

Association for Information Systems

AIS Electronic Library (AISeL)

CAPSI 2021 Proceedings

Portugal (CAPSI)

Fall 10-16-2021

Project Management Tools: an analysis of possible agile or traditional influences

Gabriel Soares dos Santos

Instituto Superior de Engenharia de Coimbra, a2019122008@isec.pt

Cristina Chuva Costa

University of Coimbra, chuva@isec.pt

Follow this and additional works at: <https://aisel.aisnet.org/capsi2021>

Recommended Citation

Santos, Gabriel Soares dos and Costa, Cristina Chuva, "Project Management Tools: an analysis of possible agile or traditional influences" (2021). *CAPSI 2021 Proceedings*. 9.

<https://aisel.aisnet.org/capsi2021/9>

This material is brought to you by the Portugal (CAPSI) at AIS Electronic Library (AISeL). It has been accepted for inclusion in CAPSI 2021 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

Ferramentas de Gestão de Projetos: uma análise sobre possíveis influências ágeis ou tradicionais

Project Management Tools: an analysis of possible agile or traditional influences

Gabriel Soares dos Santos, Instituto Superior de Engenharia de Coimbra, Portugal,
a2019122008@isec.pt

Cristina Chuva Costa, Instituto Politécnico de Coimbra, Instituto Superior de Engenharia de Coimbra, Centro de Informática e Sistemas da Universidade da Coimbra, Portugal, chuva@isec.pt

Resumo

As ferramentas de gestão de projeto funcionam como instrumentos importantes na tomada de decisões em organizações. Este estudo pretende desvendar as influências das metodologias ágeis e tradicionais no processo de conceção destas ferramentas. O estudo partiu da identificação de aplicações relevantes, análise das suas características, contraposição destas com as características das metodologias de desenvolvimento e discussão de influências adotadas na conceção das ferramentas - desde sua criação, desenvolvimento, aplicabilidade e temporalidade. Como principais resultados destaca-se que grande parte das ferramentas sofreram influências de metodologias ágeis, especialmente as mais recentes, com características colaborativas e flexíveis, assim como influências híbridas, onde uma mescla de contribuições permite um bom desempenho tanto em projetos com uma componente documental fulcral como noutros mais dinâmicos. Foi possível constatar que ferramentas influenciadas exclusivamente por metodologias tradicionais são pouco usuais. É ainda de referir que o conhecimento da influência de atuação das ferramentas facilita e apoia as tarefas de gestão de projeto.

Palavras-chave: (metodologias; ferramentas; ágeis; tradicionais; gestão de projetos.)

Abstract

Project management tools support decision making in organizations. This study aims to unveil the influences of agile and traditional methodologies in the design of these tools. The investigation started by identifying relevant applications, analyzing their characteristics, comparing them with the ones of the development methodologies, and discussing the influences adopted in the design of the applications. As main results, we highlight that a great part of the tools suffered influences from agile methodologies, especially the most recent ones, with collaborative and flexible characteristics, as well as hybrid influences, where a mix of contributions allows a good performance, both in documental projects and in more dynamic ones. It was possible to see that tools influenced exclusively by traditional methodologies are uncommon. It should also be noted that the knowledge gained about the tools characteristics facilitates and supports project management tasks.

Keywords: (methodologies; tools; agile; traditional; project management)

1. INTRODUÇÃO

Um projeto de desenvolvimento de *software* requer a utilização de práticas de engenharias eficientes, que garantam melhorias na forma como os projetos são planeados, monitorizados e controlados, de forma a criar soluções de qualidade sustentáveis e adaptadas às necessidades das organizações (Programs, 2016; Sutherland & Coplien, 2019; PMI, 2017). A aplicação destas práticas é facilitada pelo recurso a ferramentas adequadas ao suporte do processo de desenvolvimento de *software*, de forma a fomentar o máximo de automatização possível e auxiliar no desenvolvimento e nos processos de tomada de decisão ao longo do ciclo de vida do projeto. De acordo com Deming (1990), as ferramentas funcionam como um conjunto de instrumentos valiosos que suportam a gestão e administração e cuja implantação potencia a melhoria contínua do desempenho organizacional. Estas ferramentas permitem o registo e o acesso a dados sobre os projetos e disponibilizam informação acerca da sua evolução, auxiliando a tomada de decisões rotineiras e críticas (Sutherland & Coplien, 2019).

A escolha das ferramentas de gestão de projeto adequadas a cada realidade organizacional, e às características de cada projeto, potencia a sua utilização e permite otimizar processos, reduzir riscos, melhorar a qualidade dos produtos e aumentar a assertividade e efetividade dos projetos (Abrahamsson 2002; Programs, 2016; Sutherland & Coplien, 2019; PMI, 2017; Milosevic, 2001). É também importante sublinhar que o uso inapropriado dessas ferramentas pode interferir negativamente na evolução do trabalho e comprometer os resultados finais (Cash & Fox, 1992). Estudos como Kloppenborg & Opfer (2002) e Leybourne (2007) destacam que a causa de insucessos na maioria dos projetos refere-se à escolha inapropriada, desenvolvimento e/ou aperfeiçoamento de ferramentas. Winters (2003) reforça que a utilização inapropriada destas ferramentas resultam muitas vezes de falta de formação ou inexperiência dos Gestores de Projetos, má liderança, falhas na definição dos requisitos do cliente e/ou produtos, falhas na definição e gestão das expectativas dos clientes, estimativas deficientes do trabalho necessário e desalinhamento com a equipe do projeto e da organização.

Apesar dos diversos estudos realizados sobre mecanismos de gestão organizacional (Abrahamsson 2002; Cash & Fox 1992; Coombs & Pybus 1998; Pinto & Slevin 1988), existe ainda uma lacuna sobre as influências metodológicas nas ferramentas de gestão. Partindo deste contexto, e tendo por base a motivação à reflexão, compreensão e possibilidade de contribuir para a área, este estudo pretende clarificar as influências das metodologias de desenvolvimento de *software* (metodologias ágeis e tradicionais) no processo de criação e desenvolvimento de ferramentas de gestão.

2. METODOLOGIA

O estudo desenvolveu-se a partir de 4 etapas distintas: (01) identificação das ferramentas; (02) análise das suas características e funcionalidades; (03) contraposição das características das ferramentas com as metodologias de desenvolvimento (ágeis e tradicionais) e (04) discussão de influências adotadas e identificadas.

A identificação das ferramentas (etapa 01) teve como base o portal Capterra (2021)¹ e foi complementada com outros referenciais (Asana 2020; Basecamp 2021; Confluence 2021; Gazzani & Silva, 2019; JIRA Software 2020; Mishra & Mishra, 2013; Monday 2021; MS Project 2020; Slack 2020; Teamwork 2020; Trello 2020; Wrike 2021). Ao aceder ao portal, é possível selecionar o tipo de *software* acerca do qual se pretende aceder aos resultados de avaliações dos seus utilizadores. Para este estudo em particular foram realizadas pesquisas por ferramentas de Gestão de Projetos. Esta consulta foi repetida durante o período de outubro de 2020 a janeiro de 2021, no dia 20 de cada um dos meses. Os critérios de elegibilidade para a seleção das ferramentas foram o fato de serem gratuitas² e com melhor avaliação dos utilizadores, acima de 4 estrelas. É importante salientar que as buscas no portal disponibilizam dados acerca das principais características/recursos das ferramentas, como: gestão de orçamentos, ferramentas de colaboração, visualização de *Gantt*/linha do tempo, gestão de carteira, gestão de recursos, controle de marcos e gestão de ideias. Além disso, o portal permite realizar comparações entre uma ou mais ferramentas a partir destes recursos.

Durante o período observado, verificou-se um total de 287 ferramentas seriadas de acordo com os critérios de elegibilidade. Destas, as 10 com melhor avaliação foram: Slack, Trello, Basecamp, Jira, Asana, Monday, Confluence, Wrike, MS Project e Teamwork, respetivamente. Ao longo dos quatro meses de observação, estas 10 ferramentas mantiveram-se na mesma posição da seriação. Os números de avaliações de utilizadores para cada ferramenta no intervalo discutido são apresentados na Figura 01.

¹ O portal Capterra.pt foi criado em 1999, por Michael Ortener. Atualmente é um dos mais importantes fornecedores de avaliações de utilizadores sobre as diversas ferramentas de *softwares* empresariais. O *website* possui mais de 1 170 000 avaliações publicadas no *site* e milhares acrescentados a cada mês (Capterra, 2020). Para além disso, foi utilizado como fonte de informação em diversos estudos (Wasif, 2019; Möller et al., 2020)

² Período de 7 a 30 dias de utilização sem custos.

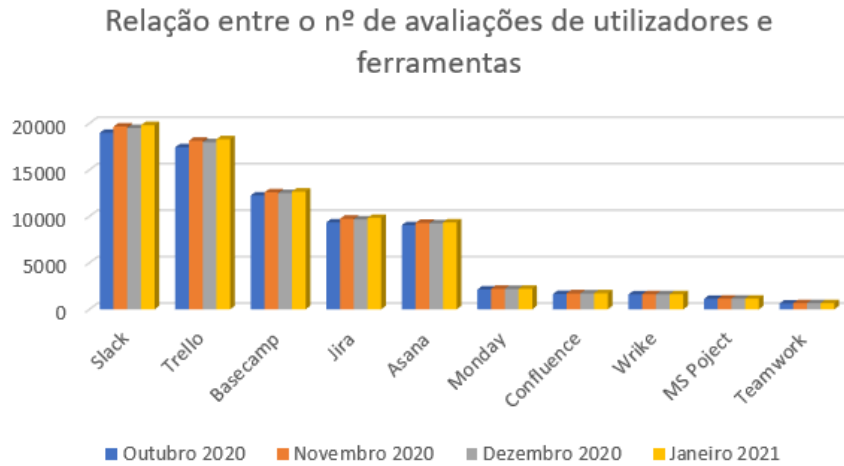


Figura 01 – Relação entre o nº de avaliações de utilizadores e ferramentas (Capterra 2020).

Após o mapeamento mensal das avaliações das ferramentas e calculada a média das avaliações dos utilizadores (Figura 02) foram seleccionadas as ferramentas a serem utilizadas neste estudo.

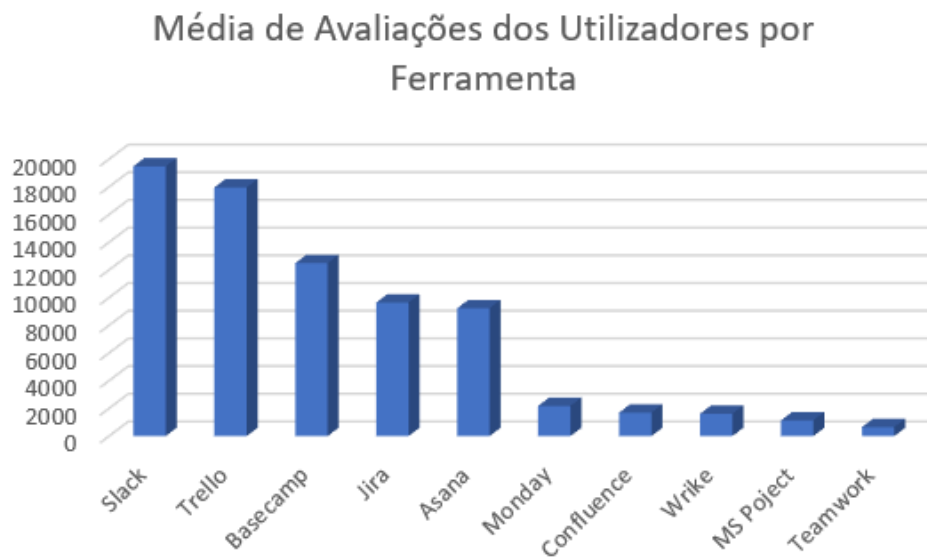


Figura 02 - Média de Avaliação dos Utilizadores por Ferramenta

De seguida, foi realizado um levantamento bibliográfico acerca das ferramentas e na etapa 03, foram intercetadas as suas características com as das metodologias ágeis e tradicionais, considerando aspetos como o nível de flexibilidade dos processos, gestão de custos e de recursos, ferramentas de apoio à comunicação, controle e gestão de tarefas, entregas, cronogramas e acompanhamento de metas.

Para colaborar com o desenvolvimento deste estudo, utilizou-se um projeto fictício padrão, com a finalidade de simular o comportamento e as especificidades de cada uma das ferramentas

conforme o ciclo de vida do projeto. Este projeto contou com 5 elementos na equipe sendo eles: 01 Gerente de Projetos; 03 Programadores; 01 *Quality Assurance (QA)*. O projeto foi configurado com a duração de 3 meses, com 30 tarefas escritas e distribuídas em 10 tarefas para cada um dos programadores, com relações de dependências entre algumas tarefas e os responsáveis por elas. Foram definidas entregas a cada 15 dias a contar da data inicial do projeto e uma entrega final 7 dias antes da data de encerramento do projeto, contando assim com 7 dias para interação com o cliente e acertos finais. No término do oitavo ciclo foram realizadas alterações no âmbito do projeto e acrescentadas 15 novas tarefas, sem alterar a data final de entrega, mas reduzindo as funcionalidades no produto final. Após a finalização de cada uma das tarefas, estas foram enviadas para o QA realizar a validação de cada uma delas e em seguida devolvê-las ao fluxo de desenvolvimento.

3. FERRAMENTAS DE PROJETOS

As ferramentas de projetos possuem inúmeras características e funcionalidades, as quais permitem automatizar a atribuição de tarefas, alocação de recursos, identificação das fases do projeto, definição e controlo de datas e marcos de um projeto, entre outros. A compreensão das suas especificidades, particularidades, processo de desenvolvimento e características é essencial para a gestão de uma organização. O conhecimento concreto destes instrumentos facilita a implementação dos modelos de gestão, assim a sua eficiência. De seguida são descritas resumidamente as ferramentas Slack, Trello, Basecamp, Jira, Asana, Monday, Confluence, Wrike, MS Project e Teamwork.

Slack – Ferramenta de comunicação lançada em 2014, que inclui a possibilidade de criar grupos e canais (públicos e privados). Os grupos funcionam como pequenas salas de conversação entre três ou mais utilizadores, enquanto os canais integram os grupos específicos de cada projeto. Para além disso, permite a troca de arquivos e pesquisas por palavras-chaves e/ou documentos enviados, assim como a integração com diversos outros aplicativos através de APIs, como calendários e *softwares* mais específicos para gestão de tarefas. O Slack pode ser acedido através de dispositivos móveis, navegadores *web* ou *desktop*. É ainda de referir que esta ferramenta possui integração com mais de 150 ferramentas como gestores de tarefas e *e-mails* (Slack, 2020).

Trello – Aplicação *web* de gestão de projetos lançada em 2011, baseada em *Kanban*. Esta aplicação permite anexar arquivos nas tarefas, atribuir tarefas a utilizadores, inserir comentários. Na versão paga, é possível criar gráficos de *Burndown*, *Gantt*, *BurnUp*, entre muitos outros. Esta ferramenta permite a integração com outras aplicações, como *Dashboards* e relatórios (Trello, 2020).

Basecamp – Ferramenta de gestão de projetos lançada em 2004, apresenta na sua versão gratuita várias funcionalidades como, monitorizar atividades, partilhar arquivos, *chat* para comunicação, envio de mensagens e área de agendamento de metas e cronogramas. A versão *trial* é suportada por uma plataforma *web* que pode ser utilizada em sistemas *Android* e *iOS* (Basecamp, 2021).

Jira – Esta ferramenta foi lançada em 2004. É utilizada pois permite: monitorizar tarefas, inserir tempo para cada uma delas, acompanhamento por *Kanban*, trocar informações e arquivos, organizar o fluxo de trabalho com vários relatórios - *Burndown*, *Burnup*, *Velocity Chart*. Os seus *Dashboards* fornecem informações importantes para os gestores, tornando-a relevante no controle de tarefas e tempo. Também permite diversas integrações com serviços externos (Jira, 2020).

Asana – Lançado em 2012, este *software* permite a criação de fluxos de atividades, listas de tarefas, comunicação através de mensagens, trocas de arquivos, implementação dos funis de vendas, recursos humanos e estratégias digitais. Esta ferramenta possui aplicações como *Kanban*, que tornam a gestão mais prática e visual. É uma plataforma *Web*, e além disso, está disponível para dispositivos móveis *Android* e *iOS* (Asana, 2020).

Monday – Ferramenta de gestão de projetos lançada oficialmente com o nome de *Monday.com* em 2017, é uma plataforma *web* e *mobile* (*Android* e *iOS*) com suporte a gestão de tarefas, prazos e metas, partilha de arquivos, quadros estilo *kanban*, gráficos, colaboração de equipe e ainda possibilita uma alta gama de integrações. A ferramenta pode ser personalizada para atender diferentes mercados e projetos. Na sua versão de avaliação é possível manter projetos individuais até 2 utilizadores (Monday, 2021).

Wrike – Ferramenta de colaboração e gestão de projetos lançada em 2007. Na sua versão de avaliação permite o controle de tarefas em tempo real, gestão de metas e cronogramas, diagrama de *Gantt*, partilha de arquivos, integrações com *Dropbox* e *Google Docs*. Esta ferramenta possui versões para *Android* e *iOS* e uma plataforma *Web* (Wrike, 2021).

Confluence – Ferramenta de colaboração e gestão lançada em 2004 pelos mesmos criadores do Jira. É uma poderosa ferramenta para documentação de projetos, organização de arquivos, troca de mensagens e partilha de informação. O Confluence é uma plataforma *Web* e pode ser integrada com várias outras aplicações, entre elas, Jira e Trello (Confluence, 2021).

Microsoft Project - Uma das mais antigas ferramentas de gestão de projeto, lançada em 1984, que contém uma variedade de relatórios de gestão que facilitam a organização de projetos, assim como a possibilidade de criar tarefas, gerir o tempo, custos, conceber gráficos de *Gantt*, criar modelos probabilísticos e diagramas de rede. É uma aplicação *Web* e *desktop* que possibilita a integração com várias aplicações (MS Project, 2020).

Teamwork – Ferramenta de gestão de projetos lançada em 2008 que possibilita criar fluxos de trabalho, manter a comunicação com as equipes de desenvolvimento através da troca de mensagens, realizar a gestão de tarefas, gerir tempo, metas, gráficos de *Gantt*, visualizar tarefas em formato *Kanban* e estabelecer relações entre tarefas. Cabe salientar, que estas funcionalidades se referem à versão de avaliação do *Teamwork* (critério de elegibilidade), pois a ferramenta na sua versão paga apresenta funcionalidades adicionais. Para além disso, pode ser acedida por navegadores *Web* e/ou dispositivos móveis e possibilita a integração com várias ferramentas como CRMs e ferramentas de comunicação (*Teamwork*, 2020).

4. CARACTERÍSTICAS DAS ABORDAGENS ÁGEIS E TRADICIONAIS

Para constatar se as ferramentas de gestão de projeto são influenciadas na sua conceção por abordagens de desenvolvimento, nomeadamente ágeis e tradicionais, o próximo passo deste estudo incidiu no estudo das características dessas abordagens.

As metodologias tradicionais, comumente conhecidas como “pesadas”, são aquelas caracterizadas por uma alta regulamentação e documentação, elevado nível de detalhe no levantamento de requisitos e rígido controle e análise de riscos. Segundo Pressman (2001), as metodologias tradicionais são estruturadas a partir etapas bem definidas e aplicadas, baseiam-se principalmente no planeamento do produto final, das atividades que serão necessárias (programação, design, etc.), definições de prazos e custos/recursos, execução, testes e implantação. Este modelo metodológico requer um processo de desenvolvimento de sistema definido, baseado em uma documentação abrangente, o que torna as atividades mais previsíveis e eficientes. Grande parte dos processos de software são planeados detalhadamente por longos períodos de tempo. Esta metodologia é utilizada em projetos onde os requisitos estão bem definidos desde o início do projeto (Cockburn, 2000).

As abordagens ágeis para o desenvolvimento de *software* evoluíram a partir do início dos anos 90, como parte de uma reação contrária as metodologias tradicionais. Essa nova abordagem passou a ser desenvolvida devido ao aumento de produtividade de diversos projetos de desenvolvimento de *software*, que conduzem os projetos à obtenção do produto final em um prazo mais curto, com menos recursos (humanos, financeiros, etc.).

Inicialmente, as metodologias ágeis eram conhecidas como metodologias leves, somente no ano de 2001, quando membros proeminentes da comunidade profissional tecnológica se reuniram no estado norte-americano de Utah e conjuntamente publicaram o chamado “Manifesto ágil” é que a comunidade passou a adotar o conceito de “metodologias ágeis”, como é conhecido atualmente (Beck et. al, 2001).

Abrahamsson et al. (2002) acreditam que estas metodologias ágeis contemplam o desenvolvimento incremental (pequenos lançamentos de *software*, com ciclos rápidos), cooperativo (o cliente e desenvolvedores trabalham constantemente juntos com comunicação direta e próxima), direto (o método em si é fácil de aprender e modificar, é bem documentado) e adaptativo (capaz de fazer mudanças a qualquer tempo). As metodologias ágeis também contemplam equipes multifuncionais que trabalham simultaneamente em várias áreas, como planejamento, análise de requisitos, design, codificação, testes de unidade e de aceitação e, ainda suportam alterações no âmbito do projeto com ele já em desenvolvimento (Sutherland & Coplien, 2019).

5. ANÁLISE E DISCUSSÃO

Conforme os estudos de Sutherland & Coplien, (2019); Royce, (1970) e Pressman, (2001), as principais características de uma gestão de projetos baseia-se na Gestão de Custos (quanto será gasto no desenvolvimento do projeto), Comunicação (direta e clara), Controle de Marcos (entregas curtas ou por etapas ao cliente), Gestão de Recursos (recursos humanos, operacionais), Gestão de Tarefas (o que deverá ser realizado e quando), Controle mudanças de Âmbito (necessidade de incluir novas tarefas no meio do projeto) e Gestão de Tempo (quanto tempo foi utilizado no desenvolvimento de cada tarefa, por desenvolvedor, ou mesmo o tempo total de duração do projeto). Estas sete características apresentam associações com as diferentes abordagens metodológicas. Destas, quatro apresentam enquadramento Tradicional e Àgil (Controle de Marcos, Gestão de Recursos, Gestão de Tarefas e Gestão de Tempo); duas apresentam enquadramento com maior ênfase na metodologia Àgil (Comunicação e Controle de Mudanças de Âmbito); e uma apresenta enquadramento com maior ênfase na metodologia Tradicional (Gestão de Custos). A atribuição realizada pode ser visualizada na Tabela 01.

Características	Metodologia	
	Ágil	Tradicional
Gestão de Custos		X
Comunicação	X	
Controle de Marcos	X	X
Gestão de Recursos	X	X
Gestão de Tarefas	X	X
Controle mudanças de Âmbito	X	
Gestão de Tempo	X	X

Tabela 01 – Características por Metodologia

A partir do enquadramento das características realizadas e da simulação do projeto fictício, fez-se a compilação das principais características e funcionalidades de cada uma das ferramentas

de gestão, de forma a constatar as influências metodológicas que influenciaram a sua conceção, nomeadamente:

- Característica Gestão de Custos: as aplicações JIRA Software, Asana, Monday, Wrike, MS Project e Teamwork, permitem o acompanhamento em tempo real da evolução do projeto com base em valores por hora de cada membro da equipe.
- Gestão de Entregas: com exceção do Slack, todas as ferramentas possibilitam a definição de *milestones*.
- Comunicação: todas as aplicações permitem trocas de mensagem e troca de arquivos dentro das tarefas, o que caracteriza uma comunicação pontual. Apenas a ferramenta Slack é exclusivamente dedicada à gestão e comunicação de informações de maneira contínua e centralizada.
- Gestão de Marcos: com exceção da ferramenta Slack, todas as outras ferramentas analisadas permitem definir *Sprints* e entregas de maneira flexível (no início ou durante o do projeto).
- Gestão de Recursos: todas as ferramentas permitem incluir membros durante o período de desenvolvimento e gestão, assim como controlar as tarefas de cada um deles, seja na atribuição de tarefas ou por meio de comunicação direta entre os gestores de projeto e a restante equipa.
- Gestão de Tarefas: todas as aplicações possibilitam a criação de tarefas, atribuição destas a membros das equipas e o acompanhamento das tarefas no fluxo do projeto, com exceção do Slack. Na ferramenta Trello, o *Kanban* é a sua principal e mais importante funcionalidade, em contrapartida, nas ferramentas JIRA Software, Teamwork e Asana esta é uma das opções de visualização de tarefas.
- Controle de Mudanças de Âmbito: com exceção das ferramentas Slack e Trello, todas suportam alterações deste nível, seja no início ou durante o projeto. Algumas ferramentas, como MS Project, Jira e Teamwork, incorporaram funcionalidades de adaptação às mudanças de âmbito ao longo do ciclo de vida dos projetos, características inexistentes nas versões iniciais.

Diante deste contexto, é possível destacar que as ferramentas lançadas após o ano de 2007 (Slack, Trello, Asana, Monday, Wrike e Teamwork) apresentam maiores influências ágeis, com maior possibilidade de implementar diferentes ciclos, comunicação entre participantes do projeto, controle de entregas, pelo que se afirmam como ferramentas mais dinâmicas.

Nas ferramentas Microsoft Project, Jira, Confluence, Basecamp e Teamwork, observou-se uma grande evolução quanto aos seus processos. Inicialmente, estas ferramentas apresentavam processos pré-definidos e pouco flexíveis (características tradicionais). Contudo, com o decorrer

dos anos, os processos e utilizadores ganharam maior autonomia, surgindo novas aplicações, como possibilidade de visualização de marcos, controle de tarefas, definições de ciclos e novos gráficos de acompanhamento (características ágeis). É possível perceber que inicialmente estas ferramentas sofreram influências das metodologias tradicionais no seu desenvolvimento. Porém, com o passar dos anos, foram-se adaptando às necessidades das metodologias ágeis, passando a assumir uma configuração com influências híbridas.

Por fim, entre as ferramentas identificadas como híbridas (Microsoft Project, Jira, Confluence, Basecamp e Teamwork), as que foram lançadas em primeiro lugar (entre os anos 1984 e 2004, Jira e Microsoft Project), apresentam maiores influências tradicionais por serem mais robustas, completas, com fluxos definidos e com possibilidade de manter o projeto sempre muito bem documentado. Estes resultados podem ser visualizados na Tabela 02.

Características							
Ferramentas	Gestão de Custos	Comunicação	Controle de Marcos	Gestão de Recursos	Gestão de Tarefas	Controle mudanças de Âmbito	Gestão de Tempo
Slack		A		AT			
Trello			AT	AT	AT		AT
Basecamp		A	AT	AT	AT	A	
Jira	T	A	AT	AT	AT	A	AT
Asana	T	A	AT	AT	AT	A	AT
Monday	T	A	AT	AT	AT	A	AT
Wrike	T	A	AT	AT	AT	A	AT
Confluence		A		AT	AT		
MS Project	T		AT	AT	AT		AT
Teamwork	T	A	AT	AT	AT	A	AT

T - Metod. Tradicional | A - Metod. Ágil

Tabela 02 – Ferramentas por Características

6. CONCLUSÃO

As metodologias de desenvolvimento de *software* têm contribuído de forma expressiva para aumentar o sucesso da gestão de projetos de *software*. De forma equivalente, as ferramentas de apoio à gestão de *software*, também contribuíram para esse sucesso, especialmente, pela sua

capacidade de integrar os *guidelines* de várias metodologias, métodos e técnicas e de os colocar à disposição dos profissionais.

O estudo realizado neste artigo demonstra que a maior parte das ferramentas (sejam elas proprietárias, parcialmente gratuitas, recentes, ou mais antigas) tentam proporcionar aos seus utilizadores um conjunto de funcionalidades que lhes permitem, de uma forma implícita, aplicar várias das orientações das *frameworks* tradicionais e ágeis. Contudo, é notório que a influência ágil se tem intensificado nos últimos anos, o que se poderá justificar com a natureza dinâmica e mais flexível dos projetos atuais (caracterizados por ambientes distribuídos, plataformas distintas, vários *stakeholders* com interesses distintos). Esta tendência é suportada pelo estudo realizado neste artigo, onde é possível perceber a tendência de domínio das *frameworks* ágeis (todas apresentam mais características ágeis do que tradicionais).

Para além disso, é possível constatar a grande competitividade entre aplicações, uma vez que várias partilham funcionalidades idênticas. Porém, tal não implica que o desaparecimento de influências de desenvolvimento tradicional, uma vez que de acordo com as características de um dado projeto, estas poderão ser extremamente úteis.

O desenvolvimento de software é uma indústria em constante evolução, o mesmo acontecendo com o sector do desenvolvimento de aplicações para apoio à gestão de projetos de *software*. Tal pode ser constatado com a capacidade de resposta da indústria à pandemia, onde o trabalho remoto passou de uma opção esporádica a algo permanente e a capacidade das organizações se reinventarem se transformou em algo crítico. Neste ambiente de incerteza, as aplicações de apoio à gestão de projetos foram um ponto de apoio das organizações, colocando as suas funcionalidades ao serviço das necessidades de cada projeto (neste cenário, maioritariamente ágeis, mas não excluindo a contribuições que um desenvolvimento de carácter mais tradicional pode proporcionar).

REFERÊNCIAS

- Abrahamsson, P., Salo, O. & Ronkainen, J. (2002) “Agile software development methods: Review and analysis”, *VTT Publications* 478:1-112.
- Asana (2020). Retrieved from https://asana.com/pt_ (12 de janeiro de 2021).
- Basecamp (2021). Retrieved from <https://basecamp.com/> (15 de janeiro de 2021).
- Beck, K.et.al.: Manifesto for Agile Software Development. Agile Alliance (2001)
- Capterra (2020). Retrieved from <https://www.capterra.pt/> (08 de outubro de 2020).
- Cash, C. & R. Fox (1992), ‘Elements of Successful Project Management’, *Journal of Systems Management*, pp. 10–12.
- Cockburn, A. (2000), “Selecting a project's methodology”, *IEEE Software*, Vol. 17, no. 4: 64-71

- Confluence (2021). Retrieved from <https://www.atlassian.com/software/confluence>
- Coombs, R., McMeekin, A., & Pybus, R. (1998). Toward the development of benchmarking tools for R&D project management. *R&D Management*, 28(3), 175-186. <https://onlinelibrary.wiley.com/doi/pdf/10.1111/1467-9310.00094> (03 de março de 2021).
- Deming (1990), E. *Qualidade: A Revolução da Administração*. São Paulo: Ed. Marques Saraiva.
- Gazzani, M. H., & Silva, K. L. (2019). As tendências da Gestão de Projetos e seus impactos na melhoria da entrega dos produtos e serviços. *Intercursos Revista Científica*, 113–128.
- JIRA Software (2020) Retrieved from <https://www.atlassian.com/software> (22 de janeiro de 2020).
- Kloppenborg, T. J.; Opfer, W. A. (2002). Forty years of project management research: trends, interpretations, and predictions. In: Slevin, D. P.; Cleland, D. I.; Pinto, J. K. *The frontiers of project management research*. Project Management Institute, p. 3-30.
- Leybourne, S. A. (2007). The changing bias of project management research: a consideration of the literatures and an application of extant theory. *Project Management Journal*, v. 38, n. 1, p. 62-73.
- Mishra, A., & Mishra, D. (2013). Software project management tools. *ACM SIGSOFT Software Engineering Notes*, 38(3), 1–4. <https://doi.org/10.1145/2464526.2464537>
- Milosevic, D. & Inman, L. (2001), ‘Impact of Project Management Standardization on Project Effectiveness’, *Engineering Management Journal*, Vol. 13, No. 4, pp. 9–16. <https://doi.org/10.1080/10429247.2001.11415133> (28 de Março de 2021).
- Monday (2021). Retrieved from <https://monday.com/> (22 de Janeiro de 2021).
- Möller, F., Guggenberger, T. M., & Otto, B. (2020). Design principles for route optimization business models: A grounded theory study of user feedback. *Proceedings of the 15th International Conference on Business Information Systems 2020 “Developments, Opportunities and Challenges of Digitization”*, WIRTSCHAFTSINFORMATIK 2020, March 2021. https://doi.org/10.30844/wi_2020_j10
- PMIMS Project (2020). Retrieved from <https://docs.microsoft.com/pt-br/project/> (19 de Fevereiro de 2021).
- Pinto, J. K. & Slevin, D. P. (1988), ‘Critical Success Factors Across the Project Life Cycle’, *Project Management Journal*.
- PMI - PROJECT MANAGEMENT INSTITUTE. *Guia PMBOK®: Um Guia para o Conjunto de Conhecimentos em Gerenciamento de Projetos*, Sexta edição, Pennsylvania: PMI, 2017.
- Pressman, R. *Engenharia de Software*. McGraw-Hill, 2001
- Programs, LLC. Raleigh, North Carolina. Sochova, Z. (2016). *The Great ScrumMaster: #ScrumMasterWay*. Addison-Wesley Professional.
- Royce, W.W. (1970), *Managing the development of large software systems: concepts and techniques*. Proc. IEEE Westcon, Los Angeles, CA.
- Slack (2020). Retrieved from <https://slack.com/intl/pt-pt/> (24 de Fevereiro de 2021).
- Sutherland, J. & Coplien, J.O. (2019). *A Scrum Book: The Spirit of the Game. The Programatic*
- Teamwork (2020). Retrieved from <https://www.teamwork.com/> (19 de Janeiro de 2021).
- Trello (2020). Retrieved from <https://trello.com/> (2 de Fevereiro de 2021).
- Wasif, A. (2019). *Human Resources Recruitment System* (Issue December). Daffodil International University.
- Wrike (2021). Retrieved from <https://basecamp.com/> (13 de Janeiro de 2021).
- Winters, F. (2003). *The Top Ten Reasons Projects Fail - Part 7*. <http://www.projectmanagement.com/articles/187449/The-Top-Ten-Reasons-Projects-Fail--Part-7-> (21 de Maio de 2021).