

UNIVERSIDADE FEDERAL DE GOIÁS
INSTITUTO DE INFORMÁTICA

RICARDO FIGUEIREDO CALDEIRA

**A adoção do MVP como ferramenta de
validação de hipóteses de mercado para
startups enxutas**

Goiânia
2013

UNIVERSIDADE FEDERAL DE GOIÁS
INSTITUTO DE INFORMÁTICA

**AUTORIZAÇÃO PARA PUBLICAÇÃO DE TRABALHO DE
CONCLUSÃO DE CURSO EM FORMATO ELETRÔNICO**

Na qualidade de titular dos direitos de autor, **AUTORIZO** o Instituto de Informática da Universidade Federal de Goiás – UFG a reproduzir, inclusive em outro formato ou mídia e através de armazenamento permanente ou temporário, bem como a publicar na rede mundial de computadores (*Internet*) e na biblioteca virtual da UFG, entendendo-se os termos “reproduzir” e “publicar” conforme definições dos incisos VI e I, respectivamente, do artigo 5º da Lei nº 9610/98 de 10/02/1998, a obra abaixo especificada, sem que me seja devido pagamento a título de direitos autorais, desde que a reprodução e/ou publicação tenham a finalidade exclusiva de uso por quem a consulta, e a título de divulgação da produção acadêmica gerada pela Universidade, a partir desta data.

Título: A adoção do MVP como ferramenta de validação de hipóteses de mercado para startups enxutas

Autor(a): Ricardo Figueiredo Caldeira

Goiânia, 01 de Agosto de 2013.

Ricardo Figueiredo Caldeira – Autor

Marcos Inácio Severo de Almeida – Orientador

Leandro Luís Galdino de Oliveira – Co-Orientador

RICARDO FIGUEIREDO CALDEIRA

A adoção do MVP como ferramenta de validação de hipóteses de mercado para startups enxutas

Trabalho de Conclusão apresentado à Coordenação do Curso de Sistemas de Informação do Instituto de Informática da Universidade Federal de Goiás, como requisito parcial para obtenção do título de Bacharel em Sistemas de Informação.

Área de concentração: Sistemas de Informação.

Orientador: Prof. Marcos Inácio Severo de Almeida

Co-Orientador: Prof. Leandro Luís Galdino de Oliveira

Goiânia
2013

RICARDO FIGUEIREDO CALDEIRA

A adoção do MVP como ferramenta de validação de hipóteses de mercado para startups enxutas

Trabalho de Conclusão apresentado à Coordenação do Curso de Sistemas de Informação do Instituto de Informática da Universidade Federal de Goiás como requisito parcial para obtenção do título de Bacharel em Sistemas de Informação, aprovada em 01 de Agosto de 2013, pela Banca Examinadora constituída pelos professores:

Prof. Marcos Inácio Severo de Almeida
Instituto de Informática – UFG
Presidente da Banca

Prof. Leandro Luís Galdino de Oliveira
Instituto de Informática – UFG

Prof. Celso Gonçalves Camilo Junior
Instituto de Informática – UFG

Todos os direitos reservados. É proibida a reprodução total ou parcial do trabalho sem autorização da universidade, do autor e do orientador(a).

Ricardo Figueiredo Caldeira

Graduando em Sistemas de Informação na UFG - Universidade Federal de Goiás. Durante sua graduação, foi monitor das disciplinas Redes de Computadores e Programação Web no Instituto de Informática da UFG.

<Dedicatória do trabalho a alguma pessoa, entidade, etc.>

Agradecimentos

Primeiramente agradeço ao meu Deus e Pai por todas as coisas boas que vivi, por que sei que o bem apenas dele é que veio. Minha felicidade eu só devo ao Pai!

Agradeço ao orientador prof. Marcos Inácio Severo de Almeida pela paciência, confiança e pela audácia de enxergar que pesquisadores de departamentos tão diferentes poderiam trabalhar em conjunto, seu apoio e ensinamentos proporcionaram a realização deste trabalho.

Agradeço em especial ao meu co-orientador prof. Leandro Luís Galdino de Oliveira, que considero meu pai dentro da academia, por acreditar em meu potencial logo no primeiro mês de minha graduação e guiar meus passos até aqui.

Agradeço aos meus pais, Efraim Caldeira e Patrícia Caldeira, pelo carinho, encorajamento e por não medirem esforços para que eu pudesse chegar até aqui. Também ao meu irmão Marcelo, pelo companheirismo mesmo estando em outra cidade. Obrigado por acreditarem em mim e serem exemplos em minha vida.

Agradeço à minha namorada, Mariana Atti, que acompanhou o início deste projeto de perto e acreditou que eu seria capaz de concluí-lo, obrigado por me auxiliar, confortar e motivar nas horas mais difíceis com seu amor e ternura. Agradeço também à sua família, que tanto admiro.

Agradeço à toda minha família e amigos.

Agradeço aos amigos que fiz dentro do Curso de Sistemas de Informação e que vou levar sempre comigo.

Considero este trabalho resultado de uma jornada que não começou na Universidade Federal de Goiás, agradeço a todos que de alguma forma colaboraram para a conclusão de mais uma etapa em minha vida.

<Epígrafe é uma citação relacionada com o tópico do texto>

**<Nome do autor da citação>,
<Título da referência à qual a citação pertence>.**

Resumo

Caldeira, Ricardo Figueiredo. **A adoção do MVP como ferramenta de validação de hipóteses de mercado para startups enxutas**. Goiânia, 2013. 43p. Relatório de Graduação. Instituto de Informática, Universidade Federal de Goiás.

Startups são concebidas com o objetivo de desenvolver um modelo de negócio repetível e escalável, entretanto os empreendedores devem ter em mente que a fase inicial de uma *startup* é marcada por um cenário de extrema incerteza. É consenso que estamos vivendo um renascimento do empreendedorismo mundial, mas a carência de um paradigma gerencial que permita a rápida validação do feedback de demandas do usuário faz com que oportunidades e iniciativas de inovação corram riscos altos. O que torna o fracasso especialmente doloroso não é apenas o dano econômico causado a funcionários, empresas e investidores; eles também são um desperdício gigantesco dos recursos mais preciosos da nossa civilização: o tempo, a paixão e a habilidade das pessoas. O movimento da *Startup Enxuta* dedica-se à impedir que o insucesso frustre novos empreendedores, para tal o produto mínimo viável (MVP) é apresentado como ferramenta de auxílio na rápida validação de hipóteses de mercado. Este trabalho propõe além de um estudo sobre o perfil de 100 *startups* goianas focado no uso do MVP, um plugin para desenvolvedores que trabalham com *Ruby on Rails* permitindo a construção de MVP's de maneira mais rápida a partir de um ambiente pré-configurado com o foco na entrega contínua.

Palavras-chave

Startup Enxuta, MVP, Entrega Contínua, Ruby on Rails.

Abstract

Caldeira, Ricardo Figueiredo. **The MVP's adoption as a tool for validating market assumptions for lean startups**. Goiânia, 2013. 43p. Relatório de Graduação. Instituto de Informática, Universidade Federal de Goiás.

Startups are designed with the goal of developing a business model repeatable and scalable, but entrepreneurs should keep in mind that the initial phase of a startup is marked by a scenario of extreme uncertainty. The consensus is that we are experiencing a revival of global entrepreneurship, but the lack of a management paradigm that allows rapid validation feedback of user demands makes opportunities and innovation initiatives run high risks. What makes it especially painful failure is not only the economic damage caused to employees, companies and investors, they are also a gigantic waste of the most precious resources of our civilization: the time, passion and people skills. The Lean Startup movement is dedicated to prevent failure frustrates new entrepreneurs, for such minimum viable product (MVP) is presented as tool to aid in the rapid validation of market assumptions. This paper proposes addition of a study on the profile of 100 startups from Goiânia focused on the use of the MVP, a plugin for developers working with Ruby on Rails allowing the construction of MVP's more quickly from an environment preconfigured with the focus continuous delivery.

Keywords

Lean Startup, MVP, Continuous Delivery, Ruby on Rails

Sumário

Lista de Figuras	11
Lista de Tabelas	12
1 Introdução	13
1.1 Contexto	13
1.2 Motivação	16
1.3 Objetivos	17
1.3.1 Objetivo Geral	17
1.3.2 Objetivos Específicos	17
1.4 Metodologia	17
1.5 Organização do Trabalho	17
2 Fundamentação Teórica	18
2.1 Empreendedorismo	18
2.2 Startup	19
2.3 Manufatura Enxuta	20
2.4 O Lean Startup	22
2.5 Customer Development	23
2.6 Desenvolvimento ágil	24
2.7 Plataformas tecnológicas como <i>commodities</i>	24
2.8 MVP: <i>Minimum Viable Product</i>	25
2.9 <i>Business Model Canvas</i>	25
2.10 <i>Lean Canvas</i>	26
2.11 <i>Feedback</i>	28
2.11.1 Ciclo de Feedback	28
2.12 <i>Entrega Contínua</i>	30
3 Perpetuus, uma <i>Gem</i> para construir MVP's	31
3.1 O conceito de Gem	31
3.2 Descrição da Perpetuus Gem	31
3.3 Requisitos de Sistema	31
3.4 Tecnologias Empregadas	32
3.4.1 Ruby	32
3.4.2 Ruby on Rails	32
3.4.3RSpec	32
3.4.4 GitHub	33
3.4.5 Heroku	33
3.4.6 Travis	33

3.5	Funcionamento da GEM	33
3.5.1	O comando <i>perpetuus setup</i>	33
3.5.2	O comando <i>perpetuus deploy</i>	36
4	OI EU SOU UM TESTE	38
4.1	TESTE	38
5	Considerações finais	39
5.1	Limitações e dificuldades encontradas	39
5.2	Conclusão	39
5.3	Trabalhos futuros	40
	Referências Bibliográficas	41

Lista de Figuras

2.1	O modelo Customer Development [Blank 2013].	23
2.2	O Business Model Canvas de Osterwalder [Osterwalder e Pigneur 2010].	26
2.3	O Lean Canvas de Ash Muraya [Maurya 2012].	27
2.4	O ciclo de <i>feedback</i> [Ries 2011].	29
2.5	Diagrama de sequência da entrega contínua [Humble e Farley 2010].	30
3.1	Tela inicial do <i>plugin</i> para o usuário inserir o nome do projeto	34
3.2	Tela inicial de definição do banco de dados a ser utilizado.	34
3.3	Tela para definição do nome do aplicativo.	35
3.4	Tela para redefinição do nome do aplicativo.	35
3.5	Resposta do servidor de hospedagem Heroku.	36
3.6	Caminho percorrido pelo código fonte até estar disponível aos clientes em forma de aplicativo.	36
3.7	<i>Deploy</i> interrompido por causa de falhas na montagem do aplicativo pelo serviço Travis.	37

Lista de Tabelas

Introdução

1.1 Contexto

O aumento da concorrência empresarial e a utilização de tecnologias nos processos produtivos transformaram a ordem econômica e social existente. Grandes empresas começaram a produzir mais com menor necessidade de mão-de-obra, realizando cortes ou redimensionando seus quadros de pessoal [Dolabela 1999]. A geração de empregos passou a ocorrer mais fortemente nas pequenas e médias empresas, cujas criações se constituíram em alternativa de ocupação para grande parcela da população que teve condições de desenvolver seu próprio negócio, seja de maneira formal ou informal[Neto e Sales 2012]. É nesse contexto que começa a ser difundido no Brasil, especialmente a partir da década de 1990, o conceito de empreendedorismo, trazendo significativas contribuições para o crescimento econômico e desenvolvimento social.

Para [Hisrich e Peters 2004] ainda não há definição unânime e consistente para o termo, porém etimologicamente a palavra empreendedorismo vem do francês *entrepreneur*, que segundo [Becker e LACOMBE 2003], tem seu significado original como sendo "o que está no meio" ou "entre compradores". Segundo os autores, desde a Idade Média o conceito esteve relacionado ao ato de assumir riscos, principalmente atividades comerciais.

As definições oscilam, entre as postulações da economia, da psicologia, sociologia e administração. [Dornelas 2008] ressalta como característica do empreendedor a iniciativa de criar um novo negócio utilizando de forma criativa os recursos disponíveis, assumindo riscos e transformando o ambiente e o contexto que o cerca. [Drucker 2008] sugere que o surgimento da economia empreendedora é um evento tanto cultural e psicológico quanto econômico e tecnológico, e que os empreendedores têm como principal instrumento a inovação para explorar oportunidades de novos negócios.

Para [Dornelas 2008] a definição de empreendedorismo é o envolvimento de pessoas e processos que, em conjunto, levam à transformação de ideias em oportunidades. E a perfeita implementação destas oportunidades leva à criação de negócios de sucesso.

Para o termo “empreendedor” encontra-se muitas definições, uma das mais antigas e que talvez melhor conceitue o espírito empreendedor é a de [[Schumpeter 1982](#), p. 55]:

O empreendedor é aquele que destrói a ordem econômica existente pela introdução de novos produtos e serviços, pela criação de novas formas de organização ou pela exploração de novos recursos e materiais.

O processo empreendedor envolve todas as funções, atividades e ações associadas com a criação de novas empresas. Em primeiro lugar, o empreendedorismo envolve o processo de geração de algo novo, de valor. Em segundo, requer a devoção, o comprometimento de tempo e o esforço necessário para fazer a empresa crescer. E em terceiro, que riscos calculados sejam assumidos e decisões críticas tomadas; é preciso ousadia e ânimo apesar de falhas e erros. [[Dornelas 2008](#)]

De acordo com [[Dornelas 2008](#)] diversas pesquisas têm mostrado que nos países onde há apoio e incentivo a novos empreendedores constata-se uma elevação do crescimento econômico e queda da taxa de desemprego. Portanto, o empreendedorismo passa a ser um dos fatores de desenvolvimento da economia e geração de empregos. Dentro desse contexto, o empreendedorismo surge mesmo como um meio para o desenvolvimento econômico e social dos países, gerando empregos e transformando empregados em empreendedores.

Podemos afirmar que no Brasil o tema ganha popularidade devido à expansão da criação de micro e pequenas empresas e pelo fato do empreendedor brasileiro ter passado a empreender por oportunidade e não mais por necessidade [[Kelley, Singer e Herrington 2012](#)]. Com a qualificação dos empreendedores, por meio de instituições como o Sebrae (Serviço Brasileiro de Apoio às Micro e Pequenas Empresas) passou-se a analisar a capacitação de empresários, como forma de reduzir os índices de mortalidade de empresas [[Dornelas 2008](#)].

O movimento empreendedor no Brasil começa então a tomar forma na década de 1990, quando entidades como Sebrae e Softex (Sociedade Brasileira para Exportação de *Software*) são criadas. Antes disso, praticamente não se discutia empreendedorismo e criação de pequenas empresas. Os ambientes político e econômico do país não eram propícios, e o empreendedor praticamente não encontrava informações para auxiliá-lo na jornada empreendedora. O Sebrae é um dos órgãos mais conhecidos do pequeno empresário brasileiro, que busca junto a essa entidade todo o suporte de que precisa para iniciar sua empresa, bem como consultorias para resolver pequenos problemas pontuais de seu negócio. O histórico da entidade Softex pode ser confundido com o histórico do empreendedorismo no Brasil na década de 1990. A entidade foi criada com o intuito de levar as empresas de *software* do país ao mercado externo, por meio de várias ações que proporcionavam ao empresário de informática a capacitação em gestão e tecnologia [[Dornelas 2008](#)].

Foi com os programas criados no âmbito da Softex em todo o país, junto a incubadoras de empresas e a universidades/cursos de ciências da computação/informática, que o tema empreendedorismo começou a despertar na sociedade brasileira. Até então, terminologias como plano de negócios eram praticamente desconhecidas e até ridicularizadas pelos pequenos empresários. Passados 20 anos, pode-se dizer que o Brasil entra na segunda década deste novo milênio com todo o potencial para desenvolver um dos maiores programas de ensino de empreendedorismo de todo o mundo, comparável apenas aos Estados Unidos, onde mais de duas mil escolas ensinam empreendedorismo. Seria apenas ousadia se não fosse possível. Ações históricas e algumas mais recentes desenvolvidas começam a apontar para essa direção [Dornelas 2008].

O empreendedorismo, que é considerado não a intenção, mas sim a atitude tomada por uma pessoa com o objetivo de inovar a partir do crescimento econômico de uma empresa ou região, pode ser realizada em qualquer ramo de atividade e em qualquer lugar do mundo, basta o indivíduo se sentir apto para esta atitude. Algumas pessoas agem de maneira empreendedora, mesmo sem saber o nome da ânsia, ou do desejo, que os motiva e leva ao desenvolvimento de atos empreendedores, devendo este ser conhecido previamente para que se busquem formas mais corretas para a implantação de uma empresa, visando sua permanência futura por um longo período [Dornelas 2008].

Para se descobrir então a combinação perfeita entre problema e solução para um produto, a *startup* precisa estabelecer um processo iterativo que permita aprendizado constante sobre os possíveis clientes e outras premissas do negócio. Esse processo é uma combinação de experimentos práticos e investigações qualitativas, buscando extrair dados para comprovar ou refutar suposições feitas pelo empreendedor [Ries 2011].

Com o objetivo de se testar tais hipóteses de negócio e ajudar empreendedores a iniciarem o processo de aprendizagem de maneira rápida, surgiu a ferramenta conhecida como *Minimum Viable Product* MVP. [Ries 2011, p. 70] conceitua o MVP da seguinte maneira:

O MVP é aquela versão do produto que permite uma volta completa do ciclo construir-medir-aprender, com o mínimo de esforço e o menor tempo de desenvolvimento.

[Ries 2011] cita o caso do fundador da *Zappos*, Nick Swinnum que queria testar sua hipótese de que os consumidores estavam preparados e dispostos a comprar calçados por meio de um *website*, em vez de despender tempo e recursos financeiros construindo a plataforma de vendas online e o estoque de calçados, Swinnum abordou lojas físicas locais, comprou um pequeno estoque de calçados destas lojas, tirou fotos e disponibilizou-as *online* para que os clientes pudessem comprar diretamente pelo *site*. Confirmou-se assim uma demanda imediata, a *Zappos* se tornaria então um exemplo de negócio de bilhões de dólares baseado no modelo de venda *online* de sapatos.

Fica claro que após a experimentação anterior, encontrou-se um modelo de negócio escalonável e repetível, cenário ideal para o surgimento de uma *startup* de sucesso. [Blank e Dorf 2012] Mas a principal conclusão que se tem é a de que não somente é falsa a afirmação de que é preciso muito capital para se testar uma ideia no mercado, como também essa mesma ideia pode ter sua viabilidade aferida em um curto espaço de tempo como comprovado no caso da *Zappos*.

1.2 Motivação

A maioria das *startups* tradicionais utilizam o modelo de desenvolvimento de produtos (*Product Development Model*) para iniciar suas atividades no mercado [Blank e Dorf 2012]. O problema com esse modelo é que investe-se tempo e dinheiro na construção de um produto de forma totalmente isolada sem qualquer contato ou certeza de que existam clientes interessados em consumi-lo, como já foi investido muito dinheiro tentativas desesperadas de remediar os fracos resultados culminam em um só resultado, a morte de mais uma empresa.

[Blank e Dorf 2012] ensinou empreendedores durante anos que para coletar fatores a respeito de clientes, mercados, fornecedores e canais de comunicação é preciso "sair do prédio". *Startups* precisam de contato extensivo com possíveis clientes para entendê-los e a melhor forma de fazer isso é conhecendo-os percorrendo o ciclo de *feedback*.

O ciclo de *feedback* de uma *startup* compreende as atividades de transformar ideias em produtos (construir), medir como os consumidores reagem (medir) e decidir se é o caso de "pivotar" ou preservar (aprender), este ciclo é importante por permitir que o fracasso, caso ocorra, possa se mostrar nos estágios iniciais de operação, permitindo rápido aprendizado e mudanças na estratégia de operação da empresa [Ries 2011].

O MVP é apresentado como ferramenta perfeita para percorrer o ciclo de *feedback* da maneira mais rápida com o menor esforço, *ries2011lean* observou a adoção dessa ferramenta na maioria das *startups* de sucesso.

A proposta deste trabalho é aplicar um estudo do uso e adoção do MVP como ferramenta de experimentação e validação de hipóteses de mercado e desenvolver então uma ferramenta de apoio à rápida construção de produtos mínimos viáveis fornecendo um ambiente automatizado de entrega de software.

1.3 Objetivos

1.3.1 Objetivo Geral

Identificar características do MVP (Produto Mínimo Viável) como método de testes de hipótese de mercado e suas vantagens e desvantagens quando comparado aos modos tradicionais de lançamento de negócio.

1.3.2 Objetivos Específicos

1. Identificar o modo como esses grupos de usuários trabalham na construção de seus negócios, bem como a demanda de tempo necessário à disponibilização das primeiras versões de seus produtos.
2. Comparar o MVP frente ao modo tradicional de construção e implantação de negócios na *Internet*.
3. Identificar as diferentes práticas de coleta de *feedback* do cliente por parte dos desenvolvedores de MVP.
4. Desenvolver um roteiro para a construção de MVP's fornecendo um roteiro e apontando ferramentas de apoio ao empreendedor na construção do produto mínimo viável.
5. Desenvolver um template de apoio à construção rápida de um produto mínimo viável *web* usando linguagem de programação *Ruby* e *framework Rails*.

1.4 Metodologia

- Aplicar questionário a empreendedores de negócios na *internet* que fazem uso do MVP e empreendedores que não fazem uso da ferramenta.
- Produzir uma *GEM* (*plugin* de apoio construído em *Ruby*) para que desenvolvedores possam construir MVP's na *Internet* com um ambiente pré-configurado para que se possa fazer uso da "entrega contínua".

1.5 Organização do Trabalho

Fundamentação Teórica

2.1 Empreendedorismo

A Escola Empreendedora nasceu no âmbito da economia. O termo *entrepreneur* foi adotado, no início do século XIX, pelo economista francês *Jean-Batiste Say* para identificar o indivíduo que transfere recursos econômicos de um setor de produtividade mais baixa para um setor de produtividade mais elevada e de maior rendimento. A intenção era ser um manifesto e uma afirmação de dissensão: o empreendedor perturba e desorganiza.

Segundo [Drucker 2008], *Say* foi um admirador de *Adam Smith*. Ele traduziu para o francês *A Riqueza das Nações* (1776), e propagou suas idéias e políticas. Entretanto, a sua contribuição pessoal para o pensamento econômico, o conceito de empreendedor e de *entrepreneurship* é independente da economia clássica, e com ela incompatível. Para *Say* a função do empresário é combinar os fatores produtivos e reuni-los. Isso só é uma atuação do tipo “especial” quando os fatores são combinados pela primeira vez — ao passo que é mero trabalho de rotina quando feito no curso da operação de um negócio.

[Schumpeter 1982] foi o primeiro economista de renome a retornar a *Say*. No seu clássico *Die Theorie der Wirtschaftlichen Entwicklung* (A Teoria do Desenvolvimento Econômico), publicado em 1911, o autor rompeu com a economia tradicional. Suas ideias postulavam que o desequilíbrio dinâmico provocado pelo empreendedor inovador é a “norma” de uma economia sadia e a realidade central para a teoria e a prática econômica. Entretanto, mesmo assumindo que o empreendedor é importante e provoca impacto, para os economistas de um modo geral, o fato de alguém se dispor a empreender é um evento “meta-econômico”, algo que influencia profundamente e molda a economia, sem fazer parte dela.

[Timmons e Spinelli 1994] consideram os empreendedores exímios identificadores de oportunidades, aqueles que são capazes de criar e de construir uma visão sem possuir uma referência prévia, isto é, são capazes de partir do nada. O empreendedorismo é um ato comportamental, de criatividade. Eles assumem riscos calculados, tentam entender seu ambiente e controlar o máximo de fatores possíveis para que seu empreen-

dimento dê certo. Para isso, os empreendedores utilizam sua habilidade de persuasão para formar uma equipe de pessoas com conhecimentos complementares, as quais buscarão implementar e gerenciar um novo negócio ou projeto empresarial para capitalizar sobre a oportunidade identificada.

Sabidamente o conceito de empreendedorismo pode assumir diferentes significados conforme o contexto social, cultural e econômico em que é discutido, porém de forma geral, podem-se identificar três abordagens principais. A primeira deriva da linha comportamental e ocupa-se em definir os perfis de personalidade do empreendedor e compreender seu comportamento. A segunda abordagem abrange a análise das habilidades e competências do empreendedor e sua dinâmica e interação no ambiente organizacional. Trata-se de uma abordagem que valoriza habilidades como senso de oportunidade, visão e compreensão abrangente do ambiente, comprometimento pessoa e profissional. A terceira abordagem refere-se à relação que se estabelece entre empreendedorismo e economia, a inovação e a disposição para correr riscos ao investir em produtos e negócios. Em suma, as três abordagens sugerem em comum que para uma sociedade capitalista perseverar, deve contar com pessoas de comportamentos, habilidades e conhecimentos que o capacitem para a identificação de oportunidades, melhoria de processos e inovação na criação de negócios. [Costa, Barros e Carvalho 2011]. Os autores são unânimes em citar a inovação, o senso de oportunidades e a disposição para assumir riscos como sendo características intrínsecas ao empreendedorismo.

Para [Baron e Shane 2007], o processo empreendedor começa quando uma ou mais pessoas reconhecem uma oportunidade como potencial para se criar algo novo. Essa oportunidade surge de um padrão complexo de condições de mudança - cenário social, político, econômico e demográfico - e deve ter potencial para gerar valor econômico, pois tal fator é muito valorizado na sociedade atual. Este juízo de valor atribuído à oportunidade faz com que a mesma se alie a padrões morais e legais vigentes, não devendo ser restringida ou impedida por eles. Para eles, o empreendedorismo procura entender como as oportunidades surgem ou são descobertas por pessoas específicas, que desenvolvem métodos para explorá-las e desenvolvê-las, transformando o cenário onde vivem. As oportunidades surgem nos processos de mudança.

2.2 Startup

A bolha da *Internet*, fenômeno observado entre 1995 e 2001, representa o momento mais importante da *internet* nos últimos dez anos. Iniciada em 1995 quando o *Netscape* fez um IPO (sigla em inglês para oferta pública inicial de ações), a bolha gerou trilhões de dólares em investimentos na *Internet* e colaborou para o desenvolvimento de novas tecnologias e *sites* atualmente conceituados, como o *Google*. Durante este período

de tempo empresas criadas a partir da *internet* tiveram seus valores super estimados pelo mercado, até empresas que tinham como objetivos negócios completamente esdrúxulos, só pelo fato de serem empresas criadas à partir da *internet* tinham uma valorização absurda [Blank e Dorf 2012].

No contexto apresentado, surgem as empresas denominadas *startups*, [Blank e Dorf 2012] definem *startups* como uma empresa, uma parceria ou organização temporária projetado para procurar um modelo de negócios repetível e escalável. Mais tarde o termo seria associado com empreendimentos tecnológicos designados para crescer rapidamente.

No auge da especulação, o índice da bolsa eletrônica de *Nova York*, a *Nasdaq*, chegou a alcançar mais de 5000 pontos, despencando pouco tempo depois. Considera-se que o auge da bolha tenha ocorrido em 10 de março de 2000. Ao longo de 2000, ela se esvaziou rapidamente, e, já no início de 2001, muitas empresas *ponto com* já estavam em processo de venda, fusão, redução ou simplesmente quebraram e desapareceram [Blank e Dorf 2012].

2.3 Manufatura Enxuta

De acordo com [Womack et al. 2004], foram *Eiiji Toyoda* e *Taiichi Ohno*, da *Toyota* os criadores do Sistema Toyota de Produção (STP ou Toyotismo). Tudo surgiu a partir de uma viagem dos executivos aos Estados Unidos com o intuito de aprender um pouco mais sobre a linha de montagem idealizada pela Ford.

Após observarem o modelo fordista, os três japoneses ficaram decepcionados com o que viram, o gigante norte americano possuía um enorme modelo de produção em massa extremamente rico, o qual utilizava enormes espaços de estoque onde era possível armazenar toda a produção, o que dificilmente conseguiria ser feito nas fábricas da Toyota no Japão.

Por outro lado, ainda nessa viagem, uma outra coisa havia chamado a atenção dos japoneses nos EUA: o modelo de prateleiras dos supermercados. Tal maneira de vender produtos ainda era inexistente no Japão. O fato de terem produtos expostos em prateleiras, na qual um consumidor simplesmente retira um produto de acordo com sua necessidade e, posteriormente, os estoquistas recolocam tal produto para substituir o que foi retirado, atraiu o interesse dos três japoneses para um modelo totalmente voltado para a premissa do *Just in time* da qual surgiu o STP, ou *Lean*. Alinhado ao fato da *Toyota* não possuir espaço para grandes galpões como os vistos nos EUA, não se podia criar grandes estoques de produção de carros. Dado esse cenário, o modelo Lean pregava antes de mais nada a redução do desperdício, dos gastos considerados desnecessários. Então, os japoneses a frente da *Toyota* listaram os 7 principais tipos de gastos existentes:

1. **Transporte** Movendo produtos que não eram na verdade necessários para executar o processamento;
2. **Inventário** Todos os componentes, trabalhos no processo e produtos finalizados não sendo processados;
3. **Movimento** Pessoas ou equipamento se movimentado ou andando mais que o necessário para executar o processamento;
4. **Espera** Espera para o próximo passo de produção;
5. **Superprodução** Produção na frente da demanda;
6. **Desperdício** Resultantes de ferramentas pobres ou de atividades de criação do design do produto;
7. **Defeitos** O esforço envolvido para inspecionar e corrigir defeitos

Para eliminar então os desperdícios, técnicas como produção em pequenos lotes, redução de *setup*, redução de estoques, alto foco na qualidade, dentre outras, eram utilizadas. Essa nova abordagem passou a ser conhecida como *Sistema Toyota de Produção*.

Apesar do STP (*Toyota Production System*) muitas vezes ser entendido como algo novo, na verdade, muitos de seus princípios são trabalhos de pioneiros como *Deming*, *Taylor* e *Skinner* [James-Moore e Gibbons 1997]. O STP foi popularizado no ocidente por [Womack et al. 2004]. Esses autores denominam o STP de Manufatura Enxuta.

Atualmente coexistem várias definições para a Manufatura Enxuta (ME). [Womack et al. 2004], por exemplo, definem ME como uma abordagem que busca uma forma melhor de organizar e gerenciar os relacionamentos de uma empresa com seus clientes, cadeia de fornecedores, desenvolvimento de produtos e operações de produção, segundo a qual é possível fazer cada vez mais com menos (menos equipamento, menos esforço humano, menos tempo, etc.). Segundo [Shah e Ward 2003], a abordagem da ME engloba ampla variedade de práticas gerenciais, incluindo *just in time*, sistemas de qualidade, manufatura celular, entre outros. Ainda de acordo com esse autor, o ponto fundamental da ME é que essas práticas devem trabalhar de maneira sinérgica para criar um sistema de alta qualidade que fabrica produtos no ritmo que o cliente deseja, sem desperdícios.

O *Sistema Toyota de Produção* tem sido, mais recentemente, referenciado como *emph*Sistema de Produção Enxuta [Ghinato 1996]. O termo *Lean* foi cunhado originalmente no livro *emph*A Máquina que Mudou o Mundo de Womack, como resultado de um amplo estudo sobre a indústria automobilística mundial realizada pelo MIT (*Massachusetts Institute of Technology, EUA*), no qual evidenciaram-se as vantagens no uso do STP. O estudo evidenciou que o STP proporcionava expressivas diferenças em relação à produtividade, qualidade, desenvolvimento de produtos e explicava o sucesso da indústria japo-

nesa na época. Nesse sentido, os cinco princípios do *Lean*, segundo [Womack et al. 2004] são:

1. Precisamente especificar o valor por produto específico;
2. Identificar o fluxo de valor para cada produto;
3. Faça o valor fluir sem interrupções;
4. Puxar;
5. Buscar a perfeição.

2.4 O Lean Startup

O *Lean Startup* é um modelo de processo de construção de produtos cada vez mais popular a medida que o tempo passa. Eric Ries propôs o modelo após observar a *IMVU Inc.* e outras *startups* fracassarem ao validar seus modelos de negócio.

Diretamente influenciado pelo sistema toyotista de produção Eric Ries propõe em 2011 uma nova forma de pensar e construir negócios na *emphinternet*, nasce daí o termo *Startup Enxuta*. Os conceitos apresentados por Ries faziam parte, inicialmente, de seu *blog* lançado em 2008 com o objetivo de reunir suas experiências de sucesso e fracasso a fim de auxiliar futuras startups [Ries 2011].

O método da *startup* enxuta é projetado para que o empreendedor aprenda a dirigir uma *startup*, nele ao invés de se projetar planos complexos, baseados em inúmeras hipóteses, fazem-se ajustes constantes por meio do ciclo de *feedback* construir-medir-aprender [Ries 2011].

Afim de acelerar o ciclo de *feedback* já nos primeiros momentos de vida da *startup*, Ries propõe a ferramenta batizada de MVP (Produto Mínimo Viável) que [Ries 2011] define como sendo o mínimo conjunto de funcionalidades que permite uma ação e aprendizado sobre os clientes ou usuários. Sua origem é advinda do mantra *release early, release often* (libere precocemente, libere frequentemente) das metodologias ágeis de desenvolvimento, prática que coloca o *feedback* real dos usuários como guia da evolução do *software* [Ries 2011].

Assim como a filosofia Lean original, o *Lean Startup* se preocupa com o não desperdício de esforço ao se criar produtos e sua premissa principal é de quanto maior a velocidade e menor o custo de cada grande iteração, maiores são as suas chances de sucesso [Ries 2011].

Apesar da prática não estar atrelada à área de *software*, é nessa área onde este assunto tem aparecido mais, principalmente em seu epicentro, o Vale do Silício. Como forma de suporte às suas premissas, a *Lean Startup* se baseia na combinação de três fortes pilares:

- **Customer Development** Processo detalhado que serve como validação de hipóteses sobre clientes, produtos e mercado.
- **Desenvolvimento ágil** Utilização de metodologias ágeis, tais como o XP e o Scrum como forma de redução de tempo de cada iteração de desenvolvimento, aumentando a velocidade do aprendizado através de feedback dos usuários.
- **Plataformas tecnológicas como *commodity*** Uso de serviços (Amazon EC2, Google AdWords, etc.), tecnologias (Integração Contínua, etc.) e frameworks (Ruby on Rails, Wordpress, etc.) que proporcionam agilidade no desenvolvimento com baixo custo de utilização.

2.5 Customer Development

Em sua essência, o *Customer Development* é uma metodologia para a *startup* achar o seu *Product Fit* (ou *Market Fit*) [Blank e Dorf 2012], que pode ser entendido como o momento em que é comprovado que existem pessoas interessadas em utilizar seu produto e que as mesmas pagariam por isso. O modelo é composto basicamente por quatro etapas, conforme a figura a seguir ilustra:

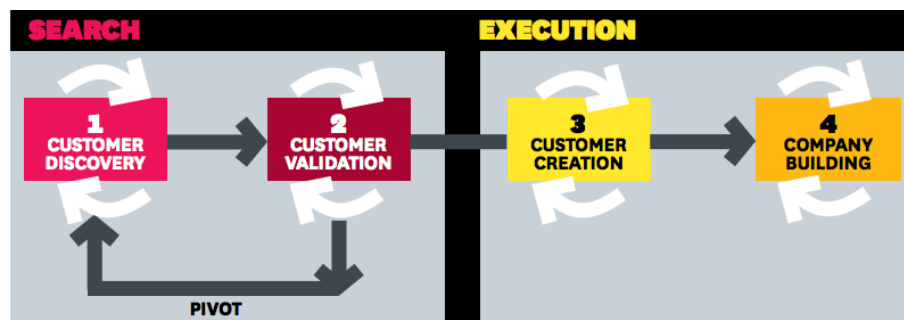


Figura 2.1: O modelo Customer Development [Blank 2013].

Customer Discovery Esta fase é caracterizada pelos testes das hipóteses de mercado e o entendimento dos problemas dos clientes pelos criadores do produto, de modo a descobrir se os clientes realmente querem o seu produto.

Customer Validation Este é o momento da validação do processo de vendas e da distribuição, onde se deve ter um modelo de negócios replicável e escalável, tentando descobrir se de fato os clientes pagariam para usar o produto.

Customer Creation É o momento onde o *Market Fit* foi atingido e quando se deve escalar as vendas.

Company Building Fase onde se constrói de fato uma empresa que se dedique ao produto. Nesta fase os clientes já foram validados, o mercado já existe e há quem pague pelo produto. É o momento onde se tem maiores insumos para conseguir investimentos financeiros externos.

2.6 Desenvolvimento ágil

Um outro pilar do Lean Startup é a prática do desenvolvimento ágil. Criado em 2001 por membros da comunidade da engenharia de software, o desenvolvimento ágil possui alguns princípios que foram descritos no que chamaram de Manifesto ágil [Beck et al. 2001]:

- **Indivíduos e interação entre eles** mais que processos e ferramentas;
- **Software em funcionamento** mais que documentação abrangente;
- **Colaboração com o cliente** mais que negociação de contratos;
- **Responder às mudanças** mais que seguir um plano.

Quando se fala de *Customer Development*, até que se atinja o *Product Fit*, diversos podem ser os ciclos existentes entre o *Customer Discovery* e o *Customer Validation*. Portanto, o tempo despendido nesse ciclo deve ser minimizado para que uma nova validação possa dizer se estamos ou não chegando onde queremos. Dessa forma, o desenvolvimento ágil se torna um grande aliado. Com seus ciclos curtos de entrega de valor e adaptabilidade a mudanças, seu manifesto encaixa bem com os princípios do *Lean Startup*.

2.7 Plataformas tecnológicas como *commodities*

O terceiro pilar do *Lean Startup* é a utilização de plataformas tecnológicas como *commodities*. Nesta área, diversas são as ferramentas que auxiliam na construção de um novo produto, como por exemplo:

- **Servidores Virtuais** Amazon EC2, Heroku, Rackspace, etc;
- **Ferramentas de Métricas** Crazyegg, Kissmetrics, Google Analytics, etc;
- **Plataformas de Publicidade** Google AdWords, etc;
- **Ferramentas de Gerenciamento** Trello, Kanbanery, Redmine, Lighthouse, etc.

O uso de ferramentas como essas podem auxiliar na concepção, na elaboração e no gerenciamento de um produto de uma *startup*. O uso de servidores virtuais, por exemplo, permite que o produto desenvolvido vá ao ar, para que os primeiros usuários possam utilizá-lo. Ferramentas como o *Google Analytics* e o *Crazyegg* auxiliam na validação das hipóteses do início do ciclo do *Customer Development* e outras como o *AdWords* podem auxiliar na captação dos primeiros recursos para que ao menos cubram os gastos gerados pela utilização dos próprios servidores virtuais. Percebe-se então como esses itens se completam e ajudam uns aos outros no andamento inicial de uma *startup*. Estes são os princípios do desenvolvimento de um produto no *Lean Startup*.

2.8 MVP: *Minimum Viable Product*

Uma das técnicas mais importantes do *Lean Startup* é o MVP [[Ries 2011](#)]. Em linhas gerais, o MVP pode ser entendido como a versão mais simples de um produto que, ao ser lançada, é capaz de gerar dados de uso sobre o mesmo através dos seus primeiros usuários (também conhecidos como *early adopters*). Seu conceito se encaixa muito bem na ideia do *Customer Development*, já que para iniciarmos o ciclo de validações das hipóteses levantadas, é preciso que liberemos uma versão inicial do produto em questão.

Por vezes, essa técnica pode gerar algum tipo de confusão pelo fato das pessoas terem certa dificuldade para encontrar o que pode ser realmente considerado como o ?Produto Mínimo Viável?. Isso pode acontecer pela simples vontade que as pessoas têm de implementar funcionalidades no produto para que ele seja ainda melhor ou até mesmo para se destacar dos concorrentes. Nesse caso, o MVP ajuda justamente para que o tempo que seria despendido na construção dessas funcionalidades não faça com o que o produto seja lançado tarde demais. Para isso, cabe às pessoas fazer uma priorização nas funcionalidades existentes, definir o que poderia ser o seu MVP, lançar a versão inicial do produto, iniciar a validação das hipóteses iniciais e entrar no ciclo novamente, até que o produto esteja maduro o suficiente para ter atingido seu *Market Fit* [[Ries 2011](#)].

2.9 *Business Model Canvas*

Uma das dúvidas existentes quando se tenta iniciar a construção do MVP de um produto, é como se documentar as hipóteses do modelo de negócios. Existem atualmente algumas ferramentas que auxiliam nessa construção mas, em linhas gerais, a maioria tenta responder a 3 grandes perguntas: *O quê?*, *Quem?* e *Como?*. Mais especificamente:

1. O quê: Problema
 - Qual é o problema que você está tentando resolver?
2. Quem: Cliente
 - Quem possui o problema?
3. Como: Produto/Mercado
 - Como você resolve esse problema?
 - O quão grande é o mercado?
 - Como você irá atingir esse cliente?
 - Como você conduz as demandas?
 - Como você vai fazer dinheiro?

O Business Model Canvas [Osterwalder e Pigneur 2010] é um modelo produzido com esse propósito. Ele se resume a um quadro dividido em blocos que representam os pontos-chave da construção de um simples modelo de negócio. Nessa abordagem, deve-se preencher o quadro identificando os valores como é possível ver na figura 2.2.

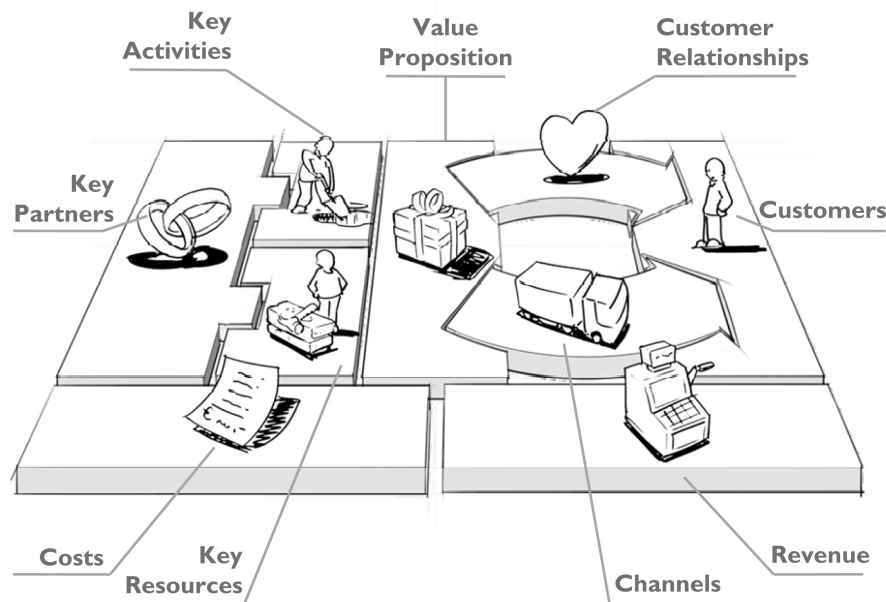


Figura 2.2: O Business Model Canvas de Osterwalder [Osterwalder e Pigneur 2010].

A proposta de modelos como este está justamente na tentativa de validação das hipóteses de produto, de clientes, de canais, de precificação, de criação de demanda, do tipo do mercado e da competição [Osterwalder e Pigneur 2010].

2.10 Lean Canvas

Ash Muraya, fundador da Spark59 e escritor, afirmou em seu *blog* que particularmente gosta do modelo de Osterwalder, dado que ele captura visualmente os componentes essenciais de um modelo de negócios. Porém, Muraya achava alguns blocos um pouco generalistas demais para *Lean Startups* e especificamente o tipo de negócio dele - aplicações *web*. Por exemplo, antes do *Product/Market Fit*, ele queria ver mais ênfase no problema/solução do que nos parceiros-chave ou num modelo de relacionamento com clientes. Algum tempo depois, Muraya conheceu o modelo criado por Rob Fitzpatrick que era uma adaptação do criado por Osterwalder, mas ainda assim achou que Fitzpatrick deixou de lado alguns blocos críticos como peças de custo/renda as quais fazem parte da construção de um modelo de negócios. Portanto, Muraya decidiu fazer seu próprio mo-

delo, que ficou conhecido como o *Lean Canvas*, um modelo adaptado do *Business Model Canvas* que, segundo o autor, está mais próximo à realidade do mundo de *startups*.

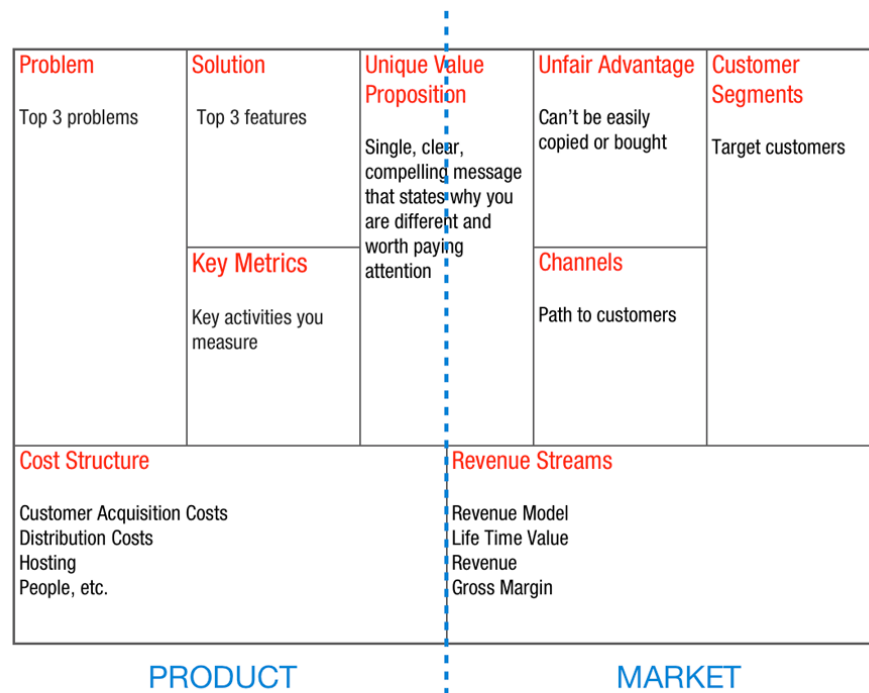


Figura 2.3: O *Lean Canvas* de Ash Maurya [Maurya 2012].

No modelo de Maurya (figura 2.3), há uma delimitação clara no meio do *canvas* sobre produto *versus* mercado e, abaixo, uma breve descrição sobre cada bloco e a ordem na qual Maurya gostaria de pensar/validar:

1. **Problem:** uma breve descrição dos três maiores problemas que você está tentando resolver;
2. **Customer Segments:** quem são os clientes/usuários do sistema? Eles podem ser segmentados?;
3. **Unique Value Proposition:** por qual razão seu produto é diferente e vale a pena ser comprado?;
4. **Solution:** qual é o MVP que apresenta o *Unique Value Proposition* acima?;
5. **Key activity:** descreve a ação-chave que o usuário irá fazer que trará a receita do produto;
6. **Channels:** lista dos canais gratuitos e pagos que você pode utilizar para atingir seu cliente;
7. **Cost Structure:** lista de todos os custos fixos e variáveis;
8. **Revenue Streams:** identifica o seu modelo de receita - inscrições, publicidade, *freemium*, etc.;

9. **Unfair Advantage:** vantagem competitiva identificada como algo que não pode ser copiado ou comprado.

2.11 Feedback

O conceito do *feedback* pode ser encontrado como apoio a diversas áreas, inclusive no desenvolvimento de *software*. Em um dos seus livros sobre *Extreme Programming* [Beck e Andres 2004, p. 34] afirma que o *feedback* é o mecanismo fundamental que permite que o cliente conduza o desenvolvimento diariamente e garanta que a equipe direcione as suas atenções para aquilo que irá gerar mais valor. Para Kent, o *feedback* está relacionado a diferentes dimensões no desenvolvimento de um sistema:

- *Feedback* do sistema: pela escrita de testes unitários, ou execução periódica de testes de integração, os programadores têm *feedback* direto do estado do sistema após novas implementações;
- *Feedback* do cliente: os testes funcionais (mais conhecidos como testes de aceitação) são escritos pelo cliente e pelos analistas de teste. Eles terão *feedbacks* concretos sobre o estado atual do sistema. Essa revisão é planejada uma vez a cada duas ou três semanas, então o cliente pode facilmente conduzir o desenvolvimento;
- *Feedback* do time: quando os clientes vêm com novos requisitos no planejamento, o time diretamente dá uma estimativa do tempo que isso irá levar para ser implementado.

[Beck e Andres 2004, p. 35] afirma ainda que *feedbacks* concretos trabalham em conjunto com a comunicação e simplicidade. Quanto mais *feedback* você tem, mais fácil de se comunicar. Se alguém tem uma objeção a algum código que você escreveu e lhe entrega um caso de teste que o quebra, isso certamente vale mais que mil horas de discussão sobre a estética do projeto. Se você está se comunicando com clareza, você vai saber o que testar e medir, aprendendo ainda mais sobre o sistema.

2.11.1 Ciclo de Feedback

A figura 2.4 representa o ciclo de *feedback* que está no centro da Startup Enxuta e guia as atividades da empresa durante suas tentativas de estabelecer um modelo de negócios repetível e escalável [Ries 2011].

Para aplicar o ciclo de *feedback* em uma *startup* é preciso primeiro identificar quais hipóteses serão testadas, à essas hipóteses dá-se o nome de "atos de fé".

Para [Ries 2011, p. 76] os dois mais importante saltos de fé são os de valor e de crescimento. Eles vão originar variáveis sintonizadas que controlam o crescimento da

Startup. Cada iteração do ciclo vai validar as hipóteses e garantir o progresso da empresa. Quando o resultados esperados forem atingidos e o processo repetido várias vezes, muda-se a direção do projeto para uma escala maior.

Uma vez evidentes as suposições do tipo ato de fé, o primeiro passo é entrar na fase de desenvolver (Build) o quanto antes um MVP que permitirá dar uma volta completa no ciclo com o mínimo de esforço e com o menor tempo de desenvolvimento, tem-se então o produto a ser testado no mercado.

Ao ingressar na fase medir (Measure) o maior desafio do empreendedor é determinar se os esforços de desenvolvimento do produto levarão a um progresso real da *startup*.

Por último e o mais importante é decidido sobre o pivot. Quando a equipe completa o ciclo Construir-Medir-Aprender, o empreendedor deve decidir, qual foi o aprendizado durante os outros passos. Se os resultados obtidos forem otimistas, então perseverar (em inglês, persevere) na mesma estratégia é uma boa escolha. Caso contrário, é aconselhado uma mudançaa na estratégia (em inglês, pivot). Outra atividade importante é a reavaliação das hipóteses, se uma delas estiver falsa, então será preciso mudá-la.

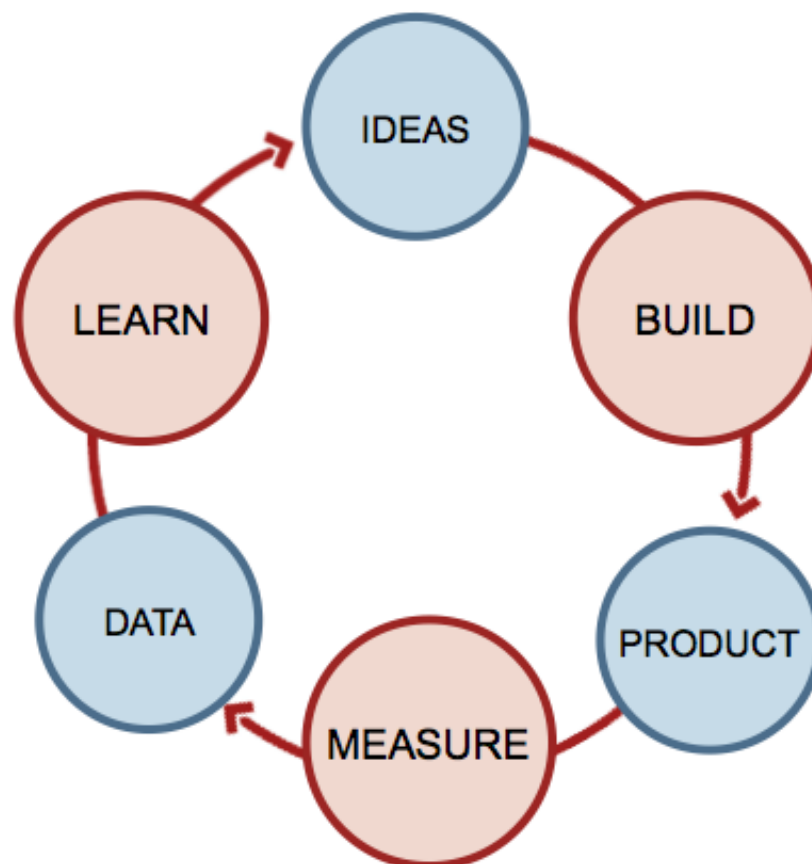


Figura 2.4: O ciclo de feedback [Ries 2011].

2.12 Entrega Contínua

A entrega contínua pode ser entendida como um padrão utilizado no desenvolvimento de software de modo a automatizar os processos de construção, testes e entrega [Humble e Farley 2010].

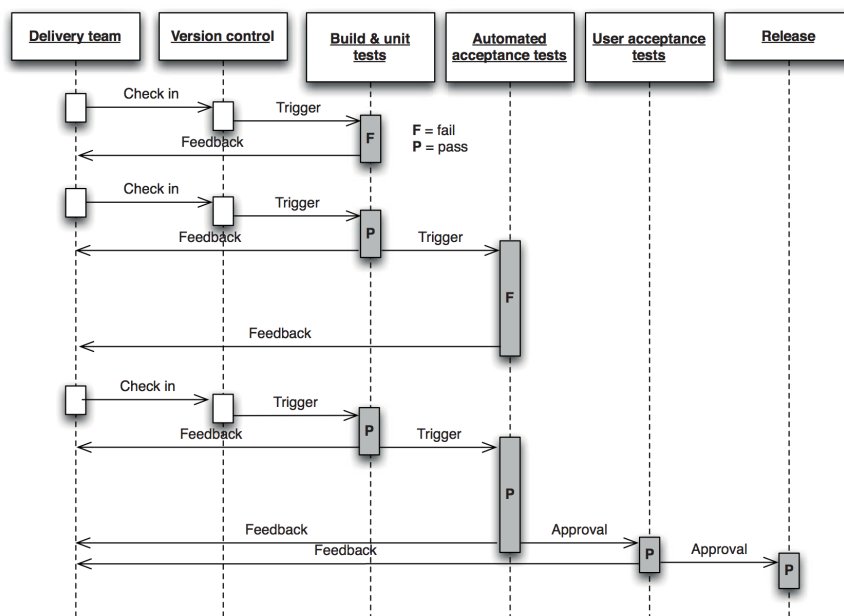


Figura 2.5: Diagrama de sequência da entrega contínua [Humble e Farley 2010].

A figura 2.5 apresenta o conjunto de validações por meio do qual um pedaço de software deve passar em seu caminho para a liberação. O código é compilado, caso necessário e, em seguida, embalado por um servidor de compilação cada vez que uma mudança é enviada a um repositório de controle de versão, em seguida, testado por um número de diferentes técnicas (possivelmente incluindo o teste manual) antes que possa ser marcado como disponível para produção.

Perpetuus, uma *Gem* para construir MVP's

3.1 O conceito de Gem

Uma Gem pode ser definida como um pacote ou uma aplicação escrito em linguagem Ruby [Berube 2007]. Essas bibliotecas podem ser divulgadas e instaladas em diversos computadores que tenham suporte à linguagem Ruby instalado.

Gems são escritas em sua maioria com o intuito de automatizar tarefas repetitivas e fornecer soluções a problemas comuns a muitos desenvolvedores.

3.2 Descrição da Perpetuus Gem

A *gem* Perpetuus foi desenvolvida para garantir que desenvolvedores *web* possam construir MVP's baseados no *framework* Ruby on Rails, isso permite que *startups* testem suas hipóteses de mercado utilizando um ambiente que acelere a produção de um produto na *internet*.

Perpetuus oferece um *template* de aplicação Ruby on Rails integrado com o *framework* RSpec para atender às boas práticas de programação orientada à escrita de testes, dentro do pacote o desenvolvedor também conta com um script para geração automática do repositório de versionamento de código utilizando o GitHub,

3.3 Requisitos de Sistema

Para utilizar o plugin desenvolvido os seguintes requisitos de sistema precisam ser preenchidos:

- Sistema operacional Unix
- Ruby 1.9 ou superior
- Rubygems 2.0 ou superior
- Cliente Git
- Heroku Toolbelt

- Bundler 1.3 ou superior

3.4 Tecnologias Empregadas

3.4.1 Ruby

Ruby é uma linguagem de programação de script interpretada, criada em 1994 por Yukihiro Matsumoto, com grande inspiração nas linguagens Python e Perl. A linguagem possui a característica de ser totalmente orientada a objeto, código aberto e com tipagem dinâmica e forte.

Existe uma filosofia por trás do Ruby, a linguagem foi desenvolvida objetivando as pessoas, buscando uma maior produtividade e fornecendo uma sintaxe muito limpa e elegante. Ruby foi projetada com o princípio da menor surpresa, tentando diminuir a frustração durante a programação. Seu criador tinha o objetivo de fazer uma linguagem que proporcionasse diversão ao programador, diminuindo as dificuldades no desenvolvimento [Flanagan e Matsumoto 2008].

3.4.2 Ruby on Rails

Antes de descrever sobre o Ruby on Rails será necessário entender o conceito de framework. Segundo [Hartl 2012], framework é uma solução para um conjunto de problemas em comum com o uso de classes e interfaces que disponibilizam objetos com funcionalidades comuns a várias aplicações. A utilização de framework pode trazer benefícios em relação à agilidade de desenvolvimento, podendo reduzir seus custos.

Com o Ruby on Rails pode-se desenvolver sistemas para a web utilizando a orientação a objeto e com a estrutura MVC (Modelo, Visualização, Controle) para construção em camadas. O framework possui alguns princípios baseados no desenvolvimento ágil, com o conceito do Do not Repeat Yourself (DRY, Não se repita), ou seja, não repetir código, e Convention over Configuration (CoC, Convenção sobre Configuração) [Akita 2006].

3.4.3 RSpec

RSpec é um *framework* de testes escrito em Ruby, permitindo que se descrevam aplicações em linguagem de domínio específico em inglês domain-specific language (DSL) [Chelimsky et al. 2010]

Como a maioria dos projetos falha em manter uma documentação atualizada, utilizar o RSpec pode ser uma excelente maneira de se documentar um projeto, pois o

resultado dos exemplos executados, quando bem escritos, formam uma especificação do projeto.

3.4.4 GitHub

O GitHub é um serviço de hospedagem para projetos de desenvolvimento de *software* que usam o sistema de controle de versão Git. GitHub oferece planos pagos para repositórios privados e contas livres para projetos de código aberto (open-source). Em maio de 2011, tornou-se o local mais popular para hospedagem de projetos de código-fonte aberto [Loeliger e McCullough 2012]

3.4.5 Heroku

Heroku é uma plataforma de hospedagem de serviços na nuvem que suporta linguagens de programação como Ruby, Java, PHP e frameworks como Ruby on Rails [Kemp e Gyger 2013].

O diferencial do Heroku está na automação da configuração necessária à criação de um ambiente para colocar uma aplicação em funcionamento, assim programadores podem focar seus esforços no desenvolvimento do *software*, ficando a cargo do Heroku juntar os módulos desenvolvidos, resolver questões de dependência e tornar o que antes era código em uma aplicação disponível aos clientes.

3.4.6 Travis

Travis é uma plataforma de serviço responsável por automatizar a montagem do código fonte que é versionado em um repositório remoto como o GitHub,

3.5 Funcionamento da GEM

3.5.1 O comando *perpetuus setup*

Para iniciar a construção de um novo projeto, o desenvolvedor utiliza o comando "perpetuus setup", assim é possível definir a estrutura inicial do MVP que será construído seguindo os passos demonstrados nas figuras 3.1, 3.2, 3.3, 3.4, 3.5:

Na figura 3.1 é pedido ao desenvolvedor que escolha um nome para o seu projeto, pois um novo diretório será criado com esse nome para guardar os arquivos contidos na estrutura inicial do projeto.

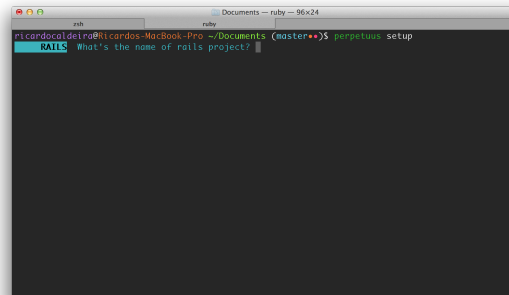


Figura 3.1: Tela inicial do plugin para o usuário inserir o nome do projeto

A figura 3.2 ilustra a etapa onde é perguntado ao desenvolvedor qual banco de dados ele quer utilizar para construir seu MVP.

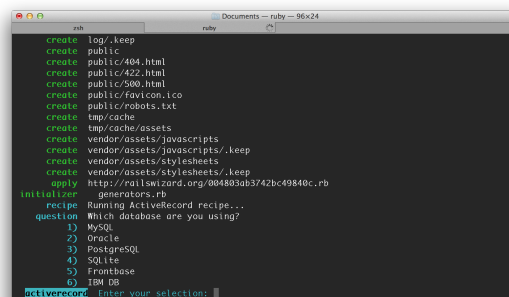
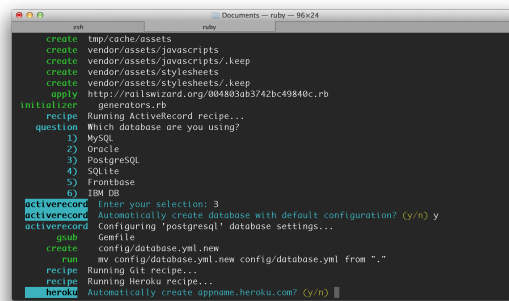


Figura 3.2: Tela inicial de definição do banco de dados a ser utilizado.

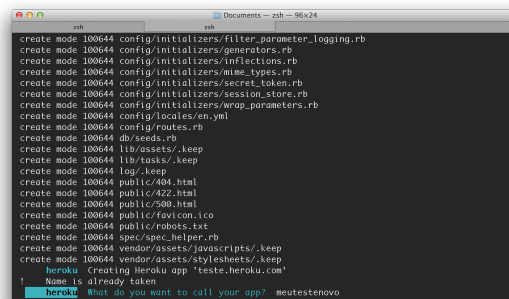
Na figura 3.3 o usuário pode deixar a cargo da ferramenta a escolha do nome para a aplicação a ser hospedada no Heroku.



```
create tmp/cache/assets
create vendor/assets/javascripts
create vendor/assets/javascripts/keep
create vendor/assets/stylesheets
create vendor/assets/stylesheets/keep
apply http://rails.wizard.org/004803ab3742bc49840c.rb
initializer generators.rb
recipe Running ActiveRecord recipe...
question Which database are you using?
1) MySQL
2) Oracle
3) PostgreSQL
4) SQLite
5) Frontbase
6) IBM DB
activerecord Enter your selection: 3
activerecord Automatically create database with default configuration? (y/n) y
activerecord Configuring 'postgresql' database settings...
gsub GeFile
create config/database.yml now
run mv config/database.yml new config/database.yml from "."
recipe Running Git recipe...
recipe Running Heroku recipe...
heroku Automatically create appname.herokuapp.com? (y/n)
```

Figura 3.3: Tela para definição do nome do aplicativo.

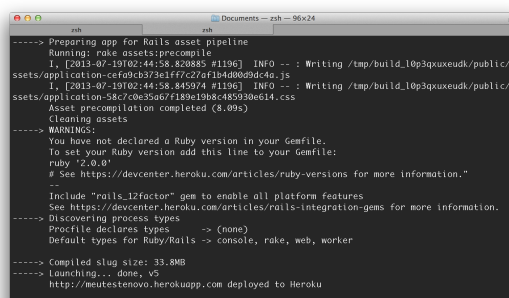
A figura 3.4 mostra o sistema pedindo ao desenvolvedor que defina outro nome para o aplicativo.



```
create mode 100644 config/initializers/filter_parameter_logging.rb
create mode 100644 config/initializers/generators.rb
create mode 100644 config/initializers/inflections.rb
create mode 100644 config/initializers/mime_types.rb
create mode 100644 config/initializers/secret_token.rb
create mode 100644 config/initializers/session_store.rb
create mode 100644 config/initializers/wrap_parameters.rb
create mode 100644 config/locales/en.yml
create mode 100644 config/routes.rb
create mode 100644 db/seeds.rb
create mode 100644 lib/assets/keep
create mode 100644 lib/tasks/keep
create mode 100644 log/keep
create mode 100644 public/404.html
create mode 100644 public/422.html
create mode 100644 public/500.html
create mode 100644 public/favicon.ico
create mode 100644 public/robots.txt
create mode 100644 spec/spec_helper.rb
create mode 100644 vendor/assets/javascripts/keep
create mode 100644 vendor/assets/stylesheets/keep
heroku Creating Heroku app 'teste.herokuapp.com'
! Name is already taken
heroku What do you want to call your app? meutestenovo
```

Figura 3.4: Tela para redefinição do nome do aplicativo.

A figura 3.5 ilustra a primeira versão hospedada no servidor de aplicação do Heroku, na última linha é possível ver o endereço na *Internet* onde o aplicativo está hospedado.



```

----- Preparing app for Rails asset pipeline
Running: rake assets:precompile
I, [2013-07-19T02:44:58.820885 #1196] INFO -- : Writing /tmp/build_10p3qxueud/public/a
ssets/application-cera0c33e1ff7c22a1b40009d4e.js
I, [2013-07-19T02:44:58.845974 #1196] INFO -- : Writing /tmp/build_10p3qxueud/public/a
ssets/application-58c7de35a67f189e198c485930e614.css
Asset precompilation completed (8.09s)
Cleaning assets
----- WARNINGS:
You have not declared a Ruby version in your Gemfile.
To set your Ruby version add this line to your Gemfile:
ruby "2.0.0"
# See https://devcenter.heroku.com/articles/ruby-versions for more information.
-----
Include "rails_12factor" gem to enable all platform features
See https://devcenter.heroku.com/articles/rails-integration-gems for more information.
----- Discovering process types
Profile declares types -> (none)
Default types for Ruby/Rails -> console, rake, web, worker
----- Compiled slug size: 33.8MB
----- Launching... done, v5
http://meutestenova.herokuapp.com deployed to Heroku

```

Figura 3.5: Resposta do servidor de hospedagem Heroku.

3.5.2 O comando *perpetuus deploy*

Ao executar o comando "perpetuus deploy" em um ambiente pré-configurado pela *gem* o código fonte produzido no computador do desenvolvedor é primeiramente versionado no repositório de código do GitHub, terminado esse processo é a vez do código produzido ser montado pelo construtor Travis, nele os testes definidos pelo desenvolvedor (usando o *framework* RSpec) são executados, caso os testes definidos passem e a aplicação consiga ser construída é a vez do código ser colocado em produção no servidor de aplicação do Heroku, ao final desse processo uma nova versão de software já está disponível aos clientes.

A figura 3.6 ajuda a ilustrar o caminho do código até se tornar um MVP disponível aos clientes.

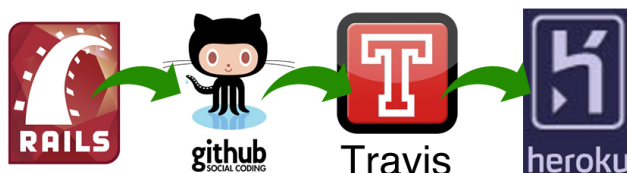
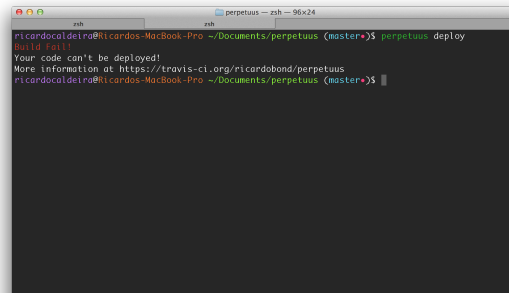


Figura 3.6: Caminho percorrido pelo código fonte até estar disponível aos clientes em forma de aplicativo.

O processo de execução descrito anteriormente exemplifica a construção de *software* baseado em entrega contínua, metodologia ideal para o desenvolvimento de MVP's por permitir rápida alteração do experimento baseada no *feedback* colhido ao final do ciclo construir–medir–aprender.

A figura 3.7 mostra um exemplo de tentativa de *deploy* da aplicação mal sucedido. Neste caso, um link é fornecido ao desenvolvedor com mais detalhes sobre os fatores que impediram o código de ser incorporado à aplicação hospedada no servidor do Heroku.

A screenshot of a terminal window on a macOS system. The window title is 'perpetuus — zsh — 96x24'. The prompt shows the user is 'ricardocaldeira' on a 'Ricardos-MacBook-Pro' in the directory '~/Documents/perpetuus' on the 'master' branch. The user has entered the command 'perpetuus deploy'. The output shows a red error message: 'Your code can't be deployed!'. Below this, it says 'More information at https://travis-ci.org/ricardobond/perpetuus'. The prompt returns to the user's shell.

```
ricardocaldeira@Ricardos-MacBook-Pro: ~/Documents/perpetuus (master) $ perpetuus deploy
Build fails!
Your code can't be deployed!
More information at https://travis-ci.org/ricardobond/perpetuus
ricardocaldeira@Ricardos-MacBook-Pro: ~/Documents/perpetuus (master) $
```

Figura 3.7: Deploy *interrompido por causa de falhas na montagem do aplicativo pelo serviço Travis.*

OI EU SOU UM TESTE

4.1 TESTE

apenas um teste com pe o e  rvores [[Hartl 2012](#)]

Considerações finais

Neste tópico estão as últimas explicações deste trabalho, quais as principais dificuldades, o que pode ser concluído e o que ele pode gerar de produtivo para startups, academia e empreendedores.

5.1 Limitações e dificuldades encontradas

A maior dificuldade encontrada durante à realização do trabalho foi na aplicação do questionário de perguntas aos empreendedores do grupo *StartupGO*, apesar de contar com mais de 1000 membros nem todos são empreendedores ou possuem uma *startup* em operação no mercado, foi preciso então fazer um levantamento das *startups* cadastradas no grupo e abordar de forma direta seus proprietários através do *chat*. Graças à ajuda do administrador da comunidade (Paolo Petrelli) foi possível então aplicar a pesquisa ao universo de 100 empresas.

A bibliografia que descreve a história por trás do nascimento do termo *startup* também se revelou um grande desafio, uma vez que a maioria do material encontrado não distinguia com clareza o período do surgimento das *startups* e empresas de capital de risco.

5.2 Conclusão

Este trabalho apresentou um estudo importante quanto à adoção do MVP como ferramenta de auxílio à rápida coleta de *feedback* das hipóteses de mercado que uma *startup* faz a respeito de seus clientes e apresentou o resultado deste estudo aplicado ao contexto das *startups* inseridas no mercado goiano.

Como forma de acelerar a construção de MVP's na internet, a *GEM Perpetuus* foi construída a fim de que desenvolvedores *web* adeptos do *framework* Ruby on Rails pudessem modificar rapidamente o MVP a partir do *feedback* de seus clientes, através do

uso de "entrega contínua" graças à disponibilização de um ambiente de automatização de tarefas no fluxo de desenvolvimento.

5.3 Trabalhos futuros

Futuramente a *gem* Perpetuus poderia suportar outros *frameworks* de desenvolvimento *web*, como *Sinatra* e o *Java Server Faces* dado que o cerne do *plugin* pode gerar *templates* para diferentes linguagens de programação.

Pesquisas futuras podem verificar se a ferramenta desenvolvida de fato acelera a construção de MVP's e se os *templates* gerados pelo *plugin* precisam oferecer suporte a outras tecnologias.

Os resultados da pesquisa aplicada às *startups* goianas podem fomentar novos trabalhos relacionados ao empreendedorismo no estado de Goiás.

Referências Bibliográficas

- [Akita 2006]AKITA, F. *Repensando a Web com Rails*. [S.l.]: Brasport, 2006.
- [Baron e Shane 2007]BARON, R. A.; SHANE, S. A. *Empreendedorismo: uma visão do processo*. [S.l.]: Thomson Learning, 2007.
- [Beck e Andres 2004]BECK, K.; ANDRES, C. *Extreme programming explained: embrace change*. [S.l.]: Addison-Wesley Professional, 2004.
- [Beck et al. 2001]BECK, K. et al. The agile manifesto. <http://www.agilemanifesto.org/principles.html>. Acesso em, v. 7, n. 08, p. 2009, 2001.
- [Becker e LACOMBE 2003]BECKER, G. V.; LACOMBE, B. M. Colocando luz em quem tem “idéias luminosas”: Competências do empreendedor de base tecnológica. *Anais 27º Encontro da Associação Nacional de Pós-Graduação em Administração. Atibaia-SP*, 2003.
- [Berube 2007]BERUBE, D. *Practical Ruby Gems*. [S.l.]: Apress, 2007.
- [Blank 2013]BLANK, S. G. Why the lean start-up changes everything. <http://bit.ly/1aqTw1l>, 2013.
- [Blank e Dorf 2012]BLANK, S. G.; DORF, B. *The startup owner’s manual: the step-by-step guide for building a great company*. [S.l.]: K&S Ranch, Incorporated, 2012.
- [Chelimsky et al. 2010]CHELIMSKY, D. et al. *The rspec book: Behaviour driven development with rspec, cucumber, and friends*. [S.l.]: Pragmatic Bookshelf, 2010.
- [Costa, Barros e Carvalho 2011]COSTA, A. M. da; BARROS, D. F.; CARVALHO, J. L. F. A dimensão histórica dos discursos acerca do empreendedor e do empreendedorismo. *RAC. Revista de Administração Contemporânea*, SciELO Brasil, n. 2, p. 179–197, 2011.
- [Dolabela 1999]DOLABELA, F. *O segredo de Luísa*. [S.l.]: Cultura Ed. Ass., 1999.
- [Dornelas 2008]DORNELAS, J. C. A. *Empreendedorismo*. [S.l.]: Elsevier Brasil, 2008.

- [Drucker 2008]DRUCKER, P. F. *Inovação e espírito empreendedor*. [S.l.]: Cengage Learning Editores, 2008.
- [Flanagan e Matsumoto 2008]FLANAGAN, D.; MATSUMOTO, Y. *The ruby programming language*. [S.l.]: O'Reilly, 2008.
- [Ghinato 1996]GHINATO, P. *Sistema Toyota de produção: mais do que simplesmente just-in-time automação e zero defeitos*. [S.l.]: EDUCS, 1996.
- [Hartl 2012]HARTL, M. *Ruby on Rails Tutorial: Learn Web Development with Rails*. [S.l.]: Addison-Wesley, 2012.
- [Hisrich e Peters 2004]HISRIC, R. D.; PETERS, M. P. *Empreendedorismo*. [S.l.]: Bookman, 2004.
- [Humble e Farley 2010]HUMBLE, J.; FARLEY, D. *Continuous delivery: reliable software releases through build, test, and deployment automation*. [S.l.]: Pearson Education, 2010.
- [James-Moore e Gibbons 1997]JAMES-MOORE, S.; GIBBONS, A. Is lean manufacture universally relevant? an investigative methodology. *International Journal of Operations & Production Management*, MCB UP Ltd, v. 17, n. 9, p. 899–911, 1997.
- [Kelley, Singer e Herrington 2012]KELLEY, D. J.; SINGER, S.; HERRINGTON, M. D. The global entrepreneurship monitor. *2011 Global Report, GEM 2011*, 2012.
- [Kemp e Gyger 2013]KEMP, C.; GYGER, B. *Professional Heroku Programming*. [S.l.]: John Wiley & Sons, 2013.
- [Loeliger e McCullough 2012]LOELIGER, J.; MCCULLOUGH, M. *Version Control with Git: Powerful tools and techniques for collaborative software development*. [S.l.]: O'Reilly Media, Inc., 2012.
- [Maurya 2012]MAURYA, A. *Running Lean: Iterate from Plan A to a Plan That Works*. [S.l.]: O'Reilly Media, 2012.
- [Neto e Sales 2012]NETO, S. P. de S.; SALES, A. H. L. *Empreendedorismo: Um modelo de liderança para o século xxi*. 2012.
- [Osterwalder e Pigneur 2010]OSTERWALDER, A.; PIGNEUR, Y. *Business model generation: a handbook for visionaries, game changers, and challengers*. [S.l.]: Wiley, 2010.
- [Ries 2011]RIES, E. *The Lean Startup: How today's entrepreneurs use continuous innovation to create radically successful businesses*. [S.l.]: Crown Pub, 2011.

- [Schumpeter 1982]SCHUMPETER, J. A. *Teoria do desenvolvimento econômico: uma investigação sobre lucros, capital, crédito, juro e o ciclo econômico*. [S.l.]: Abril Cultural, 1982.
- [Shah e Ward 2003]SHAH, R.; WARD, P. T. Lean manufacturing: context, practice bundles, and performance. *Journal of Operations Management*, Elsevier, v. 21, n. 2, p. 129–149, 2003.
- [Timmons e Spinelli 1994]TIMMONS, J. A.; SPINELLI, S. *New venture creation: Entrepreneurship for the 21st century*. [S.l.]: Irwin Homewood, 1994.
- [Womack et al. 2004]WOMACK, J. P. et al. *A mentalidade enxuta nas empresas lean thinking: elimine o desperdício e crie riqueza*. [S.l.]: Elsevier Editora, 2004.