

Práticas e Ferramentas Utilizadas em Startups de Software Paranaenses: Um Estudo Exploratório

Bruno Henrique Cavalcante¹, Liandra Dos Santos Jesus², Gislaine Camila Lapasini Leal², Renato Balancieri³, Ivaldir De Farias Junior⁴

¹Departamento de Informática – Universidade Estadual de Maringá (UEM)
CEP 87.020-900 – Maringá – PR – Brasil

²Departamento de Engenharia de Produção – Universidade Estadual de Maringá (UEM)
CEP 87.020-900 – Maringá – PR – Brasil

³Universidade Estadual do Paraná (UNESPAR) – Campus Apucarana
CEP 86.813-250 – Apucarana – PR – Brasil

⁴Universidade de Pernambuco (UPE) – Campus Garanhuns
CEP 50.100-010 – Recife – Pernambuco – Brasil

bruno.h.cavalcante@gmail.com, {ra102374,gclleal}@uem.br,
renato.balancieri@unespar.edu.br, ivaldir.junior@upe.br

Abstract. *This survey seeks to identify techniques and tools applied on software development in startups located in the Brazilian state of Paraná. Expecting to be able to contribute to the academic and industrial field, by entering an area that is still relatively unknown and presenting relevant information for them, especially for companies that are just starting out.*

Resumo. *Este estudo exploratório busca identificar as práticas e ferramentas utilizadas no desenvolvimento de software em startups do Paraná. Buscando contribuir tanto para a academia quanto para a indústria, por explorar uma área ainda pouco estudada e apresentar informações importantes para essas startups, principalmente aquelas que estão iniciando.*

1. Introdução

Startups são pequenas organizações que estão buscando um modelo de negócio que se expanda exponencialmente, por meio da oferta de um produto ou serviço [Paternoster *et al.* 2014]. É comum que as mesmas acabem por criar novos mercados, entregando ao público produtos ou serviços que conseqüentemente geram novas necessidades.

Por meio de mapeamentos sistemáticos Paternoster *et al.* (2014) e Klotins *et al.* (2015) apontaram a necessidade de aumentar as pesquisas em relação as práticas de desenvolvimento de *software*, com o intuito de replicá-las e reutiliza-las em outros setores da indústria de tecnologia. Porém, apesar do desenvolvimento de *software* ser o centro das atividades de uma *startup*, o mesmo não possui uma base de conhecimento científico bem definido. Neste contexto, o objetivo deste estudo é apresentar as práticas e as ferramentas adotadas no desenvolvimento de *software* em *startups* paranaenses.

2. Método de Pesquisa

Este trabalho foi conduzido por meio de um *survey* exploratório, um método de pesquisa quantitativo frequentemente utilizado em pesquisas empíricas na Engenharia

de *Software* como forma de obter dados de diversas fontes e explorar áreas pouco estudadas [Wohlin and Aurum 2015].

E a partir de *sites* de associações e agrupamentos de *startups* foi possível obter a lista das empresas participantes da pesquisa. Sendo que o procedimento de coleta consistiu do envio de e-mails para as *startups* encontradas, contendo a carta de apresentação e o *link* de acesso ao utilizando a ferramenta de Formulários do Google¹. Na qual, o mesmo ficou disponível por um período de 120 dias.

2. Resultados e Discussões

Os dados considerados para essa pesquisa foram 28 registros. Após a exclusão dos *outliers* por respostas incompletas, o total da amostra foi reduzido para 26.

Em relação aos 26 respondentes da pesquisa, 65% possuem mais do que 5 anos de experiência, indicando que a grande maioria dos envolvidos nas *startups* possuem uma quantidade considerável de experiência com desenvolvimento de *software*. Do restante, 15% possuem entre 3 e 5 anos, 12% possuem entre 1 e 3 anos e 8% possuem menos do que 1 ano de experiência. Sendo que 42% é especialista, 30% são graduando ou graduado, 12% mestrando, 8% Mestre, 8% doutorando ou Doutor. Isto demonstra que boa parte das pessoas que estão envolvidas com *startups* são recém-formadas ou acabaram de terminar uma pós-graduação, ou então são pessoas que decidiram não investir em uma carreira acadêmica além da graduação e da especialização. Porém, apenas 4% dos respondentes têm mais do que 5 anos de experiência com *startups*, 31% entre 3 e 5 anos, 27% entre 1 e 3 anos e 38% menos de um ano.

2.1 Caracterização das *Startups*

A Figura 1 apresenta a distribuição das *startups* de acordo com o seu ano de fundação. E por meio da linha de tendência é possível verificar o crescimento exponencial do número de *startups* fundadas entre 2010 e 2017. É importante salientar que o questionário foi disponibilizado em meados de julho de 2017, ficando disponível até meados de outubro de 2017, logo, é possível observar que a quantidade de *startups* fundadas no ano 2017 está incompleta.

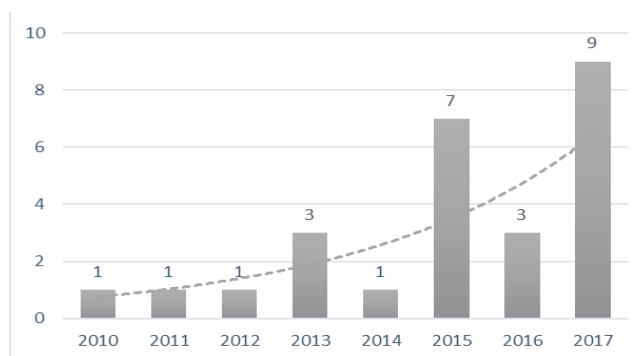


Figura 1. Ano de fundação das *startups*.

A classificação do nível de maturidade das *startups* participantes desta pesquisa, foi realizada segundo Crowne (2002), que descreve a maturidade das *startups* em três

¹ <https://goo.gl/forms/QqksndXyuNCSRPFw1>

fases: a inicial, de estabilização e de crescimento. Assim, foi identificado que 46% encontram-se em fase inicial, e que de acordo com mesmo autor estão entre a concepção do produto e a realização da primeira venda. 19% em fase de estabilização, ou seja, entre o momento em que o primeiro consumidor recebe o produto até o momento em que o produto está estável o suficiente para ser vendido a um novo consumidor sem causar sobrecarga no desenvolvimento. E 34% em fase de crescimento, ou seja, entre o fim da fase anterior e acaba quando as taxas de crescimento e fatia de mercado estão estáveis e todos os processos necessários para o desenvolvimento e venda do produto estão estabilizados [Duc *et al.* 2016].

Os dados permitem concluir que quase metade das *startups* estão em fase inicial, a minoria encontra-se na fase de estabilização, e o restante das *startups* estão na fase de crescimento sendo capazes de entregar seus produtos para mais de um cliente e em breve irão se estabilizar, tornando-se organizações maduras. Entretanto, chegar a esta fase de maturidade é um grande desafio para essas organizações, visto que a maioria destas não conseguem atingir dois anos de vida.

Independentemente do nível de maturidade da *startup*, os dados demonstram que o investimento externo não é um dos principais fatores para o sucesso. Porém, a participação das mesmas em grupo ou rede de colaboração é alta nas fases inicial e de estabilização. Na fase de estabilização o número de *startups* que fazem parte de um grupo ou rede de colaboração é maior, mas com uma diferença de apenas 4% das que não fazem parte. E segundo Santos (2016) a participação traz benefícios como, maior facilidade de comunicação entre as organizações, além de facilitar a troca de conhecimento etc., o que não faz sentido neste trabalho devido que 75% das empresas que estão na fase crescimento não fazem parte de alguma rede ou grupo.

Obviamente que somente com esses dados não é possível chegar a nenhuma conclusão, mas é possível gerar perguntas a serem respondidas em uma pesquisa mais direcionada, para compreender esse comportamento e identificar se as *startups* que chegam a essa fase não utilizam-se de parcerias ou deixam de utilizá-las; e se não utilizam-se de investidor pelo fator de não conseguirem um ou não necessitarem.

2.2 Caracterização do Desenvolvimento de Software nas *Startups*

O estudo analisou a utilização das metodologias ágeis e tradicionais, práticas e técnicas de desenvolvimento, suítes/plataformas de desenvolvimento, ferramentas de controle de versão e configuração, de gerenciamento de projeto e de gestão do conhecimento.

Em relação as metodologias ágeis, 69% das *startups* indicaram utilizá-las, e 31% não, os dados demonstram uma maior tendência de adoção aos métodos ágeis, dividido entre (*Scrum*, *Lean/Lean Startup* e *Test Driven Development* - TDD). Além disso, 1 *startup* confundiu metodologia e técnica, e indicando o *Kanban* como metodologia. Estes métodos permitem que o produto seja desenvolvido em pequenas iterações e entregas frequentes, possibilitando que sejam obtidos *feedbacks* constantes dos usuários. Isso ajuda a visualizar se o desenvolvimento do produto está indo na direção correta, permitindo que sejam realizados ajustes quando este não é o caso.

Observando os dados da Tabela 1, que apresenta um painel das práticas adotadas pelas *startups* paranaenses de *software*. Assim, na fase inicial, tem-se que *Backlog* é a prática mais utilizada seguida de padrões de código, enquanto o uso de *frameworks*

destaca-se na fase de estabilização, seguido do *Kanban*, e por fim na fase de crescimento a capacitação dos membros da equipe, o MVP e o uso de *frameworks* são as práticas mais utilizadas.

Tabela 1. Painel de práticas.

| Práticas | Fase Inicial (12) | Fase de Estabilização (5) | Fase de crescimento (9) |
|--|-------------------|---------------------------|-------------------------|
| <i>Backlog</i> | 10 (83,33%) | 1 (20,00%) | 6 (66,67%) |
| <i>Bug-tracking</i> | 1 (8,33%) | 1 (20,00%) | 3 (33,33%) |
| Capacitação dos membros da equipe | 5 (41,67%) | 1 (20,00%) | 8 (88,88%) |
| Cliente no local de desenvolvimento | 1 (8,33%) | 1 (20,00%) | 2 (22,22%) |
| Code review | 7 (58,33%) | 1 (20,00%) | 2 (22,22%) |
| Empoderamento da equipe de desenvolvimento | 2 (16,67%) | 2 (40,00%) | 5 (55,56%) |
| Entregas frequentes/contínuas | 6 (50,00%) | 2 (40,00%) | 7 (77,78%) |
| Gerenciamento de requisitos | 5 (41,67%) | 1 (20,00%) | 4 (44,44%) |
| Integração contínua | 5 (41,67%) | 2 (40,00%) | 4 (44,44%) |
| <i>Kanban</i> | 4 (33,33%) | 3 (60,00%) | 4 (44,44%) |
| MVP (<i>Minimum Viable Product</i>) | 7 (58,33%) | 1 (20,00%) | 8 (88,88%) |
| Padrões de código | 8 (66,67%) | 2 (40,00%) | 7 (77,78%) |
| <i>Planning Game</i> | 2 (16,67%) | 1 (20,00%) | 3 (33,33%) |
| Programação em pares | 1 (8,33%) | 1 (20,00%) | 2 (22,22%) |
| Prototipagem evolucionária | 3 (25,00%) | 0 (0,00%) | 3 (33,33%) |
| Releases curtos | 5 (41,67%) | 0 (0,00%) | 4 (44,44%) |
| Reuso de Software | 0 (0,00%) | 1 (20,00%) | 7 (77,78%) |
| Uso de <i>frameworks</i> | 7 (58,33%) | 4 (80,00%) | 8 (88,88%) |
| Uso de indicadores de desempenho | 1 (8,33%) | 0 (0,00%) | 6 (66,67%) |
| Uso de logs e estatísticas | 1 (8,33%) | 0 (0,00%) | 7 (77,78%) |
| Uso de soluções COTS ou <i>open source</i> | 1 (8,33%) | 2 (40,00%) | 5 (55,56%) |

O uso de *frameworks* agiliza o processo de desenvolvimento, por retirar a necessidade de implementar algo que já está pronto [Giardino *et al.* 2014]. Outra prática popular é o uso de *backlog*, sendo uma das principais práticas da metodologia *Scrum*, a metodologia mais utilizada entre os respondentes. O mesmo permite que a equipe de desenvolvimento tenha um registro sempre atualizado das tarefas que ainda devem ser realizadas.

O MVP cumpre um papel importante nos processos de uma *startup*, servindo como um artefato de *design* e como artefato recusável, além de ser o ponto chave da metodologia *Lean/Lean Startup* [Ries 2011], permitindo testar o produto com o menor investimento possível e assim, acelerar o seu lançamento. O *Kanban* é uma técnica proporciona velocidade na produção e um ciclo rápido e contínuo de *feedback* com o cliente.

A Tabela 2 apresenta um painel das ferramentas de gerenciamento de projeto, controle de versão e configuração, suítes/plataformas de desenvolvimento e de gestão de conhecimento utilizadas pelas *startups* participantes. Contudo, em relação as suítes/plataformas de desenvolvimento, é difícil argumentar quanto a sua utilização, visto que a escolha das mesmas geralmente depende de fatores particulares da

organização, de desenvolvimento ou do desenvolvedor, restrições de orçamento e público alvo também podem influenciar na decisão.

Em termos de ferramentas de gerenciamento de projeto, na fase inicial e de crescimento o *Trello* aparece como a mais popular provavelmente pelo fato de o mesmo ser organizado por meio de cartões de transitam entre painéis, além de ser uma ferramenta gratuita no plano mais básico.

Tabela 2. Apresentação das ferramentas em relação ao nível de maturidade.

| Nível de Maturidade | Número de startups | Ferramentas | | | |
|------------------------------|--------------------|--|---|--|---|
| | | Gerenciamento de Projeto | Controle de versão e configuração | Suítes/plataformas de desenvolvimento | Gestão do conhecimento |
| Fase Inicial | 12 (46%) | Trello (7) Redmine (2) Atlassian Jira (1) Asana (1) Basecamp (1) Taiga (1) Nenhuma (1) | Git (1) Github (7) Gitlab (4) Bitbucket (2) | Visual Studio (6) Eclipse (5) NetBeans (2) Atom (3) XCode (1) RadStudio Deplhi (1) | Freshdesk (1) Atlassian Confluence (1) Open KM (2) Collectiva Knowledge (2) Slack (1) Drive (1) Nenhuma (6) Google Hangouts (1) |
| Fase de estabilização | 5 (19%) | Trello (1) Redmine (1) IBM Jazz (1) Asana (1) Nenhuma (1) | Git (1) Github (2) Gitlab (1) Bitbucket (2) Subversion (2) | Visual Studio (2) Eclipse (1) NetBeans (2) JetBrains (2) XCode (1) | Freshdesk (1) Atlassian Confluence (1) World (1) Drive (1) Nenhuma (1) |
| Fase de crescimento | 9 (35%) | Trello (8) Redmine (2) Atlassian Jira (3) Google Docs (1) | Git (3) Gitlab (1) Bitbucket (3) Subversion (4) Nenhuma (1) | Visual Studio (2) Deplhi (1) Atom (2) Netbeans (2) VIM (2) JetBrains (2) PyCharm (1) PHPStorm (1) | Atlassian Confluence (3) Jira (2) Collectiva Knowledge (1) zendesk (1) Vtiger (1) Drive (1) Nenhuma (2) |

As ferramentas de controle de versão e configuração facilitam uma análise do histórico de mudanças são, portanto, poderosas ferramentas de *backup*. Além disto, por meio das versões é possível verificar quais trechos de código foram modificados, quem fez as mudanças e também métricas como, número de mudanças dentro de um período de tempo, número de erros e indicar inanição. Assim, na fase inicial e de estabilização, o *Github* foi a ferramenta mais utilizada nesse contexto, já na fase de crescimento foi o *Subversion*.

Por fim, as ferramentas de gestão do conhecimento mais utilizadas na fase inicial foram *Open KM* e *Collectiva Knowledge* e na fase de crescimento foi *Atlassian Confluence*. Já na fase de estabilização, quatro ferramentas tiveram o mesmo número indicações (*Freshdesk*, *Atlassian Confluence*, *World* e *Drive*). Sendo que, uma das

startups indicou que está estudando a respeito de qual ferramenta utilizar e 8 não utilizavam nenhuma ferramenta nesse contexto.

No caso da escolha de qual ferramenta utilizar, ocorre restrições ligadas ao perfil da organização e de seus desenvolvedores. A condução de concursos de inovação tem potencial para trazer ideias de fora da organização ao observar como outras *startups* gerenciam o seu conhecimento de *software*, a partir disto é possível traçar uma estratégia de gerenciamento mais adequada.

3. Considerações Finais

Este estudo permitiu verificar que nos últimos anos o número de *startups* fundadas no Paraná vem crescendo de forma acelerada, e também é possível verificar que o fato de a *startup* possuir investidor externo ou fazer parte de um grupo ou rede de colaboração tem pouca relação com o nível de maturidade da mesma.

No que se refere as práticas relacionadas ao desenvolvimento de *software*, os resultados demonstram que a maioria das *startups* paranaenses utilizam metodologias ágeis, com destaque para o *Scrum* e o *Lean Startup*. Ainda assim, vale notar que, por razões desconhecidas, algumas utilizam metodologias tradicionais, sendo essa uma oportunidade de pesquisa, uma vez que *startups* estão fortemente ligadas ao uso de métodos ágeis e, portanto, seria interessante entender as motivações ligadas a escolha oposta. Adotar noções e práticas mais abertas poderiam auxiliá-las com os desafios provenientes da limitação de recursos e competências.

Estes resultados são importantes pois permitem monitorar o estado de desenvolvimento das *startups* paranaenses e possibilitam que novos estudos sejam conduzidos com base nos dados adquiridos, permitindo que novas hipóteses sejam formuladas a partir do que foi encontrado. Do ponto de vista da indústria, estes resultados servem como um ponto inicial para as *startups* que estão iniciando suas operações e ainda não possuem metodologias definidas, ferramentas, técnicas ou práticas que irão utilizar em seu processo de desenvolvimento.

References

- Crowne, M. (2002), Why software product startups fail and what to do about it? Evolution of software product development in startup companies. In Engineering Management Conference, 2002. IEMC'02. 2002 IEEE International, 1, pp.338-343.
- Duc, A. N., Shah, S. M. A. and Abrahamsson, P. (2016), Towards an early stage software startups evolution model. In 42th Euromicro Conference on Software Engineering and Advanced Applications (SEAA), 2016, pp. 120–127.
- Giardino, C., Unterkalmsteiner, M., Paternoster, N., Gorschek, T. and Abrahamsson, P. (2014), What do we know about software development in startups? IEEE software, 31, 5, pp. 28–32.
- Klotins, E., Unterkalmsteiner, M. and Gorschek, T. (2015), Software engineering knowledge areas in startup companies: a mapping study. In International Conference of Software Business, pp. 245–257.
- Paternoster, N., Giardino, C., Unterkalmsteiner, M., Gorschek, T. and Abrahamsson, P. (2014), Software development in startup companies: A systematic mapping

study. The 29th International Conference on Software Engineering and Knowledge Engineering, 56, 10, pp. 1200–1218.

Ries, E. (2011), *The Lean Startup*. New York: Crown Business.

Santos, M. C. F. R. (2016), *O ecossistema de startups de software da cidade de São Paulo*. PhD thesis, Universidade de São Paulo.

Wohlin, C. and Aurum, A. (2015) Towards a decision-making structure for selecting a research design in empirical software engineering. *Empirical Software Engineering*, 20, 6, pp. 1427–1455.