



PRINCÍPIOS DA ENGENHARIA DE SOFTWARE- AULA 07

Prof.: Franklin M. Correia

1

NOS CAPÍTULOS ANTERIORES....

- Planejamento e Gerenciamento de Riscos
- Atividade programada

O QUE TEMOS PARA HOJE!!

- Análise de Requisitos
- Atividade programada

ANÁLISE DE REQUISITOS

- Analisar (v) 1. Decompor um todo em partes, componentes; fazer análise 2. Observar, examinar com minúcia; esquadrinhar 3. Examinar Criticamente.
- Requisito (s.m) 1. Condição necessária para obtenção de certo objetivo, ou para preenchimento de certo fim; quesito 2. Exigência necessária para certos efeitos.

ANÁLISE DE REQUISITOS

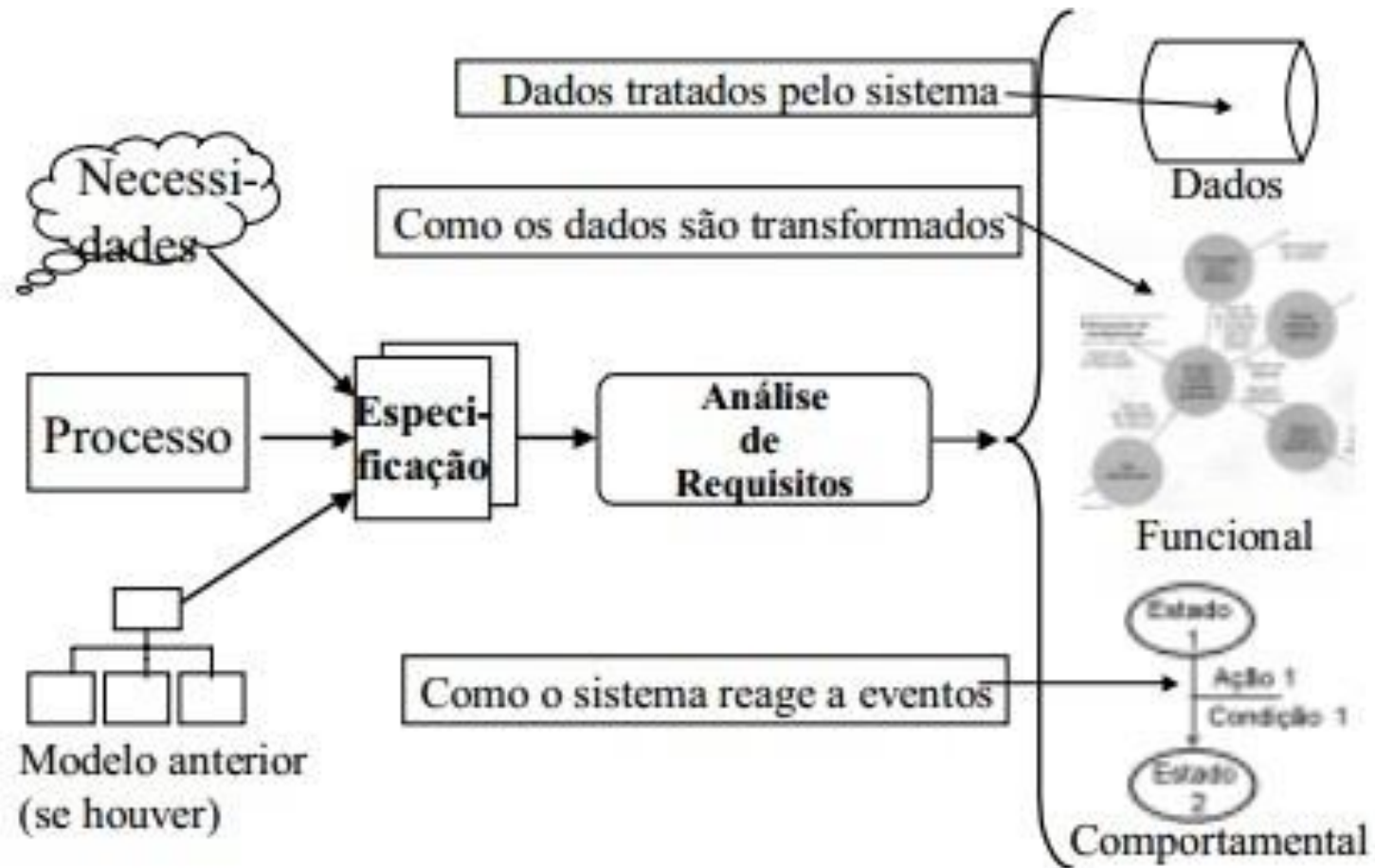
- Também chamada de Análise de Sistemas ou Engenharia de Requisitos.
 - Parte dos requisitos obtidos e revisados e do Plano de Projeto. Visa fornecer representações (modelos) de "Informação" e de "função" que:
 - Possam ser transformadas (traduzidas) em "projeto" de dados, arquitetura, interface e procedimentos;
 - Sirvam como um meio de determinar a qualidade do software.

ANÁLISE DE REQUISITOS

- O Engenheiro de Software constrói modelos usando os requisitos elicitados do usuário.
- Para entender e validar os requisitos eliciados, precisamos analisa-los de diferentes pontos de vista:
 - Aspectos de dados
 - Aspectos funcionais
 - Aspectos comportamentais

ANÁLISE DE REQUISITOS

FLUXO DE ATIVIDADES



ANÁLISE DE REQUISITOS

○ Objetivos

1. Descrever o que o usuário requisitou, de maneira técnica;
2. Estabelecer uma base para a criação do Projeto de Software;
3. Definir um conjunto de requisitos que precisam ser validados quando o software estiver pronto.

PRINCÍPIOS DA ANÁLISE DE REQUISITOS

1. Analisar (dividir) para entender o domínio das Informações de um problema que deve ser representado e compreendido;
2. Criar modelos que descrevam as informações a serem tratadas, as funções que tratarão as informações e o comportamento do sistema que será desenvolvido;
3. Os Modelos (e o problema) devem ser divididos em partições (analisados) de maneira que revele os detalhes em camadas (hierarquicamente);
4. O foco do processo deve mover-se da informação essencial para os detalhes.

ANÁLISE DE REQUISITOS - ESPECIFICAÇÃO

- A especificação é o produto (consequência) da Análise de Requisitos. Cada requisito exige uma especificação (como deve ser feito e como será validado);
- É o documento fundamental da Engenharia de Software, pois especifica tudo O QUE deverá ser projetado e construído;
- Permite que o requisitos seja gerenciáveis (planejamento, controle e feed-back);
- A revisão dos requisitos é fundamental para garantir que usuário e analista tenham a mesma “percepção” do sistema.

ANÁLISE DE REQUISITOS – PRINCÍPIOS DA ESPECIFICAÇÃO

1. Separar Funcionalidade da Implementação (não conter características de implementação).
2. Usar uma linguagem de Especificação de Sistema Orientada a Processos.
3. A Especificação do deve abranger o sistema do qual o software é um componente.
4. Especificação deve abranger o ambiente(s) no(s) qual(is) o sistema opera

ANÁLISE DE REQUISITOS – PRINCÍPIOS DA ESPECIFICAÇÃO

5. Uma Especificação de Requisitos deve ser um modelo cognitivo.
6. Uma Especificação deve ser operacional
7. A especificação do sistema deve ser expansível
8. Uma especificação deve ser localizada e fracamente acoplada

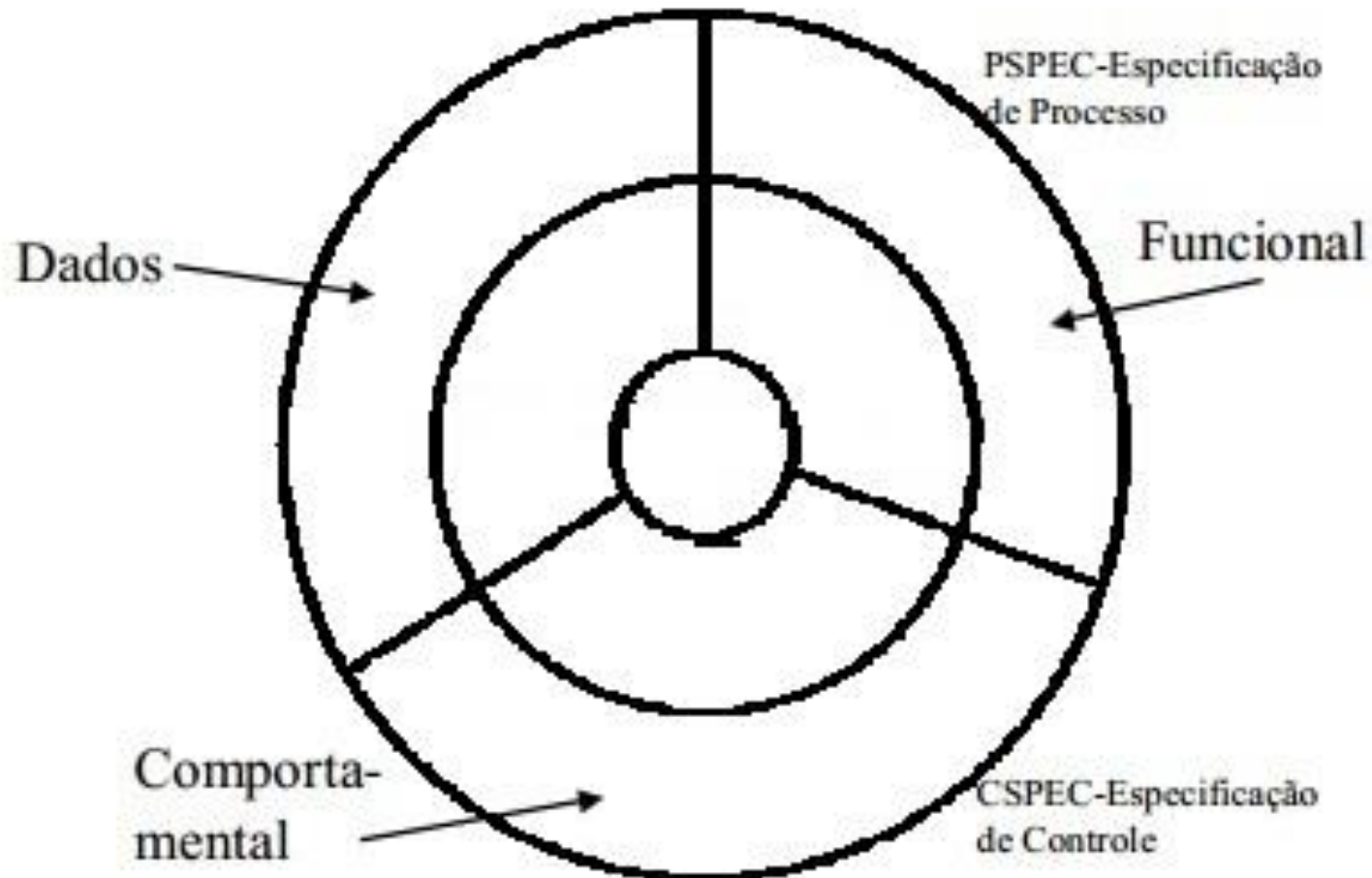
ANÁLISE DE REQUISITOS – MODELOS PREDOMINANTES

- Inicia-se com uma série de tarefas de modelagem, que pretendem obter uma especificação completa de requisitos e projeto que represente o software a ser construído.
- A Análise é um conjunto de modelos que será a primeira representação técnica de um sistema. Os modelos predominantes são:
 - Análise Estruturada (Metodologia Estruturada)
 - Análise Orientada a Objetos (Met. Orient. a Objetos)

ANÁLISE DE REQUISITOS

- Modelagem de dados:
 - Definir seus dados, atributos e relacionamentos.
- Modelagem funcional:
 - Definir como os dados são transformados (processados) dentro do sistema.
- Modelagem comportamental:
 - Definir como os eventos trata o sistema.

OS ELEMENTOS DO MODELO DE ANÁLISE



ANÁLISE (ESTRUTURADA) DE REQUISITOS

- Dicionário de Dados:
 - Repositório que contém a descrição de todos os objetos de dados alimentados ou produzidos pelo software.
- Diagrama Entidade-Relacionamento:
 - Mostra as relações (cardinalidade e modalidade) existentes entre os itens de dados.
- Diagrama de Transição de Estados:
 - Destaca o evento e seu impacto sobre o sistema (fluxo de controle)

ANÁLISE DE REQUISITOS – FERRAMENTAS DE APOIO

- Ferramentas CASE (Computer-Aided Software Engineering)
 - Aumento de Produtividade e Qualidade
- Software de Apoio ao Desenvolvimento de Software:
 - Gerenciamento de Projetos (MS-Project)
 - Análise e Projeto (Rose; System Architecture; OracleDesigner; ErWin; BpWin; Telelogic, System Architect etc.)
 - Geradores de Código (System Architect -> Dr Case)

ANÁLISE DE REQUISITOS - FERRAMENTAS DE APOIO

- Software de Apoio ao Desenvolvimento de Software:
 - Apoio a Teste, Validação e Simulação
 - Gerenciamento de Configuração e Versão
 - Documentação de Software

MODELO DE ESPECIFICAÇÃO DE REQUISITOS

1. Introdução

1. Referência ao sistema
2. Objetivos
3. Restrições ao projeto de software

2. Descrição da informação

1. Representação do fluxo da informação
2. Representação do conteúdo da informação
3. Representação da estrutura da informação
4. Descrição das interfaces do sistema

MODELO DE ESPECIFICAÇÃO DE REQUISITOS

3. Descrição funcional

- Particionamento funcional
- Descrição funcional
 - Texto explicativo do processo
 - Restrições/limitações
 - Requisitos de desempenho
 - Restrições de projeto
 - Diagramas

MODELO DE ESPECIFICAÇÃO DE REQUISITOS

4. Descrição Comportamental

- Estados do Sistema
- Eventos e Ações

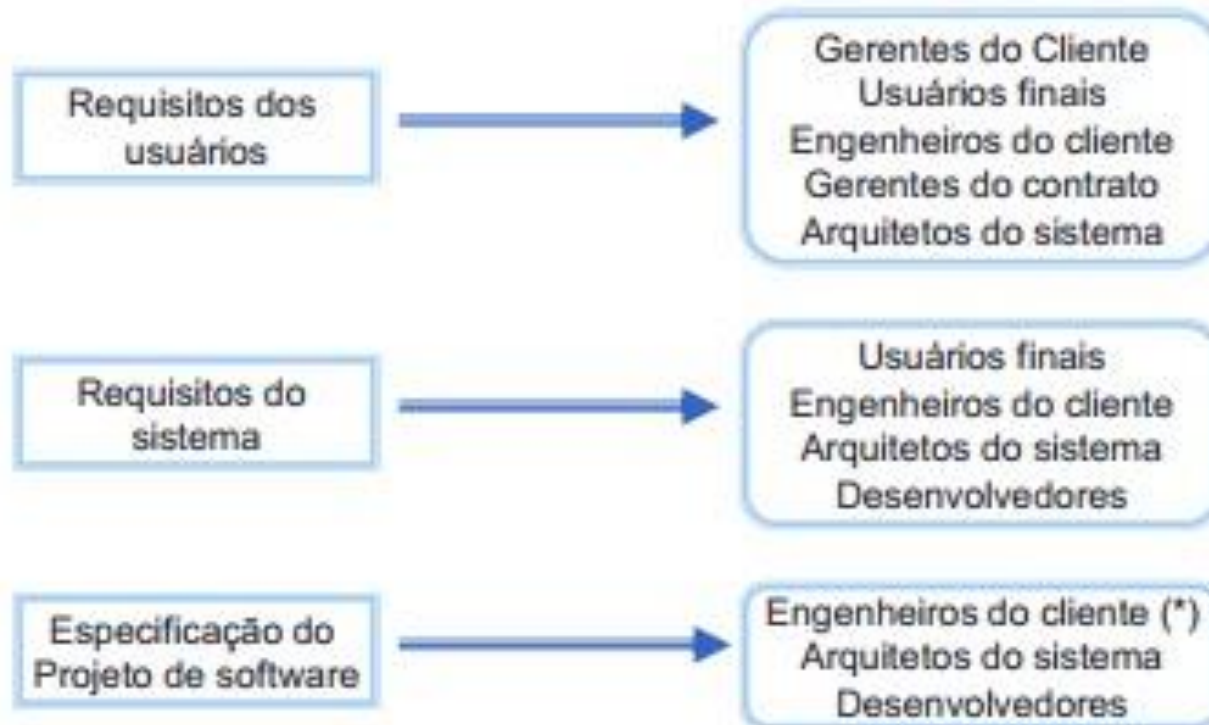
5. Critérios de validação

- Fronteiras de desempenho
- Classes de testes
- Resposta esperada do software
- Considerações especiais

EXEMPLO DE REQUISITOS

- Requisitos do usuário
 - O software deve prover meios de representar e acessar arquivos externos gerados por outras ferramentas.
- Requisitos do sistema
 - O usuário deve dispor de recursos para definir os tipos de arquivos externos.
 - Cada tipo de arquivo externo pode ter uma ferramenta associada que pode ser aplicada a ele.
 - Quando o usuário selecionar o ícone representando o arquivo externo, o efeito resultante é executar a ferramenta associada com o tipo de arquivo externo para o arquivo representado pelo ícone.

LEITORES DOS REQUISITOS



ANÁLISE DE REQUISITOS - TIPOS

○ Requisitos Funcionais

- Declarações de serviços que o sistema deveria prover, como o sistema deveria reagir a uma entrada particular e como o sistema deveria se comportar em uma situação particular.

○ Requisitos não funcionais

- Restrições de serviços ou funções oferecidas pelo sistema tais como restrição de tempo, processo de desenvolvimento, padrões, etc.

ANÁLISE DE REQUISITOS - TIPOS

○ Requisitos Funcionais

- Descreve funcionalidades ou serviços do sistema.
- Depende do tipo de software, usuários, e o tipo de sistema onde o software é usado.
- RF dos usuários podem ser declarações de alto-nível do que o sistema deveria fazer
- RF do sistema deve descrever os serviços do sistema em detalhes.

ANÁLISE DE REQUISITOS - FUNCIONAIS

- Exemplo: Sistema LYBSYS
 - Um sistema que prover uma interface única para várias bases de dados de artigos em diferentes bibliotecas.
 - Usuários podem pesquisar, fazer download e imprimir os artigos para estudo pessoal

ANÁLISE DE REQUISITOS - FUNCIONAIS

○ Exemplo:

- O usuário deve ser capaz de pesquisar em todas as bases de dados ou selecionar uma única base para sua busca.
- O sistema deve prover telas apropriadas para que os usuários possam ler os documentos acessados.
- Cada solicitação deve ter um único identificador (ID) para o qual o usuário deve ser capaz de copiar para a sua área de armazenamento permanente.

IMPRECISÕES NOS REQUISITOS

- Problemas podem surgir quando os requisitos não são estabelecidos precisamente.
- Requisitos ambíguos podem ser interpretados de diferentes formas pelos desenvolvedores e usuários.
- Por exemplo:
 - Considere o termo “Telas apropriadas”
 - Intenção do usuário: Telas com finalidades específicas para cada tipo de documento.
 - Interpretação do desenvolvedor: Prover telas de texto que mostra o conteúdo dos documentos.

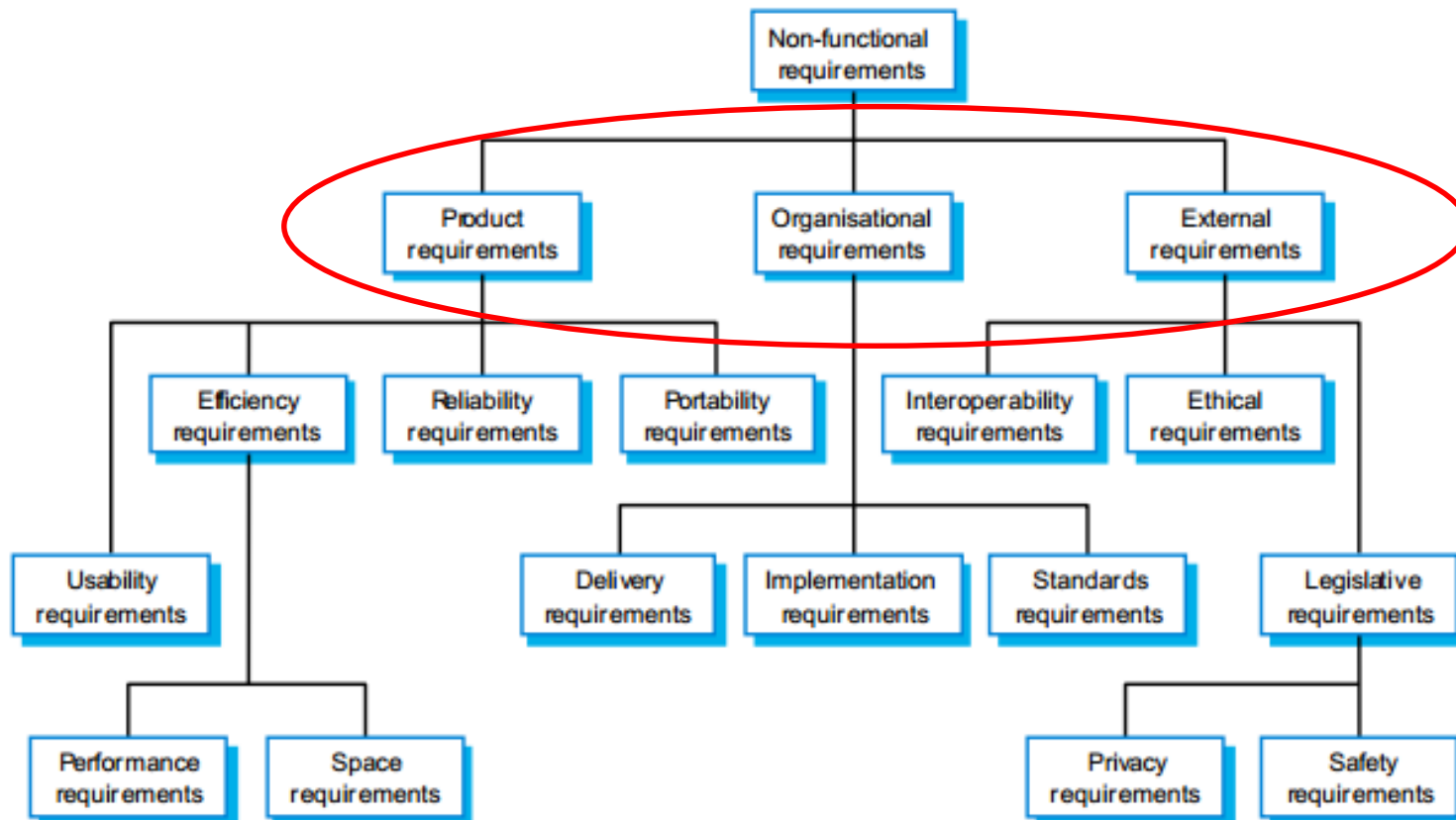
COMPLETUDE E CONSISTÊNCIA

- Os requisitos deveriam ser completos e consistentes.
- Completos:
 - Deveriam incluir descrições de todas as facilidades requeridas.
- Consistentes:
 - Não deveriam ter conflitos ou contradições nas descrições de suas facilidades.
- Na prática é muito difícil produzir um documento de requisitos completo e consistente.

ANÁLISE DE REQUISITOS – NÃO FUNCIONAIS

- Definem propriedades e restrições de sistemas, i.e. confiabilidade, tempo de resposta, armazenamento.
 - Restrições são capacidades de dispositivos de E/S, representações de sistemas, etc.
 - Requisitos de processos podem também ser obrigatórios como uma ferramenta CASE, uma linguagem de programação ou método de desenvolvimento.
 - RNF podem ser mais críticos que RF. Pois se eles não forem satisfeitos podem tornar o sistema inutilizado.

TIPO DE REQUISITOS NÃO FUNCIONAIS



TIPO DE REQUISITOS NÃO FUNCIONAIS

- Requisitos do produto
 - Especificam que o produto (software) deve ter um comportamento esperado, i.e., velocidade de execução, segurança, etc.
- Requisitos organizacionais
 - São consequência de políticas organizacionais e procedimentos, i.e. Padrões de processos, requisitos de implementação, etc.
- Requisitos externos
 - Surgem de fatores externos ao sistema e seu processo de desenvolvimento, i.e. Requisitos de interoperabilidade, requisitos legislativo, etc.

TIPO DE REQUISITOS NÃO FUNCIONAIS

- Requisitos de produtos
 - A interface do usuário para o sistema LIBSYS deve ser implementada com HTML simples sem uso de frames e Plugins.
- Requisitos organizacionais
 - O processo de desenvolvimento de sistema e documentos entregues devem estar em conformidade com os padrões e templates definidos em XYZCo-SP-STAN-95.
- Requisitos externo
 - O sistema não deve publicar nenhuma informação de cliente a não ser seu nome e número de referência para os operadores do sistema

REQUISITOS NÃO FUNCIONAIS E METAS

- RNF são difíceis de estabelecer precisamente e requisitos imprecisos são difíceis de verificar.
- Metas
 - Uma intenção geral do usuário tal como facilidade de uso.
- RNF verificáveis
 - Uma declaração usando alguma métrica que possa ser objetivamente testada.
- Metas são úteis para os desenvolvedores para satisfazer as intenções dos usuários do sistema.

REQUISITOS NÃO FUNCIONAIS E METAS

○ Meta do sistema

- O sistema deveria ser fácil de usar por controladores experientes e deveria ser organizado de tal forma que erros de uso do sistema sejam minimizados.

○ RNF verificável

- Controladores experientes deveriam ser capazes de usar todas as funções do sistema após um total de duas horas de treinamento.
- Após o treinamento o número médio de erros criados por estes controladores treinados não deveria exceder dois por dia.

REQUISITOS NÃO FUNCIONAIS E METAS

Property	Measure
Speed	Processed transactions/second User/Event response time Screen refresh time
Size	M Bytes Number of ROM chips
Ease of use	Training time Number of help frames
Reliability	Mean time to failure Probability of unavailability Rate of failure occurrence Availability
Robustness	Time to restart after failure Percentage of events causing failure Probability of data corruption on failure
Portability	Percentage of target dependent statements Number of target systems

REQUISITOS NÃO FUNCIONAIS E METAS

- Conflitos entre RNF são comuns em sistemas complexos
- Sistema de uma nave espacial
 - Para minimizar o peso, o numero de chips deveria ser minimizado,
 - Para minimizar o consumo de potência, menos chips deveriam ser usados
 - Mas, chips de baixa potência significam que mais chips devem ser usados.
- Qual o requisito mais crítico?

COMO DESCREVER OS REQUISITOS??

- Devem descrever RF e RNF de forma que sejam compreensíveis por todos os usuários, até mesmo os que não tem nenhum conhecimento técnico.
- Requisitos são definidos em linguagem natural (LN), tabelas e diagramas tal que eles possam ser entendidos por todos os usuários.

PROBLEMAS COM LINGUAGEM NATURAL

- Falta de clareza
 - Precisão X clareza
- Confusão dos requisitos
 - RNF, RF, objetivos e informações de projeto misturadas.
- Combinação de requisitos
 - Vários requisitos expressos como um único requisito.

PROBLEMAS COM LINGUAGEM NATURAL

○ Ambiguidade

- Os leitores e escritores dos requisitos devem interpretar as mesmas palavras da mesma forma. LN é naturalmente ambígua.

○ Alta flexibilidade

- A mesma coisa pode ser dita de diferentes formas na especificação.

○ Falta de modularização

- Estruturas em LN são inadequadas para estruturar requisitos do sistema.

DICAS PARA ESCREVER REQUISITOS

- Criar um formato padrão e usa-lo para todos os requisitos
 - Use a linguagem de maneira consistente. O uso da palavra “deve” para requisitos imperativas e o uso de “deveria” para requisitos desejáveis.
 - Use texto destacado para identificar partes chaves dos requisitos.
 - Evite o uso de jargões de computador.

ALTERNATIVAS PARA A LINGUAGEM NATURAL

Notação	Descrição
Linguagem natural estruturada	Depende de formulários padrões definidos para expressar a especificação dos requisitos
Linguagem de descrição de programas	Usa uma linguagem técnica com características abstratas para especificar os requisitos definindo um modelo operacional do sistema. Não é uma abordagem muito usada, embora seja útil para especificação de interfaces.
Notações Gráficas	Linguagem gráfica complementada com texto é usada para definir os requisitos funcionais.
Especificações matemáticas	Notações baseadas em conceitos matemáticos, tais como, máquinas de estado finitos ou conjuntos.

LINGUAGEM ESTRUTURADA

- A liberdade do escritor dos requisitos é limitada por templates (modelos) predefinidos.
- Todos os requisitos são escritos em um formato padrão.
- A vantagem é que é mantida a expressividade da linguagem natural, embora um grau de uniformidade seja imposto na especificação.

ESPECIFICAÇÃO BASEADA EM FORMULÁRIOS

Software de controle de insulina /SRS /3.3.2

Função: Calcular dose de insulina

Descrição: Calcule a dose de insulina quando o nível de açúcar estiver seguro (entre 3 e 7 unidades)

Entrada: Leitura atual do açúcar (r2), duas leituras anteriores (r0 e r1)

Fonte: Leitura atual (sensor) e outras leituras (memória)

Saída: CompDose – Dose calculada

Destino: Programa principal

Ação: CompDose é zero se o nível do açúcar é estável ou se o nível aumentou, mas a taxa de aumento está decrescendo. Se o nível está alto e a taxa de aumento também então CompDose é calculada dividindo a diferença entre o nível atual de açúcar e o nível anterior por 4 e arredondando o resultado. Se o resultado é zero então CompDose é igual ao valor mínimo da dose.

Depende de: Duas leituras anteriores.

Pre-condição: O reservatório de insulina contém pelo menos a dose máxima.

Pos-condição: r0 é trocado por r1 e então r1 é trocado por r2

Efeitos colaterais: Nenhum

ESPECIFICAÇÃO TABULAR

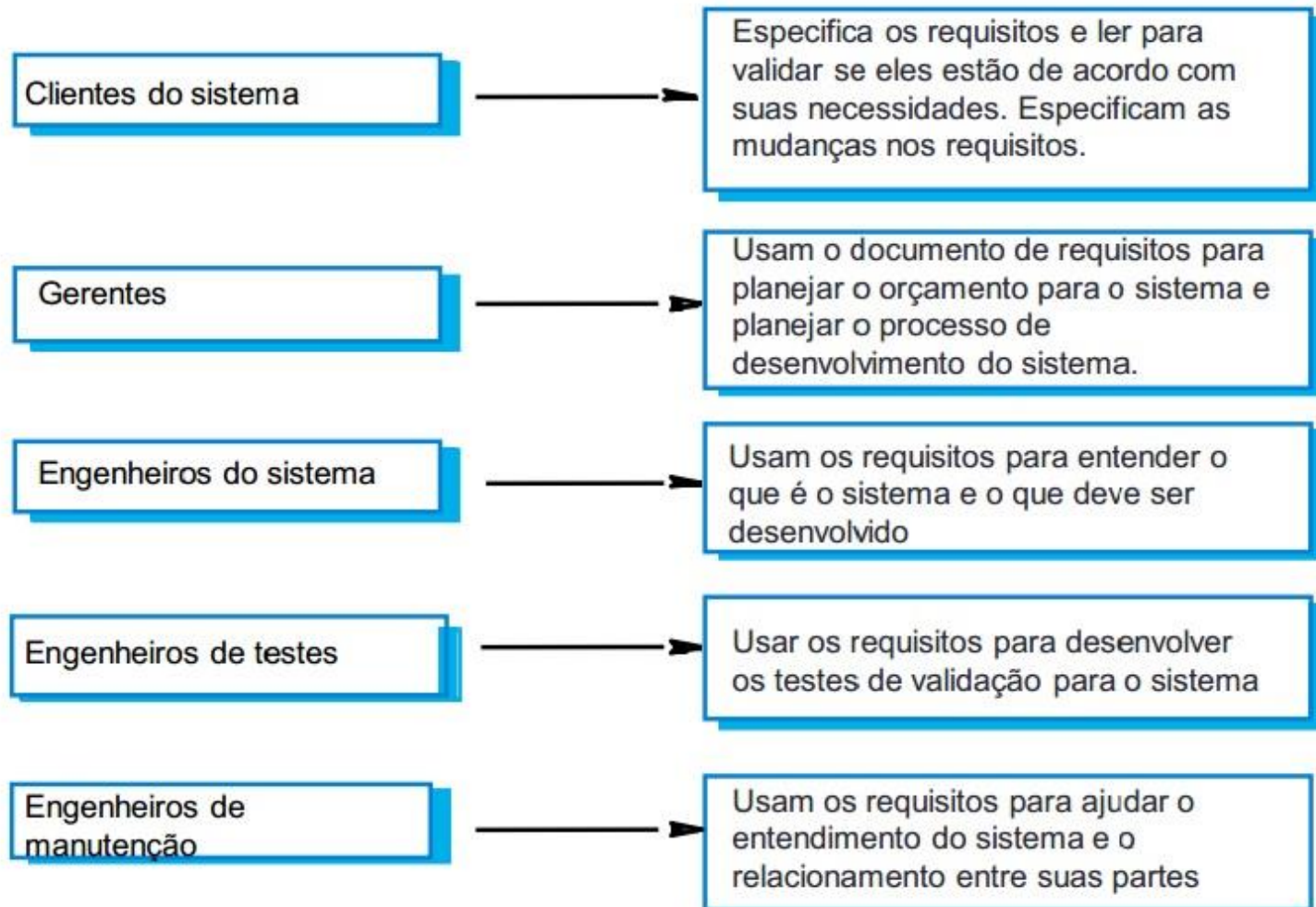
- Complementa a linguagem natural quando necessário definir alternativas.

Condição	Ação
Nível de açúcar diminuindo ($r_2 < r_1$)	CompDose = 0
Nível de açúcar estável ($r_2 = r_1$)	CompDose = 0
Nível de açúcar aumentando e a taxa de aumento diminuindo ($(r_2 - r_1) < (r_1 - r_0)$)	CompDose = 0
Nível de açúcar diminuindo aumentando e a taxa de aumento estável ou aumentando ($(r_2 - r_1) \geq (r_1 - r_0)$)	CompDose = round $((r_2 - r_1) / 4)$ Se o resultado arredondado = 0 então CompDose = DoseMinima

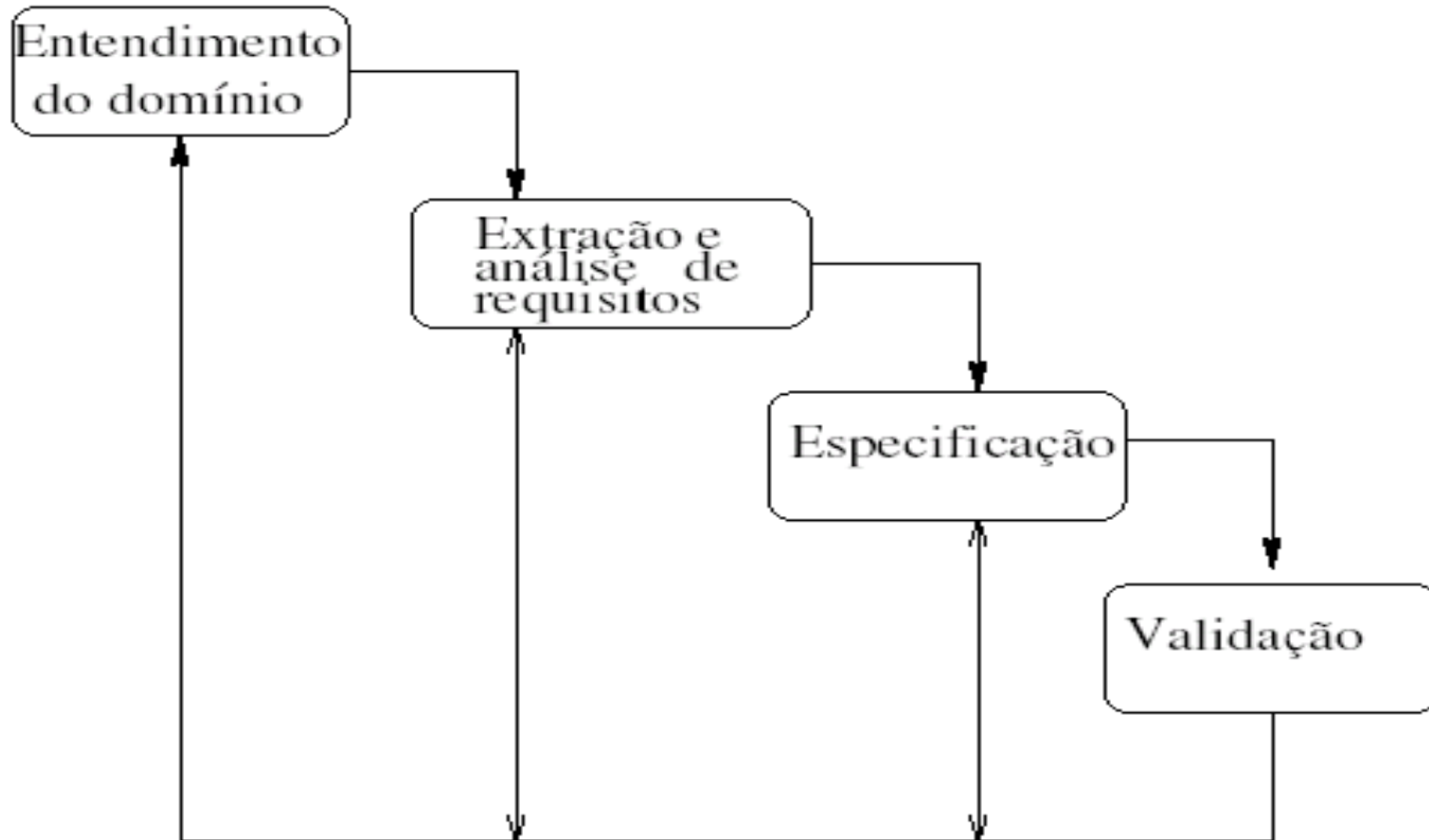
DOCUMENTO DE REQUISITOS

- O documento de requisitos é a declaração oficial do que é exigido aos desenvolvedores do sistema.
- Deveria incluir uma definição dos requisitos do usuário e a especificação dos requisitos do sistema.
- NÃO é um documento de projeto.
- Deve incluir O QUE o sistema deve fazer e não COMO deve fazer.
- Indica o PROBLEMA a ser resolvido e não a SOLUÇÃO

USUÁRIOS DE UM DOCUMENTO DE REQUISITOS



PASSOS PARA O PROCESSO DE EXTRAÇÃO DE REQUISITOS DE SOFTWARE



PASSOS PARA O PROCESSO DE EXTRAÇÃO DE REQUISITOS DE SOFTWARE

- **Entendimento do domínio:** os desenvolvedores devem entender o domínio da aplicação (documentos, livros, pessoas).
- **Extração e análise de requisitos:** acontece a descoberta, revelação e entendimento dos requisitos, através de interação com o(s) usuário(s); e desenvolvedores envolvendo:
 - Classificação e organização dos requisitos;
 - Determinação de suas prioridades;
 - Resolução de inconsistência;
 - Conflitos e a descoberta de omissões.

PASSOS PARA O PROCESSO DE EXTRAÇÃO DE REQUISITOS DE SOFTWARE

- **Especificação dos requisitos:** armazenamento dos requisitos em uma ou mais formas, incluindo linguagem formal ou semiformal, representações simbólicas ou gráficas.
- **Validação dos requisitos:** verificação dos requisitos, visando determinar se estão completos com as necessidades do usuário.

PASSOS PARA O PROCESSO DE EXTRAÇÃO DE REQUISITOS DE SOFTWARE

- Mudanças nos requisitos acontecem na maioria dos sistemas complexos (mudanças das necessidades dos usuários, outras advêm da interpretação incorreta dos requisitos do produto a ser desenvolvido.
- Requisitos incompletos, incorretos ou mal entendidos são as causas mais frequentes da baixa qualidade, ultrapassagem dos custos previstos e atraso na entrega do produto de software.

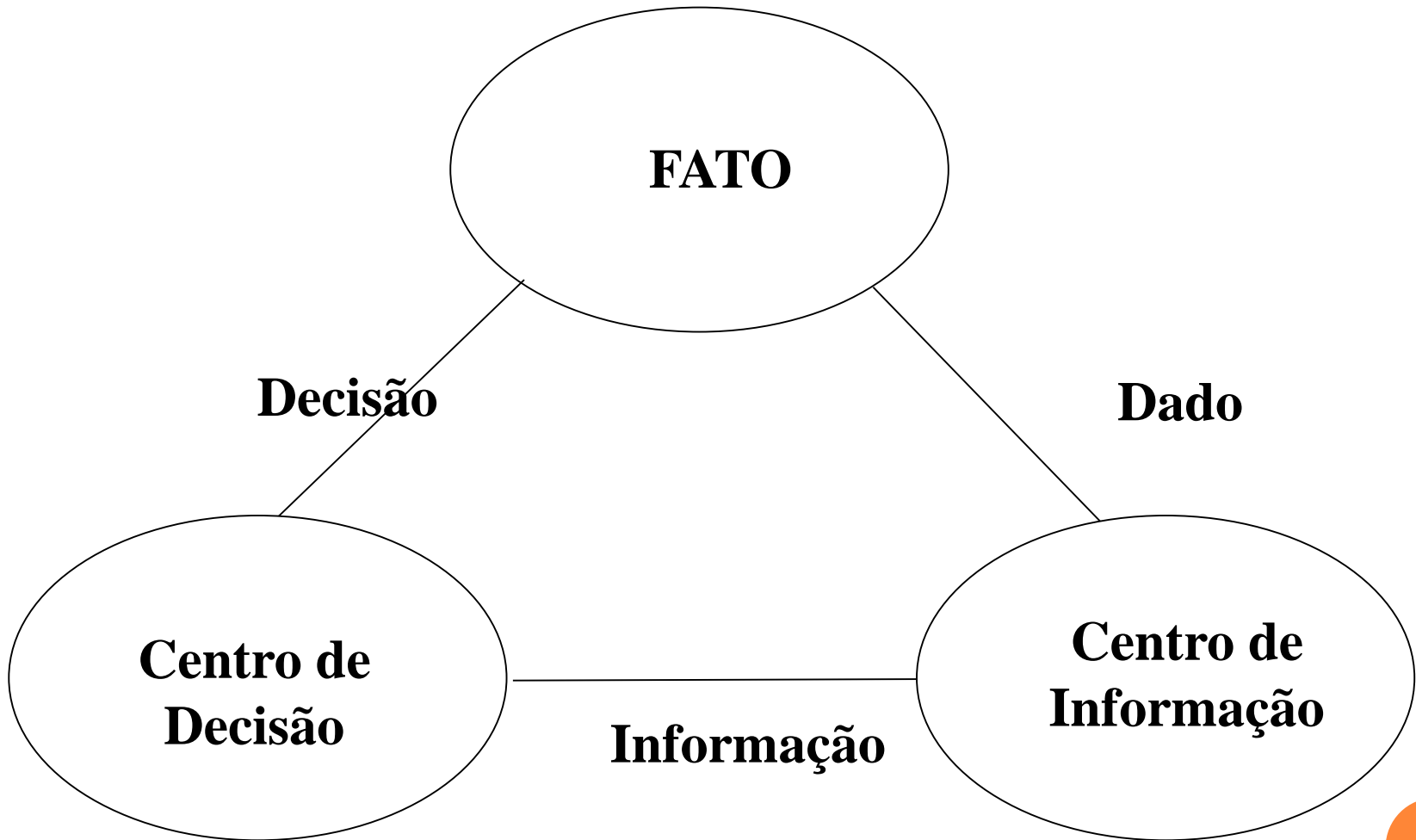
DIFICULDADES NO PROCESSO DE EXTRAÇÃO DE REQUISITOS

- Falta de conhecimento do usuário
 - O usuário não conhece sua real necessidade.
 - Desenvolvedores não conhecem o domínio do problema.
 - Diferenças entre o que os usuários querem e o que precisam.

DIFICULDADES NO PROCESSO DE EXTRAÇÃO DE REQUISITOS

- **Problemas de comportamento**
 - Conflitos e ambiguidades nos papéis → clima de insatisfação e participação menos afetiva.
- **Resultado:** custo maior, atraso no planejamento e projetos cancelados.
- **Problemas técnicos**
 - Avanço tecnológico muito rápido.

TÉCNICAS DE EXTRAÇÃO DE REQUISITOS



PROCEDIMENTO DE EXTRAÇÃO DE REQUISITOS

- **Perguntar:** identificar a pessoa apropriada.
- **Observar e inferir:** observar o comportamento dos usuários e inferir suas necessidades.
- **Discutir e formular:** discutir com os usuários suas necessidades e, juntamente com eles, formular um entendimento comum dos requisitos.

PROCEDIMENTO DE EXTRAÇÃO DE REQUISITOS

- **Negociar:** a partir de um conjunto-padrão de requisitos, negociar com os usuários quais dessas características serão incluídas, excluídas ou modificadas.
- **Identificar problemas:** investigar os problemas para identificar os requisitos que podem melhorar o produto.
- **Supor:** quando não existe usuário, ou para a criação de um produto inexistente é preciso usar intuição.

TÉCNICAS DE EXTRAÇÃO DE REQUISITOS

- **Técnicas informais** – baseada em comunicação estruturada e interação com o usuário.
 - Entrevistas
 - Questionário
 - Técnica dos 5 W's
 - Brainstorming
- **Observação**
- **Técnicas formas** – construção de um modelo conceitual do problema sendo analisado, ou de um protótipo de um produto de software a ser construído.

TÉCNICAS DE EXTRAÇÃO DE REQUISITOS

- Metodologias mais utilizadas
 - Entrevistas
 - São agendadas entrevistas com o cliente onde ele explica como funciona o processo (que será informatizado) de funcionamento da empresa.
 - Observação
 - O analista de requisitos visita a empresa durante alguns dias e observa como as coisas funcionam relatando tudo em um documento que será discutido posteriormente (Hospital Jorge Valente – Salvador-BA).

ATIVIDADE PROGRAMADA

- Crie um documento de requisitos contendo seguindo o modelo.
- Como tema do seu trabalho você pode usar o mesmo tema do trabalho de análise de risco.
- Deverá ser entregue em arquivo tipo .pdf
- Deverá ser entregue o arquivo com o seguinte formato:
AtividadeProgramadaAnaliseRequisitos_Aluno1_
Aluno2_Aluno3.pdf

ATIVIDADE PROGRAMADA

- Será enviado por e-mail um modelo de documento de requisitos que deverá ser seguido, caso não seja o valor da nota será zero.
- O documento deverá ter, no máximo, 7 requisitos para cada sessão (menos que isso será feita a proporção da nota de acordo aos requisitos entregues)
- A data de Entrega será dia 28/10 até as 23:59:59