

PDMS Ágil: Metodologia Baseada em Scrum para Uma Organização Pública

Jéssica S. Garcia¹, Joyce Aline Marins¹, Vanice Cunha², Aline Wollinger³,
Giulla Araújo³

¹Faculdade de Engenharia – Universidade Federal do Mato Grosso (UFMT)
Cuiabá – MT – Brazil.

²Instituto de Computação – Universidade Federal do Mato Grosso (UFMT)
Cuiabá – MT – Brazil.

³Tribunal de Contas do Estado de Mato Grosso (TCE-MT)
Cuiabá – MT – Brazil.

jessicagarcia132012@hotmail.com, joyce.marins@ufmt.br

vanice@ic.ufmt.br, {alinea, giulla}@tce.mt.gov.br

Abstract. *Evidences in the literature show benefits associated with development with the use of agile methodologies. The Court of Auditors of the State of Mato Grosso has adopted a methodology based on the SCRUM and it is noticeable that the use of the new methodology has promoted an increase in the quality of the systems, reduced implementation time and increased communication and transparency. This article reports the construction and implementation of this methodology, which was built through action research and evaluated from the perspective of management. The challenges reported and lessons learned are of great relevance for organizations that aim to change the traditional development paradigm to agile in their contexts.*

Resumo. *Evidências na literatura mostram benefícios associados ao desenvolvimento com o uso de metodologias ágeis. O Tribunal de Contas do Estado do Mato Grosso tem adotado uma metodologia baseada no SCRUM e é perceptível que o uso da nova metodologia promoveu o aumento da qualidade dos sistemas, diminuiu o tempo de implementação e aumentou a comunicação e a transparência. Este artigo relata o construção e implantação desta metodologia que foi construída por meio de uma pesquisa-ação e avaliada sob a perspectiva da gestão. Os desafios relatados e as lições aprendidas são de grande relevância para organizações que visam mudar o paradigma de desenvolvimento tradicional para ágil em seus contextos.*

1. Introdução

Enquanto metodologias tradicionais são baseadas em fases rigorosas com foco na elaboração de artefatos, metodologias ágeis focam na diminuição de documentos de projeto e de análise e no rápido desenvolvimento de funcionalidades de forma iterativa e flexível. Um dos métodos ágeis amplamente utilizado por equipes de desenvolvimento de software é o Scrum, que é conhecido por sua simplicidade, por exigir pouca documentação de requisitos, pela facilidade de aprendizagem e pela flexibilidade quanto

a regras [Schwaber 2004]. Na literatura são encontrados vários trabalhos que indicam dificuldades de adoção de métodos ágeis principalmente em ambientes do setor público. As principais limitações mencionadas são: estrutura hierárquica, formalidade, limitação de pessoal, necessitando que nesses ambientes ocorra uma mudança de cultura, relações e rotinas [Oliveira and Pedron 2021, Noordeloos et al. 2012, Lappi and Aaltonen 2017, Hidalgo 2019, de Souza Carvalho et al. 2011]. Portanto, a mudança do paradigma de desenvolvimento tradicional para o paradigma ágil precisa ser cuidadosamente planejada.

Este artigo relata a construção e implantação de uma metodologia ágil baseada no SCRUM que foi construída e aplicada no Tribunal de Contas do Estado de Mato Grosso por meio de uma pesquisa-ação. A metodologia construída foi chamada de PDMS ágil (Processo de Desenvolvimento e Manutenção de Software) e possui as características, os artefatos e os papéis baseados no SCRUM. O TCE-MT antes trabalhava com uma metodologia tradicional baseada no RUP (Rational Unified Process). A dificuldade em gerir a grande quantidade de documentação, o atraso nas entregas das funcionalidades e a necessidade de atender com maior rapidez as demandas do setor de Tecnologia da Informação exigiram a evolução da metodologia anteriormente utilizada para um paradigma ágil. Como contribuição teórica, os resultados aumentaram o corpo de conhecimento sobre metodologias de desenvolvimento em engenharia de software. As implicações práticas serão viáveis por meio da adoção da metodologia por outras organizações. De forma geral foi perceptível pela gestão que a adoção de uma metodologia ágil no TCE-MT aumentou a qualidade dos sistemas desenvolvidos e diminuíram o tempo de implementação.

2. Fundamentação Teórica

A engenharia de software tem tradicionalmente usado uma estrutura popular de desenvolvimento de software iterativo e incremental, onde os processos estruturados são caracterizados por um foco em práticas rigorosamente definidas, extensa documentação, planejamento e gerenciamento detalhados, como é o caso do Rational Unified Process (RUP). Estes processos são frequentemente descritos como orientados a planos, já que todo o ciclo de vida de desenvolvimento de software é planejado em uma fase inicial do projeto com base nas unidades de funcionalidade do sistema de software em desenvolvimento, conhecido como 'casos de uso' [Shafiee et al. 2020]. Já os processos de desenvolvimento ágil, como Scrum e eXtreme Programming (XP), enfatizam a importância da comunicação informal eficaz entre os desenvolvedores, a melhoria iterativa de implementações orientadas por cenários de casos de uso e a utilização de pequenas equipes de desenvolvimento coesas em grandes unidades estruturadas, acompanhando o progresso e replanejando em reuniões regulares para desenvolver produtos de forma incremental.

2.1. SCRUM

Existem várias metodologias de design e programação de software em uso hoje. Uma das mais usadas atualmente é o Desenvolvimento Ágil de Software com SCRUM. [Morandini et al. 2021, Shafiee et al. 2020]. SCRUM é um framework para o processo de desenvolvimento ágil de software com foco em práticas de gerenciamento aplicável a projetos muitas vezes complexos que possuem prazos curtos, requisitos específicos e também que exigem melhor comunicação e documentação mais rápida [Morandini et al. 2021, Shafiee et al. 2020, Streule et al. 2016, de Souza Carvalho et al. 2011]. Pesquisas anteriores mostram o porquê do Scrum ter frequentemente substituído o RUP em vários projetos

de desenvolvimento de software [Shafiee et al. 2020] apud [Dybå and Dingsøy 2008]. Os benefícios são percebidos nos ciclos de desenvolvimento mais curtos, maior satisfação do cliente, menores taxas de bugs e adaptação mais rápida aos requisitos de negócios em rápida mudança [Shafiee et al. 2020] apud [Boehm and Turner 2005, Miller and Larson 2005, Moe et al. 2010].

3. Método

De acordo com [Baskerville 1999], a Pesquisa-Ação fundamenta-se na afirmação de que processos sociais complexos podem ser melhor estudados introduzindo-se mudanças e observando-se os efeitos destas no ambiente de aplicação. A flexibilidade da Pesquisa-Ação possibilita criar e adaptar passos para orientar a condução desta abordagem. O conjunto destes passos denomina-se ciclo de vida. O ciclo de vida da pesquisa-ação adotado nesta investigação foi composto pelos seguintes passos:

- 1. Diagnóstico - Envolveu uma investigação na organização participante com o propósito de identificar características culturais e funções da equipe de desenvolvimento e características da metodologia tradicional em vigor que era chamada de PDMS (Processo de Desenvolvimento e Manutenção de Software);
- 2. Planejamento - Foi elaborada a metodologia ágil com artefatos, papéis e atividades de acordo com as especificidades do SCRUM e foi chamada de PDMS ágil.
- 3. Implementação da ação - Gradualmente o PDMS ágil que é baseado no Scrum foi implantado nos projetos de desenvolvimento e as ferramentas necessárias, por exemplo o Redmine, foram configuradas para suportar o acompanhamento dos projetos de forma ágil.
- 4. Avaliação - Os resultados da implantação foram analisados por meio de uma entrevista semi-estruturada com a gestão com o propósito de avaliar a intervenção e os benefícios alcançados.
- 5. Documentação do aprendizado - As lições aprendidas foram documentadas e compartilhadas com a equipe para melhoria e continuidade do uso da metodologia ágil.

Os procedimentos de coleta de dados utilizados durante a Pesquisa-Ação foram entrevistas semi-estruturadas e documentação [Merriam and Tisdell 2009]. Os dados foram tratados por meio de análise temática. A análise temática é um dos procedimentos clássicos para analisar o material de entrevista [Flick 2009].

4. Resultados

4.1. Diagnóstico

O TCE utilizava uma metodologia chamada de PDMS¹ (Processo de Desenvolvimento e Manutenção de Software). Esta metodologia era baseada no RUP e possuía vários artefatos a serem entregues em cada fase de desenvolvimento. Estas fases consistiam em 1. Concepção²: durante este processo, os analistas de sistemas da equipe do projeto deveriam identificar todos os requisitos do produto a partir de reuniões e entrevistas de levantamento junto aos usuários; 2. Elaboração³: esta fase envolvia uma análise detalhada sobre

¹<https://drive.google.com/file/d/1pv6HMG1bhIIzEzdhMgXliNn7dlZqYMQ6/view?usp=sharing>

²<https://shortest.link/1jpv>

³<https://shortest.link/1n5g>

as necessidades e problemas gerais do projeto e a definição de como o sistema deveria ser desenvolvido em termos tecnológicos, considerando os requisitos, limitações e restrições identificados durante a fase de Concepção; 3. Construção ⁴: A fase de Construção era dividida em iterações de acordo com a necessidade identificada. A cada iteração será gerada e testada uma nova release do sistema, contendo os casos de uso implementados até aquele momento; 4. Transição ⁵: consistia em realizar a homologação da versão final do sistema por seus usuários. A homologação do sistema deveria ser realizada considerando os itens descritos no RRH – Relatório de Resultados de Homologação. O sistema homologado junto a seus usuários era disponibilizado para implantação em ambiente de produção.

Vários motivos tornaram clara a necessidade de o TCE mudar da metodologia tradicional, anteriormente utilizada para uma metodologia ágil, como falta de visibilidade dos projetos, dificuldade de gerenciamento do conhecimento, aumento de fluxo de trabalho e por consequência, dificuldade em gerir prazos para entrega das funcionalidades. A gestora responsável pela equipe de TI relatou que o TCE estava com dificuldades de obter informações relacionadas ao andamento dos projetos que estavam sendo realizados. Essas informações não eram armazenadas e não tinham onde e nem como buscá-las. Considerando estes problemas, a quantidade e fluxo de trabalho e a carência de gerir e mensurar as informações de interesse, foi demandada a implementação de uma metodologia ágil para que se conseguisse fazer todo o acompanhamento dos projetos de desenvolvimento. O objetivo dessa mudança era realmente organizar o cenário no qual o TCE se encontrava: excesso de projetos, um profissional envolvido em vários projetos, projetos complexos com prazos para serem entregues e que não se conseguia fazer o gerenciamento de forma efetiva.

4.2. Planejamento

Considerando os problemas identificados na fase de diagnóstico, deu-se início a elaboração do PDMS ágil ⁶ baseado no SCRUM por meio de constantes reuniões com a gestora da equipe e com os líderes de desenvolvimento. Após vários refinamentos foi definido que os artefatos do PDMS ágil são:

1. Documento de Visão do Produto - O Documento de Visão do Produto estabelece as principais funcionalidades e fronteiras do sistema e registra informações que delimitam o projeto de desenvolvimento como um todo. Este artefato tem a função de um contrato entre os diferentes interessados no sistema (TCE, usuário gestor, usuário final etc.), registrando aquilo que será desenvolvido ao longo do projeto. Deve ser formalmente aprovado pelas partes envolvidas.
2. Backlog do Produto - O Backlog do Produto registra os principais requisitos a serem desenvolvidos até a entrega do produto. Este artefato lista os requisitos sem muitos detalhes e em ordem crescente dos mais importantes e a expectativa de conclusão.
3. Backlog da Sprint - O Backlog da Sprint registra o requisito a ser desenvolvido durante a sprint. Este artefato consta com uma breve história de usuário, critérios de aceitação, regras do negócio, protótipos e descrição das telas.

⁴<https://shortest.link/1n5j>

⁵ <https://shortest.link/1jpB>

⁶https://drive.google.com/file/d/10a2-fPcD1TzJ_s1sYKHJz6_bFwY-89Iu/view?usp=sharing

4. Código-Fonte - O código-fonte representa qualquer trecho de código implementado ao longo do desenvolvimento do sistema.

4.3. Implementação da ação

A implantação do PDMS ágil iniciou em novembro de 2020 com um mini-curso sobre os conceitos, as características e os papéis do SCRUM. Após isso, o Redmine foi ajustado para que os projetos executados com base no PDMS ágil fossem cadastrados. A partir daí todos os projetos demandados para a Secretaria de Sistemas de Informação do TCE-MT passaram a ser desenvolvidos com base na nova metodologia. Assim os eventos, atividades e papéis do Scrum foram adotados e seguidos conforme definido no PDMS ágil. As sprints começaram para todas as coordenações ao mesmo tempo, de modo que todas começaram juntas e terminam juntas. No fim do dia é realizada uma retrospectiva geral do que foi realizado e do que precisa ser feito no outro dia. Com a execução das sprints e dos eventos os conceitos do SCRUM tem se consolidado na equipe.

4.4. Avaliação

4.4.1. Avaliação sob a perspectiva da gestão

Foi realizada uma entrevista semi-estruturada com a gestora da equipe de TI do TCE-MT que faz o papel de Scrum Master (a qual foi dada a sigla de GTI para preservar seu anonimato) para avaliar como foi a implementação da ação, ou seja a implantação do PDMS ágil. As questões visaram investigar se houve algum impacto negativo em relação a diminuição de documentação, se houve alguma desvantagem na adoção do PDMS ágil, quais foram os benefícios obtidos com a transição, entre outros aspectos. Cada um deles é mencionado em sequência.

Considerando a grande quantidade de artefatos que eram solicitados no PDMS tradicional e considerando a diminuição drástica de documentação com o PDMS ágil, houve o questionamento se a equipe sentiu falta de algum artefato. A equipe sentiu falta de user stories. Ao ser identificada essa necessidade: “Foi feita a utilização da história de usuários e foi utilizado um livro para treinar as equipes como referência para quem nunca fez a história de usuário [GTI]. Deste modo, User Stories ficou a ser adicionado como artefato na nova versão do PDMS ágil, pois passaram a ser utilizados na escrita de novas funcionalidades, na manutenção corretiva, na manutenção evolutiva e nos casos de teste, tornando-se indispensável para a equipe de TI do TCE-MT.

A gestora não identificou desvantagens na adoção do PDMS ágil. Mas informou que alguns benefícios já foram percebidos no acompanhamento dos projetos: “como são feitos daily [Scrum diária] todos os dias, além de reuniões de planejamento, retrospectiva e revisão, o primeiro benefício foi conseguir acompanhar os projetos de uma forma mais macro, sem ser invasivo, pois a maneira como era feita antes era perguntando para o profissional como estava o andamento do projeto, pois como não tinha registros, não se conseguia filtrar o quanto ainda tinha de projeto para frente [GTI].” Também houve benefícios quanto à gestão de projetos como um todo: “começamos a realizar os planejamentos dos projetos, cronogramas e replanejamento quando necessário, e era uma questão que a equipe sentia muita falta [GTI].” Benefícios foram percebidos também quanto a evolução da equipe que passou a se empenhar mais em fazer uso do Redmine: “eu coloquei de forma muito clara, que o Redmine hoje é o nosso ambiente de gerenciamento

de projeto, se não registraram e se não tiverem o planejamento lá, não tem como ver se precisa de mais um desenvolvedor, não vai saber se vai ter que readequar a entrega porque apareceu algum impedimento [GTI].”

Mais dois benefícios percebidos foram a agilidade na identificação de problemas e a melhoria da comunicação: “conseguimos identificar e tirar esses impedimentos que levava muito tempo para saber, e com as dailies [Scrum diária] começou a dar muita agilidade a algumas coisas que demorava mais e teve muito ganho de comunicação, isso pra mim foi sensacional, conseguimos ter uma comunicação muito melhor [GTI].”

Com relação a definição de indicadores que meçam e comprovem os benefícios obtidos, a gestora informou que ainda não foram implantados. Ainda não há indicadores para mensurar os benefícios obtidos com a mudança de metodologia em termos de tempo, qualidade e custo de desenvolvimento, por exemplo. Entretanto, estes indicadores estão sendo levantados: “Há uma certa dificuldade de extrair dados do Redmine, então está sendo elaborado um relatório no Qlik Sense, para extrair essas informações e conseguir mensurar esses aspectos [GTI].”

5. Conclusão

Métodos ágeis é um tema que tem tomado grande proporção na área de engenharia de software, mas há lacunas sobre o entendimento global da dinâmica e dos conceitos inerentes à elaboração e implementação em equipes de desenvolvimento. Este artigo reportou a elaboração e a implantação de uma metodologia ágil baseada em SCRUM, denominada de PDMS ágil, por meio de uma pesquisa-ação realizada no TCE-MT. O artigo traz grandes inovações em termos de produto, pois o PDMS ágil substitui o PDMS tradicional baseado no RUP que era utilizado há anos pelo TCE-MT. Em termos de processo, a principal inovação é a nova forma de desenvolver software com base nos princípios de SCRUM escritos no PDMS ágil. Isto tem implicado, a partir das percepções da gestora na fase de avaliação, em diminuição de tempo com projetos sendo concluídos mais rapidamente, em valorização de recursos humanos com a padronização dos trabalhos e melhor planejamento dos projetos e em aumento da qualidade dos sistemas desenvolvidos. A perspectiva da gestão quanto à adoção e uso da metodologia serve de orientação para outras organizações que necessitem passar pela experiência de mudar do paradigma de desenvolvimento tradicional para o ágil. O PDMS ágil, artefato gerado pelos pesquisadores em conjunto com equipe de TI, consiste em uma importante contribuição para a literatura e pode ser útil como um ponto de partida para aqueles que quiserem construir sua própria metodologia ágil. Como trabalho futuro é planejado que o PDMS ágil seja avaliado pela equipe de TI e ajustado conforme as lições aprendidas. Métricas para acompanhamento de projetos baseados em SCRUM também estão sendo estudadas para serem adicionadas ao PDMS ágil. Os autores agradecem ao TCE-MT pela abertura e disponibilidade na realização da pesquisa e a unidade financiadora pelo financiamento do projeto.

Referências

- Baskerville, R. L. (1999). Investigating information systems with action research. *Communications of the association for information systems*, 2(1):19.
- Boehm, B. and Turner, R. (2005). Management challenges to implementing agile processes in traditional development organizations. *IEEE software*, 22(5):30–39.

- de Souza Carvalho, W. C., Rosa, P. F., dos Santos Soares, M., da Cunha Jr, M. A. T., and Buiatte, L. C. (2011). A comparative analysis of the agile and traditional software development processes productivity. In *2011 30th International Conference of the Chilean Computer Science Society*, pages 74–82. IEEE.
- Dybå, T. and Dingsøy, T. (2008). Empirical studies of agile software development: A systematic review. *Information and software technology*, 50(9-10):833–859.
- Flick, U. (2009). Introdução à pesquisa qualitativa. artmed. *Porto Alegre*, page 207.
- Hidalgo, E. S. (2019). Adapting the scrum framework for agile project management in science: case study of a distributed research initiative. *Heliyon*, 5(3):e01447.
- Lappi, T. and Aaltonen, K. (2017). Project governance in public sector agile software projects. *International Journal of Managing Projects in Business*.
- Merriam, S. B. and Tisdell, E. (2009). Qualitative research: A guide to design and implementation. san francisco, ca: Lohn wiley & sons.
- Miller, K. W. and Larson, D. K. (2005). Agile software development: human values and culture. *IEEE Technology and Society Magazine*, 24(4):36–42.
- Moe, N. B., Dingsøy, T., and Dybå, T. (2010). A teamwork model for understanding an agile team: A case study of a scrum project. *Information and Software Technology*, 52(5):480–491.
- Morandini, M., Coleti, T. A., Oliveira Jr, E., and Corrêa, P. L. P. (2021). Considerations about the efficiency and sufficiency of the utilization of the scrum methodology: A survey for analyzing results for development teams. *Computer Science Review*, 39:100314.
- Noordeloos, R., Manteli, C., and Van Vliet, H. (2012). From rup to scrum in global software development: A case study. In *2012 IEEE Seventh International Conference on Global Software Engineering*, pages 31–40. IEEE.
- Oliveira, R. L. F. and Pedron, C. D. (2021). Métodos ágeis: Uma revisão sistemática sobre benefícios e limitações. *Brazilian Journal of Development*, 7(1):4520–4534.
- Schwaber, K. (2004). *Agile project management with Scrum*. Microsoft press.
- Shafiee, S., Wautelet, Y., Hvam, L., Sandrin, E., and Forza, C. (2020). Scrum versus rational unified process in facing the main challenges of product configuration systems development. *Journal of Systems and Software*, 170:110732.
- Streule, T., Miserini, N., Bartlomé, O., Klippel, M., and De Soto, B. G. (2016). Implementation of scrum in the construction industry. *Procedia engineering*, 164:269–276.