

**ANDRESSA DE ANDRADE FREITAS**

**Conceitos e aplicações de *Lean Startup* no desenvolvimento de software em um caso de estudo.**

Monografia apresentada ao PECE – Programa de Educação Continuada em Engenharia da Escola Politécnica da Universidade de São Paulo como parte dos requisitos para conclusão do curso de MBA em Tecnologia de Software.

São Paulo  
2015

**ANDRESSA DE ANDRADE FREITAS**

**Conceitos e aplicações de *Lean Startup* no desenvolvimento de software em um caso de estudo.**

Monografia apresentada ao PECE – Programa de Educação Continuada em Engenharia da Escola Politécnica da Universidade de São Paulo como parte dos requisitos para a conclusão do curso de MBA em Tecnologia de Software.

Área de Concentração: Tecnologia de Software

Orientador: Prof. Me. Marcel Luiz Garcia de Miranda

São Paulo  
2015

## DEDICATÓRIA

*Dedico este trabalho a Deus, aos meus pais Wagner e Cristiane, ao meu irmão Andrew, à minha família e à Sonia que sempre estiveram ao meu lado, me apoiando e me dando forças pra alcançar meus sonhos e objetivos. E ao meu professor e orientador Marcel.*

## AGRADECIMENTOS

À Universidade de São Paulo – USP e a Escola Politécnica da Universidade de São Paulo - EPUSP, seu corpo docente, direção e administração que oportunizaram a janela que hoje vislumbro um horizonte superior, eivado pela confiança no mérito e ética aqui presentes.

Ao PECE – Programa de Educação Continuada em Engenharia e a todos os professores da Escola Politécnica da USP e colegas de classe que colaboraram direta ou indiretamente para a realização do trabalho.

Ao meu orientador Me. Prof. Marcel Miranda, pela orientação, compreensão, paciência e motivação.

A Deus por ter me dado saúde e força para superar as dificuldades.

Ao meus pais Wagner de Andrade Freitas e Cristiane Andréia Freitas e meu irmão Andrew de Andrade Freitas por sempre ter acreditado em mim e por todo apoio, motivação, e por ter me ensinado a ter grandes valores e princípios. Se hoje me orgulho da pessoa que sou, com certeza devo a eles.

A minha família que sempre esteve ao meu lado em todos os momentos, por todo amor, carinho, aprendizados e dedicação.

A Sonia Poloni uma grande amiga que sempre me apoiou e está ao meu lado em momentos bons e ruins, me apoiando, motivando e me ensinando.

E a todos que direta ou indiretamente fizeram parte da minha formação, o meu muito obrigada.

## RESUMO

Este estudo foi baseado nas ideias da obra do autor Eric Ries (2011) no livro *The Lean Startup* e aborda o tripé que sustenta o conceito: desenvolvimento ágil da metodologia *Lean*, *Customer Development* ou desenvolvimento de cliente e *Commodity*. Abordado por Steve Blank e Bob Dorf (2014) no livro o Manual da *Startup*, a fim de que, num contexto de incerteza, haja máxima assertividade ao entregar um novo produto ou serviço e ter sucesso nos negócios de forma sustentável utilizando inovação contínua.

Palavras-chave: Lean Startup, Customer Development, Commodity, desenvolvimento ágil, MVP (Minimum Viable Product).

## **ABSTRACT**

This study was based on the author's work ideas Eric Ries (2011) in *The Lean Startup* and the tripod that supports the concept: Agile development methodology Lean, Customer Development or development of client and Commodity. Approached by Steve Blank and Bob Dorf (2014) in the book *Startup Manual*, so that, in a context of uncertainty, there is maximum assertiveness to deliver a new product or service and to succeed in a sustainable way using business innovation continues.

Key words: Lean Startup, Costumer Development, Commodity, agile development, MVP (Minimum Viable Product).

## LISTAS DE ILUSTRAÇÕES

Figura 1 - Modelo em cascata.....	15
Figura 2 – Modelo incremental .....	17
Figura 3 – Modelo de desenvolvimento de produto .....	20
Figura 4 – Processo de desenvolvimento de clientes .....	23
Figura 5 - Fluxo de criação de <i>persona</i> .....	26
Figura 6 – Plano de modelo de negócios - <i>Canvas</i> .....	29
Figura 7 – Desenvolvimento do MVP para produtos digitais .....	31
Figura 8 – Pirâmide para gerar um <i>pivot</i> .....	33
Figura 9 – Ciclo de <i>feedback</i> .....	38
Figura 10 – <i>Devops</i> e automação de processos no ciclo vicioso de desenvolvimento versus operações .....	41
Figura 11– Primeiro modelo de negócio .....	46
Figura 12 – Relatórios do <i>Facebook</i> do Facecar .....	47
Figura 13 - Protótipos criados para o Facecar .....	48
Figura 14– Relatório sobre <i>downloads</i> disponível no Google <i>Playstore</i> .....	49

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO .....</b>	<b>9</b>
1.1	Apresentação .....	9
1.2	Objetivo .....	9
1.3	Justificativa.....	9
1.4	Organização.....	10
<b>2</b>	<b>METODOLOGIA .....</b>	<b>12</b>
<b>3</b>	<b>REVISÃO BIBLIOGRÁFICA.....</b>	<b>14</b>
3.1	<i>Startup</i> .....	14
3.1.1	Porque <i>startups</i> falham?.....	15
3.2	<i>Lean Startup</i> .....	18
3.2.1	<i>Lean thinking</i> ou pensamento enxuto.....	19
3.2.2	Desenvolvimento de cliente ou <i>Customer Development</i> .....	21
3.2.3	Estratégia de atos de fé.....	26
3.2.4	Experimentar .....	27
3.2.5	Aprender.....	37
3.2.6	<i>Devops</i> .....	39
3.2.7	Crescer.....	43
<b>4</b>	<b>ESTUDO DE CASO - FACECAR.....</b>	<b>45</b>
4.1	Primeiro <i>release</i> .....	45
4.2	Segundo <i>release</i> .....	47
4.3	Terceiro <i>release</i> .....	49
<b>5</b>	<b>CONSIDERAÇÕES FINAIS .....</b>	<b>51</b>
<b>6</b>	<b>OPORTUNIDADES PARA ESTUDOS FUTUROS.....</b>	<b>52</b>
<b>7</b>	<b>REFERÊNCIAS .....</b>	<b>53</b>

# 1 INTRODUÇÃO

## 1.1 Apresentação

É notória a quantidade de variáveis de incertezas que existem na criação de uma *Startup*, Ries (2011) propôs uma metodologia para diminuir os riscos e promover maiores chances de sucesso e para compor este modelo, utilizou recursos que vem sendo descobertos desde os tempos de constituição do sistema de produção da Toyota, e complementou utilizando ideias atuais como *Lean Thinking*, *Devops*, MVP (*Minimum Viable Product*), entre outros.

Nesta produção, está o estudo de diversos autores que defendem algumas ferramentas úteis quando se trata de inovação, baseado em Blank e Bob Dorf (2014) e outros autores como Mary Tom Poppendieck (2011).

## 1.2 Objetivo

Este trabalho propõe um entendimento da metodologia de *Lean Startup* e a apresentação das vantagens na utilização dos conceitos como: a utilização de métodos ágeis no desenvolvimento de software e uma interação constante com os usuários realizando entregas de produtos mínimos viáveis, construídos no menor tempo possível. As entregas são encaixadas no decorrer do ciclo de vida com integração contínua. Neste ciclo, os produtos criados devem ser validados, revistos e renovados com o menor custo que resolva o problema do cliente compondo um produto coeso e útil.

Propõe também o estudo de diversas metodologias e ferramentas complementares. Blank e Dorf (2014) defendem o estudo de desenvolvimento para clientes e a constituição de um modelo de negócio utilizando *Canvas*, uma técnica que permite criar visualmente uma base do negócio e promove o entendimento e a discussão por todos os envolvidos, o que complementa a ideia de Ries (2011), agregando conceitos de desenvolvimento de Mary e Tom Poppendieck (2011) e a constituição de um estudo de caso para validar estes conceitos e apresentar os resultados de um produto constituído seguindo a metodologia proposta.

## 1.3 Justificativa

De acordo com Ries (2011), as *startups* operam em um mundo de incertezas pois, não sabem quem são seus clientes e como devem ser seus produtos. Sendo assim, fica difícil prever o futuro.

Ries (2011) salienta que em um conceito clássico de desenvolvimento de software, o modelo em cascata, é defendida a ideia de haver um bom planejamento estratégico com previsões embasadas em um histórico operacional longo e estável, porém, em *startups* este mundo é irreal e por isso não funciona.

Blank e Dorf (2014) realizaram estudos que comprovam que o desenvolvimento em cascata funciona bem para um modelo maduro, porém quando se trata de inovação é uma fórmula para o fracasso e foi considerado por eles como "a principal causa da morte de *startups*", tendo elaborado ainda uma hipótese de que a maioria das histórias de fracasso e de sucesso de *startups* seguem o padrão de utilizar metodologias tradicionais que, quando criadas, não tinham o foco em inovação e abrangendo empresas com modelo de negócio, processos e público alvo bem definidos.

Baseado nesta problemática Ries (2011), combinando estudos de *marketing*, tecnologia e gestão, criou a obra *The Lean Startup* que propõe uma nova metodologia de como desenvolver softwares coesos, úteis e bem encaixados no mercado, criados num menor tempo possível com o menor custo e evitando desperdícios, dentro do cenário de incertezas gerado pela inovação.

Ries (2011), propõe que o método *Lean Startup* seja baseado em um ciclo básico de *feedback* em que é essencial construir, medir e aprender tão rápido o quanto for possível, como sendo o ideal para o mundo de *startups*. Este conceito foi criado a partir de estudos estatísticos realizados do desenvolvimento da empresa IMVU e em indústrias de manufaturas combinando ideias de *marketing*, tecnologia e gestão para designar processos ideias para eliminação de desperdícios, porém garantindo qualidade em um *time-to-market* mais rápido que, nada mais é do que medir a aceitação do produto ao mercado ou ao cliente, como defende Mary e Tom Poppendieck (2011).

#### **1.4 Organização**

Para atingir o objetivo proposto neste trabalho, serão apresentados no capítulo 2 – Metodologia – qual foi a metodologia aplicada nesta obra. No capítulo 3 – Fundamentação Conceitual – os principais conceitos de *Lean Startup* descritos por Ries (2011) e por Blank e Dorf (2014), apresentando o histórico do surgimento da metodologia e de que forma é composta o seu ciclo, incluindo também os conceitos necessários para aplicá-los na prática.

Ainda no capítulo 3, são apresentadas outras tecnologias complementares para execução do ciclo com plenitude e sucesso, e trazendo também algumas vantagens em utilizar tais ferramentas e em qual momento aplicá-las.

No capítulo 4 – Estudo de Caso – é apresentado um caso utilizando tais técnicas e quais os resultados obtidos após o uso das mesmas.

Este trabalho se encerra no capítulo 5– Conclusão – com apresentação das conclusões do modelo *Lean Startup* e aponta futuros trabalhos a serem desenvolvidos com os modelos e a metodologia aqui apresentadas.

## 2 METODOLOGIA

Para atingir o objetivo deste trabalho foi realizada pesquisas bibliográficas e o estudo literário de algumas das principais referências sobre a área, bem como sites e artigos oriundos de fontes relevante.

Sobre o tema do *Lean Startup* foram estudados livros de autores que hoje são referência no assunto, como: *The Lean Startup* (Eric Ries), *Startup Manual do Empreendedor* (Steve Blank e Bob Dorf) e *Implementing Lean Software Development* (Mary e Tom Poppendieck). Foi apresentado o ciclo que envolve a metodologia do conceito proposto por Ries (2011) e suas principais abordagens, como também, realizado um estudo complementar dos assuntos e técnicas que ajudam a compor este assunto.

Foram realizadas diversas pesquisas complementares, mas alguns dos principais temas são: o entendimento do conceito de *Lean Thinking* baseado na obra *A Mentalidade Enxuta nas Empresas: Elimine o Desperdício e Crie Riqueza* (Daniel T. Jones e James P. Womack). O assunto *Customer Development*, muito bem explicado por Blank e Dorf (2014) e complementado com a técnica de *persona* descrita na obra *The Persona Lifecycle - Keeping People in Mind Throughout Product Design* (John S Pruitt e Tamara Adlin). Ries (2011) faz ênfase na técnica de *Devops*, também complementada pelo estudo realizado em *Devops na Prática: entrega de software confiável e automatizada* (Danilo Sato). E também Ries (2011) e Bob e Dorf (2014) propõem a criação de um modelo de negócios complementando essa ideia com o estudo do livro *Business Model Generation* (Alexandre Osterwalder e Yves Pigneur).

Embora o conceito de *Lean Startup* seja abrangente, este estudo limita-se a apresentar o ciclo de vida desta metodologia e abordar alguns conceitos como prova e validação do estudo de caso de uma aplicação *mobile*, analisando, explorando e apresentando os resultados obtidos. Como sugerido por Yin (2001), o estudo de caso é usado como ferramenta de pesquisa e será utilizado não apenas as informações colhidas, mas também o planejamento, análise e conclusão.

Será feito um estudo de caso com uma abordagem qualitativa que, sob a visão de Yin (2001), é utilizada para coleta e análise dos dados para estudos gerenciais e organizacionais, o que se encaixa perfeitamente, pois, estamos fazendo prova ao conceito "*Lean Startup*".

Yin (2001) complementa ainda, que o ideal é utilizar o estudo de caso quando é levantada questões de "como" e "porque", como é o caso deste projeto que utiliza validação de conceito com estudo de caso para questões que exigem foco em acontecimentos contemporâneos e com pouco domínio e influência da situação e dos resultados.

Este estudo de caso é explanatório e implica em indicar como essas explicações podem ser aplicadas em algumas estratégias utilizando *Lean Startup*, com objetivos claros e delimitados, explicando quais foram os benefícios alcançados com a aplicação da estratégia adotada. Embora tenha a dificuldade de lidar com a transformação do modelo de referência para um estudo de caso real e a aplicação de algumas práticas para a execução do estudo de caso, o resultado obtido será baseado no que foi publicado em relação ao tema, de modo que se possa lidar com a ampla variedade de evidências de um estudo de caso.

### 3 REVISÃO BIBLIOGRÁFICA

Neste capítulo estão apresentados os diversos conceitos utilizados para o desenvolvimento do estudo de caso.

O primeiro tópico apresentará o conceito de *Startup* e identificará quais as principais falhas de sua concepção, apoiando-se nos conceitos de Ries (2011) e Blank e Dorf (2014). No segundo tópico, será apresentada a metodologia de *Lean Startup* e como ela pode ajudar a mitigar os erros, seguindo os conceitos de Ries (2011) no seu trabalho “*Hypothesis-Driven Entrepreneurship: The Lean Startup*”, explanando os principais conceitos.

Nos próximos passos serão apresentados detalhes da execução do ciclo da metodologia e as principais técnicas abordadas, conforme exposto por Mary e Tom Poppendieck (2011), Ries (2011), Blank e Dorf (2014) entre outros autores.

#### 3.1 *Startup*

Ries (2011) afirma que *startup* é uma instituição formada por um grupo de pessoas que trabalha com uma ideia diferente ou inovação com intenções lucrativas e com um modelo de negócios que gere valor, que seja repetível e escalável. Blank e Dorf (2014) concordam com Ries (2011) e explicam que ser repetível significa entregar o mesmo produto em quantidades ilimitadas sem que sofra grandes alterações e ajustes no processo para adaptação a cada cliente, e escalável no sentido de crescer progressivamente sem causar grandes impactos no custo e no modelo de negócios.

Segundo Ries (2011) e Blank e Dorf (2014), o que diferencia as *startups* de empresas tradicionais é o fato de trabalharem com inovação, em um ambiente com condições de extrema incerteza e com um portfólio de atividades que necessita de empreendedorismo e gestão para obter sucesso.

“[...]definição de *startup*: instituição humana designada a criação de novos produtos e serviços sob condições de extremas incertezas. Isso significa que empresários estão por toda parte, e a abordagem da *startup* enxuta pode funcionar em empresas de qualquer tamanho, mesmo numa de grande porte, em qualquer setor ou atividade” (RIES; 2011, p. 8).

“Uma *startup* é uma organização formada com base num modelo de negócio escalável e que possa ser repetido” (BLANK; DORF; 2014, p. XVIII).

Sob a perspectiva de Ries (2011) o conceito surgiu durante a “bolha da internet” entre 1996 e 2001 aqui no Brasil e nos Estados Unidos, na década de 90. A semântica de *start* (de

início) e *up* (para cima) está diretamente relacionada aos conceitos de inovação e empreendedorismo.

O conceito de *startup* está culturalmente ligado ao negócio on-line, porém Ries (2011) explica que esta não é uma grande verdade, pois, ao contrário do que se supõe, algumas nem sequer tem o foco na internet. Outra grande falácia é que *startup* deve ser uma empresa essencialmente pequena, pois, podem haver *startups* dentro de grandes empresas.

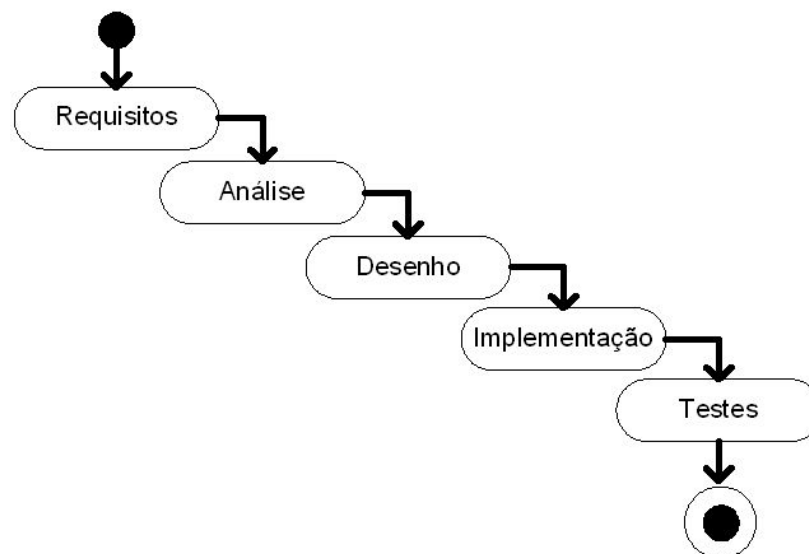
### 3.1.1 Porque *startups* falham?

Ries (2011) e Blank e Dorf (2014) contam que muitas *startups* falham porque utilizam um planejamento robusto e engessado para construir seus produtos e modelos de negócio. Estes planejamentos sofrem poucas mudanças uma vez que são estabelecidos e embasados em históricos de outras empresas, internas ou externas, e em modelos clássicos de projetos, como o *waterfall* ou modelo em cascata.

No modelo de ciclo de vida em cascata, os primeiros subprocessos são executados em estrita sequência, o que permite demarca-los com pontos de controle bem-definidos. Esses pontos de controle facilitam muito a gestão dos projetos, o que faz com que estes processos sejam, em princípio, confiáveis e utilizáveis em projetos de qualquer escala. Por outro lado, se interpretado literalmente, é um processo rígido e burocrático, em que as atividades de requisitos, análise e desenho têm de ser muito bem dominadas, pois não se permite erros. O modelo em cascata puro é de baixa visibilidade para o cliente, que só recebe o resultado final do projeto.

Na prática, é necessário permitir que em fases posteriores, haja revisão e a alteração do resultado obtido nas fases anteriores. (FILHO; 2001, p. 12)

Figura 1 - Modelo em cascata



Fonte: FILHO (2001)

Porém, este cenário não é real para o mundo de *startups* que trabalham com inovação e tecnologia pois, estas empresas ainda estão em formação e não possuem histórico significativo, além de trabalharem com produtos ou serviços inovadores sem grandes casos de sucesso no mercado, para que possam ser comprados pelo público, simplesmente pela fama que possam possuir.

Muitos destes insucessos, afirma Ries (2011), ocorrem pelo uso dos processos tradicionais que não servem para *startups* por não terem sido criados para atender inovação, principalmente a cultura de ter um planejamento robusto para um determinado produto ou serviço. “O primeiro problema é a fascinação de um bom plano, uma solida estratégia, e minuciosa pesquisa de mercado” (RIES; 2011, p. 9). Em uma empresa comum, um planejamento robusto funcionaria muito bem, explica Ries (2011), mas nas *startups* existem outras necessidades como a de modificar e evoluir, o tempo todo, suas estratégias e seus produtos ou serviços, sem perder sua visão essencial, pois, trabalha com um público incerto e não tem plena certeza de como irá atender as necessidades e desejos do mercado.

No ciclo de desenvolvimento de projetos clássicos, o modelo *waterfall*, salientado por Filho (2001), é um processo que estabelece um modelo de projetos e constrói um produto ou serviço sem que haja alterações até a conclusão de todo o ciclo, ou seja, ao cliente só é possível testar o produto quando ele já está pronto e na fase final. Porém, Ries (2011) explica que essa abordagem pode ser um desperdício quando se trata de uma *startup*, que trabalha com inovação e ainda não sabe com precisão quem são seus clientes e o que eles desejam. Blank e Dorf (2014) concordam que o público alvo de uma *startup* ainda é uma suposição e seu produto ou serviço está em constante fase de adaptação, ao contrário do modelo tradicional, que pressupõe que a empresa saiba o que seu cliente deseja, além de ter histórico e informações para comparar com o sucesso de outras empresas concorrentes. A necessidade de uma plano de negócios convencional, tradicional e robusto, faz com que o modelo de processo clássico seja falho e ultrapassado e Ries (2011), postula sobre a incapacidade de sobrevivência de um plano quando houver o primeiro contato com os clientes, e complementa: “se você não tiver testado primeiro as premissas do seu modelo de negócios fora do seu prédio, o seu plano de negócio não passa de literatura criativa”. Com isso, fica claro que a utilização deste de desenvolvimento clássico, leva a um crescimento não sustentável.

O sucesso das *startups* é baseado em ciclos de experimentação, incrementos e desenvolvimento, chamado por Ries (2011) de ciclo de *feedback*, e que pode ser

complementado por Blank e Dorf (2014) com o uso de "Desenvolvimento de Cliente" ou *Customer Development*.

Blank e Dorf (2014) e Ries (2011) concordam que parte das *startups* criam produtos que os supostos clientes não desejam usar e que não despertam interesses, ou seja, investem muito nestes produtos que, quando disponibilizados em ambiente produtivo, percebe-se que foram produzidos produtos desinteressantes para o mercado.

Por isso é muito mais útil e lucrativo criar um produto simples e ir desenvolvendo-o através dos *feedback* recebidos em um ciclo iterativo incremental, que favoreça a adequação da estratégia e do produto final, a partir da opinião dos clientes.

O processo incremental não é um método novo, Sommerville (2004) já explicava que é um processo mais apropriado do que o *waterfall*, pois produz produtos que atendem melhor às necessidades dos clientes, porém, Blank e Dorf (2014) complementam esta metodologia com o uso de *Customer Development*, e Ries (2011) utiliza este conceito para criar o *Lean Startup*.

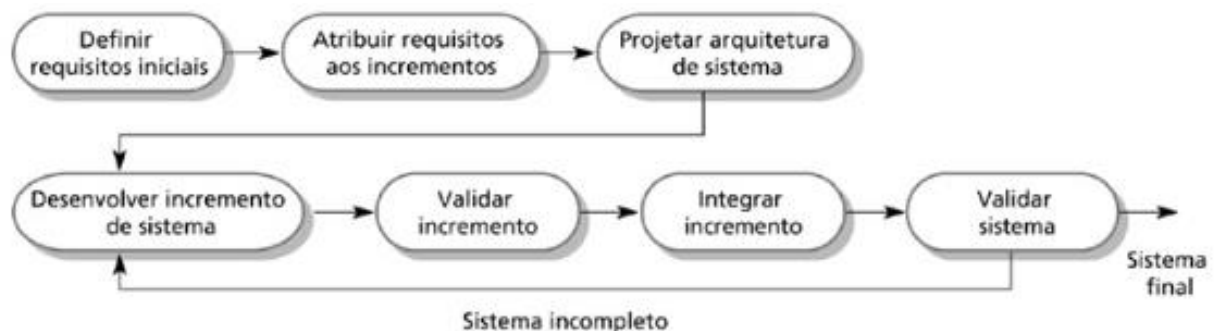
Em um processo de desenvolvimento incremental, os clientes identificam em um esboço, as funções a serem fornecidas pelo sistema. Eles identificam quais funções são mais importantes e quais são menos importantes para eles. [...]. As funções prioritárias são entregues primeiramente ao cliente.

Uma vez identificados os incrementos, os requisitos para as funções a serem entregues no primeiro incremento são definidos em detalhes, e esse incremento é desenvolvido utilizando o processo de desenvolvimento mais apropriado. [...]

As mudanças nos requisitos para o incremento atual não são aceitas.

Uma vez que um incremento é concluído e entregue, os clientes podem colocá-lo em operação. [...]. Eles podem experimentar o sistema, o que lhes facilita esclarecer seus requisitos para os incrementos subsequentes e para versões posteriores do estágio atual. (SOMMERVILLE; 2004. p. 43, 44)

Figura 2 – Modelo incremental



Fonte: Adaptado de Ian Sommerville (2004)

Ries (2011) e Blank e Dorf (2014) concordam que muitas das *startups* crescem e morrem antes do tempo, por não evoluir seu produto e sua estratégia diante dos *feedback* de mercado, em ciclos curtos.

Para que uma *startup* possa ser bem sucedida, uma série de conceitos deve ser aplicada. É necessário a composição de processos, gestão e disciplina evitando o “simplesmente faça”. Mas estes princípios culturalmente são mal adaptados ao ambiente incerto e caótico no qual se encontram as *startups*, de tal forma que é necessário que se estabeleça formas de mensurar o progresso, apesar das incertezas e de estabelecer novas estratégias e planos de ação, permitindo aos empreendedores formular previsões que possam ser testadas.

Ries (2011) afirma que hoje somos mais produtivos do que nunca devido a nossa capacidade tecnológica e administrativa, porém, há ainda muitos desperdícios e fracassos no caminho e a partir disso, surgiu o conceito de *Lean Startup* com intuito de ajudar a impedir estes insucessos.

### **3.2 *Lean Startup***

Ries (2011) explica que para criar o conceito de *Lean Startup* ou *startup* enxuta, partiu da revolução na Toyota, promovida por Taiichi Ohno e Shigeo Shingo com o pensamento enxuto.

Mary e Tom Poppendieck (2011) e Ries (2011) contam que por volta da Segunda Guerra Mundial a Toyota utilizava produção em massa para produzir peças de carros em grandes lotes e gastava enormes quantias de dinheiro para comprar e manter máquinas que suportassem esse tipo de produção em larga escala.

A Toyota ainda era uma fábrica pequena e estava tentando competir com as empresas americanas de grande porte e para isso, precisava de um novo jeito de produzir.

Poppendieck (2011) e Ries (2011) pontuam que Taiichi Ohmo era um chefe de produção que estudou os conceitos de Henry Ford, Kiichiro Toyoda e Sakichi Toyoda, os dois últimos, engenheiros têxteis que haviam revolucionado a tecelagem naquela época e que automatizaram o processo de tear da fábrica. Ohmo entendeu o processo de *just in time* e automação, e junto com Shigeo Shingo, que era um engenheiro brilhante, propôs a utilização de máquinas menores que conseguiam produzir diversos tipos de peças em pequenos lotes e criaram o Sistema Toyota de Produção.

A grande vantagem era que essas máquinas eram fáceis e rápidas de serem reconfiguradas, então Shigeo criou o conceito de SMED (*Single-Minute Exchange of Die* – troca rápida de ferramentas) e colocou-o em prática nas fábricas da Toyota (RIES, 2011).

Outro grande *insight* (entendimento súbito da solução de um problema) é que a fabricação em pequenos lotes possibilitou a identificação muito mais rápida de problemas, fomenta Ries (2011). E para isso foi criada a corda *andon*, que dava o direito a qualquer funcionário de parar o fluxo por completo e a qualquer momento, caso descobrisse um problema. Embora houvesse um custo de parar a produção repetidamente, este custo não se comparava ao lucro de corrigir rapidamente e de maneira eficiente os problemas (RIES, 2011).

Ainda sob a perspectiva de Ries (2011), dentre os princípios descobertos estão: aproveitar a criatividade e conhecimento de cada funcionário para acelerar a produção do tipo *just in time* e reduzir os custos na fabricação em lote, promovendo qualidade de dentro para fora. Poppendieck (2011) concorda com Ries (2011) e complementa que a ideia é que o sistema seja automático e a prova de erros, onde as peças deveriam chegar na linha de montagem somente quando fossem ser utilizadas.

Ries (2011) diz que desta forma a Toyota pôde atender mercados menores, fragmentados e variados até se tornar a maior fabricante de carros em 2008.

Assim a Toyota descobriu uma nova forma de desenvolver um negócio sustentável tão rápido o quanto possível, eliminando desperdícios. E dessa mesma forma, uma *startup* pode minimizar o gasto de tempo e esforço, pois é muito melhor descobrir os problemas mais cedo, defende Poppendieck (2011).

Significa “desperdício”, qualquer atividade humana que absorve recursos, mas não cria valor: erros que exigem retificação. Produção de itens que ninguém deseja, acúmulo de mercadorias nos estoques, etapas de processamentos que na verdade não são necessárias, movimentação de funcionários e transporte de mercadorias de um lugar para o outro sem propósito, grupos de pessoas em uma atividade posterior que ficam esperando porque uma atividade anterior não foi realizada dentro do prazo, e bens e serviços que não atendem às necessidades do cliente. (JONES; WOMACK, 2004 p. 3).

### **3.2.1 *Lean thinking* ou pensamento enxuto**

Jones e Womack (2004) afirmam que o melhor antídoto contra o desperdício é o pensamento enxuto, gerando valor através do alinhamento de processos e ações sequenciais sem interrupções, de forma cada vez mais rápida e eficaz, ou seja, é a maneira de oferecer maior valor aos clientes utilizando cada vez menos recursos o tão rápido quanto possível, porém,

promovendo trabalho gratificante e satisfatório através de ciclos de *feedback* imediatos sobre os esforços realizados, transformando o desperdício em valor e eficiência.

Conforme Jones e Womack (2004), por volta do pós Segunda Guerra Mundial, durante a chamada era industrial, os responsáveis pela definição do valor eram os executivos, especialistas e engenheiros que construía produtos complexos avaliados para serem o que o cliente queria, porém, despertavam pouco interesse nas pessoas, exceto para seus criadores que afirmavam que assim que explicassem como o produto funcionava, tinham certeza de que o cliente iria querer obter o produto e aqueles que não quisessem, seria devido à falta de capacidade para entender o valor do produto.

À época, as indústrias manufatureiras seguiam um modelo de lançamento de produto adotado também pelas *startups* (BLANK, DORF; 2014). À primeira vista, não houve um grande problema e foi uma boa opção para uma empresa já em operação, na qual os clientes são conhecidos, o mercado está bem definido e as bases de competitividade são claras e compreendidas. Ries (2011) e Blank e Dorf (2014) complementam que a partir deste contexto conhecido, elabora-se um plano de negócios alicerçado nas avaliações das necessidades dos clientes conhecidos, constrói-se o produto e o submete a testes *alfa/beta* (testes que garantem que o produto está de acordo com as especificações), que proporcionam *feedback* aos engenheiros para corrigir problemas eventuais até a data de lançamento.

Figura 3 – Modelo de desenvolvimento de produto



Fonte: Steve Blank e Bob Dorf (2014)

Mas como já sabemos, este contexto é irreal para *startups*, pois no início ainda não tem clientes e o fundador pode apenas fazer suposições e assumir hipóteses a respeito de quem são os clientes e o que realmente gera valor a eles. Ou seja, a *startup* é baseada em fé e palpites, que devem ser validados o mais brevemente com os clientes e rapidamente corrigir o que estiver errado.

Uma questão que se coloca é como definir o valor? Jones e Womack (2004) afirmam que o valor é criado pelo produtor, porém somente pode ser definido pelo cliente final, e só é significativo quando um produto, seja esse um serviço ou um bem, gera satisfação às

necessidades dos clientes e que fundamentalmente, os produtores deveriam repensar o valor e a perspectiva do cliente, sem esquecer que a qualidade desses produtos pode ser melhorada a medida em que os custos fundamentais vão sendo reduzidos. As *startups* já estabelecidas propõem um modelo de negócio assumindo um rol de características do produto e operam baseando-se em pesquisas com testes, para comprovar suas hipóteses preliminares, aprendem com os resultados e *feedback* e reelaboram suas hipóteses baseadas nas lições aprendidas, quando voltam a coloca-las à prova a fim de desenvolver um modelo de negócio lucrativo e sustentável.

*Lean Startup* de Ries (2011) utiliza práticas que tem como principal objetivo mitigar o desperdício e para isso, adota o conceito “pensamento enxuto” (*lean thinking*), que por sua vez, promove o desenvolvimento de produtos através de incrementos evolutivos construídos sob constantes contatos com clientes, validando e eliminando hipóteses e suposições incorretas, reduzindo assim o custo, tempo e o esforço e também criando táticas otimizadas para encontrar atração de mercado.

“Essa abordagem se desenvolve sobre diversas ideias prévias de administração e desenvolvimento de produto, incluindo a manufatura enxuta, o *design thinking*, o desenvolvimento de clientes (*customer development*) e o desenvolvimento ágil. Representa uma nova abordagem para criar a inovação contínua. “Eu a denominei de startup enxuta (*lean startup*)” (RIES; 2011, p.4).

*Lean Startup* é uma abordagem com princípios para o desenvolvimento de novos produtos, que ajuda a entender o que sua equipe deve fazer depois de formada, quais processos utilizar e quais as responsabilidades deve cumprir.

### **3.2.2 Desenvolvimento de cliente ou *Customer Development***

O Desenvolvimento de Cliente é um processo que reúne os *feedback* dos clientes sobre o produto, canais de distribuição, preço, mercado e outras características a mais que compõem o produto e utiliza essas informações para reformular e otimizar esse produto, até conseguir um modelo de negócio recorrente e escalável, afirma Blank e Dorf (2014).

Tanto Ries (2011), quanto Blank e Dorf (2014) concordam que o ponto crucial deste processo é que os fatos estão fora do escritório, e que é necessário entender as necessidades do cliente e combiná-las com os *feedback* obtidos para compor o processo de desenvolvimento do produto.

A principal diferença com o modelo tradicional de desenvolvimento de produto é que ele não contempla a opinião do cliente até os testes *Alfa/Beta* quando já é tarde. Os testes *Alfa/Beta* ou A/B ou multivariado é uma estratégia de *marketing* que oferece versões diferentes de um mesmo produto aos clientes, para afinar a estratégia de promoção deste produto (RIES, 2011).

Na ocasião do teste beta, trabalha-se com um pequeno grupo de usuários externos para testar o produto e assegurar de que ele funcione de acordo com as especificações. A área de *marketing* desenvolve um plano completo de comunicação ao mercado, configurando o site corporativo, abastecendo vendas com um completo sortimento de matérias de apoio e colocando a equipe de relações públicas para fazer sua parte que começa a contatar *blogs* para pesquisar o tempo de antecedência necessário para publicação de material na imprensa, afinando assim a estratégia ao passo que o *marketing* inicia suas atividades de promoção da marca (BLANK; DORF, 2014, p. 5).

Ries (2011) e Blank e Dorf (2014) afirmam que é necessário o contato com os clientes reais para entender, interagir e aprender com suas reações ao utilizarem o produto ou o serviço e que as principais vantagens de observar o cliente real é a possibilidade de avaliar o comportamento e aprender quais são as necessidades do mesmo, criando produtos com dados mais precisos, ao invés de somente se basear em ideias hipotéticas.

O modelo de desenvolvimento de clientes, descrito por Blank e Dorf (2014) é elaborado para resolver as principais falhas do modelo tradicional e está dividido basicamente em busca ou pesquisa para criar a hipótese do modelo de negócio, em execução do que será testado e por último, a fase da prova do que foi desenvolvido.

A primeira etapa do ciclo, que seria a de busca, divide-se em descoberta e validação pelo Cliente.

Blank e Dorf (2014) discorrem que na descoberta, os fundadores transformam inúmeras hipóteses em um plano de negócio e desenvolvem um plano de experimentações para testar as hipóteses e transformá-la em fatos. Nesse momento também é gerado o MVP (*Minimum Viable Product*) de baixa fidelidade baseado no modelo de negócio, o que também é defendido por Ries (2011). Um MVP é uma versão simplificada do produto que carece de diversos recursos, mas pode ser lançada em ambiente produtivo com o mínimo de esforço e tempo de desenvolvimento, sendo apenas um teste de conceito e das hipóteses fundamentais do modelo de negócio, que deve ser amadurecido e melhorado através dos *feedback* ou respostas recebidas dos usuários. Segundo Ries (2011), um MVP provoca um marco de aprendizagem, chamado *baseline*, e a partir deste é criado um modelo de crescimento que serve como alicerce para aprendizagem a respeito dos clientes e suas reações.

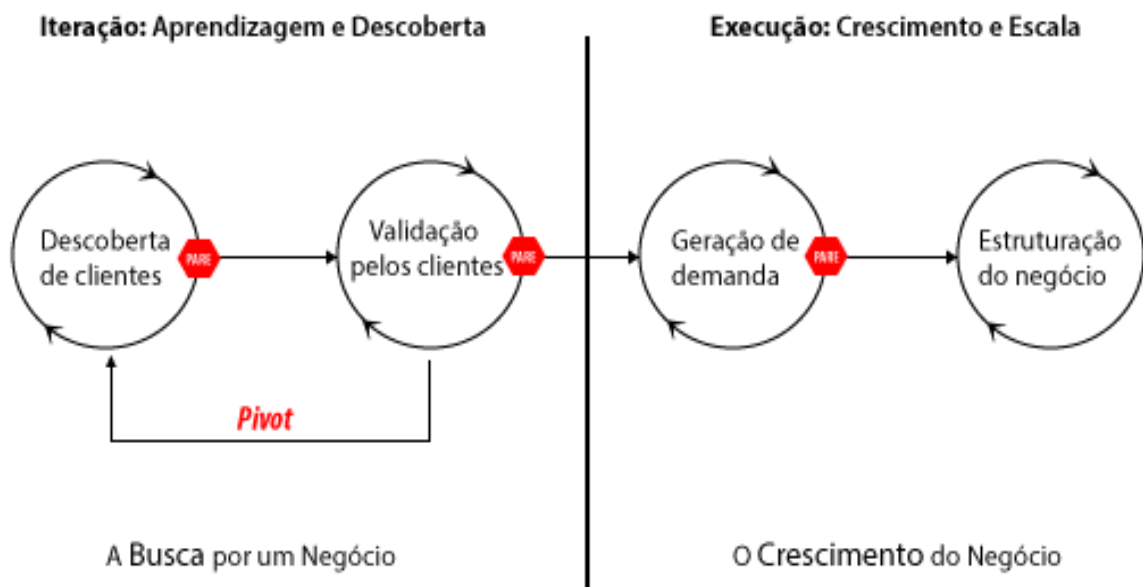
Nessa etapa é importante ir às ruas para testar a reação dos clientes a cada hipótese, para que seja possível identificar o que é uma verdade. A maior importância de ir a campo é entender os recursos que podem ter no produto e a profundidade dos problemas, para solucioná-los e assim garantir que seja comprovada a necessidade pelos consumidores.

Na validação verifica-se os resultados do modelo de negócio para saber se são repetíveis e escaláveis, disponibilizando este produto para uma escala maior de clientes, com testes mais rigorosos, mais extensos e em maior quantidade. Caso o resultado seja negativo, retorna-se à descoberta. É neste momento que é tomada a decisão de fazer um *pivot* (mudança estratégica) ou perseverar, como defende Ries (2011).

Um *pivot* é uma hipótese estratégica que representa as modificações no modelo de negócio e que podem ocorrer de diversas formas possíveis, cada uma para uma determinada finalidade. “É um tipo de mudança estruturada, projetada para testar uma nova hipótese fundamental a respeito do produto, do modelo de negócio do *lean startup*.” (RIES; 2011, p.165).

Então, construído um MVP de alta fidelidade para testar as características principais junto ao cliente, para comprovar a existência de um grupo de consumidores, identificar o comprador, implementar as estratégias de preço, demanda e distribuição (BLANK; DORF, 2014).

Figura 4 – Processo de desenvolvimento de clientes



Fonte: Steve Blank e Bob Dorf (2014)

No próximo passo, que é a geração de demanda, Blank e Dorf (2014) afirmam que é o momento de aceleração, onde há os investimentos com publicidade e propaganda com a finalidade de aumentar a demanda e a incorporação dos dados de consumo dos usuários finais e definição dos canais de venda.

E por último, a estruturação da empresa, que é o momento em que a empresa deixa de ser *startup* para se transformar em uma companhia de operação corrente. Neste momento, o modelo de negócios possibilita elevar a escala de modo sustentável, trocando a orientação informal de aprendizagem e descoberta por outra formal, com a criação de estruturas em departamentos para promover a escala (RIES, 2011).

### 3.2.2.1 *Persona*

Segundo John Pruitt e Tamara Adlin (2010), Ries (2011) e Blank e Dorf (2014), construir produtos que as pessoas querem e precisam, parece uma tarefa fácil e óbvia, mas não é.

Os usuários são complicados e tem características, necessidades e desejos variados, no entanto, é possível criar representação de perspectivas dos mais variados clientes através de modelagem de usuários fictícios e específicos, chamados de *persona* ou arquétipos (PRUITT; ADLIN, 2010). A notação de *persona* foi criada por Alan Cooper e ajudam a organizar o foco no usuário e transmitir as mesmas informações que os documentos textuais, porém, com uma abordagem muito mais ilustrativa, lúdica e didática promovendo um entendimento homogêneo a todos os membros do time que estão desenvolvendo o produto.

*Persona* são ideias sobre quem são os usuários de um determinado produto e como são as suas vidas (BLANK; DORF, 2014).

Utilizar esta técnica para representar usuário, traz uma série de benefícios pois, transcrevem uma segmentação de mercado representando quais são as premissas e conhecimentos explícitos sobre os usuários identificados, criando uma linguagem comum para representa-los. Segundo John Pruitt e Tamara Adlin (2010), representa até mesmo um grupo pequeno, permitindo manter um foco e ajudando na tomada de decisão.

Sob a perspectiva de Ries (2011), o perfil encontrado tende a ser provisório para as *startups*, pois, os clientes encontrados num primeiro momento de concepção do produto, são apenas suposições baseadas num intervalo de usuários e que a partir da execução dos ciclos da *Lean Startup*, *persona* deve ir se modificando e se adaptando para representações mais fiéis possíveis da *baseline* atual. “O perfil encontrado deve ser provisório até a estratégia mostrar,

via aprendizagem validada, que pode-se atender esse tipo de cliente de momento sustentável” (RIES, 2012; p. 90).

John Pruitt e Tamara Adlin (2012), Ries (2011), Massari (2014) e Blank e Dorf (2014) concordam que esta técnica auxilia na descoberta do perfil e interesses dos clientes, mas não substitui os testes do mundo real da experiência do usuário e nem as análises que devem ser feitas sob os resultados obtidos, pois, é uma visão generalista que não consegue prever toda a gama de complexidade de criar um produto. Porém, Ries (2011) salienta que é importante saber que nem o melhor *design* pode antever as complexidades de trazer um produto ao mundo real.

Adlin e Pruitt (2010) apresentam um *framework* de ciclo de vida de *persona* que vai se modificando a medida que se tem mais informações e variâncias dos usuários, técnica que também é proposta por Ries (2011) e Blank e Dorf (2014).

Segundo Adlin e Pruitt (2010), a criação de *persona* em séries é um processo simples onde é preciso criar um esqueleto de um personagem apresentando somente o que é mais importante, depois incorporar detalhes concretos, personalidade e uma história. É possível subdividir esse processo em seis passos:

1. **Identificação Ad Hoc** – durante uma reunião rápida com os *stakeholders*, identifique quem são os usuários pensando no que querem, no que precisam e crie *ad hoc persona*.
2. **Faça alguns filtros** – entenda quais são realmente os fatos e as premissas para cada usuário.
3. **Organize essas ideias em Data Sources** - extraia o que for relevante sobre seus usuários, domínios, relacionamentos, temas e o que há de comum entre eles. A sugestão é que se crie isso em um *board* para ajudar a visualizar as ideias com mais clareza.
4. **Crie esqueletos** – identifique quais são as categorias e subcategorias dos seus usuários.
5. **Priorize os esqueletos** – identifique quem são as *persona* mais importantes para o negócio e para o produto. Dentre essas, quais vale a pena transformar em *persona*.
6. **Transforme em persona** – enriqueça os esqueletos com dados individuais e elabore suas histórias dando personalidades e contexto.

Na imagem 5, é possível observar o fluxo de criação de arquetipos ou *persona*, proposto por Addin e Pruitt (2010), onde é demonstrado os passos citados acima, exibindo o processo que é preciso executar para transformar fontes de dados em *persona*. Uma informação bastante relevante, é que neste fluxo é proposto a criação de um diagrama de afinidade para representar as informações, promovendo assim um entendimento macro dos fatos e premissas abortados e uma maior assertividade nos resultados.

Figura 5 - Fluxo de criação de *persona*

Fonte: Addin e Pruitt (2010)

Existem algumas ferramentas de mercado que podem ajudar a identificar *persona* como a Mosaic, da Serasa Experian, que possui técnicas analíticas e estatísticas que já classificou bilhões de usuários em 29 países destacando suas principais características.

Ries (2012) e Blank e Dorf (2014) salientam que os perfil de *persona* mais interessantes para o mundo de *startups* são os clientes de perfil visionários, que estão mais suscetíveis a serem os adotantes iniciais, pois sempre querem ser pioneiros a terem produtos inovadores e estão preocupados em criar vantagem competitivas com seus concorrentes e por isso utilizam a imaginação para testar e ajudar completar o que falta.

### 3.2.3 Estratégia de atos de fé

De acordo com Ries (2011), a cada iteração é gerado um plano de negócio baseado em suposições, que toma estas hipóteses como verdades e prossegue mostrando como alavancar a visão da empresa. As suposições são denominadas atos de fé, que devem ser testadas o mais rápido possível, porque o sucesso de todo empreendimento depende deles.

Ries (2011) defende que o segredo é realizar testes rigorosos sem perder de vista a visão geral da empresa. E o primeiro desafio de um empreendedor é construir uma empresa capaz de testar estas suposições sistemicamente.

Há diversos tipos de destruição de valor que devem ser evitados. Ries (2011) exemplifica como uma empresa que cresce por meio da continua captação de investimentos e de muita propaganda paga, mas que não desenvolve um produto que gere valor.

Os primeiros planos estratégicos tendem a ser orientados a atos de fé e para converter estes instintos em dados. Os empreendedores devem “sair do prédio” como diz Steve Blank e Dorf (2014) e começar a aprender.

O fato é que as *startups* precisam de contato extensivo com possíveis clientes para entendê-los e confirmar se as hipóteses são realidade e se o cliente possui um problema importante cuja solução vale a pena.

O ideal é não se iludir com conversas com clientes esperando que eles saibam o que eles realmente querem e precisam, pois geralmente esta é uma falácia, e Ries (2011) complementa, refine seus planos interminavelmente, para não se basear em fatos errados.

### 3.2.4 Experimentar

Ries (2011) defende que em um modelo tradicional mede-se o progresso avaliando se o produto final atende aos requisitos especificados e se o ciclo de projeto está sendo executado dentro do tempo e do custo estimado. Porém, quando se trata de inovação, surgem questionamentos como: E se o que está sendo criado não causar interesse mercadológico? Estamos construindo algo que ninguém quer? E do que vale construir algo dentro do prazo e do orçamento se o produto final é desinteressante para os clientes?

Por isso, salienta Ries (2011), é essencial entender que a maior dificuldade não é desenvolver o produto, pois, com nossa tecnologia atual, quase tudo pode ser desenvolvido e as perguntas a serem respondidas são: “Esse produto deve ser desenvolvido? É possível torná-lo um negócio sustentável em torno deste conjunto de produtos e serviço?”

Para responder essas perguntas é preciso um método científico que permita decompor sistemicamente um plano de negócios, incluindo marketing, nas suas partes componentes e testar cada parte de forma em uma abordagem experimental projetada para alcançar aprendizagem, Ries (2011).

Num primeiro momento, Blank e Dorf (2014) e Ries (2011) entendem que a startup possui apenas hipóteses baseadas em atos de fé, que são os elementos mais arriscados; é aquela primeira versão do produto e com um modelo de negócio que supõe as necessidades dos futuros clientes.

O primeiro passo é decompor uma grande visão em partes e desenvolver um experimento que teste o plano de negócios e as estratégias de forma empírica, projetados para gerar aprendizagem e concluir quais partes são brilhantes e quais são absurdas. O resultado deste primeiro experimento gera suposições que Ries (2011) considera muito importantes:

**Hipótese de valor.** Testa se o produto ou o serviço realmente fornece valor aos clientes no momento em que estão sendo utilizados;

**Hipótese de crescimento.** Testa como os novos clientes dos adotantes iniciais descobrirão um produto ou serviço. Os primeiros participantes espalhariam a notícia aos outros para um crescimento viral? A intenção de construir um experimento é encontrar adotantes iniciais, que são os que sentem a necessidade pelo produto; estes clientes são mais tolerantes com erros e ávidos em dar *feedback*. É o momento de encontrar *feedback* qualitativos sobre como melhorar o programa. E mesmo que os resultados sejam negativos, estes são instrutivos e podem influenciar na estratégia.

No modelo do *lean startup* criado por Ries (2011), o processo de experimento é o primeiro produto que, se for bem-sucedido, permitirá ao gerente desencadear uma campanha adotando mais clientes iniciais a cada nova iteração, começando a construir um produto que quando estiver pronto para ser distribuído de um modo amplo, já terá estabelecido clientes e solucionado problemas reais, além de fornecer especificações detalhadas para o que precisa ainda ser desenvolvido.

#### 3.2.4.1 Modelo de negócio – CANVAS

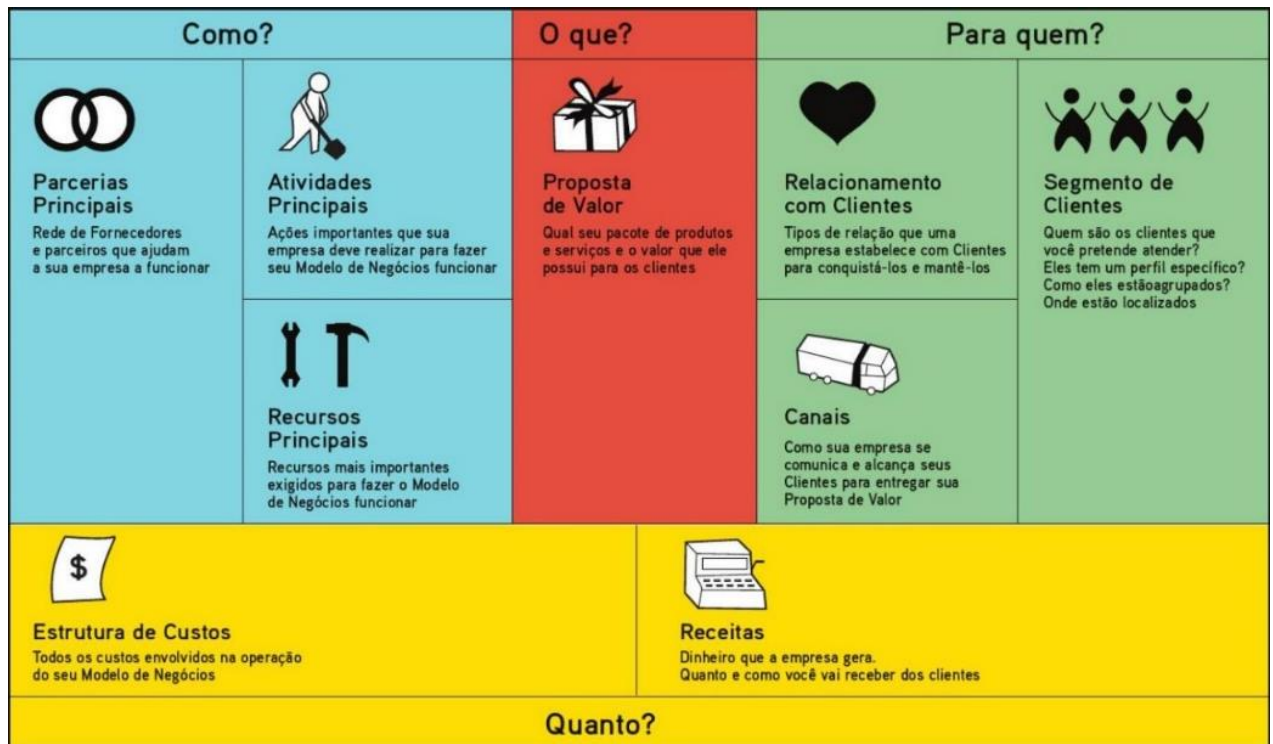
Osterwalder (2010) apresenta o modelo de negócio como sendo uma estratégia a ser implementada através de uma estrutura organizacional, processos e sistemas. Apresenta uma visão panorâmica de como as organizações criam, entregam e capturam valor. Este modelo é uma forma de descrever e manipular o modelo de negócio em uma linguagem de fácil entendimento, para criar novas estratégias de maneira dinâmica, Ries (2011) e Blank e Dorf (2014) concordam com ele.

Sob a perspectiva de Junior (2013), o modelo de negócio deve ser essencial, pragmático, curto e por isso, defende o uso de *post its*, pois eles restringem o espaço disponível para a escrita. Afirma também que deve-se escrever apenas o essencial de forma que fique fácil relacionar os elementos no *Canvas*.

Para alimentar o *Canvas*, que é o espaço em que é possível criar um protótipo do modelo mental, Junior (2013) e Ries (2011) defendem a realização de um *brainstorm*, ou seja, a reunião com pessoas de perfis distintos com a finalidade de gerar soluções inovadoras, com e sem domínio do negócio, pois, os mais experientes devem mitigar riscos e trazer sob domínio os cenários possíveis, enquanto os que não possuem tanto domínio, devem contribuir com inovação e ousadia. O ideal é que respondam as perguntas: Por que? O que? Quem? Como? Quando e Quanto? Pois são questões fundamentais e que, após respondidas permitem que qualquer um entenda o modelo de negócio.

O *Canvas* está distribuído em nove componentes de negócio idealizados por Alexander Osterwalder (2010), como mostra a figura 6.

Figura 6 – Plano de modelo de negócios - *Canvas*



Fonte: <http://www.mrafaelbatista.com.br/blog/?p=171>

A ideia é utilizar o modelo criado como um guia localizador de onde e como realizar as reformulações e apresentar como a empresa pretende ganhar dinheiro:

- 1- **Segmento de clientes** – Trata da escolha do segmento de mercado. Qual segmento de mercado servir e qual ignorar, com um nicho de clientes bem definido, pois, um negócio deve ser desenvolvido a partir da perspectiva do cliente, focando nas necessidades específicas deste grupo de clientes.
- 2- **Proposta de valor** – motivo pelo qual os clientes escolhem uma empresa. Qual problema do cliente está sendo solucionado? Qual a necessidade está sendo satisfeita? São os benefícios que uma empresa oferece para o segmento de clientes escolhido.
- 3- **Canais** – como uma empresa alcança seus clientes e quais os meios de comunicação adotados para entregar a proposta de valor. São os pontos de contato com o segmento de clientes, e devem promover o conhecimento dos produtos e serviços oferecidos pela empresa, levando valor e fornecendo suporte ao cliente após a compra.

- 4- **Relacionamento com Cliente** – tipos de relações que são estabelecidas com o segmento dos clientes, que variam de automatizadas a pessoais. Deve-se entender o foco deste relacionamento, se a ideia é conquistar, reter ou ampliar o número de clientes.
- 5- **Receitas** – o lucro gerado a partir dos segmentos selecionados e deve responder à questão de qual valor o segmento de clientes está disposto a pagar e de que maneira. Deve envolver também a seleção de fontes diferentes de receita.
- 6- **Recursos principais** – recursos cruciais para fazer o negócio funcionar e que garantam e sustentem a proposta de valor, o relacionamento com o cliente, alcance nos mercados e obtenção de receitas.
- 7- **Atividades Principais** – ações que uma empresa deve realizar para garantir que o negócio funcione, por exemplo: desenvolvimento de *software*, cadeia de fornecimento, resolução de problemas, etc.
- 8- **Parcerias Principais** – alianças que otimizem seu modelo de negócio, que podem ou não ser concorrentes, compradores, fornecedores, *joint ventures*.
- 9- **Estrutura de custos** – custos mais importantes para garantir a operação do modelo de negócio. A razão deste item ser o último a ser preenchido reside no fato de que o preenchimento dos demais, facilita e norteia o preenchimento deste item.

#### 3.2.4.2 MVP – Produto Mínimo Viável

*Lean startup*, criada por Ries (2011), propõe a construção de protótipos rápidos que incorporem a ideia principal e que representem as características imaginadas pelos fundadores, projetadas para validar algumas suposições de mercado.

Ries (2011), Blank e Dorf (2014) e Massari (2014) defendem a ideia de construir um Produto Mínimo Viável (MVP) que seja aceitável mercadologicamente, mas evitando o “simplesmente faça”. O ideal é que este produto seja criado através de iterações e incrementos resultantes de métodos ágeis e composto pela menor e mais relevante quantidade de recursos possível do que for considerado indispensável.

Um produto mínimo viável (MVP) é aquela versão do produto que permite a volta completa no ciclo de *feedback*, afirma Ries (2011), resultando num produto composto por funcionalidades essenciais e fundamentais, apenas construído com o mínimo de esforço possível e dentro de um curto tempo de desenvolvimento. Existem alguns tipos de MVP cada

um possui uma finalidade e uma ação a ser tomada em momento específico, conforme ilustrado na figura 7.

Figura 7 – Desenvolvimento do MVP para produtos digitais

Fase	Página	Ação	Objetivo
Prepare-se para Engajar o Cliente	200	-Construa MVP de Baixa fidelidade -Engaje os clientes remetendo baixo tráfego para o MVP.	Veja se a visão da necessidade/problema coincide com a dos clientes e o quanto essa questão é importante para eles.
Teste do Problema através do MVP de Baixa Fidelidade	211	-Aumente gradualmente o número de convites para o MVP. Estude mais de perto o comportamento deles quando chegam e avalie suas atividades. -Mantenha encontros pessoais com os clientes. -Considere se pode haver aumento de escala.	-Entenda o problema/necessidade que está solucionando e como explicá-lo. -Os clientes se importam?
Teste do MVP de Alta Fidelidade	237	-Abra as portas e convide mais clientes a participar -Observe a velocidade de ativação dos clientes.	-Determine se os clientes irão engajar-se, comprar o produto ou usar o site ou app. (teste da solução) -Descubra muitos earlyvangelists apaixonados e entusiasmados que acreditam que o produto resolveu seus problemas.
Otimize a atração de mais clientes	380	Deixe o "sinal aberto" para iniciar a corrida para atrair clientes.	-Otimize a estratégia de atração de clientes.

Fonte: Steve Blank e Bob Dorf (2014)

Baseado no manual da *startup* apresentado por Blank e Dorf (2014), a ideia primordial de construir um MVP de baixa fidelidade seria criar uma hipótese de geração de valor para o negócio e colocá-la em prática o mais rápido possível, para entender se a proposta gera o retorno esperado, e Ries (2011) complementa afirmando servir para avaliar se foi identificado com precisão o problema que incomoda o cliente.

Para que isso seja possível, é preciso identificar a visão que a empresa tem do produto a curto, médio e longo prazos, mapeando quais são as melhorias que os *stakeholders* gostariam de implementar para alcançar os objetivos. Ries (2011) afirma que também é preciso identificar quais recursos e benefícios que são gerados através da solução criada.

Blank e Dorf (2014) afirmam que um MVP com alta fidelidade verifica se o produto está no caminho correto de solucionar o problema do cliente e juntos, tendem a estabilizar a relação mercado versus produto.

Um MVP testa o interesse do cliente e permite que seja possível rearticular e reformular as estratégias sempre que necessário e construir assim, um modelo de negócio satisfatório, estabelecendo *baseline* a cada iteração e refinando as soluções consecutivamente.

Tanto Ries (2011) quanto Blank e Dorf (2014), defendem que as principais vantagens de construir um MVP é evitar o desperdício de horas de engenharia e desenvolvimento para colocar o produto na mão dos clientes visionários o mais rápido possível, para aprender o máximo possível num curto espaço de tempo, o que elimina todo recurso, processo ou esforço que não contribui diretamente para aprendizagem. “...se pegarmos um caminho errado, teremos ferramentas necessárias para descobrir outro caminho” (RIES, 2011, p.165).

Mas um ponto de atenção é que o MVP não é um protótipo e nem é projetado necessariamente para responder perguntas técnicas ou de design, e sim validar hipóteses fundamentais do negócio.

Ries (2011) demonstra que os MVP variam em complexidade e em caso de dúvida, defende a construção de algo simples. Desde testes ou um pouco mais que um anúncio até protótipos iniciais reais, inclusive com problemas e recursos ausentes, o MVP construído tem que ter qualidade, principalmente se houver algum motivo crítico e ou de riscos altos (que podem causar algum tipo de lesão ou acidentes, por exemplo).

#### **MVP – Concierge**

Ries (2011) defende a construção de um MVP–*Concierge*, ou seja um serviço personalizado para clientes, incluindo visitas do presidente da empresa que atende pessoalmente os clientes. Essa estratégia é elaborada como forma de aprendizagem para testar suposições contribuindo para construção de um modelo de crescimento.

#### **MVP – Mocado**

Ou ainda construir o que Ries (2001) chama de MVP–*Mocado*, um MVP que tem resultados forçados e padrões. Ries (2011) propõe, que esse tipo de MVP pode ser utilizado quando a solução requer uma elaboração muito complexa e logicamente desafiadora (se produzir uma solução em pequena escala, pode testar estas hipóteses).

### **3.2.4.3 Pivot**

Estabelecer uma *baseline* é importante para se entender o real cenário atual e baseado nele, é possível tomar decisões de negócios.

Após estabelecer a *baseline*, Ries (2011) fomenta que a *startup* pode dar o próximo passo que é a regulagem do motor, sendo que, todas essas regulagens devem refletir na força motora

de crescimento. Ou seja, deve-se estabelecer um objetivo e entender a situação atual, entender o que falta para chegar neste objetivo e fazer os ajustes necessários para alcançá-lo. Segundo Ries (2011), toda *startup* tem um ideal em comum que é criar um empreendimento próspero e que consiga mudar o mundo. Para alcançar essa visão criam-se estratégias que inclui um modelo de negócio, plano de produto, *benchmarking*, projeção e prospecção dos clientes.

O produto é o resultado final da estratégia e o momento de mudá-la é chamado de *pivot*. Por meio do processo de condução é possível gerar um *pivot* ou perseverar; decisão tomada a partir dos *feedback* dos clientes.

Um produto é criado por constantes otimizações com a inclusão ou retirada de recursos e funcionalidades. Ries (2011) e Blank e Dorf (2014) defendem que a cada nova versão criada não se acrescente muitas funcionalidades de uma única vez e que é necessário entender quais são as prioridades no momento e modificar o produto para atender essas prioridades e somente modificá-lo novamente após as alterações realizadas serem testadas e validadas com clientes. Essas modificações são resultantes dos ciclos de *feedback* que são os ajustes realizados para melhor solucionar o problema do cliente.

As modificações estratégicas são definidas por Ries (2011) como “ajustes no motor” e ocorrem com uma menor frequência, são os chamados *pivot*. É uma mudança um pouco mais consistente recorrente de modificações no plano de negócio e não apenas otimizações. No entanto, a visão dominante dificilmente muda, pois, é a essência do negócio.

Figura 8 –Pirâmide para gerar um *pivot*



Fonte: O autor

Ries (2011) defende que tomar a decisão de gerar um *pivot* ou perseverar é difícil, pois envolve o emocional das pessoas envolvidas no projeto e requer participação da equipe de desenvolvimento de produtos como da liderança comercial e de conselheiros externos. A equipe de desenvolvimento deve apresentar para o restante, quais foram as iniciativas tomadas para otimização do produto e quais foram os resultados consequentes dessas medidas, e se atingiram a expectativa criada e se não atingiram o porquê, hipoteticamente.

Enquanto que a liderança comercial deve apresentar os *feedback* dos clientes e os detalhes de todas as suas conversas com potenciais clientes e com os concorrentes, os conselheiros externos servem para demonstrar uma visão de quem está de fora dos problemas apresentados.

A tabela abaixo é baseada na visão abordada por Ries (2011) e representa os principais tipos:

Tabela 1 - Tipos de *pivot*

<b>Tipos de <i>pivot</i></b>	<b>Descrição</b>
<b>Zoom-in</b>	No planejamento havia uma gama de recursos que compunham o produto como um todo, porém ocorre quando apenas um recurso isolado torna-se o produto como um todo
<b>Zoom-out</b>	Este é o caso da situação inversa, onde um produto completo, torna apenas um recurso determinado de uma gama maior de recursos.
<b>Segmento de cliente</b>	A empresa conseguiu solucionar o problema real estipulado no modelo de negócios, porém o público alvo, é diferente do esperado.
<b>Necessidade do cliente</b>	As vezes no meio do percalço, logo percebe-se que o problema que está tentando solucionar não é tão importante mercadologicamente, porém por ser um perfil claro de cliente percebe-se que há problemas mais importantes a serem resolvidos de um produto completamente diferente
<b>Plataforma</b>	De maneira geral, cria-se uma nova plataforma para o um produto matador ou um <i>killer app</i> , que são produtos atrativamente baratos que alcançam a compra de um mais caro (BLANK; DORF, 2014) porém, grande parte dos casos terceiros acabam entendendo que também podem tirar proveito desta plataforma e então, participam do desenvolvimento deste novo <i>pivot</i> .

<b>Arquitetura de negócios</b>	<p>Conceito de Geoffrey Moore (1991), geralmente composto de duas arquiteturas principais e os <i>pivot</i> podem mudar de arquitetura.</p> <p>B2B (<i>Business to Business</i>) – uma arquitetura de alta margem e baixo volume (modelo de sistemas complexos).</p> <p>B2C (<i>Business to Commerce</i>) – baixa margem e alto volume. Ex.: <i>Appliance</i> Google</p>
<b>Captura de valor</b>	Modelos de monetização ou receita. Pode haver frequentemente mudanças de forma que as empresas captam valor que variam de alcance do negócio, produto e estratégias de marketing.
<b>Motor de crescimento</b>	De maneira geral, a empresa muda sua estratégia para obter um crescimento mais rápido ou mais lucrativo. E pode ser composta de três tipos: viral, recorrente e pago.
<b>Canal</b>	Mecanismos em que os produtos estão disponíveis ao cliente que pode ser um canal de vendas ou de distinção, pois um ano pode obter sucesso com um ou outro.
<b>Tecnologia</b>	Maneira de chegar a mesma solução utilizando tecnologias diferentes, daí pode avaliar qual destas traz maior rentabilidade e ou desempenho. Pode se tratar de uma inovação de sustentação ou uma melhoria incremental.

Fonte: o autor

Segundo Ries (2011), cada tipo de *pivot* se tem que estar correlacionado com uma pensamento estratégico e com seu público alvo e devem ser utilizados para alavancar a estratégia adotada e tomar um certo cuidado com as analogias feitas com cases parecidos, pois, é difícil ter uma fórmula correta quando se trata de inovação; é necessário avaliar se o que funcionou para uma determinada área, empresa, produto ou setor, também serve para alcançar os objetivos traçados e se existir essas hipóteses, o ideal é que seja testada por um novo produto mínimo viável, com mudanças específicas e bem projetadas, e caso seja adotado o caminho errado é necessário agilidade para descobrir um novo.

#### 3.2.4.4 Medir

Seguindo a diretriz de Ries (2011), o conceito de entender que uma *startup* está evoluindo e quanto está sendo produtivo esse crescimento, significa medir a aprendizagem validada e obtida através dos esforços, o que não reflete necessariamente na quantidade de trabalho executado ou na quantidade de entregas produzidas. Produtividade é descobrir qual o modo

sistemático das coisas certas para desenvolver. E para conseguir medir produtividade é necessário que a startup entenda como ela está agora e estabeleça uma *baseline*, complementa Ries (2011), confrontando-a com as verdades reveladas pela avaliação feita pelos clientes e em seguida, descobrir qual é o ideal esperado e qual o planejamento estratégico para se aproximar do ideal refletido nos planos de negócios.

Clientes reais precisam dizer à *startup* se utilizariam os produtos criados e porquê. Ries (2011) salienta que entrevistas e acompanhamento de comportamentos dos clientes são muito importantes neste momento de concepção, e igualmente através dos resultados, construir novos experimentos que conduzam às métricas a serem utilizadas.

#### 3.2.4.4.1 Contabilidade para inovação

Segundo Ries (2011) e Blank e Dorf (2014), contabilizar os resultados obtidos é a melhor forma de avaliar se a *startup* está aprendendo a desenvolver um negócio sustentável. A contabilidade para inovação tem início no processo de converter os atos de fé em um modelo financeiro quantitativo e funciona em três passos.

Ries (2011) e Blank e Dorf (2014) concordam que primeiramente, deve-se construir um MVP para estabelecer dados reais referentes a *baseline* para entender o cenário atual e entender o quanto falta para atingir o objetivo, assim como, com o passar do tempo, medir o progresso.

Depois, completa Ries (2011), é importante regular o direcionamento da estratégia da *baseline*, entendendo se ela está se desenvolvendo na direção correta e caso não esteja, realizar todos os ajustes para alcançar o ponto de decisão, porém, isso pode exigir muitas tentativas.

E por último, Ries (2011) complementa, criar um *pivot* ou perseverar, medir o progresso alcançado e decidir se a empresa está no rumo certo e se faz sentido continuar. Senão a equipe gerencial deve evoluir a estratégia e construir uma nova *baseline* com todos os ajustes necessários.

Durante este processo é possível obter dados que ajudam a medir o progresso que deve ser contabilizado e apresentado em relatórios para que todos da equipe tenham o mesmo entendimento. Mas, estes relatórios devem ser acionáveis, acessíveis e auditáveis para que sejam úteis, defende Ries (2011).

**Acionável:** significa que devem demonstrar causa e efeito evitando métricas de vaidade, que são novos clientes resultantes de campanhas de *marketing* e de relações públicas. Ries (2011), salienta a não utilizar os resultados de destas campanhas na fase inicial, pois o produto ainda não está mercadologicamente maduro e utilizar os esforços de *marketing* para aumentar

os números gera uma força de tração ilusória, que dura pouco tempo e traz uma ideia errônea de sucesso.

Ries (2011), defende que é necessário estabelecer o que é considerado uma unidade de medição para conseguir que todos os envolvidos tenham a mesma percepção e proporção dos elementos que propõe o projeto e para aumentar a clareza e assertividade, ou seja, caso seja um site, o que é considerado uma visita? “Quando a causa e o efeito são entendidos com clareza, as pessoas são capazes de aprender a partir de ações” (RIES, 2011, p.144).

**Acessíveis:** os relatórios precisam ser entendidos por funcionários e gerentes; devem estar em um repositório em que todos os interessados tenham acesso e ser utilizado para embasar uma tomada de decisão, de forma que represente *feedback* autênticos. Ries (2011) orienta a utilização de relatórios de corte para representar esses *insights*, este relatório ajuda para identificar e analisar os resultados de um grupo de pessoas que experimentam o mesmo evento em um determinado intervalo de tempo.

**Auditados:** os dados precisam ser confiáveis e auditados pelos gerentes que devem conferir amostragens de dados reais com clientes, pois, esses dados geralmente são construídos por pessoas que não dominam o negócio ou por sistemas intermediários.

### 3.2.5 Aprender

Ainda a hipótese de estar-se construindo algo que ninguém queira. Segundo Ries (2011), o resultado consequentemente seria apenas as lições aprendidas e os prejuízos de um trabalho duro, porém, aprendizagem é uma ótima oportunidade quando se trata de inovação. Oportunidade de aprender e entender o que os clientes realmente querem e não apenas o que se supôs que eles quisessem. No *Lean Startup* isso se chama aprendizagem validada e esta é usada como medida de progresso e eliminação do desperdício.

“Startups existem não apenas para fabricar coisas, ganhar dinheiro ou mesmo atender clientes. Elas existem para aprender a desenvolver um negócio sustentável. Essa aprendizagem pode ser validada cientificamente por meio de experimentos frequentes que permitem aos empreendedores testar cada elemento da sua visão” (RIES, 2011, p.8)

“Aprendizagem validada é o processo de demonstrar empiricamente que uma equipe descobriu verdades valiosas acerca de perspectivas de negócio presentes e futuros de uma *startup*” (RIES, 2011, p.34).

Quando se realiza entrevistas com os clientes reais, geralmente comete-se um grande erro ao fazer perguntas diretas do que esperam do produto; o de achar que os clientes sabem o que

querem. O correto é analisar o comportamento e entender se as sugestões dadas são mesmo úteis, trabalhando com inteligência alinhada as necessidades reais e não apenas suposições estratégicas sem embasamento empírico.

### 3.2.5.1 Ciclo de *Feedback* – Construir-Medir-Aprender

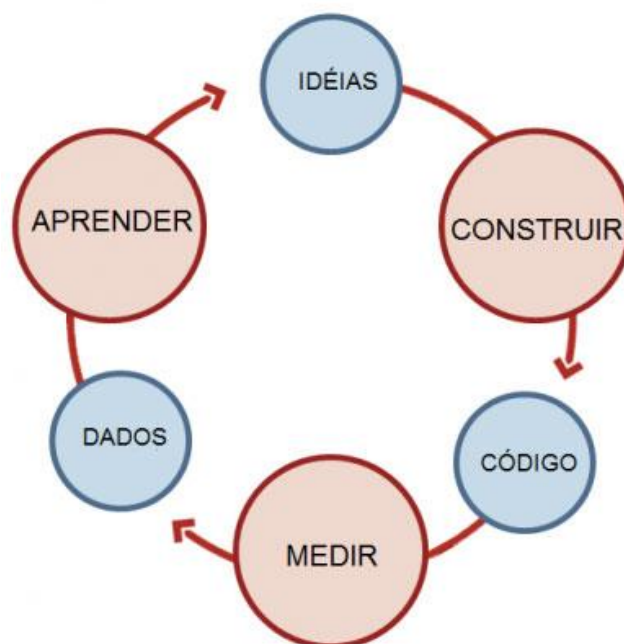
O método de *Lean Startup* criado por Ries (2011) é planejado de forma cíclica; é um ciclo de vida baseado em *feedback* para construir-medir-aprender e em contraste com o modelo tradicional em cascata, onde tudo é projetado nos mínimos detalhes e sofrem poucas alterações, no ciclo de *feedback* se tem inúmeras hipóteses que são testadas a cada iteração.

Segundo Ries (2011), O ciclo funciona com o surgimento de atos de fé, da construção de um modelo básico de negócios e a construção um produto mínimo viável disponibilizando-o o mais rápido possível para os clientes, validando se a estratégia pensada está correta, diminuindo as estratégias que são falhas, e iterando o ciclo a partir do *feedback* recebido.

As *startups* transformam as ideias em produtos, e estes são apenas experiências e a aprendizagem de como desenvolver uma empresa sustentável é o melhor resultado desses experimentos.

“...os produtos que uma startup desenvolve são experimentos; a aprendizagem sobre como desenvolver uma empresa sustentável é o resultado destes experimentos” (RIES, 2011, p.69).

Figura 9 – Ciclo de *feedback*



Fonte: Eric Ries (2011)

O principal objetivo de execução deste ciclo é concentrar as energias para minimizar o tempo total de execução completo do ciclo, sem que se perca em qualidade. Lembrando sempre que de nada adianta construir algo que ninguém quer dentro do prazo e do orçamento.

O método recomendado por Ries (2011) para essa medição chama-se contabilidade para inovação, e permite observar através dos dados gerados pela iteração anterior, se os esforços de ajustes do motor estão dando resultados.

Por fim, há um *pivot*, e a partir deste é preciso decidir se irá gerar uma nova estratégia ou perseverar com a mesma. Se perceber que a hipótese gerada é falsa, será o momento de realizar mudanças importantes de rumo a uma nova estratégia.

Ries (2011) defende que cada iteração deste ciclo é uma tentativa de construir um motor de crescimento e em validar as hipóteses de valor, dando origens a variáveis de ajustes que controlam o motor, que seriam os ajustes das estratégias. E assim que este motor estiver funcionando, ou seja, assim que for construído um modelo de negócio sustentável, deve-se investir em alavancar para marchas maiores, com fortes investimentos em *marketing* e relações públicas.

### 3.2.6 *Devops*

Segundo Ries (2011), com o uso de *Devops* é possível criar um sistema imunológico com proteções automáticas e para a metodologia *lean* é o mundo ideal, pois, diminui os custos com operações e desenvolvimento, diminui o desperdício e aumenta a qualidade das mudanças realizadas permitindo a construção de um sistema complexo e robusto.

A ideia é criar um sistema imunológico com proteções automáticas e monitoramento contínuo, através do qual é possível verificar se o produto se comporta de acordo com o previsto, caso contrário, o erro ou a mudança é removida rapidamente, garantindo a integridade do negócio.

Para Sato (2014), existem três termos importantes e complementares em *devops*: *continuous integration* ou integração contínua, *continuous delivery* ou entrega contínua, *continuous deployment* ou implantação contínua.

Sato (2014), afirma que atrasar o *deploy* para produção, significa atrasar a competitividade e adaptabilidade mercadológica, por isso grandes empresas realizam vários *deploy* por dia, também chamado de entrega contínua ou *continuous delivery*. Realizar um *deploy* é equivalente a implantação de um software em português, que para ser realizado necessita de um provisionamento, e a melhor forma de realizar as etapas é com a automatização.

“O *build* de um sistema envolve todas as tarefas necessárias para conseguir executá-lo, como por exemplo: compilação, *download*, e resolução de dependências, vinculação com bibliotecas, empacotamento, cálculo de métricas de qualidade, etc.” (SATO, 2014, p. 132)

Para Ries (2011), a implantação contínua ou *continuous deployment* é transformar a implantação em ambiente produtivo em um ato trivial sendo automático e ocorrendo com frequência.

Integração contínua ou *continuous integration* para Sato (2014), é o processo de unir uma nova versão produzida com a versão anterior, controlando as alterações realizadas e garantindo a integridade do código fonte produzido, com a ajuda de um controlador de versões.

Os códigos fontes produzidos anteriormente, ficam armazenados em um repositório central denominados *branch/trunk*, e são muito utilizados para gestão destes artefatos. A *trunk* é o repositório principal e geralmente tem acesso restrito a pessoas com o papel de integrador, enquanto o *branch* é comumente utilizado para fazer rascunhos e testar as alterações que podem ser guardadas ou jogadas fora quando necessário e depois que estas alterações são realizadas, testadas e validadas elas são integradas com *trunk*.

Esta técnica requer versionamento, testes unitários automatizados, testes de integração, uma boa comunicação e coordenação entre os membros, devendo evitar o acúmulo de alterações locais em que as mudanças não estejam visíveis pelo restante da equipe.

A integração contínua surgiu com a metodologia de desenvolvimento ágil *Extreme Programming* ou XP, que postula que as alterações realizadas sejam construídas, integradas e testadas rapidamente, e o time precisa ser maduro e disciplinado para executar este trabalho.

“Integração contínua é uma das práticas originais de XP que encoraja desenvolvedores a integrar seu trabalho frequentemente para que seja possível que problemas sejam detectados e corrigidos rapidamente. Assim que um desenvolvedor termina uma tarefa cujo código pode ser compartilhado com a equipe, ele precisa seguir um processo disciplinado para garantir que suas mudanças não vão introduzir problemas e que elas vão funcionar corretamente com o resto do código” (SATO, 2014, p. 140).

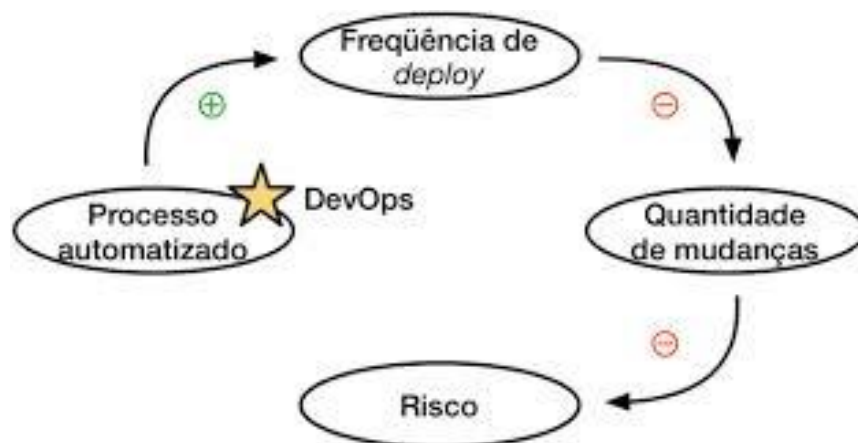
Sob a perspectiva de Sato (2014), realizar um *deploy* significa levar os códigos construídos para a produção, sejam eles do ambiente de desenvolvimento, testes ou homologação, realizando um processo cíclico e vicioso de desenvolvimento e operações, onde cada equipe tenta defender seus interesses. No ambiente de desenvolvimento de software, são

realizadas as integrações das códigos fontes para a criação dos softwares, sendo neste momento que o desenvolvedor realiza os testes para verificar se a aplicação funciona de acordo que o que havia sido especificado pela equipe de negócios, enquanto que no ambiente de testes são realizados os testes de execução da aplicação e realizados as validações de integração com as aplicações já existentes e se todos os requisitos foram atendidos e por último, mas não menos importante, é realizada a publicação desta versão em um ambiente de homologação, onde o representante do cliente ou a equipe de negócios, realiza seus testes com a estratégia de usabilidade, garantindo que o que foi codificado corresponde ao esperado e atende as necessidades do usuário.

O time de desenvolvimento tenta implantar novas mudanças e operações, a título de tentar minimizar as alterações sistêmicas e mitigar o impacto dos riscos no ambiente produtivo. *Devops* tem por finalidade modificar essa cultura e transformar estes conflitos em colaboração, entrega contínua e automação de processos.

Sato (2014) defende que a entrega contínua diminui a quantidade de mudanças em cada *deploy*, por isso facilita o processo de identificação de qualquer erro causado em ambiente produtivo e em utilizar processos automatizados de *deploy* que minimizam a quantidade de erros humanos, tornando a execução mais confiável e dando autonomia para ser realizada a qualquer momento.

Figura 10 – *Devops* e automação de processos no ciclo vicioso de desenvolvimento versus operações



Fonte: Danilo Sato (2014)

A ideia é criar uma cultura colaborativa entre as pessoas que devem atuar como membros de uma mesma equipe e com habilidades complementares.

“Seu objetivo é criar uma cultura de colaboração entre as equipes de desenvolvimento e de operações permitindo aumentar o fluxo de trabalho completo - maior frequência de *deploy* - ao mesmo tempo aumentar a estabilidade e a robustez do ambiente de produção” (SATO, 2014, p. 6).

Além de uma mudança cultural, *Devops* promove a automação de testes, de processos e de outras atividades, com a finalidade de transformar a Tecnologia da Informação (TI) em um agente de capacitação de negócios.

Sato (2014) recomenda que o ambiente de desenvolvimento se aproxime da produção, para que seja possível evitar conflitos e também defende que o ciclo de vida do software se inicia quando o usuário começa a utilizá-lo, postulando também que os servidores que utilizam máquinas virtuais e ferramentas de mercado facilitam o trabalho de gestão dos ambientes. Salienta, que é necessário realizar monitoração para garantir qualidade, e que esta atividade também deve ser automática, a tal ponto que avisa os administradores caso encontre algum problema em ambiente produtivo. Mas, por outro lado, o time de desenvolvimento deve ajudar, utilizando testes automatizados e *Test-Driven Development* (TDD) ou Desenvolvimento Dirigido à Testes com verificações granulares para identificar eventuais falhas mais rapidamente.

Ries (2011) também defende o uso de ferramentas para executar todos os testes de integração e gerar um pacote para integração a cada modificação, processo esse também chamado de *commit*.

Beck (2009) afirma que o TDD é basicamente adicionar um teste rapidamente ao código e verificar se ele funciona, caso não, é preciso acrescentar mudanças e executar todos os testes novamente e se estiver correto, apenas será preciso modificar a estrutura interna do código sem alterar o seu comportamento para remover duplicações.

A cada mudança gerada deve haver uma cobertura de testes ou *test coverage* para garantir a qualidade e a integridade das correções. E todas as integrações devem ficar no mesmo repositório de código (*trunk*) e devem haver verificações para garantir a integridade sistêmica. E quando é gerado um *build* mal sucedido, os desenvolvedores devem voltar sua atenção para a correção da mesma forma que ocorre com a corda *andon*, defende Ries (2011).

Sato (2014) afirma que a medida que o sistema vai sofrendo modificações e ficando mais complexo, o processo do *build* fica mais complicado, por isso é muito importante investir em automação, e incluir neste processo os testes automatizados, para garantir que o sistema irá funcionar corretamente, já que este pode conter defeitos ou até funcionalidades incompletas. Isso garante que as mudanças funcionem e que podem ser integradas ao restante do código.

É necessário investir em automação e padronização para diminuir a possibilidade de erros humanos, enfatiza Sato (2014), explicando que ambientes devem ser reinstalados por processos automatizados de provisionamento, configuração e *deploy*, utilizando versionamento de códigos fontes, escrevendo códigos limpos, realizando testes, empacotamento e distribuindo responsabilidades em módulos para diminuir o acoplamento.

“Provisionamento é comumente utilizado por empresas de telecomunicações e por equipes de operações para se referir as etapas de preparação inicial de configurações de um novo recurso” (SATO, 2014, p. 63). Segundo o autor, tais processos devem variar de acordo com as necessidades de cada empresa, mas em termos gerais, representam todas as atividades necessárias para que um servidor possa ser utilizado independentemente da finalidade.

Uma solução muito utilizada neste processo é o *cloud* ou computação em nuvem, que é a utilização de hardware virtualizado e que consiste em executar diversos sistemas operacionais em um mesmo computador, essa técnica permite restaurar ambientes em minutos, afirma Sato (2014).

Outro importante recurso promovido por essas ferramentas é a possibilidade de execução automática de *scripts*, executados a partir de um conjunto de diretivas para gerenciamento de configurações, que promove ganho de qualidade e performance com a automatização dos processos.

Em conclusão, *devops* é aplicar um método científico para melhoria contínua com a finalidade de evoluir o processo de entrega de software e modificá-lo para que seja mais eficiente.

### 3.2.7 Crescer

Segundo Ries (2011), depois da construção de um modelo maduro de sistema, onde o modelo de negócios está estabelecido e já se teve uma série de usuários iniciais, o próximo passo é construir um “motor de crescimento” ou apenas, alavancar o crescimento do produto tornando-o um produto comercial e uma *startup* sustentável. Sustentável, com a interpretação de conseguir se sustentar no longo prazo e caracterizar uma regra simples: “Os novos clientes surgem das ações dos clientes passados” (RIES, 2011, p.196).

“O motor de crescimento é o mecanismo que as startups utilizam para alcançar o crescimento sustentável” (RIES, 2011, p.196).

O motor de crescimento tem variáveis de ajustes que seriam as hipóteses de valor e de crescimento que servem para realizar ajustes e controlam o crescimento da *startup*, onde cada

iteração é uma tentativa de funcionar o motor. Um motor que, conforme gira diversas vezes, gera combustão e completa o ciclo cada vez mais rápido impulsionando o crescimento da empresa.

“Essas fontes de crescimento sustentável movimentam ciclos de *feedback* que denominei motores de crescimento” (RIES, 2011, p197).

Funcionar o motor tem a finalidade de alavancar e impulsionar o produto, utilizando práticas de vendas, *marketing* e negócios:

**Boca a boca:** é a representação do entusiasmo dos clientes satisfeitos em utilizar determinado produto, ou seja, as pessoas vão divulgando o produto para seus contatos sociais gerando crescimento.

**Efeito colateral da utilização do produto:** é o status social promovido pelo produto ou tendências da moda, bens de luxo, produtos que promovem a consciência de si mesmos ou prazer sempre que usados.

**Publicidade financiada:** maioria das empresas utilizam esse recurso, onde é investido valores financeiros para gerar a publicidade a ser paga como resultado da receita recorrente, como o capital de investimento.

**Por compra ou uso repetido:** alguns produtos são projetados para serem comprados diversas vezes, como planos de assinaturas, por exemplo.

Os motores de crescimento, segundo Ries (2011), são projetados para promover um conjunto de métricas a serem avaliadas e sob as quais são necessários concentrar energias e gerar grandes experimento que conduzam a aprendizagem validada.

## 4 ESTUDO DE CASO - FACECAR

Foi construído um estudo de caso onde foi aplicada a metodologia do *Lean Startup* de Ries (2011). O objetivo foi construir um software de mercado utilizando os princípios enxutos, construindo também um modelo de negócio maduro pronto para o crescimento e a construção de uma *Startup*.

### 4.1 *Primeiro release*

O Facecar é um aplicativo para segurança automotiva, construído para a plataforma Android com a principal funcionalidade de bloquear os veículos, caso o sistema identifique que o veículo está sendo roubado ou furtado. Funciona como um rastreador inteligente, capaz de identificar a posição do veículo se necessário ou desejado pelo usuário.

O principal objetivo é evoluir para a construção de uma *startup* para segurança automotiva.

Foram executados e descritos no decorrer desta monografia, os ciclos da metodologia Construir-Medir-Aprender do *Lean Startup*.

Em sua concepção, o Facecar foi projetado para ser um software robusto com diversas funcionalidades, sendo algumas de segurança e outras de caráter utilitário, porém, seguindo a metodologia *lean* de Ries (2011), entendeu-se que a função mais importante que o aplicativo podia promover, seria o desligamento do motor e a localização do veículo, sendo este então, o primeiro “ato de fé”.

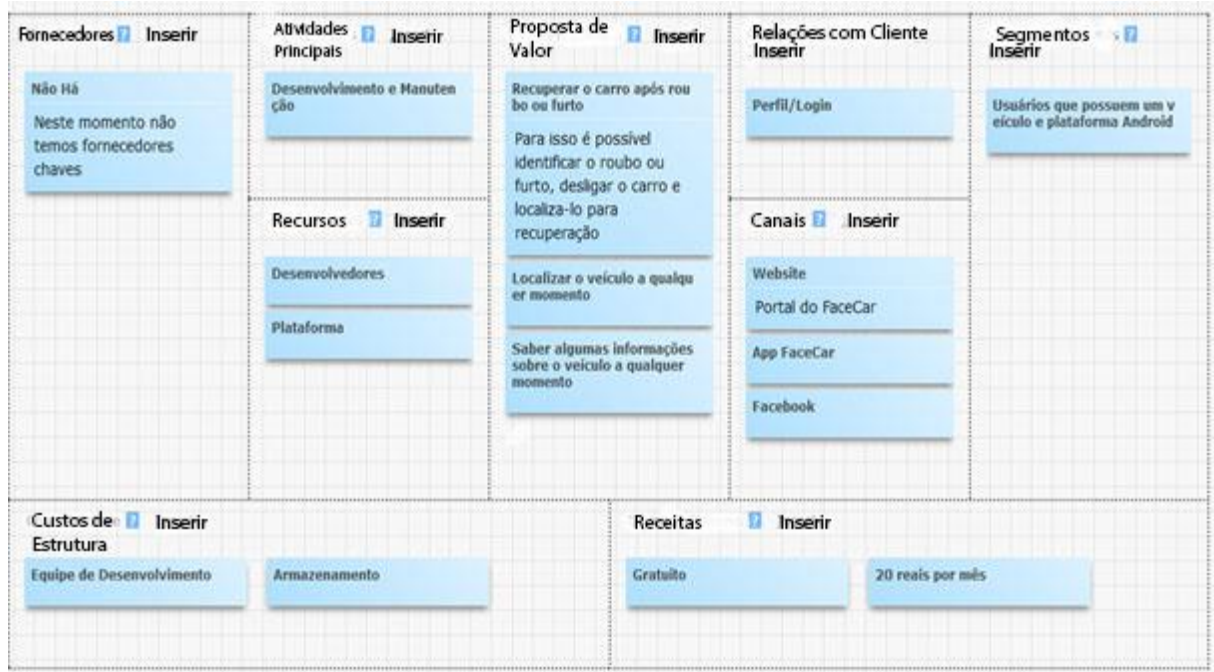
Para eliminar o desperdício e validar apenas o essencial em um primeiro momento, defendidos por Poppendieck (2011) e por Blank e Dorf (2014), foi construído um modelo de negócio baseado nesta hipótese de valor.

Para transcrever o modelo de negócio, foi utilizado o modelo *Canvas*, conforme é possível observar na figura 11, que segue as orientações de Alexander Osterwalder (2010). A aplicação desta etapa permitiu que toda a equipe entendesse melhor o projeto, unificando o foco do trabalho.

Para facilitar as possíveis mudanças do negócio, foi utilizada uma ferramenta online chamada *Canvanizer* que possibilita a criação de modelos de negócios virtuais que podem ser facilmente modificados se houver necessidade. E com a execução desta etapa foi possível identificar qual era a proposta de valor que existia naquele momento, e expor essas informações

aos participantes permitiu aos envolvidos terem um entendimento mais consistente da estratégia que estava sendo tomada.

Figura 11– Primeiro modelo de negócio



Fonte: O Autor

Foi construído um MVP de baixa fidelidade, como orientado por Ries (2011) e Blank e Dorf (2014) e disponibilizado na rede social do *Facebook* para um público selecionado de pessoas. Sendo assim, ficou entendido que, seguindo os princípios de *Lean Thinking* e de *just in time*, apenas as funcionalidades de localizar e possibilitar ao usuário desligar o carro, caso fosse identificado uma possibilidade de furto ou roubo, seriam a essência do produto e que, naquele momento, a ideia era responder se a proposta de valor era válida em disponibilizar o aplicativo ao mercado, o mais rápido possível.

Logo na primeira semana, houve mais de 100 visualizações da publicação feita no *Facebook*, conforme é possível ver na figura 12. As pessoas que gostaram da ideia proposta de valor do aplicativo e contribuíram com diversos comentários, com elogios, dúvidas, ainda quiseram complementar a ideia com solicitações de implementação de funcionalidades que ainda não haviam sido pensadas. Ou seja, foi possível entender que a hipótese proposta era válida, como Ries (2011) sugere, e não somente isso, mas como também foi possível pensar quem seriam os adotantes iniciais do produto proposto, já que trouxe resultados positivos e significativos. Uma curiosidade sobre esses resultados, é que as pessoas se mostraram

receptivas e interessadas em contribuir com suas ideias e ficavam ansiosas para saber se era possível implementar e quanto tempo demoraria para publicar as novas funcionalidades.

Figura 12 – Relatórios do *Facebook* do Facecar

<input type="checkbox"/>	Publicações	Alcance	Curtidas,...	Publicada
<input type="checkbox"/>	 Galera, boa noite ! Fiz um app que vai ...	187	10	4 de agosto de 2015 às... Andressa Andrade
<input type="checkbox"/>	 Olá, seja bem vindo ao FaceCar Conh...	183	8	4 de agosto de 2015 às... Andressa Andrade

Fonte: *Facebook* (2015)

Foi feito um *brainstorm*, conforme orientado por Ries (2011) para que todos da equipe pudessem saber dos resultados e tomarmos a decisão de perseverar ou criar um *pivot*. Por unanimidade, foi decidido manter essa estratégia e construir um MVP para validar a hipóteses dos adotantes iniciais. Outra decisão tomada foi com relação ao produto mínimo viável que, para naquele momento, entendeu-se que o mais adequado seria criar uma MVP *concierge* com tratamento especial dos usuários iniciais, por se tratar de um software de segurança automotiva, seguindo assim o modelo proposto por Ries (2011).

## 4.2 Segundo *release*

Com algumas decisões tomadas, estabeleceu-se então a primeira *baseline*. Também foi decidido investir na construção do aplicativo, com um MVP de fidelidade maior. A proposta era entender se, de todas as pessoas que gostaram da ideia e quais realmente estavam dispostas a se tornar um usuário do produto. A partir disso, construiu-se o aplicativo com as seguintes funcionalidades:

- **Status do carro atual** – é possível verificar se o motor está ligado ou desligado, se as portas estão abertas ou fechadas e algumas informações técnicas como sinais de GPS e GSM.
- **Monitoração do veículo** – foi criado um monitoramento que funciona como o alarme de um carro, ou seja, assim que ligado, monitora a posição do carro e avisa se o veículo se locomoveu indevidamente.

- **Localizar o veículo** – essa funcionalidade permite que o veículo seja localizado em qualquer momento.

- **Desligar o motor** – através do monitoramento do veículo, quando necessário, é exibido um sinal de alerta permitindo que o usuário desligue o carro, se preciso.

O aplicativo foi construído com a ferramenta Android Studio da empresa Google e seguindo padrões de *devops* proposto por Sato (2014) e Ries (2011), tanto para operações quanto para desenvolvimento, visando garantir a entrega contínua, a integração contínua e a implantação contínua.

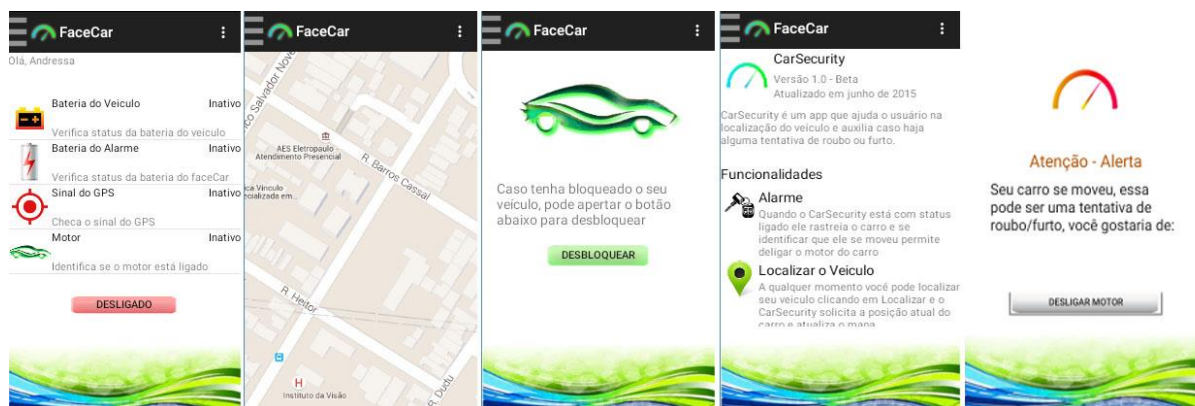
Para a estrutura de operações, foram utilizadas ferramentas de versionamento que neste caso foi o *Git*, o uso de disponibilização e compartilhamento dos arquivos foi utilizado a ferramenta *Hiroku* e para publicação da versão foi utilizado o *software Jenkins*.

Quanto ao desenvolvimento, foi tomada como diretriz a utilização de um código limpo e bem estruturado, respeitando a arquitetura MVC.

“O padrão MVC é composto por três camadas distintas, cada uma com suas responsabilidades, justamente visando separar a lógica do negócio da sua aplicação da interface gráfica, facilitando a manutenção do código, caso seja necessário criar novas versões da interface gráfica, ou também alterar o código, que por sua vez estará centralizado e desvinculado da interface gráfica” (MILANI, 2012, p. 68).

Foram desenvolvidos protótipos e validados com os envolvidos após efetuada a construção e a disponibilização do aplicativo em ambiente produtivo no *Playstore*, um repositório de aplicativos, da empresa Google.

Figura 13 - Protótipos criados para o Facecar



Fonte: O Autor

Logo na primeira versão, percebemos que havia um erro no nome do *build* gerado, que é utilizado para montar o endereço para acessar o software ou a URL (*Uniform Resource Locator*) e os usuários podiam perceber o erro quando efetuavam o *download* do aplicativo. Logo foi decidido gerar uma nova versão corrigindo o erro.


Foi essencial perceber o erro neste momento, pois quando é modificado o nome é preciso efetuar todo o processo novamente, como se fosse criado um novo aplicativo e o histórico se perde, como é possível ver na figura 14, o que influencia diretamente na reputação do aplicativo. Este erro funcionou como na corda *andon* citado no case da Toyota, mencionado por Ries (2011) e Poppendieck (2011), onde foi identificado o problema, e parou-se todo o processo para corrigi-lo rapidamente.

### 4.3 Terceiro release

Nos primeiros 3 meses foram alcançados aproximadamente 20 *downloads* do aplicativo e comentários no *Playstore*. Porém, destes, apenas 3 se propuseram a pagar pelo hardware a ser instalado do veículo para permitir o funcionamento do aplicativo por completo.

Neste terceiro release, um dos clientes teve seu carro furtado e foi possível recuperá-lo em torno de 40 minutos após o incidente.

Figura 14– Relatório sobre *downloads* disponível no Google *Playstore*

NOME DO APP	PREÇO	INSTALAÇÕES ATUAIS/TOTAL DE INSTALAÇÕES	CLASS. MÉDIA / TOTAL	FALHAS E ANRS	ÚLTIMA ATUALIZAÇÃO	STATUS
 FaceCar 1.0	Gratuito	1 / 3	★ 5,00 / 1	—	03/08/2015	Não publ...
 FaceCar 1.1	Gratuito	8 / 26	★ 5,00 / 4	—	04/08/2015	Publicado

Fonte: Google *Playstore*

Assim que o automóvel foi furtado, o usuário entrou em contato com a empresa para ajudar a recuperar o veículo, pois, teve dificuldades de entender como o aplicativo funcionava e como deveria interagir com o mesmo.

Foi dado o devido suporte VIP pelo CCO (*Chief Customer Officer*) da *startup*, conforme recomendado por Ries (2011), que assegurou o uso correto do aplicativo e o sucesso na localização e recuperação do automóvel.

Ao realizar este estudo de caso ficou notório os problemas com a usabilidade do aplicativo, que quando identificados logo foram solucionados, pois, tratava-se de um MVP com o escopo limitado e também, foram identificados os problemas de comunicação, as informações não estavam disponíveis e claras referente ao uso do produto, e embora os usuários iniciais tenham sido receptivos a ideia proposta, não conseguiram entender como utilizar o aplicativo.

## 5 CONSIDERAÇÕES FINAIS

Em conclusão, a análise realizada dos princípios da metodologia de *Lean Startup* e de sua aplicação no desenvolvimento de uma startup, mostrou que a mesma apresenta os pilares essenciais que ajudam a identificar os pontos que precisam ser adequados para tornar o desenvolvimento enxuto e de melhoria contínua com ciclos iterativos e incrementais, o que idealmente resultam em um produto construído junto ao cliente que mediante o entendimento, ajustes e pleno atendimento as suas necessidades, obtém-se um nível alto de satisfação por parte dos usuários, aumentando assim as chances de prosperidade da startup, mesmo que com um público alvo em formação e mediante tantas variáveis e incertezas.

Esta metodologia provê o desenvolvimento para o cliente ou o *Customer Development*, cujo principal foco é entender o que ele precisa e não o que ele acha que quer, construindo um produto inovador, de maneira ágil que, na medida que se desenvolve, amadurece seu modelo de negócio juntamente com o mercado.

O pensamento enxuto gerado a partir do uso desta metodologia, promove a agilidade necessária para a construção e validação de uma hipótese de valor gerada a partir de uma necessidade identificada e de uma solução proposta, tornando possível entender se realmente é uma ideia válida, com a construção e disponibilização em ambiente produtivo de um mínimo produto viável e a identificação de erros e falhas, promovendo um ambiente de baixa complexidade que permite que correções sejam feitas rapidamente alavancando o sucesso da startup.

O *Lean Startup* é composto por uma gama de recursos, práticas e metodologias complementares que trazem resultados muito positivos, diminuindo a complexidade e aumentando a assertividade, tanto com a validação do conceito inicial, quando na evolução do sistema como todo.

## 6 OPORTUNIDADES PARA ESTUDOS FUTUROS

Refletindo sobre as limitações deste trabalho, também surgem a reflexão de oportunidades futuras, que além de um nível mais amplo e embasado deste assunto, como também a execução de mais iterações e melhorias construídas a partir dos *feedback* dos usuários, como também uma pesquisa de campo acompanhada presencialmente, avaliando as dificuldades com usabilidade e um estudo de mercado traçando perfis completos dos usuários até realmente validar o sucesso e a lucratividade do aplicativo como um todo, proposto pelo conceito da metodologia.

Foi possível realizar a evolução de maturidade do estudo de caso, com o ciclo proposto neste trabalho, transformando este aplicativo em uma plataforma robusta e assertiva, construída desde sua concepção, com foco no cliente e nos usuários e suas necessidades e problemas, mapeando novamente o perfil de *persona* que este *software* teria neste momento e pensando nos conceitos apresentados sobre *customer development*. Tal plataforma deverá utilizar os recursos e técnicas propostas com *devops* e desenvolvimento ágil, com práticas de desenvolvimento e versionamento de pacotes, integração contínua, entre outras. E validando os resultados obtidos através de análise dos resultados coletados e uma aposta contínua nessa plataforma implementada e projetada totalmente pensando no usuários.

## 7 REFERÊNCIAS

- ALENCAR, P. L. C et al. Empreendedorismo Start Up: um Estudo de Caso em uma Empresa de Tecnologia no Estado do Pará. In: SIMPÓSIO DE EXCELENCIA EM GESTÃO E TECNOLOGIA. 9. 2012. Rio de Janeiro. Pará: Faculdade de Castanhal, Pará, 2012. Disponível em: <<http://www.aedb.br/seget/arquivos/artigos12/30616273.pdf>> [Acesso em 14 de novembro de 2015]
- BECK, K. TDD Desenvolvimento Guiado por Testes. Tradução de Jean Felipe Patikowski Cheiran. Porto Alegre: Editora Bookman, 2009. 240 p.
- BLANK, S; DORF, B. Startup Manual do Empreendedor. Rio de Janeiro: Alta Books, 2014. 572 p.
- BLANK, S; SANT'ANNA C. Do Sonho à Realização Em 4 Passos - Estratégias Para Criação de Empresas de Sucesso. 1ª ed. São Paulo: Évora, 2012. 368 p.
- CANVANIZER. Business Model Generation. Alemanha, 2015. Disponível em: <<http://canvanizer.com/>>. [Acesso em: 3 dezembro 2015]
- FADEL, A. C., SILVEIRA, H. da M. Metodologias ágeis no contexto de desenvolvimento de software: XP, Scrum e Lean. 2010. p.26. Programa de Pós-Graduação da Universidade Estadual de Campinas. p.26. 2010. Disponível em: <[https://www.academia.edu/617187/Metodologias\\_%C3%A1geis\\_no\\_contexto\\_de\\_desenvolvimento\\_de\\_software\\_XP\\_Scrum\\_e\\_Lean?auto=download](https://www.academia.edu/617187/Metodologias_%C3%A1geis_no_contexto_de_desenvolvimento_de_software_XP_Scrum_e_Lean?auto=download)>. [Acesso em 12 de novembro de 2015].
- FILHO, D. L. B. Experiências com desenvolvimento ágil. 2008. Dissertação (Mestrado) – Instituto de Matemática e Estatística da Universidade de São Paulo, São Paulo, 2008. Disponível em: <<http://www.ime.usp.br/~dairton/files/Dissertacao-DairtonBassi.pdf>>. [Acesso em 14 de novembro de 2015.]
- FILHO, W. de P. P. Engenharia de Software – Fundamentos, métodos e padrões. 2ª ed. São Paulo: LTC, 2001.
- GIT. Fast-version-control, 2015. Disponível em: <<https://git-scm.com/>>. [Acesso em: 10 dezembro de 2015]
- GOTHELF, J. S; SEIDEN, J. Lean UX: Applying Lean Principles to Improve User Experience. O'Reilly Media, 2013. 152 p.
- HEROKU. Cloud Application Platform, 2015. Disponível em: <<http://heroku.com/>>. [Acesso em: 12 dezembro 2015]
- JENKINS. Build Great Things at Any Scale, 2015. Disponível em: < <https://jenkins.io/>>. [Acesso em: 10 dezembro de 2015]

- JONES, D. T., WOMACK, P. J. A mentalidade enxuta nas empresas: Lean Thinking elimine o desperdício e crie riqueza. Tradução por Simon & Schuster. 6ª ed. Rio de Janeiro: Elsevier, 2004.
- JUNIOR, F. J. Project Model Canvas: Gerenciamento de Projetos sem Burocracia. 1ª ed. Rio de Janeiro: Elsevier, 2013. 232 p.
- LUSSARI, J. B. Lean Startup. Business & Marketing School ESIC. 2013. Disponível em: <[www.administradores.com.br/producao-academica/lean-startup/6353/download/](http://www.administradores.com.br/producao-academica/lean-startup/6353/download/)>. [Acesso em 14 de novembro de 2015].
- MASSARI, V. L. Gerenciamento Ágil de Projetos. 1ªed. Rio de Janeiro: Brasport, 2014. 387 p.
- MILANI, A. Programando para iPhone e iPad: Aprenda a construir aplicativos para o IOS. 2ª ed. São Paulo: Novatec, 2012. 472 p.
- MOORE, G. A. Crossing the Chasm. 1ª ed. Harper Business, 1991.
- OSTERWALDER, A.; PIGNEUR, Y. Business Model Generation – A Handbook for Visionaries, Game Changers, and Challengers. 1º ed. John Wiley Trade, 2010. 258 p.
- PRUITT, J. S; ADLIN, T. The Essential Persona Lifecycle – Your Guide to Building and Using Persona. 1ª ed. Amsterdam: Elsevier, 2010. 235 p.
- PRUITT, J. S; ADLIN, T. The Persona Lifecycle- Keeping People in Mind Throughout Product Desgin. 1ª ed. Amsterdam: Elsevier, 2006. 724 p.
- POPPENDIECK, M; POPPENDIECK, T. Implementing Lean Software Development. Concept to Cash, 2011. 1ª ed. Londres: Pearson. 306 p.
- RIBEIRO, G. Lean Startup – Análise exploratória sobre a utilização por novas empresas brasileiras. 2014. Dissertação (Mestrado) - Fundação Getúlio Vargas – Escola de Administração de Empresas de São Paulo, 2014. Disponível em: <<http://bibliotecadigital.fgv.br/dspace/bitstream/handle/10438/13114/Ribeiro-Lean-Startup-MPA-FGV-2014.pdf?sequence=1&isAllowed=y>>. [Acesso em 14 de novembro de 2015].
- RIES, E; EISENMANN, T. R; DILLARD, S. Hypothesis-Driven Entrepreneurship: The Lean Startup. Harvard Business School Background, 2011. Disponível em: <<https://hbr.org/product/Hypothesis-Driven-Entrepr/an/812095-PDF-ENG>>. [Acesso em 14 de novembro de 2015].
- RIES, E. The Lean Startup. How today’s entrepreneurs use continuous innovation to create radically successful businesses. 1ª ed. New York: Crown Business, 2011. 338p.
- SATO, D. Devops na prática: entrega de software confiável e automatizada. 1ª ed. São Paulo: A Casa do Código. 2014. 248 p.
- SOMMERVILLE, I. Engenharia de Software. Tradução de André Mauricio de Andrade Ribeiro. Revisão Técnica de Kechi Hirama. 6ª ed. São Paulo: Pearson Education do Brasil, 2004.

TRIMI, S; MIRABENT, J. Business model innovation in entrepreneurship. LCC, 2012.  
Disponível em: <[http://www.turkuamk.fi/media/filer\\_public/2014/01/31/trimi\\_and\\_berbegal-mirabent\\_2012.pdf](http://www.turkuamk.fi/media/filer_public/2014/01/31/trimi_and_berbegal-mirabent_2012.pdf)> [Acesso em 14 de novembro de 2015].

YIN, R. K. Estudo de Caso Planejamento e Métodos. 2ª ed. Bookman, 2001. 205 p.