

Análise e Desenvolvimento de Sistemas Engenharia de Software I

Engenharia de Requisitos



Agenda

- Requisitos de Software
- Engenharia de Requisitos

Agenda

- Requisitos de Software
- Engenharia de Requisitos

Requisitos de Software

- **O que é um requisito?**
 - **Requisitos** para um **sistema** são **descrições** do que o sistema deveria **realizar** – os **serviços** que ele **provê** e as **restrições** sobre sua **operação**;
 - Existem **vários níveis** de **requisitos**: um requisito **pode variar** desde uma **instrução abstrata** de **alto nível** ou de uma **restrição do sistema** até uma **especificação matemática** funcional;
 - Os **requisitos** podem servir uma **dupla função**:
 - Pode ser a **base** para uma **proposta** de um **contrato** – ser **genérico** e, portanto, deve estar **aberto à interpretação**;
 - Pode ser a **base** para o **contrato em si** – portanto deve ser **definido em detalhe**.

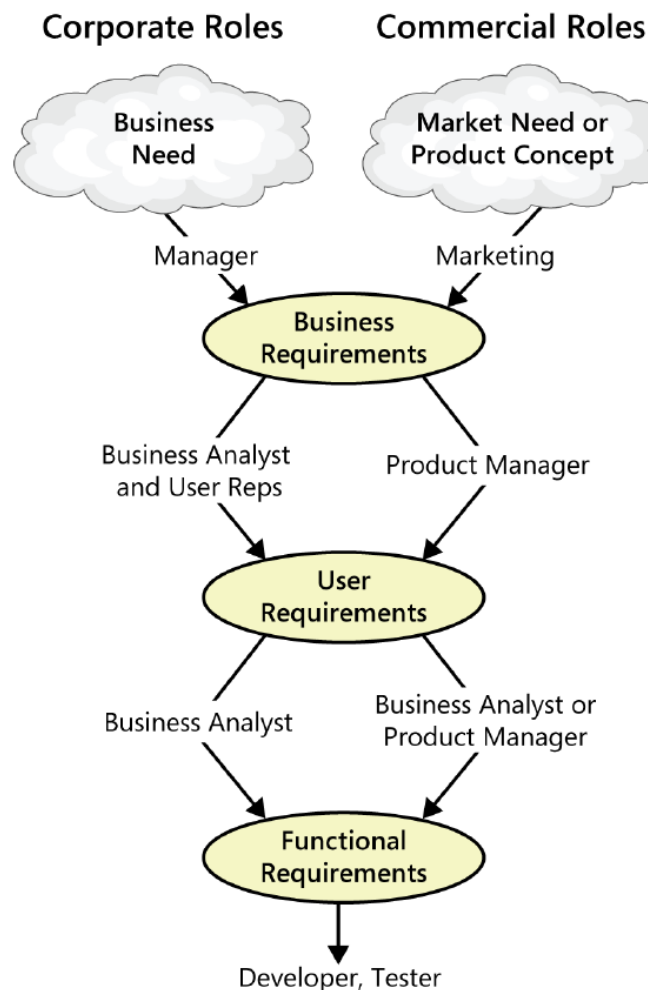
Requisitos de Software

- **Níveis de requisitos**
 - **Requisitos de negócios**
 - **Objetivos de negócios** de alto nível de uma **organização** ou de um **cliente** que necessita de um sistema.
 - **Requisitos de usuário**
 - São **descrições** em **linguagem natural** / **diagramas** dos **serviços** que o **sistema** deverá **prover**, **bem como** suas **restrições operacionais** – **voltado** para os **clientes**.
 - **Requisitos do sistema**
 - **Constitui** um **documento estruturado** **estabelecendo descrições detalhadas** das **funções** do **sistema**, **serviços** e **restrições operacionais**;
 - **Definem** o que deve ser **implementado** de modo a fazer **parte** de um **contrato** entre o **cliente** e o **desenvolvedor**.

- **Visão geral** (WIEGERS; BEATTY, 2013)

Requisitos de Software

- Níveis de requisitos
 - Aplicação (WIEGERS; BEATTY, 2013)



Requisitos de Software

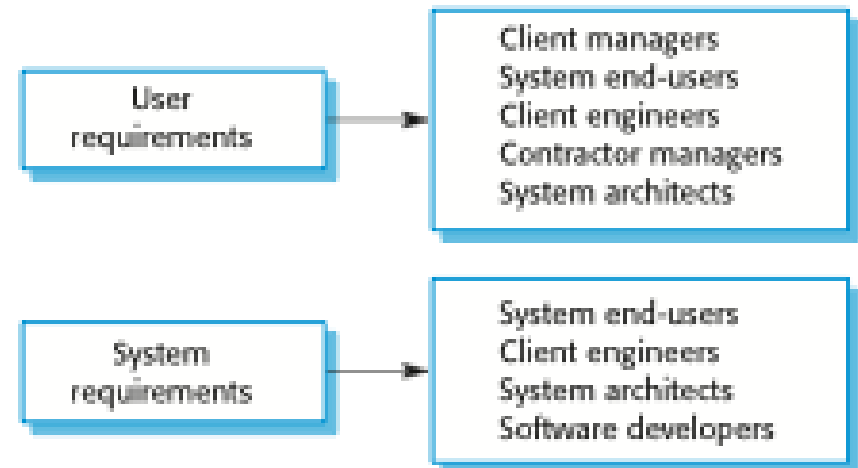
- **Níveis de requisitos**
 - **Exemplo (SOMMERVILLE, 2011)**

User requirement definition

1. The MHC-PMS shall generate monthly management reports showing the cost of drugs prescribed by each clinic during that month.

System requirements specification

1.1 On the last working day of each month, a summary of the drugs prescribed, their cost and the prescribing clinics shall be generated.
 1.2 The system shall automatically generate the report for printing after 17:30 on the last working day of the month.
 1.3 A report shall be created for each clinic and shall list the individual drug names, the total number of prescriptions, the number of doses prescribed and the total cost of the prescribed drugs.
 1.4 If drugs are available in different dose units (e.g. 10mg, 20 mg, etc.) separate reports shall be created for each dose unit.
 1.5 Access to all cost reports shall be restricted to authorized users listed on a management access control list.

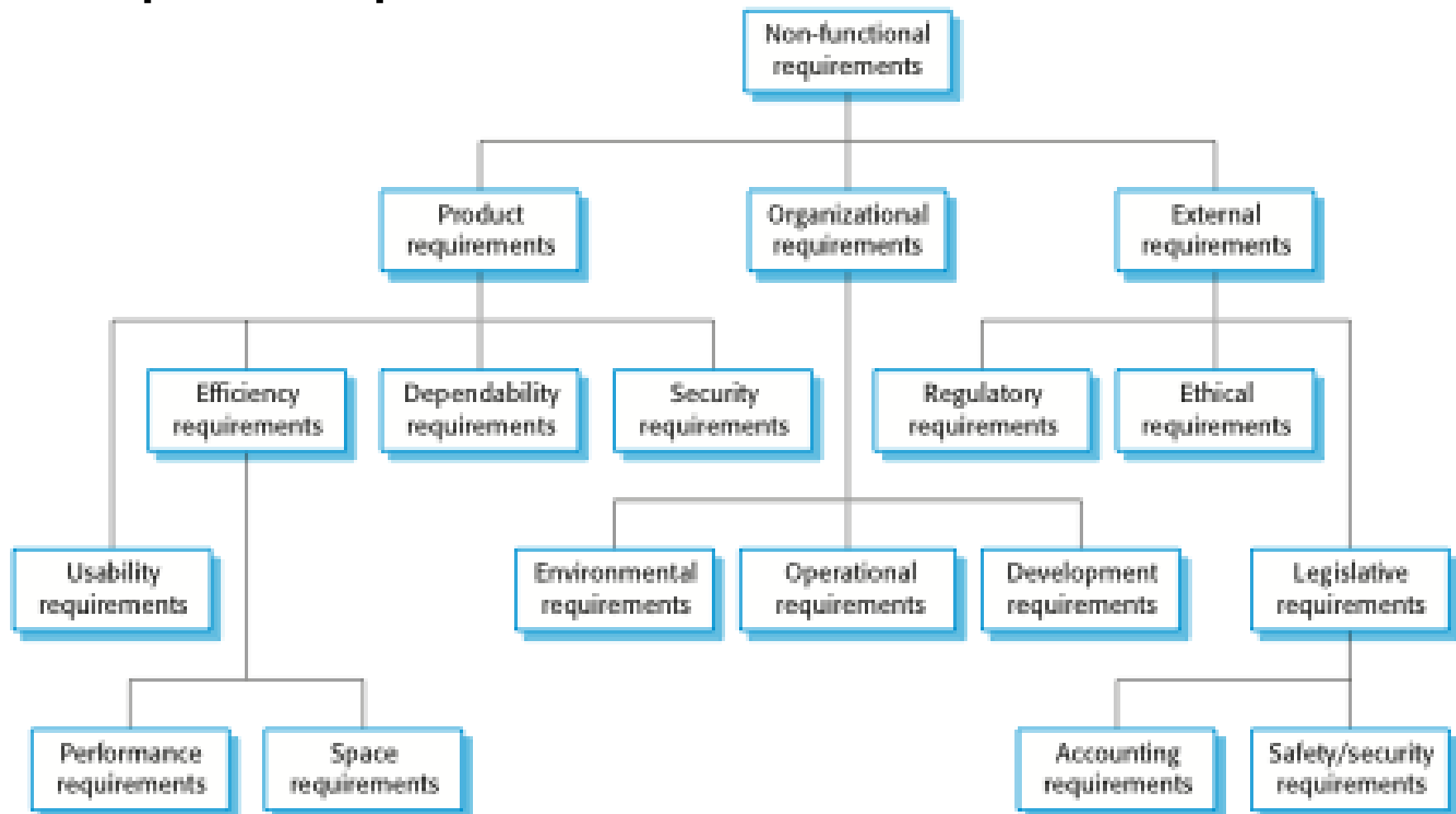


Requisitos de Software

- **Classificação dos requisitos (SOMMERVILLE, 2011)**
 - **Requisitos funcionais**
 - Descrevem **o que o sistema** deverá **prover**, como o **sistema** deverá **reagir a entradas particulares** e como o **sistema** deverá **se comportar em situações particulares**;
 - Descrevem **também** o que o **sistema não** deverá **prover**.
 - **Requisitos não funcionais**
 - São **propriedades** ou **qualidades** que o **produto** deve **apresentar** – são **restrições sobre serviços** ou **funções** do **sistema** (tempo de resposta, número de transações simultâneas etc);
 - Se **aplicam** sobre o **produto** como um **todo**.
 - **Requisitos do domínio**
 - Provenientes do **domínio** da **aplicação** do **sistema** e que **refletem características** desse **domínio** (leis, regulamentações, fórmulas físicas etc).

Requisitos de Software

- **Classificação dos requisitos (SOMMERVILLE, 2011)**
 - **Tipos de requisitos não funcionais**



Requisitos de Software

- **Classificação dos requisitos (SOMMERVILLE, 2011)**
 - **Requisitos não funcionais – ISO9126**
 - **Funcionalidade:** abrange aspectos da adequabilidade, precisão, interoperabilidade, compatibilidade e segurança do software
 - **Confiabilidade:** abrange aspectos da maturidade, recuperação e tolerância a falhas
 - **Utilização:** abrange aspectos da capacidade de aprendizado, capacidade de entendimento e operabilidade do software
 - **Eficiência:** abrange aspectos do comportamento (temporal e utilização de recursos)
 - **Sustentabilidade:** abrange aspectos de manutenção, como estabilidade, capacidade de entender o sistema, capacidade de realizar alterações e testes
 - **Portabilidade:** abrange aspectos de instalação, adaptação, e capacidade de substituição.

Requisitos de Software

- **Imprecisão dos requisitos**
 - **Requisitos ambíguos** podem ser **interpretados** de maneiras diferentes pelos **desenvolvedores** e **usuários**;
 - **Exemplo:**
 - *O usuário deve ser capaz de pesquisar as listas de consultas para todas as clínicas.*
 - **O que se quer dizer com “pesquisar”?**
 - **Intenção do usuário** – a pesquisa de um nome do paciente em todas as consultas de todas as clínicas;
 - **Interpretação do desenvolvedor** – procurar por um nome do paciente em uma clínica individual. O utilizador escolhe a clínica, em seguida, procurar.

Requisitos de Software

- **Características de bons requisitos**
 - **Completo**
 - Eles devem incluir **descrições** de **todas** as **necessidades** do **sistema**.
 - **Consistente**
 - **Não** deve **haver conflitos** ou **contradições** nas **descrições** das **necessidades** do sistema.
 - **Problema**
 - Na **prática**, é **impossível produzir** um **documento** de requisitos **completo** e **consistente**.

Requisitos de Software

■ Exercício em sala – I

- Considere um **sistema de caixa eletrônico** com os seguintes **requisitos**:
 - Permitir o saque de quantias até R\$500,00 por dia;
 - Executar qualquer transação financeira em até 30 segundos;
 - Imprimir extrato de conta-corrente dentro de um intervalo escolhido pelo usuário;
 - Garantir que em uma transação de transferência entre contas o valor debitado da conta origem seja igual ao valor creditado da conta destino;
 - Pagar conta a partir da leitura do código de barras da conta.
- **Q: classificar os requisitos** acima em **funcionais, não funcionais e de domínio**.

Requisitos de Software

■ Exercício em sala – II

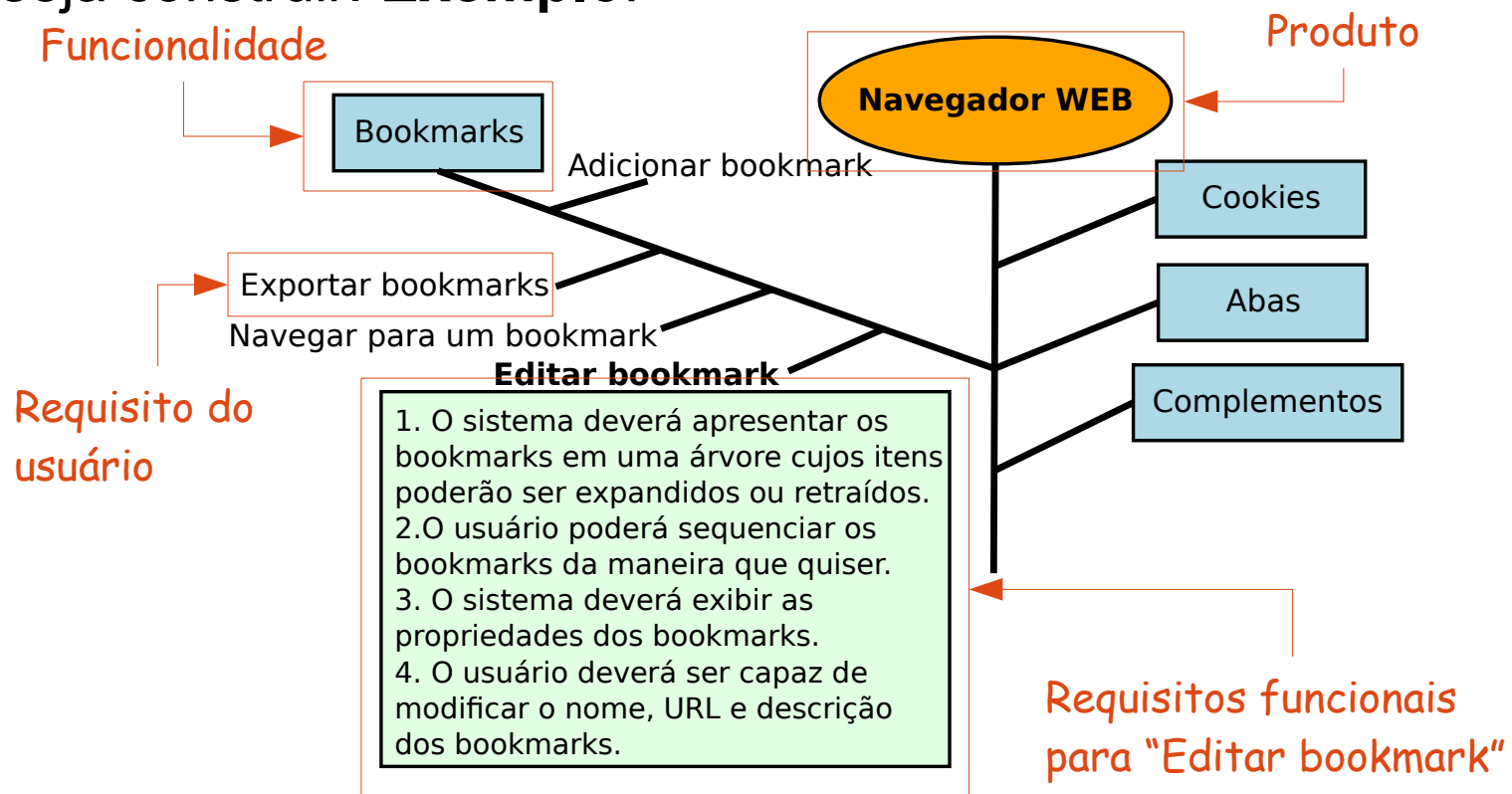
- Imagine que você desenvolverá um sistema tipo *kiosk* (sistema interativo com o usuário), que será instalado em diversas ruas de uma cidade para oferecer serviços úteis aos cidadãos;
- Elabore um conjunto inicial dos principais requisitos que este sistema deveria oferecer no formato de uma tabela de requisitos, conforme descrito a seguir:

ID Requisito	Descrição	Prioridade (Alta/Comum/Baixa)
R0001	O sistema deverá ...	Alta
...		

Requisitos de Software

Exercício em sala – III

- Uma **ferramenta visual** que relaciona **funcionalidade** (*feature*), **requisitos do usuário** e **requisitos funcionais** é a **árvore de funcionalidades** (*feature tree*), que especifica um produto que se deseja construir. **Exemplo:**



Requisitos de Software

- **Exercício em sala – III (cont.)**
 - **Elaborar uma árvore de funcionalidades para um produto de software destinado à recolocação de candidatos a empregos.** O sistema poderá ser **operado** pelas seguintes **categorias de usuários**:
 - Candidatos;
 - Empresas;
 - Administradores do sistema em questão.
 - Especificar os requisitos funcionais de 5 (cinco) requisitos do usuário, escolhidos à vontade.

Agenda

- Requisitos de Software
- Engenharia de Requisitos

Engenharia de Requisitos

- **Definição (PRESSMAN, 2010)**
 - **Começa** durante a **atividade de comunicação** e continua na **atividade de modelagem**;
 - **Constrói** uma **ponte entre** os **requisitos** do sistema e o **projeto** (*design*) e **construção** do software;
 - **Permite analisar**
 - O **contexto** em que o **software executará**;
 - As **necessidades específicas** que tanto **projeto** quanto a **construção** deverão **abordar**;
 - As **prioridades** que **orientarão** a **ordem** em que o **trabalho** será **executado**;
 - A **informação, função e comportamento** que terá um **impacto** profundo sobre o **projeto** resultante.

Responde O
QUÊ o
sistema deve
fazer e não
COMO!

Engenharia de Requisitos

- **Tarefas (PRESSMAN, 2010)**
 - **Concepção;**
 - **Elicitação;**
 - **Elaboração;**
 - **Negociação;**
 - **Especificação;**
 - **Validação;**
 - **Gestão de Requisitos.**
- **Algumas** destas tarefas **podem ocorrer em paralelo** e são todas **adaptadas** para as **necessidades do projeto**.

Engenharia de Requisitos

- **Tarefas**

- **Concepção**

- **Faz-se aqui uma série de perguntas para se estabelecer:**
 - Uma **compreensão** básica do **problema**;
 - As **pessoas** que **desejam** uma **solução**;
 - A **natureza** da **solução** que se **pretende** obter;
 - A **eficácia** da **comunicação preliminar** e **colaboração** entre o **cliente** e o desenvolvedor.
 - **Estas questões são necessárias para:**
 - **Identificar** os **interessados**;
 - **Reconhecer** diversos **pontos de vista**;
 - **Trabalhar** na busca da **colaboração**;
 - **Quebrar o gelo** e **iniciar a comunicação**.

Engenharia de Requisitos

- Tarefas

- Elicitação

Ou **levantamento**
de requisitos

- É uma tarefa cuja **difículdade ocorre** devido à:
 - **Problemas de escopo** na **identificação** dos **limites** do **sistema** ou a **especificação** de **muitos detalhes** técnicos, **em vez** de **objetivos gerais** do **sistema**;
 - **Problemas de compreensão** do **que se quer**, do **que é** o **domínio** do **problema**, e do **que** o **ambiente** de **computação** **considerará** (omitindo informações óbvias);
 - **Problemas de volatilidade**, porque os **requisitos** **mudam** ao **longo** do **tempo**.
 - **Duas atividades conhecidas** para a **elicitação** de requisitos:
 - **Coleta colaborativa** de **requisitos**;
 - **Desdobramento da Função Qualidade** (**QFD** – *Quality Function Deployment*).

Engenharia de Requisitos

- Tarefas

- Elicitação

- Coleta colaborativa de requisitos

- **Reuniões** realizadas com a **participação** de engenheiros de **software**, **clientes** e **outras partes** interessadas;
 - **São estabelecidas regras** para **preparação** e **participação**;
 - Uma **agenda suficientemente formal** é sugerida para **cobrir todos os pontos importantes**, mas **informal** o **suficiente** para **encorajar o livre fluxo de idéias**;
 - Um “**facilitador**” (**cliente**, **desenvolvedor** ou **de fora**) **controla a reunião**;
 - São utilizados “**mecanismos de definições**” tais como **planilhas**, **flip charts**, **post-its**, **boletim eletrônico**, **sala de chat**, ou algum outro fórum virtual;
 - O **objetivo** é **identificar o problema**, **propor elementos da solução**, **negociar diferentes abordagens** e **especificar um conjunto preliminar de requisitos** da solução;

Engenharia de Requisitos

- Tarefas

- Elicitação

- QFD

- É uma **técnica de gestão de qualidade** que traduz as **necessidades do cliente** em **requisitos técnicos de software**;
 - Ela **ênfatiza** uma **compreensão do que é valioso** para o **cliente** e, em **seguida**, **implanta** esses **valores durante** todo o **processo de engenharia** por meio de **funções, informações e tarefas**;
 - Ela **identifica três tipos de requisitos**:
 - **Requisitos normais**: são os **objetivos e metas estabelecidos** para um **produto** ou **sistema**;
 - **Requisitos esperados**: estão **implícitos** no **produto** ou **sistema** - não declarados explicitamente;
 - **Requisitos fascinantes**: **funcionalidades** que vão **além** das **expectativas do cliente**.

Engenharia de Requisitos

- **Tarefas**

- **Elicitação**

- **Artefatos gerados** - depende do sistema em questão:
 - Uma **declaração de necessidade e viabilidade**;
 - Uma **declaração limitando o escopo** para o sistema ou produto;
 - Uma **lista de clientes, usuários e outras** partes interessadas que participaram no levantamento de requisitos;
 - Uma **descrição do ambiente técnico** do sistema;
 - Uma **lista de requisitos** (organizados por função) e as **restrições de domínio** que se aplicam a cada item;
 - Um **conjunto de cenários de uso preliminares** (na forma de **casos de uso**), fornecendo **informações** sobre o **uso** do **sistema** ou **produto** sob **diferentes condições** de operação;
 - **Quaisquer protótipos desenvolvidos** para melhor compreender os requisitos.

Engenharia de Requisitos

- **Tarefas**

- **Elaboração**

- Durante a **elaboração**, o **engenheiro de software** **expande e refina** as **informações obtidas** durante a **iniciação** e **elicitação**;
 - A **elaboração** se **concentra** no **desenvolvimento** de um **modelo técnico** **refinado** das **funções** do **software**, **recursos** e **limitações**;
 - É uma **tarefa de modelagem de análise**:
 - Os **casos de uso** são **desenvolvidos**;
 - **Classes** de **domínio** são **identificadas juntamente** com seus **atributos** e **relacionamentos**;
 - **Diagramas** de **máquina de estado** são usadas para **capturar** a **o ciclo de vida dos objetos**.
 - O **resultado final** é um **modelo** de **análise** que define os **domínios funcional**, de **informação** e **comportamental** do **problema**.

Engenharia de Requisitos

- Tarefas

- Negociação

- Durante a **negociação**, o engenheiro de software **concilia** os **conflitos** entre o que o **cliente** quer e **o que pode** ser **alcançado** **dados** os **recursos limitados** de **negócios**;
 - **Requisitos** são **classificados** (**priorizados**) pelos **clientes**, **usuários** e outras **partes interessadas**;
 - Os **riscos** **associados** a cada **requisito** são **identificados** e **analisados**;
 - **Estimativas** “**grosseiras**” do **esforço** de desenvolvimento são **feitas** e **usadas** para **avaliar** o **impacto** de cada **requisito** no **custo** de **projeto** e **prazo** de entrega;
 - Utilizando uma **abordagem iterativa**, os **requisitos** são **eliminados**, **combinadas** e/ou **modificados** de **modo** satisfazer as partes.

Engenharia de Requisitos

- Tarefas

- Especificação

- É o artefato final produzido pelo engenheiro de requisitos;
 - Serve como **base** para as **atividades posteriores** de engenharia de **software**;
 - **Descreve** as **funcionalidades** e o **desempenho** de um **sistema** baseado em **computador** e as **restrições** que irão **reger** o **seu desenvolvimento**;
 - **Formaliza** os **requisitos informativos, funcionais e comportamentais** do **software** proposto, **tanto** um formato **gráfico** quanto **textual**;
 - A **especificação** é tipicamente **representada** por um **documento** denominado **Especificação de Requisitos de Software (SRS – Software Requirements Specification)** – **documento** contendo uma **descrição detalhada** de **todos** os **aspectos** do **software** a ser construído.

Engenharia de Requisitos

- **Tarefas**

- **Validação**

- Na **validação**, os **artefatos produzidos** como **resultado** da engenharia de requisitos são **avaliados** pela sua **qualidade**;
 - A **especificação** é **examinada** para **assegurar** que
 - **Todos** os **requisitos** de **software** foram **descritos** sem **ambiguidade**;
 - **Inconsistências**, **omissões** e **erros** foram **detectados** e **corrigidos**;
 - Os **artefatos** **estão em conformidade** com as **normas** estabelecidas para o **processo**, o **projeto** e **produto**;
 - A **revisão técnica formal** serve como o principal **mecanismo** de **validação** de **requisitos**:
 - Os membros incluem engenheiros de software, clientes, usuários e outras partes interessadas.

Engenharia de Requisitos

- Tarefas

- Validação

- Questões levantadas na **validação** de requisitos:

- Cada **requisito** está **consistente** com o **objetivo global** para o sistema/produto?
 - **Todos** os **requisitos** foram **especificados** no **nível adequado** de **abstração**? Isto é, **existem requisitos** que possuem um **nível de detalhe técnico inapropriado** para este **estágio**?
 - O **requisito** é **realmente necessário** ou **representa** uma **característica adicional** que pode ser **não essencial** para o objetivo do sistema?
 - Cada **requisito** está **limitado** e é **não ambíguo**?
 - Cada **requisito** tem **atribuição**? Isto é, há uma **fonte** para **cada requisito**?

Engenharia de Requisitos

- **Tarefas**

- **Validação**

- **Questões levantadas na validação de requisitos (cont.):**
 - Há **conflito entre alguns requisitos**?
 - **Cada requisito é realizável no ambiente técnico** que vai abrigar o **sistema** ou produto?
 - **Cada requisito é testável, depois de implementado**?
 - **Demonstração, teste real, análise ou inspeção.**
 - Será que o **modelo de requisitos reflete adequadamente a informação, função e comportamento** do sistema a ser **construído**?
 - Será que o **documento de especificação de requisitos** foi **organizado de forma que expõe informações** cada vez **mais detalhadas** sobre o **sistema**?

Engenharia de Requisitos

- **Tarefas**

- **Gestão de requisitos**

- No gerenciamento de requisitos, a **equipe do projeto realiza um conjunto de atividades que permitem identificar, controlar e acompanhar requisitos e as mudanças nos requisitos a qualquer momento** durante a execução do projeto;
 - Cada **requisito é atribuído um identificador único**;
 - Os **requisitos** são então **colocados em uma ou mais tabelas de rastreabilidade**;
 - Estas **tabelas** podem ser **armazenadas numa base de dados** que se **relacionam com características, fontes, dependências, subsistemas e interfaces** para os requisitos;
 - Uma **tabela de rastreabilidade** também é **adicionada no fim da especificação de requisitos** de software.

Referências bibliográficas

- PRESSMAN, R. S. **Software engineering: a practitioner's approach**. 7. ed. New York: McGraw-Hill Higher Education, 2010.
- SOMMERVILLE, I. **Software Engineering**. 9. ed. Boston, MA: Pearson, 2011.
- WIEGERS, K. E.; BEATTY, J. **Software Requirements**. 3. ed. Redmond, WA: Microsoft Press, 2013.