



# Ágil na Prática

## 1. Intro (20min)

**Why**

- TI é uma área fantástica e você faz parte dela!
- Tudo é muito dinâmico
- Precisamos conhecer o que há de novo para tirar o melhor proveito da tecnologia e realizar mais!
- Sair desta aula com um melhor entendimento de como o mercado desenvolve software hoje
- Quem sabe, sair inspirado a começar um novo projeto
- Entender como o mercado de desenvolvimento de software usa métodos ágeis na prática
- Esclarecer quais as principais práticas adotadas
- Montar um ambiente de desenvolvimento profissional

**How**

**Mapa**

**What**

**Engenharia de Software não é Engenharia Civil**

- Mas os requisitos estão SEMPRE mudando!
- Portanto, o software precisa ser construído de forma que possa mudar
- Mais do que isso, você precisa se organizar para abraçar a mudança

**Refs**

- Qual o principal problema das metodologias tradicionais?
- Ágil?
- Fazer correndo?
- Não! Responder rápido às mudanças
- Mudar é caro demais
- Principal diferença: software deve estar SEMPRE adequado aos requisitos do usuário
- http://pt.slideshare.net/webgoal/desenvolvimento-gil-na-prtica-agile-tour-2011-poos-de-caldas
- http://www.devmedia.com.br/conceitos-basicos-sobre-metodologias-ageis-para-desenvolvimento-de-software-metodologias-classicas-x-extreme-programming/10596

## 2. Práticas ágeis (30min)

Quais as práticas fundamentais para abraçar a mudança?

Quais as práticas mais importantes?

Principais frameworks/metodologias?

**Refactoring**

- Lembre-se: abraça a mudança!
- Evolução de código
- Débito técnico/Cleaning
- TDD
- Testes unitários, integração e aceitação
  - JUnit
  - Sem barreira nítida
  - Pequenos, médios e grandes
    - P 60s
    - M Disco
    - G Rede, etc
- Resolve problemas com regressões
- Não é perda de tempo!
  - menos depuração
  - facilidade na refatoração
  - menos impactos para o cliente
  - mais tranquilidade

**Conjunto de Testes**

**Automação/Ferramentas**

**DevOps**

- Chef / Puppet / Ansible
- Git / SVN
- Jenkins
- Vagrant
- Packer
- Docker
- Refs
  - Escalando o pé de feijão... → <https://prezi.com/q7hdsqevzf5/escalando-o-pe-de-feijao-estrategias-open-source-para-automacao-na-nuvem/>
  - Fantástica Fábrica de Chocolate → <https://prezi.com/989nklbvo/a-fantastica-fabrica-de-deployment/>
  - Piratas do DevOps → <https://prezi.com/yjyj1lbb6ri/piratas-do-devops-navegando-em-aguas-misteriosas/>

**ALM - Application Lifecycle Management**

- Gerenciamento de requisitos Redmine
- Gestão de código / Controle de Versão
  - SVN
  - GIT → [readwrite.com > 2013 > 09 > 30 > Understanding-github-a-journey-for-beginners-part-1](http://readwrite.com > 2013 > 09 > 30 > Understanding-github-a-journey-for-beginners-part-1)
- Construção e dependências Maven
- Gerenciamento de arquétipos Maven
- Integração Contínua
  - Jenkins
  - Resolve problemas com integrações
- Repositório de Componentes Nexus
- Ferramenta para Desenvolvimento Eclipse IDE com plugins
- Ref MC134 - Introdução a ALM e Testes com ferramentas open source → <https://www.youtube.com/watch?v=t4YYQHGRzA>

**Caminho para o Continuous Delivery**

- 1 build
- 2 teste
- 3 integração contínua
- 4 melhorar cobertura de testes
- 5 deploy contínuo para staging
- 6 monitoramento produção
- 7 deploy contínuo para parte do cluster
- 8 propagação automática, baseado no monitoramento

Refs: Entrevista com Jonas Abreu <http://www.infoq.com/br/interviews/desenvolvimento-agil-software>

Práticas para aumentar a Produtividade

- Pair programming Aumenta o foco
- Pomodoro

## 4. Projeto (1h)

**Ref**

- Outros
  - <https://netbeans.org/kb/docs/web/hibernate-webapp.html>
  - <https://css-tricks.com/app-from-scratch-1-design/>
  - <https://www.codecademy.com/learn/learn-angularjs>
    - NearMe 1
    - NearMe 2
    - NearMe 3
  - [codeproject.com > Artigos > 546960 > HTML-Quick-Start-Web-Application](http://codeproject.com > Artigos > 546960 > HTML-Quick-Start-Web-Application)
- Se fosse ágil...
  - MVP (5min)
  - Product Backlog (5min)
  - kanban (5min)
  - 3 sprints (20min cada)
    - Sprint 1 Práticas-foco Design Thinking/Prototyping
    - Sprint 2 Práticas-foco Arquitetura/Design
    - Sprint 3 Práticas-foco

O que faltou de essencial?

- Refactoring Exemplo
- TDD JUnit
- Build/Deploy na VM
  - Config Docker/Vagrant
  - Jenkins

## 6. Próximos passos (5min)

- Coding Dojos
- Hackatons
- PMI-ACP <http://www.projectbuilder.com.br/blog-pb/entry/pratica/gerenciamento-agil-de-projetos-uma-visao-pratica>
- <https://developer.mozilla.org/en-US/docs/Web/Tutorial>
- [devzone.zend.com > 6 > Php-101-php-for-the-absolute-beginner](http://devzone.zend.com > 6 > Php-101-php-for-the-absolute-beginner)