

O Impacto do Uso de Métodos Ágeis no Processo de Ensino-Aprendizagem de Engenharia de Software

Cecília Keiko Adati Tomomitsu
Departamento de TI - Faculdade de Tecnologia de São Paulo
ckt_tomodati@yahoo.com.br

Aristides Novelli Filho
Departamento de TI - Faculdade de Tecnologia de São Paulo
anfmeq@uol.com.br

Abstract: This article highlights the verifications accomplished in the research project on the adoption impact of the agile methods in the software engineering teaching-learning process in the Faculdade de Tecnologia de São Paulo. It presents them the agile and traditional approaches and the how much these approaches are adequate, under apprentice's perspective, in the software process learning. It verifies that the traditional approach it constitutes in a model that subsidizes the adoption of the agile methods, that is a statement of the principles, more than of the formal rigidity. To the final it concludes that the agile approach results in apprentice performance improvement but the base offered by the traditional approach is important.

Resumo: Este artigo destaca as constatações realizadas no projeto de pesquisa sobre o impacto da adoção dos métodos ágeis no processo de ensino-aprendizagem de engenharia de software no âmbito dos cursos de tecnologia em processamento de dados da Faculdade de Tecnologia de São Paulo. Apresenta as abordagens ágil e tradicional (modelo cascata) e o quanto essas abordagens são adequadas, sob a perspectiva do aprendiz, na aprendizagem do processo de software. Constata-se que a abordagem tradicional constitui-se num referencial que subsidia a adoção dos métodos ágeis, que se vale de princípios e valores, mais do que do rigor formal. Ao final conclui-se que a abordagem ágil resulta na melhora do desempenho do aprendiz mas a fundamentação oferecida pela abordagem tradicional é importante.

Palavras-Chaves: Engenharia de Software, Métodos Ágeis, Ensino-Aprendizagem

Introdução

Num contexto caracterizado pela celeridade, o processo de ensino e aprendizagem do ciclo de vida do software, a partir de ciclos encurtados, exige dos aprendizes saberes e habilidades que muitas vezes ainda não estão presentes. Este artigo sintetiza algumas constatações que foram realizadas durante o projeto de pesquisa realizado na Fatec-SP [1] com a finalidade de verificar o impacto da adoção dos métodos ágeis no processo ensino-aprendizagem de engenharia de software no âmbito do curso de tecnologia em

processamento de dados. Buscou-se identificar variáveis que impactam o processo de aprendizagem e que podem corroborar para a assimilação adequada do processo de software, visando o aumento de a sua eficácia.

O projeto de pesquisa foi conduzido por meio da constituição de grupos de trabalho orientados para usar as abordagens tradicional e ágil com o objetivo de desenvolver determinados artefatos de software definidos em quatro estudos de caso, ao longo de um ano. Além do aspecto metodológico, procurou-se observar o impacto do uso de software gerador de código associado às abordagens tradicional e ágil.

Abordagens metodológicas tradicional e ágil

A engenharia de software surgiu num contexto onde a crise de software se apresentava e, decorrente disso, sistemáticas de trabalho mais consistentes e formais, inspiradas na engenharia, estavam sendo concebidas para solucionar problemas de desenvolvimento de sistemas computacionais, que tendem a ser a cada dia maiores e mais complexos e que acompanham a comunidade de desenvolvedores e *stakeholders*¹ de forma crônica [2].

Boehm [3] ao apresentar a evolução da engenharia de software, destaca:

- na década de 50 a engenharia de software tendia a ser parecida com a engenharia de hardware;
- na década de 60 o processo de software era considerado arte;
- na década de 70 o desenvolvimento caracterizou-se pelo processo cascata e pela formalidade;
- na década de 80 o processo de software privilegiou a produtividade e a escalabilidade;
- na década de 90 caracterizou-se pelos processos concorrentes ao invés de apenas processos seqüenciais;
- nos anos 2000 o processo de software orienta-se pela agilidade, os métodos ágeis.

A abordagem tradicional, modelo cascata, formalmente empregado desde a década de 70, e a abordagem ágil, proposta recente de desenvolvimento, constituem o foco deste projeto de pesquisa.

O modelo de ciclo de vida clássico ou cascata, descrito pela primeira vez por Winston W. Royce em 1970 [4] [5], que denominaremos a partir deste ponto como tradicional, caracteriza-se pelo seu caráter preditivo, prescritivo, seqüencial, burocrático, rigoroso, orientado a processos e dados, formal e controlado, que tem o sucesso alcançado desde que esteja em conformidade com o que foi planejado [6] [7].

Essa metodologia de desenvolvimento e manutenção de sistemas orienta o aprendiz na realização de todos os processos e produção dos respectivos artefatos, servindo-lhe como um guia que o dirige até o final do projeto. Essa característica prescritiva constitui-se num roteiro que permite ao aprendiz uma visibilidade de todas as fases do projeto.

Os modelos prescritivos fornecem estabilidade, controle e organização a

¹ O termo “stakeholders” foi criado para designar todas as pessoas ou empresas que, de alguma maneira, são influenciadas pelas ações de uma organização.

uma atividade que pode, se deixada sem controle, tornar-se bastante caótica [6].

O rigor formal, em especial no que se refere à documentação, é uma das características que permeiam essa abordagem. Privilegia-se a elaboração de uma boa documentação de todos os elementos obtidos e produzidos em todas as fases dos projetos, além dos respectivos artefatos. Tudo aquilo que é documentado como pertencente ao escopo do projeto é considerado objeto do desenvolvimento.

Em fevereiro de 2001, um grupo de 17 metodologistas formou a *Agile Software Development Alliance* (www.agilealliance.org), freqüentemente citada apenas como *Agile Alliance*, para enfrentar os diversos desafios encontrados pelos desenvolvedores de software. Este grupo de pessoas definiu um manifesto para encorajar melhores meios de desenvolver software e, com base nesse manifesto, formulou um conjunto de princípios que definem critérios para os processos de desenvolvimento ágil de software [8].

Atualmente, a velocidade tem sido uma das marcas da sociedade moderna. Os produtos têm seu ciclo de vida encurtado. Esse ritmo acelerado da evolução tecnológica, das organizações e do ambiente de negócios de maneira geral [3], tem incentivado a busca de novas abordagens que dão ênfase a agilidade para um melhor posicionamento competitivo. A competição acirrada tem exigido respostas cada dia mais rápidas de todos os competidores. As tecnologias têm tido aumento significativo nas suas capacidades e rapidez. Nesse contexto, onde a velocidade constitui-se num dos ingredientes para o alcance de uma melhor posição competitiva pelas organizações, é essencial que os processos permitam produzir com a necessária celeridade. Assim, o requisito prazo impõe, como condição necessária, a adoção de processos ágeis para a obtenção de produtos em cronogramas cada vez mais apertados.

Os métodos ágeis caracterizam-se pelo seu caráter adaptativo e orientado para pessoas. São várias as metodologias que são classificadas como ágeis, entre elas destacam-se a XP (*Extreme Programming*) e a Scrum, que foram os processos utilizados neste projeto de pesquisa.

A engenharia de software surgiu com o intuito de solucionar a denominada crise de software. Independentemente de ser uma crise ou problema crônico, o fato é que as dificuldades persistem [6] e o desenvolvimento de software constitui-se numa tarefa muitas vezes árdua e que tem levado à muitos insucessos. Esses fatos justificam o grande interesse de todos os *stakeholders*, particularmente a comunidade de desenvolvedores, em buscar abordagens metodológicas que possam ajudar a eliminar ou minimizar significativamente as dificuldades enfrentadas a cada projeto.

Sommerville [9] destaca que as características específicas de determinada abordagem metodológica podem ser de grande valor para a solução de problemas em determinados contextos. Dessa forma tanto a abordagem tradicional quanto a ágil têm o seu valor.

Assim, características como formalidade, ênfase na documentação, cumprimento rigoroso das fases estabelecidas e produtos definidos, características da abordagem tradicional, não concorrem com a abordagem ágil, cujo conjunto de valores e princípios servem para resolver problemas cujas especificidades exigem uma outra abordagem.

Deve ser levado em consideração o fato das abordagens ágeis serem fortemente centradas nas pessoas, pois, como destaca Cockburn [10], fatores como talento, habilidades, conhecimentos, experiências, amizade e comunicação

podem ser aspectos-chave na execução de projetos. Por outro lado, o fato de caracterizarem-se pela adaptabilidade e serem focadas em resultados, tornam-se alternativas que devem ser consideradas na escolha de uma abordagem.

Estudos de Caso: Aplicação dos métodos tradicional e ágil

Os alunos do Curso de Processamento de Dados da FATEC-SP, dos turnos matutino e vespertino, nos anos 2006 e 2007, foram convidados a participar deste projeto de pesquisa. As variáveis observadas consideraram a maturidade acadêmica de cada aluno e, principalmente, a sua percepção dos elementos que facilitam e dificultam os aspectos relacionados aos processos de software estudados e aplicados no desenvolvimento dos sistemas em laboratório.

Os recursos de infra-estrutura que influem na aprendizagem da engenharia de software considerados sob a ótica do aluno foram neutralizados pela adoção do mesmo ambiente para todos os participantes da pesquisa.

As aplicações dos métodos, tradicional e ágil, foram observadas “in loco” e também avaliadas a partir de relatórios de progresso elaborados pelos alunos e professores.

Os requisitos funcionais aplicados aos estudos de caso forneceram subsídios para a obtenção de indicadores relacionados aos processos e aos produtos desenvolvidos. Os indicadores foram úteis para avaliar qual o impacto da adoção de métodos ágeis na aprendizagem da engenharia de software.

Os estudos de caso foram executados por equipes compostas de alunos com experiência acadêmica na abordagem tradicional, uma vez que já estavam cursando a terceira disciplina de análise e projetos do curso, porém, sem experiência anterior em métodos ágeis. Para tanto, antes do início da experiência, foi dada uma visão geral dessa nova abordagem, enfatizando os aspectos relevantes para o propósito do trabalho.

Resultados dos Estudos de Caso

Os objetivos da aplicação dos estudos de caso consistiram em verificar, na prática, os impactos da adoção das abordagens ágil e tradicional na aprendizagem do processo de software, sintetizada nas figuras 1, 2 e 3. O resultado evidencia que os grupos que nortearam a condução dos projetos pela abordagem ágil conseguiram atender as funcionalidades solicitadas (AT) numa proporção significativamente maior; e os grupos que adotaram a abordagem tradicional tiveram maior grau de insucesso.

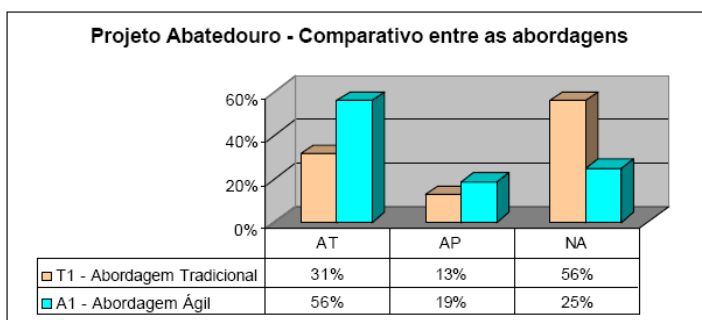


Figura 1 – Comparativo entre as abordagens – Projeto Abatedouro

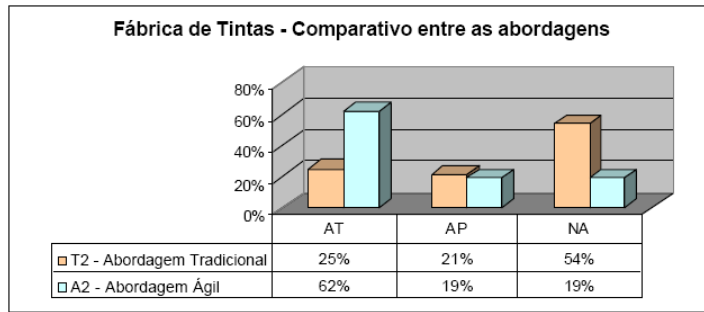


Figura 2 – Comparativo entre as abordagens – Projeto Fábrica de Tintas

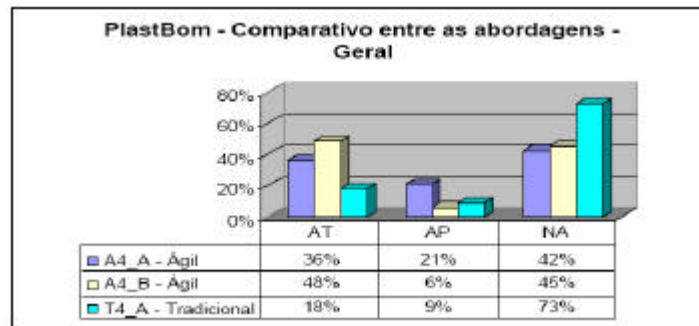


Figura 3 – Comparativo entre as abordagens – Projeto Plastbom

Cabe destacar que foi observado que um grupo que adotou a abordagem ágil (A1_D) e outro que adotou a abordagem tradicional (T1_D) conseguiram alcançar pleno êxito nas tarefas que lhes foram atribuídas conforme figura 4.

		Projeto 1									
		A1_A	T1_A	A1_B	T1_B	A1_C	T1_C	T1_D	A1_D	A1_E	A1_F
Funcionalidades											
Cadastrar componentes		NA	AT	AT	AP	NA	AP	AT	AT	NA	NA
Cadastrar estrutura do produto		NA	NA	AT	NA	NA	NA	AT	AT	AP	NA
Cadastrar planejamento		NA	NA	AT	NA	AT	NA	AT	AT	AP	NA
Calcular necessidades		NA	NA	AP	NA	AT	NA	AT	AT	NA	NA
Legenda		Abordagem Ágil			Delphi		JAVA				
		Abordagem Tradicional			Visual Basic 6.0						
		AT - Atendida									
		AP - Atendida parcialmente									
		NA - Não atendida									

Figura 4 – Projeto Abatedouro – Funcionalidades implementadas

A análise dos integrantes desses grupos evidenciou que os participantes tinham postura diferenciada em relação aos demais, corroborando com Boehm (2002), ou seja, que a presença nas equipes de participantes classificados como “*premium people*” [11] contribui significativamente para o sucesso de projetos de software, independentemente da abordagem empregada.

Acresce-se a esse resultado as observações e respostas de questionários aplicados durante a execução dos estudos de caso [1], que são sintetizadas a seguir:

- os valores e práticas da abordagem ágil contribuíram para a troca de experiências entre os alunos, uma vez que incentivaram a comunicação entre os envolvidos;
- tanto na abordagem ágil quanto na tradicional sobressaíram as competências individuais e a postura adotada frente aos trabalhos a serem executados;
- o pré-conhecimento da abordagem tradicional mostrou-se um facilitador para a execução das atividades baseadas na abordagem ágil;
- a liderança exercida por algum integrante foi fator de sucesso, mesmo não havendo hierarquia formal nas equipes;
- a auto-organização constituiu-se numa capacidade de grupo essencial;
- a abordagem tradicional foi considerada pelos alunos como a melhor abordagem para a aprendizagem inicial de processos de software.

Considerações Finais

O projeto de pesquisa teve como objetivo verificar o impacto do uso de métodos ágeis no processo de ensino e aprendizagem da engenharia de software no contexto do curso de tecnologia de processamento de dados da Fatec-SP [1].

A partir dos resultados, podem ser feitas as seguintes observações:

- os métodos ágeis são atraentes, pois orientam a realização das atividades com menos formalidade, documentação, e mostram-se como uma abordagem que pode contribuir quando se pretende focar o saber-fazer, ou seja, o desenvolvimento das habilidades;
- a abordagem tradicional, ciclo de vida clássico, por ser prescritiva, definida e seqüencial, oferece uma visão de processo mais clara e produtos melhor especificados, facilitando a aprendizagem;
- a realização dos trabalhos a partir da abordagem ágil é facilitada a partir do pré-conhecimento da abordagem tradicional;
- a produtividade a partir dos métodos ágeis é maior do que pela abordagem tradicional;

Diante dos resultados e, mesmo considerando as possíveis influências que as histórias de vida de cada participante tenham sobre esses resultados, é possível afirmar que a adoção dos métodos ágeis resulta em melhora no desempenho do aprendiz. Pode-se afirmar, também, que a adoção de geradores de código, independente da abordagem metodológica, traz benefícios ao aprendizado e ao domínio de processos de desenvolvimento de software.

Vale ressaltar que a abordagem tradicional serviu como contraponto à abordagem ágil, uma vez que os participantes reconheceram a importância do seu pré-conhecimento.

Como constatado ao longo dos resultados apresentados, o fator humano constitui-se num aspecto essencial para o sucesso dos projetos. É importante ressaltar que os avanços tecnológicos afetam desenvolvedores, usuários e processos utilizados no desenvolvimento de sistemas [6], mas as pessoas envolvidas que trazem consigo seus próprios estilos de trabalho, diferenças importantes quanto a habilidades, criatividade, regularidade, consistência e espontaneidade propiciam também, importante influência [12]. Essa constatação

é corroborada por Boehm [11] que enfatiza que a eficácia da adoção das abordagens ágeis deve merecer uma avaliação cuidadosa pelo fato de ter, no componente humano, um ponto importante para o sucesso de sua adoção.

Para concluir, espera-se que os estudos realizados contribuam para a melhoria dos resultados obtidos no ensino de engenharia de software na Fatec-SP e sirvam como incentivo para novos estudos práticos sobre o processo de ensino-aprendizagem desta disciplina que apresenta muitas dificuldades nos cursos de tecnologia da informação.

Referências

- [1] Tomomitsu, Cecília K. A. *Impacto do O Impacto do Uso de Métodos Ágeis no Processo de Ensino-Aprendizagem de Engenharia de Software*. São Paulo, 2008. Dissertação (Mestrado) – Centro Estadual de Educação Tecnológica Paula Souza.
- [2] Pressman, Roger S. *Engenharia de software*. 3ª. edição. São Paulo: Makron Books, 1995.
- [3] Boehm, Barry. A view of 20th and 21st century software engineering. *ICSE 2006*.
- [4] Royce, W. "Managing the Development of Large Software Systems", In: *IEEE WESCON, IEEE Press, pp. 1-9, San Francisco, 1970*.
- [5] Gustafson, David A. *Schaum's Outlines of Theory and Problems of Software Engeneering*. USA: McGraw-Hill, 2002.
- [6] Pressman, Roger S. *Engenharia de software*. 6ª. edição. São Paulo: McGraw-Hill, 2006.
- [7] Paula Filho, Wilson de Pádua. *Engenharia de software*. Rio de Janeiro: LTC – Livros Técnicos e Científicos S.A., 2003.
- [8] [http:// www.agilemanifesto.org/principles.html](http://www.agilemanifesto.org/principles.html), acesso em 10/11/2007
- [9] Sommerville, Ian. *Engenharia de Software*. São Paulo: Addison Wesley, 2003.
- [10] Cockburn, A.; Highsmith, J. Agile Software Development: The People Factor", *Computer*, Nov. 2001, p.131-133.
- [11] Boehm, Barry. Get Ready for Agile Methods, with Care. *IEEE 2002*.
- [12] Cockburn, A. *Agile Software Development: The corporative game*. USA: Addison Wesley, 2006.