

OLIVIER THIERRY FRANCK CHANCE

**AUMENTAR AS CHANCES DE SUCESSO DE UMA STARTUP NO SETOR DOS
APLICATIVOS MÓVEIS : UMA APLICAÇÃO DO MÉTODO LEAN STARTUP**

Trabalho de Formatura apresentado à Escola
Politécnica da Universidade de São Paulo para
obtenção do diploma de Engenheiro de Produção.

SÃO PAULO

2015

OLIVIER THIERRY FRANCK CHANCE

**AUMENTAR AS CHANCES DE SUCESSO DE UMA STARTUP NO SETOR DOS
APLICATIVOS MÓVEIS : UMA APLICAÇÃO DO MÉTODO LEAN STARTUP**

Trabalho de Formatura apresentado à Escola
Politécnica da Universidade de São Paulo para
obtenção do diploma de Engenheiro de Produção.

Orientador: Prof. Dr. Guilherme Ary Plonski

SÃO PAULO

2015

Catálogo-na-publicação

Chance, Olivier Thierry Franck

AUMENTAR AS CHANCES DE SUCESSO DE UMA STARTUP NO
SETOR DOS APLICATIVOS MÓVEIS : UMA APLICAÇÃO DO MÉTODO LEAN
STARTUP / O. T. F. Chance -- São Paulo, 2015.

102 p.

Trabalho de Formatura - Escola Politécnica da Universidade de São
Paulo. Departamento de Engenharia de Produção.

1.Lean Startup I.Universidade de São Paulo. Escola Politécnica.
Departamento de Engenharia de Produção II.t.

AGRADECIMENTOS

Agradeço, em primeiro lugar à minha família, que sempre me apoiou em todos os momentos em que precisei e tem extrema importância em tudo que conquistei até hoje.

Em segundo lugar, gostaria de agradecer a minha namorada, Ingrid, que esteve sempre ao meu lado, me ajudou na redação deste trabalho e foi uma força inspiradora.

Queria agradecer também o Professor Ary Plonski por ter me orientado e auxiliado na realização deste trabalho.

Agradeço minha família e amigos do Brasil, em especial Raphaël, Zuca, Sergio, Mario, Mélina, Hadrien, Pierre e FXs, que através de dicas e conselhos me ajudaram muito durante minha jornada no Brasil.

Por fim, gostaria de agradecer a toda a equipe da empresa MatchUp, que confiou em mim para a realização deste projeto.

Everybody has a plan until they get punched in the mouth.

Mike Tyson

RESUMO

Este trabalho apresenta uma metodologia de desenvolvimento de empresas *startup* no setor dos aplicativos móveis, focada nas primeiras fases da sua vida, nomeadamente da primeira ideia do produto até o lançamento no mercado. O aplicativo de *dating* MatchUp, desenvolvido pelo autor deste trabalho e pertencente à uma *holding* brasileira focada nas tecnologias da internet, serve de ilustração para a metodologia. O autor do trabalho, por ser um dos dois fundadores da *startup* MatchUp, foi encarregado de definir uma estratégia de criação do produto, além da construção do produto em si. Este trabalho apresenta a estratégia implementada. Mais precisamente, o autor mostra um caminho, passo a passo, a fim de aumentar as chances de sucesso da empresa, ao aumentar a probabilidade de uma boa adequação do produto com o mercado.

O método *lean startup*, desenvolvido principalmente por Eric Ries, constitui o fio condutor deste trabalho. Ele é complementado por ferramentas tradicionais de estratégia de empresas (Porter, PESTEL), de gerenciamento de projeto (Matriz de criticidade), de geração de ideias (Carta de empatia, entre outras), de gerenciamento de projeto informático (*Extrem Programming*) e outras ferramentas desenvolvidas pelo autor do trabalho.

Palavras chaves: *Startup*, Empreendedorismo, Modelo de Negócios, Desenvolvimento de Clientes, Desenvolvimento de Produto, Desenvolvimento de *Software*.

ABSTRACT

This paper presents a startup development methodology in the mobile applications industry, focused on the early stages, including the first idea to the product launch. The MatchUp dating application, developed by this author and property of a Brazilian holding company focused on Internet technologies, is an illustration for the methodology. The author of this work, as one of the two founders of the startup MatchUp, was tasked to define a product creation strategy in addition to the construction of the product itself. This paper presents the strategy implemented. More precisely, the author shows a path step by step, in order to increase the company's chances of success by increasing the probability of a good adaptation product/market.

The lean startup method, developed mainly by Eric Ries, is the guiding thread of this paper. This method is supplemented by other tools, including company policy (Porter, PESTEL), project management (criticality matrix), idea generation (empathy Charter, among others), IT project management (Extrem Programing) and others tools developed by the author of this paper.

Keywords: Startup, Entrepreneurship, Business Model, Customer Development, Product Development, Software Development.

LISTA DE FIGURAS

Figura 1: A pirâmide da <i>startup</i> segundo Ries	27
Figura 2: Ciclo tradicional de desenvolvimento de produto	27
Figura 3: O <i>Lean canvas</i>	29
Figura 4: Os riscos no lean canvas.....	31
Figura 5: O ciclo Construir - Medir – Aprender	33
Figura 6: <i>Pirate metrics</i>	35
Figura 7: O método do brainstorming.....	37
Figura 8: Mapa de empatia	39
Figura 9: A análise PEST.....	40
Figura 10: As 5 forças de Porter	41
Figura 11: Abordagem clássica de gerenciamento de um projeto de <i>software</i>	43
Figura 12: O ciclos da abordagem eXtrem Programing	44
Figura 13: Esquema geral do trabalho	49
Figura 14: <i>Timeline</i> do projeto.....	55
Figura 15: <i>Timeline</i> do projeto.....	57
Figura 16: Mapa de empatia para o segmento de cliente dos jovens de 18 à 30 anos, que se deslocam regularmente	59
Figura 17: <i>Timeline</i> do projeto.....	64
Figura 18: As classes sociais no Brasil, classificação em 5 classes.....	65
Figura 19: <i>Timeline</i> do projeto.....	70
Figura 20: <i>Lean canvas</i> completo	71
Figura 21: <i>Timeline</i> do projeto.....	74
Figura 22: Script da entrevista sobre os problemas	78
Figura 23: Script da entrevista sobre as soluções	82
Figura 24: <i>Timeline</i> do projeto.....	85
Figura 25: Script da entrevista sobre o PVM.....	88
Figura 26: <i>Timeline</i> do projeto.....	91
Figura 27: Ilustração de duas classes, como caixas pretas, com 3 inputs e 3 outputs para a primeira e 3 inputs e 1 output para segunda.....	96

LISTA DE TABELAS

Tabela 1: As cinco primeiras empresas mundiais em função da capitalização (31 de março de 2014)	18
Tabela 2: Matriz de criticidade	30
Tabela 3: Principais problemas e soluções achadas.....	62
Tabela 4: Matriz de criticidade do projeto MatchUp.....	75
Tabela 5: Classificação das hipóteses segundo sua criticidade	76
Tabela 6: Vantagens e Desvantagens dos aplicativos atuais de dating.....	80
Tabela 7: Nota dada a adequação problema-solução.....	89
Tabela 8: Resultado da entrevista no diz respeito ao PVM	90
Tabela 9: Ilustração de uma pilha de funcionalidades (P1), com o primeiro ciclo CMA	94
Tabela 10: Ilustração de uma pilha de funcionalidades (P1), com o primeiro ciclo CMA, incluindo as classes.....	95
Tabela 11: Ilustração de duas classes, para a funcionalidade 2 do ciclo 1	96
Tabela 12: Medidas mínimas à atingir para parar o desenvolvimento do produto.....	97

LISTA DE GRÁFICOS

Gráfico 1: Intensidades das 5 forças de Porter no setor dos aplicativos de dating	68
Gráfico 2: Classificação das respostas dos entrevistados segundo a solução	83
Gráfico 3: Repartição dos usuários em função da disposição de pagar o serviço	84
Gráfico 4: Classificação média do Problema i em função da frequência de deslocamento.....	87
Gráfico 5: Medidas <i>Pirate Metrics</i> segundo o número do ciclo	98

LISTA DE ABREVIATURAS

PUV : Proposição única de valor

Cr : Criticidade

Pr : Probabilidade

Gr : Gravidade

CMA : Construir – Medir - Aprender

PVM : Produto Viável Mínimo

PEST : Política, Econômica, Social e Tecnológica

UX : *User Experience*

UI : *User Interface*

F : Funcionalidade

T : Tempo de um ciclo CMA

Prd : Prioridade

H : Hora

κ : Classe (programação)

P : Pilha de funcionalidades

K_v : Coeficiente de viralização

XP : *Extrem Programming*

SUMÁRIO

1. INTRODUÇÃO	18
1.1. CONTEXTO	18
1.2. PROBLEMA	19
1.3. O OBJETIVO DO TRABALHO	20
1.4. O ESTÁGIO	21
1.5. A ESTRUTURA DO TRABALHO	21
1.6. JUSTIFICATIVA DO TRABALHO	22
1.6.1. JUSTIFICATIVA DA ABORDAGEM	22
1.6.2. INSERÇÃO DO TRABALHO NA ENGENHARIA DE PRODUÇÃO	22
2. REVISÃO BIBLIOGRÁFICA	24
2.1. STARTUP	24
2.2. LEAN STARTUP	25
2.2.1. OS PONTOS CHAVES	25
2.2.2. OS PILARES DO METODO	27
2.3. MÉTODOS DE GERAÇÃO DE IDEIA	36
2.3.1. BRAINSTORMING	36
2.3.2. MAPA DE EMPATIA	37
2.4. FERRAMENTA DE ANÁLISE ESTRATÉGICA	39
2.4.1. ANÁLISE PEST	39
2.4.2. AS FORÇAS DE PORTER	41
2.5. GESTÃO DE PROJETO INFORMÁTICO : O MÉTODO EXTREM PROGRAMING	43
2.5.1. OS CICLOS DE DESENVOLVIMENTO XP	43
2.5.2. AS VALORES CHAVES	45
2.5.3. AS PRATICAS DO XP	46
3. METODOLOGIA	49
3.1. ESQUEMA GERAL	49
3.2. O DETALHE DOS PASSOS DA CRIAÇÃO DA STARTUP	50
3.2.1. ADEQUAÇÃO PROBLEMA SOLUÇÃO	51
3.2.2. ADEQUAÇÃO PROBLEMA-MERCADO	54
3.2.3. LANÇAMENTO	54
4. RESULTADOS	55
4.1. A IDEIA INICIAL	55
4.1.1. COMPOSIÇÃO DA EQUIPE E DURAÇÃO DO PASSO	55
4.1.2. CONTEXTO	55
4.1.3. DETALHES DA IDEIA INICIAL	56
4.2. ESTRUTURAÇÃO DAS HIPÓTESES	56
4.2.1. OBJETIVO DA ETAPA	56
4.2.2. COMPOSIÇÃO DA EQUIPE E DURAÇÃO DA ETAPA	57
4.2.3. O CLIENTE DO APLICATIVO E SEUS PROBLEMAS	58
4.2.4. AS SOLUÇÕES IMAGINADAS	60
4.2.5. A PROPOSTA UNICA DE VALOR DO APLICATIVO	63
4.3. ANÁLISE DO SETOR ESCOLHIDO	63
4.3.1. OBJETIVO DESSA ETAPA	63
4.3.2. COMPOSIÇÃO DA EQUIPE E DURAÇÃO DA ETAPA	64
4.3.3. ANÁLISE PEST	64
4.3.4. ANÁLISE DE PORTER	67
4.3.5. LIÇÕES RETIRADAS PELO PROJETO	69

4.4. ELABORAÇÃO DO LEAN CANVAS	69
4.4.1. OBJETIVA DESSA ETAPA	69
4.4.2. COMPOSIÇÃO DA EQUIPE E DURAÇÃO DA ETAPA	69
4.4.3. CLIENTES, PROBLEMAS, SOLUÇÕES E PUV	70
4.4.4. CANAIS DE DISTRIBUIÇÃO	70
4.4.5. FLUXO DE CAIXA	72
4.4.6. ESTRUTURA DOS CUSTOS	72
4.4.7. METRICAS	72
4.4.8. VANTAGEM COMPARATIVA	73
4.5. PRIMEIRAS CONFRONTAÇÕES	73
4.5.1. OBJETIVA DESSA ETAPA	73
4.5.2. COMPOSIÇÃO DA EQUIPE E DURAÇÃO DA ETAPA	73
4.5.1. CONFRONTAÇÃO COM ESPECIALISTAS	74
4.5.2. CONFRONTAÇÃO COM CLIENTES ALVOS NO DIZ RESPEITO AOS PROBLEMAS	74
4.5.3. CONFRONTAÇÃO COM CLIENTES ALVO NO DIZ RESPEITO AS SOLUÇÕES	81
4.6. ELABORAÇÃO DO PRIMEIRO PVM	85
4.6.1. OBJETIVO DESSA ETAPA	85
4.6.2. COMPOSIÇÃO DA EQUIPE E DURAÇÃO DA ETAPA	85
4.6.3. AS CARACTERISTICAS DO PVM	86
4.6.4. CONFRONTAÇÕES COM OS EARLY-ADOPTER	86
4.7. ITERAÇÕES SEGUNDO OS CICLOS CMA	91
4.7.1. OBJETIVO DESSA ETAPA	91
4.7.2. COMPOSIÇÃO DA EQUIPE E DURAÇÃO DA ETAPA	91
4.7.3. CONSTRUIR O PRODUTO	92
4.7.4. DEFINIR QUANDO ACABA O PROJETO	97
<u>5. CONCLUSÃO</u>	<u>99</u>
<u>6. REFERÊNCIAS BIBLIOGRÁFICAS</u>	<u>100</u>

1. Introdução

1.1. Contexto

Em 2009, Jan Koum e Brian Acton, dois veteranos engenheiros da empresa Yahoo, criam o WhatsApp, um aplicativo móvel de conversação instantânea. No mês de outubro do ano 2013, quatro anos depois do lançamento, WhatsApp anunciou ter 350 milhões de usuários ativos por mês, 50 a mais do que em agosto. No mês de dezembro eram 400 milhões, passando a 450 milhões em fevereiro de 2014. Nessa data, o número de mensagens enviadas por dia pelo aplicativo era igual ao número de mensagens enviadas pelo serviço de SMS: 10 bilhões. [HOLSON, 2014]

No dia 19 de fevereiro de 2014, cinco anos após a sua criação, a empresa Facebook comprou a WhatsApp por 19 bilhões de dólares, ou seja, 40 dólares por usuário ou 350 milhões de dólares por colaborador. [HOLSON, 2014]

Essa história de sucesso extremamente rápido de uma empresa de tecnologia não é única. Youtube, um serviço de compartilhamento de vídeos, foi criado em fevereiro de 2005 e adquirido pela Google por 1,65 bilhão de dólares em outubro de 2006, um ano e meio depois. [REUTERS, 2006] Vine, um outro serviço de compartilhamento de vídeos, foi criado em junho de 2012 e adquirido pelo Twitter no mês de Outubro de 2012, 4 meses depois, por 30 milhões de dólares. [ISAAC et al., 2012]

Basta olhar para as maiores capitalizações do mundo para constatar que as empresas de tecnologia, apesar por possuírem uma historia recente na economia mundial, podem ter um peso muito importante na mesma.

Classificação	Nome	Capitalização (em G\$)	Tipo	Criação
1	Apple	469	Tecnologia	1976
2	Exxon Mobil Corp	416	Petróleo	1870
3	Google Inc	409	Tecnologia	1998
4	Microsoft	318	Tecnologia	1975
5	Berkshire Hathaway	286	Banco	1839

Tabela 1 : As cinco primeiras empresas mundiais em função da capitalização (31 de março de 2014), Fonte: Fortune, 2014

Considerando as cinco maiores empresas segundo a capitalização de mercado, três são empresas de tecnologia e têm menos de 40 anos.

Estes sucessos são extremamente midiaticizados. Lançado em 2010, o filme *The Social Network* conta a história de Mark Zuckerberg, fundador da Facebook, que conheceu um sucesso comercial importante [IMBD, 2010]. Num intervalo de somente dois anos, em 2013 e 2015, saíram dois filmes sobre Steve Jobs, o fundador da empresa Apple. [IMBD, 2014]

Todos estes filmes, números gigantes e artigos de imprensa que falam de tal aquisição, podem fazer acreditar que o sucesso é fácil e que uma boa ideia basta para construir uma *startup* que se torne uma empresa bem sucedida.

No entanto, há um fato menos conhecido : 75% das *startups* falham. [BLANK, 2013]

1.2. Problema

Considerando essa porcentagem de falha, é interessante fazer o questionamento se esses sucessos e fracassos são previsíveis.

Por definição, uma *startup* opera num contexto muito incerto. Com efeito, ela tem um plano de negócio que pouco ou nunca foi provado no passado. O risco de falha, nesse contexto, é muito mais elevado do que para uma empresa cujo negócio já foi testado com números, dados e tecnologias existentes. Em suma, a sorte teria, portanto, um lugar privilegiado no sucesso ou fracasso de uma *startup*.

Nessa lógica, seria inútil, até contraproducente, promover qualquer sistema de gerenciamento ou regras de trabalho, pois a única coisa importante é a ideia inicial e como ela se desenvolverá ao decorrer do projeto. Essa imagem faz parte do imaginário coletivo, em que uma *startup* é criada por um grupo de amigos tendo uma ideia boa e desenvolvendo-a sem muito planejamento, sem quadro de trabalho definido ou gerenciamento.

Na verdade, essa imagem é errônea. Uma *startup* opera num contexto muito

incerto e é justamente por isso que precisa de um quadro bem definido. Segundo Eric Ries, cujas teses sobre o *Lean startup* vão apoiar este trabalho, criar uma *startup* é sinônimo de promover um tipo de gerenciamento. [RIES, 2011]

Existem inúmeras razões que podem explicar a taxa elevada de fracasso das *startups* :

- Criação de um produto ou serviço que ninguém quer.
- Ausência total de gestão ou tentativa de aplicar uma gestão tradicional.
- Falta de reatividade de modo a responder as mudanças que ocorrem.
- Tempo de execução longo demais.
- Um projeto ambicioso demais, sem chance de sucesso considerando a tecnologia atual ou a experiência da equipe.

Essas causas todas - a lista não é exaustiva - levam à única causa final do fracasso: a falta de caixa.

1.3. O objetivo do trabalho

O objetivo deste trabalho é propor uma metodologia que permita aumentar as chances de sucesso de uma *startup*, em particular achar um problema que vale a pena ser resolvido, assim como uma solução adequada para esse problema. Este trabalho foca nas primeiras fases da vida de uma *startup* de aplicativo móvel, da definição da primeira ideia até a criação do produto que será lançado no mercado.

Todas as causas de fracasso supramencionadas são evitáveis a partir do momento em que os colaboradores aplicam regras específicas de gerenciamento. Esse conceito foi inicialmente desenvolvido por Eric Ries em 2008 : o *Lean startup*. [RIES, 2011]

Segundo Ries, “o objetivo de uma *startup* é descobrir a coisa certa a criar – a coisa que os clientes querem e pela qual pagarão- o mais rápido possível” [RIES, 2011]. Será também o foco deste trabalho descobrir quais são as regras e técnicas para alcançar este objetivo.

Mais concretamente, trata-se de dar uma metodologia, através de diferentes passos estruturados, que permita a validação do caminho de uma *startup*, a fim de limitar a maior parte dos riscos que podem surgir nas primeiras fases da sua vida.

1.4. O estágio

O autor deste trabalho foi contratado no primeiro dia da criação da *startup* MatchUp, que tem como objetivo criar um aplicativo móvel de paquera. Sua missão era definir uma metodologia de gerenciamento e a aplicar com o intuito de aumentar as chances de sucesso do produto no seu lançamento no mercado. O autor foi também o primeiro programador do aplicativo. Não se tinha ao início do projeto uma ideia clara ou precisa do produto a desenvolver, ou seja, ele iniciaria o projeto do zero. Esse estágio tinha um prazo de seis meses, o que correspondeu ao tempo entre a geração da primeira ideia e a produção do aplicativo a colocar no mercado.

A equipe era inicialmente composta de um programador e aumentou para quatro programadores e um *designer*.

1.5. A estrutura do trabalho

Este trabalho é composto de duas partes.

A primeira parte é uma revisão bibliográfica de dois conceitos importantes e que respondem às duas questões primordiais. Em primeiro lugar, o que é uma *startup*? Ou seja, quem poderá aplicar o método apresentado neste trabalho. Em segundo lugar, o que é o *Lean startup*? De onde vem e o que é exatamente o método que será usado neste trabalho?

A segunda parte constitui o conteúdo principal do trabalho, que abordará diferentes técnicas. Este é focado nos primeiros passos da criação de uma *startup*, sendo o primeiro a geração da ideia inicial, que finalmente foi selecionada pela equipe como a criação de um aplicativo de *dating*. A partir da ideia, é necessário criar um cenário, que Maurya (2012) chama de *lean canvas*, um modelo de negócio breve que cabe em apenas uma página. O objetivo deste segundo passo é escrever da forma mais clara e concisa quais serão:

- O ou os problemas que a empresa enfrentará
- A solução selecionada
- O que ela tem de única
- Os clientes
- Os canais para atingi-los
- A estrutura de custos

- A fonte de lucro
- Os indicadores chaves para avaliar o fracasso/sucesso da empresa
- O conjunto de vantagens que a empresa tem comparativamente à concorrência

No entanto, um *lean canvas* é composto por hipóteses que, portanto, devem ser testadas. A base do *lean startup* é o ciclo Construir-Medir-Aprender, que permite testar sistematicamente as hipóteses da *startup*. Com essa finalidade, é necessário construir algo para mostrar aos clientes - pode ser um desenho, um protótipo, até o produto final - coletar dados, medir a pertinência das hipóteses e tirar finalmente uma aprendizagem dessa confrontação. Estes ciclos devem demorar o menor tempo possível. Para atingir esse objetivo, uma técnica de gestão de projeto focado no desenvolvimento de aplicativo por equipe de pequeno tamanho foi usada: o *extreme programming*. Depois de vários ciclos Construir-Medir-Aprender o aplicativo ficou pronto. A estratégia de lançamento constitui o próximo passo, que não será apresentada neste trabalho.

1.6. Justificativa do trabalho

1.6.1. Justificativa da abordagem

Uma *startup*, como qualquer empresa, tem recursos limitados: dinheiro, tempo, mão de obra, entre outros. Para a maioria delas, pode demorar semanas, meses ou até anos antes de receber a primeira receita, enquanto os gastos aparecem no primeiro dia. Quanto mais durar esse período de despesas sem receita, maior é a probabilidade de falência da *startup*. Sendo assim, ela deve encontrar o jeito mais rápido possível de uma existência economicamente perene. Desta maneira, a *startup* deve achar, antes da falência, um problema relevante, ou seja, que tenha uma solução tecnicamente possível, com um mercado bastante amplo, em que a empresa pode se implementar e prosperar. Essa procura de um modelo de negocio viável é justamente o objetivo do *Lean startup*. A partir de um primeiro modelo de negocio que fica baseado somente em hipóteses, o empresário deve testar, validar ou refutar cada uma delas para obter fatos e construir, por várias iterações, uma empresa que gera lucros.

1.6.2. Inserção do trabalho na engenharia de produção

Este trabalho abrange vários conceitos estudados durante a formação de um

engenheiro de produção: gestão de projeto, inovação, melhora de processo e serviço, otimização.

A criação de uma *startup* é um processo que permite aos empregadores colocarem em prática todos estes conceitos diferentes. Trata-se de um terreno muito fértil para um engenheiro de produção.

2. Revisão bibliográfica

2.1. Startup

Não há como definir facilmente, claramente e precisamente uma *startup*, já que não existem critérios ou normas reconhecidos para tal. O American Heritage Dictionary (2013) proporciona uma definição muito geral: “uma *startup* é um negócio que começou suas operações há pouco tempo”. Mesmo a denominação deste tipo de empresa apoia esta definição. No entanto, não é precisa o suficiente, tendo como exemplo um restaurante clássico que acabou de ser criado e não pode ser considerado como *startup*.

A Associação Brasileira de Startups (ABS) propõe uma definição mais clara: é “uma empresa de base tecnológica, com um modelo de negócios repetível e escalável, que possui elementos de inovação e trabalha em condições de extrema incerteza”.

Base tecnológica: Segundo Natalie Robehmed (2013), cada *startup* se apoia em tecnologia para vender seu serviço ou produto. No entanto, a tecnologia não constitui necessariamente o produto ou o serviço vendido. Por exemplo, Uber é uma *startup* que relaciona motoristas privados com clientes por meio de um aplicativo –apoia-se na tecnologia para oferecer um serviço que já existia antes mesmo dos primeiros computadores aparecerem.

Repetível: Segundo Gitahy (2010), ser repetível significa ter a capacidade de vender o produto ou serviço em escala potencialmente ilimitada, sem muitas customizações ou adaptações para o cliente.

Escalável: Para Gitahy (2010), ser escalável é um ponto fundamental para cada *startup* e significa ter a capacidade de crescer sempre mais, com custos sempre menores por unidade de produção, permitindo assim uma margem de lucro sempre maior.

Inovação: A inovação tem papel principal para uma *startup*, que a usa a fim de fomentar o crescimento que a diferencia das empresas clássicas.

Incerteza: A incerteza significa que é muito difícil saber se a ideia dará certo, já que uma *startup* proporciona um novo produto, um novo serviço ou um novo jeito de produzir esse serviço ou produto, o que conduz a empresa a se envolver em um ambiente de grande incerteza.

Segundo a ABS, existem mais de 10 mil *startups* no Brasil, que levantaram em fundos cerca de R\$ 1,7 bilhão em 2012. Em 2011, mais de R\$ 750 milhões foram investidos nessas *startups*, contra R\$ 350 milhões no período de 2005 a 2008, segundo um estudo da Fundação Getúlio Vargas (2011).

2.2. Lean Startup

2.2.1. Os pontos chaves

Este trabalho é principalmente baseado no trabalho de Eric Ries, *The Lean startup* (2011), Ash Maurya, *Running Lean* (2012) e Steve Blank, *Four steps to the epiphany* (2006).

Segundo Ries, há duas causas principais que explicam o fracasso de uma *startup*:

- O foco exacerbado dos empregadores em um modelo de negócio ou em uma análise de mercado inicial, enquanto os planos de uma *startup* devem evoluir. Com efeito, no início da sua vida, é extremamente difícil saber quais serão realmente seus clientes ou seu mercado. Essa é, também, a hipótese principal de Steve Blank e Ash Maurya.
- Os empregadores tentam aplicar metodologias de gerenciamento tradicional ou adotam uma atitude de “só fazer as coisas” e evitam qualquer forma de gerenciamento ou planejamento.

O *lean startup* foca justamente na melhora do gerenciamento de uma empresa em uma situação de extrema incerteza. Numerosos empregadores recusam qualquer forma de gerenciamento, sob o pretexto de que é sinônimo de muito trabalho administrativo. No entanto, de acordo com Ries, empreendedorismo e gerenciamento são totalmente compatíveis e devem coabitar a fim de garantir a agilidade e o sucesso

da empresa.

A origem do *lean startup* é a *lean manufacturing revolution* liderada por Taiichi Ohno e Shigeo Shingo, da empresa Toyota. Essa técnica implicava em mudar radicalmente a maneira de gerenciar o sistema de produção. Este teve que: se basear na inventividade de cada trabalhador, na redução dos lotes, na produção *just in time*, no controle dos inventários e na aceleração dos ciclos de produção. [OHNO, 1978]

O *lean startup* adapta essas ideias no contexto do empreendedorismo. Os pontos chave são:

- O *lean startup* propõe aos empregadores avaliar o trabalho de um jeito diferente do das empresas tradicionais. Na produção industrial, o progresso é medido pela produção de bens de qualidade. O *lean startup* usa uma outra medida, “o conhecimento fidedigno”.
- Todo o que é feito e experimentado tem que trazer esse conhecimento, é a base do método. Como a empresa evolui num contexto de extrema incerteza, é esse conhecimento o que permitirá a empresa fazer as escolhas certas para sobreviver e crescer, alimentando o que Eric Ries chama de “motor de crescimento” da empresa.
- A fim de maximizar esse conhecimento, é importante criar uma equipe interoperacional: uma equipe encarregada do desenvolvimento do produto, vendas, marketing e distribuição, em que todos devem trabalhar juntos.
- Essas equipes farão hipóteses e as testarão. É inútil fazer um plano a longo prazo no início. Deve-se, ao contrário, seguir ciclos curtos “construir-medir-aprender” para obter continuamente conhecimentos e diminuir as consequências dos erros.
- Caso seja impossível continuar com a estratégia inicial, pois os ciclos mostraram que era errada, a *startup* deve “pivotar”. Mais precisamente, Ries divide uma *startup* em três andares de uma pirâmide. A base é a visão dos empregadores, criar algo que possa mudar o mundo. Para atingir essa visão, a *startup* usará uma estratégia, o segundo andar, que inclui o modelo de negócio. O produto, terceiro e último andar, é o resultado dessa estratégia. Os ciclos construir-medir-aprender testarão a estratégia permanentemente e a ajustarão, a fim de criar um produto que atinja as necessidades do mercado.

Nessa pirâmide, o “pivot” corresponde a uma mudança profunda do segundo andar, ou seja, da estratégia, enquanto o primeiro andar - a visão - nunca muda e o terceiro - do produto - evolui sempre, seguindo os ciclos de aprendizagem.

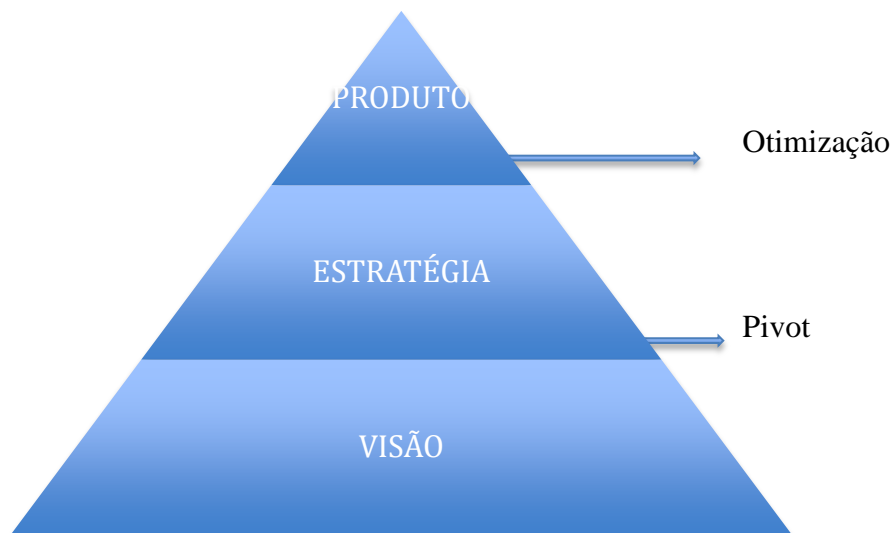


Figura 1: A pirâmide da *startup* segundo Ries

Fonte: Ries (2011)

2.2.2. Os pilares do método

2.2.2.1. Criar o plano

Aprender, portanto, é o fator chave de sucesso do método *lean startup*. Segundo Blank (2005), o primeiro passo é identificar os clientes e o mercado. No entanto, raramente os empregadores tentam descobrir isso em primeiro lugar. Com efeito, para Blank, o método tradicional de desenvolvimento de um novo projeto é:



Figura 2: Ciclo tradicional de desenvolvimento de produto

Fonte: Blank, 2005

As quatro fases são:

- *Conceito*: A primeira fase consiste principalmente em moldar a ideia, criar um modelo de negócio, verificar se é tecnicamente possível construir o produto ou propor o serviço e planejar a parte financeira da empresa. Nesta fase, tudo se apoia em hipóteses que vêm de discussões ou análises de mercado.
- *Desenvolvimento do produto*: Nesta segunda fase, a empresa já é dividida entre diferentes departamentos. As discussões sobre a estratégia iniciada na primeira fase acabaram; esta fase é o tempo da construção do produto ou serviço em si.
- *Testes Alpha/Beta*: Esta fase permite aos engenheiros verificarem que o produto não apresenta *bugs* e que, tecnicamente, corresponde às expectativas. A equipe de *marketing* desenvolve um plano completo de comunicação.
- *Lançamento*: O produto ou serviço está pronto e será confrontado com o mercado.

Segundo Ries, é nessa ultima fase que, muitas vezes, os empregadores percebem que construíram um produto ou serviço que poucas pessoas, ou ninguém, realmente quer.

Para Blank, se as *startups* fracassam por falta de clientes e não por um fracasso no desenvolvimento do produto, por que neste esquema, tem um processo de desenvolvimento de produto e nenhum processo pelo desenvolvimento da clientela?

O cliente e o mercado devem ser o primeiro conhecimento da *startup*.

Obviamente, os clientes e o mercado não podem ser considerados independentemente do produto ou serviço a criar ou dos outros componentes de um modelo de negócio clássico. Mas, segundo Maurya, é inútil criar um tal modelo de negócio de várias dezenas de páginas enquanto este tenha por base somente hipóteses ou a intuição do empregador. Em vez disso, Maurya propõe usar o *lean canvas*, adaptado do canvas proposto por Alexander Osterwalder e Yves Pigneur (2010). Trata-se de apresentar o modelo de negócio com só uma página. Segundo os autores, isso permite, além de evitar perder tempo com hipóteses não validadas, favorecer o *brainstorming* dentro da equipe.

O *lean canvas* apresenta-se desta forma:

1	Soluções	Proposição única de valor	Vantagem comparativa	Segmento de cliente
	4		5	
	Métricas		Canais	
	8		9	
Estrutura dos custos			Fluxo de caixa	
7			6	

Figura 3: O *Lean canvas*

Fonte: Osterwalder & Pigneur, 2010

Os diferentes pontos e hipóteses a formular são :

Problema: Quais são os problemas principais que os futuros clientes enfrentam?

Segmento de cliente: Quais são os diferentes grupos de clientes ou organizações que são confrontados com estes problemas?

Proposição única de valor (PUV): Por que esses clientes comprariam o produto da empresa?

Solução: Quais são as características principais do produto ou serviço que apoiam essa PUV?

Vantagem comparativa: Algo que não pode ser copiado ou comprado facilmente, tornando-se o diferencial entre os concorrentes.

Fluxo de caixa: Como a empresa vai ganhar dinheiro?

Estrutura dos custos: Quais são os custos que enfrentará a empresa?

Métrica: Indicadores que permitem seguir a evolução da empresa.

Canais: Como a empresa vai comunicar e entrar em contato com seus clientes para proporcionar-lhes a PUV?

2.2.2.2. Identificar as partes críticas do plano

Uma vez o *lean canvas* realizado, ele deve ser testado. No entanto, segundo Ries (2011), existem hipóteses em demasia a serem verificadas nessa fase da criação da *startup*. Sendo assim, algumas deverão ficar hipotéticas. Para identificá-las, Maurya (2012) sugere apoiar-se na distinção que existe entre risco e incerteza. Segundo Douglas Hubbard (2010), a diferença é:

- *Incerteza*: A falta de uma certeza absoluta, ou seja, a existência de mais de uma possibilidade.
- *Risco*: Um estado de incerteza em que uma ou mais das possibilidades implica uma perda, um desastre ou uma consequência realmente indesejável.

As incertezas não são as prioridades. Os empregadores devem estar focados nos riscos, pelo menos ao início da vida da *startup*. Uma ferramenta que pode ser usada neste sentido é a matriz de criticidade. Essa abordagem origina-se na indústria. Cada risco do plano tem uma probabilidade *Pr* de acontecer e uma gravidade *Gr* associada. É portanto possível classificar os riscos segundo a matriz [TACOMA, 2014]:

		Gravidade				
		1	2	3	4	5
Probabilidade	5					
	4					
	3					
	2					
	1					

Tabela 1: Matriz de criticidade

Fonte: Tacoma, 2014

A criticidade *Cr* é o produto da probabilidade *Pr* pela gravidade *Gr*, $Cr = Pr \times Gr$. Quanto maior a *Cr*, maior tem que ser a atenção dos empregadores. Nessa matriz, as hipóteses que ficam nas células mais escuras serão testadas em primeiro lugar. [TACOMA, 2014].

Segundo Maurya, existem três tipos de risco:

Risco no diz respeito ao produto (P): criar um bom produto.

Risco no diz respeito ao cliente (C): construir um caminho até o cliente.

Risco no diz respeito ao mercado (M): construir uma empresa viável.

Estes riscos integram-se no *lean canvas*, cada parte deles identificam-se com um dos citados, como é possível ver na seguinte tabela:

Problemas	Soluções	Proposição	Vantagem comparativa	Segmento de cliente
P	P	P	M	C
	Métricas		Canais	
	P		C	
Estrutura dos custos		Fluxo de caixa		
M		M		

Figura 4: Os riscos no lean canvas

Fonte : Maurya, 2012

Nesta fase, logo depois da criação de um ou mais planos, quando os riscos estão sendo classificados, Maurya sugere apoiar-se em consultores - podendo ser especialistas numa área relevante para o negocio, investidores, entre outros - para ter uma visão diferente da visão dos empregadores. No entanto, para o autor, é ainda cedo demais para chamar clientes potenciais, sabendo que ainda não há o suficiente para ser apresentado.

2.2.2.3. Iterar segundo o ciclo Construir-Medir-Aprender

Segundo Ries, a base do método *lean startup* é a aprendizagem, que o autor chama de “fidedigna” ou “validada”, sendo assim um conhecimento ao qual o empregador pode confiar, pois este foi provado através de testes e experimentações. O autor recomenda o aprendizado mais cedo possível.

Ele usa o exemplo de Zappos para enfatizar este ponto. Zappos é um *website de e-commerce* fundado em 1999 por Nick Swinmurm. O conceito inicial era a venda

exclusiva de sapatos na internet. Ao invés de começar a construir estoque, elaborar contratos com marcas de sapatos, o criador da empresa queria testar as hipóteses dele imediatamente, a primeira delas era saber se haviam clientes com vontade de comprar sapatos na internet. O fundador realizou acordo com lojas de sapatos da cidade onde morava, iniciou propondo a ideia de tirar fotos dos sapatos das lojas e vendê-los em seu *website*. Em caso de venda, ele buscava o sapato na loja e cuidava da entrega ao cliente. Rapidamente, Zappos tirou a conclusão que de fato existia uma demanda para a venda de sapato na internet. Somente a partir desta informação, Zappos começou a criar o próprio estoque e efetivar acordos com marcas de sapatos. Deste jeito, Nick Swinmurm aprendeu muito mais do que se ele tivesse realizado análises de mercado ou um modelo de negócio de dezenas de páginas, sendo assim:

- Zappos recebeu dados à respeito dos clientes muito mais precisos, pois estava observando clientes de verdade.
- Zappos interagiu com clientes para aprender as necessidades deles o mais perto possível.
- Zappos pôs-se numa posição em que pudesse ser surpreendido pelo comportamento dos clientes. Por exemplo, o que acontece se o cliente quiser devolver ou trocar o produto?

Portanto no método *lean startup*, as empresas não devem esperar antes de testar as hipóteses. Segundo Ries, existem dois tipos principais de hipóteses, as de valor e as de crescimento. A hipótese de valor testa se um produto ou serviço realmente proporciona valores aos clientes uma vez que estes o utilizam. A hipótese de crescimento testa como os novos clientes poderão descobrir este produto ou serviço.

Como o Ries demonstrou com o caso Zappos, uma experimentação é mais do que uma pesquisa teórica, é também um primeiro produto. No momento em que estiver pronto para ser distribuído mais amplamente, já existirão clientes estabelecidos, já terá enfrentado verdadeiros problemas e obterá especificações a respeito do que tem realmente para construir. Contrariamente ao contexto da planificação estratégica, essas especificações terão como raízes verdadeiras experiências com o cliente.

Portanto, segundo Ries, uma *startup* é um catalizador que transforma ideias e hipóteses em produto. E a medida que os clientes interagem com os produtos, estes geram dados. Estes dados são qualitativos - o que os clientes gostam e o que eles não gostam - mas também quantitativos - quantos clientes usam os produtos e os acham valiosos. Estes dados proporcionam aprendizagens, a partir destes novos

conhecimentos, surgem novas ideias e hipóteses para testar. Este ciclo é a base do trabalho do método *lean startup*.

O ciclo construir - medir - aprender (CMA) representa-se assim:

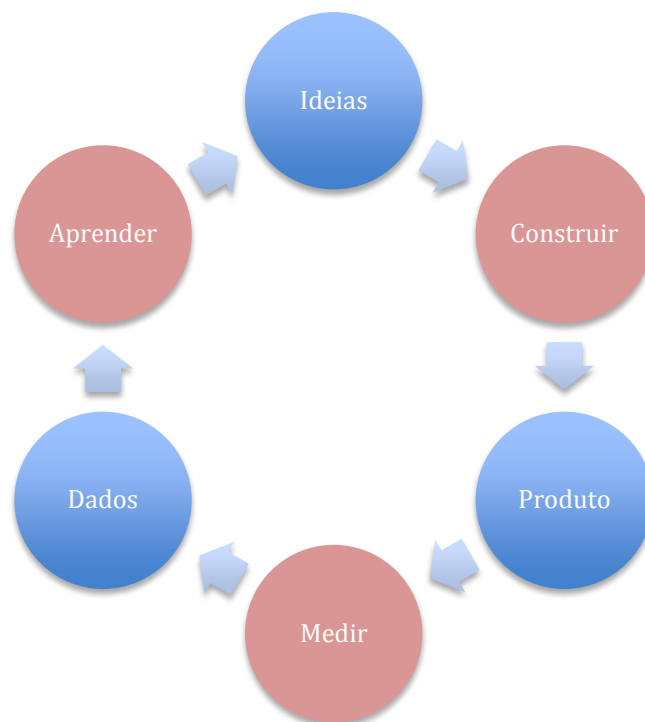


Figura 5: O ciclo Construir - Medir – Aprender

Fonte: Ries (2011)

Segundo Ries, a maioria das pessoas tem uma formação que enfoca em um dos elementos do ciclo. Por exemplo, os engenheiros aprendem construir produtos do jeito mais eficiente possível. Alguns gerentes ou vendedores são muito bons para tirar informações ou conhecimentos úteis dos clientes. Ou seja, para melhores resultados, é necessário criar uma equipe com pessoas de formação ou especialidades diferentes.

2.2.2.4. Construir o produto viável mínimo

Uma vez as primeiras hipóteses afirmadas ou negadas, o próximo passo consiste em construir o Produto Viável Mínimo (PVM) que tem como função continuar iterando segundo o ciclo CMA. O PVM é a versão do produto que permite uma volta completa do ciclo com um mínimo de esforço e o mínimo de tempo para desenvolver

o produto. O PVM não ajuda somente validar o design do produto ou as hipóteses técnicas, mas também testar as hipóteses fundamentais do lean canvas.

Os primeiros produtos não precisam ser perfeitos pois, segundo Blank, são adotados pelos “*early-adopters*”, um certo tipo de cliente que aceita, e até prefere, uma solução incompleta.

A complexidade do PVM não é escolhida por acaso. Segundo Maurya (2012), a regra é que tudo que não pode proporcionar um conhecimento relevante não deve aparecer no PVM.

O passo depois da construção do produto no ciclo CMA é a medida. A contabilidade clássica é um processo usado em todas as empresas grandes a fim de determinar objetivos para cada divisão. Essa abordagem não tem quase nenhuma utilidade para as *startups*. As previsões a respeito dos *startups* não podem ser precisas. Em vez de determinar objetivos em termos de números de clientes ou lucros, as *startups* devem determiná-los através de aprendizagem validada. É o que Ries chama de contabilidade da inovação. Isso permite à *startup* saber objetivamente se está mais perto do objetivo de crescer com um modelo viável.

A contabilidade da inovação é composta por três etapas. O primeiro passo é usar um PVM para obter dados reais sobre a situação da empresa. O segundo passo consiste em corrigir o “motor de crescimento” para chegar num ponto em que a empresa pode escolher entre perseverar ou *pivotar*, que é a última etapa. Se usar o pivot, a empresa deve começar novamente o início do processo. Ela sabe que deu certo se as correções do motor de crescimento são mais eficientes depois do pivot.

Segundo Maurya (2012), a ferramenta, *pirate metrics*, é muito relevante para medir se os objetivos da *startup* foram atendidos :

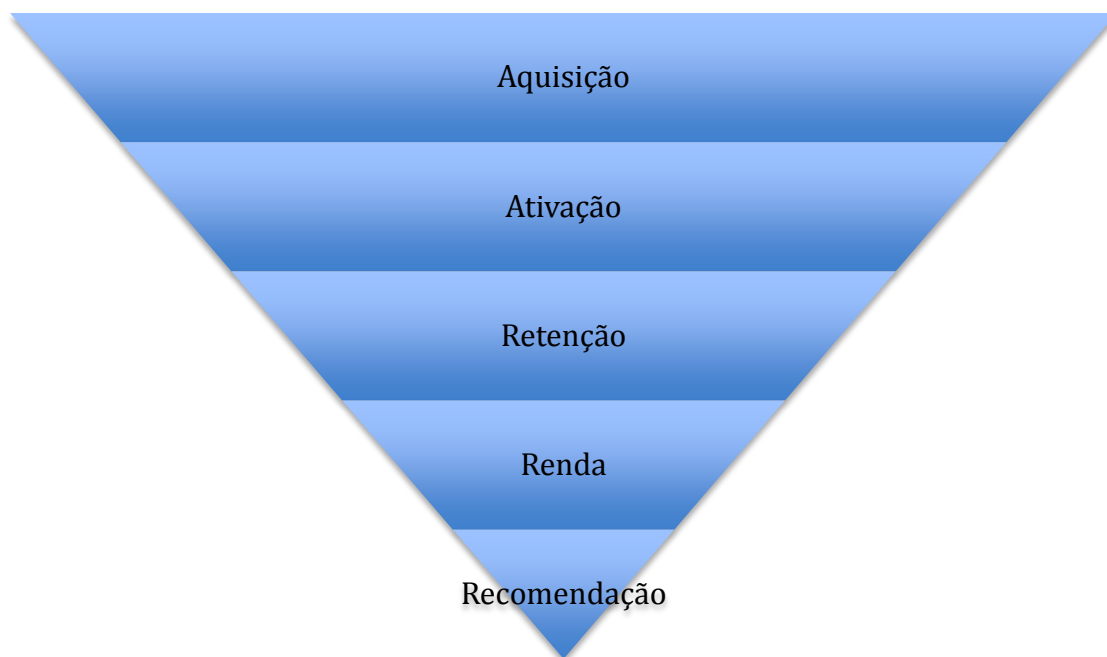


Figura 6: *Pirate metrics*

Fonte: Mc Clure, 2012

Trata-se de um conjunto de métricas que permitem medir progressos e identificar os pontos fracos ou fortes da *startup* durante o ciclo da vida do cliente. Essas métricas são :

Aquisição: É o momento em que a *startup* consegue transformar um visitante em cliente potencial.

Ativação: É o momento em que o cliente potencial realiza a primeira experiência boa dele com a empresa.

Retenção: Ela serve para medir as utilizações repetidas do produto ou serviço.

Renda: É o momento em que a *startup* é paga.

Recomendação: É o momento em que os clientes da empresa começam a recomendar o produto para outros clientes potenciais.

2.2.2.5. Crescer

Essas métricas, usando com outros testes, contato permanente com o cliente e o PVM, permitem acessar o terceiro item do ciclo PVM, a aprendizagem. Essa aprendizagem ajudará para ajustar o produto, ter novas ideias e começar um novo ciclo.

Depois de vários ciclos CMA e se a estratégia da *startup* for boa, as métricas

descritas vão aproximar-se, atingir ou até ultrapassar os objetivos da empresa. Começa o tempo para a empresa crescer, agora que achou um produto que sustenta a necessidade real do mercado.

O *lean startup* descreve técnicas para ajudar viralizar um produto ou serviço. O esquema geral não é diferente do que já foi apresentado. Trata-se de fazer hipóteses e testá-las para atingir objetivos em termos de número de clientes, entre outros. Portanto, não é o assunto deste trabalho, que foca nas primeiras fases da vida de uma *startup*.

2.3. Métodos de geração de ideia

2.3.1. Brainstorming

O *brainstorming* é uma técnica formalizada de resolução criativa de problema sob a direção de um coordenador. A técnica foi criada em 1939 por Alex Osborn, vice-presidente da agência de propaganda americana BBDO. Na origem, era um método de reunião de grupo preparada e usada para achar um número importante de ideias publicitárias e promocionais para os clientes e futuros clientes da agência. [MILLER, 2012]

Hoje, essa técnica é também usada fora do mundo da propaganda mesmo se o foco – achar um número importante de ideias originais – for o mesmo.

Dois princípios de base definem o *brainstorming*: a suspensão do julgamento e uma pesquisa mais ampla possível.

Estes dois princípios traduzem-se em quatro regras:

- Não criticar
- Não se limitar no que diz respeito a originalidade das ideias
- Combinar e aperfeiçoar as ideias dos outros
- Buscar o máximo de ideias possíveis

A técnica envolve 3 etapas principais

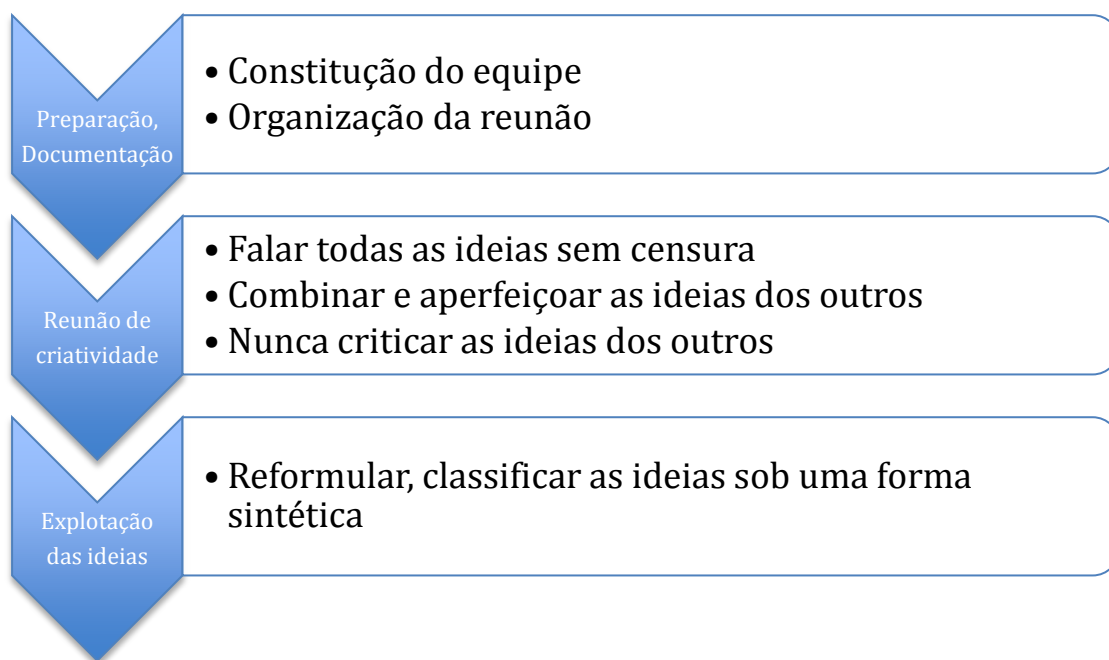


Figura 7 : O método do brainstorming

Fonte : Miller, 2012

2.3.2. Mapa de empatia

Segundo Osterwalder e Pigneur (2010), o mapa de empatia é uma ferramenta que permite entender o perfil de um “cliente alvo”.

Ter um mapa de empatia que abrange nomeadamente o ambiente do cliente, seu comportamento, suas preocupações e suas aspirações, permitindo ajudar nas escolhas estratégicas e na elaboração do *lean canvas*. [OSTERWALDER ET AL., 2010]

Os pontos chaves do mapa de empatia são:

O que o cliente vê?

Essa parte descreve o que o cliente vê no ambiente dele.

- Como é?
- Quem faz parte deste ambiente?
- Quem são seus amigos?
- Que ofertas já estão propostas?
- Quais são os problemas que ele enfrenta?

O que ele ouve?

Essa parte descreve como o ambiente influencia o cliente.

- O que falam os amigos do cliente?
- Quem o influencia realmente e como?
- Quais são as mídias em que ele confia?

O que ele pensa e sente realmente?

Essa parte foca o que acontece na mente do cliente.

- O que importa realmente para ele?
- Quais são as emoções dele?
- Quais são seus sonhos e aspirações?

O que o cliente fala e o que ele faz?

Essa parte é sobre o que o cliente fala ou como se comporta em público.

O que ele teme?

- Quais são suas maiores frustrações?
- Quais são os obstáculos que ele enfrenta para fazer o que quer?
- Quais são os riscos que ele pode temer?

O que ele espera?

- O que o cliente quer ou o que realmente quer alcançar?
- Quais são seus critérios de sucesso?
- Quais são suas estratégias para acertar os seus desejos?



Figura 8: Mapa de empatia

Fonte: Osterwalder & Pigneur, 2010

2.4. Ferramenta de análise estratégica

2.4.1. Análise PEST

A análise PEST – acrônimo para Política, Econômica, Social e Tecnológica – permite seguir os riscos e as oportunidades que poderia encontrar uma empresa e seu mercado. A particularidade dessa matriz é a sua visão global do ambiente. Ela tem enfoque em quatro grandes fatores:

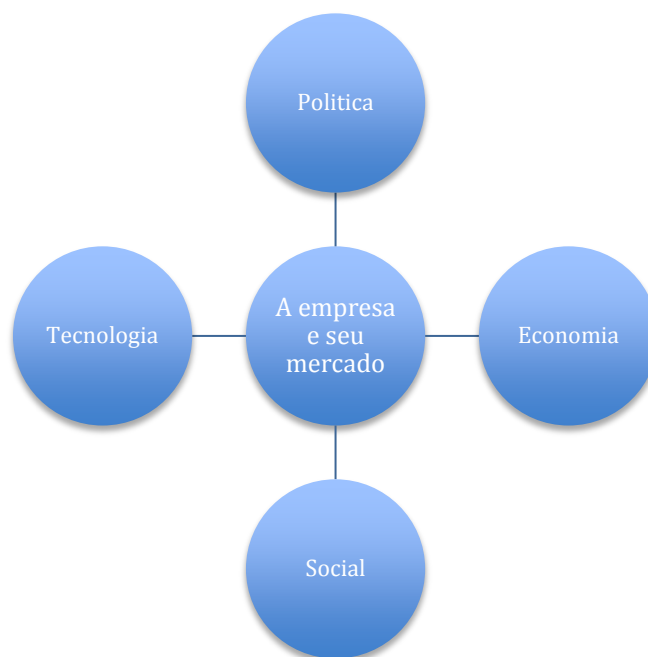


Figura 9: A análise PEST

Fonte: Realizado pelo autor deste trabalho

O ambiente político

Ele intervém em dois níveis: nacional com as decisões tomadas pelo governo (exemplo: subvenção, novas leis...) e internacional pelas decisões tomadas pelos outros países ou conjunto de países ou pelas instituições internacionais (FMI por exemplo).

O ambiente econômico

Essa parte da análise inclui todas as variáveis e fatores que influenciam o poder de compra e as despesas dos clientes ou fornecedores.

O ambiente social

Ele inclui as várias características da população (repartição por idade, estrutura familiar por exemplo) assim como as modas e tendências que poderiam influenciar a venda do produto ou serviço.

O ambiente tecnológico

Ele corresponde as forças que criam novas tecnologias, novos produtos e que influenciam direta ou indiretamente na capacidade das empresas de inovar.

2.4.2. As forças de Porter

O modelo das cinco forças de Porter foi elaborado em 1979 por Michael E. Porter, professor de estratégia na universidade de Harvard.

É uma ferramenta de análise estratégica do ambiente concorrencial de uma empresa. Segundo Porter, o desempenho da mesma depende da sua capacidade de enfrentar, influenciar e resistir às pressões do seu ambiente concorrencial. Com efeito, o objetivo principal de uma empresa é obter uma vantagem competitiva no seu mercado, o que pode ser medido pela sua capacidade em gerar lucros. [PORTER, 1998]

Este modelo tem como objetivo permitir identificar as forças presentes no ambiente concorrencial bem como suas intensidades, de adaptar a estratégia para obter uma vantagem competitiva, e assim realizar lucros maiores em comparação das outras empresas da área de atuação. [PORTER, 1998]

As 5 forças de Porter são :

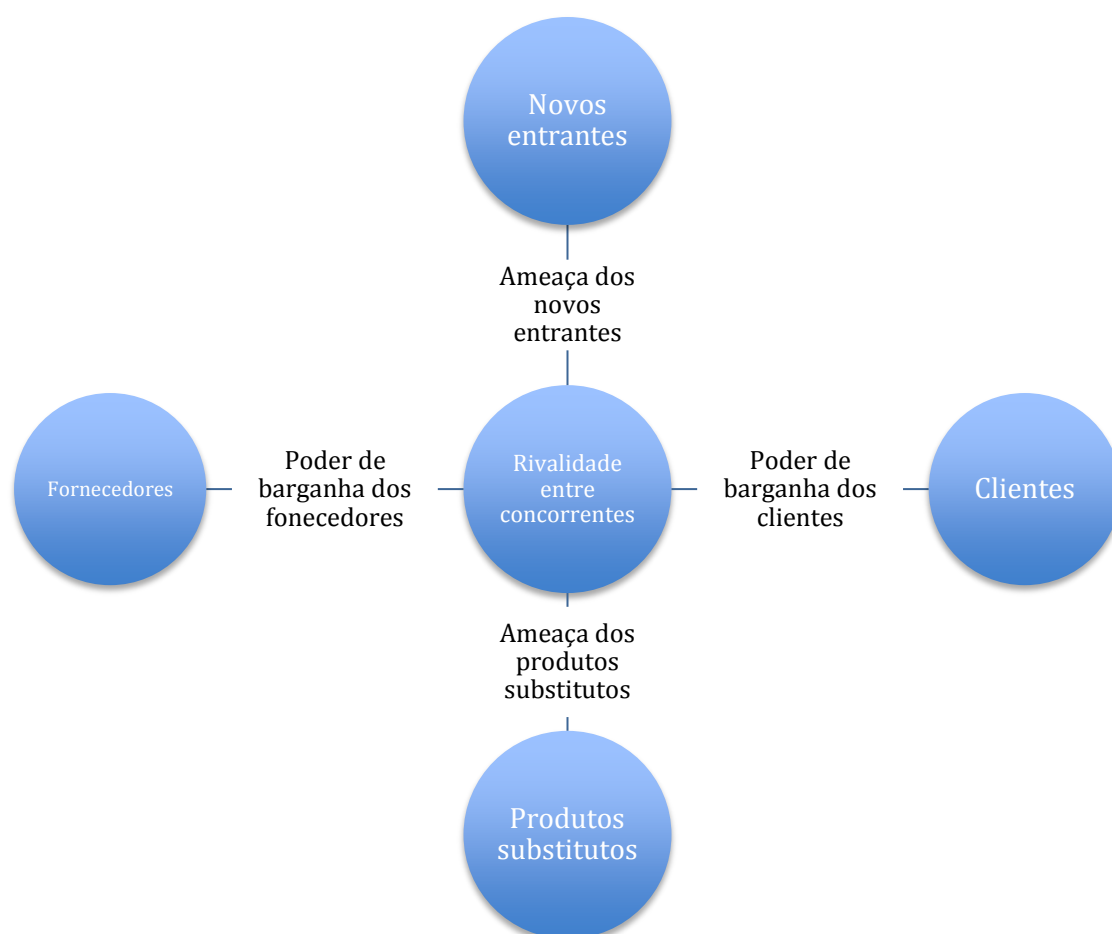


Figura 10: As 5 forças de Porter, Fonte: Porter, 1998

A rivalidade entre concorrentes

A concorrência entre as empresas, num mesmo setor, é um dos fatores que determina a atratividade dele. As empresas lutam para manter ou aumentar suas cotas no mercado. A concorrência evolui com o desenvolvimento do setor, sua diversidade ou a existência de barreira à entrada. Trata-se também de uma análise do número de concorrentes, dos produtos, da identidade das marcas, das forças e fraquezas, das estratégias, das quotas do mercado, entre outros.

A ameaça dos novos entrantes

Quanto mais houver novos entrantes, menor é a rentabilidade do setor. As barreiras à entrada protegem o mercado dos novos entrantes. Entre outras:

- Economia de escala
- Custo elevado à entrada
- Dificuldade para ter um acesso aos canais de distribuição
- Vantagens de custo não ligadas com o tamanho da empresa (contatos privilegiados com uma empresa chave por exemplo)
- Leis, regulamentações...

O poder de barganha dos clientes

Quanto mais elevado o poder de barganha dos clientes, menor fica a rentabilidade do setor, pois eles impõem as exigências deles em termos de preço, de qualidade ou de serviço. Entre outros, este poder fica maior quando:

- Tem compradores importantes no setor
- As empresas no setor são pequenas e indiferenciadas
- O custo para mudar de empresa é baixo
- Os clientes são unidos

A ameaça dos produtos substitutos

A existência de um produto substituto é uma ameaça para a empresa. Um tal produto pode surgir de uma inovação (por exemplo, o email X o fax) ou não (por exemplo, o leite X o suco de laranja). Quanto maior é a eficácia do produto substituto e menor seu preço, maior fica a ameaça.

O poder de barganha dos fornecedores

O poder de barganha dos fornecedores influencia também a rentabilidade do setor. Fornecedores poderosos podem impor com maior facilidade o preço, a qualidade, a quantidade do produto ou serviço prestado. Ao contrário, se houver numerosos pequenos fornecedores, o poder deles fica mais fraco.

2.5. Gestão de projeto informático: o método *Extrem Programming*

O *Extreme Programming* - XP - é uma disciplina de desenvolvimento de *software* baseada em 5 valores chaves : comunicação, simplicidade, coragem, feedback e respeito. Ela funciona ao levar toda a equipe juntas através de práticas simples e se insere perfeitamente na abordagem *lean*.

2.5.1. Os ciclos de desenvolvimento XP

Segundo Kent Beck e Cynthia Andres (2004), a abordagem tradicional baseada na sequência [Especificação => Concepção => Realização => Validação] concentra a maioria das decisões ao início do projeto.



Figura 11: Abordagem clássica de gerenciamento de um projeto de *software*

Fonte: Beck et al., 2004

A abordagem pode parecer justa: o cliente quer garantias sobre o que obterá ao final do projeto e o chefe de projeto deseja ter as informações necessárias para a organização da equipe.

Segundo Kent Beck e Cynthia Andres (2004), as equipes que evoluem em um ambiente imprevisível ou complexo sabem como é difícil ficar no caminho das decisões iniciais. O cliente conclui que as necessidades mudaram ou a equipe descobre na fase de implementação erros de especificação ou de concepção que comprometem os planos de desenvolvimento. A abordagem clássica supõe a ausência de mudanças e essa suposição revela-se finalmente dispendiosas.

Os criadores de XP descobriram práticas de organização de equipe e

programação que permitem deixar o *software* extremamente maleável – à tal ponto que vira mais vantajoso o fazer evoluir progressivamente do que buscar uma especificação e realizá-lo logo ao início. A partir dessa constatação, eles elaboraram uma abordagem que difunde o processo de decisão durante todo o projeto graças a uma sequência de ciclos iterativos curtos:



Figura 12: O ciclos da abordagem Extrem Programing

Fonte: Beck et al., 2004

O cliente ganha muito com essa abordagem. Ao invés de ver a intervenção dele só na fase inicial do projeto, ele integra totalmente o projeto para virar o piloto dele. Em cada iteração, ele escolhe as funcionalidades a implementar, colabora com a equipe para definir as necessidades em detalhes e recebe uma nova versão do *software* que integra as evoluções.

Essa abordagem apresenta varias vantagens em termos de gestão de projeto :

- O cliente tem uma grande visibilidade à respeito do avanço dos desenvolvimentos.
- O cliente usa o *software* como suporte de reflexão para as escolhas das funcionalidades a implementar
- A ordem de implementação das funcionalidades não é guiada pelas restrições técnicas, mas pelos pedidos do cliente. Este pode então focar os esforços da equipe nas funcionalidades mais importantes já no início do projeto.

A abordagem iterativa é portanto a principal vantagem do método XP.

O XP enfoque numa redução significativa da duração do ciclo de desenvolvimento, ou seja do tempo entre o momento em que é decidido implementar uma funcionalidade e o momento em que esta é colocada em produção.

Num projeto XP, este tempo corresponde exatamente na duração de uma iteração, ou seja tipicamente duas à três semanas:

- O primeiro da iteração é focado na reunião de planificação, em que o cliente e a

equipe concordam sobre o que tem que ser implementado na iteração. No final do dia, a equipe tem a disposição uma lista precisa de tarefas para realizar.

- Depois, a equipe organiza-se a fim de realizar as tarefas. Ela cuida do acompanhamento das tarefas, assim como as atividades de análise das necessidades, de concepção, de implementação e dos testes. É importante ter em mente que não tem mudança até o final da iteração.
- Depois dessas duas semanas, a equipe entrega uma nova versão do *software* ao cliente. Este *software* é robusto, testado e a estrutura dele é deixada limpa para que as próximas evoluções fiquem pouco caras.

Essa organização extremamente reativa impõe novas restrições a respeito da equipe, diferentes das praticas tradicionais de desenvolvimento de *software*. Por exemplo:

- Como o conteúdo do produto se decide só ao longo das iterações, não é mais conveniente definir a implementação do produto ao inicio do projeto. Nessa abordagem, a equipe tem que poder fazer emergir a concepção ao longo do desenvolvimento para seguir as direções dadas pelo cliente.
- Em termos de organização de equipe, não é mais conveniente separar módulos do *software* por programador, pois o cliente pode pedir para os mesmos trabalharem em um modulo só. Em suma, todos os programadores devem ter a possibilidade de trabalhar em qualquer parte do *software*

XP proporciona soluções concretas nessas problemáticas, com um conjunto de praticas que formam um sistema coerente e eficiente.

2.5.2. As valores chaves

Segundo Kent Beck e Cynthia Andres (2004), a abordagem XP é baseada sobre 5 valores chaves.

A comunicação

É o jeito fundamental para evitar os problemas. As práticas preconizadas pela abordagem XP impõem uma comunicação intensiva. Os testes, a programação em pares obrigam os desenvolvedores, os chefes e os clientes a se cominarem muito entre si.

A simplicidade

O jeito mais fácil de atingir os objetivos é também o melhor. Antecipar as extensões futuras é uma perda de tempo. É mais fácil fazer evoluir um aplicativo simples.

O feedback

A volta de informação é primordial para o desenvolvedor e o cliente. Os testes unitários indicam se o código funciona. Os testes funcionais dão o avanço do projeto. As entregas frequentes permitem testar as funcionalidades rapidamente.

A coragem

Algumas mudanças pedem muita coragem. As vezes é preciso mudar a arquitetura do projeto, jogar no lixo código a fim de produzir um código melhor ou tentar uma nova técnica. A coragem permite sair de uma situação inadequada. Pode ser difícil mais a simplicidade, o feedback e a comunicação fazer com que essas tarefas se tornem mais acessíveis.

O respeito

Cada programador pode trabalhar em qualquer parte do projeto e em particular partes escritos pelos outros programadores. Cada desenvolvedor tem que ter este fato em mente para escrever sempre um código de qualidade, que não dá trabalho demais ao próximo programador.

2.5.3. As praticas do XP

Essas cinco valores chaves se declinam em treze praticas que se fortalecem mutualmente.

Cliente no local

Um representante do cliente deve ser presente durante a vida do projeto no local de desenvolvimento. Ele deve ter os conhecimentos do usuário final e ter uma visão global do resultado a obter. Ele realiza o trabalho habitual dele mas fica disponível para atender qualquer pergunta da equipe.

Jogo da planificação

O cliente cria cenários para as funcionalidades que deseja obter. A equipe avalia o tempo necessário para colocá-lo em prática. O cliente seleciona depois os cenários em função das prioridades e do tempo disponível.

Integração perpetuel

Quando uma tarefa acaba, as mudanças são imediatamente integradas ao produto completo. A sobrecarga de trabalho, que está ligada à integração de todos os elementos antes da entrega, é portanto evitada. Os testes facilitam muito essa integração: quando os testes aceitam, a integração acaba.

Pequena entrega

As entregas devem ser feitas com uma frequência elevada, tipicamente 2 semanas. A integração permanente e os testes reduzem bastante o custo de entrega.

Ritmo viável

A equipe não faz horas a mais. Se for o caso, a planificação tem que ser avaliada de novo. Um desenvolvedor cansado entrega um trabalho de qualidade baixa.

Testes unitários

Antes de colocar uma nova funcionalidade, o desenvolvedor escreve um teste que verificará se o seu programa comporta-se como foi planejado. Este teste será conservado até o final do projeto, até que a funcionalidade não seja mais necessária. Em cada modificação do código, os testes escritos por todos os desenvolvedores são lançados, o que permite saber se alguma coisa não funciona.

Concepção simples

O objetivo de uma iteração é implementar os cenários selecionados pelos clientes e só isso. Considerar as próximas evoluções é uma perda de tempo. Os testes permitem mudar a arquitetura mais tarde se for necessário. Quanto mais simples fica o aplicativo, mais fácil fica a evolução dele nas próximas iterações.

Utilização de metáforas

Metáforas e analogias são usadas para descrever o sistema e seu funcionamento. O funcional e o técnico entendem-se melhor quando concordam nos termos que usam.

Refactoring

Melhora sistemática da qualidade do código sem modificar o seu comportamento. O código é refactorizado, não para o cliente, mas para a equipe mesmo. Ao ficar mais claro, um código de boa qualidade permite trabalhar de um modo mais eficiente e portanto mais rápido.

Apropriação coletiva do código

A equipe é coletivamente responsável pelo aplicativo. Cada desenvolvedor pode fazer mudanças em cada parte do código, mesmo aquelas que ele não escreveu. Os testes servem para indicar quando algo foi feito errado.

Padronização de código

Como todos os desenvolvedores trabalham no código inteiro, é indispensável estabelecer e respeitar normas de padronização para as variáveis, classes, métodos, objetos, entre outros.

Programação pareada

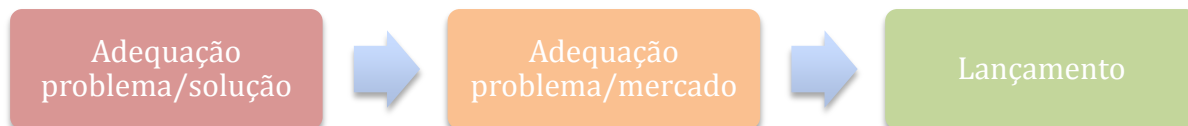
A programação é feita com par de desenvolvedores. O primeiro é chamado o “driver” ou piloto e tem o teclado. É ele que trabalha no código a escrever. O segundo chamado *partner* ajuda ao sugerir novas possibilidades ou ao detectar problemas. Os programadores mudam frequentemente de parceiro, o que permite melhorar o conhecimento coletivo do aplicativo assim como a comunicação dentro da equipe.

3. Metodologia

3.1. Esquema geral

O esquema geral deste trabalho pode ser apresentado assim:

Macro-etapas :



Micro-etapas:

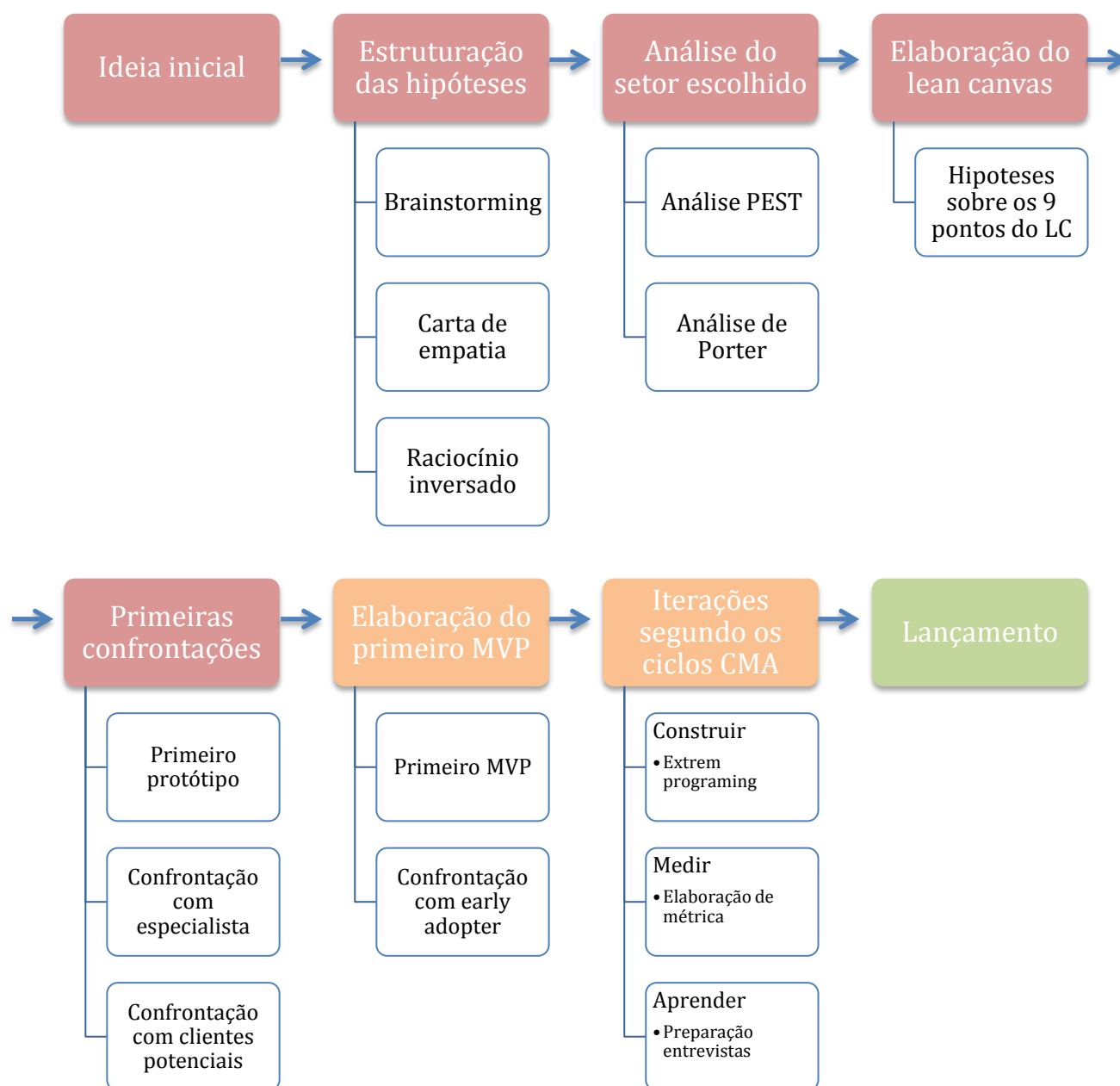


Figura 13: Esquema geral do trabalho, Fonte: Elaborado pelo autor

Este trabalho tenta proporcionar uma metodologia para acertar nas primeiras fases da criação de uma *startup* no setor da criação de *software*. Essa metodologia apoia-se principalmente no método *lean startup* desenvolvido por Steve Blank (2005), Eric Ries (2011) e Ash Maurya (2012). O trabalho de Alexander Osterwalder e Yves Pigneur (2010) sobre os modelos de negócios constitui um ponto de partida para a metodologia proporcionada.

Se o tempo for um elemento chave em qualquer projeto, é ainda mais importante nas primeiras fases de vida de uma *startup*, tendo em vista que até o lançamento do produto a empresa não gera receita. A *startup* deve achar o caminho até os clientes da maneira mais eficiente possível. É o objetivo da metodologia proposta neste trabalho.

Os principais passos são indicados na última figura. Seis meses foram necessários para realizar o conjunto dessas etapas. É também o tempo que se deu Ries para lançar o produto da empresa IMVU, um serviço de avatar online, e que ilustra o método *lean startup* no seu estudo (2011).

A metodologia proposta neste trabalho é ilustrada pela criação de um aplicativo móvel. No entanto, o produto em si e seus aspetos técnicos são secundários. O objetivo deste trabalho não é descrever ou explicar como criar um *software*, mas como criar uma *startup*, trata-se de um outro nível de abstração. Portanto, muitos elementos são apresentados de um jeito genérico. Isso também permite garantir a confidencialidade dos dados, que podem ser confidenciais. No entanto, este trabalho não pode permanecer teórico e alguns elementos de gestão de projetos próprio a criação de *software* são abordados, é nomeadamente o caso do método *extrem programming*. Essa exceção não impede o leitor interessado pela criação de uma empresa alcançando um outro tipo de produto ou serviço, entender os passos e os conceitos da presente metodologia.

3.2. O detalhe dos passos da criação da startup

A metodologia apresentada neste trabalho é composto por oito etapas. Como este é, em primeiro lugar, um trabalho de gerenciamento, cada etapa, além das explicações, é caracterizada por um tempo de execução e uma organização de equipe. Estes pontos são partes integrantes da metodologia mesmo se não aparecem na

descrição dos passos à seguir, pois são comuns a todos. No entanto, aparecerão na apresentação dos resultados.

3.2.1. Adequação problema solução

As 5 primeiras partes dessa metodologia servem para realizar o que Maurya (2012) chama de adequação problema/solução, ou seja, achar um problema que realmente vale a pena ser resolvido. Em cada uma dessas micro-etapas, o empregador deve avaliar se essa adequação faz sentido. A micro-etapa 5 constitui a conclusão dela.

Segundo Maurya (2012), um problema vale a pena ser resolvido se corresponde à três critérios :

- O produto/serviço interessa os clientes?
- Será que são inclinados a compra-lo? Do contrário, quem o compraria no lugar deles? (economicamente viável)
- Este problema pode ser resolvido? (é tecnicamente, legalmente, ... possível?).

3.2.1.1. Ideia inicial

Qualquer empresa é criada a partir de uma ideia. Essa ideia é o ponto de partida. É o único passo que não terá método associado em particular.

3.2.1.2. Estruturação das hipóteses

A primeira micro-etapa – a ideia inicial - permite ter uma primeira indicação do que a empresa quer criar ou do problema que ela quer resolver. No entanto, falta estrutura.

Essa segunda etapa tem, portanto, dois objetivos principais:

- Proporcionar de forma clara e precisa o problema a resolver, clientes potenciais, soluções possíveis e uma PUV, ou seja os quatros primeiros blocos do *lean canvas*.
- Aproveitar de um primeiro grupo de reflexão para avaliar o problema a resolver. Para servir estes objetivos, essa metodologia propõe usar técnicas de dinâmica

de grupo.

Três técnicas serão abordadas:

- O brainstorming
- O raciocínio invertido: consiste em imaginar a pior solução possível para um problema e inverter as características achadas.
- O mapa de empatia: ferramenta que ajuda ter um melhor conhecimento do ambiente do cliente, seu comportamento, suas preocupações e suas aspirações.

3.2.1.3. Análise do setor escolhido

Uma vez o problema, as soluções possíveis, os clientes alvos e a PUV clarificados, o próximo passo é estudar o ambiente em que a *startup* vai atuar. Com esta finalidade, dois estudos serão apresentados. O primeiro é a análise PEST, acrónimo de análise Política, Económica, Social e Tecnológica, e consiste num enquadramento de fatores macroambientais.

O segundo estudo é a análise de Porter, que destina-se à análise da competição entre empresa.

Essas análises tem 2 objetivos:

- Entender como funciona o ambiente do setor económico em que a *startup* tenta se inserir e como se articulam suas principais forças concorrenciais.
- Avaliar se as ideias proporcionadas nas partes 1 e 2 dessa metodologia são relevantes tendo em vista a análise do ambiente. Por exemplo, tal ideia poderia ser ilegal em tal país ou região do mundo ou um concorrente muito mais poderoso poderia proporcionar alguma coisa muito parecida.

É importante ter em mente que a *startup* MatchUp evolui num setor relativamente novo e o conjunto de conclusões que saiam dessa terceira parte da metodologia estão sujeito a importantes mudanças. Por causa disso, embora essas análises sejam importantes, elas terão um peso menor do que se a empresa MatchUp evoluísse numa área ou setor mais tradicional. É neste sentido que o método *lean startup* é relevante : segundo Ries (2011), uma hipótese, uma análise ou um estudo não tem valor antes de ser testados, o que será feito na micro etapa 5, “primeiras confrontações”.

3.2.1.4. Elaboração de lean canvas

As partes 1, 2 e 3 permitem levar varias hipóteses, em particular sobre os problemas dos clientes alvos, as soluções possíveis e a PUV. No entanto, o *lean canvas* fica incompleto. O objetivo desta parte é propor uma descrição global da empresa ao completar as caixas vazias dele.

Nessa metodologia, o *lean canvas* tem varias características:

- Segundo Osterwalder e Pigneur (2010), o *lean canvas* tem que ser conciso, caber numa pagina só. Isso permite destilar a essência do *lean canvas* e não se dispersar com elementos secundários.
- Segundo Maurya (2012), o *lean canvas* tem que focar no presente. Os modelos de negócios tradicionais tentam prever o futuro, mas o objetivo do método *lean startup* é aprender com hipóteses. Se a hipótese puder ser testada somente no futuro, ela não tem valor e não tem razão de ser no ciclo CMA, base do presente trabalho.
- Segundo Osterwalder e Pigneur (2010), existem varias estratégias para começar um *lean canvas*. O “epicentro” da inovação pode ser os recursos, a oferta, a finança... Na abordagem lean, como o cliente fica no centro da estratégia, ele é também o ponto inicial da construção do *lean canvas*. É inclusive por essa razão que o mapa de empatia tem uma importância na geração de novas ideias da parte 2 da metodologia.

3.2.1.5. Primeiras confrontações

Depois de ter completado o *lean canvas*, um primeiro protótipo é construído a fim de testar as hipóteses mais arriscadas dele e finalmente tirar uma conclusão sobre a adequação problema-solução. Um especialista e clientes potenciais são confrontados com o protótipo. Se essa confrontação revela-se positiva e a adequação problema-solução foi achada – o que é o objetivo dos cinco primeiros passos -, essa metodologia deste trabalho propõe ir para o próximo passo. No caso contrário, os empregadores devem escolher um dos outros *lean canvas* do passo quatro, tendo em conta as aprendizagens do passo cinco.

As aprendizagens da presente etapa permitem também determinar o conjunto mínimo de funcionalidades para criar o PVM do passo seis.

3.2.2. Adequação problema-mercado

As duas próximas partes dessa metodologia permitem achar o que Maurya (2012) chama de adequação problema-mercado. A questão fundamental nessa parte é de saber se a solução achada realmente atende ao problema do melhor jeito possível.

3.2.2.1. Elaboração do primeiro PVM

A partir das primeiras confrontações com clientes da parte 5 dessa metodologia, um PVM é construído. Ele constitui a primeira iteração de vários ciclos CMA da parte 6. O objetivo é poder confrontar um produto viável com os “early-adopter”. Como este trabalho apresentara, é este tipo de cliente que é o mais adequado para avaliar o caminho inicial de uma *startup* na adequação problema-mercado.

3.2.2.2. Iterações segundo os ciclos CMA

Os ciclos CMA da etapa 6 constituem o motor de crescimento da *startup*, ou seja, permitem trazer sempre mais conhecimento. Para reduzir ao máximo o tempo entre dois ciclos consecutivos, o método Extrem Programing foi aplicado.

A fim de medir os progressos, métricas serão definidas.

Finalmente, clientes são confrontados com o produto construído em cada ciclo.

Estes ciclos finalizam quando as métricas atingem os objetivos fixados. É tempo de lançar o produto no mercado.

3.2.3. Lançamento

O lançamento e sua estratégia de marketing não são descritos neste trabalho e aparecem aqui somente para perspectivar em um contexto maior.

4. Resultados

4.1. A ideia inicial

4.1.1. Composição da equipe e duração do passo

Uma pessoa sugeriu a ideia inicial e a duração deste passo é arbitrariamente escolhida como sendo um dia – é com efeito difícil dar uma duração ao processo de geração de uma primeira ideia. Isto marca o início do projeto MatchUp.

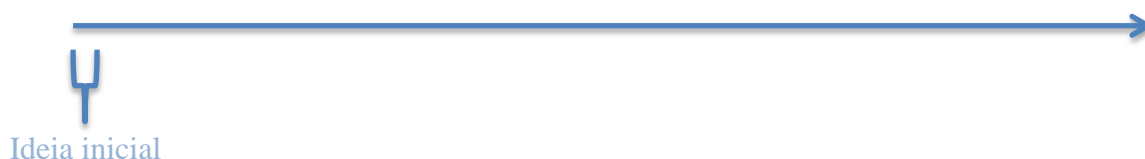


Figura 14: *Timeline* do projeto, Fonte: Elaborado pelo autor

4.1.2. Contexto

A ideia inicial foi sugerida pelo futuro chefe do projeto MatchUp. Nessa época, existiu vários aplicativos móveis de dating, ou seja, usados pelos usuários para encontrar novas pessoas. Apesar do sucesso de um em particular – Tinder – e da multidão de aplicativos parecidos [YODER, 2014], nenhum de todos pareceram agradar totalmente os usuários. O potencial em termo de tamanho de mercado apareceu como gigante. A ideia era portanto criar um produto que atenderia realmente as necessidades e desejos dos usuários.

4.1.2.1. Benchmark: o caso Tinder

A base de inspiração da nova ideia era o aplicativo Tinder. Este é um aplicativo de dating lançado em setembro de 2012 por 4 americanos. Tinder recusa revelar o número exato de usuários, mas precisava ter recebido um milhão de novos usuários nos Estados-Unidos durante os 60 dias entre dezembro 2013 e janeiro 2014. O aplicativo faz desfilar os perfis de usuários segundo diferentes critérios, inclusive o sexo e a posição geográfica. O usuário tem que indicar se ele os curtem ou não pelo intermediário de um “*swipe*” – movimento do dedo pela direita ou esquerda na tela. Quando a atração é recíproca, os dois usuários podem começar a conversar. [STAMPLER, 2014]

Os fatores chaves do sucesso do Tinder pareceram ser:

- A simplicidade para se cadastrar, menos de 5 segundos via Facebook.
- A simplicidade de utilização com o movimento “*swipe*”.
- A impossibilidade de receber mensagens de pessoas que não foram aprovadas pelo usuário antes.
- A possibilidade de escolher uma distância máxima entre os outros usuários e um segmento de idade alvo.

4.1.3. Detalhes da ideia inicial

Considerando estes dados, a primeira ideia era criar um aplicativo de paquera simples e intuitivo, com as seguintes funcionalidades:

- Conjunto das funcionalidades do Tinder, descritas na parte 4.2.1.2 deste trabalho.
- Possibilidade de mudar de localização, para ver os perfis de pessoas de outras cidades do mundo.
- Possibilidade de criar filtros a fim de ver perfis mais relevantes.

4.2. Estruturação das hipóteses

4.2.1. Objetivo da etapa

A primeira ideia constitui o ponto inicial da presente metodologia. Mas esta é pouca estruturada.

O objetivo é portanto construir um conjunto de hipóteses coerentes e colocá-las em um *lean canvas* a fim de testá-las numa primeira confrontação com clientes.

Para construir este conjunto de hipóteses, Osterwalder e Pigneur (2011) recomendam usar técnicas de grupos que envolvam o pensamento visual. Segundo os dois autores, essa técnica permite, através da utilização de desenhos e Post-itTM, alimentar as discussões ao construir algo que faz sentido. Isto melhora a lógica e a coerência do conjunto de ideias propostas. Como o objetivo dessa primeira macro etapa, adequação problema-solução, é construir um *lean canvas* para testar na segunda macro-etapa, é importante notar a este respeito, que o modelo proposto só pode ganhar em clareza ao usar um pensamento visual, assim como mostrar melhor as articulações que o anima.

Precisamente, o objetivo nessa etapa é achar em primeiro lugar quais serão os clientes do aplicativo. Segundo Blank (2006), é a etapa mais importante para todos os criadores de empresa. Osterwalder e Pigneur (2011) recomendam usar um mapa de empatia a fim de realmente entender o cliente e assim ficar com uma certeza maior que constitui um alvo certo para a *startup*, assim como, se for o caso, poder propor um *lean canvas* coerente com este cliente. Segunda Maurya (2012), uma vez este identificado, assim como seus problemas e as soluções as quais ele já tem acesso, uma proposição única de valor deve ser proposta. Finalmente, as funcionalidades essenciais do produto serão definidas, graças ao brainstorming e a técnica do raciocínio invertido.

4.2.2. Composição da equipe e duração da etapa

Seis pessoas participaram do brainstorming, com duração de um dia. O autor deste trabalho lidou as discussões.

Geração de novas ideias



Figura 15: *Timeline* do projeto, Fonte: Elaborado pelo autor

Segundo Osterwalder e Pigneur (2011), para tirar ao máximo benefício dessas técnicas, é recomendado ter uma equipe diversificada em termos de idade, conhecimento dos clientes potenciais, competência profissional. Além disso, os autores recomendam também aos participantes estudar o assunto da reunião antes que esta comece. É neste sentido que os membros da equipe foram selecionados para participar deste processo. Cada participante tinha também que passar a metade de um dia se informando sobre o que pode ser interessante ao mercado do dating móvel. Isto pode parecer vago, mas permite a cada participante não ter os mesmos dados no momento da reunião e permite garantir a diversidade da equipe.

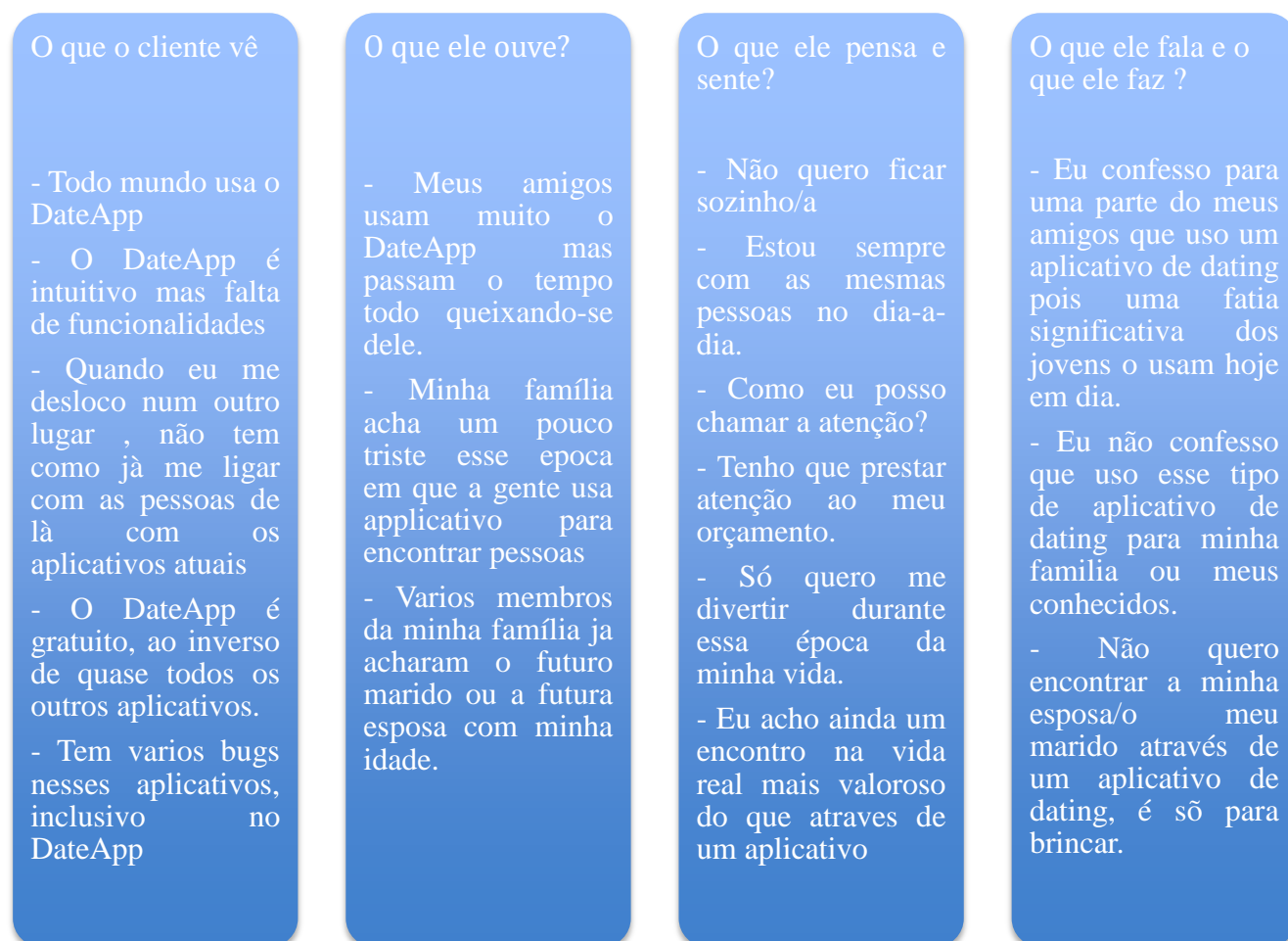
4.2.3. O cliente do aplicativo e seus problemas

Um primeiro brainstorming é organizado a fim de achar vários segmentos de cliente alvo. Três segmentos foram selecionados :

- Os estudantes de 18 à 25 anos.
- Os jovens ativos, solteiros, de categoria A/B.
- Os jovens de 18 à 30 anos, que se deslocam regularmente.

Para cada um deles, um mapa de empatia é realizada. A próxima figura é a da última proposta - os jovens de 18 à 30 anos, que se deslocam regularmente – segmento de cliente finalmente selecionado no final da macro etapa um da metodologia deste trabalho.

O mapa de empatia é realizado na primeira pessoa. [OSTERWALDER E PIGNEUR , 2011]



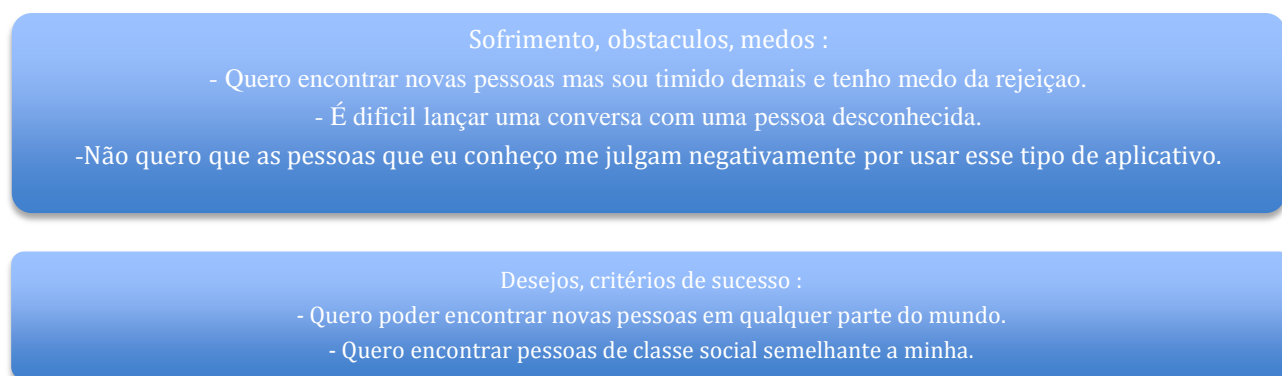


Figura 16: Mapa de empatia para o segmento de cliente dos jovens de 18 à 30 anos, que se deslocam regularmente

Fonte: Elaborado pelo autor

O mapa de empatia permite formular hipóteses sobre os problemas enfrentados pelo usuário selecionado, as soluções as quais ele já tem acesso e seus limites, ou seja os blocos 1 e 2 do *lean canvas*.

Sobre o segmento de cliente

Trata-se dos jovens de 18 a 30 anos que se deslocam regularmente, uma vez por mês ou mais, por vários motivos, entre outros:

- O cliente é um estudante que estuda na semana numa cidade e mora com a família em outra no fim de semana.
- O cliente é um jovem ativo que se desloca muito por causa do trabalho.
- O cliente é um jovem que viaja regularmente por prazer.

Além deste critério, o cliente alvo é um celibatário querendo fazer novos encontros em qualquer lugar que ele frequenta seguindo, por exemplo, os critérios supra mencionados.

Sobre os problemas enfrentados

Maurya (2011) recomenda listar os principais problemas enfrentados pelo segmento de cliente, por ordem de importância.

- O medo da rejeição e a timidez são obstáculos psicológicos fortes no processo de fazer novos encontros. (Problema A)
- As soluções atuais não integram a possibilidade do usuário deslocar-se em qualquer parte do mundo. (Problema B)

- Os clientes querem ver perfis mais adequados com o seu nível de instrução ou classe social. (Problema C)
- É difícil iniciar uma conversa com um desconhecido (Problema D)
- O usuário não quer que seu perfil seja visto por uma parte de seus amigos ou conhecimentos (Problema E)

Sobre as soluções já existentes

Tinder constitui o principal concorrente e o passo três da primeira macro etapa permite estudar com profundidade o ambiente de atuação da empresa MatchUp.

Sobre os early adopters

Segundo Ries (2011), o método lean foca-se no presente e portanto, ao início, nos “early adopters” do produto, eles constituem o grupo mais receptivo ao produto. Eles têm as mesmas características do que o cliente alvo supra descrito, mas além disso, é brasileiro – o primeiro mercado atacado – e adepto às novas tecnologias e às novidades em termo de aplicativos, ele usa diariamente Tinder ou uma solução semelhante.

4.2.4. As soluções imaginadas

Uma vez o tandem “cliente – problemas” clarificado, as soluções para resolver os problemas destacados são propostas. Trata-se do bloco quatro do *lean canvas*. Existem várias técnicas de grupos que permitem, a partir de um problema, achar soluções pertinentes. Entre outros, as equipes podem explorar: o *brainstorming*, o método dos 6 chapéus e o método 6-3-5. [MILLER, 2012] A equipe responsável pelo projeto MatchUp usou o método do raciocínio revertido, associado com um *brainstorming*. Essa técnica foi nomeadamente usada pela empresa SNCF – empresa ferroviária francesa – para melhorar o serviço aos clientes, e também pela Ecole Centrale de Paris – escola de engenharia francesa – para achar ideias inovadoras na construção de um novo campus destinado aos estudantes. [CHENEVIER et al., 2014] Ela consiste em tentar de imaginar para cada problema a pior solução que existe. Basta depois de inverter suas características para possuir uma solução que afasta-se ao máximo do que o usuário não quer.

Essa técnica apresenta, quando associado com um brainstorming, numerosas vantagens :

- Por concentrar-se no que o usuário realmente não quer, o raciocínio revertido permite pensar e proteger-se das futuras críticas no diz respeito ao produto.
- Ele é complementar do brainstorming que concentra-se em que o usuário realmente quer para resolver os seus problemas.
- Por essência, as ideias propostas com este técnica são absurdas, pois tem que definir as piores soluções. Isto permite a equipe encarregada neste processo livrar-se das críticas ou dos julgamentos que constituem um freio para a inovação.

O resultado da técnica do raciocínio revertido, associado com o brainstorming aparece na tabela seguinte:

Problemas principais	Solução do raciocínio inversão	Solução do brainstorming
<i>Problema A</i> O medo da rejeição e a timidez são obstáculos psicólogos fortes no processo de fazer novos encontros	O usuário recebera uma notificação cada vez que outro não mostrou interesse nele.	O aplicativo proporá anonimamente perfis ao usuário.
<i>Problema B</i> As soluções atuais não integram a possibilidade do usuário de deslocar-se em qualquer parte do mundo	O usuário será bloqueado na posição atual dele.	O usuário terá acesso à um mapa e um calendário a fim de informar os deslocamentos dele.
<i>Problema C</i> Os clientes querem ver perfis mais adequados com o seu nível de instrução ou classe social	O usuário será exposto unicamente aos perfis menos relevantes no diz respeito aos desejos dele.	O usuário terá que informar os estudos e trabalho dele.

Problema D É difícil iniciar uma conversa com um desconhecido	O aplicativo incitará, do jeito mais frio e menos divertido, entrar em contato com um outro usuário. Os interesses em comum entre dois usuários que mostraram uma atração recíproca serão indicados.
Problema E O usuário não quer que seu perfil seja visto por uma parte de seus amigos ou conhecimentos	Nada será feito de jeito anônimo, o aplicativo mandará notificações nas redes sociais (principalmente Facebook e Twitter) do usuário. O usuário terá a possibilidade de bloquear os perfis indesejados

Tabela 2: Principais problemas e soluções achadas

Fonte: Elaborado pelo autor

A partir deste conjunto de dados, a solução selecionada é:

- O aplicativo propõe vários perfis ao usuário, que pode aceitar ou rejeitar cada um deles. O usuário nunca sabe se ele foi rejeitado, ele recebe uma notificação só quando acontecer uma atração mutual. (Solução ao problema A)
- O usuário deve informar ao aplicativo de seus planos em termo de deslocamento a fim de propor perfis correspondentes (Solução ao problema B)
- O usuário deve criar um perfil com nomeadamente o estudo dele, assim como o seu trabalho atual. O usuário terá a opção de ver perfis correspondentes com critérios definidos. (Solução ao problema C)
- Uma integração com Facebook permite conhecer os interesses de cada usuário. Além disso, cada usuário terá que informar o aplicativo sobre um fato insólito e engraçado sobre ele. Antes de começar uma conversar, este fato é mostrado ao outro usuário. (Solução ao problema D)

- O usuário tem a possibilidade de bloquear perfis indesejados. Além disso, o aplicativo informará ao usuário que nunca haverão publicações em seu nome em qualquer rede social. (Solução ao problema E)

4.2.5. A proposta única de valor do aplicativo

Segundo Maurya (2011), o próximo passo é determinar a proposta única de valor da empresa. A PUV é o que constitui a diferença do produto, o que faz dele um produto único que merece a atenção do mercado. Essa PUV tem que se apoiar dos problemas selecionados.

Segundo Al Ries e Jack (2001), é mais importante focar nos benefícios do que nas funcionalidades. Neste sentido, a PUV eficiente é composta de três componentes: o benefício final para o consumidor (obrigatório), um período preciso (facultativo) e uma resposta às objeções (facultativa). Por exemplo, a PUV de Domino's pizza – maior rede de entrega de pizza do mundo – é: Sua pizza quente entregue em 30 minutos ou você não pagará pela mesma.

Segundo Maurya (2001), a PUV tem que responder as três perguntas: em que consiste seu produto, quais são seus clientes e porque eles têm que o comprar?

Considerando estas duas abordagens, a PUV selecionada para o produto MatchUp é:

“Encontrar novas pessoas, onde quer que esteja”

A PUV insere-se no bloco 3 do *lean canvas*.

4.3. Análise do setor escolhido

4.3.1. Objetivo dessa etapa

A etapa anterior permitiu levar várias hipóteses sobre o segmento de cliente alvo, os problemas que ele enfrenta, as soluções possíveis. Este conjunto permitiu definir finalmente uma PUV.

Essa etapa é dedicada a entender melhor o ambiente da empresa. Ou seja, são apresentadas: a análise PEST e a das forças de Porter, que são ferramentas chaves para essa análise estratégica.

4.3.2. Composição da equipe e duração da etapa

Uma pessoa, o autor, participou dessa etapa, de duração de um dia.



Figura 17: *Timeline* do projeto, Fonte: Elaborado pelo autor

4.3.3. Análise PEST

A análise PEST é uma ferramenta dedicada a identificar a influência – positiva ou negativa – que podem exercer, sobre uma organização, os fatores macroambientais. Como o early adopter é brasileiro, essa análise foca no Brasil.

4.3.3.1. Os fatores políticos

Por um lado, o Brasil é relativamente mal classificado ao nível mundial no que diz respeito a facilidade de criar uma empresa. No mês de junho de 2014, o World Bank Group, maior banco de desenvolvimento do mundo, classificou o Brasil no 120º lugar sobre 190 países analisados. Em particular, ele tem a posição 167 na categoria *starting a business*. [WORLD BANK GROUP, 2014]

Por outro lado, desde julho de 2014, o Brasil impõe aos fabricantes de celulares colocar aplicativos nacionais neles. Desde o início de julho, os aparelhos precisam oferecer pelo menos 30 aplicativos nacionais. Em dezembro, este número saltou para 50. [MINISTERIO DAS COMUNICAÇÕES DO BRASIL, 2014]

4.3.3.2. Os fatores econômicos

Por um lado, os anos 90 (crescimento médio de 1,9% por ano) e 2000 (3,4 %) foram prósperos para o Brasil, o que permitiu ao PIB aumentar 68%, reduzir as desigualdades e desenvolver uma classe média de 100 milhões de pessoas.

Por outro lado o desempenho econômico desde o ano 2011 freou. (2,5% em 2011, 1% em 2012, 2,5% em 2013, previsões : 0,2% em 2014 , 0,8% em 2015).

Como foi o caso dos anos 90 e 2000, um crescimento robusto permite um crescimento da classe média e assim do número de habitantes possuindo um celular, o que aumenta o número potencial de usuário do aplicativo. [THE WORLD BANK, 2014]

4.3.3.3. Os fatores sociais

Idade e sexo

O Brasil tem aproximativamente, em 2014, 202 milhões de habitantes. Em 2010, 18% da população tinha entre 20 e 29 anos de idade, ou seja 34 milhões de pessoas (36 milhões em 2014). Tem aproximativamente o mesmo número de mulheres e de homens para essa parte da população (49.6% H / 50.4% M) [INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA, 2014]

Classe social

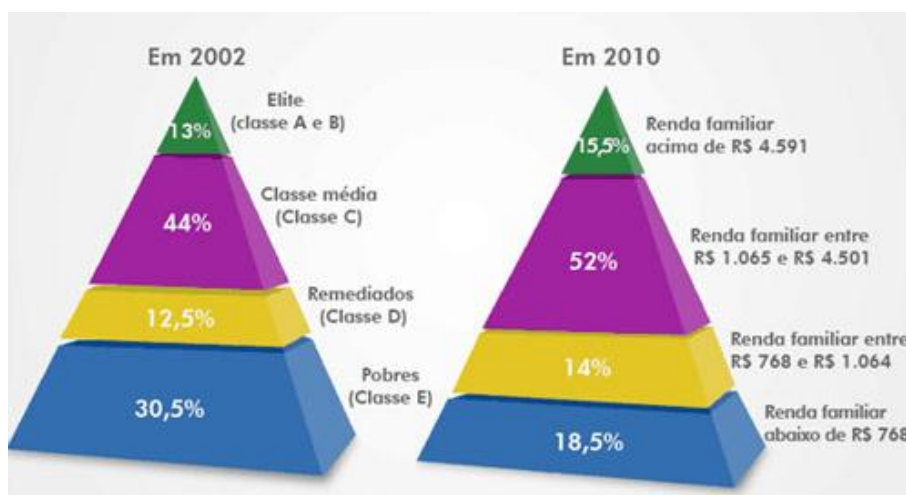


Figura 18 : As classes sociais no Brasil, classificação em 5 classes

Fonte : FGV, 2011

Segundo à FGV, a pobreza despencou desde 2002. Com isso o miolo da pirâmide engordou e agora é maioria absoluta.

Status de relacionamento

Segundo a Pesquisa Nacional por Amostra de Domicílio (PNAD) de 2013, considerando a população de 15 anos ou mais, 49,2% do total eram solteiros, 38,6% casados e o resto é divorciado, desquitado ou separado judicialmente.

Habito com celulares

Em 2014, segundo o *MMA Nielsen Mobile Report Brasil* (2014):

- Os donos de celulares são majormente da classe A(13%) e B(45%). C e D representam 39% e E, o resto, 3%.
- As pessoas entre 18 e 34 anos de idade representam quase a metade dos usuários (49%)
- A maioria dos usuários de celulares trabalha em tempo integral (55% deles) ou são estudantes (26%)
- No mês de fevereiro, o smartphone era, para 33% dos usuários, a principal forma de acesso na internet. Há cinco meses atrás era para 25%.

4.3.3.4. Os fatores tecnológicos

Forma de acesso à internet

O número de conexões móveis cresceu de 233 mil (em 2007) para 7,2 milhões em 2013. A projeção é de 9,3 milhões em 2014, superando os 15 milhões depois dos Jogos Olímpicos.

Em 2012, 3% dos internautas brasileiros tem uma velocidade de banda larga de 128 a 512 Kbps; 45% tem 512 Kbps a 2 Mbps; 27% usa 2 Mbps a 8 Mbps. No 3º trimestre de 2013, a velocidade média da conexão atingiu 2,7 Mbps. [IBOPE MEDIA, 2013]

No mês de fevereiro de 2015, 70.9% dos municípios do Brasil tinham acesso à 3G, o que representa 92.1% da população coberta. [TELECO, 2015]

Ferramenta de criação de software

Um número importante de ferramentas de criação de *software* são livres e gratuitos de utilização, é particularmente o caso do:

- Serviço integrado de cadastro pelo Facebook
- Serviço de localização e de mapa via Google Maps
- Serviço de mensagens via Whatsapp

4.3.4. Análise de Porter

Esta é uma ferramenta de análise estratégica do ambiente concorrencial de uma empresa. As 5 forças são:

Os clientes

O custo para mudar de empresa/aplicativo é baixo, até inexistente. No entanto, a empresa MatchUp tenta criar algo novo e único que permite abaixar o peso deste componente do poder de barganha. Além disso, não há clientes que têm uma importância capital para a empresa, todos possuem um peso igual e são numerosos.

Um ponto importante no estudo de um aplicativo social é o efeito de rede, ou seja o fenômeno pelo qual a utilidade real de uma técnica ou de um produto depende da quantidade dos usuários. Segundo a lei de Metcalfe, se a utilidade for proporcional com o número dos usuários, isso significa que o valor da rede é proporcional ao quadrado do número de usuários. [KLEINBERG et al., 2010] Portanto, o poder de barganha dos clientes é tanto mais fraco quando o aplicativo tem muitos usuários.

Os fornecedores

Existem diferentes tipos de fornecedores:

- Fornecedores de ferramenta de criação de *software* : Quase todas as ferramentas usadas são gratuitas e livres de utilização. Existem muitos fornecedores.
- Fornecedores de acesso aos clientes, lojas virtuais: Existem três sistemas operacionais principais: *Android* (95% do mercado brasileiro), *Windows Phone* (6%) e *iOS* (4.7%). O valor destes

sistemas é ligado ao número de aplicativos compatíveis. [HIGA, 2014]

Portanto as lojas incentivam todos os programadores a desenvolver seus aplicativos para o seu sistema operacional.

Os produtos substitutos

Os produtos substitutos são os websites de paquera. No entanto, os smartphones tornam-se sempre mais os meios privilegiados de acesso à internet [29]. Além disso, um certo número de funcionalidades úteis para uma ferramenta de paquera – geolocalização por exemplo - são acessíveis unicamente graças a um smartphone.

Os novos entrantes

Para criar um aplicativo, um computador e uma conexão de internet são suficientes. Além disso, é fácil colocar um aplicativo nas lojas virtuais *Android*, *iOS* ou *Windows Phone*. Portanto, as barreiras na entrada são poucas. No entanto, o efeito de rede é primordial para este tipo de aplicativo. A massa crítica a atingir é o principal freio e as despesas, em marketing principalmente, para atingí-la podem ser bem maiores do que o custo do desenvolvimento do aplicativo em si.

Rivalidade entre concorrentes

Quatros aplicativos de dating chegaram a uma massa de usuários superior à 10 milhões de usuários : Tinder, Lovoo, Badoo e MeetMe. Existem inúmeros aplicativos de dating com menos usuários. [YODER, 2014]

O conjunto dessas cinco forças de Porter permite criar um gráfico que representa a intensidade destes componentes no ambiente dos aplicativos de paquera.

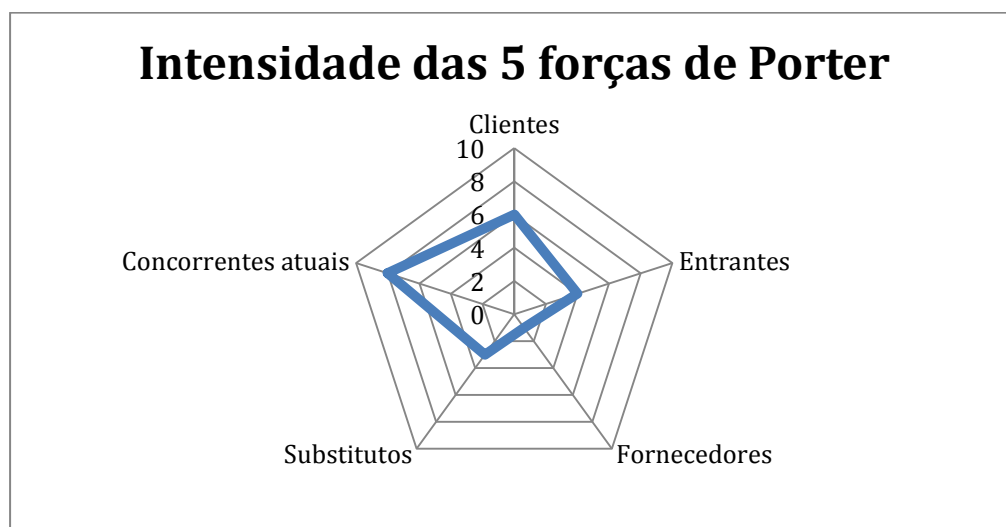


Gráfico 1: Intensidades das cinco forças de Porter no setor dos aplicativos de dating

Fonte: Elaborado pelo autor

4.3.5. Lições retiradas pelo projeto

Privilegiar os jovens entre 18 e 30 anos aparece como uma estratégia interessante, já que são os principais usuários de smartphones. Além disso, ao privilegiar os jovens que se deslocam frequentemente, a empresa foca nos brasileiros de classe A e B, os que têm mais acesso aos smartphones mais recentes, proporcionando funcionalidades interessantes para um aplicativo de dating.

As infraestruturas para ter um acesso de boa qualidade na internet a partir de um smartphone melhoram cada vez mais no Brasil, o que torna coerente uma primeira implementação neste país. Além disso, o programa de incitamento a criar aplicativo nacional dá uma chance maior ao aplicativo de atingir mais usuários.

A população do Brasil, sendo grande, permite, mesmo ao focar numa parte restrita dela, alcançar um número enorme de usuários – 36 milhões entre 20 e 30 anos de idade e 600 000 ao considerar apenas as classes A e B.

A principal dificuldade que aparece é vencer o efeito de rede e atingir uma massa crítica suficiente para tornar o aplicativo relevante para os usuários. Com efeito, o poder dos fornecedores, dos novos entrantes e dos produtos substitutos parece baixo. Os concorrentes sérios são aqueles que conseguiram justamente atingir essa massa crítica.

4.4. Elaboração do lean canvas

4.4.1. Objetiva dessa etapa

Trata-se da penúltima etapa do primeiro macro passo da metodologia presente. Através da utilização de um *lean canvas*, essa etapa deve propor uma descrição global da *startup*. O conjunto das hipóteses inscritas neste *lean canvas* tem que ser testado na próxima macro etapa da metodologia.

4.4.2. Composição da equipe e duração da etapa

Uma pessoa, o autor, participou dessa etapa, de duração de um dia.

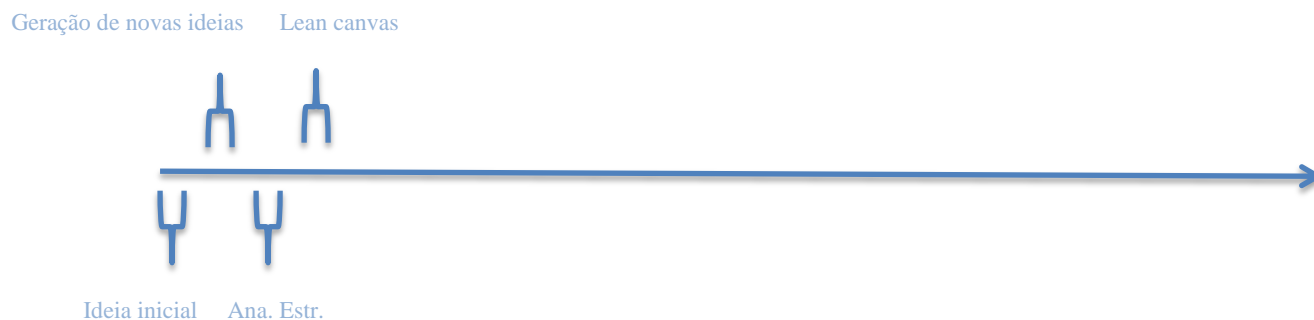


Figura 19: *Timeline* do projeto, Fonte: Elaborado pelo Autor

4.4.3. Clientes, problemas, soluções e PUV

Os quatro primeiros blocos já foram propostos na micro etapa 2 da metodologia e serão diretamente colocados no *lean canvas* final.

4.4.4. Canais de distribuição

Um dos critérios chaves de sucesso do aplicativo MatchUp é a capacidade dele atingir uma massa crítica suficiente.

Os canais naturais de distribuição são as lojas virtuais de aplicativos dos principais sistemas operacionais: Google Play para Android, AppStore para iOS e a loja de Windows Phone.

Os canais de viralização selecionados são: a rede social *Facebook* e o sistema de propaganda *Google AdWord*. Este conjunto de ferramentas tendo que mandar o cliente potencial no website de apresentação do aplicativo, que permite o baixar.



Figura 20: *Lean canvas* completo

Fonte: elaborado pelo autor

4.4.5. Fluxo de caixa

Ainda segundo a ideia do que o efeito de rede é o componente mais crítico de um tal aplicativo de dating, parece importante não desencorajar os primeiros usuários a baixar e usar o aplicativo ao cobrar, enquanto as principais empresas têm uma política Premium – ou seja, a utilização do aplicativo é gratuita, porém limitada às funcionalidades mais básicas, o resto sendo cobrado. Neste fim, o usuário poderá usar o aplicativo gratuitamente, mas será limitado à 5 conversas simultâneas e 2 destinos onde pode encontrar novas pessoas, além de seu lugar de origem. Se o usuário quiser desbloquear o aplicativo e tiver um acesso ilimitado à todas as funcionalidades, ele terá que pagar 10 reais por mês, o que corresponde a uma política de preço parecida com os concorrentes do setor.

4.4.6. Estrutura dos custos

Os custos são calculados de uma maneira simples e permitem ter uma primeira ideia da massa de usuários necessária a fim de atingir o limiar de rentabilidade.

Os custos são:

- Os custos dos servidores: Solução *Amazon AWS* : 1 R\$/100 usuários/mês
- Os custos dos recursos humanos: $4 * 5.000 = 20.000$ R\$/mês
- Os custos diversos (aluguel, eletricidade, computadores...): 4.000 R\$/mês

Se 1% dos usuários se cadastram no Premium, isso permite gerar um fluxo de caixa de: 10 \$R /100 usuários/mês. Ou seja, sem considerar o custo de aquisição do usuário, o limiar de rentabilidade é atingido com 270 000 usuários.

4.4.7. Métricas

As métricas usadas são as *pirate metrics* descritas por Dave McClure:

Aquisição: O usuário baixa o aplicativo no seu celular.

Ativação: Se cadastra e descobre o aplicativo pela primeira vez.

Retenção: Este volta uma segunda vez.

Renda: O usuário começa a pagar para ter acesso a mais funcionalidades.

Recomendação: Contente com o serviço, manda convites para seus amigos via o sistema do aplicativo.

4.4.8. Vantagem comparativa

A maior vantagem comparativa da empresa MatchUp é o conjunto de contatos dos fundadores, nomeadamente nas duas empresas de tecnologias *Google* e *Twitter*.

O *lean canvas* completo é apresentado na figura 98.

4.5. Primeiras confrontações

4.5.1. Objetiva dessa etapa

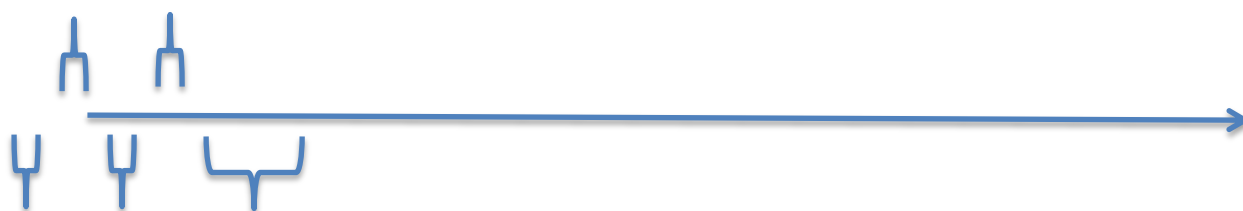
Trata-se da última etapa do primeiro macro passo da presente metodologia. O objetivo é confirmar que o *lean canvas* selecionado na etapa anterior vale a pena ser criado, em termos de tempo e dinheiro. Com efeito, se o *lean canvas* passar nessa última etapa, a equipe começa a trabalhar na produção de um produto viável mínimo, passo que necessita mais tempo do que formular hipóteses e colocá-las em um *lean canvas*.

Três tipos de confrontações acontecem nessa etapa. A primeira é uma confrontação com clientes potenciais e tem como objetivo validar as hipóteses sobre o segmento de cliente, os problemas e a proposição única de valor do *lean canvas*. A segunda envolve também clientes, mas foca nas soluções, na fonte de receita e nos canais de distribuição. A terceira confrontação acontece com especialistas e deve permitir validar a coerência global do *lean canvas* da etapa anterior.

4.5.2. Composição da equipe e duração da etapa

Duas pessoas, cujo o autor, participaram desta etapa, que demorou uma semana.

Geração de novas ideias // *Lean canvas*



Ideia inicial // Ana. Estr. // 1ª confrontações

Figura 21: *Timeline* do projeto, Fonte: Elaborado pelo autor

4.5.1. Confrontação com especialistas

Para essa confrontação, dois elementos são necessários. O primeiro já foi realizado na última micro-etapa, é o *lean canvas* completo. O outro elemento é um protótipo do produto. No método *lean*, este não é necessariamente uma versão funcional do produto. No caso da empresa MatchUp, o protótipo é um conjunto de desenhos das telas do aplicativo. Este protótipo é um suporte que permite receber um feed back dos especialistas.

No caso do MatchUp, o especialista é um responsável do design na empresa Twitter. Ele confirmou a relevância do *lean canvas* e trouxe a sua esperteza para o projeto MatchUp essencialmente na aérea da “experiência usuário” - UX – e “interface usuário” – UI. Trata-se de dois elementos chaves do jeito de usar o aplicativo para que a experiência do usuário seja a melhor possível.

4.5.2. Confrontação com clientes alvos no diz respeito aos problemas

4.5.2.1. Riscos medidos

Três riscos são avaliados durante as entrevistas sobre os problemas.

O ‘risco produto’: qual é o problema que a empresa tenta resolver? Para responder essa questão, o cliente tem, nessa entrevista, que classificar 3 problemas identificados como prioritários. Tem também o ‘risco mercado’: quem são os concorrentes? Durante a entrevista, o cliente dá sua solução para resolver estes problemas. Finalmente tem o ‘risco cliente’: quem é afetado por este problema? O entrevistador deve achar o perfil do cliente.

4.5.2.2. Hipóteses testadas

Segundo Maurya (2012), as confrontações com clientes devem permitir confirmar ou negar as hipóteses colocadas no *lean canvas*. É impossível testar todas essas hipóteses numa mesma confrontação, especificamente sobre os problemas. Sendo assim, as hipóteses são classificadas segundo a probabilidade que estas estejam erradas e o grau de risco pelo projeto, se for o caso.

Maurya recomenda selecionar as três hipóteses que apresentam o risco maior para o projeto. No entanto, neste trabalho, é a criticidade que permite classificá-las.

As notações são tiradas do *lean canvas* completo da figura 20.

		Gravidade				
		1	2	3	4	5
Probabilidade	5					
	4		P4			P2
	3			P3		
	2					
	1	P5			P1	

Tabela 3: Matriz de criticidade do projeto MatchUp

Fonte: elaborado pelo autor

A hipótese sobre o problema um tem uma probabilidade muito baixa de estar errada, já que foi a hipótese principal do aplicativo Tinder, aplicativo que encontrou um sucesso importante. Além disso, tem uma gravidade importante, pois é o componente principal da experiência do usuário.

A hipótese sobre o problema dois aparece como a mais crítica. Com efeito, ela é baseada sobre conversações anteriores dos fundadores com usuários de aplicativos de paquera, e não sobre estudos rigorosos. Além disso, é totalmente ligado com a proposição única de valor da empresa, ou seja é o problema maior que o aplicativo pode resolver.

A hipótese sobre o problema 3 apoia-se também das conversas ou intuições dos fundadores e também dos comentários dos usuários disponíveis nas lojas virtuais Google Play ou AppStore. Mas como constitui só funcionalidades a mais a implementar e não o elemento chave do aplicativo, a gravidade dessa hipótese estar errada é menos importante do que para o problema 1 ou 2.

A hipótese sobre o problema 4 tem uma probabilidade igual ao problema 2, pela mesma razão, no entanto, a gravidade é baixa.

A classificação é:

Hipótese	Criticidade = Probabilidade X Gravidade
Problema 2	20
Problema 3	9
Problema 4	8
Problema 1	4
Problema 5	1

Tabela 4: Classificação das hipóteses segundo sua criticidade

Fonte: elaborado pelo autor

Os problemas selecionados para ser testados são: 2, 3 e 4.

Segundo Maurya (2012), a fim de testar essas hipóteses, elas devem ser enunciadas sob uma forma refutável. Uma hipótese refutável é a soma de dois elementos:

[ação específica reproduzível] + [resultado mensurável esperado]

Problema 2:

As confrontações da primeira micro etapa revelarão (ação específica reproduzível) que a ausência de possibilidade de ver pessoas de outros lugares nos outros aplicativos é uma causa de frustração para os usuários (resultado mensurável esperado)

Problema 3:

As confrontações da primeira micro etapa revelarão que os usuários de aplicativos de paquera esperam deles uma filtragem dos usuários, nomeadamente no que diz respeito a educação e ao emprego.

Problema 4:

As confrontações da primeira micro etapa revelarão que os usuários desejam uma ajuda para iniciar uma conversa.

4.5.2.3. As entrevistas

No total, 20 pessoas foram entrevistadas. Todos foram submetidos ao mesmo script descrito na figura 22.

Acolher (2 min)

O entrevistador recebe o testador e explica o que vai acontecer durante a entrevista.

Coletar os dados pessoas (2 min)

As primeiras questões perguntadas permitem testar o segmento de cliente. As informações recolhidas são: a idade, frequência de deslocamento, profissão e/ou estudo e status de relacionamento.

Contar uma historia (2 min)

O entrevistador cria o contexto para ilustrar os 3 problemas selecionados.

Estabelecer uma classificação dos problemas (3 min)

O entrevistador pede para o cliente alvo classificar os 3 problemas:

- O que você acha da impossibilidade de ver perfis de usuários de outras regiões do que a sua própria? (P2)
- O que você acha da pertinência dos perfis dos aplicativos atuais? (P3)
- Como solteiro, você acha difícil começar uma conversa com uma pessoa desconhecida? (P4)

Descobrir o ponto de vista do cliente (7 min)

O entrevistador deixa o entrevistado falar sobre os problemas que enfrenta, quer sejam os P2, P3 e P4, quer sejam outros. Este fala também sobre as soluções atuais que usa.

Terminar a entrevista (2 min)

Dar um *pitch* do produto para interessar o cliente.

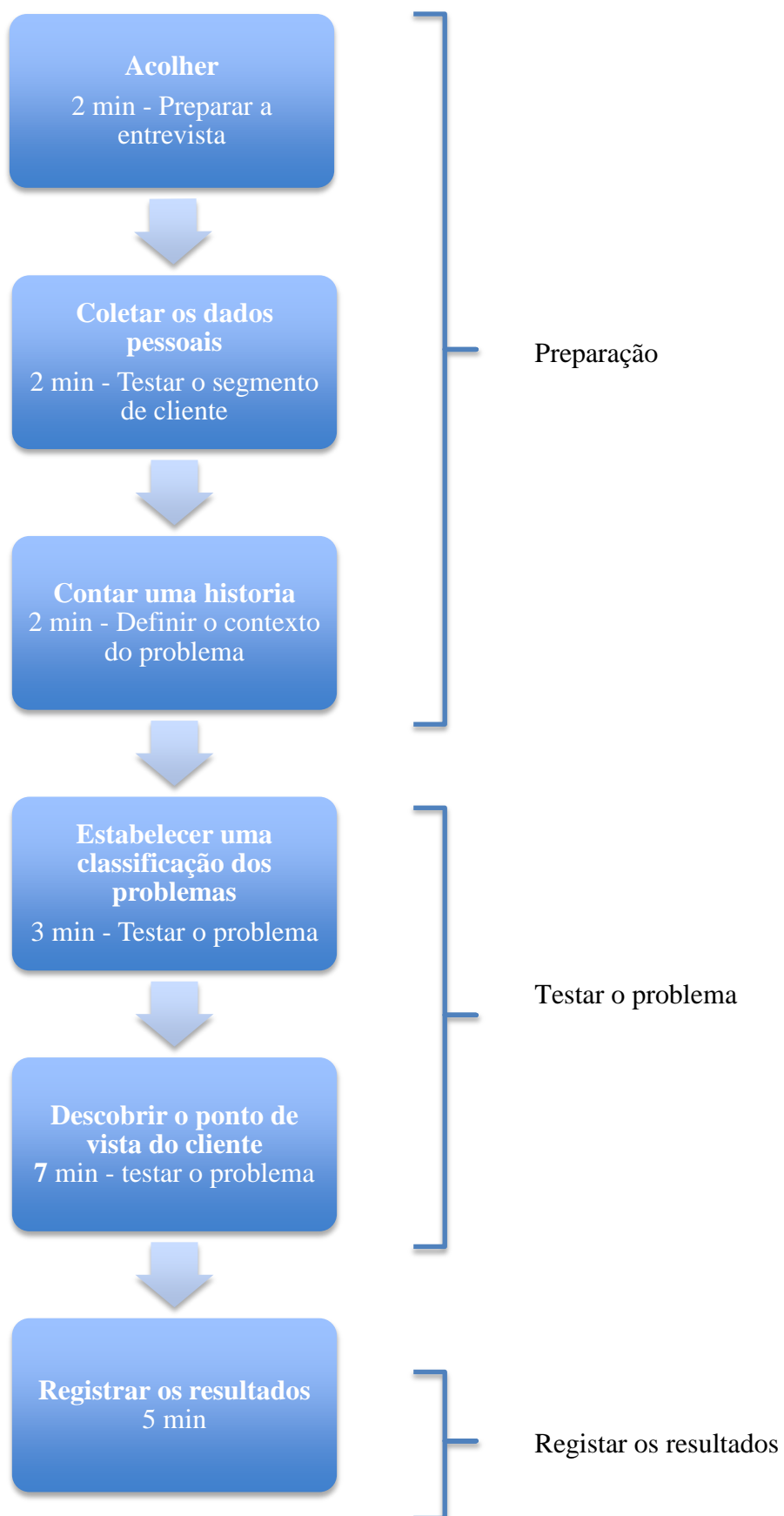


Figura 22: Script da entrevista sobre os problemas, Fonte: adaptado de Maurya, 2012

4.5.2.4. Aprendizagem sobre os problemas enfrentados pelos usuários

O problema maior para as 20 pessoas interrogadas é o problema 3, ou seja, a vontade deles de ver perfis mais adequados. Em média, foi classificado em posição 1.65 (média das 20 classificações, a posição 1 era dada para o problema maior e 3 para o menor). Vem depois o problema 2, a vontade de poder escolher o lugar onde encontrar pessoas, com uma classificação de 1.80. Finalmente, o problema 4, a dificuldade para iniciar uma conversa com um desconhecido, é classificado em último lugar dos maiores problemas com uma posição média de 2.65.

Sobre as soluções atuais dos entrevistados:

- 90 % usam o Tinder como principal aplicativo de paquera.
- 10 % usam o MeetMe como principal aplicativo de paquera e 5 % como segundo.
- 10% usam Lovoo e Badoo como segundo aplicativo de paquera.
- 5% usam Happn como segundo aplicativo de paquera.

O ponto de vista dos entrevistados sobre os aplicativos atuais fica na tabela 6.

Nessa tabela, para cada vantagem ou desvantagem, o número de pessoas que a proporcionou é colocado entre parênteses.

É notável que a metade das pessoas pesquisadas nunca tentaram um outro aplicativo de paquera além do Tinder. A razão avançada para justificar isso é que o Tinder é tanto percebido como um aplicativo de paquera, quanto como um jogo.

A relação entre as vantagens do Tinder e a simplicidade, a experiência do usuário chega como primeira. A base importante de usuários aparece como um apoio importante ao sucesso do aplicativo, já também que é um ponto fraco dos concorrentes dele. Para este grupo de usuários, a utilização gratuita do aplicativo aparece em ultimo lugar das vantagens e a utilização cobrada de certos outros aplicativos foi destacada por apenas um usuário.

	Tinder	Outros aplicativos
Vantagens	<ul style="list-style-type: none"> - Simplicidade, UX (17) - Grande número de usuários (13) - Lúdico (12) - Utilização gratuita (6) 	<ul style="list-style-type: none"> - Uma adequação mais forte com os desejos de encontro (10)
Desvantagens	<ul style="list-style-type: none"> - Simples demais, falta de funcionalidades (15) - Perfis que não correspondem aos desejos (14) - Não tem a possibilidade de achar pessoas de outros lugares do mundo (12) - Falta de reatividade dos outros usuários (7) - Bugs (1) 	<ul style="list-style-type: none"> - Cheio de bugs (8) - Pouco usuários (8) - Aplicativos cobrados (1)

Tabela 5: Vantagens e Desvantagens dos aplicativos atuais de dating

Fonte: Elaborado pelo autor

A falta de funcionalidade aparece como a principal desvantagem do Tinder, nomeadamente, não tem como filtrar os perfis ou como mudar de localização. É possível que essas duas respostas foram dadas tantas vezes por causa dos 3 problemas que foram apresentados uns minutos antes na entrevista, mas isso confirma também que são problemas de verdade para os usuários. A falta de reatividade dos outros usuários, ou seja, a ausência de respostas as mensagens ou a má vontade de começar uma conversa era um problema para 7 das 20 pessoas interrogadas.

Finalmente, os usuários não tem paciência para os aplicativos que apresentem bugs, o principal ponto fraco dos concorrentes do Tinder.

4.5.3. Confrontação com clientes alvo no diz respeito às soluções

4.5.3.1. Riscos medidos

Três riscos são também avaliados durante as entrevistas para as soluções. Tem o ‘risco produto’: quais são as soluções para os 3 problemas destacados? Tem o ‘risco mercado’: qual é o modelo de tarifa? Finalmente tem o ‘risco cliente’: quem é o *early adopter*?

4.5.3.2. As Entrevistas

As 20 pessoas já entrevistadas para a parte dos problemas foram entrevistadas de novo e submetidos ao mesmo script descrito na figura 23.

A diferença deste script com o das soluções fica nas partes:

Demonstrar (10 min)

Para cada problema, o entrevistador ilustra como o aplicativo MatchUp quer o resolver. Para isso, ele usa o mesmo protótipo do que o que serve para a demonstração com especialistas.

O entrevistador busca saber quais são as funcionalidades que fazem mais sentidos, menos sentidos ou são indispensáveis e quais são as que são ausentes.

Testar o preço (5 min)

O entrevistador propõe o preço ao entrevistado. Ele anota se o entrevistado rejeita, aceita sem resistência ou com resistência.

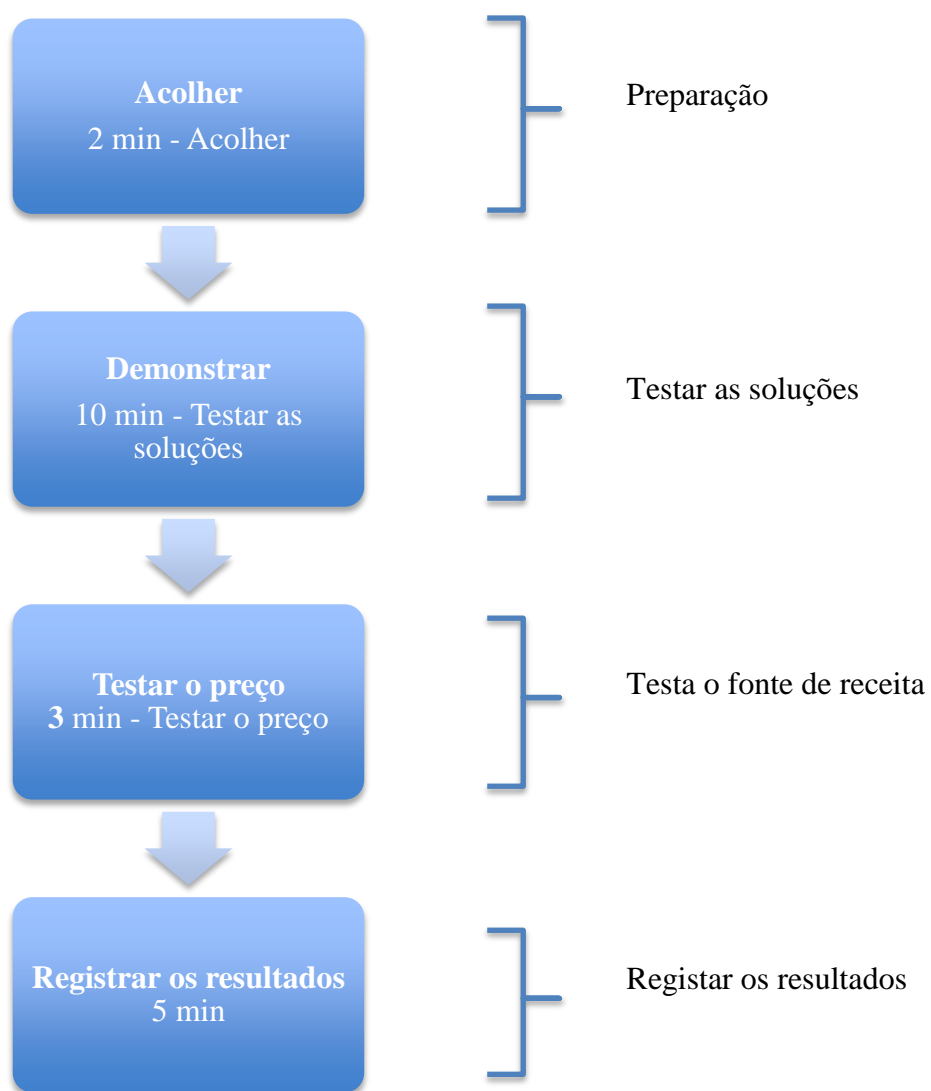


Figura 23: Script da entrevista no diz respeito às soluções

Fonte: Adaptado de Maurya, 2012

4.5.3.3. Aprendizagens sobre as soluções

Para cada solução os 3 problemas selecionados e os comentários dos entrevistados são classificados em 4 categorias no gráfico 2.

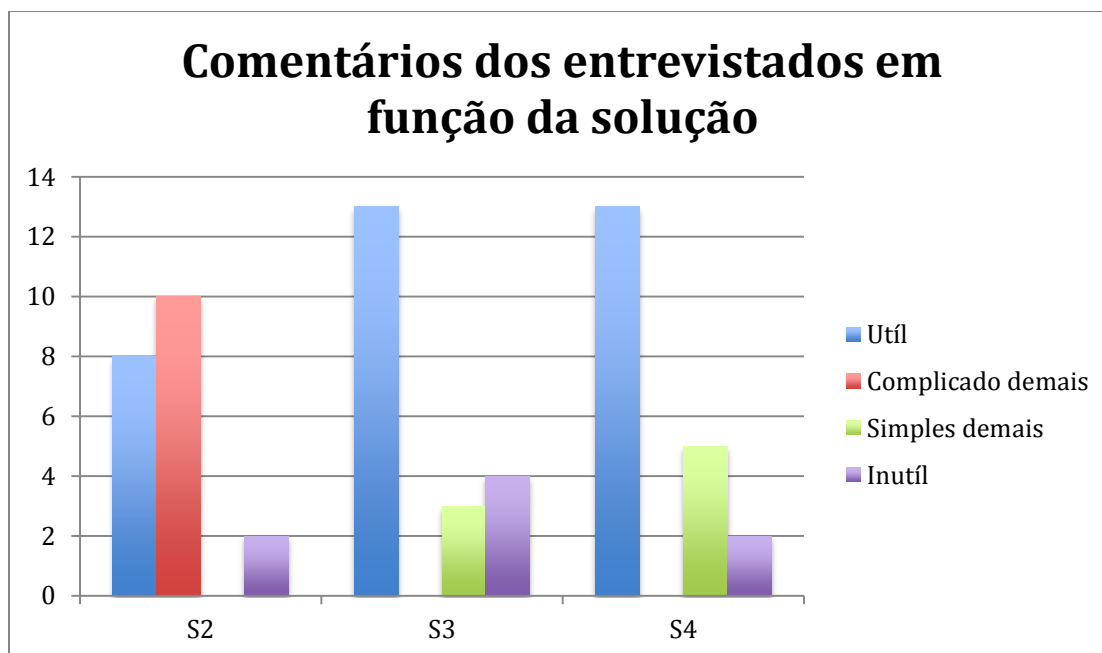


Gráfico 2: Classificação das respostas dos entrevistados segundo a solução

Fonte: Elaborado pelo autor

A solução 2, ou seja, a possibilidade do usuário indicar onde e quando ele vai se deslocar, para assim o aplicativo propor perfis de pessoas mais adequadas com a localização dele, apareceu como útil para 40% dos entrevistados, complicada demais para 50% e inútil para 10%. Os entrevistados que acharam a solução complicada demais propuseram a possibilidade para o usuário indicar vários lugares onde ele pode ir, seja frequentemente, ou pontualmente sem indicar uma data específica. Além disso, a questão da proteção dos dados apareceu como um receio, pois o aplicativo teria acesso ao conjunto dos deslocamentos dos usuários, o que constitui obviamente dados pessoais.

A solução 3, a possibilidade para o usuário indicar o estudo e/ou o trabalho dele para o propor perfis mais adequados, apareceu como útil para 65% dos entrevistados, simples demais para 15% e inútil para 20%. As pessoas que acharam a solução simples demais propuseram que a triagem seja feita segundo o grau de sucesso do usuário, já que estes aplicativos focam mais na aparência física do que no nível de educação.

A solução 4, a integração com Facebook simplificando a descoberta dos interesses em comum entre usuários e a possibilidade de colocar um fato insólito para iniciar um conversa com mais facilidade, apareceu como útil para 65% dos usuários,

simples demais para 25% e inútil para 10%. A integração dos interesses em comum via Facebook foi um fracasso para Tinder segundo os entrevistados. Além disso, eles acham a funcionalidade do fato insólito interessante, mas pensam que é possível ainda o melhorar para o tornar mais divertido.

A Repartição dos usuários em função da disposição de pagar o serviço é a seguinte:



Gráfico 3: Repartição dos usuários em função da disposição de pagar o serviço

Fonte: Elaborado pelo autor

A grande maioria dos entrevistados, 80 % (16 pessoas), não pagariam o serviço, 10 % não deram uma resposta clara e 10% pagaria o serviço se o produto estivesse disponível no momento da entrevista.

As entrevistas permitiram confirmar que o *lean canvas* selecionado era relevante. Com efeito, o conjunto de entrevistados, selecionados por pertencer ao segmento de cliente alvo, reagiu positivamente às entrevistas e mostrou um interesse à ideia e ao protótipo. Além disso, as entrevistas em relação aos problemas mostraram um certo número de fontes de insatisfação dos usuários de aplicativos de paquera, nomeadamente no diz respeito aos problemas selecionados P2, P3 e P4. As entrevistas

em relação as soluções mostraram, mesmo se umas alterações devem ser introduzidas, que são relevantes e úteis para a maioria dos usuários potenciais entrevistados.

Essa etapa constituiu o último passo da primeira micro etapa, a adequação problema/solução. Visto a reação dos entrevistados, a equipe encarregada pelo projeto MatchUp resolveu continuar com este *lean canvas* e construir um PVM cujo as características, apresentados ao próxima passo, serão alinhados com ele e as aprendizagens adquirida nessa etapa.

4.6. Elaboração do primeiro PVM

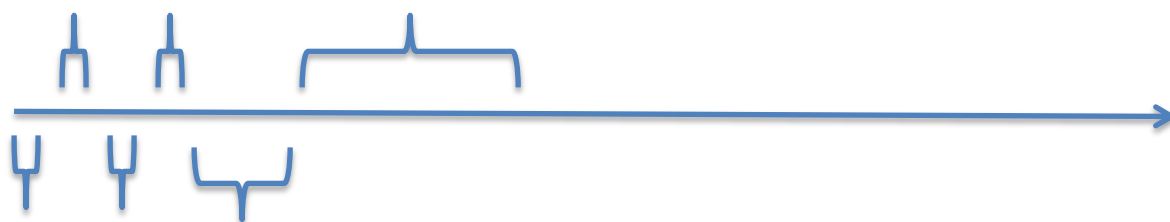
4.6.1. Objetivo dessa etapa

Nessa etapa, um primeiro produto será proposto aos early adopters. Ele é criado segundo às especificações do *lean canvas* e a partir dos resultados da parte anterior. O PVM deve apresentar as funcionalidades mínimas para responder a PUV.

4.6.2. Composição da equipe e duração da etapa

Duas pessoa, cujo o autor, participaram dessa etapa, que teve duração de um mês.

Geração de novas ideias // Lean canvas // Elaboração do PVM



Ideia inicial // Ana. Estr. // 1ª confrontações

Figura 24: *Timeline* do projeto, Fonte: Elaborado pelo autor

4.6.3. As características do PVM

O passo anterior ensinou à equipe do projeto MatchUp que a experiência do usuário do aplicativo Tinder contribuiu significativamente ao seu sucesso, nomeadamente a impressão dada ao usuário que é tanto uma ferramenta de encontro quanto um jogo. Portanto, o sistema de seleção dos outros usuários será semelhante ao do Tinder e fará parte do PVM.

Como o PVM deve responder a PUV, a solução 2 será implementada. Mas, como os entrevistados o sugeriam, trata-se somente da possibilidade para o usuário escolher vários lugares sem escolher uma data específica.

As soluções 3, 4 e 5 serão implementadas nos ciclos CMA posteriores, pois não tem a ver com a PUV.

Uma funcionalidade de mensagens minimalista fará também parte do PVM, pois é necessário para este tipo de aplicativo – os usuários devem poder comunicar caso apareça um interesse comum.

Finalmente, haverá uma página web para apresentar o aplicativo e permitir que o download seja realizado.

A forma de gerenciar a implementação das características no produto e a criação do *software* será vista no próximo passo, “iterações segundo os ciclos CMA”, justamente focado na gestão do projeto.

4.6.4. Confrontações com os early-adopter

A partir das 20 pessoas entrevistadas na micro etapa 1, 11 foram selecionados como early adopters.

As entrevistas mostraram que, o que pode parecer óbvio, tem uma correlação positiva entre o interesse para resolver o problema 2 e a frequência de deslocamento, como o mostra no gráfico 4:

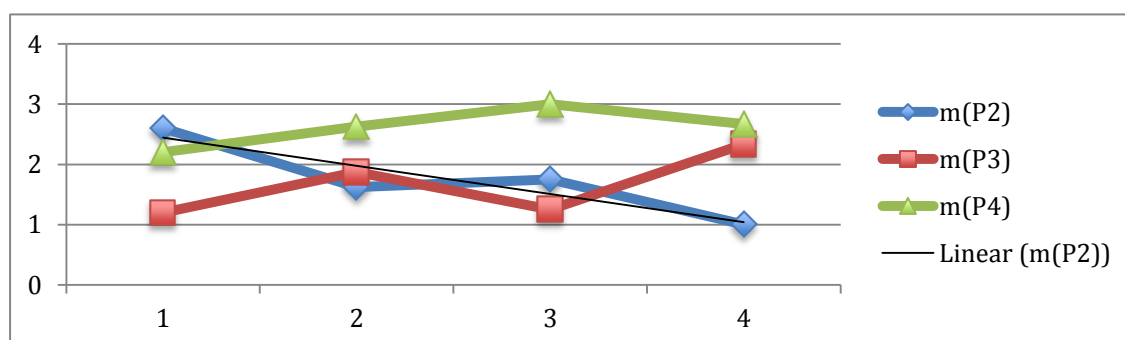


Gráfico 4: Classificação média do Problema i em função da frequência de deslocamento

Fonte: Elaborado pelo autor

Portanto, o critério de seleção dos early adopter é $C = C1$ e/ou $C2$, com:

$C1$: *frequência de deslocamento – classificação do problema 2 > 0*

$C2$: *usuário compraria o produto*

Além destes 11 primeiros early adopters, nove novos entrevistados foram selecionados segundo critérios semelhantes para manter um número de 20 pessoas a entrevistar para cada ciclo CMA.

Segundo o modelo *lean startup*, o PVM é lançado no mercado (Ries, 2011). A metodologia proposta neste trabalho propõe lançar o produto depois de umas iterações segundo os ciclos CMA. No caso do projeto MatchUp, foram 6 iterações. Essa decisão é justificada pelo fato que o projeto é baseado sobre o efeito de rede. Antes de atingir uma massa crítica de usuários, o aplicativo aparece como quase inútil. Atingir uma massa crítica rapidamente necessita um esforço de marketing importante. Aparece como pouco relevante concentrar este esforço para um produto com tão pouco valor agregado. Além disso, uma das forças deste projeto é uma base importante de entrevistados, o que permite aprender durante os ciclos CMA sem ter que lançar o produto no mercado.

O script da entrevista é apresentado na figura 25.

Acolher (1 min)

O entrevistador recebe o entrevistado e explica o que vai acontecer durante a entrevista.

Testar o site (2 min)

O site é considerado como o canal de entrada para o aplicativo, sua vitrina. Portanto, ele permite testar a métrica “aquisição”. O entrevistador mostra o site do aplicativo ao entrevistado, e cada um deve responder se baixaria o aplicativo ao entrar neste site. Uma vez que as 20 pessoas foram entrevistados, a métrica “aquisição” é calculada, é a porcentagem das pessoas testadas que baixariam o aplicativo.

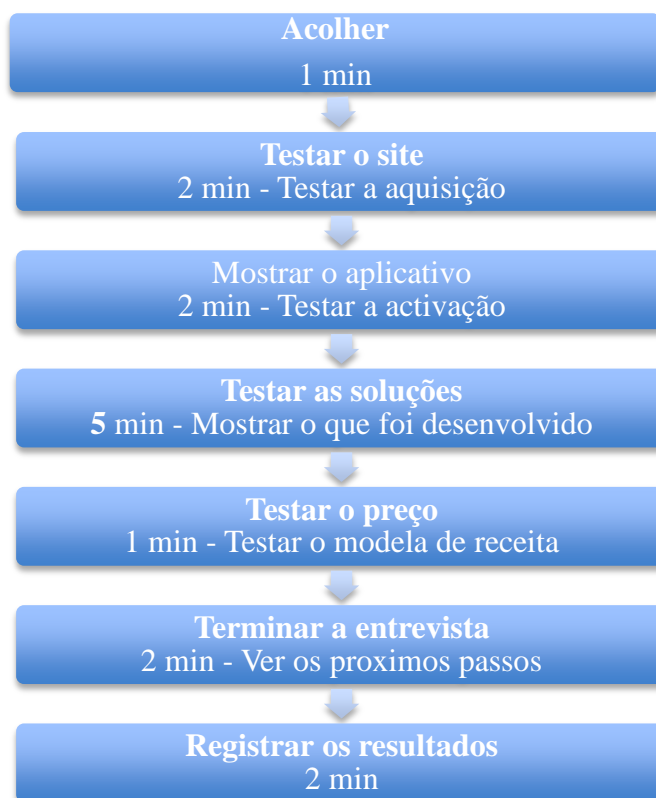


Figura 25: Script da entrevista no diz respeito ao PVM

Fonte: Adaptado de Maurya, 2011

Mostrar o aplicativo (2 min)

O testador deve usar o aplicativo sem a ajuda do entrevistador. Isso permite testar a UX e a UI. Após a exploração do aplicativo, o entrevistado deve dizer se ele usaria este aplicativo se fosse instalado no celular dele. Portanto, trata-se de testar a métrica “ativação”. Uma vez que a entrevista com as 20 pessoas for finalizada, a métrica “ativação” é calculada, é a porcentagem de entrevistados que usariam efetivamente o aplicativo.

Testar as soluções (5 min)

O entrevistador mostra precisamente o que foi desenvolvido no ciclo. Neste primeiro ciclo, o entrevistador foca na solução 2. Para cada funcionalidade mostrada, o usuário dá uma nota que caracteriza a adequação problema/solução, ou seja, se a solução proposta responde realmente ao problema considerado.

Nota	Adequação problema-solução
1	Péssima
2	Ruim
3	Media
4	Boa
5	Ótima

Tabela 6: Nota dada à adequação problema-solução

Fonte: Autor deste trabalho

Uma vez que os 20 testadores foram entrevistados, a nota média é calculada. A quantidade R_i , elaborada pelo autor, é calculada para cada funcionalidade i :

$$R_i = \frac{\text{Nota media da solucao } i}{\sqrt{\text{Criticidade do problema } i}}$$

O problema é considerado como resolvido, e a solução é encontrada quando R_i é maior ou igual do que 1. Se R_i fosse menor, deve-se achar uma outra solução no próximo ciclo, com a ajuda das informações recolhidas dos testadores no ciclo atual.

Testar o preço (1 min)

O entrevistador pergunta ao testador se pagaria o preço definido no *lean canvas*. Isso permite testar a métrica “renda”. Uma vez que os 20 testadores foram entrevistados, a métrica “renda” é calculada, é a porcentagem de pessoas entrevistadas que pagariam efetivamente para usar o aplicativo.

Terminar a entrevista (2 min)

O entrevistador pergunta ao entrevistado quais são suas prioridades para melhorar o produto, quais são as funcionalidades que ele queria ver implementadas na próxima entrevista.

O resultado da entrevista no diz respeito ao PVM aparece na tabela 8.

Usuário	Nota S2	Aquisição	Ativação	Renda
1	3	1	0	0
2	3	0	0	0
3	4	1	0	0

4	3	1	1	0
5	2	0	0	0
6	3	1	0	0
7	3	0	0	0
8	4	1	1	0
9	5	1	0	0
10	4	1	0	0
11	5	0	0	0
12	3	0	0	0
13	3	1	0	0
14	3	0	0	0
15	4	1	1	0
16	4	1	0	0
17	5	0	0	0
18	3	1	0	0
19	4	1	0	0
20	4	0	0	0
Resultado	Nota media : 3,6 Criticidade : 20 R = 0.80 < 1	60 %	15 %	0 %

Tabela 7: Resultado da entrevista no diz respeito ao PVM

Fonte: Elaborado pelo autor

Sobre a aquisição

Aqueles testadores que não baixariam o aplicativo denunciaram uma falta de design do site, em comparação com os standards atuais. Do outro lado, a PUV escolhida no *lean canvas* e aparecendo ostensivamente convenceu a grande maioria dos entrevistados.

Sobre a ativação

A UX e a UI são denunciadas como ruins. O aplicativo aparece como contra-intuitivo, a funcionalidade principal de mudar de localização meio escondida. Estes pontos explicam uma ativação tão baixa.

Sobre a renda

Ninguém pagaria para usar este aplicativo para as razões já citadas na análise da ativação.

Sobre a solução 2

Mesmo se a nota média não é ruim, ela não é suficiente comparado com a criticidade importante do problema 2. Os testadores denunciaram nomeadamente uma utilização difícil dessa funcionalidade e o fato dela estar escondida.

4.7. Iterações segundo os ciclos CMA

4.7.1. Objetivo dessa etapa

Na etapa anterior, o processo das entrevistas e a medição das métricas foram apresentados, ou seja a parte Medir e Aprender dos ciclos CMA. Essa última parte do trabalho foca no Construir assim como a articulação entre os diferentes ciclos.

4.7.2. Composição da equipe e duração da etapa

Quatro pessoas, cujo o autor, participaram dessa etapa, que demorou quatro meses.

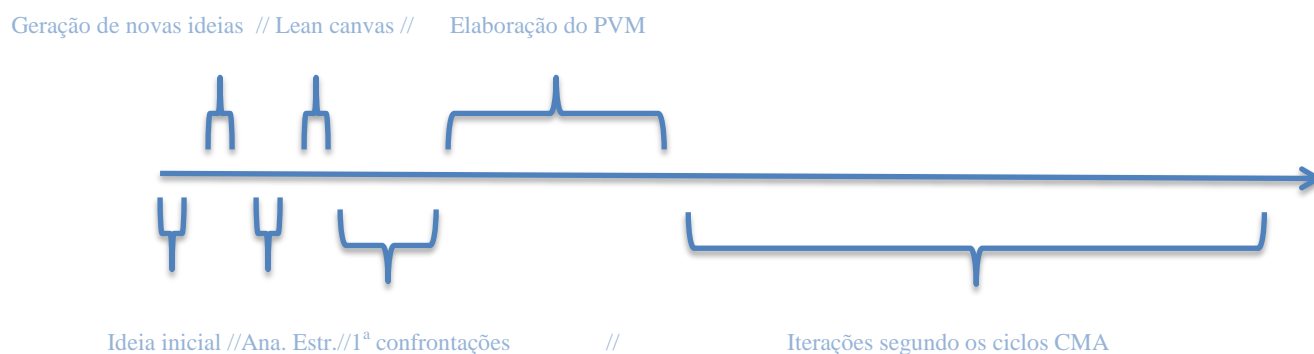


Figura 26: *Timeline* do projeto, Fonte: Elaborado pelo autor

4.7.3. Construir o produto

O desenvolvimento do aplicativo segue as regras do extrem programming, método apresentado na revisão bibliográfica deste trabalho, além da metodologia a seguir, desenvolvida pelo autor do trabalho. O desenrolar de um ciclo começa com a planificação do que será desenvolvido nele. Com esta finalidade, o conjunto de funcionalidades potenciais é classificado por importância.

Mais precisamente, no final das entrevistas do ciclo $i - 1$, os usuários dão uma nota para cada funcionalidade F_j desenvolvida nele. Se o coeficiente R_j é menor do que 1, a funcionalidade F_j será automaticamente melhorada no ciclo i . Essas funcionalidades tem uma prioridade igual à 1 (prioridade maximal). Além dessas notas dadas, os entrevistados, no ciclo $i - 1$, dão uma funcionalidade que eles querem ver no próximo ciclo. Se o testador não tiver uma funcionalidade em mente, o entrevistador propõe uma lista de funcionalidades (basicamente, S3, S4, S5), o testador deve escolher, dentro dessa lista, a funcionalidade que é mais importante para ele. Cada funcionalidade potencial recebe uma nota, que corresponde no total de testadores que propuseram essa funcionalidade para ser desenvolvida. Estas serão classificados por nota, a que obtiver a maior nota recebe uma prioridade igual à 2, a segunda, uma prioridade igual à 3 e assim por diante.

Cada ciclo tem uma duração de 3 semanas. 4 programadores trabalham nessa parte do projeto. Segundo as regras do extrem programming, os desenvolvedores trabalham em par. Portanto, serão duas equipes de programadores. O tempo total de programação disponível por ciclo é:

$$T = \text{Número de equipes} * \text{duração de um ciclo} \\ * \text{tempo de trabalho de um programador}$$

ou seja:

$$T = 2 * 3 * 30 = 180 \text{ horas}$$

Cada funcionalidade F_j , além de uma Prioridade Prd_j , recebe um numero H_j de horas provável para ficar pronta. Em cada inicio de ciclo, a equipe dispõe então de um conjunto $C = (\{F_1, Prd_1, H_1\}, ..., \{F_j, Prd_j, H_j\}, ..., \{F_N, Prd_N, H_N\})$ de funcionalidades. Essas funcionalidades são classificadas por prioridades crescentes,

segundo o esquema de uma pilha. Em cima da pilha, tem a funcionalidade de propriedade maximal ($Prd = 1$), vem em segundo lugar a segunda funcionalidade mais importante ($Prd \in \{1,2\}$, 1 se tem duas ou mais funcionalidades do ciclo $i - 1$ para refazer, 2 se não) e assim por diante. A pilha P tem portanto os elementos de C , mais é ordenada (sendo assim, os indícios mudaram):

$$P = [..., \{F_i, Prd_i, H_i\}, \dots, \{F_j, Prd_j, H_j\}, \dots] \mid \text{se } i < j \Rightarrow Prd_i < Prd_j$$

Neste caso, F_1 tem a prioridade maximal e F_N a prioridade mínima.

As funcionalidades que podem ser desenvolvidas no ciclo i devem obedecer essa equação:

$$\sum_j H_j < T_i = T = 180 \text{ horas}$$

Uma segunda pilha P_i é construída, é composta das funcionalidades que serão desenvolvidas durante o ciclo i .

F_1 é colocada em cima da pilha. Se $T - H_1 > H_2$, então F_2 é colocada em segundo lugar na pilha. Se $T - H_1 - H_2 > H_3$, F_3 é colocada em terceiro lugar. Este procedimento é repetido até o indicio k com:

$$T_i - \sum_{j=1}^{k-2} H_j \geq H_{k-1} \cap T_i - \sum_{j=1}^{k-1} H_j < H_k$$

A funcionalidade F_k não pode ser implementada no ciclo i pois a carga total de horas de trabalho ultrapassaria T , o total de horas disponível de trabalho. No entanto, é possível que tenha uma funcionalidade F_l (com $l > k$) que verifica:

$$T_i - \sum_{j=1}^{k-1} H_j \geq H_l$$

A funcionalidade F_l , mesmo com uma importância menor do que F_k , será implementada no ciclo i . Este procedimento é repetido até que todas funcionalidades forem testadas.

Os programadores dispõem portanto de uma pilha de funcionalidades para implementar no ciclo : $P_i = [F_1, F_2, F_3, \dots, F_{k-1}, F_k, F_l, \dots]$.

Para ilustrar, é possível ter como exemplo o primeiro ciclo CMA depois do desenvolvimento do Produto Mínimo Viável:

Funcionalidade	Prioridade	Carga horaria (em horas)
F1 = S2	1	30
F2 = Serviço de mensagens completo	2	80
F3 = Pagina de perfil	3	60

Tabela 8: Ilustração de uma pilha de funcionalidades (P1), com o primeiro ciclo CMA

Fonte : Elaborado pelo autor

É importante especificar que F_1, F_2, \dots, F_3 são funcionalidades que o usuário do aplicativo entende, ou seja, que fazem sentido para ele, mesmo sem ter qualquer conhecimento da ciência da computação. Como no final das 3 semanas dos ciclos, é este usuário que deve ver um resultado e o avaliar, é primordial que essas funcionalidades sejam assim, compreensíveis. No ponto da vista dele, uma funcionalidade é, por assim dizer, indivisível. Mas do ponto da vista do programador, essas funcionalidades são divisíveis. Por exemplo:

$$F_2 = \kappa_1 + \kappa_2 + \dots + \kappa_9$$

Com $F_1 =$ Sistema de mensagens instantâneas para conversar entre usuários (compreensível por qualquer usuário)

$\kappa_1 =$ Serviço de WebSocket – lado Servidor (não necessariamente compreensível por qualquer usuário)

$\kappa_2 =$ Connection manager – lado Cliente

:

$\kappa_3 =$ Modelo de uma mensagem – lado Cliente

$\kappa_9 =$ Modelo de uma mensagem – lado Servidor

$(\kappa_1, \kappa_2, \dots, \kappa_9)$ representa assim o conjunto de funcionalidades - ou “classes” - indivisíveis do ponto de vista do programador.

De fato, existem vários tipos de linguagem de programação. A linguagem usada para programar os aplicativos moveis é do tipo “linguagem orientada objeto”. Basicamente, isso significa que um programa completo, o aplicativo no caso de MatchUp, é um conjunto de funcionalidades elementares, chamadas “classes” que podem ser desenvolvidas independentemente uma das outras. Isso significa que qualquer uma das classes κ_1 à κ_9 pode ser desenvolvida em primeiro lugar ou sem

que as outras estejam prontas. Mais geralmente, se o aplicativo completo pode ser descrito:

$$Aplicativo = \sum_{i=1}^{N=18} F_i = \sum_{j=1}^{M=135} \kappa_j$$

Então a ordem de desenvolvimento das κ_j não faz diferença para o tempo de desenvolvimento do aplicativo ou a facilidade para o desenvolver. Isso significa que não tem tarefas bloqueantes.

Segundo tudo o que foi avançado, aparece claramente que as funcionalidades não são desenvolvidas como blocos individuais, mas divididas segundo elementos menores, as classes. Portanto, a tabela 10 pode assumir essa forma:

Funcionalidade	Classe	Prioridade	Duração (horas)
1	1	1	8
	:	1	:
	7	1	2
2	1	2	30
	:	2	:
	9	2	2
3	1	3	40
	:	3	:
	4	3	1

Tabela 9: Ilustração de uma pilha de funcionalidades (P1), com o primeiro ciclo CMA, incluindo as classes

Fonte: Elaborado pelo autor

O trabalho dos programadores é organizado de uma forma simples. Quando um par de desenvolvedores acaba uma classe, ele a testa e cuida de uma nova classe, aquela que fica em cima da pilha P_i .

Para que este método de gestão de projeto funcione, as diferentes classes devem ser compatíveis. Para entender o que isso significa, é interessante imaginar as classes como caixas pretas com *inputs* e *outputs*.

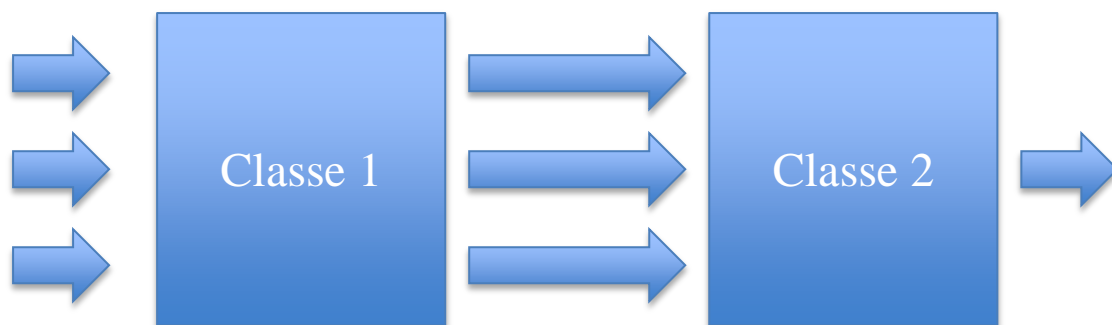


Figura 27: Ilustração de duas classes, como caixas pretas, com 3 *inputs* e 3 *outputs* para a primeira e 3 *inputs* e 1 *output* para segunda

Fonte: Autor deste trabalho

As classes se encaixam umas nas outras, os *outputs* de uma classe constituem os *inputs* da outra. Portanto, é importante que estes *inputs* e *outputs* estejam definidos ao início de cada ciclo. De fato, eles são definidos no primeiro dia de cada ciclo, ao mesmo tempo que as funcionalidades de cada classe. A forma de transformar os *inputs* em *outputs*, dentro de uma classe, não é importante e são os pares de desenvolvedores que a definem no momento que eles programam as classes.

A tabela 11 ilustra a figura 2, com classes da funcionalidade 2 do ciclo P1:

Classe 1		Classe 2	
Modelo de uma mensagem		Modelo de uma célula de mensagem	
Essa classe fica responsável por transformar dados que vem do servidor (uma mensagem) num formato compreensível por um usuário.		Essa classe fica responsável por transformar dados de uma mensagem em uma representação gráfica dela na tela do usuário.	
Input (3)	Output (3)	Input (3)	Output (1)
Data da mensagem no formato UNIX (número de segundos entre o 1/1/1970 e a data da criação da mensagem)	Data da mensagem no formato DD/MM/AAAA, HH/MM		
Sobrenome do expedidor da mensagem	Sobrenome do expedidor		
Mensagem	Mensagem		

Tabela 10: Ilustração de duas classes, para a funcionalidade 2 do ciclo 1

Fonte: Elaborado pelo autor

Finalmente, uma vez que todas as classes, e portanto as funcionalidades, foram desenvolvidas a parte “Construir” do ciclo CMA acaba. Começam depois as componentes Medir e Aprender já apresentadas no último passo dessa metodologia.

4.7.4. Definir quando acaba o projeto

No final de cada ciclo, as métricas Aquisição, Ativação e Renda são medidas. A equipe definiu o resultado mínimo a atingir para considerar que o desenvolvimento do produto é suficiente para o lançar no mercado.

Aquisição	Ativação	Renda
90 %	70 %	5 %

Tabela 11: Medidas mínimas à atingir para parar o desenvolvimento do produto

Fonte: Elaborado pelo autor

Os 90% da aquisição são justificados pelo fato que os entrevistados são early adopters dos aplicativos de paquera. Portanto, eles não apresentam nenhuma reserva para baixar e testar este tipo de programa. Uma taxa elevada é esperada.

O limiar da ativação é um pouco mais baixo e foi calculado ao usar o coeficiente de viralização:

$$K_V = N_C \times \text{ativação}$$

Onde N_C é o número médio de convites enviado por cada usuário para experimentar o aplicativo. [G. WEINBERG, J. MARES, 2014]. Para conseguir viralizar um produto, K_V deve ser maior do que 1. Com efeito, se N_0 é o número inicial de usuários, se cada usuário manda N_C convites aos amigos/conhecidos dele, então o número n_i de usuários adquiridos ao ciclo i de envio de convites é:

$$n_i = N_0(N_C \times \text{ativação})^i = N_0 \times K_V^i$$

O número total de usuários depois de i ciclos, segundo este modelo é portanto dado:

$$N_i = N_0 \times \sum_{k=0}^i K_V^k$$

Que é uma soma dos termos de uma progressão geométrica, convergente se $K_V < 1$ e divergente se $K_V \geq 1$.

O objetivo para o marketing do MatchUp é atingir um N_C entre 1 e 2 ($N_C = \frac{3}{2}$), a ativação é portanto dado por:

$$\frac{3}{2} \times \text{ativação} \geq 1 \Rightarrow \text{ativação} \gtrsim 67 \%$$

A renda, 5%, corresponde ao que foi definido no *lean canvas*.

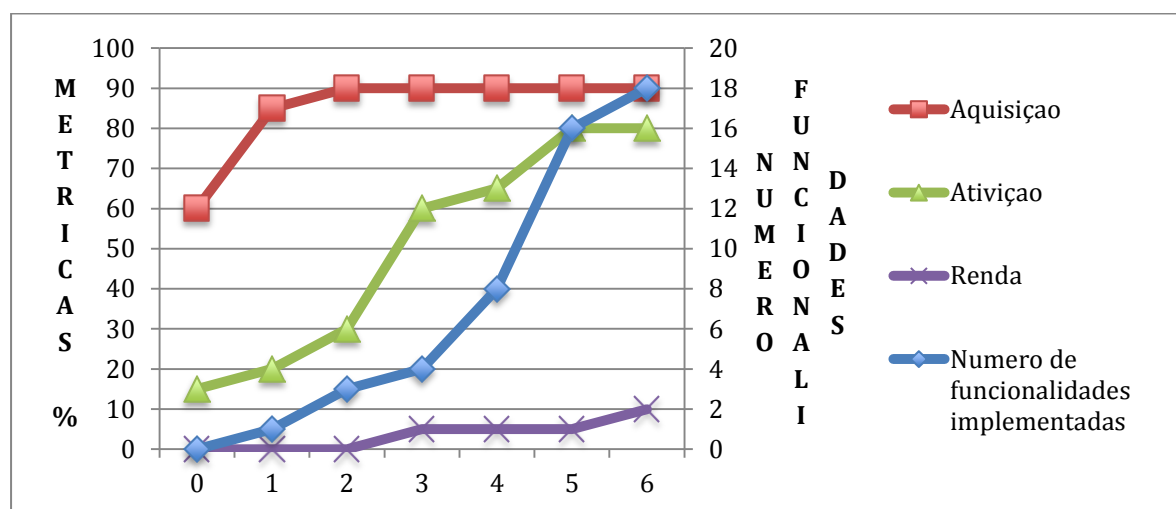


Gráfico 5: Medidas *Pirate Metrics* segundo o número do ciclo

Fonte : Autor deste trabalho

O gráfico 5 apresenta a evolução das métricas segundo os ciclos CMA. O objetivo em termos de aquisição foi atingido em pouco tempo, 2 ciclos, e portanto, não foi mais medido depois. O gap importante entre o ciclo 2 e 3, no diz respeito a ativação, é relacionado com uma real compreensão do problema 2 e portanto de uma solução 2 bem adaptada. Já no ciclo 5, todas as métricas foram atingidas, um último ciclo foi adicionado para regularizar os últimos bugs do ciclo anterior, ou seja, nenhuma nova funcionalidade foi colocada no produto, só uma melhora das existentes.

Seis ciclos, ou sete com o ciclo do PVM, foram necessários para obter um produto pronto para o mercado. No total, demorou um pouco menos do que 6 meses para o desenvolver, o que correspondeu ao objetivo da equipe.

5. Conclusão

Os resultados da última parte mostraram que os objetivos, em termos de métricas, foram atingidos. Tal como referido nessa metodologia, o produto, para ser sujeito ao efeito de rede, não foi lançado na etapa do PVM, mas depois de seis ciclos CMA. Isso apresentou a vantagem de não gastar desnecessariamente com marketing de viralização para um produto com pouco valor agregado. Em contrapartida, o produto foi construído a partir de uma amostra de poucos usuários potenciais, 20 no caso do MatchUp. Os resultados obtidos na última parte deste trabalho são, portanto, valores otimistas dos resultados esperados no mercado real, já que o conjunto de usuários que recomendava as melhoras e as funcionalidades esperadas era também o conjunto de usuários que testava e notava o progresso do aplicativo. Para obter resultados mais realísticos, o autor sugere, aos criadores de *startup* interessados em aplicar esta metodologia, dividir os entrevistadores entre estes dois diferentes grupos – os que recomendam e os que testam - a fim de obter uma base de entrevistados maior.

Para além disso, mesmo que este trabalho apresente uma metodologia para as primeiras fases da vida de uma *startup*, o leitor se dá conta de que a abordagem proposta continua além deste tempo. O objetivo desta metodologia é sempre propor um produto que corresponde aos desejos dos clientes. Portanto, os ciclos CMA não se terminam ao lançamento do produto, mas continuam a serem aplicados ininterruptamente – com clientes bem reais desta vez. Esta é a abordagem que será aplicada para o projeto MatchUp, uma vez a estratégia de lançamento definida e o produto disponível nas lojas virtuais, que correspondem aos próximos passos da *startup*.

6. Referências Bibliográficas

ANDRES, Cynthia. BECK, Kent. Extreme Programming Explained: Embrace Change, Addison-Wesley, 2004

BLANK, Steve. The Four Steps to the Epiphany, K&S Ranck, 2006

BLANK, Steve. Why the Lean startup changes everything, Harvard Business Review, 2013

CHENEVIER, Emmanuel. DEBOIS, François. GROFF, Arnaud. La boîte à outils de la créativité, Dunod, 2014

EASLEY, David. KLEINBERG, Jon. Networks, Crowds, and Markets, Cambridge University Press, 2010

FORTUNE. FT500 2013, Fortune, 2013

GITAHY, Yuri. O que é uma Startup?, Exame, 2010

HIGA, Paulo. Windows Phone ultrapassa iOS no Brasil e fica em segundo lugar entre os sistemas mais vendidos, Technoblog, 2014

HOLSON, Parmy. The Rags-To-Riches Tale Of How Jan Koum Built WhatsApp Into Facebook's New \$19 Billion Baby, Forbes, 2014

HUBBARD, Douglas. How to Measure Anything: Finding the Value of Intangibles in Business, 2010

ISAAC Mike, KAFKA Peter. Twitter Buys Vine, a Video Clip Company That Never Launched, AllThingsD, 2012

IMBD. The Social Network, Imbd, 2011

MAURYA, Ash. Running Lean: Iterate from Plan A to a Plan That Works, O'Reilly Media, 2012

MILLER, Brian. Quick Brainstorming Activities for Busy Managers: 50 Exercises to Spark Your Team's Creativity and Get Results Fast, Amacom, 2012

MC CLURE, Dave. Pirate metrics, 500hats, 2007

OHNO, Taiichi. Toyota Production System: Beyond Large-Scale Production, Productivity Press, 1988

OSTERWALDER, Alexander. PIGNEUR Yves. Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers, John Wiley and Sons, 2010

PORTER, Michael E. Competitive Strategy: Techniques for Analyzing Industries and Competitors, Free Press, 1998

REUTERS. Google closes \$A2b YouTube deal, Reuters, 2006

RIES, Al. TROUT, Jack. Positioning: The Battle for Your Mind, Mc Graw Hill Education, 2001

RIES, Eric. The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses, First Edition, 2011

ROBEHMED, Natalie. What is a Startup, Forbes, 2013

TACOMA, Zach. Criticality Analysis Made Simple, Reliabilityweb.com, 2014

YODER, Steve. How Online Dating became a \$2 billion Industry, The Fiscal Times, 2014