

# Comparando Quantitativamente Processos de Software - Experiências com Práticas Ágeis

Igor M. Pereira, Vicente J. P. Amorim, Lucas C. Lima

Departamento de Computação e Sistemas (DECSI) - Universidade Federal de Ouro Preto (UFOP)

Caixa Postal 24 – 35.931-008 – Minas Gerais – MG – Brasil

{igormuzetti, vjpamorim, lucaslima2004}@gmail.com

**Resumo.** *A adaptação de processos de software em projetos reais é comumente reportada através de dados qualitativos. Contudo, dados quantitativos podem ser usados para auxiliar a tomada de decisão e a comparação de alternativas. Este trabalho tem o objetivo de verificar a melhoria de um processo de software utilizando técnicas de processos experimentais sistemáticos. As práticas de engenharia de software utilizadas no processo de software observado são oriundas de métodos ágeis. Os resultados indicam que as práticas ágeis de construção e testes influenciam significativamente na melhoria de processos e este controle pode ser realizado com um baixo custo em pequenas organizações.*

**Abstract.** *An adaptation of software processes to real projects is commonly communicated through qualitative data. However, quantitative data can be used to aid decision making and a set of alternatives. This work aims to verify an improvement of a software process using techniques of systematic experimental processes. Software engineering practices used in the observed software process come from agile methods. Results indicate that the construction and testing practices significantly influence process improvement and this control can be carried out at a low cost in small organizations.*

## 1. Introdução

Melhorar a qualidade do produto pode ser o objetivo principal de uma iniciativa de melhoria de processo de software. Diversas organizações alcançam melhorias na qualidade de seus produtos controlando diversos atributos de qualidade através da adaptação de seus processos. Tais adaptações podem ser realizadas através do seguimento de modelos e normas para melhoria de processos, como o uso do CMMI, da SPICE, entre outros [García-Mireles *et al.*, 2015]. Estudos apontam que o uso do MPS.BR, Modelo para Melhoria do Processo de Software Brasileiro, também impacta positivamente na qualidade dos produtos desenvolvidos em organizações hospedadas no setor público e privado e em ambientes de pesquisa e desenvolvimento. As organizações compreendem que para gerenciar a qualidade de seus produtos, seus processos precisam ser definidos, seguidos e avaliados sistematicamente [Weber *et al.*, 2015].

A adaptação de processos de software para projetos reais é usualmente realizada. Essa personalização busca atender as características específicas dos diversos tipos de projetos de software e é conhecida pela comunidade de engenharia de software seja na academia ou em entidades desenvolvedoras [Kalus, 2013]. É comum encontrar relatos

de experiência e estudos de caso que reportam lições aprendidas, positivas e negativas, ou discutem as diversas adaptações das práticas, técnicas e métodos que compõem os processos de software [Turk, 2005; Fuggetta, 2014]. Entretanto, a comparação de processos pode ser realizada também através de uso de técnicas de processos experimentais sistemáticos que auxiliam a tomada de decisão e a comparação de alternativas através da sumarização de dados e outras técnicas estatísticas [Jain, 1991].

Entidades desenvolvedoras de software sabem que para gerenciar a qualidade dos seus produtos, elas precisam definir processos, segui-los e avaliá-los sistematicamente [Svensson *et al.*, 2012]. Este trabalho compara a evolução de um processo de software formado por práticas ágeis de desenvolvimento durante um ano em um projeto real. O projeto foi executado no laboratório de pesquisa em computação, denominado iMobilis, que fica hospedado no Instituto de Ciências Exatas e Aplicadas (ICEA) da Universidade Federal de Ouro Preto (UFOP). O projeto é caracterizado como inovador e surgiu de uma parceria entre a UFOP e uma grande empresa privada atuante na região que incentiva a pesquisa. Participam do projeto cinco alunos dos últimos períodos dos cursos de Engenharia da Computação e Sistemas de Informação, três professores destes cursos e um especialista em Engenharia da Computação (aluno formado). Como é uma equipe constituída por dez pessoas que já se conhecem a algum tempo, a comunicação face a face e a auto-organização são muito realizadas. Tal fato incentivou o uso de práticas ágeis na composição do processo a ser seguido no projeto.

Práticas ágeis tentam resolver diversas dificuldades enfrentadas por equipes de desenvolvimento, como entregar um produto final que realmente gere valor ao cliente e adaptar um processo às mudanças que ocorrem em um projeto que o cliente participa ativamente do seu desenvolvimento [Jalali *et al.*, 2012]. De acordo com Gomes (2014) o conjunto de práticas ágeis existente pode ser comparado com uma caixa de ferramentas, na qual, para cada projeto, um subconjunto de práticas pode ser mais eficiente. O autor também as classificou em seis grandes grupos: requisitos, projeto, construção, testes, processo, organização, aprendizado e gestão ágil. O processo observado neste trabalho evoluiu de práticas de requisitos, processo e organização para em um segundo momento além desses grupos, de construção e testes. Ou seja, no primeiro momento o foco do processo era mais voltado para o controle e monitoramento do projeto e o segundo momento o foco foi em técnicas essenciais de desenvolvimento em equipe para entregar software em funcionamento.

A inserção e adaptação de práticas de engenharia de software em um processo acontecem pela busca da melhor qualidade dos produtos almejada pelas entidades desenvolvedoras. O alcance de uma melhor qualidade dos produtos concede melhor posição no mercado e fidelidade de clientes e usuários, o que pode ser muito vantajoso, seja para projetos desenvolvidos por entidades privadas ou públicas [Weber *et al.*, 2015]. Projetos desenvolvidos por equipes ágeis, muitas vezes encontradas em pequenas empresas, *startups* e laboratórios de tecnologia e inovação, não possuem uma equipe específica para o acompanhamento rotineiro da melhoria de qualidade de seus processos e produtos. Contudo, essas informações são muito úteis para auxiliar essas entidades a tomar decisões e decidir sobre alternativas. Este trabalho apresenta uma análise de métricas coletadas pela própria equipe empiricamente em um projeto durante um ano. A análise feita conta com técnicas de estatística descritiva e indicaram que o segundo momento do processo se apresentou significativamente melhor do que o primeiro momento.

De acordo com o problema e o contexto especificado, esta pesquisa responde à seguinte questão: Como comparar estatisticamente processos de software seguidos por pequenas equipes? Por isso, o objetivo geral desta pesquisa foi analisar as médias das métricas de processo coletadas durante um ano com o propósito de caracterizar as práticas ágeis que influenciam na melhoria do processo de software seguido do ponto de vista de desenvolvedores de software no contexto de projetos reais desenvolvidos por equipes pequenas.

O restante deste relato está organizado conforme proposto para estudos de caso conforme sugere Runeson *et al.*, (2012). Inicialmente, é discutida uma revisão da literatura na Seção 2. A seguir, na Seção 3, o estudo de caso é descrito. Na Seção 4 os resultados obtidos são apresentados e discutidos. Por fim, na Seção 5 são apresentadas as considerações finais, as possibilidades de trabalhos futuros e as limitações do estudo.

## 2. Revisão da Literatura

Unterkalmsteiner *et al.*, (2012) realizaram uma revisão sistemática na literatura sobre medições e avaliações de iniciativas de melhoria de processos de software. Eles identificaram os tipos de métricas de software mais utilizadas nas iniciativas de melhoria e descobriram que as do tipo processo, como produtividade durante o desenvolvimento e taxa de defeitos por fase de desenvolvimento são as mais utilizadas. Eles também relataram que o uso do CMMI (*Capability Maturity Model Institute*) é o mais abordado nessas iniciativas. Entretanto este tipo de controle é usado por empresas já certificadas em altos níveis do CMMI, sendo elas de médio ou grande porte. Esta pesquisa analisa sistematicamente métricas do tipo processo coletadas com um baixo custo por uma equipe pequena que deseja manter a melhoria contínua de seu processo.

Kalus e Kuhrmann (2013) fizeram uma revisão sistemática e identificaram e discutiram quarenta e nove critérios comumente utilizados na adaptação de práticas de engenharia de software em processos seguidos por projetos reais. Eles concluíram que as práticas precisam sim ser adaptadas aos diferentes tipos de projetos, mas não existe uma definição de quais critérios básicos precisam ser considerados em uma adaptação. Este trabalho também usou a adaptação de práticas descritas na literatura para as particularidades do processo observado. Os critérios que envolvem o ambiente externo, como participação dos usuários e clientes e objetivos como integração contínua e testes, levantadas pelos autores representam as práticas que caracterizam a melhoria do processo obtida nesta pesquisa.

Magalhães *et al.*, (2015) apresentam resultados preliminares referentes às linhas de processo de software do processo de análise de desempenho e uma ferramenta para instanciar e executar o processo de análise de desempenho. Segundo os autores este tipo de atividade deve ser realizada com o auxílio de técnicas e ferramentas estatísticas. Contudo, o estudo de Magalhães e colaboradores não apresenta resultados de projetos reais. Este trabalho não utiliza um software para o controle estatístico do processo, mas usa técnicas estatísticas que podem ser aplicadas às métricas de software coletadas durante a execução de projetos reais.

Equipes ágeis são normalmente formadas por times pequenos que dão maior importância às pessoas e interações do que para processos e ferramentas. Por este motivo, elas não costumam seguir um processo padrão completo, elas costumam adaptar as práticas de engenharia de software conforme as particularidades de seus

projetos. Fontana *et al.*, (2015) discutem como a melhoria de processos para estes tipos de equipes pode ser medido. As autoras concluíram que as equipes de desenvolvimento de software ágeis evoluem naturalmente. Elas analisaram a evolução das práticas de nove equipes e descobriam que o processo era, como esperado, peculiar. Cada equipe observada adotou práticas baseadas em suas circunstâncias e melhorou as práticas com base nos desafios que enfrentaram. Desta maneira, este trabalho não foca em apresentar como o time observado adaptou as novas práticas inseridas no processo para melhorá-lo. Foi considerado que outros times podem adaptar as mesmas práticas de maneira diferente, o que se quis demonstrar neste trabalho foi que elas realmente impactaram significativamente na melhoria do processo.

Reconhece-se a relevância e a importância que a melhoria de processos de software e o desenvolvimento ágil ganharam no campo da engenharia de software. Ambas são abordagens que aumentam a eficiência e a eficácia de uma organização de desenvolvimento de software e aprimoram seus produtos. Santana *et al.*, (2015) apresentam em seu trabalho a identificação e caracterização de melhoria de processos em ambientes ágeis através de uma revisão sistemática da literatura, incluindo 423 trabalhos publicados entre 2001 e Março/2013. Os autores concluíram que conduzir iniciativas de melhoria em ambientes ágeis é diferente do tradicional, tornando-se necessário adaptar as abordagens de melhorias existentes ou criar novos métodos. Neste trabalho foi adaptado um método de comparação quantitativa de processos sistemáticos em um processo de software que utiliza práticas ágeis, cujo intuito foi descobrir se o desempenho havia melhorado com a inserção de novas práticas orientadas mais ao desenvolvimento e testes enquanto que anteriormente as práticas utilizadas eram mais voltadas à gestão.

É comum ao implantar um processo de software em um projeto real, no primeiro momento, escolher práticas de engenharia de software mais voltadas para o gerenciamento de projetos e no segundo momento incorporar o processo com técnicas mais focadas ao desenvolvimento de engenharia de software. Por exemplo, muitas organizações implantam primeiramente o Scrum em seus projetos, que é uma metodologia focada na gestão ágil. Depois de implantado é que se elas aplicam práticas da Programação Extrema para dar maior valor à qualidade dos produtos entregues [Chaves, 2011]. Porém, estudos empíricos que comprovem estatisticamente estas evoluções de processos, não são fáceis de encontrar.

### 3. Metodologia

Esta seção apresenta a metodologia seguida para alcançar o objetivo deste trabalho. O laboratório de computação móvel e sistemas embarcados, iMobilis<sup>1</sup>, fica hospedado no campus da Universidade Federal de Ouro Preto (UFOP) em João Monlevade. O iMobilis desenvolve projetos de pesquisa em computação que estão relacionados à construção de soluções para plataformas móveis e sistemas embarcados com o auxílio de práticas de engenharia de software. É neste ambiente que foi realizado o projeto real do estudo de caso desta pesquisa, conhecido como *quasi-experimento*, segundo Wolhin *et al.*, (2012). Os projetos do iMobilis são atrelados à programas de incentivo à pesquisa fomentados pelo governo federal e pela própria UFOP e com parcerias público-privadas que a UFOP possui com empresas do setor privado que atuam na região.

---

<sup>1</sup> [www.imobilis.ufop.br](http://www.imobilis.ufop.br)

As ferramentas de apoio utilizadas para coleta de métricas foram: a Dokuwiki<sup>2</sup> para gestão da documentação dos projetos, o GitLab<sup>3</sup> é utilizado para o gerenciamento de configuração e mudança e o uma própria para gestão dos projetos em geral. Todas as ferramentas utilizadas foram dadas prioridades para projetos *open-source*, apenas a ferramenta de gerenciamento de projetos ainda não possui licença, pois está sendo finalizada como outro projeto do iMobilis. Acredita-se que outras ferramentas correlatas a estas podem ser utilizadas em replicações deste estudo.

O método deste estudo contou com um processo experimental sistemático realizado para comparar dois momentos de um processo de software durante um ano de execução de um projeto real executado por uma mesma equipe de dez colaboradores. O intuito foi avaliar se a inserção de práticas ágeis focadas no desenvolvimento e testes impacta significativamente na melhoria do processo anteriormente composto por práticas de gestão. O processo experimental sistemático foi seguido pelos seguintes passos:

- i. Entendimento do problema e identificação da questão que se desejava responder e definição dos objetivos.
  - O problema que se desejava resolver foi como verificar que ao longo de projetos reais se a inserção de práticas ágeis de desenvolvimento de software influencia a melhoria do processo.
  - A questão que necessitava de uma resposta era: Como comparar momentos de um processo ou processos diferentes, executados por pequenas organizações que não possuem muitos recursos para aplicarem técnicas dos mais altos níveis dos modelos de maturidade seguidos no mercado, com dados quantitativos?
  - O principal objetivo da pesquisa foi verificar se as médias das métricas de dedicação e velocidade coletadas durante quinze meses em projetos que seguem ritmo correlato a este são úteis para demonstrar o quão significativo um processo é do outro.
- ii. Seleção de métricas. As métricas selecionadas foram de dedicação e velocidade relativas.
  - Elas indicam o quanto os colaboradores trabalharam no projeto em razão do que eles deveriam trabalhar e o quanto eles entregavam de valor ao cliente em razão do que eles negociavam que iriam entregar, respectivamente [Pereira *et al.*, 2013]. Todas as métricas foram coletadas semanalmente pelo scrum *master* acessando as ferramentas de apoio ao processo.
- iii. Identificação dos parâmetros.
  - Os parâmetros considerados foram todas as práticas ágeis presentes nos dois momentos do processo e mais as novas práticas inseridas somente no segundo momento. São elas: Reuniões diárias, de demonstração, de

---

<sup>2</sup> [www.dokuwiki.org](http://www.dokuwiki.org)

<sup>3</sup> <https://gitlab.com>

retrospectivas, iterações, *sprint backlog*, *burdown chart*, *product backlog*, *planning poker*, times pequenos, ambientes de trabalho compartilhado, cliente interno e *scrum master*, essas são práticas mais voltadas para o gerenciamento de projetos. E as práticas focadas no desenvolvimento que foram inseridas no segundo momento foram: desenvolvimento dirigido por comportamento, entregas frequentes, testes automatizados de aceitação e exploratórios.

- iv. Decidir quais os parâmetros seriam estudados.
  - Os parâmetros escolhidos foram as quatro práticas focadas no desenvolvimento presentes no segundo momento do processo: desenvolvimento dirigido por comportamento, entregas frequentes, testes automatizados de aceitação e exploratórios.
- v. Seleção das técnicas.
  - Sumarização de dados, verificação que os dados seguem uma distribuição normal e comparação de observações não pareadas.
- vi. Seleção da carga de trabalho.
  - A carga de trabalho escolhida foi a média das duas métricas de todos os colaboradores durante os seis primeiros meses que representam o primeiro momento do processo e projeto e os outros nove meses posteriores que correspondem ao segundo momento do processo e projeto. Esses dados foram coletados nas ferramentas de gerencia de configuração e de projetos utilizadas pela equipe e não dependem de ferramentas específicas.
- vii. Execução dos experimentos.
  - Os experimentos foram executados durante quinze meses.
- viii. Análise, interpretação e apresentação dos resultados e da conclusão.
  - Esses últimos passos foram realizados no final dos quinze meses de processo.

#### 4. Resultados

Os dados considerados nesta comparação de experimentos não pareados foram as médias das métricas de dedicação e velocidade relativas coletadas durante os quinze meses de projeto. Eles podem ser visualizados nas Tabelas 1 e 2.

**Tabela 1. Média da dedicação e velocidade relativa da equipe durante os sprints do Processo A**

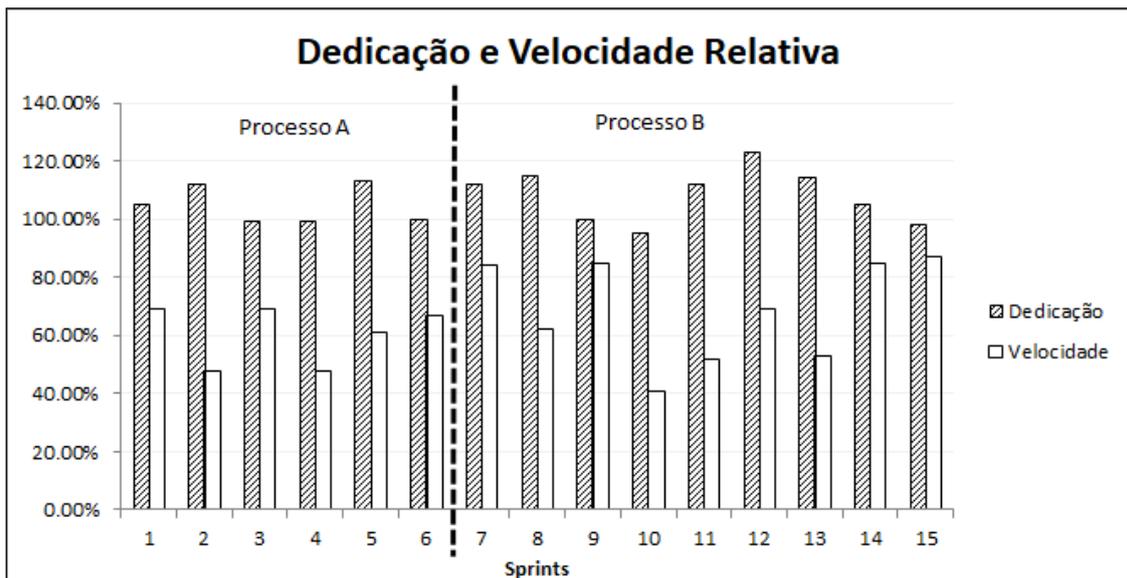
	Processo A	
	Dedicação	Velocidade
Sprint 1	105%	69%
Sprint 2	112%	48%
Sprint 3	99%	69%

Sprint 4	99%	48%
Sprint 5	113%	61%
Sprint 6	100%	67%

**Tabela 2. Média da dedicação e velocidade relativa da equipe durante os sprints do Processo B**

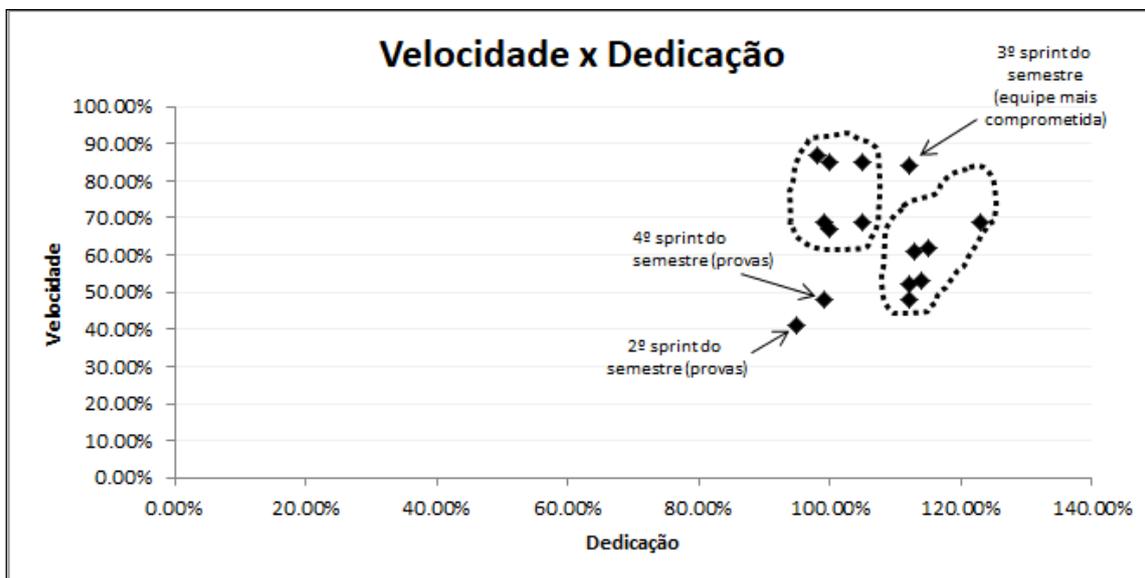
	Processo B	
	Dedicação	Velocidade
Sprint 7	102%	84%
Sprint 8	115%	62%
Sprint 9	100%	85%
Sprint 10	95%	41%
Sprint 11	112%	52%
Sprint 12	123%	69%
Sprint 13	114%	53%
Sprint 14	105%	85%
Sprint 15	98%	87%

Estes dados também estão apresentados na Figura 1. Para a métrica de dedicação relativa, a média simples da amostra dos seis sprints iniciais (Processo A) foi de 104,67% com desvio padrão ( $S_a$ ) 0.064. Para a métrica de velocidade relativa, a média simples da amostra do Processo A foi de 60.33% com desvio padrão ( $S_a$ ) 0.099. Para os outros nove sprints (Processo B), a média simples da amostra de dedicação relativa foi 108.22% com desvio padrão ( $S_b$ ) de 0.092. A velocidade relativa para o Processo B ficou na média de 68.67% com desvio padrão ( $S_b$ ) 0.174.



**Figura 1. Gráfico das médias das métricas de dedicação e velocidade relativa ao longo dos sprints**

A Figura 2 ilustra a simulação de uma *clusterização* que destaca cada componente e *outlier*:



**Figura 2. Componentes e outliers dos parâmetros estudados**

Cada componente, pontos dentro da linha pontilhada, corresponde a fatos corriqueiros em projetos reais desenvolvidos em laboratórios de pesquisa em computação inseridos em universidades. É de se esperar que os colaboradores (alunos) dediquem próximo àquilo que foi acordado e que terminem atividades que eles disseram que iriam terminar em um determinado período de tempo. O fato de dedicarem muito mais que o esperado, não faz com que as atividades sejam terminadas da maneira correta. Não é intuito do processo, exigir uma dedicação além da aceitável para finalização das atividades. Então, quando existe uma dedicação próxima aos 100%, a velocidade da equipe costuma ser satisfatória e acima da média. Quando a dedicação é muito superior aos 100%, a velocidade relativa tende a cair, ou seja, pode estar ocorrendo um mal entendido de como resolver as atividades ou as mesmas estão em uma granularidade maior que deveriam ser.

Cada semestre possui quatro sprints e normalmente as avaliações das disciplinas que os alunos cursam acontecem após o primeiro mês de aula e no último mês do semestre. Por isso, é esperado que nestes momentos a dedicação ou a velocidade caiam. Normalmente, para equipes maduras em seguir um tipo de processo em projetos anteriores, o primeiro e terceiro sprints de um semestre tendem a ser mais produtivos [Pereira *et al.*, 2013]. Isto explica os *outliers* encontrados.

Para seguir com o protocolo da técnica de comparação de sistemas com observações não pareadas, a diferença entre as médias simples foi calculada,  $\bar{x}_a - \bar{x}_b$ , obtendo-se -3.55 para a dedicação e -8.34 para a velocidade. O desvio padrão das diferenças também foi calculado:

$$S = \sqrt{\frac{S_a^2}{n_a} + \frac{S_b^2}{n_b}} \rightarrow S = 0.039 \text{ (dedicação) e } 0.069 \text{ (velocidade)}$$

Após isto, os graus efetivos de liberdade foram calculados:

$$V = \frac{\left(\frac{n_a^2}{s_a} + \frac{n_b^2}{s_b}\right)^2}{\frac{1}{n_a - 1} * \left(\frac{n_a^2}{s_a}\right)^2 + \frac{1}{n_b - 1} * \left(\frac{n_b^2}{s_b}\right)^2} - 2 \rightarrow V = 13 \text{ (dedicação) e } 11 \text{ (velocidade)}$$

Verificamos através dos gráficos *quantil-quantil* que ambas as distribuições, dedicação e velocidade relativas, são linearmente relacionadas, ou seja, os dados seguem uma distribuição normal. A Tabela 2 e as Figuras 3 e 4 apresentam os dados e os gráficos usados para verificar as distribuições normais.

**Tabela 3. Dados das métricas de dedicação e velocidade relativas**

<i>Quantil-Quantil</i> Dedicação Relativa			<i>Quantil-Quantil</i> Velocidade Relativa		
i	fi	Dedicação	i	fi	Velocidade
1	0.040983607	0.95	1	0.040983607	0.41
2	0.106557377	0.98	2	0.106557377	0.48
3	0.172131148	0.99	3	0.172131148	0.48
4	0.237704918	0.99	4	0.237704918	0.52
5	0.303278689	1.00	5	0.303278689	0.53
6	0.368852459	1.00	6	0.368852459	0.61
7	0.43442623	1.05	7	0.43442623	0.62
8	0.5	1.05	8	0.5	0.67
9	0.56557377	1.12	9	0.56557377	0.69
10	0.631147541	1.12	10	0.631147541	0.69
11	0.696721311	1.12	11	0.696721311	0.69
12	0.762295082	1.13	12	0.762295082	0.84
13	0.827868852	1.14	13	0.827868852	0.85
14	0.893442623	1.15	14	0.893442623	0.85
15	0.959016393	1.23	15	0.959016393	0.87

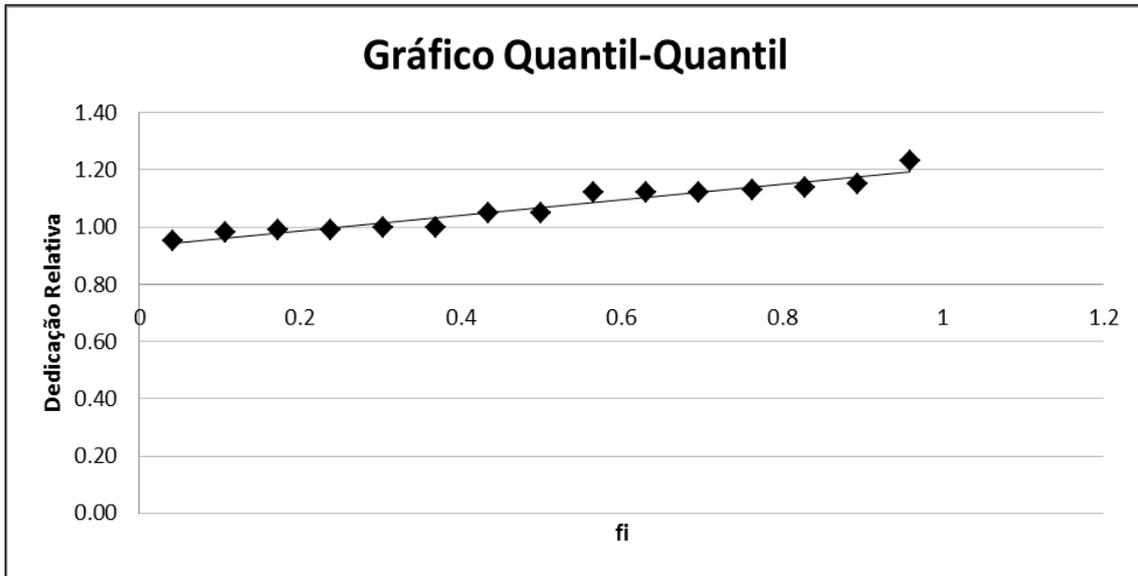


Figura 3. Gráfico *Quantil-Quantil* para os dados de dedicação relativa

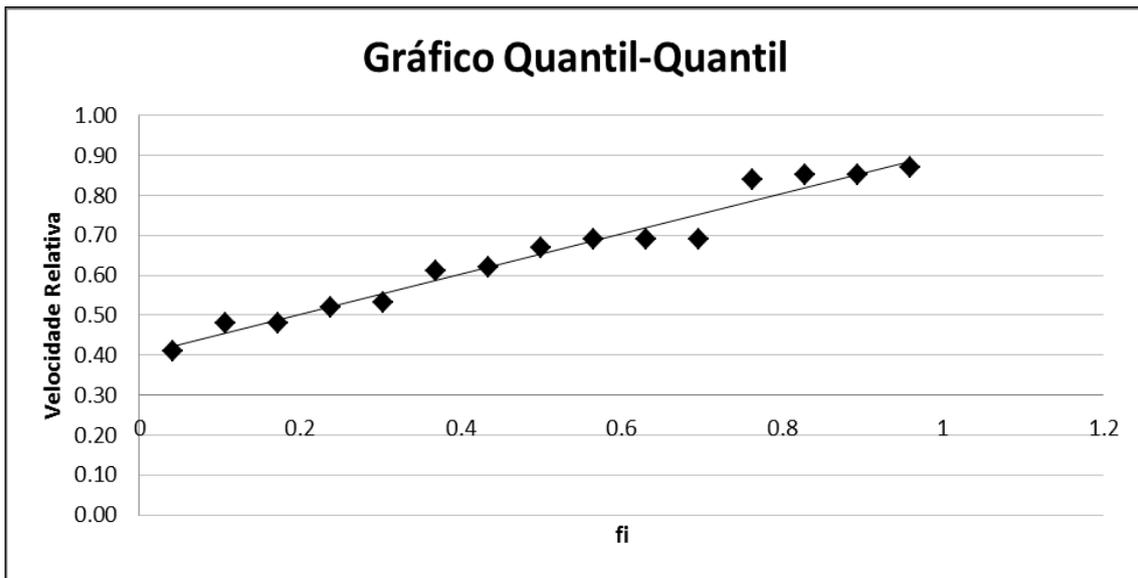


Figura 4. Gráfico *Quantil-Quantil* para os dados de velocidade relativa

Por isso, utilizamos o *teste-t* para calcular o intervalo de confiança:

$$(\bar{x}_a - \bar{x}_b) \mp t_{[1-\frac{\alpha}{2};v]}^s$$

Considerando 90% de confiança o intervalo para a dedicação fica [-3.48; -3.62]. Para 90% de confiança o intervalo para a velocidade fica [-8.21; -8.46]. Como ambos os intervalos não incluem zero, o sinal da diferença das médias indica que o Processo B, é melhor significativamente que o Processo A em ambas as métricas. Ou seja, as práticas envolvidas no Processo B que não estão presentes em A, favoreceram a melhoria do processo seguido.

## 5. Considerações Finais

O objetivo de replicar um estudo de caso em engenharia de software ainda carece de um conjunto de conceitos padronizados, terminologias, métodos adequados e em como essas informações são compartilhadas. Magalhães *et al.*, (2014) considera que esses fatos podem ser considerados como impactos negativos sobre replicações de estudos em nossa área. Para tratar esta situação, este estudo envolveu um esforço que apoiou a observação de processos não pareada que pode ser replicada em outros projetos reais executados por pequenas organizações que não possuem ainda um controle estatístico de seus processos.

Observou-se que quando foca-se em práticas de gestão de projetos e depois em práticas mais técnicas de engenharia de software, para adaptar à realidade de uma organização, o direcionamento da adaptação quando guiado pelas métricas organizacionais que os envolvidos querem atingir, influencia positivamente na caminhada aos objetivos organizacionais. Embora o processo utilizado tenha seguido a implantação de práticas de gerenciamento de projetos em primeiro lugar e em segundo as práticas mais técnicas, os autores sugerem que esta mesma abordagem pode ser utilizada com outros subconjuntos de práticas.

Percebeu-se que a adaptação das práticas cujo foco é o gerenciamento de projetos, que foram aplicadas no primeiro momento do processo, também alcançou resultados positivos. Conclui-se que subconjuntos dessas práticas podem sim ajudar na organização geral de um projeto e na obtenção de uma garantia que uma determinada ordem é seguida pela equipe. Ou seja, adaptar primeiramente práticas dessa natureza ou garantir que elas sejam seguidas podem favorecer o início de uma melhoria de processo. Considerando as práticas ágeis utilizadas no Processo A desta pesquisa, destaca-se as seguintes características percebidas pelos membros da equipe: As reuniões diárias, de demonstração e retrospectivas promovem a comunicação informal e constante entre a equipe. O seguimento do *burndown chart* e a realização do *poker* do planejamento deixam o planejamento e o controle das atividades e entregáveis do projeto transparente para toda equipe. Usufruir de um *scrum master*, cliente interno, ambiente compartilhado e um time pequeno, favorece a coesão da equipe durante a execução do projeto deixando claro as diferentes responsabilidades.

Os resultados estatísticos obtidos validam a premissa que a inserção de práticas de engenharia de software mais técnicas em um processo conhecido e bem seguido pelos envolvidos induz a resultados melhores considerando o comprometimento e o amadurecimento da equipe. Quando a equipe já estava acostumada com o Processo A e gerando bons resultados de dedicação e velocidade relativas, as práticas focadas no

desenvolvimento e entrega serviram para elevar o nível da maturidade dela. O desenvolvimento dirigido por comportamento promoveu melhor concepção do que deveria ser feito pelos desenvolvedores e testadores, deixando o cliente mais confortável na ideia que a equipe realmente havia entendido o que era para ser feito em um sprint. Isso ocasionou mais assertividade nas validações feitas pelo cliente. Com isso, as entregas foram mais frequentes, pois novas versões realmente eram desenvolvidas para atender os valores esperados pelo cliente. Os testes automatizados de aceitação e exploratórios auxiliaram na documentação do processo de desenvolvimento e em uma melhor qualidade das versões entregues.

Entretanto, acredita-se que o momento do Processo B foi significativamente melhor que o momento do Processo A, porém não seria possível aplicar as práticas do Processo B sem antes ter aplicado e a equipe dominado as práticas do Processo A. Conclui-se que houve de fato uma melhoria do processo de software. Isso foi validado estatisticamente nesta pesquisa a medida que o processo, após maduro em práticas de gerência de projetos, incorporou práticas técnicas de desenvolvimento. A observação das métricas de dedicação e velocidade relativa mostraram isso.

É interessante considerar como limitação desta pesquisa a falta de mais resultados entre métricas o que poderia viabilizar uma observação considerando a distribuição  $z$ , o que poderia oferecer um grau maior de confiança. É possível realizar como trabalho futuro a execução de diferentes cenários variando em cada um, práticas ágeis diferentes. Com isso, seria possível através de um projeto  $2^k$ , obter quais práticas ágeis mais influenciam nas métricas coletadas e quais interações entre elas poderiam ser mais incentivadas. Apesar de parecer pouco seis amostras de um momento do processo e nove do segundo momento, cada uma é uma média de dez colaboradores geradas em sprints de quatro ou cinco semanas. Ou seja, esses dados são referentes a quinze meses de execução de projeto. Eles podem servir para a continuação do mesmo, indicando que novas práticas técnicas podem ainda ser inseridas assim como novas métricas também podem ser monitoradas. Também é possível utilizá-los como referências para projetos futuros.

## Referências

- Chaves, N.L.S, Santos, G., Cerdeiral, C., Cabral M.L., Cabral, R., Schots, M., Nunew, E., Rocha, A.R. (2011) “Lições Aprendidas em Implementações de Melhoria de Processos em Organizações com Diferentes Características”. In: Workshop Anual do MPS - WAMPS.
- Fontana, R. M., Meyer Jr, V., Reinehra, S., Malucellia A. (2015). “Progressive Outcomes: A Framework for Maturing in Agile Software Development”. The Journal of Systems and Software pages 88–108.
- Fuggetta, A. and Nitto, E. I. (2014). “Software Process”. In: Future of Software Engineering (FOSE). Pages 1-12. Hyderabad, India — May 31 – June.

- García-Mireles, Gabriel Alberto, *et al.* (2015) "Approaches to promote product quality within software process improvement initiatives: a mapping study." *Journal of Systems and Software* 103. p.150-166.
- Gomes, A. F. (2014) "Agile – Desenvolvimento de Software com Entregas Frequentes e Foco no Valor de Negócio". Casa do Código.
- Jain, R. (1991). "The Art of computer Systems Performance Analysis – Techniques for Experimental Design, Measurement, Simulation and Modeling". Wiley Professional Computing.
- Jalali, S. and Wholin, C. (2012) "Global Software Engineering and Agile Practices: A Systematic Review". *Journal Software Evolution and Process*. Vol.24. Pages 643–659.
- Kalus, G. and Khurmann, M. (2013) "Criteria for Software Process Tailoring: A Systematic Review". In: *International Conference on Software and System Process (ICSSP)*. Pages 171-180.
- Magalhães, C. V. C., Silva, F. Q. B., Santos, R. E. S., Suassuna, M. (2014) "Investigations about replication of empirical studies in software engineering: A systematic mapping study" *Information and Software Technology*.
- Magalhães, R. F., Gonçalves, T. G., Rocha, A. R., Santos, G., Oliveira, K. M. (2015). "Instanciação e Execução das Atividades do Processo de Análise de Desempenho de Processos de Software". In: *XIV Simpósio Brasileiro de Qualidade de Software (SBQS)*.
- Pereira, I. M., Tiago, G. S. C., Pereira, R. R. (2013) "Developing Innovative Software in Brazilian Public Universities: Tailoring Agile Processes to the Reality of Research and Development Laboratories". In: *4th Annual International Conference on Software Engineering & Applications (SEA)*.
- Runeson, P., Host, M., Rainer, A., Regnell B. (2012) "Case Study Research in Software Engineering: Guidelines and Examples". John Wiley & Sons.
- Santana, C., Queiroz, F., Vasconcelos, A., Gusmão, C. (2015) "Software Process Improvement in Agile Software Development: A Systematic Literature Review" In: *41st Euromicro Conference on Software Engineering and Advanced Applications*.
- Svensson, R. B., Gorschek, T., Regnell, B., Torkar, R., Shahrokni, A. and Feldt, R. (2012) "Quality Requirements in Industrial Practice—An Extended Interview Study at Eleven Companies". In: *IEEE Transactions on Software Engineering*, Vol. 38, Nº 4.
- Turk, D., France, R., Rumpe, B. (2005) "Assumptions Underlying Agile Software Development Processes" *Journal of Database Management*, Volume 16, No. 4, pp. 62-87.

- Unterkalmsteiner, M., Gorschek, T., Moinul Islam, A. K. M., Cheng, C. K., Permadi R, B., and Feldt, R. (2012) “Evaluation and Measurement of Software Process Improvement - A Systematic Literature Review”. IEEE Transactions on Software Engineering, Vol. 38, N° 2, March.
- Weber, K. C., Macedo, M. M., de Oliveira, N. H. F., Barroso, E. T., Duarte, V. C. (2015). “Impactos Socio-econômicos no Brasil do Modelo MPS-SW para Melhoria de Processos de Software”. In: XIV Simpósio Brasileiro de Qualidade de Software (SBQS).
- Weber, K. C., Macedo, M. M., Duarte, V. C., Oliveira, N. H. F., Barroso, E. T. (2015). “Pesquisa ‘MPS Cidadão’: Impactos Econômicos e Sociais da Melhoria de Processos de Software no Brasil Usando o Modelo MPS-SW”, Softex.
- Wholin, C., Runeson, O. Höst, M., Ohlsson, M. C., Regnell, B., Wesslén, A. (2012) Experimentation in Software Engineering. Springer. E-Book.