas sessões de autorreflexão, você desenvolverá cada vez mais perguntas, se concentrará em aspectos específicos etc., até chegar à sua própria lista de verificação personalizada. Ou, se você se tornar um especialista em autorreflexão, poderá identificar os tópicos a serem refletidos no decorrer do seu trabalho. No entanto, como iniciante (inexperiente em autorreflexão), tome cuidado com a simplificação excessiva: deitar-se na cama, fingir que está pensando em si mesmo e fazer uma autorreflexão profissional pode resultar em um cochilo relaxante à tarde, em vez de insights sobre si mesmo e sua proficiência em elicitação!

Hint 5.3.2:

Portanto, aqui estão algumas dicas de reflexão para iniciantes:

- Inicie sua atividade de autorreflexão em frente a uma boa xícara de café ou chá (ou a bebida que preferir); tenha suas anotações e dados de entrada disponíveis (consulte a Dica 5.3.3).
- Planeje seu processo de autorreflexão: Sobre qual questão você quer refletir primeiro? Inicialmente, use uma lista de verificação existente de perguntas para reflexão, como a fornecida na Dica 5.3.1.
- Faça anotações sobre o resultado de seu raciocínio.
- Atenha-se ao tópico selecionado! Evite mudar o tópico antes de ter abordado todos os seus aspectos. Considere sistematicamente todos os aspectos da pergunta selecionada.
- Não pense apenas em problemas e erros e em como evitá-los no futuro; capture também critérios e lições aprendidas relacionadas a sucessos/resultados positivos.

– Relacionar seu pensamento a conceitos conhecidos e teorias estabelecidas. Se necessário, diga que você precisa aprender mais sobre um método específico depois. Também é possível consultar um livro-texto (ou a Internet) para avaliar uma ação de elicitação ou resolução de conflitos realizada. Certifique-se de não se perder nos detalhes de uma fonte escrita. Retorne rapidamente à pergunta de reflexão que foi o motivo da excursão pela literatura.

Hint 5.3.3:

Como insumo para sua autorreflexão, você pode primeiro coletar dados sobre você e seu comportamento profissional:

- Se possível, faça um vídeo ou uma gravação de voz de uma atividade de elicitação que você realiza (p. ex., uma entrevista).
- Peça feedback a seus colegas sobre aspectos específicos de seu desempenho profissional. Por exemplo, um colega pode observá-lo entrevistando uma stakeholder ou moderando um workshop. – Ou um especialista analisa suas anotações de observação ou avalia um questionário que você criou. Ler os resultados e discuti-los com o observador/avaliador pode fornecer informações valiosas para sua autorreflexão.
- Peça feedback às stakeholders. Isso pode ser feito com uma breve entrevista (p. ex., imediatamente após uma entrevista de elicitação) ou com um breve questionário. No entanto, tenha cuidado para não misturar atividades de elicitação e resolução de conflitos com a solicitação de feedback sobre seu comportamento. No caso de conflitos ou relutância em fornecer informações sobre o tópico de elicitação, os participantes podem não estar preparados para apoiar seu processo de autorreflexão.
- Aplique uma abordagem centrada no ser humano ao seu próprio trabalho: por exemplo, testar um questionário que você desenvolveu com alguns stakeholders preliminares pode fornecer percepções que não apenas melhoram o questionário em si, mas também servem como insumo para a reflexão sobre o seu desempenho.
- Peça a um Engenheiro de Requisitos sênior para ser seu instrutor.
- Aplique a conhecida técnica do diário em seu próprio trabalho. Fazer anotações sobre problemas, ações bem-sucedidas, feedback informal, lições aprendidas etc. ajuda você a se lembrar ao realizar o processo de reflexão. Uma folha de avaliação dos recursos previamente definidos também é um instrumento de medição adequado [SmMa2011].

A coleta de informações das pessoas ao seu redor às vezes é chamada de feedback 360° [LeLu2009].

5.4 Oportunidades de desenvolvimento pessoal

A experiência prática insuficiente é muitas vezes apresentada como motivo para não aplicar uma técnica específica de elicitação ou de resolução de conflitos. Essa atitude pode ser compreensível em termos de sucesso do projeto (o Engenheiro de Requisitos aplica as técnicas que conhece melhor para garantir o sucesso do projeto); em termos de desenvolvimento pessoal, no entanto, essa atitude não é útil, pois o Engenheiro de Requisitos nunca aprenderá técnicas desconhecidas e, assim, ampliará sua caixa de ferramentas. Na seção a seguir, apresentamos duas boas práticas que permitem a aplicação de novas técnicas em projetos em andamento: aplicação em ambientes de baixo risco e aplicação em paralelo a uma técnica conhecida.

Uma técnica desconhecida pode ser aplicada em um ambiente de baixo risco para minimizar os efeitos negativos, caso a técnica forneça o resultado esperado. O que constitui uma configuração de baixo risco depende, é claro, do contexto do projeto. As características típicas de um ambiente de baixo risco podem ser:

- Aplicar uma técnica somente a um subgrupo reduzido de stakeholders. Por exemplo, o Engenheiro de Requisitos aplica o aprendizado apenas a um pequeno número de stakeholders.
- Aplicação de uma técnica por um período de tempo limitado. Por exemplo, o Engenheiro de Requisitos planeja uma observação de campo muito curta (p. ex., uma hora) com determinados stakeholders.
- Aplicação de uma técnica em um ambiente amigável. Por exemplo, um Engenheiro de Requisitos que trabalhou por muito tempo em um projeto e estabeleceu um bom relacionamento com os stakeholders. Em um ambiente como esse, a aplicação de técnicas desconhecidas geralmente é mais fácil.
- Aplicação de uma técnica a um aspecto do sistema que não é considerado crítico. Por exemplo, a edição do perfil do usuário em uma loja virtual pode ser considerada um recurso não crítico. Portanto, o Engenheiro de Requisitos pode usar uma técnica desconhecida para obter requisitos para esse recurso.

Hint 5.4.1:

Se você aplicar uma técnica desconhecida, seja honesto com seus acionistas. Deixe claro para eles que a técnica aplicada é nova para você e peça-lhes um feedback depois. Essa abordagem honesta com os stakeholders reduz a pressão sobre você. Os stakeholders normalmente reconhecem se você está inseguro com uma nova técnica e geralmente estão dispostos a lhe dar feedback para melhorar suas habilidades nessa área.

Além disso, é possível aplicar uma técnica desconhecida em paralelo a uma técnica conhecida para reduzir o risco de falha. A aplicação paralela pode ocorrer de várias formas:

- **Paralelismo síncrono:** caso um grande grupo de participantes precise ser abordado durante a elicitação de requisitos, um subgrupo menor desses participantes pode aplicar uma técnica desconhecida.

Por exemplo, 30 participantes devem participar de um workshop de criatividade. Um subgrupo de 6 participantes pode ser convidado a executar uma técnica desconhecida (p. ex., 6 chapéus pensantes), enquanto a maioria aplica uma técnica conhecida (p. ex., brainstorming).

- **Paralelismo assíncrono:** se não for possível (ou desejável) dividir um grupo de participantes para executar duas técnicas em paralelo, as duas técnicas podem ser executadas sequencialmente. Por exemplo, cinco usuários finais de um novo sistema devem participar da elicitação de requisitos por meio de entrevistas. Em tal situação, o Engenheiro de Requisitos pode planejar uma observação de campo adicional logo após cada entrevista.

Hint 5.4.2:

Se você aplicar uma técnica desconhecida em paralelo a uma técnica conhecida, planeje uma comparação explícita dos resultados de ambas as técnicas com os participantes. Dessa forma, seus acionistas também podem se beneficiar dos novos resultados.

5.5 Aprender com a experiência anterior - aprendizado contínuo

Os componentes essenciais de um processo de treinamento pessoal que promove o aprendizado com base em experiências anteriores são:

- *Medição regular de seu próprio perfil de habilidades:* A consciência de seus próprios pontos fortes e fracos em termos de perfil de habilidades é a base para um desenvolvimento posterior bem-sucedido. Análises regulares (p. ex., por meio de questionários de autoavaliação ou conversas com clientes ou colegas) de seu próprio perfil de habilidades promovem a conscientização de seus próprios pontos fortes e fracos.
- *Medidas de treinamento:* Para aprimorar seu perfil de habilidades, é necessário realizar educação adicional, treinamento ou coaching em uma ou mais técnicas de elicitação/resolução de conflitos ou nas habilidades necessárias (p. ex., treinamento de liderança ou motivacional).
- *Aprimoramento no trabalho cotidiano:* Medidas de treinamento dedicadas são o primeiro passo para aprimorar suas próprias habilidades. No entanto, o progresso substancial só é alcançado por meio da aplicação e da prática no trabalho cotidiano. O aprimoramento contínuo de suas habilidades deve, portanto, ser parte integrante de seu trabalho prático. É possível obter um bom progresso se o aprimoramento das habilidades individuais for promovido (p. ex., aprimoramento das habilidades de liderança em workshops) durante um período prolongado (pelo menos 4 semanas).
- *Medidas de mentoria:* Uma alternativa ao treinamento intensivo é aprender um método com um mentor experiente. Normalmente, você pode primeiro ajudar seu mentor durante a aplicação de uma técnica e aprender observando o comportamento dele. Posteriormente, seu mentor delega responsabilidades a você, observa seu desempenho e fornece feedback.

Dicas 5.5.1:

Desenvolva e cultive seu caminho pessoal de aprimoramento contínuo das habilidades de Engenharia de Requisitos. Aprender os fatos da maioria das técnicas é fácil; desenvolver a excelência na aplicação dessas técnicas é outra história. Não existe uma escada predefinida para o céu (ou excelência em RE)! Use os elementos descritos nesta seção como uma caixa de ferramentas para encontrar seu caminho de aprimoramento pessoal.

6 Referências e leituras adicionais

- [A4qu2018] Alliance for Qualification: A4Q Design Thinking Foundation Syllabus. <https://isqi.org/de/55-a4q-design-thinking-foundation-level.html>, 2018. Last visited October 2021.
- [AII1977] Alexander, S. Ishikawa, M. Silverstein: A Pattern Language – Towns – Buildings – Construction. Oxford University Press, New York, 1977.
- [Alexander2005] I. F. Alexander: A Taxonomy of Stakeholders – Human Roles in System Development. International Journal of Technology and Human Interaction, Vol 1, 1, 2005, pages 23–59.
- [BaGL1996] V.R. Basili, S. Green, O. Laitenberger, et al.: The Empirical Investigation of Perspective-Based Reading. Empirical Software Engineering 1: 133. <https://doi.org/10.1007/BF00368702>, 1996. Última revisão em Fevereiro de 2019.
- [BaCC2015] K. Baxter, C. Courage, K. Caine: Understanding Your Users – A Practical Guide to User Research Methods, Morgan Kaufmann; 2nd edition, 2015.
- [BaGr2005] R. Bandler, J. Grinder: The Structure of Magic, Vol. 1 – A Book about Language and Therapy; 1st edition, 2005.
- [BeHo1998] H. Beyer, K. Holtzblatt: Contextual Design – Defining Customer-Centered Systems. Morgan Kaufmann, 1998.
- [Beveridge 1957] W. I. Beveridge.: The Art of Scientific Investigation. The Blackburn Press, 1957.
- [Design Council 2007] Design Council: 11 lessons: managing design in 11 global brands – A study of the design process. <https://www.designcouncil.org.uk/search/?query=Double+Diamond+model>. Última revisão, março de 2024.
- [BiAB2006] Stefan Biffel, Aybuke Aurum, Barry Boehm: Value-Based Software Engineering. Springer-Verlag, Berlin, 2006.
- [Bitkom2017] Bitkom: Role Model Digital Design, <https://www.bitkom.org/Bitkom/Publikationen/Role-Model-Digital-Design.html>. Última revisão, março de 2024.
- [Boehm2006] B. Boehm: A View of 20th and 21st Century Software Engineering. 28th international conference on Software engineering (ICSE '06), p. 12–29, 2006.
- [Bourne2015] L. Bourne: Making Projects Work: Effective Stakeholder and Communication Management. CRC Press, 2015.
- [Brls2005] J. Brown, D. Isaacs: The World Café: Shaping Our Futures Through Conversations That Matter, Berrett-Koehler Publishers, 2005.
- [Brown2009] T. Brown: Change by Design – How Design Thinking Transforms Organizations and Inspires Innovation. HarperCollins, 2009.

- [Buxton2007] B. Buxton: Sketching User Experiences – Getting the design right and the right design. Morgan Kaufmann, São Francisco, 2007.
- [BuBu2005] T. Buzan, B. Buzan: Das Mind-Map-Buch – Die beste Methode zur Steigerung Ihres geistigen Potenzials, Morgan Kaufmann; 5. aktualisierte Auflage, 2005.
- [BuHe2019] S. Bühne, A. Herrmann: Handbook Requirements Management According to the IREB Standard (versão 1.1.0), IREB e.V., Karlsruhe, 2019.
<https://www.ireb.org/downloads/#cpre-advanced-level-requirements-management-handbook>. Last visited May 2022
- [Cambridge2017] Cambridge Dictionary
<https://dictionary.cambridge.org/dictionary/english/abstraction>. Última revisão em Fevereiro de 2019.
- [Carrol2003] JM. Carrol: Making Use. The MIT Press, 2003.
- [Chernak2012] Y. Chernak: Requirements reuse: the state of the practice. In: Proceedings of the 2012 IEEE international conference on software science, technology & engineering (SwSTE), p. 46–53
- [Cockburn2001] A. Cockburn: Writing Effective Use Cases. Addison–Wesley, 2001.
- [CRCN2014] A. Cooper, R. Reimann, D. Cronin, C. Noessel: About Face: The Essentials of Interaction Desig. 4th Edition, John Wiley & Sons, Inc, Hoboken, 2014.
- [CoSh2007] T. Colburn, G. Shute: Abstraction in Computer Science. 169–184, 2007.
- [Cooper2004] A. Cooper: The Inmates Are Running the Asylum–Why High–tech Products Drive Us Crazy and How to Restore the Sanity. Que, Indianapolis, 2004.
- [Couger1996] J. D. Couger: Creativity and Innovation in Information Systems Organizations. Byod & Fraser, 1996.
- [CrOB2006] O. Creighton, M. Ott, B. Bruegge: Software Cinema: Video-based Requirements Engineering. 14th IEEE International Requirements Engineering Conference (RE'06), 2006.
- [CHQW2016] T. Cziharz, P. Hruschka, S. Queins, T. Weyer: Handbook of Requirements Modeling According to the IREB Standard (versão 1.3), IREB e.V., Karlsruhe, 2016. <https://www.ireb.org/en/downloads/#handbook-cpre-advanced-level-requirements-modeling>. Última revisão, março de 2024.
- [DeBono2006] E. DeBono: Edward DeBono's Thinking Course – Powerful Tools to Transform Your Thinking. BBC Active, Londres, 2006.
- [DeDe2011] K. Dewalt, B. Dewalt: Participant observation – A guide for fieldworkers. 2nd Edition. AltaMitra Press, Plymouth, UK, 2011.
- [DDP2021] IREB e.V.: Syllabus DDP Foundation Level, version 1.0.1.
<https://www.digitaldesign.org/syllabus>. Última revisão, março de 2024.
- [EIOs2017] R. Elamin, R. Osman: Towards Requirements Reuse by Implementing Traceability in Agile Development, em: 2017 IEEE 41st Annual Computer

Software and Applications Conference (COMPSAC) Itália,
<http://ieeexplore.ieee.org/abstract/document/8029969/?part=1>, 2017. Última
revisão, março de 2024.

- [FiUP2012] R. Fisher, W. Ury, B. Patton: Getting to Yes – Negotiating an agreement without giving in, 3ª ed. rev. Random House Business, Nova York, 2012.
- [GHJV1994] E. Gamma, R. Helm, R. Johnson, J. Vlissides: Design Patterns. Elements of Reusable Object-Oriented Software. Prentice Hall, 1994.
- [Glas1999] F. Glas: Confronting Conflict – A first-aid kit for handling conflict. Hawthorn Press, Gloucestershire, 1999.
- [Glas2004] F. Glas: Konfliktmanagement: Ein Handbuch für Führungskräfte, Beraterinnen und Berater. 11ª ed., Freies Geistesleben, Stuttgart, 2004.
- [GleA2020] M. Glinz et al: Handbook for the CPRE Foundation Level according to the IREB Standard (Versão 1.0.0). IREB e.V., Karlsruhe, 2020.
<https://www.ireb.org/en/downloads/#cpre-foundation-level-handbook>.
Última revisão, março de 2024.
- [Glin2020] M. Glinz: A Glossary of Requirements Engineering Terminology (Version 2.0.0). IREB e.V., Karlsruhe, 2020. <https://www.ireb.org/en/downloads/#cpre-glossary>. Última revisão, março de 2024.
- [GoBe2015] L. Goldin, D. M. Berry: Reuse of requirements reduced time to market at one industrial shop: a case study, Requirements Engineering Journal 2013. In: Requirements Engineering–Volume 20, Issue 1, March 2015.
- [Goodwin2009] K. Goodwin: Designing for the Digital Age. Wiley, New York, 2009.
- [Gothelf2013] J. Gothelf: Lean UX – Applying Lean Principles to Improve User Experience. O'Reilly, Sebastopol, 2013.
- [Gottesdiener2002] E. Gottesdiener: Requirements by Collaboration: Workshops for Defining Needs, Addison–Wesley Professional, 2002.
- [GoWo2005] T. Gorschek, C. Wohlin: Requirements Abstraction Model, Requirements Engineering Journal 2005. In Requirements Engineering Volume 11, Issue 1, Pages 79 – 101, December 2005.
- [Groen et al.2017] E. C. Groen, N. Seyff, R. Ali, F. Dalpiaz, J. Doerr, E. Guzman, M. Hosseini, J. Marco, M. Oriol, A. Perini, M. Stade: The Crowd in Requirements Engineering – The Landscape and Challenges, IEEE Software, vol. 34, no. 2, pp. 44–52, 2017.
- [GrKo2016] E. C. Groen, M. Koch: How Requirements Engineering can benefit from crowds – Driving innovation with crowd-based techniques. Requirements Engineering Magazin Vol. 2. <https://re-magazine.ireb.org/issues/2016-2-take-the-broader-view/how-requirements-engineering-can-benefit-from-crowds>. 2016. Última revisão, março de 2024.
- [HaPy2012] R. Hartson, P.S. Pyla: The UX Book: Process and Guidelines for ensuring a Quality User Experience. Morgan Kaufmann, São Francisco, 2012.

- [IREB2020] IREB Certified Professional for Requirements Engineering Foundation Level Syllabus (version 3.0.1). IREB e.V., Karlsruhe, 2020.
<https://www.ireb.org/en/downloads/#cpre-foundation-level-syllabus-3-0>.
Ultima revisão, março de 2024.
- [IsNe2013]A. Ishizaka, P. Nemery: Análise de decisão multicritério. Métodos e software. John Wiley & Sons, Ltd, Chichester, 2013.
- [ISO25010] ISO/IEC 25010:2011: Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models. International Organization for Standardization, Geneva, 2011.
- [ISO29148] ISO/IEC/IEEE29148: Systems and software engineering — Life cycle processes — Requirements Engineering. International Organization for Standardization, Geneva, 2011.
- [ISO9241.11] ISO9241: Ergonomic requirements for office work with visual display terminals (VDTs) -- Part 11: Guidance on usability. International Organization for Standardization, Geneva, 1998.
- [ISO9241.210] ISO9241: Ergonomics of human-system interaction -- Part 210: Human-centred design for interactive systems. International Organization for Standardization, Geneva, 2010 (replaces ISO 14407-210: 1999).
- [JoDö2004] I. John, J. Dörr: Requirements Engineering, basierend auf existierenden Systemen, em: G. Böckle, P. Knauber, K Pohl, K Schmid. "Software-Produktlinien- Methoden, Einführung und Praxis", dpunkt.Verlag, Heidelberg 2004, p.153-163.
- [KaBP2002] C. Kaner, J. Bach, B. Pettichord: Lessons Learned in Software Testing – A Context-Driven Approach. Wiley, New York, 2002.
- [Katie2017] Katie: Pril Gets Pranked. Social Media for Business Performance, <https://smbp.uwaterloo.ca/2017/05/pril-gets-pranked/>, 2017. Última revisão em Fevereiro de 2019.
- [Klaus2007] P. Klaus: The Hard Truth About Soft Skills – Workplace Lessons Smart People Wish They'd Learned Sooner. HarperCollins Publishers, Nova York, 2007.
- [KnZK2016] J. Knapp, J. Zeratsky, B. Kowitz: Sprint – How to Solve Big Problems and Test New Ideas in Just Five Days. Simon & Schuster. 2016.
- [Koelsch2016] G. Koelsch: Requirements Writing for System Engineering. Apress, 2016.
- [Koes1964] A. Koestler: The Act of Creation. Last Century Media, 2014.
- [Kumar2013] V. Kumar: 101 Design Methods – A Structured Approach for Driving Innovation in Your Organization. Wiley, 2013.
- [Kvale2008] S. Kvale: Doing Interviews. SAGE, 2008.
- [Laue2014] K. Lauenroth: What does it mean to say „requirement“?–An inquiry into the abilities of the human mind and the meaning of the word „requirement“.

Requirements Engineering Magazin Vol. 1. <http://re-magazine.ireb.org/issues/2014-1-learning-to-fly/what-does-it-mean-to-say-requirement>. Última revisão, março de 2024.

- [LeLL2018] M. Lewrick, P. Link, L. Leifer: The Design Thinking Playbook: Mindful Digital Transformation of Teams, Products, Services, Businesses and Ecosystems. Wiley, Nova Jersey, 2018.
- [LeLu2009] R. Lepsinger, A.D. Lucia: The Art and Science of 360 Degree Feedback (A arte e a ciência do feedback de 360 graus). 2ª ed., Wiley, São Francisco, 2009.
- [LiHB2003] W. Lidwell, K. Holden, J. Butler: Universal Principles of Design. Rockport, 2003.
- [LiFi2012] S. L. Lim, A. Finkelstein: StakeRare: Using Social Networks and Collaborative Filtering for Large-Scale Requirements Elicitation, em IEEE Transactions on Software Engineering, vol. 38, no. 3, pp. 707–735, 2012.
- [LiOg2011] J. Liedtka, T. Ogilvie. Designing for Growth: A Design Thinking Tool Kit For Managers. Columbia University Press, 2011.
- [LoSL2017] H. van Loenhoud, P. Steiger, K. Lauenroth: The goal is to solve the problem – Some thoughts on problems and goals in the context of Requirements Engineering. Revista RE 2017–02. <https://re-magazine.ireb.org/articles/the-goal-is-to-solve-the-problem>, 2017. Última revisão em Fevereiro de 2019.
- [Maiden et al.2010] N. Maiden, S. Jones, I. Karlsen, R. Neill, K. Zachos, A. Milne: Requirements Engineering as Creative Problem Solving: A Research Agenda for Idea Finding. 18th IEEE International Requirements Engineering Confereneec (RE), 2010.
- [MaGi2001] N. Maiden, A. Gizikis: De onde vêm os requisitos? IEEE Software Vol. 18, No. 5, 2001, S.10–12.
- [McCo2006] S. McConnell: Software Estimation – Demystifying the Black Art, Microsoft Press, 2006.
- [McEl2017] K. McElroy: Prototyping for Designers: Developing the Best Digital and Physical Products. O’Reilly, 2017.
- [GrBa2017] J. McGrath, B. Bates: The Little Book of Big Management Theories. 2ª edição, Pearson Business, 2017.
- [Mill2009] R.E. Miller: The Quest for Software Requirements. MavenMark Books, 2009.
- [Moor2014] C. W. Moore: The Mediation Process – Practical Strategies for Resolving Conflicts. 4th ed., John Wiley & Sons, Hoboken, 2014.
- [Osbo1948] A. F. Osborn: Your creative power: How to use imagination. C. Scribner's Sons, 1948. (Accessed as digital reprint: Read Books Ltd. (epub eBook), abril de 2013)
- [Owen2008] H. Owen: Open Space Technology: A User's Guide. Berrett-Koehler Publishers, 3ª ed., 2008.

- [DsSch2012] An Introduction to Design Thinking PROCESS GUIDE. Hasso Plattner Institute of Design. <https://hpi.de/en/school-of-design-thinking/design-thinking.html>. Ultima revisão, março de 2024.
- [Pohl2010] K. Pohl: Requirements Engineering – Foundations, Principles, and Techniques. Springer, 2010.
- [PoRu2015] K. Pohl, C. Rupp: Requirements Engineering Fundamentals: A Study Guide for the Certified Professional for Requirements Engineering Exam – Foundation Level – IREB compliant, Rocky Nook, Santa Barbara, 2015.
- [Port2013] S. Portigal: Entrevistando usuários: How to Uncover Compelling Insights. Rosenfeld Media, Brooklyn, 2013.
- [Rein1997] D. Reinertsen: Managing the Design Factory: A Product Developers Tool Kit. Free Press, 1997.
- [RiFl2014] M. Richter, M. Flückiger: User-Centred Engineering – Creating Products for Humans. Springer, Berlin Heidelberg, 2014.
- [Robertson2001] S. Ian Robertson: Problem Solving. Psychology Press, 2001.
- [Robles2012] M.M. Robles: Executive Perceptions of the Top 10 Soft Skills Needed in Today's Workplace. Business Communication Quarterly 75(4) p. 453–465, 2012.
- [RoRo2013] S. Robertson, J. Robertson: Mastering the Requirements Process: Getting Requirements Right. Third Edition, Pearson Education, London, 2013.
- [Rosenberg2015] M. B. Rosenberg: Nonviolent Communication – A Language of Life. 3rd rev. ed., Puddle Dancer Press (US), Encinitas, 2015.
- [RuCh2008] J. Rubin, D. Chisnell: Handbook of Usability Testing– How to Plan, Design, and Conduct Effective Tests. Wiley; Indianapolis, 2008.
- [Rupp et al.2014] C. Rupp, die SOPHISTen: Requirements-Engineering und -Management – Aus der Praxis von klassisch bis agil. 6ª ed., Carl Hanser Verlag, München, 2014. (selected chapters English version see <http://www.sophist.de/en/infopool/downloads>. última revisão, fevereiro de 2019.)
- [Schramm1971] W.L. Schramm: How communication works, in W.L. Schramm, ed., (1971). The Process and Effects of Mass Communication, rev. ed., University of Illinois Press, 1971.
- [Schulz von Thun 1981] F. Schulz von Thun: Miteinander reden 1 – Störungen und Klärungen. Psychologie der zwischenmenschlichen Kommunikation. Rowohlt, Reinbek, 1981.
- [Shackel1991] B. Shackel: Usability – Context, Framework, Definition, Design and Evaluation. In B. Shackel & S. Richardson (Eds.): Human Factors for Informatics Usability (p. 21–37), University Press, Cambridge, Reino Unido, 1991.
- [ShRP2007] H. Sharp, Y. Rogers, J. Preece: Interaction Design: Beyond Human-Computer Interaction. John Wiley & Sons, Hoboken, 2007.

- [ShWe1971] C.E. Shannon & W. Weaver: The Mathematical Theory of Communication. University of Illinois Press, 1971.
- [SiOp2005] G. Sindre, A. L. Opdahl: Eliciting security requirements with misuse cases. Requirements Engineering Journal, Vol. 10, No. 1, 2005.
- [SmMa2011] S. Smith, R. Mazin: The HR Answer Book: An Indispensable Guide for Managers and Human Resources Professionals. 2ª edição, AMACOM, Nova York, 2011.
- [Snijders et al.2015] R. Snijders, F. Dalpiaz, S. Brinkkemper, M. Hosseini, R. Ali, A. Özüm: REfine: A Gamified Platform for Participatory Requirements Engineering, Proc. 1º Workshop Internacional de Engenharia de Requisitos Baseados em Multidões (CrowdRE 15), p. 1–6, 2015.
- [Snyder2003] C. Snyder: Paper Prototyping – The Fast and Easy Way to Design and Refine User Interfaces. Morgan Kaufmann, 2003.
- [TiSi2017] S. Tiwari, S. Singh Rathore: A Methodology for the Selection of Requirement Elicitation Techniques. Em: arXiv e-prints, 2017.
<https://arxiv.org/abs/1709.08481>. Última revisão, março de 2024.
- [UXQB2017] UXQB Advanced Level – Usability Testing and Evaluation (CPUX–UT), Version 1.07, February 2017.
- [Walten et al.2015] D.D. Walten, G.J. Roedler, K.J. Forsberg, R.D. Hamelin, T.N. Shortell: Systems Engineering Handbook – A Guide for System Life Cycle Process and Activities. Wiley, 2015.
- [Warfel2009] T. Warfel: Prototyping – A Practitioner's Guide. Rosenfeld Media, 2009.
- [Wiki2017] Wikipedia Principle of abstraction:
https://en.wikipedia.org/wiki/Principle_of_abstraction. Última revisão, março de 2024.
- [Withall2007] S. Withall: Software Requirements Patterns. Microsoft Press, 2007.
- [YoAs2015] M. Yousuf, M. Asger: Comparison of Various Requirements Elicitation Techniques. In: International Journal of Computer Applications, Vol. 116, No. 4, 2015.