**Youtube:**

**Fronteiras da Engenharia de Software - Podcast**

Host: Adolfo Neto (UTFPR)

Co-host: Gustavo Pinto (UFPA)

Equipe: Marco Tulio Valente (UFMG), Danilo Ribeiro (Zup), Leonardo Fernandes (IFAL), Fabio Petrillo (Univ. Quebec), Marcela dos Santos (Univ. Quebec)

**Introdução ao Episódio**

[ADOLFO] Olá ouvintes, eu sou Adolfo Neto, professor da Universidade Tecnológica Federal do Paraná.

[GUSTAVO] Oi pessoal, tudo bem? Eu sou Gustavo Pinto, professor da UFPA, e este é mais um episódio do Fronteiras da Engenharia de Software, um podcast feito para refletir sobre o presente e futuro da engenharia de software

[ADOLFO] Fronteiras da Engenharia de Software é um podcast de divulgação científica que discute mensalmente diversificados temas da Engenharia de Software. Os episódios abordam desde questões fundamentais da engenharia de software até os mais recentes avanços na área.

[GUSTAVO] Esse já é o nosso sexto episódio, e hoje contamos com a presença do Uirá Kulesza, que é professor na UFRN. Uirá lidera o grupo de pesquisa CASE, Collaborative & Automated Software Engineering (CASE Lab - http://caseufrn.github.io/)

[ADOLFO] Uirá

Nos últimos anos, Uirá tem trabalhado bastante com Integração Contínua. Vem com a gente!

>> Ver <http://caseufrn.github.io/publications/>

**Episódio 06: CI, com Uirá Kulesza (UFRN)**

**Parte 1: Apresentação [5 min, estimativa]**

[ADOLFO] Olá ouvintes, para este sexto episódio, contamos com a presença de Uirá Kulesza, que é professor na UFRN.

Tudo bem, Uirá? Você pode se apresentar para as pessoas que nos escutam?

**Parte 2 - Integração Contínua [20 min, estimativa]**

[GUSTAVO] Uirá, explica pra gente o que é integração contínua?

[GUSTAVO] Quais são os pilares da integração contínua? O que que eu preciso ter de ferramentas e práticas que podem me apoiar para ter um ambiente saudável de integração contínua? Qual é o (ou tem algum) setup ideal de um projeto pra integração contínua fazer sentido?

[ADOLFO] Quais são os principais benefícios da integração contínua? Pq eu devo praticar na minha empresa?

[ADOLFO] Esses benefícios fazem sentido quando eu trabalho num grupo pequeno? Ou trabalho sozinho?

[GUSTAVO] Quais são os desafios que integração contínua traz para os desenvolvedores? Coisas como conflitos de integração (ou merge conflicts) passaram a ser problemas comuns. Tem algum outro tipo de problema que é intrínseco de IC?

[GUSTAVO] A resolução de conflitos de integração é comumente feita de maneira manual (tenho que ir lá e escolher qual parte do código vai ficar/vai embora). Como está o status atual das ferramentas que apoiam essa atividade de resolução de conflitos de integração? Dá pra usar no dia a dia? Ou elas dão mais trabalho do que ajudam?

[ADOLFO] Uma das vantagens de sistemas de controle de versão como git é que conseguimos separar o trabalho (via branches) e assim podemos trabalhar em uma funcionalidade com mais tranquilidade, sem impactar o trabalho de ninguém (caso eu precise fazer commits de WIP). No entanto, se eu passar 15 dias trabalhando numa branch da minha funcionalidade, quando eu quiser integrar essa funcionalidade na branch principal, eu vou ter uma dor de cabeça, pq a branch principal já deve ter avançado muito enquanto eu trabalhava na minha própria branch. Na perspectiva de integração contínua, como que eu poderia resolver esse tipo de problema? Ou ainda é um problema em aberto?

[ADOLFO] Existe alguma “rule of thumb” (regra de ouro) de um limite de tempo que eu preciso para integrar com a branch principal? Ou isso varia de projeto para projeto ou organização para organização?

OPCIONAL [GUSTAVO] Qual a diferença entre integração contínua e entrega contínua? E deployment contínuo? São conceitos diferentes? Relacionados?

**Parte 3 - Pesquisas em CI [20 min, estimativa]**

[GUSTAVO] O Kent Beck lá no livro de XP de 2004 já falava sobre integrar e testar código em intervalos menores. Ferramentas de integração contínua (como travis e o circleCI) existem há pelo menos algumas décadas. O Jenkins por exemplo foi introduzido em 2011. No entanto, só nos últimos poucos anos (2--3 anos) que começaram a aparecer muitos trabalhos científicos sobre IC. Você tem alguma ideia de pq houve toda essa leva de trabalhos recentemente?

OPCIONAL [ADOLFO] Integração contínua tem algum impacto (positivo?) em alguma outra atividade de desenvolvimento de software? Por ex, será que o número de bugs diminui após adotar IC? Será que eu consigo ter entregas mais rápidas depois de adotar IC? Faz sentido?

[GUSTAVO] Vocês tiveram um trabalho recente sobre integração contínua e evolução de código de teste. Você pode explicar um pouco mais sobre o trabalho pra gente? Qual é a relação entre os dois?

OPCIONAL [ADOLFO] Outro trabalho interessante é sobre o impacto de integração contínua no tempo de vida do PR. Qual foi a ideia desse trabalho? O uso de integração contínua acelera o tempo de vida do PR?<Caso a discussão do trabalho acima tome muito tempo, podemos pular esse trabalho>

[GUSTAVO] Para quem quer implantar CI em sua equipe, quais as recomendações/dicas/sugestões você poderia dar?

[ADOLFO] Quais são os problemas científicos abertos, quando se fala de integração contínua?

**Parte 4 - Indicações**

[GUSTAVO] Alguma indicação para nossos ouvintes?

Posts do Martin Fowler

Livro Extreme Programming Explained

Acompanhar sites como InfoQ e similares

Cursos sobre configuração de pipelines e automação de testes.

**Parte 5: Conclusão [3 min, estimativa]**

[ADOLFO] Para você, qual é a próxima fronteira da engenharia de software?

**Texto para divulgação**

Sites do Uirá

* <https://www.dimap.ufrn.br/~uira/>
* <https://sigaa.ufrn.br/sigaa/public/docente/portal.jsf?siape=1644456>
* <https://scholar.google.com/citations?user=-lb3qmgAAAAJ&hl=pt-BR>
* <http://lattes.cnpq.br/0189095897739979>
* <https://twitter.com/uirakulesza>

**Extra (opcional)**:

