

## Saberes e práticas contemporâneas em gestão e inovação na Educação Profissional e em Sistemas Produtivos

### Comparação entre o Método Ágil e o Tradicional no Desenvolvimento de *Software*

Claudia Regina Freneda Silva Giglio<sup>1</sup>,  
Napoleão Verardi Galegale<sup>2</sup>,  
Marília Macorin de Azevedo<sup>3</sup>

**Resumo** - Este estudo pretende demonstrar que a utilização da metodologia ágil pode ser mais produtiva do que a metodologia tradicional na construção de *software* a partir da utilização de amostras de dois projetos da área de tecnologia de um grande banco brasileiro, um utilizando metodologia ágil e o outro, metodologia tradicional. A produtividade foi calculada baseada nas horas estimadas e realizadas na construção de artefatos de sistemas. O método utilizado foi a análise quantitativa descritiva, com cálculos estatísticos a partir das amostras coletadas. Como resultado, a utilização da metodologia ágil mostrou ser mais eficiente e a hipótese que o tempo anterior na função de desenvolvedor aumenta a produtividade não demonstrou ser importante, pelo fato da correlação ser baixa.

**Palavras-chaves:** Desenvolvimento ágil, Metodologia ágil, *Scrum*, Produtividade de *Software*

**Abstract** – This study intends to demonstrate that the use of the agile methodology can be more productive than the traditional methodology in the construction of software from the use of samples of two technology projects of a large Brazilian bank, one using agile methodology and the other, Traditional methodology. Productivity was calculated based on the estimated hours and realized in the construction of systems artifacts. The method used was the descriptive quantitative analysis, with statistical calculations from the samples collected. As a result, the use of the agile methodology proved to be more efficient and the hypothesis that the previous time in the developer function increases productivity was not shown to be important, because the correlation is low.

**Keywords:** *Agile development, Agile Methodologies, Scrum, Software productivity*

---

<sup>1</sup> Centro Paula Souza, claugiglio67@gmail.com

<sup>2</sup> Centro Paula Souza, nvg@galegale.com.br

<sup>3</sup> Centro Paula Souza, marilia.azevedo@fatec.sp.gov.br

## Saberes e práticas contemporâneas em gestão e inovação na Educação Profissional e em Sistemas Produtivos

### 1. Introdução

Após a crise econômica mundial de 2008, muitas empresas verificaram nos métodos ágeis uma maneira de construir sistemas com mais eficiência e menor custo, influenciadas pela expansão dos investimentos em *software* e do aumento expressivo do uso de aplicativos via *Mobile Banking*. Com a 'Era Digital', as metodologias tradicionais de desenvolvimento de *software* não atendiam mais a necessidade de entregas rápidas, com menor custo e foco no mercado, tornando o desenvolvimento ágil o melhor método para o aumento da produtividade das equipes na indústria de *software* (EBERT; ABRAHAMSSON; OZA, 2012).

Com a popularidade do desenvolvimento ágil e o seu *framework* mais utilizado no mercado, o SCRUM, as empresas, principalmente no setor financeiro bancário, que é o estudo de caso deste artigo, adotaram a metodologia ágil no desenvolvimento dos sistemas concomitantemente com o desenvolvimento tradicional, já utilizado amplamente desde o início do mainframe, por isso, este estudo utiliza dados de projetos utilizando as duas metodologias distintamente, em equipes separadas, com medições e profissionais específicos.

A questão de pesquisa que motiva este estudo é: A utilização de métodos ágeis no desenvolvimento de *software* pode aumentar a eficiência do desenvolvedor?

Por meio da análise quantitativa descritiva de dados das amostras de dois projetos do banco, o projeto A, utilizando a metodologia ágil e o Projeto B, utilizando a metodologia tradicional, e a partir de cálculos estatístico com o intervalo de confiança para inferir o valor médio da população a partir das amostras e o Coeficiente de Correlação de Pearson, para medir a correção entre o tempo de experiência na função de desenvolvedor, e a produtividade apresenta-se as seguintes hipóteses:

H<sub>0</sub> – O projeto A (desenvolvimento ágil) tem maior produtividade que o Projeto B (desenvolvimento tradicional);

H<sub>1</sub> – Quanto maior o tempo de experiência do desenvolvedor, maior é a produtividade no desenvolvimento de *software*.

### 2. Referencial Teórico

A fundamentação teórica a seguir descreve o cenário atual da tecnologia bancária no Brasil com relação à evolução do mobile banking e dos investimentos em *software* e hardware. Em seguida, são mostradas as diversas metodologias de desenvolvimento de *software* nos últimos cinquenta anos, onde metodologias não prescritivas evoluíram para as prescritivas, como a cascata e espiral, até os métodos ágeis, muito difundidos hoje através do *framework* SCRUM. Por fim, a fundamentação teórica mostra a relação dos métodos ágeis com a produtividade.

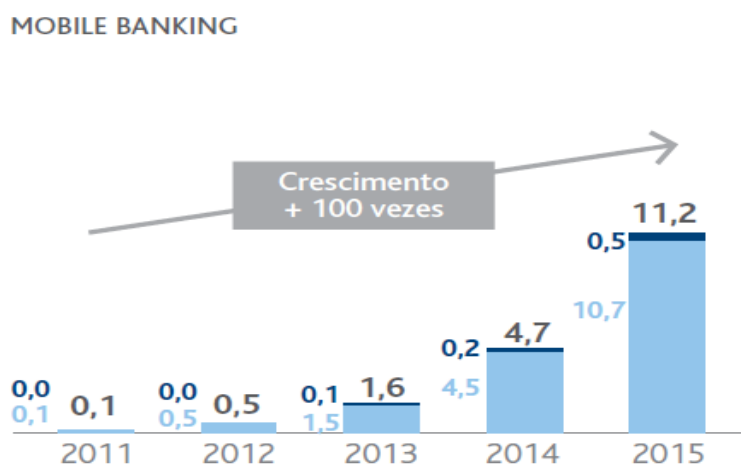
## Saberes e práticas contemporâneas em gestão e inovação na Educação Profissional e em Sistemas Produtivos

### 2.1 Tecnologia Bancária

Pesquisa realizada com 17 bancos, representando 93% da indústria bancária do Brasil, revela que os investimentos em tecnologia em 2015 foram de R\$ 19 bilhões, estando na sétima posição entre as dez economias maiores do mundo em gastos absolutos em TI no setor bancário. Essa posição se fortaleceu a partir do avanço da informatização nas redes bancárias, a integração das informações entre os bancos e o Banco Central e agências reguladoras e a enorme diversidade de produtos oferecidos a dezenas de milhões de consumidores (RELATÓRIO FEBRABAN, 2016).

A pesquisa também revela dados impressionantes da forte expansão do Mobile Banking, que aumentou 100 vezes em comparação com 2011, chegando a 11,2 bilhões em volume de transações efetuadas (Gráfico 1). Esse volume foi devido ao aumento do acesso da população à internet, o uso crescente dos *smartphones*, a inserção da mobilidade nos hábitos da população e a maior sofisticação da disponibilidade de serviços via aplicativos para dispositivos digitais.

Gráfico 1 – Mobile Banking



Fonte: Relatório Febraban (2016, p. 24)

Em termos de desafios para o futuro, os bancos enfrentam uma forte concorrência das *Fintechs*, como vem sendo chamadas as empresas emergentes do setor financeiro. O termo *Fintech* surgiu da combinação das palavras em inglês *financal* (finanças) e *technology* (tecnologia). Esse nome, por si só, resume bem a ideia: *fintech* é toda empresa que oferece serviços financeiros que se diferenciam pelas facilidades proporcionadas pela tecnologia e, com efeito, pela internet (INFOWESTER, 2016).

## Saberes e práticas contemporâneas em gestão e inovação na Educação Profissional e em Sistemas Produtivos

### 2.2 Metodologias de desenvolvimento de *Software*

As metodologias, chamadas também de modelos prescritivos de processo foram originalmente propostas para colocar ordem no caos do desenvolvimento de *software* (Pressman, 2006) porque fornecem estabilidade, controle e organização a uma atividade que não pode ser deixada sem controle.

A metodologia de desenvolvimento de *software* mais utilizada nos últimos anos é a Cascata. Ela foi criada nos anos 70 por Winston e Royce, para propor a criação de um processo organizado, sequencialmente com diversas fases bem definidas, são elas: Requisitos do Sistema, Requisitos do *Software*, Análise, Desenho do Programa, Codificação, Testes e Operação (TERLIZZI e BIANCOLINO, 2014).

Pesquisas de Kumar e Bathia (2014) mostram que os processos de desenvolvimento de sistemas tradicionais não são tão eficientes quanto os ágeis em razão do tradicional priorizar a documentação, as regras ao invés das equipes e enfatizar o planejamento ao invés da construção. Já os métodos ágeis enfatizam o planejamento adaptativo, isto é, como as entregas são curtas e o contato com o usuário é constante, as alterações são rápidas e precisas, de forma iterativa.

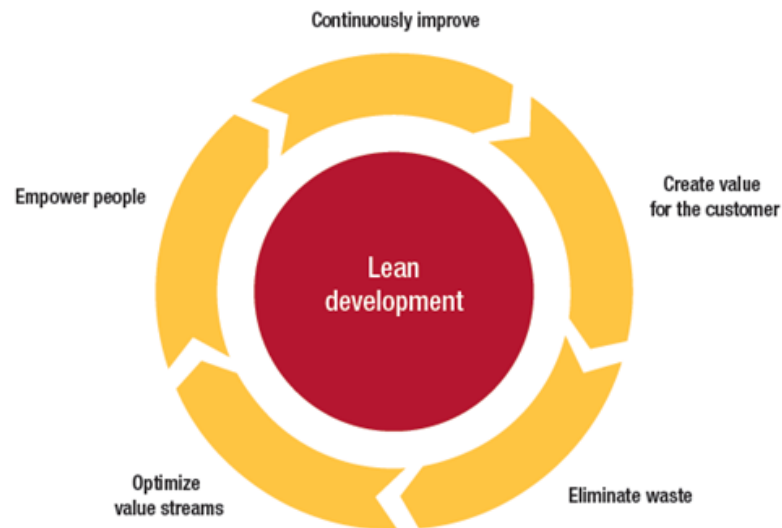
### 2.3 Métodos ágeis

Os métodos ágeis ficaram populares após o Manifesto Ágil lançado em 2001 por um grupo de estudiosos e prometiam ser mais adaptativos e flexíveis em relação aos tradicionais, tendo como princípios o desenvolvimento colaborativo, a mentalidade '*lean*', evitando trabalhos e documentações desnecessárias e clientes/stakeholders participando ativamente na construção do *software*, guiando a sua evolução (EBERT, ABRAHAMSSON, OZA, 2012).

A **Figura 1** explica melhor o ciclo de desenvolvimento utilizando os princípios ágeis, chamados também de *Lean* (enxuto):

## Saberes e práticas contemporâneas em gestão e inovação na Educação Profissional e em Sistemas Produtivos

Figura 1 – Ciclo de desenvolvimento *Lean*



Fonte: EBERT, ABRAHAMSSON, OZA (2012)

A **Figura 1** mostra que o ciclo de desenvolvimento *Lean* segue o conceito de melhoria contínua (*Continuously improve*), criação de valor para o cliente (*create value for the customer*), identificação e eliminação de desperdícios no processo (*eliminate waste*), melhoria nas interfaces (*optimize value streams*) e empoderamento das pessoas (*empower people*).

Segundo Pressman (2006), muitos conceitos ágeis são simples adaptações de bons conceitos de engenharia de *software* e há muito a ser ganho considerando ambas as escolas.

A indústria brasileira de *software*, principalmente a bancária, está no início da utilização do desenvolvimento ágil nos sistemas legados, que foram construídos com a metodologia tradicional e hoje estão sendo adaptados para serem conectados aos novos sistemas construídos pelo modelo ágil.

### 2.4 Métodos ágeis e produtividade

A produtividade no desenvolvimento de *software* é muito difícil de ser medida e depende de vários fatores, desde a motivação do programador, a experiência geral da equipe, até o seu gerenciamento (BARROS, 2010), dependendo diretamente da plataforma tecnológica utilizada, que deve ser de amplo conhecimento de todos os desenvolvedores.

Não há um consenso para a medição da produtividade no desenvolvimento de *software*, tanto no desenvolvimento ágil quanto no tradicional, porque o desenvolvimento de *software* é uma atividade humana com várias incertezas desde o início do projeto. Os fatores de maior influência na produtividade da equipe ágil

## Saberes e práticas contemporâneas em gestão e inovação na Educação Profissional e em Sistemas Produtivos

são o *Team Designer* e o *turnover* de membros da equipe. Problemas com *Team Designer* foram identificados como a diversidade da equipe, a personalidade e experiência anterior de cada membro e as alocações dos papéis na equipe. O *turnover*, que é a troca de membros da equipe, provoca uma ruptura no desenvolvimento do projeto e consequentes atrasos, porque o novo membro levará um certo tempo para conhecer o funcionamento do projeto e começar a produzir adequadamente (MELO et al, 2013).

### 3 Método

O método utilizado neste estudo é o quantitativo descritivo com fontes primárias de dados a partir de duas amostras por conglomerado divididas em duas subpopulações distintas do banco: uma amostra que utiliza dados do projeto A que utilizou o método de desenvolvimento ágil (*Scrum*) e a outra amostra do projeto B que utilizou a metodologia em Cascata (*Waterfall*). A coleta da amostra não foi probabilística e sim intencional.

#### 3.1 Análise dos dados

A análise dos dados utiliza fonte de dados primários – registro de horas e medições dos artefatos dos projetos.

- a) Variável dependente: Produtividade: calculada a partir das horas realizadas dividida pelas horas estimadas definidas pelo Guia de métricas de serviços do banco, na construção de um artefato pelo desenvolvedor. Uma variável dependente pode ser alterada pelas variáveis independentes ou pelas variáveis intervenientes.
- b) Variável independente: Número de meses que o desenvolvedor atua na função: A experiência anterior do desenvolvedor é uma variável que atua diretamente na produtividade, porque a produção dos artefatos deve ser mais rápida por causa do das experiências vividas em outros projetos.

#### 3.2 Estudo de caso

O estudo de caso examina um fenômeno em seu meio natural, com fontes de evidência (indivíduos, grupos, organizações) que podem ser obtidas pelo emprego de diferentes técnicas de coleta de dados (entrevistas, dados secundários como planilhas, atas, etc.) (SILVA, 2004).

O estudo de caso foi realizado na área de tecnologia do Banco B, um banco Brasileiro que está listado no DJSI (Índice Dow Jones de Sustentabilidade da Bolsa de Valores de Nova York). Possui mais de cinco mil agências no Brasil e está presente em mais de 20 países. Com uma carteira de 64 milhões de clientes, conta

## Saberes e práticas contemporâneas em gestão e inovação na Educação Profissional e em Sistemas Produtivos

com mais de 66 mil pontos de atendimento, sendo que está presente em 99,8% dos municípios brasileiros.

### 3.3 Guia de métricas de serviços do Banco B

O Guia de métricas do Banco B determina os esforços relativos ao serviço de desenvolvimento e manutenção de *software*, tendo em vista instruções normativas internas e recomendações de órgãos reguladores.

Em geral, os produtos dos serviços de *software* são conhecidos como “artefatos”. Portanto, os artefatos são os resultados tangíveis dos serviços de desenvolvimento e manutenção de *software* (Guia de métricas de serviços do Banco B, 2014).

A unidade de referência adotada (em horas) se baseia em cinco elementos principais: complexidade, esforço, tempo (horas), produtividade e qualificação da mão-de-obra. Com base em histórico de produtividade e a exemplo de outras metodologias de medição, os esforços relacionados às atividades de alteração equivalem a um percentual das de criação (50%).

## 4. Resultados e discussão

### 4.1 Amostras

A primeira amostra é referente ao projeto A, onde foram selecionados 86 registros de construção de artefatos novos e alterados, programas em linguagens de programação mainframe utilizando metodologia ágil (*Scrum*). A segunda amostra é do projeto B, com 166 registros relativos à construção de artefatos utilizando a metodologia tradicional em cascata. Após retirar os dados extremos, exceção no desenvolvimento de artefatos, a amostra A ficou com 82 registros e a B com 157 registros.

Os dados coletados dos dois projetos são: *Sprint* (Projeto A), Nome do artefato, Tipo do artefato (Linguagem em que ele foi codificado), USTI - Tempo estimado para a construção do artefato de acordo com o Guia de Métricas de Serviço do Banco B, quantidade de horas realizadas para a construção do artefato, nome do analista e tempo em meses de experiência na função de desenvolvedor.

### 4.2 Cálculo da produtividade média

A partir dos dados das amostras, foram realizados os cálculos estatísticos demonstrados na **Tabela 1**.

## Saberes e práticas contemporâneas em gestão e inovação na Educação Profissional e em Sistemas Produtivos

**Tabela 1** – Cálculos estatísticos do desenvolvimento ágil e tradicional

Nome	Projeto A (%)	Projeto b (%)	Descrição
Média	60,6	6,4	Média aritmética das amostras
Desvio Padrão	30,7	39,0	Grau de dispersão da coleta em relação a média
Coefficiente de Variação	0,5	6,1	Divisão do desvio padrão pela média.
Intervalo de Confiança	6,6	6,1	É utilizado para se inferir o valor da média da População
IC Negativo	54,0	0,3	Valor menor do IC, representativo da média da População.
IC Positivo	67,3	12,5	Valor maior do IC, representativo da média da População.

**Fonte:** Resultado da pesquisa, 2017.

Para uma estimativa confiável da média populacional a partir das amostras, foi utilizada a seguinte fórmula:

$$n = \left( \frac{Z_{\sigma/2} \cdot \sigma}{E} \right)^2 \quad \text{Eq.(2)}$$

Na Eq. (2), temos que  $n$  = número de artefatos da amostra;  $Z_{\sigma/2}$  = Grau de confiança desejado (95%);  $\sigma$  = Desvio padrão das amostras e  $E$  = Margem de erro, que identifica a diferença máxima entre a média amostral e a verdadeira média populacional.

No Projeto A a margem de erro máxima desejada é de 6,6%, com desvio padrão de 30,7% o que resulta em uma amostra confiável de 82 artefatos coletados.

Para o Projeto B, a margem de erro desejada é de 6,1%, com desvio padrão de 39% e amostra confiável de 157 artefatos coletados.

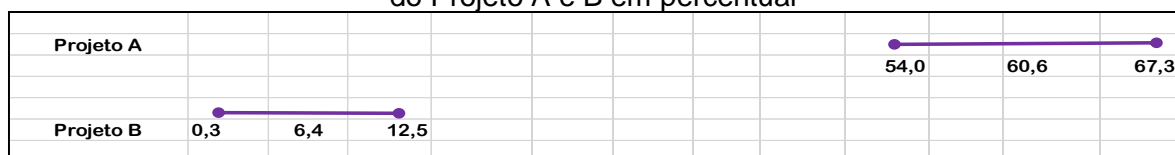
O valor 0 (zero) de produtividade indica que o tempo estimado para a construção do *software* foi o mesmo do realizado.

O cálculo da média mostra que o Projeto A resultou em uma produtividade quase dez vezes maior que o projeto B.

### 4.3 Resultados baseados no intervalo de confiança

Conforme a comparação dos intervalos de confiança da **Tabela 1** referentes aos Projetos A e B, verifica-se que os dois não tem interseção nos valores, portanto, prova-se que o Projeto A tem valores de produtividade bem superiores ao Projeto B, como exibido no **Gráfico 2**:

**Gráfico 2** – Comparação dos valores do intervalo de confiança negativo, médio e positivo do Projeto A e B em percentual



**Fonte:** Resultado da pesquisa, 2017.



## Saberes e práticas contemporâneas em gestão e inovação na Educação Profissional e em Sistemas Produtivos

Conforme o resultado apresentado no **Gráfico 2**, a  $H_0$  - O projeto A (desenvolvimento ágil) tem indicadores de produtividade melhores que o Projeto B (desenvolvimento tradicional) foi provada.

### 4.4 Coeficiente de correlação de Pearson

O coeficiente de correlação de Pearson mede o grau da correlação (e a direção dessa correlação - se positiva ou negativa) entre duas variáveis de escala métrica que podem assumir valores entre -1 e 1 (FARBER e LARSON, 2016).

Se o coeficiente for = 1, significa que há uma correlação perfeita positiva entre as duas variáveis, isto é, se uma variável aumenta, a outra variável aumenta, se uma variável diminui, a outra variável diminui.

Para uma correlação negativa entre as variáveis, se uma aumenta, a outra sempre diminui. A fórmula do coeficiente de correlação de Pearson é calculada conforme a Eq. (2):

$$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{(\sum (x_i - \bar{x})^2)(\sum (y_i - \bar{y})^2)}} \quad \text{Eq. (2)}$$

A variável X é a produtividade e a variável Y é o tempo de experiência na função. Existe correlação entre duas ou mais variáveis quando as alterações sofridas por uma são acompanhadas por modificações (para mais ou para menos) nas outras. Ou seja, no caso de duas variáveis x e y os aumentos (ou diminuições) em x correspondem a aumentos (ou diminuições) em y.

Os valores calculados para a verificação da Correlação variam de -1 a +1, conforme o **Quadro 1**:

**Quadro 1** – Valores das correlações mínimas e máximas de Pearson

Valor calculado	Descrição
+1	Correlação positiva perfeita
+0,70 a +0,99	Correlação positiva muito forte
+0,50 a +0,69	Correlação positiva substancial
+0,30 a +0,49	Correlação positiva moderada
+0,10 a +0,29	Correlação positiva baixa
+0,01 a +0,09	Correlação positiva ínfima
0,00	Nenhuma correlação
-0,01 a -0,09	Correlação negativa ínfima
-0,10 a -0,29	Correlação negativa baixa
-0,30 a -0,49	Correlação negativa moderada
-0,50 a -0,69	Correlação negativa substancial
-0,70 a -0,99	Correlação negativa muito forte
-1	Correlação negativa perfeita

**Fonte:** Farber e Larson (2016)

## Saberes e práticas contemporâneas em gestão e inovação na Educação Profissional e em Sistemas Produtivos

De acordo com o resultado do cálculo da correlação feito no *software* Excel, para o Projeto A o coeficiente é  $-0,02$  (correlação negativa ínfima) e para o projeto B é  $-0,18$  (correlação negativa baixa), não provando a  $H_1$ , que o tempo de experiência na função de desenvolvedor aumenta a produtividade e a influência é mínima da variável interveniente 'tempo de experiência na função de desenvolvedor de *software*'.

### 5. Considerações Finais

Em resposta à questão de pesquisa apresentada sobre a utilização de métodos ágeis no desenvolvimento de *software*, se a mesma aumenta a eficiência do desenvolvedor, foram abordados os aspectos do desenvolvimento de *software* na indústria bancária, a evolução das metodologias de desenvolvimento até a utilização dos métodos ágeis, especificamente o *Scrum*.

A partir de dois projetos na área de tecnologia bancária, o primeiro utilizando metodologia ágil e o segundo tradicional, foi provado a partir do intervalo de confiança da variável produtividade que o método Ágil foi mais eficiente que o Cascata ( $H_0$ ), corroborando com os estudos de Ebert; Abrahamsson; Oza (2012), que a utilização dos princípios '*Lean*' tornam o desenvolvimento mais eficiente (empoderamento das equipes, redução de desperdícios, foco no mercado e com o acompanhamento do usuário). A Hipótese 1 não foi provada, sendo que para os dois projetos, a experiência anterior do desenvolvedor tem pouca influência na produtividade, conforme resultados utilizando o coeficiente de correlação de Pearson.

Existe uma limitação deste trabalho em relação às amostras, que poderiam ser maiores, mas em razão do Banco B estar iniciando os seus projetos com o método ágil, não foi possível conseguir amostras maiores. A situação particular das amostras não pode ser generalizada para todas as situações de desenvolvimento de *software* e suas particularidades, afinal os projetos eram diferentes e não foram mapeadas todas as possíveis variáveis que possam ter influenciado os scores de produtividade.

Outra limitação é em relação à utilização da USTI, única métrica utilizada pelo banco. A mesma pesquisa deve ser realizada em outro banco para obtermos resultados mais abrangentes.

Este artigo pretende contribuir com a indústria de *software* em relação a persistência na utilização dos métodos ágeis, mesmo com problemas práticos ou falta de apoio dos membros da equipe, que podem não se interessar na participação nas reuniões diárias ou integração com os outros membros da equipe.

As práticas de desenvolvimento ágeis são recentes e precisam de uma maior maturidade na sua utilização.

Novos estudos devem ser realizados para identificar os pontos críticos no processo de desenvolvimento ágil como sugestão de melhorias e estudos futuros.

## Saberes e práticas contemporâneas em gestão e inovação na Educação Profissional e em Sistemas Produtivos

### REFERÊNCIAS

- BARROS, E. **Catálogo de fatores que influenciam a produtividade no desenvolvimento de software**. 153f. Dissertação (Mestrado) – Universidade Federal do Pernambuco – CIN – Centro de Informática, 2010.
- EBERT, C; ABRAHAMSSON, P.; OZA, N. **Lean Software Development**. IEEE *Computer Society*, 2012.
- FARBER, L.; LARSON, R. **Estatística aplicada**. São Paulo. Pearson Education do Brasil, 2016. 656p.
- INFOWESTER. **O que é fintech?** Disponível em: <<http://www.infowester.com/fintech.php>>. Acesso em: 15 nov. 2016.
- KUMAR, G.; BHATIA.PK. **Comparative analysis of software engineering models from traditional to modern methodologies. Fourth International Conference on Advanced Computing & Communication Technologies**. IEEE, 2014.
- MELO, C.; CRUZES, D.; KON, F.; CONRADI, R. **Interpretative case studies on agile team productivity and management. Information and Software Technology**. Departamento de Ciência da Computação, Universidade de São Paulo, São Paulo, 2013.
- MONTAIN GOAT SOFTWARE. **What is Scrum?** Disponível em: <<http://www.mountangoatsoftware.com/agile/scrum>>. Acesso em 30 out. 2016.
- PRESSMAN, R. **Engenharia de Software**. São Paulo. Editora McGraw-Hill, 2006. 720p.
- RELATÓRIO FEBRABAN. **Pesquisa FEBRABAN de Tecnologia Bancária 2015**. São Paulo, 2015. Disponível em: <<https://portal.febraban.org.br/pagina/3106/48/pt-br/pesquisa>>. Acesso em: 16 out. 2016.
- SILVA, R. **O teletrabalho e suas influências na qualidade de vida no trabalho**. Dissertação. Universidade de São Paulo, 2004.
- TERLIZZI, M.; BIANCOLINO, C. **Projeto de Software no Setor Bancário: Scrum ou Modelo V**. TAC. Rio de Janeiro, v. 4, n. 1, art. 4, pp. 46-58, Jan./Jun. 2014.