



UNIVERSIDADE
FEDERAL DO CEARÁ
CAMPUS DE RUSSAS

Arquitetura de Software

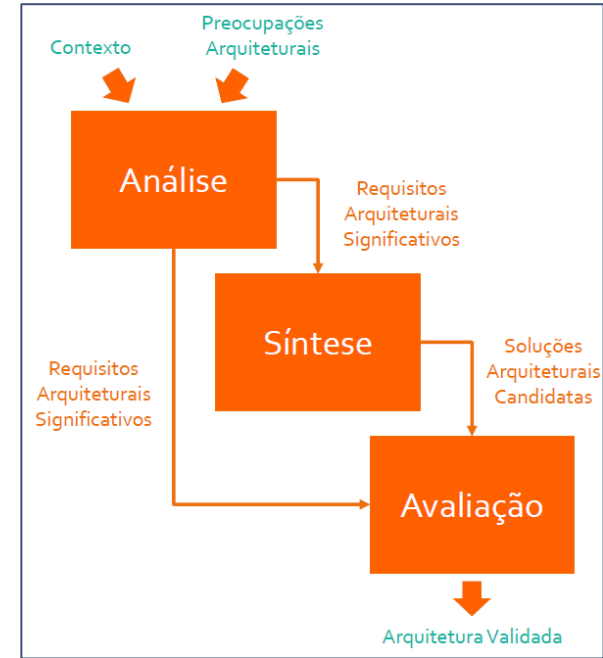
Arquitetura e Requisitos

Profa. Dra. Anna Beatriz Marques

Contextualização

O projeto da arquitetura é o elo **crítico** entre o projeto e a engenharia de requisitos.

- O processo de design arquitetural considera as etapas de **Análise**, **Síntese** e **Avaliação**.
- A etapa de Análise arquitetural articula os **requisitos significativos para a arquitetura** com base nos problemas que essa arquitetura deverá solucionar e o contexto



1

Tipos de Requisitos

Relação com Arquitetura de
Software



Tipos de Requisitos de Software

- **Requisito funcional:** define o que o sistema deve fazer e como o sistema deve reagir a estímulos em tempo de execução.
- **Requisito não-funcional:** define critérios de qualidade aplicados aos requisitos funcionais ou ao sistema como um todo.
- **Restrições:** define decisões de design que não podem ser alteradas, ou seja, decisões previamente tomadas.



Como a Arquitetura de Software atende aos tipos de requisitos?

Requisito funcional

- Alocação de responsabilidades aos elementos arquiteturais

Requisito não-funcional

- Definição de estruturas de elementos arquiteturais
- Definição dos comportamentos e interações entre os elementos arquiteturais

Restrições

- Assumir as decisões de design tomadas
- Alinhar tais decisões às decisões de design arquiteturais

2

Requisitos Arquiteturais

Conceitos

O que são Requisitos Arquiteturais?

São as **condições ou capacidades** que um sistema deve possuir **em termos de arquitetura** para atender às demandas funcionais exigidas na **especificação do sistema**





Requisitos Arquiteturais - Definições

- Requisitos de **potencial valor para a arquitetura** de um sistema
- São **informações relevantes** em termos de arquitetura para a parte funcional de um sistema
- Um requisito arquitetural **pode ser explícito ou implícito**
- Na maioria das vezes **costumam ser implícitos**
 - Devem ser explorados com os stakeholders



Exemplos de Requisitos Arquiteturais

- O sistema será acessado por usuários de outros países
- Uma ajuda online será necessária
- A persistência será manuseada por um banco de dados não-relacional
- O sistema deve estar disponível 24 horas nos 7 dias da semana
- O sistema deve proteger os dados dos usuários



Exemplos de Requisitos Arquiteturais

O sistema será acessado por usuários de outros países

- Uma ajuda online será necessária

Requisito
arquitetural funcional

- A persistência será manuseada por um banco de dados não-relacional
- O sistema deve estar disponível 24 horas nos 7 dias da semana
- O sistema deve proteger os dados dos usuários



Exemplos de Requisitos Arquiteturais

- O sistema será acessado por usuários de outros países
- Uma ajuda online será necessária
- A persistência será manuseada por um banco de dados não-relacional
- O sistema deve estar disponível 24 horas nos 7 dias da semana
- O sistema deve proteger os dados dos usuários

Requisito
arquitetural funcional

Restrição
arquitetural



Exemplos de Requisitos Arquiteturais

- O sistema será acessado por usuários de outros países
- Uma ajuda online será necessária Requisito arquitetural funcional
- A persistência será manuseada por um banco de dados não-relacional Restrição arquitetural
- O sistema deve estar disponível 24 horas nos 7 dias da semana
- O sistema deve proteger os dados dos usuários Requisito arquitetural não-funcional

“ *A inclusão de requisitos arquiteturais, muito provavelmente, resultará em uma arquitetura diferente daquela em que esses requisitos não foram incluídos.*



Seleção de Requisitos Arquiteturais



3

Elicitação de Requisitos Arquiteturais

Alguns métodos de elicitação existentes



Desafios na Elicitação de Requisitos Arquiteturais

- Documento de requisitos geralmente é negligenciado
- Requisitos arquiteturais assumem muitas vezes, mas nem sempre, a forma de requisitos não-funcionais
- Pouca informação do documento de requisitos é útil para o arquiteto de software
- É necessário explorar os requisitos arquiteturais durante o processo de Engenharia de Requisitos

A person with short brown hair, seen from the back, is looking at a wall covered in various architectural design documents, sketches, and photographs. The documents include floor plans, diagrams, and text. The person is wearing a grey and black striped sweater. The wall is a light-colored surface, possibly a corkboard or a wall with pins. The overall scene suggests a creative or professional workspace focused on architecture or design.

**Quais técnicas podem ser adotadas
para identificar requisitos
arquiteturais?**



Como identificar requisitos arquiteturais?

- Análise do documento de requisitos
- Workshop de atributos de qualidade
- Entendimento dos objetivos de negócio
- Criação de árvore de utilidade
- Técnica FURPS+



Análise de Documento de Requisitos

- Importante **fonte inicial** de requisitos arquiteturais
- Porém:
 - A maioria do conteúdo de um documento de requisitos não afeta a arquitetura
 - As informações úteis para um arquiteto de software não estão no documento de requisitos





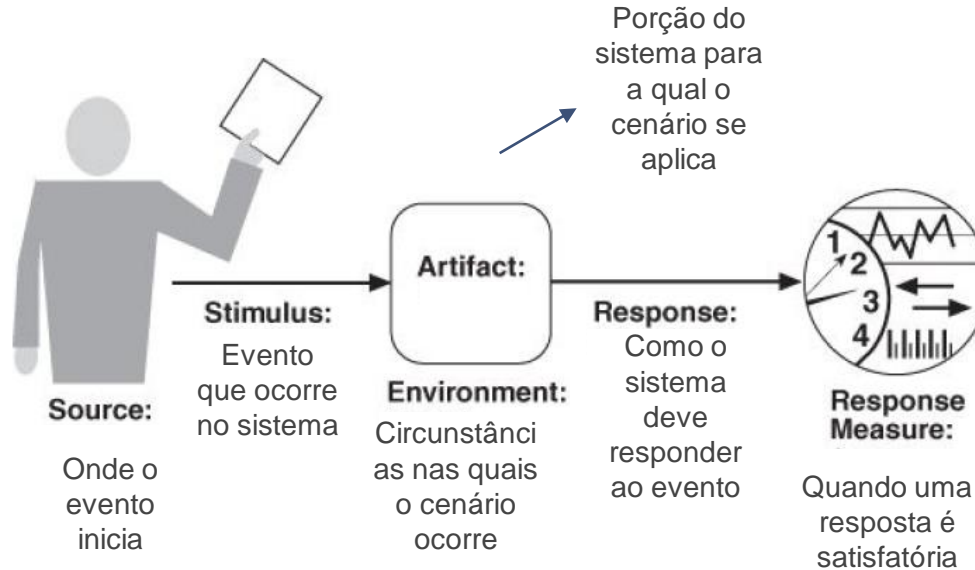
Workshop de Atributos de Qualidade (QAW)

- Método facilitador para engajar os stakeholders a explorar atributos de qualidade antes da arquitetura do software ser projetada.
- Os stakeholders fornecem suas necessidades e expectativas em relação aos atributos de qualidade.
- Devem participar entre 5 a 30 stakeholders.
- O método adota um conjunto de cenários de atributos de qualidade.



Workshop de Atributos de Qualidade (QAW)

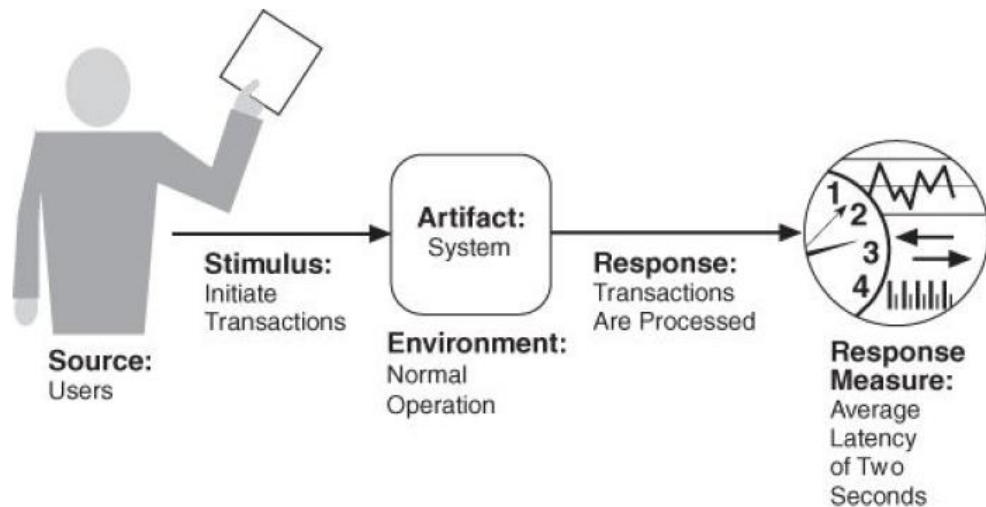
Exemplo de cenário genérico





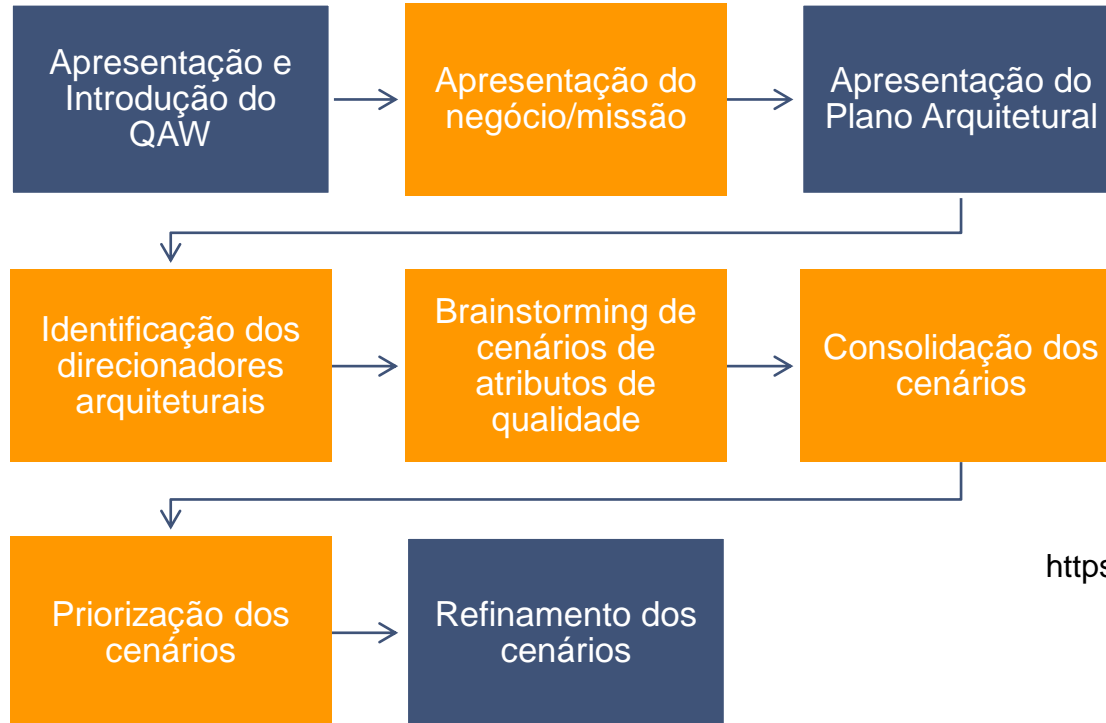
Workshop de Atributos de Qualidade (QAW)

Exemplo de cenário para o atributo Desempenho





Workshop de Atributos de Qualidade (QAW)

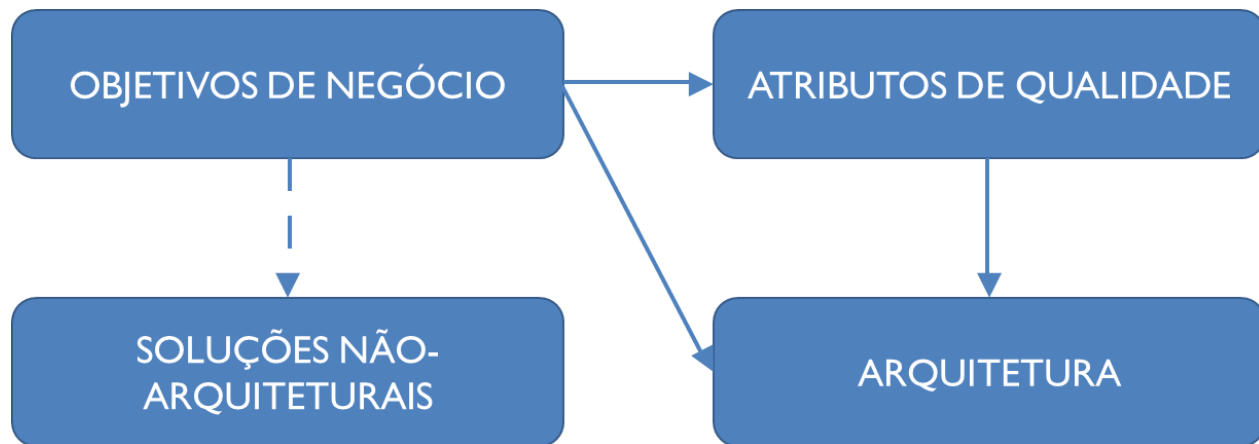


<https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=474329>



Entendimento de objetivos de negócio (PALM - Pedigreed Attribute eLicitacion Method)

- Método para identificação de objetivos de negócio que podem derivar requisitos arquiteturais
- Adota um template para especificação de objetivos de negócio





PALM (Pedigreed Attribute eLicitacion Method)

Como descrever objetivos de negócio que impactam na arquitetura?

Template:

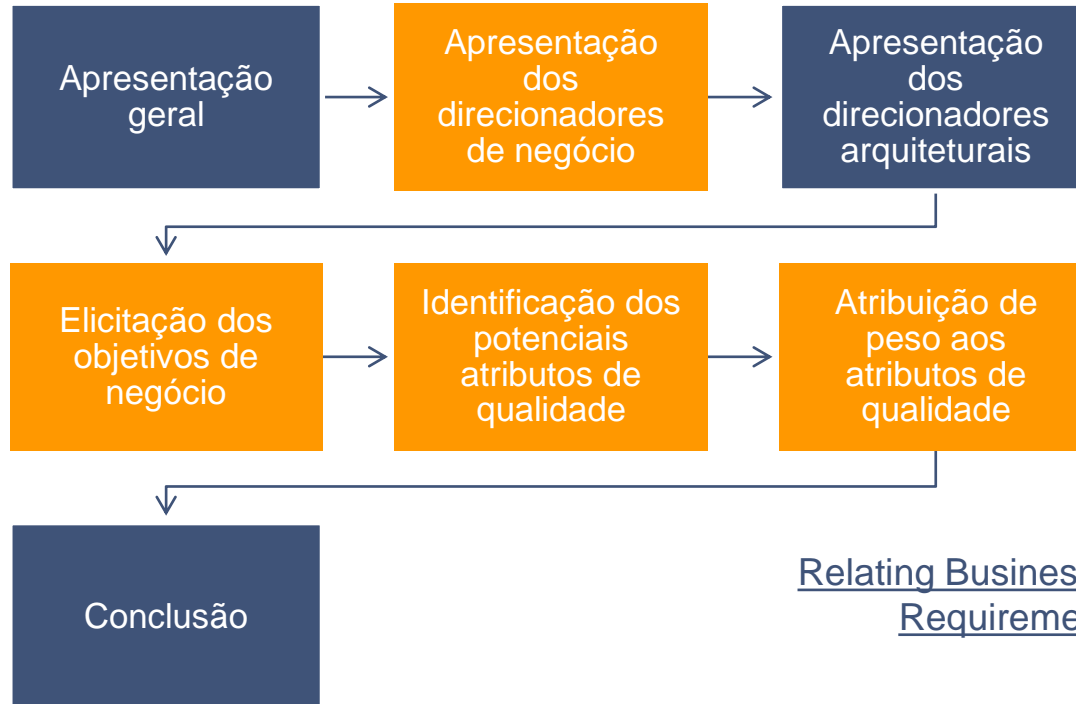
Para <**o sistema em desenvolvimento**>, o <**stakeholder**> deseja que <**objeto**> atenda <**objetivo**> no contexto de <**ambiente**> e será satisfeito se <**medida**>

Exemplo de objetivo de negócio:

Para <**a Cabine Pets**>, o <**usuário**> deseja que <**o sistema**> atenda <**à solicitação de visualização de pets para adoção em uma cidade informada**> no contexto de <**quando o sistema estiver em carga de pico**> e será satisfeito se <**a solicitação for atendida em 3 segundos**>



PALM (Pedigreed Attribute eLicitacion Method)



Relating Business Goals to Architecturally Significant Requirements for Software Systems (cmu.edu)

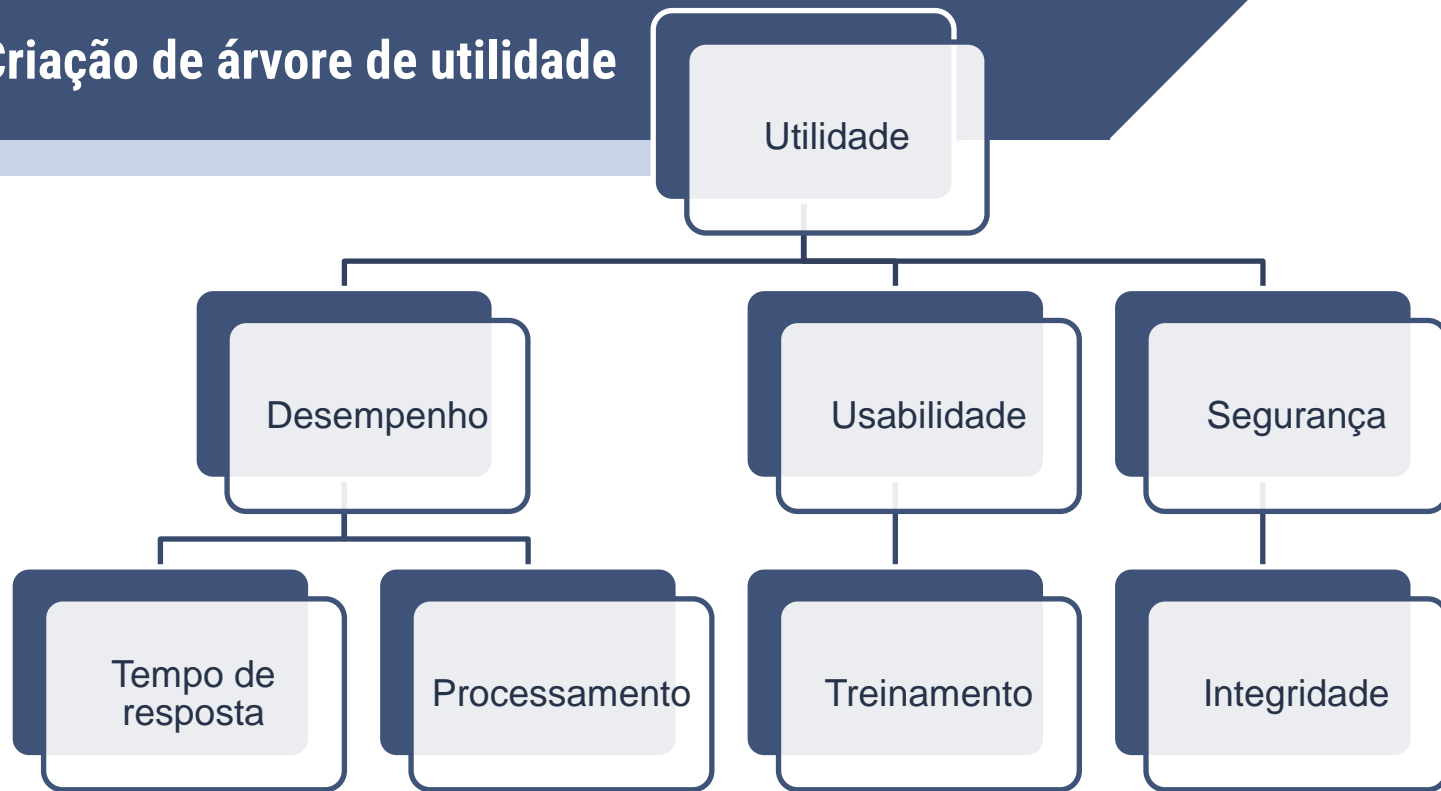


Criação de árvore de utilidade

- Criação de uma árvore para os atributos de qualidade prioritários do sistema
- Refinamento dos atributos de qualidade
- Atribuição de valores a cada atributo em termos de valor de negócio e impacto na arquitetura



Criação de árvore de utilidade



O tempo de resposta a uma alteração de senha é menor que 0,75 segundos (A,M)

Um novo usuário com experiência de no mínimo 2 anos no negócio torna-se proficiente no sistema em menos de 1 semana (M, B)



Técnica FURPS+

■ Sistema de classificação de requisitos arquiteturais

FURPS

Functionality (Funcionalidade)

Usability (Usabilidade)

Reliability (Confiabilidade)

Performance (Desempenho)

Supportability (Suportabilidade)

‘+’

Requisitos de Design

Requisitos de
Implementação

Requisitos de Interface

Requisitos Físicos



Categorias FURPS+

Functionality (Fucionalidade)

A categoria de funcionalidades representa os principais requisitos funcionais de um produto. Entretanto nem todos os requisitos tem alguma importância arquitetural. Ex. Prover acesso à aplicação via smartphone é mais significativo do que controlar estoque.

Exemplos de Requisitos arquiteturalmente significativos

Função	Descrição
Internacionalização	Prover facilidades para suportar múltiplas línguas.
Segurança	Prover serviços para proteger informações e acesso.
Impressão	Prover facilidades para impressão.
Posso ajudar Online	Prover capacidades de ajuda online.
Cross browsing	Prover capacidade de acesso por múltiplos browsers.
E-mail	Prover serviços de envio e recebimento de e-mail.
Workflow	Prover suporte à trâmite de documentos e ciclos de aprovação.



Categorias FURPS+

Usability (Usabilidade)

A usabilidade é interessada em aspectos estéticos e interatividade na interface com o usuário. Acredite, essa é uma categoria que pode determinar o sucesso ou o fracasso de um projeto. Muitas vezes por questões de prazo e verba é colocada em segundo plano.

Reliability (Confiabilidade)

Essa categoria é interessada em aspectos voltados para disponibilidade, precisão do sistema em realizar suas funções e a habilidade para se recuperar de falhas

Performance (Execução)

Essa categoria é interessada no rendimento da aplicação em termos de tempo de resposta, tempo de recuperação de falhas, tempo de inicialização e tempo de encerramento.

As categorias URPS, descrevem requisitos que são em sua maioria arquiteturalmente significativos.



Categorias FURPS+

Supportability (Suportabilidade)

- Essa categoria é interessada nos aspectos de testabilidade do sistema, adaptabilidade, manutenabilidade, compatibilidade, configurabilidade, instabilidade, e localizabilidade. Esses termos são aportuguesados. Talvez, não encontrados nos dicionários

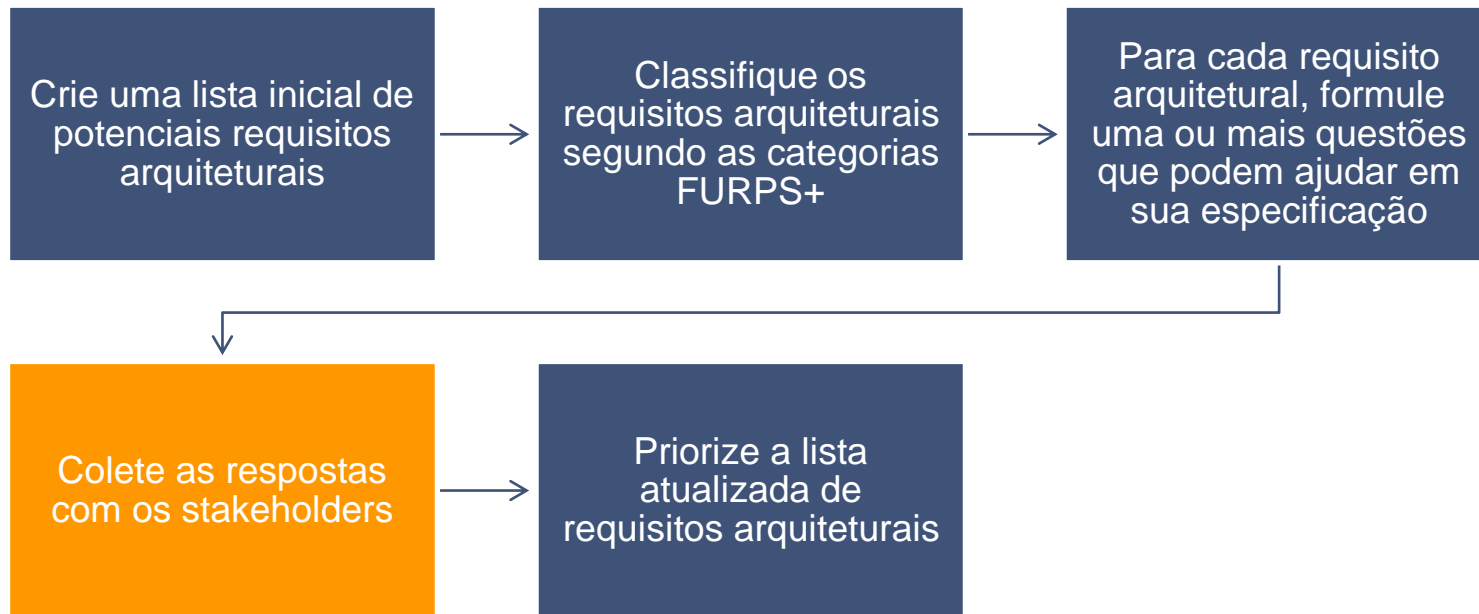
“+”

- O acrônimo “+” é reservado a identificação de novas categorias que normalmente possuem significado arquitetural

- Requisitos de projeto - Categoria reservada a restrições de projeto
- Requisitos de implementação - Categoria reservada a restrições de código ou construção de sistemas
- Requisitos de interface - Categoria reservada a interações com elementos externos ao sistema.
- Requisitos físicos - Categoria reservada as restrições impostas por hardware, de qualquer ordem.



Técnica FURPS+



Referências

- Bass, L., Clements, P., & Kazman, R. (2003). *Software architecture in practice*. Addison-Wesley Professional.
- Sommerville, I. (2011). *Engenharia de Software-9ª Edição*. São Paulo: Pearson Prentice Hall.
- Barbacci, Mario., Ellison, Robert., Lattanze, Anthony., Stafford, Judith., Weinstock, Charles., & Wood, William. (2003). Quality Attribute Workshops (QAWs), Third Edition (CMU/SEI-2003-TR-016). Retrieved January 11, 2021, from the Software Engineering Institute, Carnegie Mellon University website: <http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=6687>
- Clements, Paul., & Bass, Len. (2010). Relating Business Goals to Architecturally Significant Requirements for Software Systems (CMU/SEI-2010-TN-018). Retrieved January 11, 2021, from the Software Engineering Institute, Carnegie Mellon University website: <http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=9347>
- Eeles, P. (2005). Capturing architectural requirements. IBM Rational developer works. Disponível em: https://www.researchgate.net/profile/Peter_Eeles2/publication/329760910_Capturing_Architectural_Requirements/links/5c197761458515a4c7e8bae1/Capturing-Architectural-Requirements.pdf



Obrigada