

DISCIPLINA:

QUALIDADE E GOVERNANÇA EM PROJETOS DE SOFTWARE

AULA:

16 – CONTROLE DE VERSÃO EM PROJETOS DE SOFTWARE APLICADO

PROFESSOR:

RENATO JARDIM PARDUCCI

PROFRENATO.PARDUCCI@FIAP.COM.BR

GIT HUB PARA ADMINISTRAR FONTES DE SOFTWARE



<https://youtu.be/Z76STqABbqs>

PONTO CRÍTICO DA GESTÃO DE MUDANÇA DE SOFTWARE

Um grande problema existente na atualização de software é o de se alterar um arquivo fonte sem rever outros arquivos fonte associados (que referenciam o conteúdo do primeiro arquivo alterado).

PONTO CRÍTICO DA GESTÃO DE MUDANÇA DE SOFTWARE

Imagine a situação a seguir:

- Um **usuário revê o seu requerimento** funcional e...
- Um **analista elimina um Caso de Uso** do modelo de Casos de Uso de um sistema em desenvolvimento, porém...
- **Classes foram definidas para tratar esse Caso de Uso** que foi removido e agora, elas perderam o sentido, e...
- **Tabelas em banco de dados foram criadas para persistir os dados das Classes** de Entidade e Códigos em JAVA foram construídos para tratar os Métodos das Classes – tudo isso perdeu o sentido com o fim do Caso de Uso.

Vamos ter problemas com desperdício de trabalho e erro na manutenção futura do software se isso não for harmonizado!

PONTO CRÍTICO DA GESTÃO DE MUDANÇA DE SOFTWARE

Imagine essa outra situação:

- Um **programador alterou parâmetros de entrada da sua aplicação** (componente de software que implementa o método de uma Classe de Objeto) só que...
- **Essa aplicação é chamada por 13 outras**, sendo 3 referentes a Métodos da própria Classe e 10 referentes a Métodos de outras Classes de Objetos do sistema.

Vai dar problema!

PONTO CRÍTICO DA GESTÃO DE MUDANÇA DE SOFTWARE

Seja em modificações ou exclusões de dados de modelagem ou código de aplicação, é preciso verificar sempre quais os outros componentes de software ou de documentação externa ao software que precisam ser revisados!

Esse gerenciamento se chama **Garantia da Integridade Referencial dos Componentes de Software.**

INTEGRIDADE REFERENCIAL DE REQUISITOS

A integridade referencial pode ser garantida por matrizes de relacionamento entre componentes do software (fontes de código de aplicação e documentação).

Esse controle de integridade deve nascer junto com a lista de requisitos (backlog de produto).

INTEGRIDADE REFERENCIAL DE REQUISITOS

Exemplo de Matriz de Integridade Referencial de Requisitos:

Identificação do Requisito	Identificação do requisito relacionado
Calcular o imposto de renda da pessoa física	Informar o ganho por fonte pagadora
Calcular o imposto de renda da pessoa física	Informar bens e direitos
Calcular o imposto de renda da pessoa física	Informar movimento financeiro tributável
Calcular o imposto de renda da pessoa física	Exportar dados para a receita federal

Requisitos relacionados com o
Cálculo do Imposto de Renda

INTEGRIDADE REFERENCIAL DE REQUISITOS

Outra forma de representar a Matriz de Integridade Referencial de Requisitos:

	Requisitos associados (afetados em caso de modificação dos requisitos na coluna vertical à esquerda)									
Requisito	RF1	RF2	RF3	RF4	RF5	RN1	RN2	RN3	RN4	RN5
RF1										
RF2										
RF3										
RF4										
RF5										
RN1										
RN2										
RN3										
RN4										
RN5										

*Marcar comum "X" as associações válidas que implicam em revisões em caso de definição e redefinição

INTEGRIDADE REFERENCIAL DE REQUISITOS

Exemplo de Matriz de associação entre códigos fonte de programas de aplicação:

Identificação do Fonte	Identificação do fonte relacionado
Pessoa.Calculo_GanhoSalarial()	Pessoa.Calculo_IR()
Pessoa.Calculo_GanhoSalarial()	Pessoa. Consulta_Salario()
Pessoa.Calculo_GanhoSalarial()	ReceitaFederal.Export_Ganho()

Programas que usam o Calculo de Ganho Salarial da Classe Pessoa

INTEGRIDADE REFERENCIAL DE REQUISITOS

O mesmo tipo de matriz pode ser usado para associar documentos da engenharia do software ou de gerenciamento de projetos:

Identificação do Documento	Identificação do doc. relacionado
Plano_Projeto_SI_Comercial_V9.0.MPP	Pacote_SI_Comercial_V1.0.ZIP
Plano_Projeto_SI_Comercial_V9.0.MPP	Apresentacao_SI_Comercial.PPT
Plano_Projeto_SI_Comercial_V9.0.MPP	História_Usuário_SI_Comercial. DOCX
Plano_Projeto_SI_Comercial_V9.0.MPP	Modelo_SI_Comercial. ASTAH

Documentos que implementam as ações do Plano do Projeto do SI

INTEGRIDADE REFERENCIAL DE REQUISITOS

Podem existir matrizes de **relacionamento dos Requisitos listados para o Produto de software e os componentes de documentação, código fonte e estruturas de dados** associados:

Identificação do Requisito	Identificação do componente relacionado
Calcular o imposto de renda da pessoa física	Diag. Caso de Uso – Calculo IRPF Tabela IRPF Dicionário de dados IRPF

Componentes associados ao requisito de calcula o IR de pessoa física

INTEGRIDADE REFERENCIAL DE REQUISITOS

O controle de Integridade Referencial permite RASTREABILIDADE de requisitos de software:

Rastreabilidade: permite a fácil determinação dos antecedentes e consequências de todos os requisitos.

Para trás: deve ser possível localizar a origem de cada requisito. Deve-se sempre saber **por que existe cada requisito, e quem ou o que o originou**. Isso é importante para que se possa avaliar o impacto da mudança naquele requisito e dirimir dúvidas na interpretação.

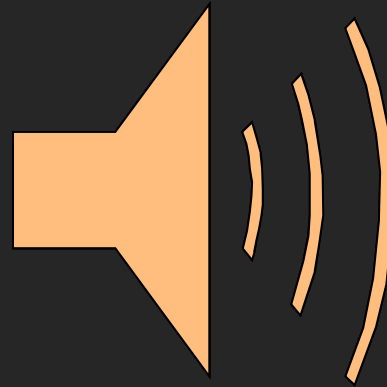
Para frente: deve ser possível localizar **quais os resultados do desenvolvimento que serão afetados** por cada requisito. Isso é importante para garantir que **os itens de análise, desenho, código e testes cubram todos os requisitos** e nada mais do que eles, e para localizar os itens que serão afetados por uma mudança nos requisitos.

INTEGRIDADE REFERENCIAL DE REQUISITOS

Geralmente é impraticável gerenciar a integridade referencial de componentes de software sem o auxílio de um software.

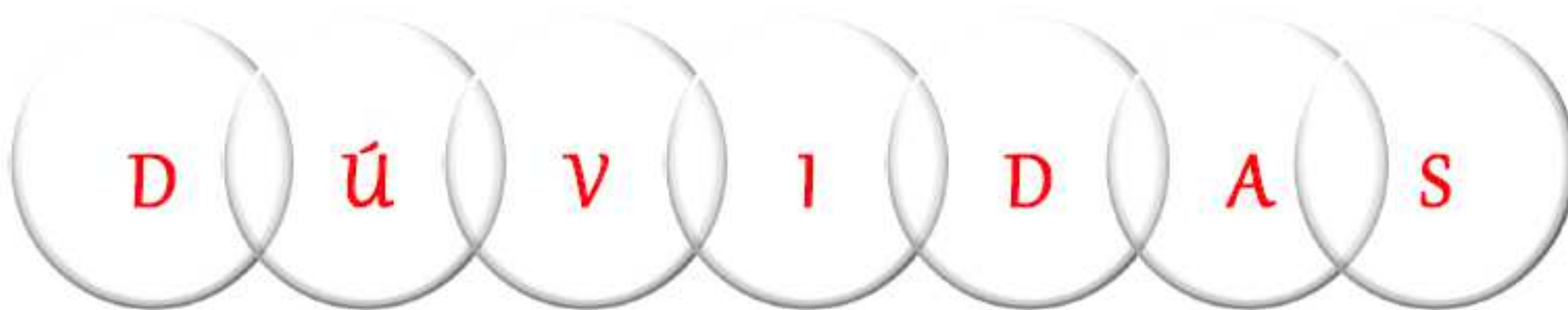
Muitos softwares de apoio ao processo completo da Engenharia (CASE) já possuem o controle referencial integrado.

Nesses softwares, quando um componente é modificado, uma lista de verificação dos documentos associados é disponibilizada com um status de verificação pendente e o desenvolvedor deve confirmar que todas as revisões foram feitas.



**ESCUTE O PODCAST NO CANAL DO PROFESSOR
GERENCIAMENTO DE INTEGRIDADE REFERENCIAL DE REQUISITOS**

<https://youtu.be/DqVDICy05DI>



EXERCÍCIOS extras



Resolvidos

1. EXERCÍCIO COM RESOLUÇÃO EM SALA DE AULA

Crie uma Matriz de Integridade Referencial que sirva para associar qualquer tipo de fonte com quaisquer outros tipos de fonte.

Ela deve permitir cruzar documentos texto com diagramas, diagramas com códigos fonte, um diagrama com outro, um componente de diagrama com outro, um código fonte com outro, uma estrutura de dados com outra.

2. EXERCÍCIO COM RESOLUÇÃO EM SALA DE AULA

Você está desenvolvendo um projeto que utilizará sensores de luz e temperatura, os quais serão colocados em um ambiente qualquer para capturar sinais desse local. Os sensores estarão conectados a um dispositivo Arduíno cada um e esses dispositivos transmitirão dados para um computador ao qual serão conectados por uma rede de dados qualquer.

Os dados exportados por esses dispositivos a cada minuto, serão guardados em arquivos que ficam à disposição para consumo de informações em um sítio na Internet.

Um software de retaguarda irá acessar os arquivos na nuvem e fazer uma associação de temperatura e luminosidade do ambiente.

Se a luminosidade superar 10.000 lumens e o calor exceder 60 graus Célsius, o software emitirá uma mensagem de alerta de incêndio.

1º) Organizem-se em duplas;

2º) Criem as duas tabelas de registro de dados a seguir no desktop de um membro da equipe:

- Tabela – Registro Arduino 1: contém a tupla [ID-Local-Monitorado, Data, Hora, Minuto, Temperatura em Graus Kelvin]
- Tabela – Registro Arduino 2: contém a tupla [ID-Local-Monitorado, Data, Hora, Minuto, Luminosidade em Lumens]

Essas são as tabelas que ficariam na nuvem!

3º) Criem o programa que lê as tabelas e faz a análise de incêndio iminente, gerando o alerta.

4º) Coloquem os fontes (ZIP) no GitHub.

5º) Um membro da equipe altera o código usando os recursos do GitHub (criem um Branch)

6º) Os três membros da equipe tentam alterar ao mesmo tempo o código fonte.

7º) Façam as atualizações na cópia Master.

CONTROLE DE VERSÃO EM PROJETOS DE SOFTWARE

FIM

PROFESSOR:
RENATO JARDIM PARDUCCI

PROFRENATO.PARDUCCI@FIAP.COM.BR