

Elicitação

Modelagem

Análise

Requisitos – Aula 07

Professores: Milene Serrano e Maurício Serrano

Agenda

- › Considerações Iniciais
- › Elicitação de Requisitos
 - Subatividades
 - › abstraindo a metodologia escolhida...
 - Técnicas
 - Priorização
- › Considerações Finais

Considerações Iniciais

Considerações Iniciais

Requisito

- Requisito representa uma condição necessária para a obtenção de certo objetivo, ou para o preenchimento de certo fim; quesito.
- Requisitos não incluem somente necessidades dos usuários, mas também aquelas que surgem de padrões organizacionais, governamentais e industriais em geral.
- Portanto, trata-se de uma declaração que identifica uma capacidade, característica física, ou fator de qualidade que ressalta uma necessidade de um produto ou processo para o qual uma solução será adotada.
- Assim sendo, foca nas necessidades não só de produto, mas também de processo, além de que um requisito pode, também, ser um atributo de qualidade.

Considerações Iniciais

Cientes da
definição de
requisitos, surgem
perguntas:



O que é elicitar requisitos?
Como elicitar requisitos?
Como priorizar requisitos?

Considerações Iniciais

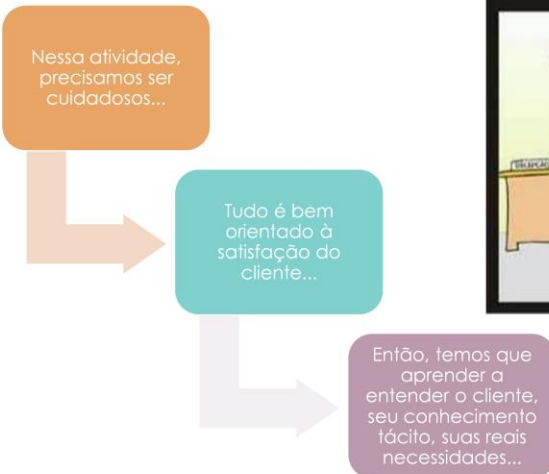
O que é elicitar requisitos?



O termo elicitar pode ser definido como: definir, **tornar explícito**, obter o máximo de informação sobre o objeto em questão.

Considerações Iniciais

Nem tudo pode ser interpretado ao pé da letra...



Nem tudo pode ser interpretado ao pé da letra...

Considerações Iniciais

Interessados podem não ter ideia precisa e clara sobre o que querem...

Interessados costumam ter dificuldades para descreverem seus conhecimentos sobre o domínio do problema...

Interessados, entre eles, possuem pontos de vista diferentes...

Interessados e equipe técnica de software também possuem pontos de vista diferentes...

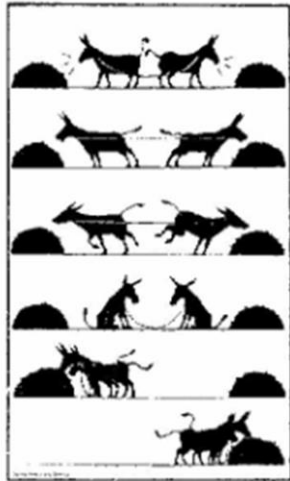
Interessados podem se negar a participar do processo de elicitação, ou mesmo podem fornecer informações errôneas.



Típicas dificuldades...

Típicas dificuldades...

Considerações Iniciais



Conflitos de interesse...
VIEW POINT RESOLUTION – Focaremos mais a frente, quando falarmos
sobre validação de requisitos.
<http://www-di.inf.puc-rio.br/~julio/Slct-pub/ieee-tse.pdf>

Conflitos de interesse...

VIEW POINT RESOLUTION – Focaremos mais a frente, quando falarmos
sobre validação de requisitos.

<http://www-di.inf.puc-rio.br/~julio/Slct-pub/ieee-tse.pdf>



Elicitação de Requisitos

Subatividades

Subatividades

Subatividades, abstraindo a metodologia utilizada...

Identificação de fontes de informação

Coleta de fatos

Comunicação

Subatividades, abstraindo a metodologia utilizada...

Subatividades

- › **Faz** coleta de fatos
- › **Faz** identificação de fontes de informação
- › **Faz** comunicação
- › **Faz/Usa** ferramentas
- › **Usa** pessoal
- › **Usa** métodos
- › **Depende de** pontos de vista

Subatividades, abstraindo a metodologia utilizada...

Subatividades, abstraindo a metodologia utilizada...

Coleta de Fatos



Veremos com mais detalhes em Técnicas de Elicitação...

Veremos com mais detalhes em Técnicas de Elicitação...

Identificação das Fontes de Informação

Universo de Informação

- Contém toda informação necessária

Agentes

- Atores, Usuários

Outras fontes de informações

- Documentação do macro-sistema
- Políticas
- Manuais
- Memos, atas, contratos...
- Livros
- Outros sistemas da empresa
- Sistemas externos
- ...

Identificação das Fontes de Informação

Importante

Priorizar as fontes de informação



Principais Heurísticas:

Atores mais relevantes

Documentos mais mencionados

Rede de comunicação entre os componentes do macro-sistema

...

Comunicação



Fundamental

Atividade fundamental para que a elicitação tenha sucesso;



Trata-se da comunicação entre interessados bem como entre interessados e Engenheiros de Software



Diretrizes:

- É necessário entender o contexto, encontrando o senso comum.
- É necessária uma linguagem adequada.
- É necessário estabelecer um nível de abstração que seja capaz de viabilizar a participação ativa dos interessados.
- É necessária a retro-alimentação, refinando sucessivamente os requisitos com base nessa comunicação.





Elicitação de Requisitos

Técnicas

Técnicas

Brainstorming

Storytelling

Prototipação

Entrevista

Questionário

Observação

Introspecção

Análise de Protocolo

Análise de Discurso

Etnografia

Brainstorming



Brainstorming é uma das técnicas de reuniões em grupo mais conhecida para levantamento de requisitos.

Ela consiste em uma reunião de especialistas de diversos setores, sendo que cada componente tem a função de estimular ao outro a criação de ideias, visando à resolução do problema em questão.

Deve-se evitar criticar as ideias oriundas de um brainstorming, permitindo que os interessados participem efetivamente.

Normalmente, utiliza-se essa técnica bem inicialmente, quando pouco do produto, do processo e do projeto é conhecido.

Objetiva a geração de novas ideias, estimulando os participantes a interagir, expor suas ideias, compartilhar seus opiniões.

Uma sessão bem sucedida de Brainstorming acorda um conjunto de boas ideias, no qual os participantes sentem que cada um contribuiu de alguma maneira para a solução do problema.

Brainstorming contribui significativamente na fase de concepção de um sistema.

Storytelling



Storytelling é uma técnica que se baseia na utilização de histórias em grupos de pessoas, incluindo organizações, como um método de comunicação para que os envolvidos possam compartilhar conhecimento.

Emprega-se o uso de técnicas de contagem de histórias que inspiram e motivam os participantes através de linguagens mais cotidianas, criando entretenimento durante o processo de construção do conhecimento.

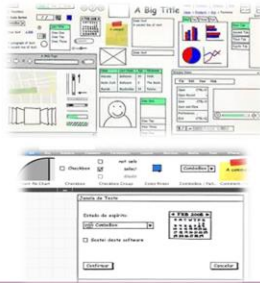
Os participantes são estimulados a narrar histórias relacionadas aos fatos, cujo conhecimento se quer elicitar. As histórias obtidas são compartilhadas com o grupo de pessoas envolvidas, o que resulta em compartilhamento de conhecimentos, agregação de novos conhecimentos e, por fim, em aprendizado.

Com formato assíncrono e distribuído, essa técnica pode fazer uso de ferramentas computacionais, permitindo maior acesso aos participantes. Isso contribui para aplicação dessa técnica mesmo quando os integrantes não estão no mesmo lugar e/ou ao mesmo tempo.

A validação da história por outros integrantes agrega valor na coleta de informações; as quais costumam ser mais autênticas, confiáveis, pois são formadas por diferentes pontos de vista.



Prototipação



A prototipação é uma técnica que permite construir um protótipo inicial do sistema proposto.

Usada para demonstrar conceitos, experimentar opções de projeto, conhecer mais sobre o problema e suas possíveis soluções.

Auxilia no desenvolvimento rápido e iterativo, permitindo que os clientes e/ou usuários possam usar o sistema o mais cedo possível.

Durante o uso, podem ser revelados erros e omissões nos requisitos propostos, podendo os usuários apresentar novos requisitos.

Possibilitam, portanto, a validação dos requisitos encontrados bem como a verificação da viabilidade do projeto proposto.

Comumente utilizados na elaboração do projeto de interfaces gráficas, permitindo a participação ativa do usuário, sendo possível acordar dificuldades e facilidades no uso do sistema. Por exemplo: detalhes quanto à complexidade dos menus e posicionamento dos elementos gráficos.

Quando o protótipo é apenas uma casca da interface, pode causar a impressão aos clientes de que o software está preparado para uso, decepcionando-os quando revela-se que esse protótipo não pode efetivamente ser utilizado.

Entrevista

A entrevista é uma técnica muito utilizada. Busca-se encontrar os requisitos ideais para o sistema.



Assim, são formuladas questões. Essas, uma vez filtradas, acordam possíveis requisitos do sistema.

interview



Basicamente, existem dois tipos de entrevistas: (i) **entrevistas fechadas**, nas quais perguntas são definidas previamente e os interessados irão respondê-las da forma que foram especificadas; e (ii) **entrevistas abertas**, nas quais há um roteiro predeterminado de questões. Esse roteiro vai sendo modificado, ao longo da entrevista, com a finalidade de obter uma maior compreensão sobre as necessidades dos interessados.

Trata-se, portanto, de uma técnica mais flexível, permitindo adquirir informações de caráter subjetivo.



Além disso, aproxima o engenheiro de requisitos do usuário do sistema, conferindo ao usuário atuação no desenvolvimento do sistema.

Custo e esforço são demandados para entrevistar inúmeros interessados, tornando-se uma dificuldade para lidar com tantos dados obtidos.

Questionário

Uma técnica capaz de lidar com grande número de pessoas. Portanto, é muito utilizada quando há indisponibilidade física, dispersão dos envolvidos ou, até mesmo, quando há necessidade de um levantamento estatístico sob qualquer aspecto envolvendo os interessados.



Pode ser utilizado para identificar problemas na organização bem como propor possíveis melhorias dada a implantação do sistema.

Criar um questionário não é uma tarefa simples. É preciso usar uma metodologia para a formulação das questões, de acordo com o perfil dos interessados que irão responder o questionário.

Planejar conteúdo, formato, ordem e valorizar simplicidade, clareza e não ambiguidade das questões são de suma relevância na elaboração de um questionário.



Prever antecipadamente e dúvidas que podem surgir e evitar questionários longos e cansativos complementam as diretrizes para a boa elaboração de um questionário.



Por trabalhar com muitas pessoas, trata-se de uma técnica mais inflexível, pois impossibilita a análise de questões subjetivas, as quais costumam ser de grande valia para o pleno entendimento dos problemas.

É possível montar questionários com perguntas discursivas, procurando explorar essa análise de aspectos mais subjetivos. Entretanto, tais questionários não poderão ser aplicados para muitos interessados. Adicionalmente, não terão suas análises automatizadas, dado que seria necessário recorrer a técnicas de análise em linguagem natural; sendo essas muito mais complexas e custosas.



Observação



A observação é uma técnica que possibilita observar, no próprio ambiente, no qual o software será implantado, as tarefas sendo executadas pelos interessados, sem interferir no ambiente.

Obtem-se os requisitos com base nas tarefas realizadas pelos interessados.

A etnografia, a qual será abordada em breve, é vista como uma técnica de observação usada para compreender requisitos sociais e organizacionais implícitos de um sistema. Nesse caso, observa-se o dia-a-dia da organização, visando conhecer e anotar as tarefas dos envolvidos.

Por fim, existe o conceito de Observação Participativa. Aqui, cabe interferência do Engenheiro de Software no ambiente em estudo.



Introspecção

Introspecção é uma técnica muito rica e profunda. Consiste em entender quais propriedades o sistema deve possuir para que seja um sucesso. Demanda o Engenheiro de Requisitos imaginar o que ele gostaria, se ele tivesse que desempenhar uma dada tarefa, com os equipamentos disponíveis e demais recursos.

Pode ser muito útil, mas tem o problema de que a introspecção de um especialista, de um área de atuação diferente, não ser adequada para refletir a experiência/atuação real dos interessados.



Análise de Protocolo

Análise de protocolo solicita ao sujeito que engaje em alguma tarefa e concorrentemente, em voz alta, explique o processo que realiza.

Seria uma espécie de verbalização direta de um processo cognitivo específico.

Permite analisar algo sob um dado ponto de vista.

Pode ser bastante útil, se bem aplicada.

Mas, pode também causar timidez no orador.

A extração de registro de tarefas pode ainda ser via vídeo ou notas escritas, além do uso do áudio.

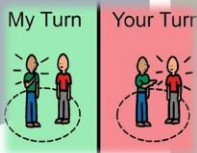


Análise de Discurso



Pode ser realizada com:
análise de conversação,
turntaking e pares adjacentes.

Análise de conversação é
algo base para permitir análise
em um grupo, interagindo, se
socializando. Os turnos das
conversas não são pré-
estabelecidos.



Turn Taking



Turntaking estabelece uma
conversação orientada à
criação de uma ordem social.
Deve ter um orador por turno,
o qual não deve ser
interrompido. Após uma pausa
mais longa, tempo ou outro
critério, o turno é mudado,
dando espaço a outro orador.
Permite conhecer diferentes
pontos de vista, identificar
conflitos de interesse e outros
ganhos, quando bem
aplicada.

Pares adjacentes é uma
conversação envolvendo
atuantes de dois lados. O
típico modelo de conversa
baseado em pergunta-
resposta é um exemplo desse
tipo de conversação. Trata-se
de uma forma mais
organizada para prover uma
conversação.



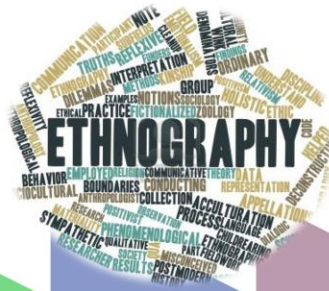
Etnografia

É uma técnica das ciências sociais que se mostrou bastante útil no entendimento dos processos reais, realizados nas organizações.

Visão mais completa e ajustada ao contexto.

Demanda tempo e há pouca sistematização do processo.

Foco na combinação de observação e entrevistas abertas, permitindo conhecer os envolvidos.

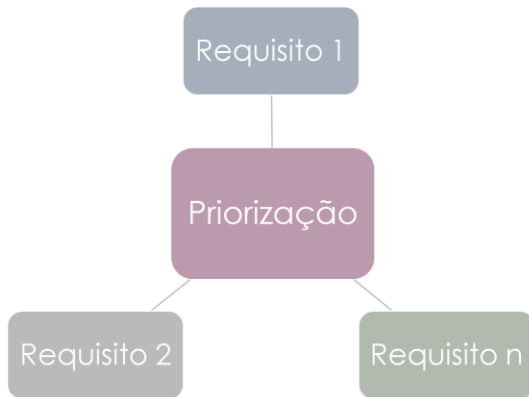




Elicitação de Requisitos

Priorização

Priorização



Dado que são elicitados vários requisitos, para saber por onde começar o desenvolvimento, de forma a gerar, desde o começo, valor agregado aos clientes a cada entrega, torna-se necessário priorizar esses requisitos.

Priorização

Como atuar?



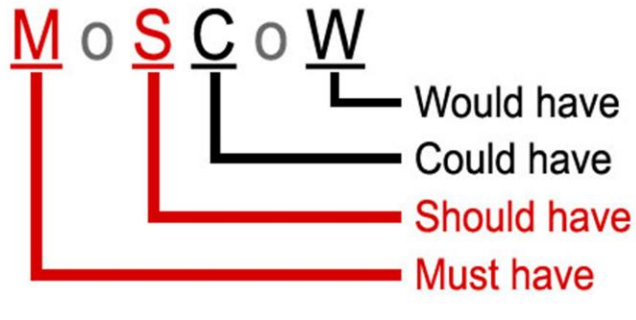
Existem várias formas de estabelecer a prioridade dos requisitos.

Mas, normalmente, são utilizadas tabelas, nas quais os requisitos são categorizados em sendo de: Alta Prioridade, Média Prioridade e Baixa Prioridade.

Lembrando que essa categorização não é tomada com base exclusivamente na percepção do Engenheiro de Software. Principalmente, devem ser levadas em conta as expectativas dos clientes.

Os entregáveis precisam ter valor agregado aos clientes...

MoSCoW



MoSCoW - Técnica muito conhecida para priorização de requisitos.

Exemplo de técnica simples...

Técnica simples...

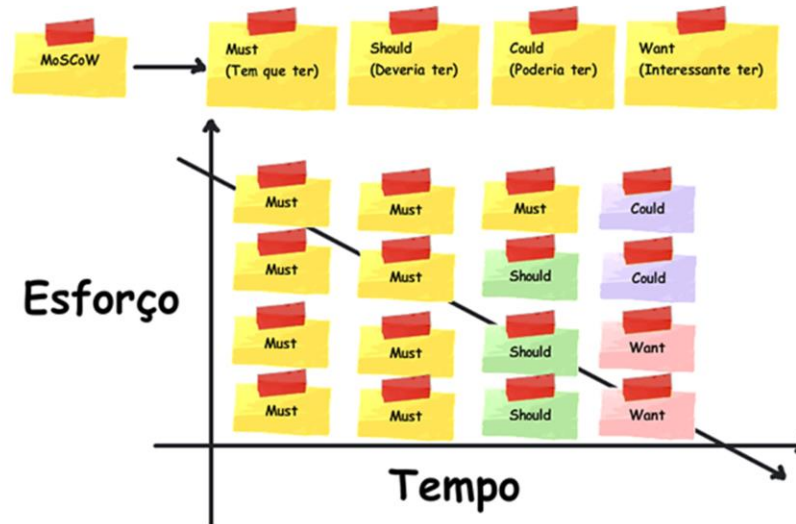
MoSCoW

Must	Should	Could	Would / Want / Won't
• Prioritários/Críticos para o negócio, como: (i) que atendem norma legal (ex: normativas da bolsa de valores, banco central, agências do governo e Receita Federal); e (ii) que afetam a imagem e credibilidade da empresa. Se um dos itens não é concluído e entregue, o projeto não pode ser considerado como sendo concluído com sucesso.	• Importantes, mas não são necessários (do ponto de vista estratégico) para entrega neste momento. Pode-se ter outro meio de se atender a necessidade classificada como SHOULD, ou pode-se esperar um pouco para ser trabalhado (ex: a conclusão de outro projeto em andamento reduzirá ou eliminará a necessidade do projeto demandado).	• Desejáveis, mas não são necessários (do ponto de vista estratégico) e podem melhorar a satisfação do cliente com algum esforço de desenvolvimento. Estes itens, geralmente, podem ser atendidos dada a disponibilidade de tempo e recursos.	• Menos críticos, com menor retorno sobre o investimento ou não adequados para serem realizados. Decisões que devem ter a concordância dos clientes. Pode ser usado "Would Like (Gostaria)" para conferir melhor entendimento.

A técnica é muito simples: para cada requisito ou item do seu *backlog*, você deverá atribuir uma das quatro letras M, S, C ou W; e cada uma delas tem um significado diferente.

Técnica simples...

MoSCoW



Técnica simples...

MoSCoW



PRODUCT BACKLOG									
Prioridade	ID	Descrição	Comentários	Valor de Negócio	Size	Points	Sprint	Status	
Alta	1	Como um usuário do módulo de contas a pagar gostaria de consultar todas as contas a pagar com filtro por data para facilitar e agilizar os pagamentos	Alto valor de negócio para o cliente, pois facilita o dia a dia do financeiro	1 - Must Have	1 - Small	?	1	Fila	
Média	2	Como um usuário do módulo de contas a pagar gostaria de emitir via sistema um relatório consolidado de todas as contas a pagar por data para enviá-lo ao Diretor Financeiro		2 - Should Have	1 - Small	?	1	Fila	
Média	3	Como um usuário do módulo de contas a pagar gostaria de visualizar gráficos de pizza dos balanços de pagamentos para demonstração em apresentações de status	Seria bom ter esta funcionalidade, mas podemos esperar para implementá-la, pois demonstrações deste tipo acontecem um vez a cada semestre	3 - Could Have	1 - Small	?	2	Fila	
Baixa	4	Como usuário do módulo de contas a pagar gostaria que me fosse enviado automaticamente um email de alerta toda vez que uma conta a pagar estiver faltando 3 dias para vencer para evitar atrasos nos pagamentos	Como alternativa, o usuário poderá extrair relatórios diários (estória ID01) para saber as contas que estão para vencer	4 - Won't Have for Now	1 - Small	?		Fila	

Técnica simples...

MoSCoW

Dis advantage

Vantagens

- Fácil compreensão.
- Fácil de se trabalhar.
- Linguagem simples.
- Permite a participação de todos os interessados.
- Não requer conhecimento prévio da técnica.

- Alguns critérios podem ser ignorados.
- Subjetividade quanto às priorizações de cada interessado.
- Encoraja decisões com viés mais político do que técnico.
- É necessário que o interessado conheça de fato o negócio.

Desvantagens

Técnica simples...

First Things First: Prioritizing Requirements



Envolvidos/Participantes

Gerente, liderando o processo, arbitrando conflitos e ajustando as entradas dos participantes.

Representantes dos clientes, classificando os benefícios e as fragilidades.

Representantes de desenvolvimento, liderando a parte técnica com avaliação de custos e riscos.



É importante:

Equilibrar os benefícios de cada função contra seus custos.

Definir as implicações que serão acarretadas na arquitetura.

Alinhar requisitos e regras de negócio.

Estabelecer risco técnico e demais dificuldades associadas a um dado requisito.

Estabelecer a granularidade na qual priorizar os requisitos. Não é recomendado que a lista de requisitos ultrapasse a casa das dezenas.

Karl E. Wiegers. "First Things First: Prioritizing Requirements". Software Development, September 1999.

<http://www.processimpact.com/articles/prioritizing.html>

Exemplo de técnica mais elaborada...

Técnica mais elaborada...

First Things First: Prioritizing Requirements

Passo 1. Faça uma lista de todos os requisitos em uma planilha. Se algo está logicamente ligado (ou seja, só é possível implementar B se A for implementada), incluir apenas o recurso A na análise.

Passo 2. Estime o benefício relativo que cada recurso fornece ao cliente ou ao negócio em uma escala de 1 a 9, onde 1 é o menos significativo e 9 o máximo. É fundamental a participação dos representantes dos clientes.

Passo 3. Estime a penalidade que o negócio sofreria, se o recurso não for incluído. Usar escala de 1 a 9, onde 1 significa nenhuma penalidade e 9 indica uma grande desvantagem. Por exemplo, deixar de cumprir uma regulamentação do governo pode ser uma grande penalidade, mesmo que o benefício para o cliente seja baixo.

Passo 4. A coluna Valor Total é a soma do (Benefício Relativo * Peso Relativo) e da (Penalidade Relativa * Peso Relativo). Caso seja necessário, os pesos – em um primeiro momento iguais – podem ser ajustados.

Passo 5. Estime o custo relativo de implementação de cada característica, em uma escala que varia de um mínimo de 1 a um máximo de 9. Os desenvolvedores devem classificar os custos com base em fatores como a complexidade de implementação, a interface de usuário necessária, a capacidade potencial de reutilização de telas ou código, e os níveis de testes e documentação necessários.

Passo 6. Estime o grau relativo de risco a cada requisito em uma escala de 1 a 9. Uma estimativa de 1 significa que eles podem "programar dormindo", enquanto que 9 indica sérias preocupações sobre a viabilidade, a disponibilidade de pessoal com o conhecimento necessário, ou o uso de ferramentas e tecnologias desconhecidas. Custo e risco são ponderados igualmente. Entretanto, esses pesos podem ser ajustados.

Passo 7. Calcule a prioridade para cada requisito usando Prioridade igual a:

$$\text{valor \%} / (\text{custo \%} * \text{Peso custo} + \text{riscos \%} * \text{Peso Risco})$$

Passo 8. Ordene a lista em ordem decrescente de prioridade. As características no topo da lista têm o melhor equilíbrio entre valor, custo e risco, e, portanto, devem ser priorizadas. Deve-se rever visando saber se os ratings e a sequência resultante não violam algo mais.

Técnica mais elaborada...

First Things First: Prioritizing Requirements

Peso Relativo	2	1			1		0,5		
Funcionalidades	Benefício Relativo	Penalidade Relativa	Valor Total	Valor %	Custo Relativo	Custo %	Risco Relativo	Risco %	Prioridade
A	5	3	13	8,4	2	4,8	1	3	1,345
B	9	7	25	16,2	5	11,9	3	9,1	0,987
C	5	5	15	9,7	3	7,1	2	6,1	0,957
D	2	1	5	3,2	1	2,4	1	3	0,833
E	4	9	17	11	4	9,5	4	12,1	0,708
F	4	3	11	7,1	3	7,1	2	6,1	0,702
G	6	2	14	9,1	4	9,5	3	9,1	0,646
H	9	8	26	16,9	7	16,7	8	24,2	0,586
I	3	4	10	6,5	4	9,5	2	6,1	0,517
J	7	4	18	11,7	9	21,4	7	21,2	0,365
Totais	54	46	154	100	42	100	33	100	

Representantes dos Clientes

Equipe de Desenvolvedores

Técnica mais elaborada...

First Things First: Prioritizing Requirements

Stakeholders	Sales					Marketing					Implementation Team		
Feature Name	Relative Benefit	Relative Penalty	Volatility	Urgency	Criticality	Relative Benefit	Relative Penalty	Volatility	Urgency	Criticality	Implementation Effort	Implementation Confidence	Dependency Risk
Customizable Interface													
Custom Contact Attributes													
Multi-Lingual													
Email Marketing													
Time-Tracking													
Pipeline Tracking													
Lead Tracking													
Email Integration													
Calendar Integration													
Campaign Management													
Desktop Access													
Web-Based Access													
Mobile Access													
Overall Ease of Use													
Project Management													
Task Management													
Workflow Automation													
Analytics Engine													

Técnica mais elaborada...

ROI - Return On Investment

$$\text{ROI} = \text{Benefício/Custo}$$

		Valor			Valor total	Valor percentual (Benefício)	Estimativa	Custo percentual (Custo)	Prioridade (ROI)
		Aumento da satisfação do cliente	Aumento das vendas	Aumento de visitantes					
Épicos ou temas	Integração com mídias sociais	2	7	10	19	31,67%	13	18%	177,82%
	Alerta de preço desejado	8	7	6	21	35,00%	20	27%	127,75%
	Compra com um clique	6	9	5	20	33,33%	40	55%	60,83%
	Total				60	100%	73	100%	

Exemplo de técnica mais elaborada...

Técnica mais elaborada...

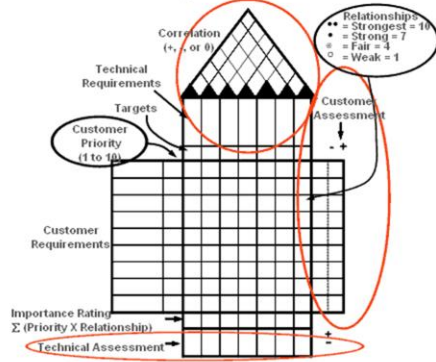
Técnicas mais Avançadas



Exemplos de técnicas bem mais elaboradas...

Técnicas bem mais elaboradas...

QFD Matrix



Quality Function Deployment

Correlation	
++	Strong positive
+	Positive
-	Negative
--	Strong negative

Correlation used to show how items within a list affect one another

Relationship		
Strong	⊙	9
Medium	○	3
Weak	△	1

Customer-
assigned value
from 1 to 9
used to weight
relationship
value

Matrix extended to score competitors against customer requirements

*Compressed Tree Diagram
used to identify customer
requirements*

Attractive table setting	Clean	Cloth	8	⊙				⊗	Δ	○	176	14	9	4	9		
		Cutlery	9		⊙				⊗	Δ	○	198	16	7	6	9	
	Elegant	Cloth	6					⊗	○	○	○	126	10	2	4	5	
		Decor	7		⊗	⊗	○	○	○	⊗	⊗	294	23	6	7	5	
		Furniture	4				Δ			⊗	○	52	4	4	8	4	
	Correct	Layout	5									⊗	180	14	6	8	6
		Cutlery	8			Δ	⊗	⊗	Δ		⊗	232	18	3	8	9	

Relationship x Priority
totals for columns
show importance of
each item in satisfying
customer requirements

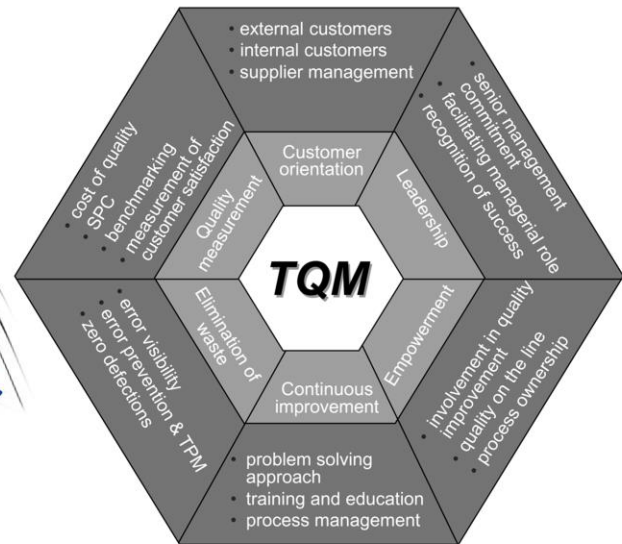
A percentage makes Importance easier to interpret and to carry forward

Relationship x Priority totals for rows shows attention given to customer requirements

Customer requirements for table-setting versus effort in restaurant procedures

Por curiosidade...
Quality Function Deployment

TQM



Source: Silvestro, 1998

Por curiosidade...
Total Quality Management



Considerações Finais

Considerações Finais

- › Nessa aula, foi apresentada a atividade de elicitación de requisitos, com detalhamento quanto às:
 - subatividades, abstraindo a escolha de uma metodologia específica.
 - técnicas, e
 - questões de priorização.
- › Continuem os estudos!



Referências

Referências

Bibliografia Básica

1. [Ebrary] Young, Ralph. **Requirements Engineering Handbook**. Norwood, US: Artech House Books, 2003.

2. [Open Access] Leite, Julio Cesar Sampaio do Prado. **Livro Vivo - Engenharia de Requisitos**. <http://livrodeengenhariaderequisitos.blogspot.com.br/> (último acesso: 2017)

3. [Ebrary] Chemuturi, Murali. **Mastering Software Quality Assurance : Best Practices, Tools and Technique for Software Developers**. Ft. Lauderdale, US: J. Ross Publishing Inc., 2010.

4. **Software & Systems Requirements Engineering: In Practice** - Brian Berenbach, Daniel Paulish, Juergen Kazmeier, Arnold Rudorfer (Livro bem completo mas, não tem exemplar físico na biblioteca, nem mesmo consta na Ebrary)

5. **Requirements Engineering and Management for Software Development Projects** - Murali Chemuturi (Livro bem completo mas, não tem exemplar físico na biblioteca, nem mesmo consta na Ebrary)

Referências

Bibliografia Complementar

1. [BIBLIOTECA – 15 exemplares] Pfleeger, Shari Lawrence. Engenharia de Software: Teoria e Prática. 2ª. Edição. São Paulo: Prentice Hall, c2004. xix, 535 p. ISBN 978858791831

2. [BIBLIOTECA – 3 exemplares] Withall, Stephen. Software Requirement Patterns. Redmond: Microsoft Press, c2007. xvi, 366 p. ISBN 978735623989.

3. [BIBLIOTECA - vários exemplares] Leffingwell, 2011, Agile Software Requirements, <http://www.scaledagileframework.com/> (último acesso: 2017)

4. [Ebrary] Evans, Isabel. Achieving Software Quality Through Teamwork. Norwood, US: Artech House Books, 2004.

5. [Ebrary] Yu, Eric, Giorgini, Paolo, and Maiden, Neil, eds. Cooperative Information Systems: Social Modeling for Requirements Engineering. Cambridge, US: MIT Press, 2010.

6. [Open Access] Slides disponíveis em: <https://www.wou.edu/~eltonm/Marketing/PP%20Slides/> (último acesso: 2017)



Dúvidas?

Orientações?

Sugestões?

FIM

mileneserrano@unb.br ou mileneserrano@gmail.com
serrano@unb.br ou serr.mau@gmail.com