

**USP - UNIVERSIDADE DE SÃO PAULO**  
**EACH – ESCOLA DE ARTES, CIÊNCIAS E HUMANIDADES**  
**BSI – BACHARELADO EM SISTEMAS DE INFORMAÇÃO**

**ACH2028 – Qualidade de Software – 02/2022**

***Prof. Marcelo Medeiros Eler***

**Atividade 01 – Inspeção de Requisitos**

O objetivo desta atividade é exercitar a prática de inspeção de software, mais especificamente a inspeção de requisitos. Embora muitos processos não gerem documentos de requisitos formalizados, a prática de investigar defeitos típicos de um documento de requisitos é útil para detectar problemas em qualquer tipo de requisito e fornece aos inspetores conhecimentos necessários para escrever requisitos de maior qualidade, qualquer que seja o formato.

Este trabalho pode ser feito em grupos de três pessoas utilizando técnicas de leitura para inspecionar o documento de requisitos do SISTEMA DE APOIO À ESCRITA (SAPES). Cada integrante do grupo pode escolher uma técnica diferente ou podem usar a mesma técnica – ou leitura baseada em perspectiva ou checklist (ver guia na próxima página). Se o grupo escolher a técnica baseada em perspectiva, deve-se indicar no relatório do trabalho qual aluno desempenhou qual papel. Cada integrante do grupo deve fazer a análise separadamente e depois deve-se fazer uma compilação dos defeitos encontrados para gerar o relatório final. O resultado desta inspeção deve ser uma lista dos defeitos encontrados, indicando a classe de defeito (Omissão, Fato Incorreto, Inconsistência, Ambiguidade, Informação Estranha, Diversos) e uma indicação de quais avaliadores encontraram o defeito (não precisam identificar quem é o avaliador 1, 2, 3, etc).

Exemplo da lista de defeitos:

Número	Requisito	Tipo de Defeito	Descrição	Avaliadores (que detectaram o defeito)
1	RF1	Omissão	...	1,2
2	RF24	Fato Incorreto	...	1,3
3	RF13	Omissão	...	3
...	...	...	...	2,3

## **SOFTWARE QUALITY CONSULTING**

**<http://www.swqual.com>**

### **Requirements Inspection Checklist**

- 1 – Os requisitos fazem uma separação clara entre funções e dados?
- 2 – Os requisitos definem todas as informações que serão exibidas para o usuário?
- 3 – Os requisitos definem como o sistema e os usuários devem reagir a situações de erro?
- 4 – Cada requisito está escrito de forma clara, concisa e não ambígua?
- 5 – Cada requisito é testável?
- 6 – Existem requisitos ambíguos ou implícitos?
- 7 – Existem requisitos conflitantes?
- 8 – Existem áreas que deveriam ser tratadas pelos requisitos e que atualmente não estão descritas?
- 9 – Existem requisitos de desempenho (tempo de resposta, requisitos de armazenamento, memória) definidos?
- 10 – Se os requisitos referem-se a cadeias de decisões complexas, eles estão expressos de uma forma que facilita sua compreensão (como tabelas e árvores de decisão, por exemplo)?
- 11 – Requisitos de atualização do software foram especificados?
- 12 – Existem requisitos com detalhamento excessivo e desnecessário?
- 13 – As restrições de tempo real foram especificadas com detalhes suficientes?
- 14 – A precisão e acurácia dos cálculos foram especificadas?
- 15 – É possível desenvolver um conjunto de teste detalhado com base nas informações fornecidas pelos requisitos? Se não, quais informações estão faltando?
- 16 – As suposições (ou premissas) e as dependências foram claramente especificadas?
- 17 – Os requisitos apresentam todas as informações indicadas no sumário (se houver)?

Possíveis tipos de defeitos: omissão, fato incorreto, inconsistência, ambiguidade, informação estranha e diversos/outros.