



Instituto Federal de Educação, Ciência e Tecnologia da Bahia - IFBA
Departamento de Ciência da Computação
Tecnólogo em Análise e Desenvolvimento de Sistemas

Engenharia de requisitos

André L. R. Madureira <andre.madureira@ifba.edu.br>
Doutorando em Ciência da Computação (UFBA)
Mestre em Ciência da Computação (UFBA)
Engenheiro da Computação (UFBA)

Relembrando a classificação de requisitos ...

- Requisitos podem ser classificados de duas formas:
 - **Classificação por nível de detalhamento**
 - Requisitos de usuário
 - Requisitos de sistema

Relembrando a classificação de requisitos ...

- Requisitos podem ser classificados de duas formas:
 - **Classificação por nível de detalhamento**
 - Requisitos de usuário
 - Requisitos de sistema
 - **Classificação por funcionalidade**
 - Requisitos funcionais
 - Requisitos não-funcionais

Classificação por nível de detalhamento

Requisitos de usuário:

descrição abstrata através de diagramas ou linguagem natural

1. O MHC-PMS deve gerar relatórios gerenciais mensais que mostrem o custo dos medicamentos prescritos por cada clínica durante aquele mês.

Requisitos de sistema:

descrição detalhada dos requisitos
(*especificação funcional*)

- 1.1 No último dia útil de cada mês deve ser gerado um resumo dos medicamentos prescritos, seus custos e as prescrições de cada clínica.
- 1.2 Após 17:30h do último dia útil do mês, o sistema deve gerar automaticamente o relatório para impressão.

Classificação de requisitos por funcionalidade

Requisitos funcionais:

descrevem os serviços fornecidos, reação à entradas específicas e comportamento

Requisitos não-funcionais:

restrições sobre os serviços ou funções do sistema

RF1: Pesquisa de listas de agendamento por clínica

RF2: Geração diária de lista de pacientes com consulta confirmada

RF3: Identificação de usuários do sistema por número de 8 dígitos

RNF1: Sistema disponível de segunda a sexta-feira, das 08h30 às 17h30.

RNF2: Autenticação dos usuários com seus cartões de identificação.

RNF3: Disposições de privacidade dos pacientes e dos seus dados.

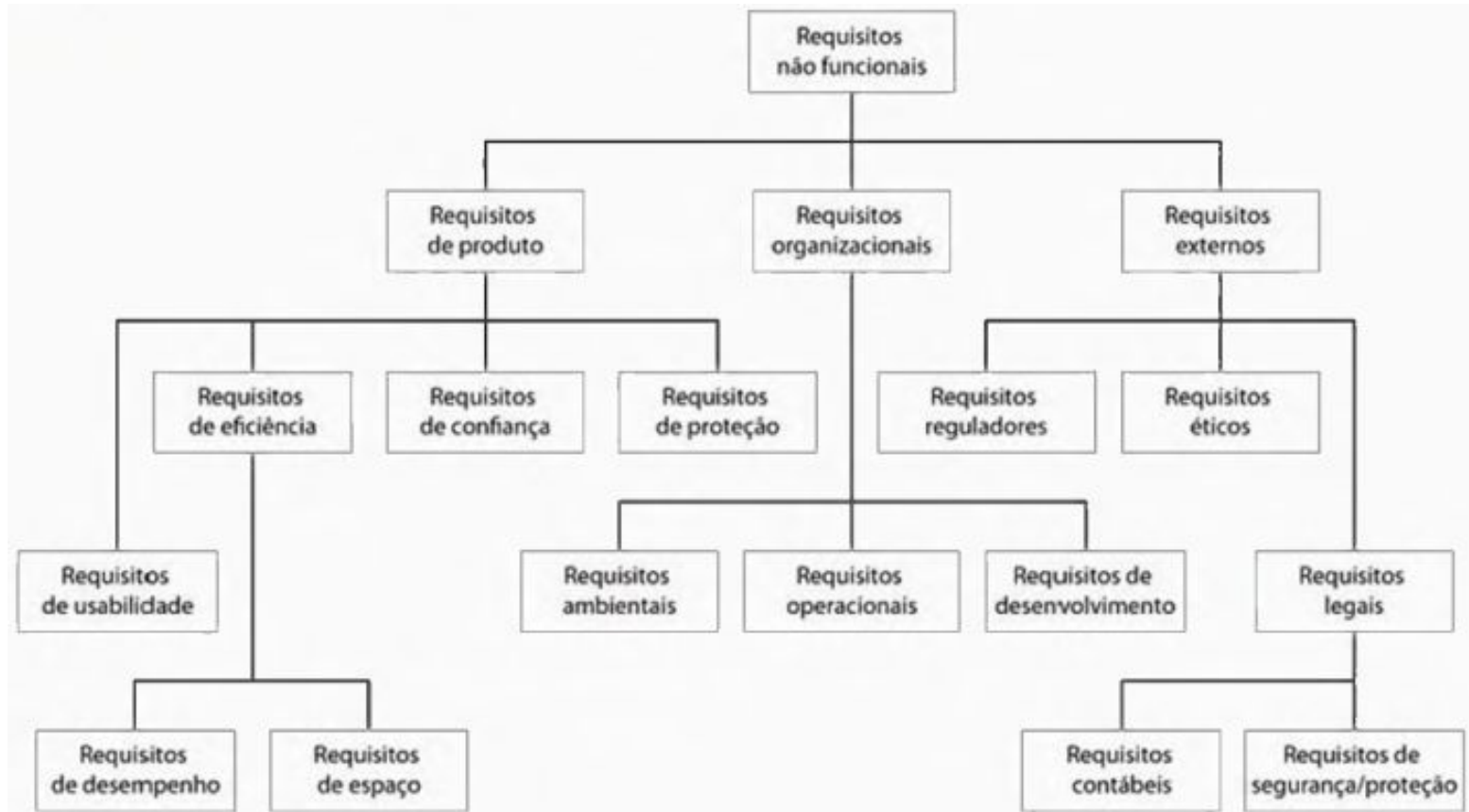
Requisitos não-funcionais

- Podem afetar a arquitetura geral de um sistema
 - **Ex:** para garantir que os requisitos de desempenho sejam atendidos, precisamos
 - Organizar o sistema para minimizar a comunicação entre os componentes
 - **Ex:** Modulo de autenticação (login) -> Banco de dados

Classificação de requisitos não-funcionais

- **Requisitos do produto:** associados ao comportamento do software
 - **Ex:** desempenho, memória consumida pelo sistema, etc
- **Requisitos organizacionais:** derivados das políticas e procedimentos da organização do cliente e do desenvolvedor
 - **Ex:** linguagens de programação, ambiente operacional, etc
- **Requisitos externos:** derivam de fatores externos ao sistema e seu processo de desenvolvimento
 - **Ex:** requisitos reguladores, legais, éticos, etc

Classificação de requisitos não-funcionais



Dependência de requisitos

- Os requisitos não são independentes uns dos outros
 - Requisitos podem gerar ou restringir outros requisitos
 - **Ex:** Suponha um sistema com requisito de usuário 1 (**RU1**) de *“limitação de acesso a usuários autorizados”*
 - **RU1** parece um requisito não-funcional, mas ele gera outros requisitos, como o requisito funcional 1 (**RF1**) de *“autenticação de usuário no sistema”*
 - **RF1** é funcional pois requer implementação de telas (login) e estruturas de dados (usuario, senha)

Imprecisão nos requisitos

- **A imprecisão na especificação de requisitos causa problemas**
 - Um desenvolvedor de sistemas pode interpretar um requisito ambíguo de uma maneira que simplifique a sua implementação
 - A implementação pode não atender à demanda do cliente
 - Definir novos requisitos e realizar alterações no sistema
 - Atrasos de entrega e aumento nos custos

Exemplo de Imprecisão nos requisitos

- **Requisito:** Usuário deve ser capaz de buscar as listas de agendamentos para todas as clínicas
 - **Implementação 01:** Dado um nome de paciente, o sistema procura por esse nome em todos os agendamentos de todas as clínicas

Exemplo de Imprecisão nos requisitos

- **Requisito:** Usuário deve ser capaz de buscar as listas de agendamentos para todas as clínicas
 - **Implementação 01:** Dado um nome de paciente, o sistema procura por esse nome em todos os agendamentos de todas as clínicas
 - **Implementação 02:** usuário escolhe uma clínica e realiza a pesquisa
 - Envolve mais entradas do usuário e necessita de mais tempo que a primeira implementação

Propriedades dos requisitos

- **Compleitude**

- Todos os serviços exigidos pelo usuário devem ser definidos

- **Consistencia**

- Os requisitos não devem ter definições contraditórias

É praticamente impossível alcançar completude e consistência dos requisitos em sistemas grandes ou complexos

Razão 01: Facilidade de cometer erros e omissões

Razão 02: Esses sistemas possuem muitos *stakeholders*, com necessidades diferentes (muitas vezes inconsistentes)

Diretrizes para completude e consistência de requisitos

- **Padronização na descrição dos requisitos (especificação estruturada)**
 - Definições de requisitos aderem a um formato padronizado
 - *Templates* (modelos) para especificar os requisitos de sistema
 - **Racional:**
 - Torna menos provável omissões na descrição
 - Facilita a verificação dos requisitos

Método VOLERE

(ROBERTSON e ROBERTSON, 1999)

- Método para padronização da descrição dos requisitos
- Os requisitos de usuário são inicialmente **escritos em cartões**, um requisito por cartão
- Cada cartão tem um conjunto de campos, por exemplo:
 - Função (comportamento esperado)
 - Dependências de outros requisitos
 - Entrada / Saida
 - Colaterais / Ação / Condições

Bomba de insulina/Software de controle/SRS/3.3.2

Função	Calcula doses de insulina: nível seguro de açúcar.
Descrição	Calcula a dose de insulina a ser fornecida quando o nível de açúcar está na zona de segurança entre três e sete unidades.
Entradas	Leitura atual de açúcar (r2), duas leituras anteriores (r0 e r1).
Fonte	Leitura atual da taxa de açúcar pelo sensor. Outras leituras da memória.
Saídas	CompDose — a dose de insulina a ser fornecida.
Destino	Loop principal de controle.
Ação	CompDose é zero se o nível de açúcar está estável ou em queda ou se o nível está aumentando, mas a taxa de aumento está diminuindo. Se o nível está aumentando e a taxa de aumento está aumentando, então CompDose é calculado dividindo-se a diferença entre o nível atual de açúcar e o nível anterior por quatro e arredondando-se o resultado. Se o resultado é arredondado para zero, então CompDose é definida como a dose mínima que pode ser fornecida.
Requisitos	Duas leituras anteriores, de modo que a taxa de variação do nível de açúcar pode ser calculada.
Pré-condição	O reservatório de insulina contém, no mínimo, o máximo de dose única permitida de insulina.
Pós-condições	r0 é substituída por r1 e r1 é substituída por r2.
Efeitos colaterais	Nenhum.

Diretrizes para completude e consistência de requisitos

- **Destaque as partes fundamentais (**negrito**, *itálico*, **cores**)**
- **Use linguagem consistente para distinguir requisitos**
 - **Requisitos obrigatórios:** o sistema DEVE fornecer
 - São essenciais
 - **Requisitos desejáveis:** o sistema PODE fornecer
 - São não essenciais

Propriedades dos requisitos

- **Compleitude**

- Todos os serviços exigidos pelo usuário devem ser definidos

- **Consistencia**

- Os requisitos não devem ter definições contraditórias

- **Testabilidade**

- Os requisitos devem ser, sempre que possível, testáveis

Exemplos de requisitos testáveis

- **Contra-exemplo (requisito NÃO TESTÁVEL):**
 - O sistema com boa usabilidade pelo pessoal médico
 - Ele deve ser organizado de tal maneira que os erros dos usuários sejam minimizados

Exemplos de requisitos testáveis

- **Contra-exemplo (requisito NÃO TESTÁVEL):**

- O sistema com boa usabilidade pelo pessoal médico
- Ele deve ser organizado de tal maneira que os erros dos usuários sejam minimizados

- **Exemplo:**

- Equipe médica deve ser capaz de usar todas as funções do sistema após 4 horas de treinamento.
- Após treinamento, a média de erros cometidos por usuários experientes não deve exceder 2 / hora de uso do sistema.

Métricas para avaliação de requisitos

Propriedade	Medida
Velocidade	Transações processadas/segundo Tempo de resposta de usuário/evento Tempo de atualização de tela
Tamanho	Megabytes Número de chips de memória ROM
Facilidade de uso	Tempo de treinamento Número de <i>frames</i> de ajuda
Confiabilidade	Tempo médio para falha Probabilidade de indisponibilidade Taxa de ocorrência de falhas Disponibilidade
Robustez	Tempo de reinício após falha Percentual de eventos que causam falhas Probabilidade de corrupção de dados em caso de falha
Portabilidade	Percentual de declarações dependentes do sistema-alvo Número de sistemas-alvo

Documento de requisitos de software

- É o objetivo final dos processos de engenharia de requisitos
- Também chamado de **especificação de requisitos de software**
- Declaração oficial do que deve ser implementado, contendo
 - **Requisitos de usuário**
 - **Requisitos de sistema**
- Pode ser subdividido em vários documentos, conforme a quantidade de requisitos

Norma IEEE para documento de requisitos (IEEE, 1998)

Capítulo	Descrição
Prefácio	Define os leitores do documento, descreve o histórico de versões e justificativa para criação de uma nova versão, resume as mudanças feitas de uma versão para outra
Introdução	Descreve as funções e interações do sistema com demais softwares, e como o sistema colabora para atingir os objetivos de negócio ou do cliente que o encomendou
Glossário	Define termos técnicos, sem supor experiência ou conhecimentos prévios do leitor
Requisitos de usuário	Descreve de alto nível de requisitos funcionais, não-funcionais, normas de produto e processos a serem seguidas

Essa é uma norma **genérica**, que PODE ser adaptada conforme necessário para o projeto, metodologia de desenvolvimento e políticas da empresa

Norma IEEE para documento de requisitos (IEEE, 1998)

Arquitetura do sistema	Visão de alto nível da arquitetura do sistema, com os módulos e funções correspondentes
Requisitos do sistema	Descreve requisitos funcionais e não-funcionais detalhadamente.
Modelos do sistema	Descreve graficamente o relacionamento entre os componentes do sistema e entre o sistema e o meio externo
Evolução do sistema	Descreve os pressupostos do sistema, bem como mudanças previstas para a evolução do sistema
Apêndices	Fornece descrição de hardware, banco de dados e demais necessidades exigidas para o funcionamento do sistema

Finalidade do documento de requisitos

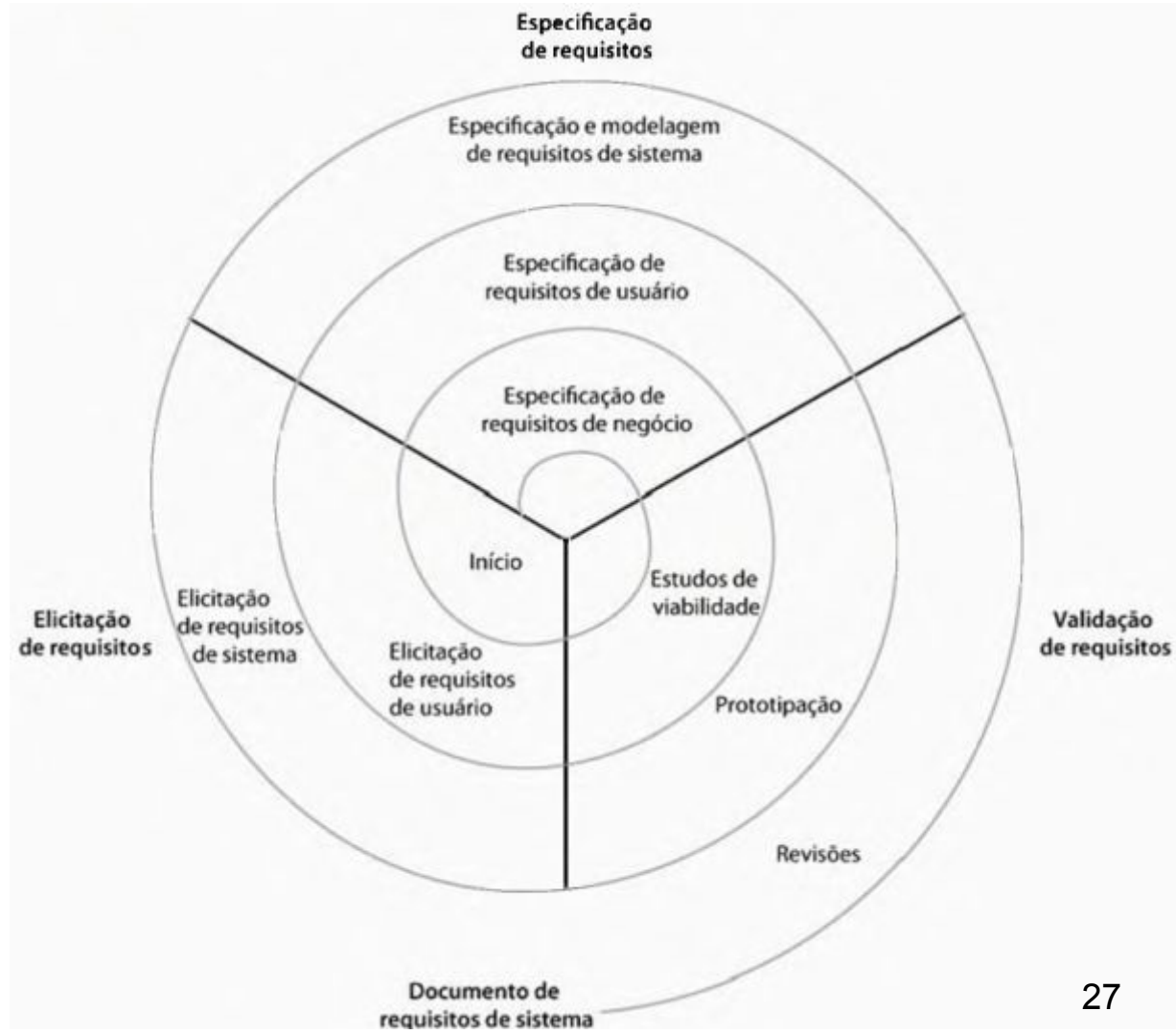
Usuário	Utilizam o documento de requisitos para
Clientes	Especificar e verificar requisitos
Gerentes	Planejar propostas e processo de desenvolvimento
Engenheiros de sistema	Entender o funcionamento do sistema
Engenheiros de testes	Desenvolver testes de validação do sistema
Engenheiros de manutenção	Entender o funcionamento do sistema

Engenharia de requisitos

- É um processo que visa construir a especificação do sistema (**documento de requisitos**), composto pelas seguintes atividades:
 - **Estudo de viabilidade:** análise de viabilidade do sistema
 - **Elicitação e análise:** descoberta e análise de requisitos
 - **Especificação:** documentação dos requisitos
 - **Validação:** verificação dos requisitos

Visão em espiral da Engenharia de Requisitos

As atividades da engenharia de requisitos são **intercaladas**, sendo representadas como uma espiral



Estudo de viabilidade

- O software pode ser concebido, com as restrições legais, éticas e demais restrições aplicáveis?
- Qual o potencial do software de melhorar o modelo de negócios ou a lucratividade da empresa?
- Custo do sistema justifica seu benefício potencial para empresa?

Elicitação e análise de requisitos

- Engenheiros de software trabalham com clientes e usuários finais para obter informações sobre
 - Domínio da aplicação
 - Serviços oferecidos pelo sistema
 - Desempenho do sistema
 - Restrições de hardware
 - Demais informações importantes sobre o sistema

Elicitação e análise de requisitos

- Composta pelas seguintes atividades:
 - **Descoberta de requisitos**
 - **Classificação e organização**
 - **Priorização e negociação**
 - **Especificação**

Elicitação e análise de requisitos

Descoberta de requisitos:
interação com os stakeholders do sistema
para descobrir seus requisitos

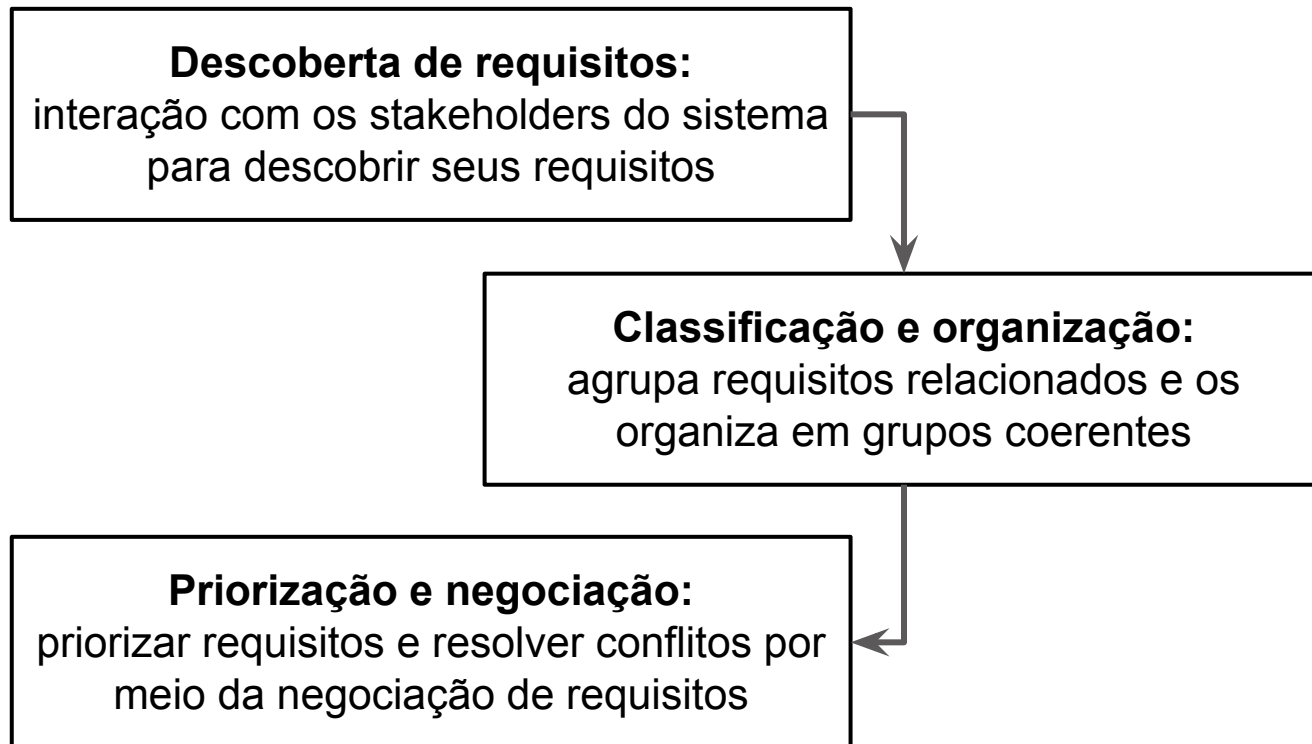
Elicitação e análise de requisitos

Descoberta de requisitos:
interação com os stakeholders do sistema
para descobrir seus requisitos

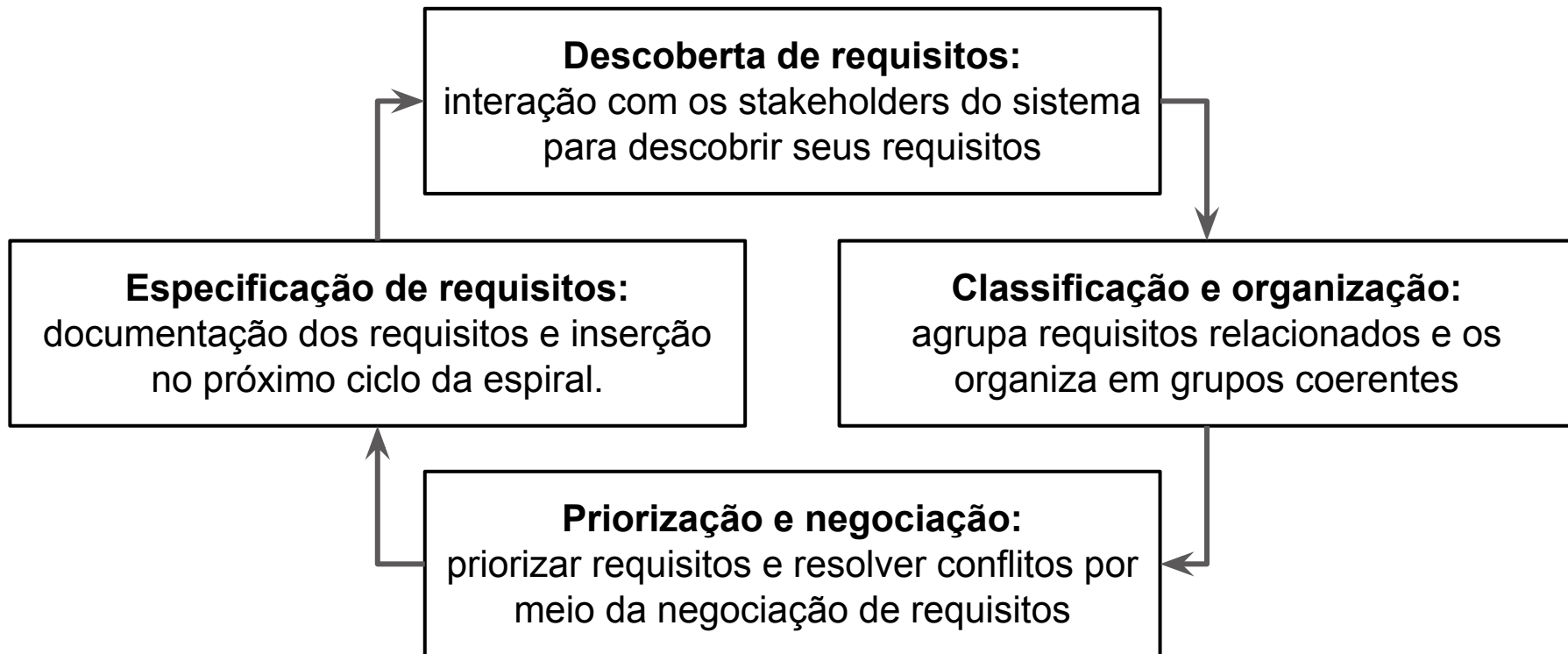


Classificação e organização:
agrupa requisitos relacionados e os
organiza em grupos coerentes

Elicitação e análise de requisitos



Elicitação e análise de requisitos



Técnicas de elicitação e análise de requisitos

- Existem várias técnicas para descobrir e analisar requisitos
- As mais utilizadas são
 - **Entrevistas** com *stakeholders* (questionários)
 - **Cenários** de uso (descrição textual)
 - **Casos** de uso (diagramas UML)
 - **Etnografia** (observação dos *stakeholders*)

Entrevistas

- A equipe de engenharia de requisitos questiona os stakeholders sobre
 - O sistema que usam no momento
 - O sistema que será desenvolvido.
- Requisitos surgem a partir das respostas a essas perguntas
- Entrevistas podem ser classificadas como:
 - **Formais ou informais**
 - **Fechadas ou abertas**



Entrevistas

- Podem ser classificadas como:
 - **Formais ou informais**
 - **Formais:** organizadas previamente (data e horário)
 - **Informais:** sem um agendamento ou organização prévios
 - **Fechadas ou abertas**
 - **Fechadas:** conjunto predefinido de perguntas
 - **Abertas:** não existe questionário (agenda) definido

Entrevistas

- Na prática, as entrevistas com stakeholders costumam ser uma mistura dos tipos de entrevista (aberta, fechada, informal e formal)
- Permitem obter conhecimento acerca da:
 - Compreensão global sobre as tarefas dos *stakeholders*
 - Como eles podem interagir com o novo sistema
 - Dificuldades que eles enfrentam com os sistemas atuais

Limitações das Entrevistas

- Entrevistas são ineficazes, para estabelecer requisitos de domínio e da organização, pois há
 - **Dificuldade na compreensão dos requisitos de domínio**
 - **Dificuldade na elicitación de requisitos e restrições organizacionais**

Dificuldade na compreensão de requisitos de domínio

- Dificuldade no entendimento de terminologias e jargões específicos
 - Especialistas em aplicações usam terminologias e jargões específicos para um domínio
 - **Ex:** (bancos) *spread* bancário, amortização, ROI, ROE
- *Stakeholders* omitem informações que eles consideram triviais, dado a sua familiaridade com certos assuntos
 - **Ex:** (entrevista *bibliotecário*) Omissão da necessidade de catalogar livros antes de adiciona-los ao acervo da biblioteca

Dificuldade na elicitação de requisitos e restrições organizacionais

- Há relações de poder sutis entre diferentes pessoas da organização
 - A estrutura organizacional real é diferente da teórica
 - **Ex:** Presidente de um país, deputados, vereadores e juízes
 - **Organização:** Nação / País
 - Presidente deveria ser a figura de maior relevância
 - Porém a maioria das decisões é tomada pelos deputados, vereadores e juízes
 - Presidente como figura pública representativa

Cenários

- Descrições de exemplos de sessões de interação (usuário <-> sistema)
- Podem incluir:
 - Descrições de fluxo de eventos do cenário
 - Descrições de erros e exceções
 - Estado final do sistema
 - etc
- Podem ser escritos como texto, diagramas, telas, etc

Suposição inicial:

O paciente é atendido em uma clínica médica por uma recepcionista; ela gera um registro no sistema e coleta suas informações pessoais (nome, endereço, idade etc.). Uma enfermeira é conectada ao sistema e coleta o histórico médico do paciente.

Normal:

A enfermeira busca o paciente pelo sobrenome. Se houver mais de um paciente com o mesmo sobrenome, o nome e a data de nascimento são usados para identificar o paciente.

A enfermeira escolhe a opção do menu para adicionar o histórico médico.

A enfermeira segue, então, uma série de *prompts* do sistema para inserir informações sobre consultas em outros locais, os problemas de saúde mental (entrada de texto livre), condições médicas (enfermeira seleciona condições do menu), medicação atual (selecionado no menu), alergias (texto livre) e informações da vida doméstica (formulário).

O que pode dar errado:

O prontuário do paciente não existe ou não pôde ser encontrado. A enfermeira deve criar um novo registro e registrar as informações pessoais. As condições do paciente ou a medicação em uso não estão inscritas no menu. A enfermeira deve escolher a opção 'outros' e inserir texto livre com descrição da condição/medicação.

O paciente não pode/não fornecerá informações sobre seu histórico médico. A enfermeira deve inserir um texto livre registrando a incapacidade/relutância do paciente em fornecer as informações. O sistema deve imprimir o formulário-padrão de exclusão afirmando que a falta de informação pode significar que o tratamento será limitado ou postergado. Este deverá ser assinado e entregue ao paciente.

Outras atividades:

Enquanto a informação está sendo inserida, o registro pode ser consultado, mas não editado por outros agentes.

Estado do sistema na conclusão:

O usuário está conectado. O prontuário do paciente, incluindo seu histórico médico, é inserido no banco de dados e um registro é adicionado ao *log* do sistema, mostrando o tempo de início e fim da sessão e a enfermeira envolvida.

Limitações de Cenários

- Descrição textual dificulta a visualização das interações do entre usuários e sistema
 - Diagramas e ilustrações facilitam o entendimento do problema
 - Requisitos podem ser mais facilmente identificados através de inspeção visual de interações

Casos de uso

- Identifica os atores envolvidos em uma interação
 - **Atores:** pessoas ou outros sistemas
- Descrevem a interação dos atores com o sistema
- Documentados por um diagrama de casos de uso (UML)
- O conjunto de casos de uso representa todas as possíveis interações que serão descritas nos requisitos de sistema

Recepcionista do médico(a)



Registrar
paciente

Ver
informações
pessoais

Gerente



Exportar
estatísticas

Gerar
relatórios

Enfermeira



Ver
registro

Editar
registro

Médico(a)



Agendar
consulta

**Classes de
interação
(elipses)**

**Atores
(figuras “palito”)**

Etnografia

- Técnica de observação usada para compreender e extrair requisitos dos processos operacionais
 - Um analista faz uma imersão no ambiente de trabalho (*contexto social e organizacional*) em que o sistema será usado
 - O trabalho do dia a dia é observado
 - Analista faz anotações sobre as tarefas reais em que os participantes estão envolvidos



Etnografia

- Ajuda a descobrir **requisitos implícitos** do sistema que refletem as formas reais com que as pessoas trabalham
 - Pessoas têm dificuldade de expressar os detalhes do seu trabalho
 - Procedimentos e detalhes podem ser omitidos, por serem considerados tarefas ou informações “triviais”
 - Tarefas feita de maneira “automática” tendem a ser omitidas ou explicadas de maneira superficial

Etnografia

- Ajuda a descobrir **requisitos implícitos** do sistema que refletem as formas reais com que as pessoas trabalham
 - Pessoas não compreendem a relação do seu trabalho com o restante da organização
 - *“Como cada tarefa contribui para o todo?”*
 - *“Pessoas tem visão micro do problema, o analista imerso no ambiente traz a visão macro”*

Requisitos reais que não estão formalmente descritos no projeto do sistema

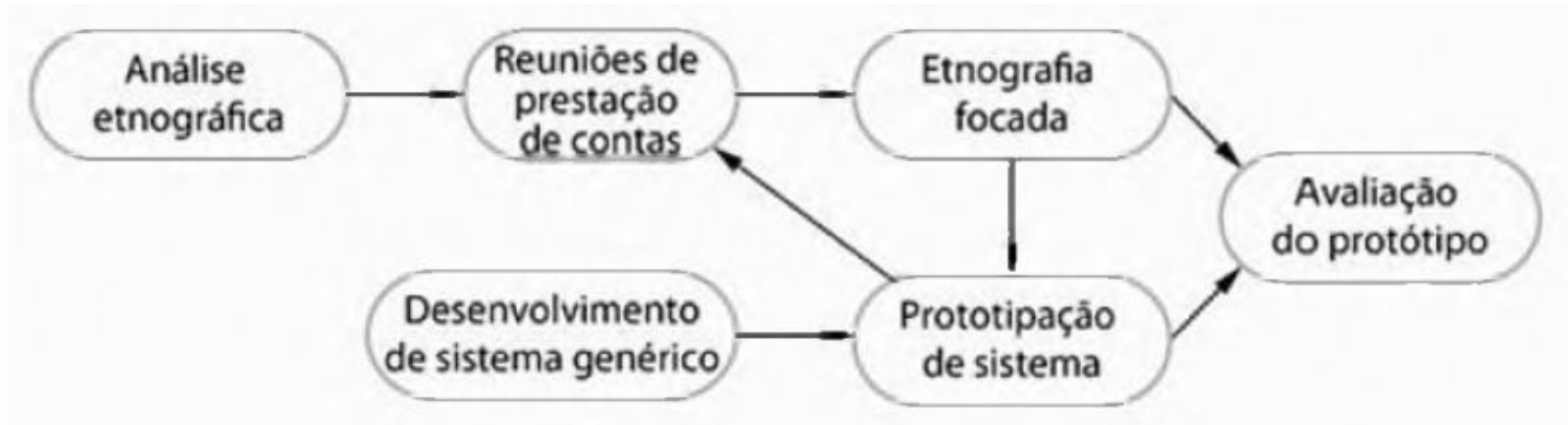
- **Ex:** controladores de tráfego aéreo podem desligar um sistema de alerta de conflitos que detecta aeronaves com rotas em colisão
 - **Racional 01:** otimizar o gerenciamento do espaço aéreo, ao colocar deliberadamente a aeronave em caminhos conflitantes, por um curto período de tempo
 - **Racional 02:** o alarme de alerta distrai seu trabalho
 - Eles se asseguram que os aviões sejam afastados da rota conflitante antes que surjam problemas

Requisitos reais derivados da cooperação entre pessoas

- **Ex:** controladores de tráfego aéreo conseguem prever o número de aeronaves que entrarão em seu setor de controle
 - **Como:** analisando o trabalho de outros controladores de tráfego
 - **Porque:** modificar as estratégias de controle, dependendo do volume de trabalho previsto.
 - **Requisito:** o sistema deve permitir aos controladores de um setor alguma visibilidade do trabalho em setores adjacentes.

Etnografia e prototipação

- Técnicas de etnografia e prototipação podem ser utilizadas em conjunto para melhorar os processo de elicitação de requisitos



Limitações da Etnografia

- Etnografia nem sempre é capaz de descobrir requisitos organizacionais ou de domínio
 - Foco da técnica está no usuário e suas interações (e não no sistema e seu domínio de aplicação)
- **Logo, a etnografia não é uma abordagem completa de elicitação**
 - Ela complementa outras abordagens, como análise de casos de uso, cenários, entrevistas, etc

Validação de requisitos

- É o processo de verificar se os requisitos definem o sistema que o cliente realmente quer
- **Racional:** Altos custos de retrabalho quando erros nos requisitos são descobertos durante o desenvolvimento ou na utilização do sistema
 - Mudanças nos requisitos resultam em alterações no projeto e na implementação do sistema
 - Além disso, o sistema deve ser retestado após as mudanças

Tipos de validação / verificação de requisitos

- **Verificações de consistência**
- **Verificações de completude**
- **Verificações de realismo**
- **Verificabilidade**

Tipos de validação / verificação de requisitos

- **Verificações de consistência**

- Requisitos não devem entrar em conflito
 - Não há restrições contraditórias
 - Não há descrições diferentes da mesma função do sistema

- **Verificações de completude**

- Todas as funções e restrições pretendidas pelo usuário devem estar contidas no documento de requisitos

Tipos de validação / verificação de requisitos

- **Verificações de realismo**

- Se asseguram que realmente os requisitos descritos podem ser implementados, dentro do orçamento e cronograma previstos, com as tecnologias existentes

- **Verificabilidade**

- Requisitos devem permitir a escrita de testes que demonstrem que o sistema entregue atende a cada requisito especificado
- **Racional:** reduzir os conflitos entre o cliente e o contratante

Técnicas de validação de requisitos

- **Revisões de requisitos:** Analise sistemática dos requisitos em busca de erros e inconsistências
- **Prototipação:** modelo executável do sistema é fornecido para os usuários finais e clientes, para verificarem se o mesmo atende às suas necessidades
- **Casos de teste:** construir testes para cada requisito
 - A dificuldade no projeto de um teste normalmente significa que os requisitos serão difíceis de serem implementados e devem ser reconsiderados.

Referencial Bibliográfico

- SOMMERVILLE, Ian. **Engenharia de Software**. 6. ed. São Paulo: Addison-Wesley, 2003.
- PRESSMAN, Roger S. **Engenharia de Software**. São Paulo: Makron Books, 1995.
- JUNIOR, H. E. **Engenharia de Software na Prática**. Novatec, 2010.