



Processo de Engenharia de Requisitos

Engenharia Software

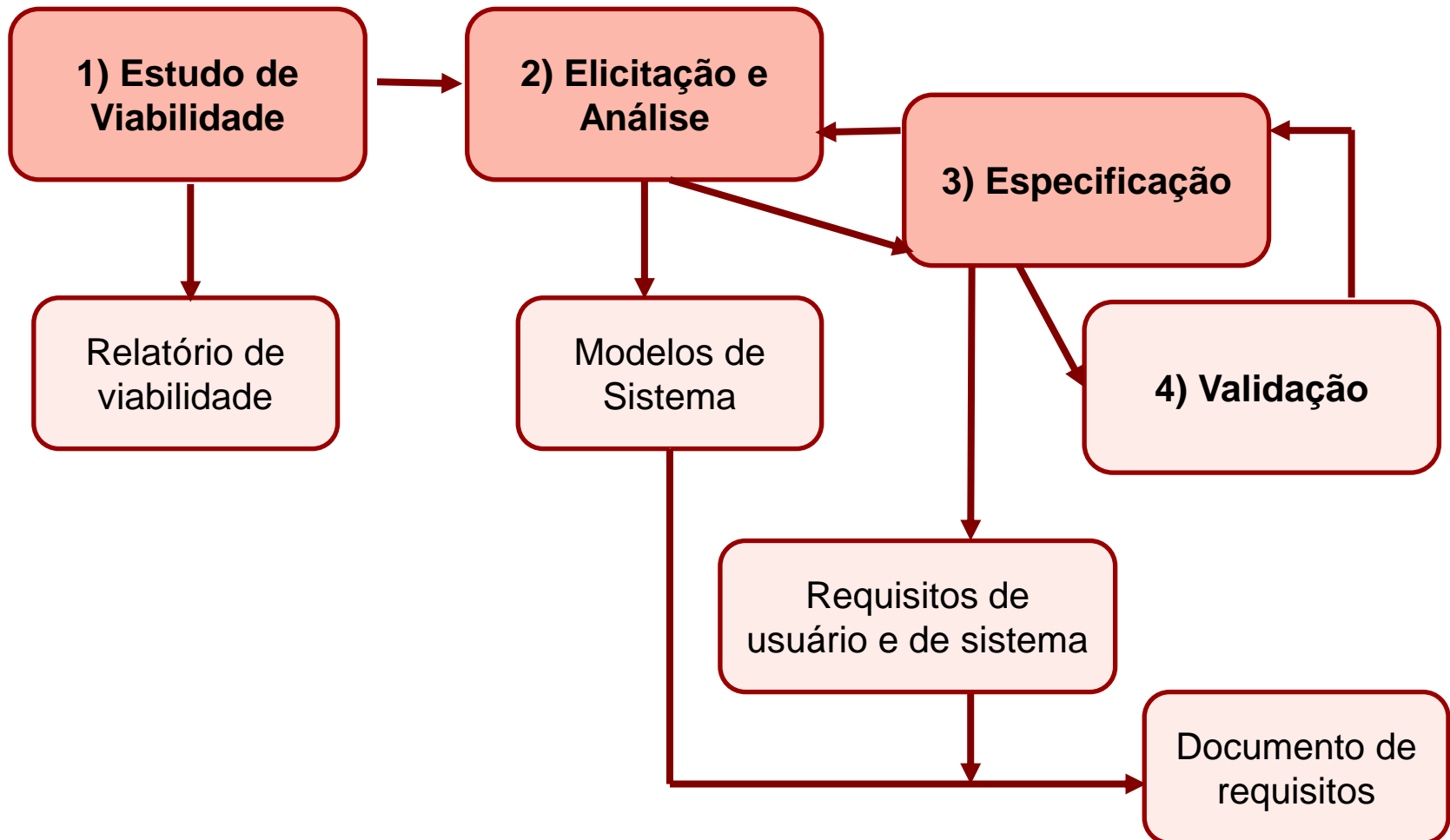
Prof. Me. Emerson Ap. Mouco Jr.

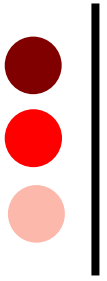


Introdução

- O Processo de Engenharia de Requisitos apresenta as quatro etapas:
 - estudo de viabilidade;
 - elicitação (levantamento) e análise de requisitos;
 - especificação de requisitos;
 - validação de requisitos

Processo de Engenharia de Requisitos – Visão de Sommerville





1) Estudo de viabilidade



Estudo de viabilidade

- Tem por objetivo obter uma estimativa sobre a possibilidade das necessidades do usuário serem atendidas usando-se tecnologias atuais de software e hardware.
- É analisada a rentabilidade do sistema proposto, a partir do ponto de vista do negócio, e se o sistema pode ser desenvolvido com o orçamento disponível.



Estudo de viabilidade

- O estudo de viabilidade é um estudo curto e focalizado que deve ser feito no início do processo da Engenharia de Requisitos.
- Ele deve responder três perguntas fundamentais:
 1. O sistema contribui para os objetivos globais da organização?
 2. O sistema pode ser implementado dentro do cronograma e orçamento usando a tecnologia atual?
 3. O sistema pode ser integrado com outros sistemas utilizados?



Estudo de viabilidade

- Se a resposta a qualquer uma dessas perguntas for não, provavelmente não se deve prosseguir com o projeto.
- Além disso, é importante analisar também:
- É mais viável desenvolver um novo sistema ou utilizar alguma tecnologia existente no mercado?
- Quais os riscos envolvidos?
- Qual será o diferencial do sistema para os outros já existentes no mercado?
- Qual o real problema do cliente? O sistema a ser desenvolvido irá resolver **O PROBLEMA** do cliente?



Tipos de viabilidade

- **Viabilidade organizacional:** tem relação aos benefícios que o software está trazendo à organização, como a aderência dos usuários e alinhamento com os objetivos estratégicos da organização;
- **Viabilidade operacional:** verifica se o sistema está de acordo com o esperado;
- **Viabilidade econômica:** análise entre o custo de desenvolvimento e os benefícios após implementação do projeto (custo-benefício).



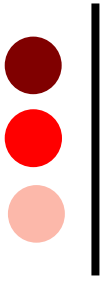
Tipos de viabilidade

- **Viabilidade técnica:** está relacionada ao suporte técnico que a organização oferecerá para o desenvolvimento do projeto; restrições da equipe ou da tecnologia; necessidade de se investir em pesquisas antes de realizar o projeto; etc.
- **Viabilidade de cronograma:** cruzamento entre as atividades levantadas e o tempo estimado para realizá-las; definição de marcos do projeto; impacto de atrasos.
- **Outras viabilidades:** legal, cultural, marketing, etc.



Relatório de Viabilidade

- O relatório de viabilidade tem o objetivo de documentar a avaliação do projeto sob os aspectos: operacional, técnico, econômico e organizacional.
- O documento deve conter:
 1. Introdução
 2. Objetivo
 3. Escopo
 4. Diagnóstico atual
 5. Requisitos
 6. Alternativas Propostas
 7. Alternativa Recomendadas
 8. Cronograma
 9. Conclusão

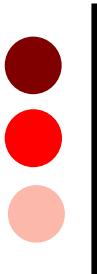


2) Elicitação / Levantamento e Análise



Elicitação/Levantamento e Análise

- É responsável pela derivação dos requisitos do sistema por meio da observação dos sistemas existentes, além de discussões com os potenciais usuários e compradores e análise de tarefas.
- São desenvolvidos os modelos do sistema e os protótipos, visando ao entendimento do sistema especificado



Levantamento de Requisitos (Visão Pressman)

- Os analistas de sistemas realizam um estudo detalhado dos requisitos levantados na fase anterior (concepção).
- A partir desse estudo, são construídos modelos para representar o software a ser desenvolvido.
- Nesta etapa são produzidos alguns artefatos que auxiliam nessa tarefa.



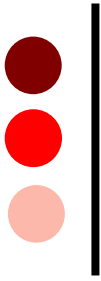
Artefatos produzidos

- Declaração da necessidade e da viabilidade;
- Definição do escopo do sistema ou do produto;
- Lista de clientes, usuários e outros interessados (*stakeholders*) que participaram do levantamento de requisitos;
- Descrição do ambiente técnico do sistema;
- Lista de requisitos (preferencialmente organizada por função) e as restrições de domínio que se aplicam a cada um deles;
- Conjunto de cenários de uso, que fornecem informações sobre o uso do sistema;
- Protótipos desenvolvidos para melhor definição dos requisitos, que também podem ser decorrentes da análise dos requisitos.

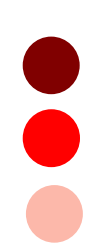


Análise de Requisitos

- Há várias técnicas e notações para se definir um modelo de requisitos, no paradigma orientado a objetos, tem se destacado a linguagem UML,



2.1 - Técnicas de Levantamento de Requisitos



Técnicas de Levantamento de Requisitos

- As principais técnicas de Levantamento de requisitos são:
 - Pesquisa
 - Entrevista
 - Questionário
 - Observação
 - Seminário



Pesquisa

- Verificar fisicamente os acontecimentos.
- Evita frequências de erros;
- Identifica quais os grandes problemas;
- Esclarece dúvidas;



Entrevista

- Um dos melhores meios de levantamento de dados (analista e usuário).
- Para uma boa entrevista:
 - Deixar claro os objetivos da entrevista;
 - Planejar local, ambiente, duração;
 - Preparar um roteiro dos assuntos abordados;
 - Adequar a entrevista ao nível do entrevistado;
 - Planejar as perguntas;
 - Escolher os entrevistados (pessoas chaves, de fácil acesso, motivadas a cooperar);



Dicas

- Entrevistar primeiramente o “chefe” do departamento;
- Conquistar a confiança do entrevistado;
- Usar perguntas simples;
- Ouvir mais do que falar;
- Anotar as informações coletadas;
- Confirmar as informações no final da entrevista.





O que perguntar durante um entrevista

Ex.: Vídeo Locadora.

- Explique o processo atual de locação de fitas/vídeos. O processo com o sistema seria diferente?
- Dúvidas: Haverá jogos ou DVDs disponíveis pra locação também?
- Dúvidas: Como seria o processo de locação de jogos? É igual de fitas?
- Dúvidas: Os DVDs são tratados de forma diferente das fitas? Como serão tratados?
- Haverá possibilidade de haver datas de devolução diferentes para cada fita, numa mesma locação?
- Quantos dígitos deverá ter no código de uma fita? O código pode ajudar na classificação ou localização?

Questionários

- Preparado em formulários, distribuídos aos usuários e recolhido posteriormente.
- Geralmente combinado com outras técnicas.
- Ex: Aos atendentes da locadora.



Cargo:

Data:

1. Descreva os passos para se locar um fita.

2. Quais as informações importantes sobre uma fita?

3. Quantos clientes você atende por dia na locadora?

4. Quantas fitas você acredita que a locadora tem hoje?

5. Em que atividade do seu trabalho, você acredita que um sistema computacional poderia ajudar?



Observação ou Etnografia

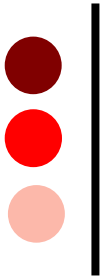
- Etnografia é uma técnica de observação usada para compreender os processos operacionais e ajudar extrair requisitos de apoio aos processos.
- Auxilia para confirmar as informações recebidas/coletadas;
- Esta técnica não interrompe o trabalho das pessoas envolvidas;
- Basta observar como as pessoas trabalham e participam do processo da empresa.



Seminário



- É uma reunião com as pessoas envolvidas no PDS;
- É necessário um planejamento quanto à convocação dos usuário chave (agendamento, horário, local);
- Elaborar a documentação (ata);
- O analista é o mediador; (fechamentos e acordos firmados)

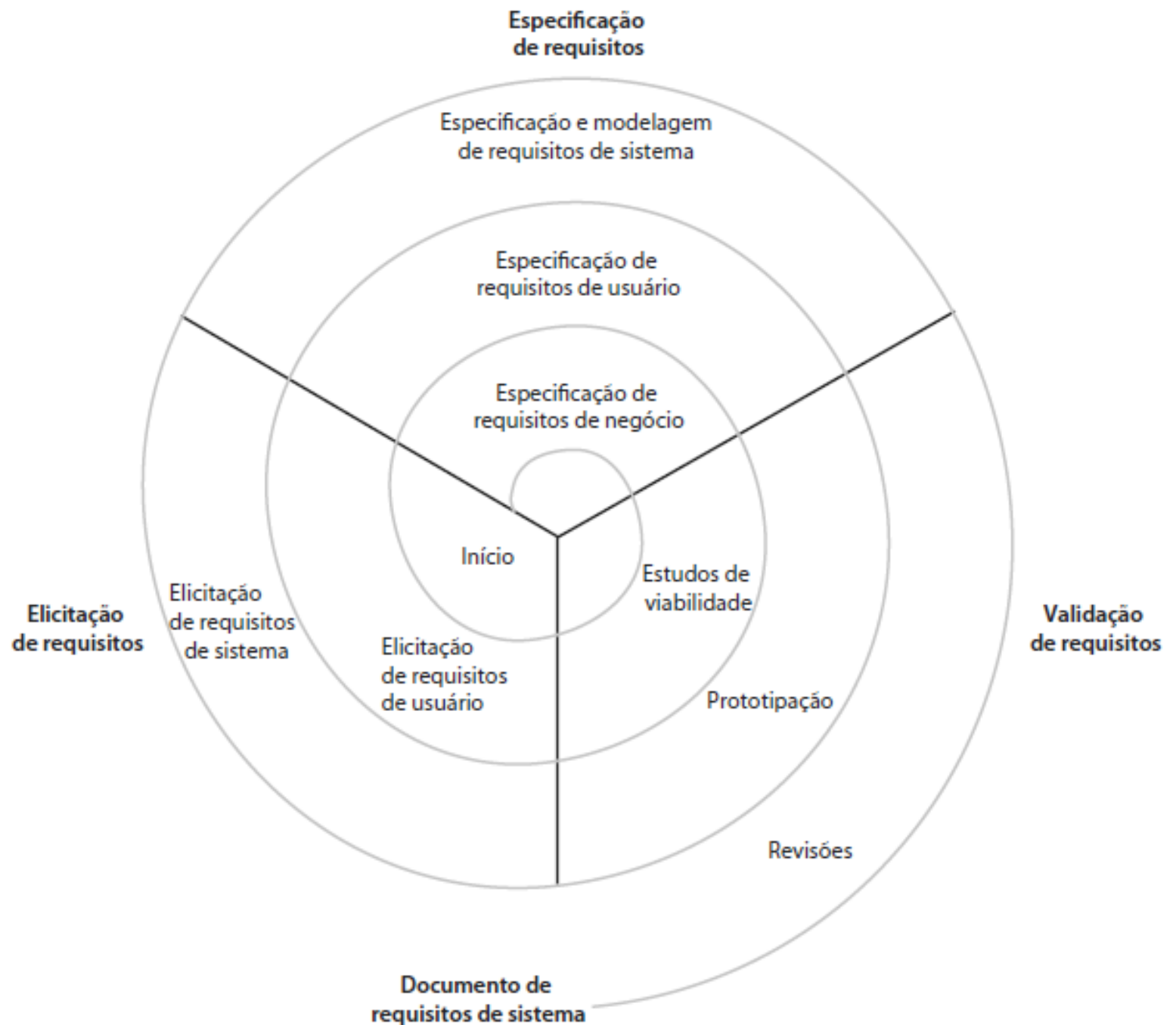


3) Especificação



3) Especificação

- Traduz as informações obtidas durante a atividade de análise em um documento que defina um conjunto de requisitos.
- Pode ser um documento escrito, um conjunto de modelos gráficos, um modelo matemático formal, um modelo de cenários de uso, um protótipo, etc.



Exemplo – Pressman p. 129

INFORMAÇÕES



Modelo de especificação de requisitos de software

Uma especificação de requisitos de software (*software requirements specification*, SRS) é um documento criado quando uma descrição detalhada de todos os aspectos do software a ser construído deve ser especificada antes de o projeto começar. É importante notar que uma SRS formal nem sempre é por escrito. Na realidade, há várias ocasiões em que o esforço gasto em uma SRS talvez fosse mais bem aproveitado em outras atividades de engenharia de software. Entretanto, quando um software for desenvolvido por terceiros, quando uma falta de especificação criar graves problemas de negócio, ou quando um sistema for extremamente complexo ou crítico para o negócio, será justificável uma SRS.

Karl Wiegers [Wie03] da Process Impact Inc. desenvolveu uma planilha bastante útil (disponível em www.processimpact.com/process_assets/srs_template.doc), que pode servir como diretriz para aqueles que precisam criar uma SRS completa. Uma descrição geral por tópicos é apresentada a seguir:

Sumário

Histórico de revisão

1. Introdução

- 1.1 Propósito
- 1.2 Convenções do documento
- 1.3 Público-alvo e sugestões de leitura
- 1.4 Escopo do projeto
- 1.5 Referências

2. Descrição geral

- 2.1 Perspectiva do produto
- 2.2 Características do Produto

- 2.3 Classes de usuários e características
- 2.4 Ambiente operacional
- 2.5 Restrições de projeto e implementação
- 2.6 Documentação para usuários
- 2.7 Hipóteses e dependências

3. Características do sistema

- 3.1 Características do sistema 1
- 3.2 Características do sistema 2 (e assim por diante)

4. Requisitos de interfaces externas

- 4.1 Interfaces do usuário
- 4.2 Interfaces de hardware
- 4.3 Interfaces de software
- 4.4 Interfaces de comunicação

5. Outros requisitos não funcionais

- 5.1 Necessidades de desempenho
- 5.2 Necessidades de proteção
- 5.3 Necessidades de segurança
- 5.4 Atributos de qualidade de software

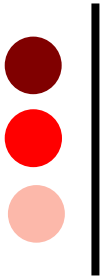
6. Outros requisitos

Apêndice A: Glossário

Apêndice B: Modelos de análise

Apêndice C: Lista de problemas

Uma descrição detalhada de cada tópico SRS pode ser obtida fazendo-se o download da planilha SRS na URL citada anteriormente neste quadro.



4) Validação



4) Validação

- Verifica os requisitos quanto ao realismo, consistência e completude.
- Os erros no documento de requisitos são inevitavelmente descobertos, e o documento deve ser ajustado para correção dos problemas encontrados.
- Ela se sobrepõe à análise, uma vez que está preocupada em encontrar problemas com os requisitos.



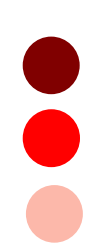
Tipos de validação

- **Verificações de validade:** é pensar como o usuário vai identificar as funções necessárias, adicionais ou diferentes. Como as funções irão atender os diversos stakeholders com diferentes necessidades
- **2. Verificações de consistência:** requisitos no documento não devem entrar em conflito, ou seja, não deve haver restrições contraditórias ou descrições diferentes da mesma função do sistema.
- **3. Verificações de completude:** o documento de requisitos deve incluir requisitos que definam todas as funções e as restrições pretendidas pelo usuário do sistema.



Tipos de validação

- **Verificações de realismo:** consideram o orçamento e o cronograma para o desenvolvimento do sistema.
- **5. Verificabilidade:** um conjunto de testes que demonstrem que o sistema será entregue e atende a cada requisito especificado.



Técnicas de validação de requisitos

- 1. Revisões de requisitos:** é a análise sistemática dos requisitos por uma equipe de revisores que verifica erros e inconsistências.
- 2. Prototipação:** é um modelo executável do sistema para demonstrar aos usuários finais e clientes. Estes podem experimentar o modelo para verificar se ele atende a suas reais necessidades.
- 3. Geração de casos de teste:** Os requisitos devem ser testáveis, ou seja, é realizado um desenvolvimento de testes a partir dos requisitos do usuário antes de qualquer código a ser escrito, como acontece com a metodologia ágil XP - *Extreme Programming*.



Exemplo - Mentcare

- **Definição dos requisitos do usuário:**

- 1- O sistema Mentcare deve gerar relatórios de gestão mensais, mostrando o custo dos medicamentos prescritos por cada clínica naquele mês.



Exemplo - Mentcare

○ Definição dos requisitos do sistema:

- No último dia útil de cada mês, deve ser gerado um resumo dos medicamentos prescritos, seu custo e a clinica que os prescreveu.
- O sistema deve gerar o relatório para impressão após as 17h30 do último dia útil do mês.
- Deve ser criado um relatório para cada clinica, listando o nome de cada medicamento, a quantidade total de prescrições, a quantidade de doses prescritas e o custo total dos medicamentos prescritos.
- Se os medicamentos estiverem disponíveis em dosagens diferentes (por exemplo, 10 mg, 20 mg etc) devem ser criados relatórios diferentes para cada dosagem.
- O acesso aos relatórios de medicamentos deve ser restrito aos usuários autorizados, conforme uma lista de controle de acesso produzida pela gestão.



Exemplo - Mentcare

○ Exemplos de stakeholders envolvidos:

- ✓ pacientes cujas informações estão registradas no sistema e familiares desses pacientes;
- ✓ médicos responsáveis por avaliar e tratar os pacientes;
- ✓ profissionais de enfermagem que coordenam as consultas com os médicos e administram alguns tratamentos;
- ✓ recepcionistas que marcam as consultas dos pacientes;
- ✓ equipe de TI responsável pela instalação e manutenção do sistema;



Tipos de requisitos

○ Requisitos funcionais

- São **declarações** que o serviço deve fornecer, o modo que o sistema deve reagir a determinadas entrada de dados do cliente. Em alguns casos também deve declarar o que o sistema **NÃO DEVE FAZER**.



Tipos de requisitos

○ Exemplos – Requisitos funcionais

- Um usuário deve poder fazer uma busca na lista de consultas de todas as clínicas.
- O sistema deve gerar, a cada dia e para cada clínica, uma lista de pacientes que devam comparecer às consultas naquele dia.
- Cada membro da equipe que utiliza o sistema deve ser identificado exclusivamente por seu número de funcionário de oito dígitos.



Tipos de requisitos

○ Requisitos não funcionais

- São **restrições** sobre os serviços que o sistema e/ou funções possui. Restrições de tempo, processos ou padrões. **Normalmente não se aplicam ao sistema todo**, mas a características individuais ou serviços.



Tipos de requisitos

○ Exemplos – Requisitos não funcionais

- Requisitos do produto
- Requisitos organizacionais
- Requisitos externos



Tipos de requisitos

○ Requisitos de produto

- O sistema deve ficar disponível para todas as clínicas durante o expediente normal (08:30 às 17:30 – de segunda a sexta).
- O tempo máximo que o sistema pode ficar fora do ar durante o expediente é 5 minutos!



Tipos de requisitos

○ Requisitos organizacionais

- Os usuários do sistema devem se identificar utilizando cartão de identificação fornecido pelo cadastro no sistema.



Tipos de requisitos

○ Requisitos externos

- O sistema deve implementar segurança para a privacidade do paciente, manter os dados em sigilo.



Métricas para requisitos não funcionais

Propriedade	Métrica
Velocidade	Transações processadas por segundo
	Tempo de resposta do usuário/evento
	Tempo de atualização da tela
Tamanho	Tamanho do projeto em Megas
Facilidade de uso	Tempo de treinamento
	Número de quadros de ajuda
Confiabilidade	Tempo médio até a falha
	Probabilidade de indisponibilidade
	Taxa de ocorrência de falhas
	Disponibilidade (SLA)



Métricas para requisitos não funcionais

Propriedade	Métrica
Robustez	Tempo para iniciar após a falha
	Porcentagem de eventos causados por falha
	Probabilidade de corromper os dados em caso de falha
Portabilidade	Porcentagem de declarações dependentes do sistema-alvo
	Número de sistemas-alvo



Entrevistas

○ Entrevistas quantitativas

- Perguntas fechadas
- Normalmente utiliza-se escalas
- Ex: Sim ou Não, Totalmente insatisfeito até Totalmente satisfeito etc.



Entrevistas

○ Entrevistas qualitativas

- Perguntas abertas
- Visam compreender o fenômeno com maior profundidade
- Entrevistando a pessoa pessoalmente, por vídeo, chamada telefônica etc.

○ Entrevistas qualitativas etnográficas



Entrevistas

○ Entrevistas qualitativas etnográficas

- Segue o padrão da qualitativa, porém com a vivência no ambiente de estudo
- Compreender o ambiente que – no nosso caso – o sistema será inserido



Entrevistas

○ Entrevistas qualitativas etnográficas

- Segue o padrão da qualitativa, porém com a vivência no ambiente de estudo
- Compreender o ambiente que – no nosso caso – o sistema será inserido



Referências

- BIZAGI (Empresa). **Process Modeling**. 2016. Vídeo com restrição de cadastramento no ambiente elearning.com. Disponível em: <<http://elearning.bizagi.com/my/intermedia.php?lang=en>>. Acesso em: out. 2017.
- GUEDES, G. **UML 2 – Uma Abordagem Prática**. São Paulo: Novatec, 2009.
- LOWE, David e PRESSMAN, Roger S. **Engenharia Web - Um Enfoque Profissional**. LTC, 2011.
- PRESSMAN, R. S. **Engenharia de Software**. São Paulo: McGraw Hill, 2009.
- PAULA FILHO, W. P. **Engenharia de Software: Fundamentos, Métodos e Padrões**. 3. ed. Rio de Janeiro: LTC, 2012.
- SOMERVILLE, I. **Engenharia de Software**. São Paulo: Addison Wesley Brasil, 2007.
- VAZQUES, C. E.; SIMÕES, G.S. **Engenharia de Requisitos: Software Orientada a Negócios**. Rio de Janeiro. Brasport: 2016.
- CARDOSO, R.M., MASSON, F.P., PEDRINI, S.C. **Sistema de atendimento ao excepcional – Apae**. 1998. (Trabalho de Graduação). Faculdades Salesianas. Araçatuba.
- Monitoria de Engenharia de Software. **Estudo de viabilidade do software**. Disponível em: <<https://monitoriadeengenhariadesoftware.wordpress.com/2016/09/06/estudo-de-viabilidade-de-software/>>