

Engenharia de Software – 2020/2 EARTE

Jordana S. Salamon

jordana.salamon@ufes.br

jssalamon@inf.ufes.br

jordanasalamon@gmail.com

<http://inf.ufes.br/~jssalamon>

Departamento de Informática

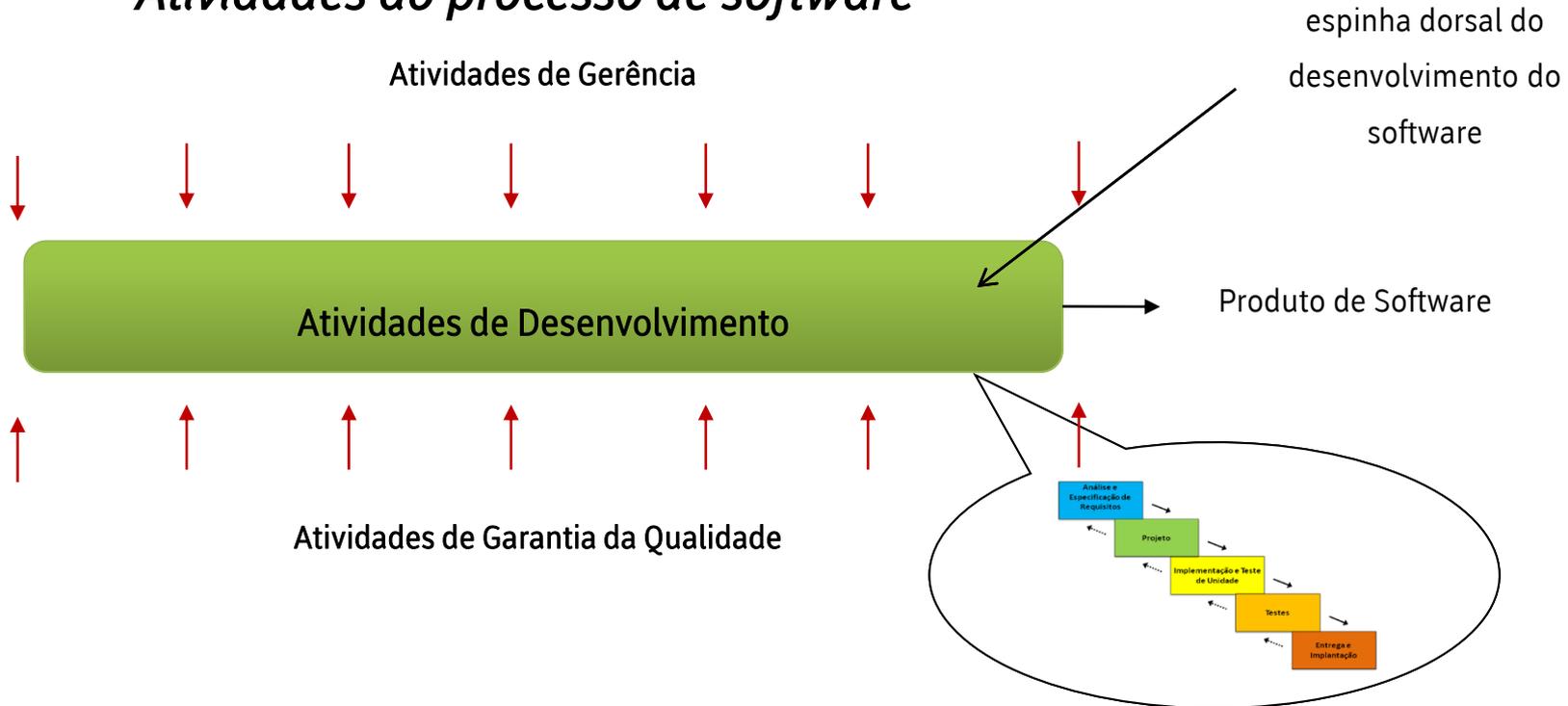
Centro Tecnológico

Universidade Federal do Espírito Santo



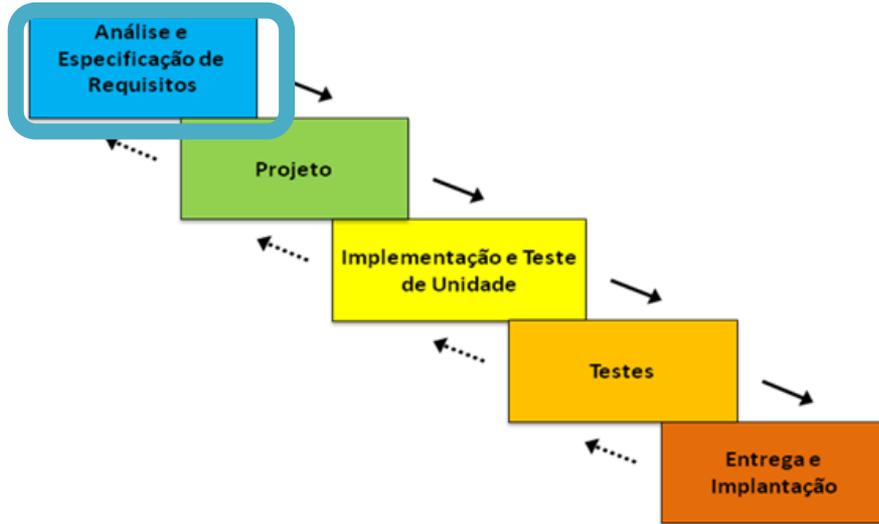
Especificação e Análise de Requisitos

Atividades do processo de software



Especificação e Análise de Requisitos

Atividades do processo de software



Em um desenvolvimento de software, a primeira coisa a ser feita é capturar os requisitos que o sistema a ser desenvolvido tem de tratar.

- Um entendimento dos requisitos do software é essencial para o sucesso de um projeto de desenvolvimento de software.

Especificação de Requisitos

- Os requisitos devem ser inicialmente levantados e descritos de maneira sucinta para permitir definir o escopo do sistema.
- Depois, os requisitos devem ser refinados em detalhes, as funções e o desempenho do software devem ser especificados e as interfaces e restrições que o software deve atender devem ser estabelecidas.
- Modelos dos dados e do comportamento do sistema devem ser elaborados e os principais artefatos produzidos devem ser submetidos à avaliação da qualidade.

Especificação de Requisitos



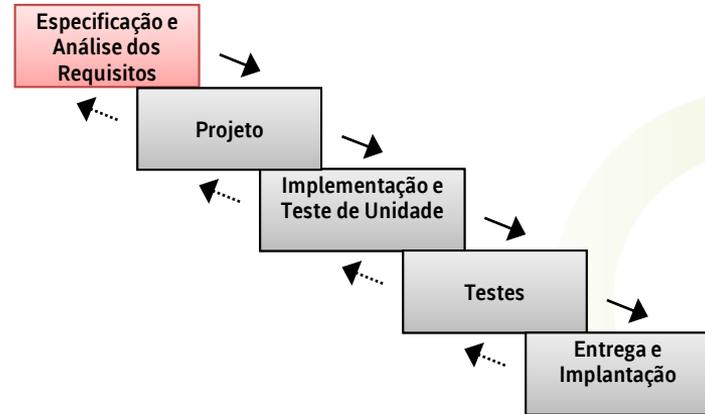
Que requisitos o software deve atender?



- Quem são os envolvidos?
- Quais são suas necessidades em relação ao software?

Mas...

O que é um requisito?



Especificação de Requisitos

Requisitos

Os requisitos de um sistema definem o que o sistema deve fazer e as circunstâncias sob as quais deve operar.

Ou, em outras palavras...

São as funções que um sistema deve incorporar e as restrições que devem ser satisfeitas.

Especificação de Requisitos

Requisitos

Uma das principais medidas do sucesso de um sistema de software é o grau no qual ele atende aos requisitos para os quais foi construído.

Especificação de Requisitos

Alguns dos problemas que surgem durante o processo de engenharia de requisitos são resultado da falha em fazer uma separação clara entre esses diferentes níveis de descrição. Os requisitos do usuário e os requisitos do sistema podem ser definidos da seguinte forma:

1. Requisitos do usuário são declarações, em uma linguagem natural somada a diagramas, de quais serviços o sistema deve fornecer aos usuários do sistema e as restrições sob as quais ele deve operar.
2. Os requisitos do sistema são descrições mais detalhadas das funções, serviços e restrições operacionais do sistema de software. O documento de requisitos do sistema (às vezes chamado de especificação funcional) deve definir exatamente o que deve ser implementado. Pode ser parte do contrato entre o comprador do sistema e os desenvolvedores de software.

Especificação de Requisitos

Tipos de Requisitos

Funcionais: apontam as funções que o sistema deve fornecer e como o sistema deve se comportar em determinadas situações.

Ex.: O sistema deve registrar locações, indicando o cliente, os itens locados, a data da locação, a data de devolução e o valor da locação.

Não Funcionais: descrevem restrições sobre as funções oferecidas, tais como restrições de tempo, de uso de recursos etc. Alguns requisitos não funcionais dizem respeito ao sistema como um todo e não a funcionalidade específica.

Ex.: A consulta ao acervo da locadora deve estar disponível pela Internet, a partir dos principais navegadores disponíveis no mercado. (requisito de portabilidade)

Especificação de Requisitos

Tipos de Requisitos

Além desses requisitos, é importante considerar também *Requisitos de Domínio*.

Requisitos de domínio (ou regras de negócio) são provenientes do domínio de aplicação do sistema e refletem características e restrições desse domínio. Eles são derivados do negócio que o sistema se propõe a apoiar e podem restringir requisitos funcionais existentes ou estabelecer como cálculos específicos devem ser realizados, refletindo fundamentos do domínio de aplicação (SOMMERVILLE, 2011).

Ex.: Em um sistema de matrícula de uma universidade, uma importante regra de negócio diz que um aluno só pode se matricular em uma turma de uma disciplina se ele tiver cumprido seus pré-requisitos.

Especificação de Requisitos

Tipos de Requisitos

Toda organização opera de acordo com um extenso conjunto de políticas corporativas, leis, padrões industriais e regulamentações governamentais. Tais princípios de controle são coletivamente designados por regras de negócio.

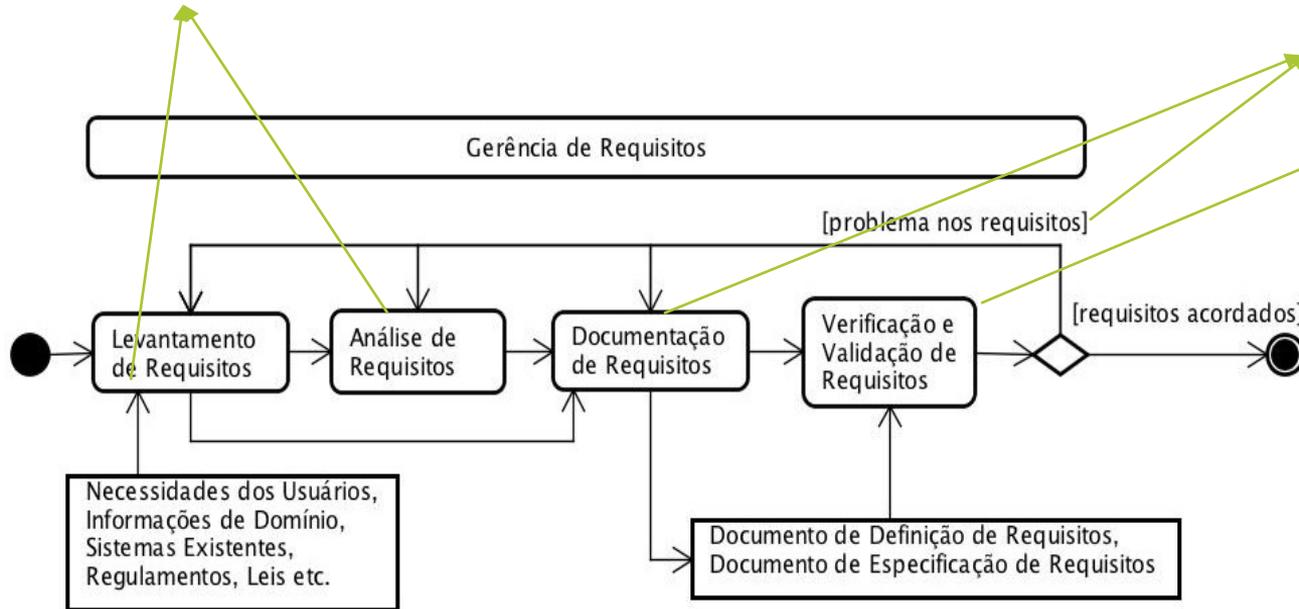
- Uma regra de negócio é uma declaração que define ou restringe algum aspecto do negócio, com o propósito de estabelecer sua estrutura ou controlar ou influenciar o comportamento do negócio.
- Sistemas de informação tipicamente precisam fazer cumprir as regras de negócio. Ao contrário dos requisitos funcionais e não funcionais, a maioria das regras de negócio origina-se fora do contexto de um sistema específico. Assim, as regras a serem tratadas pelo sistema precisam ser identificadas, documentadas e associadas aos requisitos do sistema em questão.

Especificação de Requisitos

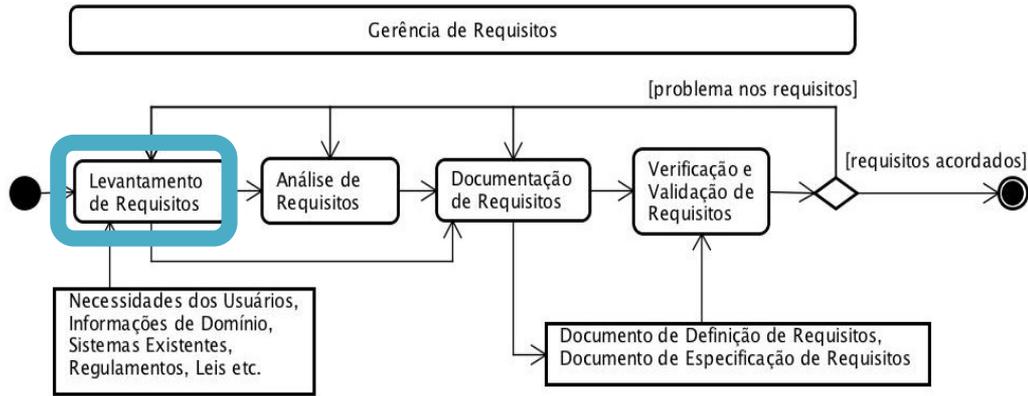
Processo de Engenharia de Requisitos

Definem O QUE o software deve fazer.

Tratados pela
Documentação,
Garantia da
Qualidade e
Gerência de
Configuração

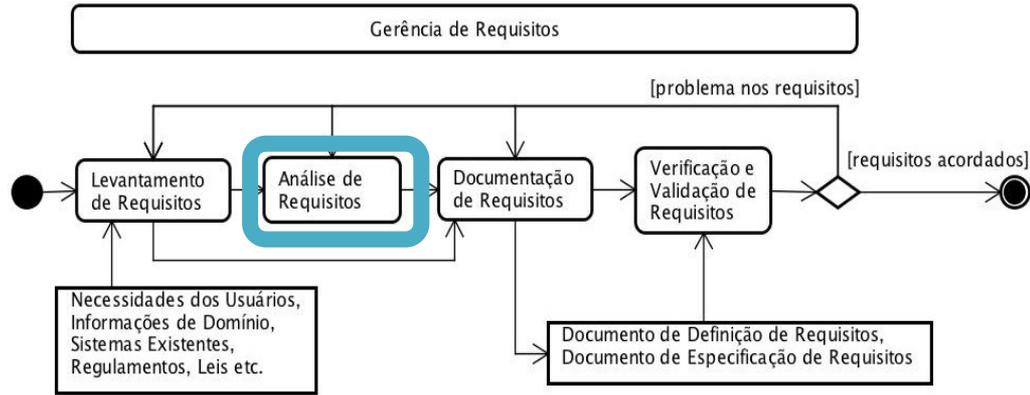


Especificação de Requisitos



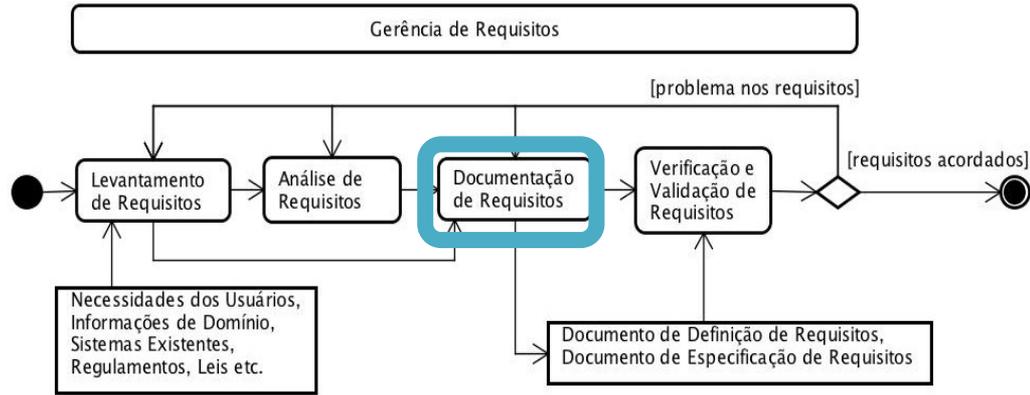
- Nesta fase, os usuários, clientes e especialistas de domínio são identificados e trabalham junto com os engenheiros de requisitos para entender a organização, o domínio da aplicação, os processos de negócio a serem apoiados, as necessidades que o software deve atender e os problemas e deficiências dos sistemas atuais. Os diferentes pontos de vista dos participantes do processo, bem como as oportunidades de melhoria, restrições existentes e problemas a serem resolvidos devem ser levantados.

Especificação de Requisitos



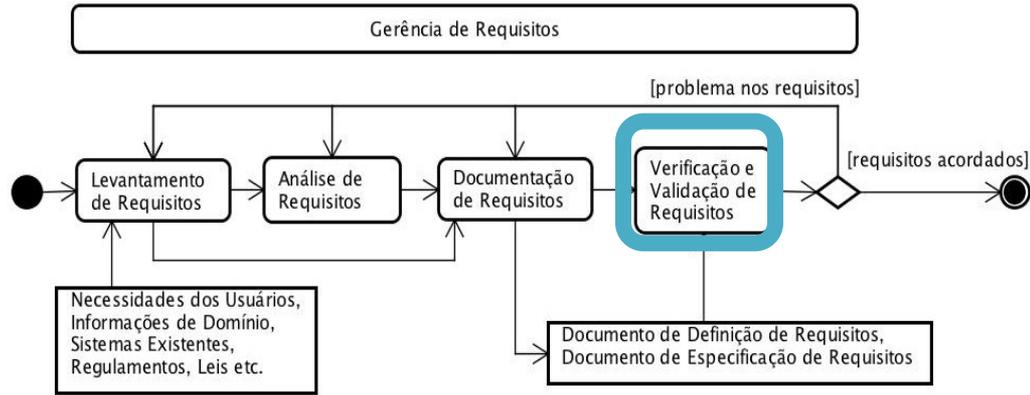
- Esta fase visa estabelecer um conjunto acordado de requisitos consistentes e sem ambiguidades, que possa ser usado como base para as atividades subsequentes do processo de software. Para tal, diversos tipos de modelos são construídos. Assim, a análise de requisitos é essencialmente uma atividade de modelagem. A análise de requisitos pode incluir, ainda, negociação entre usuários para resolver conflitos detectados.

Especificação de Requisitos



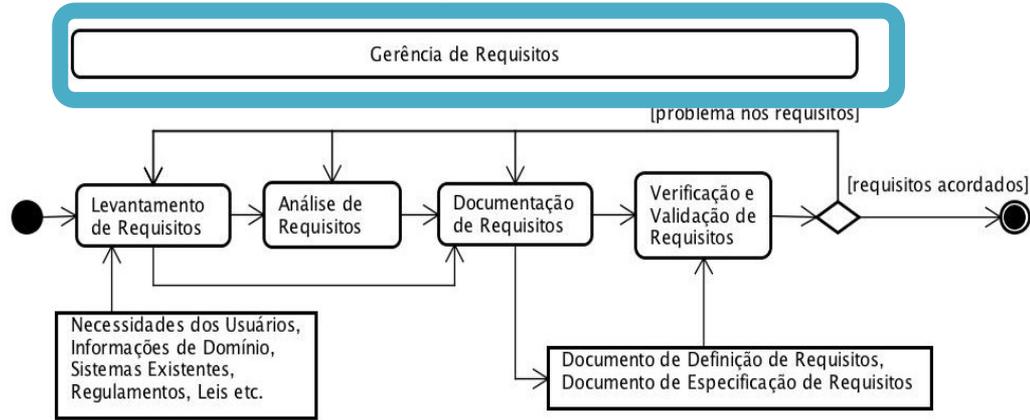
- Esta fase é a atividade de representar os resultados da Engenharia de Requisitos em um documento (ou conjunto de documentos), contendo os requisitos do software e os modelos que os especificam.

Especificação de Requisitos



- A **verificação** de requisitos avalia se os requisitos estão sendo tratados de forma correta, de acordo com padrões previamente definidos, sem conter requisitos ambíguos, incompletos ou, ainda, requisitos incoerentes ou impossíveis de serem testados. Já a **validação** diz respeito a avaliar se os requisitos do sistema estão corretos, ou seja, se os requisitos e modelos documentados atendem às reais necessidades de usuários e clientes.

Especificação de Requisitos



- Essa fase se preocupa em gerenciar as mudanças nos requisitos já acordados, manter uma trilha dessas mudanças e gerenciar os relacionamentos entre os requisitos e as dependências entre requisitos e outros artefatos produzidos no processo de software, de forma a garantir que mudanças nos requisitos sejam feitas de maneira controlada e documentada.

Levantamento ou Elicitação de Requisitos

- O levantamento de requisitos é uma atividade de descoberta de informações.
- Diversas fontes podem ser pesquisadas, dentre elas material bibliográfico, manuais de funcionamento da organização, documentos da organização e as pessoas envolvidas, dentre elas especialistas do domínio, clientes e usuários.
- É uma atividade complexa que não se resume somente a perguntar às pessoas o que elas desejam e também não é uma simples transferência de conhecimento.

Levantamento ou Elicitação de Requisitos

- Algumas técnicas que podem ser empregadas: entrevistas, questionários, prototipação, investigação de documentos, observação, dinâmicas de grupo etc.
- Importante:
- Enfoque em uma visão do cliente / usuário.
- Ainda não se está procurando definir a estrutura interna do sistema, mas sim procurando saber que funcionalidades o sistema deve oferecer ao usuário e que restrições elas devem satisfazer.

Levantamento ou Elicitação de Requisitos

Quatro dimensões devem ser consideradas:

- Entendimento do domínio da aplicação: entendimento geral da área na qual o software a ser desenvolvido está inserido;
- Entendimento do problema: entendimento dos detalhes do problema específico a ser resolvido com o auxílio do sistema a ser desenvolvido;
- Entendimento do negócio: entender como o sistema afetará a organização e como contribuirá para que os objetivos do negócio e os objetivos gerais da organização sejam atingidos;
- Entendimento das necessidades e das restrições dos interessados: entender as demandas de apoio para a realização do trabalho de cada um dos interessados no sistema, entender os processos de trabalho a serem apoiados pelo sistema e o papel de eventuais sistemas existentes na execução e condução dos processos de trabalho.

Análise de Requisitos

- Enfoca a estrutura interna do sistema (procura definir o que o sistema tem de ter internamente para tratar adequadamente os requisitos levantados).
- É uma atividade de construção de modelos.
- Um modelo é uma representação de alguma coisa do mundo real, uma abstração da realidade, e, portanto, representa uma seleção de características do mundo real relevantes para o propósito do sistema em questão.

Análise de Requisitos

Modelos são fundamentais no desenvolvimento de sistemas.

Tipicamente eles são construídos para:

- focar os aspectos chave, em detrimento de detalhes irrelevantes;
- possibilitar o estudo do comportamento do sistema;
- facilitar a comunicação entre membros da equipe de desenvolvimento e clientes e usuários;
- possibilitar a discussão de correções e modificações com o usuário;
- servir como base para a tomada de decisão;
- Fornecem uma estrutura para as atividades da ER, sendo a base para a geração do Documento de Especificação de Requisitos

Análise de Requisitos

- Exemplos de modelos como abstrações da realidade



Mapa Político



Mapa Turístico

Análise de Requisitos

No desenvolvimento de sistemas, há duas perspectivas principais:

- *Perspectiva estrutural:* tem por objetivo descrever as informações que o sistema deve representar e gerenciar. Provê uma visão estática das informações que o sistema necessita tratar. Ex.: diagramas de classes e modelos ER.
- *Perspectiva comportamental:* visa especificar as ações (funcionalidades / serviços) que o sistema deve prover, bem como o comportamento de certas entidades do modelo estrutural em relação a essas ações. Ex.: diagramas de casos de uso, diagramas de estados, etc.

Análise de Requisitos

Modelos de sistemas podem ser construídos em diferentes níveis de abstração

- *Conceitual*: considera características do sistema independentes do ambiente computacional (hardware e software) no qual o sistema será implementado. Modelos conceituais são construídos na atividade de análise de requisitos.
- *Modelo lógico*: trata características dependentes de um determinado *tipo* de plataforma computacional. Essas características são, contudo, independentes de produtos específicos. Tais modelos são típicos da fase de projeto.
- *Modelo físico*: leva em consideração características dependentes de uma plataforma computacional específica, isto é, uma linguagem e um compilador específicos, um sistema gerenciador de bancos de dados específico, o hardware de um determinado fabricante etc. Tais modelos podem ser construídos tanto na fase de projeto detalhado quanto na fase de implementação.

Análise de Requisitos

- Para realizar a Análise de Requisitos, é preciso escolher o paradigma de desenvolvimento a ser seguido.
- Paradigmas de desenvolvimento estabelecem a forma de se ver o mundo e, portanto, definem as características básicas dos modelos a serem construídos.

Por exemplo:

Paradigma Estruturado: adota uma visão de desenvolvimento baseada em um modelo entrada-processamento-saída. No paradigma estruturado, os dados são considerados separadamente das funções que os transformam e a decomposição funcional é usada intensamente.

Paradigma Orientado a Objetos: parte do pressuposto que o mundo é povoado por objetos, ou seja, a abstração básica para se representar as coisas do mundo são os objetos.

Documentação de Requisitos

- Os requisitos devem ser registrados em um Documento de Definição de Requisitos.



Introdução

Propósito do Sistema

Descrição do Minimundo

Requisitos do Usuário

Entendível pelo usuário

Documentação de Requisitos

- Estrutura do Documento:
 - **Introdução** (*contém a organização do documento*)
 - **Propósito do Sistema** (*contém o propósito do sistema descrito de forma bastante objetiva – tipicamente, em um parágrafo*)
 - **Descrição do Minimundo** (*visão geral do domínio, do problema a ser resolvido, bem como as principais ideias do cliente sobre o sistema a ser desenvolvido*)
 - **Requisitos de Usuário** (*requisitos de usuário em linguagem natural*)

Documentação de Requisitos

Requisitos Funcionais (o que o sistema deve fazer?)

Ex:

Identificador	Descrição	Prioridade	Requisitos Relacionados
RF01	O sistema deve registrar locações, indicando o cliente e os itens locados, bem como a data e o valor da locação e a data de devolução prevista de cada item.	Alta	RF06,RF09,RN01,RN08,RN11,RN12,RN13,RN14,RN16,RNF03,RNF04
RF02	O sistema deve registrar devoluções, indicando os itens devolvidos e a data de devolução	Alta	RF01,RN02,RN08,RNF03,RNF04
RF03	O sistema deve registrar os pagamentos de locações	Alta	RF01,RF02,RN01,RN08,RN09,RNF04,RNF05,
RF04	O sistema deve registrar a reserva de filmes a clientes, permitindo indicar, ainda, o tipo de mídia desejado	Média	RF06,RF09,RF01,RF02,RN10,RN16,RN17,RNF04
RF05	O sistema deve permitir o cancelamento de uma reserva, tanto pelo usuário, quanto automaticamente pelo sistema, quando expirado o prazo para retirada do item, de acordo com política da empresa.	Média	RF05,RN03,RN15,RNF04

Documentação de Requisitos

Regras de Negócio (regras que devem ser obedecidas pelo sistema)

Ex:

Identificador	Descrição	Prioridade	Requisitos Relacionados
RN01	O sistema deve permitir que sejam dados descontos nas locações, bem como que sejam ampliados os prazos de devolução de itens, em função da política da empresa.	Média	RN14
RN02	O sistema deve cobrar multa para itens devolvidos com atraso, segundo a seguinte fórmula: $M = n * VL$, onde M é o valor da multa, n é o número de dias de atraso e VL é o valor de locação do item.	Alta	
RN03	Reservas canceladas pelo sistema não deverão ser efetivamente excluídas pelo sistema, mas sim marcadas como expiradas.	Média	
RN04	O sistema deve manter o histórico de locações e, portanto, clientes que tenham feito locações não poderão ser excluídos.	Alta	

Documentação de Requisitos

Tipos de regras de negócio

Fatos ou invariantes *Ex.: Todo pedido tem uma taxa de remessa.*

Restrições *Ex.: Um aluno só pode tomar emprestado, concomitantemente, de um a três livros.*

Ativadores de Ações *Ex.: Se a data para retirada do livro é ultrapassada e o livro não é retirado, então a reserva é cancelada.*

Inferências *Ex.: Se o usuário não devolve um livro dentro do prazo estabelecido, então ele torna-se um usuário inadimplente.*

Computações *Ex.: Aplica-se um desconto progressivo se mais do que 10 unidades forem adquiridas. De 10 a 19, o desconto é de 10%. Acima de 19, o desconto é de 25%.*

Documentação de Requisitos

Tipos de regras de negócio

Restrições de Integridade em Modelos de Dados (também são regras de negócio) *Ex: Um aluno só pode ser matricular em turmas de disciplinas que compõem a grade curricular do curso que esse aluno cursa.*

Documentação de Requisitos

Ao contrário de requisitos funcionais e não funcionais, regras de negócio não são passíveis de serem capturadas por meio de perguntas simples e diretas, tal como “Quais são suas regras de negócio?”

Normalmente surgem quando está se tentando entender os requisitos. Muitas só vão surgir quando os requisitos forem detalhados. Algumas questões que podem ajudar:

Políticas: Por que é necessário fazer isso desse jeito?

Regulamentações: O que o governo requer?

Fórmulas: Como este valor é calculado?

Modelos de Dados: Como essas entidades de dados estão relacionadas?

Ciclo de Vida de Objetos: O que causa uma mudança no estado desse objeto?

Decisões de Atores: O que o usuário pode fazer a seguir?

Decisões de Sistema: Como o sistema sabe o que fazer a seguir?

Eventos: O que pode (e não pode) acontecer?

Documentação de Requisitos

Requisitos Não Funcionais (tipicamente, restrições a serem obedecidas em relação a: segurança, portabilidade, desempenho, usabilidade, interoperabilidade,...)

Ex:

Identificador	Descrição	Categoria	Escopo	Prioridade	Requisitos Relacionados
RNF01	O sistema deve controlar o acesso às funcionalidades. Funcionalidades para controlar o acervo da locadora devem ser restritas a administradores. Funcionalidades de atendimento a clientes devem estar restritas a atendentes. Funcionalidades de consulta ao acervo devem estar disponíveis na Internet.	Segurança de Acesso	Sistema	Alta	
RNF02	A consulta ao acervo deve estar disponível pela Internet, a partir dos principais navegadores disponíveis no mercado.	Portabilidade	Funcionalidade	Média	
RNF03	Os itens devem ser identificados por um código de barras, sendo possível a leitura dos mesmos usando dispositivos de leitores de código de barras.	Facilidade de Operação	Funcionalidade	Alta	

Documentação de Requisitos

Há muitos atributos de qualidade que podem ser importantes para um sistema.

Uma boa estratégia para levantar requisitos não funcionais de produto consiste em explorar uma lista de potenciais atributos de qualidade que a grande maioria dos sistemas deve apresentar em algum nível.

Por exemplo, o modelo de qualidade externa e interna de produtos de software definido na norma ISO/IEC 25010, utilizado como referência para a avaliação de produtos de software, define oito características de qualidade:

Documentação de Requisitos

- *Aptidão Funcional (Functional Suitability)*: grau em que o produto provê funções que satisfazem às necessidades explícitas e implícitas, quando usado em condições especificadas.
- *Confiabilidade/Disponibilidade*: grau em que o produto executa as funções especificadas com um comportamento consistente com o esperado, por um período de tempo. A confiabilidade está relacionada com os defeitos que um produto apresenta e como este produto se comporta em situações consideradas fora do normal.
- *Usabilidade*: grau em que o produto apresenta atributos que permitem que o mesmo seja entendido, aprendido e usado, e que o tornem atrativo para o usuário

Documentação de Requisitos

- *Eficiência de Desempenho*: capacidade de o produto manter um nível de desempenho apropriado em relação aos recursos utilizados em condições estabelecidas;
- *Segurança*: grau em que informações e dados são protegidos contra acesso por pessoas ou sistemas não autorizados, bem como grau em que essas informações e dados são disponibilizados para as pessoas ou sistemas com acesso autorizado.
- *Compatibilidade*: capacidade do produto de software de trocar informações com outras aplicações e/ou compartilhar o mesmo ambiente de hardware ou software;

Documentação de Requisitos

- *Manutenibilidade*: capacidade do produto de software de ser modificado.
- *Portabilidade*: refere-se à capacidade do software ser transferido de um ambiente de hardware, software ou operacional para outro;

Documentação de Requisitos

- RNFs podem ser classificados em:
- *Requisitos de produto*: especificam o comportamento do produto (sistema). Referem-se a atributos de qualidade que o sistema deve apresentar, tais como confiabilidade e usabilidade.
- *Requisitos organizacionais*: são derivados de metas, políticas e procedimentos das organizações do cliente e do desenvolvedor. Incluem requisitos de processo (padrões de processo e modelos de documentos que devem ser usados), requisitos de implementação (tal como a linguagem de programação a ser adotada)etc.
- *Requisitos externos*: referem-se a todos os requisitos derivados de fatores externos ao sistema e seu processo de desenvolvimento. Podem incluir requisitos de interoperabilidade com sistemas de outras organizações, requisitos legais (tais como requisitos de privacidade) e requisitos éticos.

Documentação de Requisitos

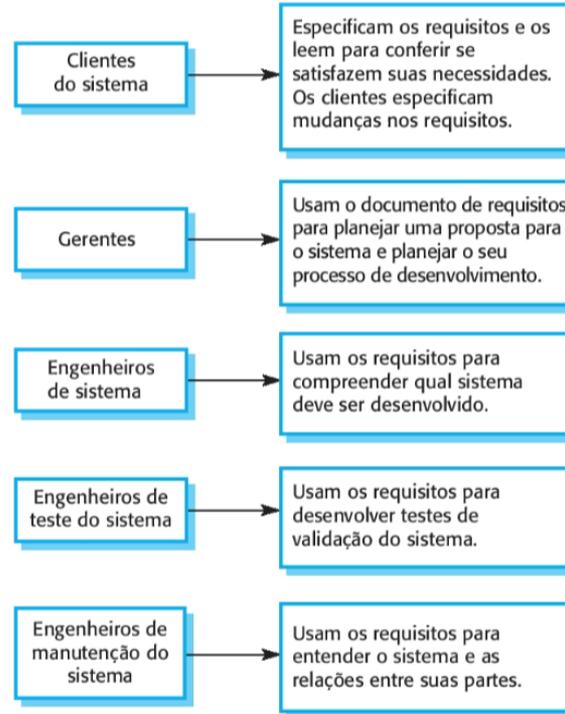
- Como escrever requisitos?
- Identificar os requisitos pelo seu tipo e número (ex.: RF01, RNF02, RN05)
- Usar frases do tipo:
 - O sistema deve.... (função ou característica requerida)
 - O sistema pode.... (função ou característica desejada)
 - O sistema não deve...
- Escrever frases completas, com a gramática, ortografia e pontuação correta. Procurar manter frases e parágrafos curtos e diretos.
- Usar os termos consistentemente. Se necessário, defina-os em um glossário.
- Preferir a voz ativa (o sistema deve fazer alguma coisa) à voz passiva (alguma coisa deve ser feita).

Documentação de Requisitos

- Sempre que possível, identificar o tipo de usuário. Ex.: se o usuário no caso for o caixa do banco, indicar claramente “o caixa do banco deve [...]”.
- Evitar termos vagos, que conduzam a requisitos ambíguos e não testáveis, tais como “rápido”, “adequado”, “fácil de usar” etc.
- Escrever requisitos em um nível consistente de detalhe.
- Escrever requisitos individualmente testáveis. Um requisito bem escrito deve permitir a definição de um pequeno conjunto de testes para verificar se o requisito foi corretamente implementado.
- Evitar longos parágrafos narrativos que contenham múltiplos requisitos. Divida um requisito desta natureza em vários menores.

Documentação de Requisitos

Usuários de um documento de requisitos.



- Fonte: (SOMMERVILLE, 2019)

Verificação e Validação de Requisitos

- A validação de requisitos é o processo de avaliar se os requisitos realmente definem o sistema que o cliente realmente deseja. Ele se sobrepõe à análise, pois se preocupa em encontrar problemas com os requisitos. A validação de requisitos é importante porque erros em um documento de requisitos podem levar a grandes custos de retrabalho quando esses problemas são descobertos durante o desenvolvimento ou depois que o sistema está em serviço.
- O custo de corrigir um problema de requisitos fazendo uma alteração no sistema é geralmente muito maior do que reparar erros de projeto ou codificação. A razão para isso é que uma mudança nos requisitos geralmente significa que o design e a implementação do sistema também devem ser alterados. Além disso, o sistema deve ser testado novamente.
- Durante o processo de avaliação de requisitos, diferentes tipos de avaliações devem ser realizados nos requisitos do documento de requisitos. Essas avaliações incluem:
- Fonte: (SOMMERVILLE, 2019)

Verificação e Validação de Requisitos

- 1. *Checagens de validade*: Um usuário pode pensar que um sistema é necessário para executar certas funções. No entanto, mais reflexão e análise podem identificar funções adicionais ou diferentes que são necessárias. Os sistemas têm diversas partes interessadas com diferentes necessidades e qualquer conjunto de requisitos é inevitavelmente um compromisso em toda a comunidade de partes interessadas.
- 2. *Checagens de consistência*: Os requisitos do documento não devem entrar em conflito. Ou seja, não deve haver restrições contraditórias ou descrições diferentes da mesma função do sistema.
- 3. *Checagens de integridade*: O documento de requisitos deve incluir requisitos que definam todas as funções e as restrições pretendidas pelo usuário do sistema.
- 4. *Checagens de realismo*: Usando o conhecimento da tecnologia existente, os requisitos devem ser checados para garantir que possam realmente ser implementados. Essas checagens também devem levar em conta o orçamento e o cronograma para o desenvolvimento do sistema.
- Fonte: (SOMMERVILLE, 2019)

Verificação e Validação de Requisitos

- 5. *Verificabilidade*: Para reduzir o potencial de disputa entre o cliente e o contratado, os requisitos do sistema devem sempre ser escritos de forma que possam ser testados. Isso significa que você deve ser capaz de escrever um conjunto de testes que possa demonstrar que o sistema entregue atende a cada requisito especificado.
- Fonte: (SOMMERVILLE, 2019)

Referências

- SOMMERVILLE, I. Engenharia de Software. 10. ed. São Paulo: Pearson Prentice Hall, 2019
- Notas de Aula do Prof. Ricardo Falbo
- Slides da Professora Monalessa Perini Barcellos (<http://www.inf.ufes.br/~monalessa/ensino/engenharia-de-software/>)

Engenharia de Software – 2020/2 EARTE

Jordana S. Salamon

jordana.salamon@ufes.br

jssalamon@inf.ufes.br

jordanasalamon@gmail.com

<http://inf.ufes.br/~jssalamon>

Departamento de Informática

Centro Tecnológico

Universidade Federal do Espírito Santo

