



UNIVERSIDADE FEDERAL DO ESTADO DO RIO DE JANEIRO

CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA

ESCOLA DE INFORMÁTICA APLICADA

Empreendedorismo digital: estudo empírico de padrões organizacionais em novas empresas de software.

Danton Coelho

Orientador

José Ricardo Cereja

RIO DE JANEIRO, RJ – BRASIL

JULHO DE 2015

Empreendedorismo digital: estudo empírico de padrões organizacionais em novas empresas de software.

Danton Coelho

Projeto de Graduação apresentado à Escola de
Informática Aplicada da Universidade Federal do
Estado do Rio de Janeiro (UNIRIO) para obtenção do
título de Bacharel em Sistemas de Informação.

Aprovada por:

José Ricardo Cereja (UNIRIO)

Flávia Santoro (UNIRIO)

RIO DE JANEIRO, RJ – BRASIL.

JULHO DE 2015

Agradecimentos

Agradeço a todos os seres vivos e espirituais que de alguma forma participaram da minha jornada, em especial aqueles que tiveram papel ativo na minha concepção como ser humano, criação como indivíduo social e educação como agente de transformação. Dedico este trabalho ao meu Pai, cuja alma lutou bravamente por muitos anos para permanecer no campo da matéria e hoje desfruta da paz eterna e certamente regozija-se com a conclusão deste ciclo.

RESUMO

Esta monografia tem como objetivo contribuir para o estudo de padrões organizacionais em novas empresas de software caracterizadas como *startups*, através do um estudo empírico baseado em entrevistas junto aos colaboradores e fundadores de startups de software. Concluída a pesquisa se observou 27 diferentes padrões organizacionais. Destes, cinco foram analisados com maior profundidade com base em uma versão adaptada do formulário de Coplien. Os padrões identificados sugerem uma preocupação acentuada dos empreendedores em desenvolver soluções com alto potencial de comercialização em detrimento das práticas formais de engenharia de software. Não foi observada forte adesão aos processos formais, nem mesmo aos métodos ágeis na forma estrita na qual estão definidos. Os empreendedores de startups de software estão focados em desenvolver software, independente de “como”, e conseguir clientes que paguem para utilizá-los.

Palavras-chave: empreendedorismo, empresa de software, padrões organizacionais.

ABSTRACT

This paper aims to contribute to the study of organizational patterns in new software companies characterized as *startups*. An empirical study based on interviews with employees and founders of software startups was conducted to achieve that. The research observed 27 different organizational patterns. Of these, five were analyzed in greater depth based on an adapted version of Coplien form. The identified patterns suggest a marked concern of entrepreneurs in developing solutions with high marketing potential to the detriment of formal practices of software engineering. There was no strong adherence to formal processes, not even the agile methods in the strict manner in which they are defined. The software startups entrepreneurs are focused on developing software, regardless of "how" and get customers to pay to use them.

Keywords: entrepreneurship, software startup, organizational patterns

Índice

1	Introdução.....	11
1.1	Motivação.....	11
1.2	Objetivos	12
1.3	Organização do texto.....	13
2	Revisão Bibliográfica.....	14
2.1	Inovação e empreendedorismo digital.....	14
2.2	Gestão de negócios tradicionais e startups	16
2.3	Engenharia de software e startups	17
2.4	Padrões organizacionais em empresas de software	18
3	Desenvolvimento do estudo	20
3.1	Planejamento e Organização	20
3.1.1	Estrutura organizacional do trabalho.....	21
3.2	Metodologia de pesquisa	22
3.2.1	Detalhamento do processo de pesquisa	23
3.2.1.1	Revisão da literatura.....	23
3.2.1.2	Definição dos critérios de seleção das Startups	24
3.2.1.3	Planejamento das entrevistas.....	25
3.2.1.4	Execução das entrevistas.....	25
3.2.1.5	Identificação dos padrões.....	26
3.2.1.6	Seleção dos padrões	27
3.2.1.7	Detalhamento dos padrões	27
3.2.1.8	Formatação e apresentação dos padrões.....	28
3.3	Padrões candidatos identificados.....	28
3.3.1	Metas	28
3.3.1.1	Foco do produto	28
3.3.1.2	Proposta de valor única	29

3.3.1.3 Evitar customizações.....	29
3.3.2 Cultura	29
3.3.2.1 Liberação controlada de atualizações.....	29
3.3.2.2 Criar a cultura de desenvolvimento.....	30
3.3.3 Financiamento	30
3.3.3.1 Autofinanciamento	30
3.3.3.2 Busca por investidores	30
3.3.3.3 Identificar compradores.....	30
3.3.4 Cliente	31
3.3.4.1 Focar em um nicho específico de clientes.....	31
3.3.4.2 Comunicação com o cliente	31
3.3.5 Organização.....	31
3.3.5.1 Organização horizontal	31
3.3.5.2 Não crescer em pessoal	31
3.3.6 Competência.....	32
3.3.6.1 Formar parcerias.....	32
3.3.6.2 Manter a vanguarda.....	32
3.3.6.3 Compartilhar ideias	32
3.3.7 Equipe.....	32
3.3.7.1 Desenvolvedores polivalentes	32
3.3.7.2 Começar com uma equipe pequena.....	33
3.3.8 Processo.....	33
3.3.8.1 Melhorias de processos no momento certo	33
3.3.8.2 Customize práticas ágeis	33
3.3.8.3 Falhar rápido	34
3.3.9 Projeto	34
3.3.9.1 Desenvolver apenas o que é necessário.....	34

3.3.9.2 Usar arquitetura do produto extensível	34
3.3.10 Testes	34
3.3.10.1 Buscar por erros.....	34
3.3.10.2 Automatizar testes	35
3.3.11 Tecnologia.....	35
3.3.11.1 Uso de tecnologias genéricas não proprietárias.....	35
3.3.11.2 Tecnologias escaláveis	35
3.3.11.3 Tecnologias familiares.....	35
3.4 Considerações finais.....	36
4 Análise dos dados.....	37
4.1.1 Criar a cultura de desenvolvimento.....	37
4.1.2 Autofinanciamento	38
4.1.3 Organização Horizontal.....	40
4.1.4 Proposta de valor única	42
4.1.5 Desenvolver apenas o que é necessário.....	43
5 Conclusão	45
5.1 Considerações finais.....	46
5.2 Limitações do projeto	46
5.3 Trabalhos futuros.....	46
Apêndice 1: Formulário de entrevista.....	50

Índice de Tabelas

Tabela 3-1 Padrão organizacional (definição)	22
Tabela 3-2 Relação de startups convidadas por país	25
Tabela 3-3 Questionários respondidos e entrevistas realizadas	26

Índice de Figuras

Figura 2-1 Temas recorrentes em startups de software conforme Giardino et al (2014)	16
Figura 2-2 Relação dos 14 padrões organizacionais destacados por Dande <i>et al</i> (2014).	
.....	19
Figura 3-1 Fluxograma do processo de pesquisa.....	23

1 Introdução

Empreender é uma missão cujo sucesso vai contra as probabilidades. O índice de mortalidade de startups pode chegar a 95% (Gage, 2012). Estabelecer empresas digitais inovadoras baseadas em software tem sido um caminho cada vez mais atraente aos jovens e somente em 2012 foram abertas cerca de 60 mil novas empresas neste nicho (Lee, 2013). O trajeto do empreendedor é permeado por fórmulas e modelos acadêmicos, receitas miraculosas, literatura efervescente e mais exemplos de fracasso do que sucesso (Silva, 2013). Um elemento comum a todos que se arriscam é a extrema incerteza que governa suas empreitadas (Hirai, 2010).

A incerteza é inerente à inovação que consequentemente caracteriza uma startup tecnológica (Ries, 2011). Entendê-la e reduzi-la tem sido o foco de diversos trabalhos acadêmicos (Silva, 2013; Coleman et al, 2014) e publicações editoriais (Ries, 2011; Blank & Dorf 2012). No entanto ainda há pouco material produzido no Brasil ou disponível em português que aborde a realidade do empreendedor na área de software. E mesmo no mundo ainda é um campo de pesquisa em franca ascensão (Gustafsson & Qvillberg, 2012).

Este trabalho traz um estudo empírico realizado em *startups* de software buscando identificar práticas comuns no dia a dia do negócio em diversas categorias, tais como gestão de pessoas ou qualidade software, entre outras. O objetivo maior é identificar padrões organizacionais em novas empresas de software visando ampliar o conhecimento disponível nesta área.

1.1 Motivação

Iniciar um novo negócio implica em colocar-se a disposição de variáveis incontrolláveis sobre as quais o empreendedor tem pouca gerência, como o desejo do

consumidor, a situação econômica do país, ou as forças da natureza (Ries, 2011). No entanto existe um outro conjunto de variáveis que são determinadas diretamente pelas decisões do empreendedor como o segmento de cliente alvo, o modelo de negócio ou o modelo de gestão de processos (Blank & Dorf, 2012).

Para tomar decisões de negócio o empreendedor pode cercar-se de manuais de administração, metodologias reconhecidas, *frameworks* e afins, além da sua própria intuição e da opinião alheia, como destacado em Blank & Dorf (2012).

No entanto, a disponibilidade de material de referência que aborde especificamente melhores práticas organizacionais com foco em novas empresas de software é escassa no Brasil. Ao observar que em 2014 no Brasil existiam mais de 10 mil empresas neste nicho, entre startups e empresas maduras, que conjuntamente movimentaram US\$ 11,2 bilhões (IDC, 2015), e que não foi encontrado trabalho acadêmico nacional que aborde de forma pragmática as idiossincrasias de uma startup de software, surgiu a motivação para a realização deste projeto de fim de curso.

1.2 Objetivos

O objetivo deste projeto de fim de curso é realizar uma pesquisa empírica abrangente com startups nacionais e internacionais, visando identificar práticas de trabalho nos diversos campos da gestão organizacional e da engenharia de software, assim como foi realizado por Dande et al (2014) em *startups* finlandesas e suíças. A partir dos dados coletados espera-se identificar padrões organizacionais em startups de software.

Espera-se que a partir do conhecimento destes padrões os atuais e futuros empreendedores possam enriquecer seu processo de tomada de decisão, bem como ter uma visão ampla das características comuns neste tipo de negócio. Com isto os empreendedores e as pessoas que possuem desejo de conduzir seu próprio negócio poderão exercer um julgamento mais criterioso e embasado sobre os referidos padrões, potencialmente reduzindo o risco de insucesso de seu negócio.

No campo da pesquisa acadêmica brasileira, espera-se contribuir com as áreas de pesquisa em padrões organizacionais em organizações jovens de base tecnológica.

1.3 Organização do texto

O presente trabalho está estruturado em capítulos e, além desta introdução, será desenvolvido da seguinte forma:

- Capítulo II: apresenta a revisão bibliográfica citando os trabalhos acadêmicos, publicações editoriais e outras fontes de pesquisa relevantes no âmbito desta monografia, organizados por área de interesse.
- Capítulo III: aborda o desenvolvimento do trabalho, desde a metodologia adotada na condução da pesquisa, raciais empregados, organização do estudo até o resultado preliminar da pesquisa trazendo a relação dos padrões organizacionais candidatos ao detalhamento e análise.
- Capítulo IV: relaciona os padrões organizacionais selecionados para análise, estruturados nas dimensões Categoria, Problema, Forças, Consequências, Solução e História real.
- Capítulo V: Conclusões – Reúne as considerações finais, assinala as contribuições da pesquisa e sugere possibilidades de aprofundamento posterior.

2 Revisão Bibliográfica

Neste capítulo é apresentada a revisão bibliográfica relevante a consecução deste trabalho. Buscou-se dividir o capítulo nos conceitos considerados importantes para o entendimento e desenvolvimento do trabalho.

A primeira parte aborda conceitos gerais e fatos relacionados ao fenômeno do empreendedorismo digital, que está intimamente ligado à inovação tecnológica, como já era observado desde Drucker (1974) e Schumpeter (1975) até Ries (2011). Em seguida aborda-se de forma mais generalista princípios de gestão organizacional, fazendo-se um breve contraponto entre as nuances da gestão tradicional e de startups de software como relatado por Ries (2011) e Blank & Dorf (2012).

Contextualizado o negócio em si, passamos para um breve estudo sobre engenharia de software, área de conhecimento fundamental e inerente a qualquer empreendimento com viés de software, seja sua aplicação tácita ou formal.

A revisão bibliográfica é finalizada com a revisão de trabalhos sobre padrões organizacionais nas empresas de software, tema diretamente relacionado aos objetivos deste trabalho.

2.1 Inovação e empreendedorismo digital

Conforme Schumpeter (1975), citado por Barbieri et al., (2004), é importante desambiguar inovação e invenção, pois não são sinônimas, e que a inovação não necessariamente é algo novo. Conforme o autor “(...) *inovação não é sinônimo de invenção. Inovação é possível sem que se possa identificar como uma invenção. E uma invenção não necessariamente induz a uma inovação. A invenção por si só não produz nenhum efeito econômico relevante. Mesmo quando a inovação resulta de uma invenção, que pode ter acontecido autonomamente como uma resposta a uma dada situação de negócio, as ações são do ponto de vista econômico e sociológico duas*

coisas completamente diferentes, mesmo quando por acaso são executadas pela mesma pessoa”.

De acordo com 3ª Edição no Manual de Oslo, editada pela Organização para a Cooperação e Desenvolvimento Econômico (OCDE) cujo objetivo é conduzir uma melhor compreensão do processo de inovação, a própria inovação consiste em melhorar produtos ou serviços, agregar valor ao método organizacional, às práticas do negócio interna e externamente (OECD, 2004).

Drucker (1974) contribui através de seus estudos sobre empreendedorismo inovador que: *“Os executivos terão de aprender a construir e administrar uma organização inovadora. Terão de aprender a construir e administrar um grupo humano capaz de prever o novo, de transformar suas ideias em tecnologia, produtos e processos capazes e desejosos de aceitar o novo”* (Drucker, 1974, p.71).

Ries (2011) e Blank & Dorf (2012) protagonizam o estudo de startups, e ambos associam a inovação como característica fundamental deste tipo de organização. As características das startups estão também relacionadas à metáfora do “oceano azul” tratada por Chan Kim e Renée Mauborgne. Nos oceanos azuis a “competição é irrelevante, pois as regras do jogo ainda não estão definidas” (Kim & Mauborgne, 2005). O desdém em relação à competição parte do pressuposto que o produto ou serviço é inovador, logo está abrindo um novo mercado, com possibilidades até então inéditas para os consumidores.

Blank & Dorf (2012) argumentam que a inovação para ser bem sucedida depende também de um modelo de negócios inovador. Osterwalder & Pigneur (2010), descrevem o modelo de negócio como a forma na qual uma organização cria, entrega e captura valor. Essenciais para o crescimento “sustentável” de um negócio.

Empreender na área de software tem sido uma alternativa bastante atraente aos jovens, como observou Giardino et al (2014). A relativa baixa necessidade de investimento inicial e estrutura empresarial, aliada à facilidade logística intrínseca ao software usando a Internet como canal de distribuição, são alguns dos fatores de atratividade. Giardino et al (2014) sintetizou as temas recorrentes entre startups de software, ilustradas na Figura 2-1.

Recurrent themes in software startups.

Theme	Description
Lack of resources	Economical, human, and physical resources are extremely limited.
Highly reactive	Startups are able to quickly react to changes in the underlying market, technologies, and product (compared to more established companies).
Innovation	Given the highly competitive ecosystem, startups need to focus on and explore highly innovative segments of the market.
Uncertainty	Startups deal with a highly uncertain ecosystem under different perspectives: market, product features, competition, people, and finance.
Rapidly evolving	Successful startups aim to grow and scale rapidly.
Time pressure	The environment often forces startups to release fast and to work under constant pressure (terms sheets, demo days, investors' requests).
Third-party dependency	Due to lack of resources, startups heavily rely on external solutions to build their product: external APIs, open source software, outsourcing, COTS, and so on.
Small team	Startups start with a small number of individuals.
One product	Company activities gravitate around one product/service only.
Low-experienced team	A good part of the development team is formed by people with less than five years of experience and often recently graduated students.
New company	The company has been recently created.
Full organization	Startups are usually founder-centric, and everyone in the company has big responsibilities, with no need for upper management.
Highly risky	The failure rate of startups is extremely high.
Not self-sustained	Especially in the early stage, startups need external funding to sustain their activities (venture capitalist, angel investments, personal funds, and so on).
Little working experience	The basis of an organizational culture isn't present initially.

Figura 2-1 Temas recorrentes em startups de software conforme Giardino et al (2014)

2.2 Gestão de negócios tradicionais e startups

Ries (2011) afirma que os métodos tradicionais de administração não se aplicam as Startups, isto porque estes métodos baseiam-se em previsões que funcionam melhor em ambientes estáticos, em detrimento de ambientes onde as Startups estão inseridas, em que as mudanças são frequentes e os riscos maiores.

Segundo Blank & Dorf (2012) *startups* não são simplesmente versões menores de grandes empresas. Baseado nisso, uma empresa iniciante, ligada a qualquer tipo de negócio, que não esteja focada para a inovação de produtos ou serviços e, portanto dispensada para os riscos de incertezas, não é uma startup. Meira (2013) acrescenta que “*nem todo novo negócio é uma Startup, apesar de toda Startup ser um novo negócio*”.

2.3 Engenharia de software e startups

A partir do mapeamento sistemático de Paternoster et al (2014), verificou-se que Coleman & O'Connor e Kajko-Mattson & Nikitina, com diversos trabalhos na área, representam as influências mais significativas para o estudo dos processos e práticas adotadas por startups no desenvolvimento de software.

Coleman & O'Connor (2007), citado em Paternoster et al (2014), relata que XP (eXtreme Programming) é a metodologia de desenvolvimento de software mais utilizada nas empresas iniciantes, devido ao seu processo de baixo custo e baixas exigências de documentação. Outras práticas ágeis são exploradas, tais como Scrumban (Kuvinka 2011 citado em Paternoster, 2014).

Paternoster (2104) explica que em qualquer dos casos, a natureza flexível das startups não estimula a adesão rigorosa a qualquer metodologia específica, mostrando uma predileção por práticas de desenvolvimento mais oportunistas, como *pair-programming* (Deias, 2002 citado em Paternoster et al, 2014) ou *backlog* (Tingling, 2007 citado em Paternoster et al, 2014).

De fato, processos são adaptados para as características específicas que caracterizam cada contexto de desenvolvimento de acordo com Paternoster citando Coleman (2008) e May (2012). Como exemplo, em Bosch et al (2013) fica explícita esta atitude do “não-processo” em startups: “(...) *advocate for adjusting the Lean startup methodology to accommodate the development of multiple ideas and integrate them when time for their testing and validation is too long. This concurs with the practice of allocating varying effort for formalizing specifications, design, documentation and testing in tailored development methodologies [S. Ambler 92, J. Zettel et al. 2001, E. Carmel 94], emphasizing the importance of minimal process management.*”.

“*Done is better than perfect*” (feito é melhor do que perfeito, em tradução livre) e “*move fast and break things*” (mova-se rápido e quebre coisas, em tradução livre) são frases de efeito praticadas repetidamente em startups como relatado por Giardino et al. (2014), que estudou cerca de 200 práticas de trabalho associadas a engenharia de software em startups. Entre as práticas mais comuns estão o uso de *frameworks* bem disseminados para mudar rapidamente o produto de acordo com as tendências do mercado, uso de prototipação evolutiva e experimentação com componentes existentes,

engajamento de toda a equipe na definição dos objetivos, uso de métricas padronizadas para rapidamente aprender com as demandas e *feedbacks* dos usuários, validação contínua e interativa do produto por meio dos clientes iniciais e entrega contínua de valor, com foco nas funcionalidades principais e clientes pagantes.

2.4 Padrões organizacionais em empresas de software

O conceito de “padrão”, da forma como é abordado no trabalho de Dande et al. (2014), principal referência deste projeto de conclusão de curso, originou-se da Arquitetura. Originalmente, um padrão foi apresentado como um construto dependente de um contexto, que inclui um problema e uma solução reprodutível para tal problema na concepção de edifícios (Alexander et al, 1977).

Desde meados dos anos de 1990, os padrões têm sido extensivamente utilizados em engenharia de software. Coleções de padrões têm sido apresentadas na literatura e na Internet para várias subáreas da engenharia de software, como a análise, design, interfaces de usuário, desenvolvimento web, arquitetura de software, entre outras, como observado por Paternoster et al (2014).

Padrões também foram aplicados na organização e processo de desenvolvimento de software conforme Coplien et al (2005).

Dande et al (2014) realizou um estudo empírico com startups da Finlândia e Suíça a fim de observar a ocorrência de padrões organizacionais especificamente em novas empresas de software. Foram observados um total de 63 padrões organizacionais preliminares, dos quais 14 foram detalhados, conforme pode ser observado na figura. O estudo foi concluído com uma análise de interação entre os padrões observados.

Create the development culture before processes [#54]
Self-funding [#62]
Validate idea [#63]
Keep customer communication simple and natural [#45]
Don't grow in personnel [#57]
Flat Organization [#3]
Unique value proposition [#38]
Start with small and experienced team and expand as needed [#64]
Start with a competence focus and expand as needed [#59]
Time process improvements right [#32]
Develop only what is needed now [#27]
Anything goes in product planning [#9]
Bughunt [#34]
Choose Scalable Technologies [#25]

Figura 2-2 Relação dos 14 padrões organizacionais destacados por Dande *et al* (2014).

3 Desenvolvimento do estudo

Neste capítulo é apresentado o processo de trabalho e metodológico empregado na obtenção dos dados e processamento dos resultados. A partir do planejamento original, onde é abordada a localização conceitual deste trabalho e a forma de organização dos elementos constituintes do estudo. Segue-se então para a apresentação da estrutura organizacional em si, onde se caracteriza o elemento primário que se busca obter com a pesquisa – o padrão organizacional. Apresentado o modelo adotado para formatar o padrão, a metodologia de pesquisa é apresentada em detalhes, incluindo os raciais envolvidos nas definições subjetivas, bem como o fluxo de execução e processamento dos dados.

Finalmente, são apresentados os resultados preliminares obtidos, nomeados como padrões candidatos, os quais são relacionados um a um. Por fim as considerações finais encerram o capítulo fornecendo uma breve análise qualitativa que levam aos padrões organizacionais detalhados no Capítulo 4.

3.1 Planejamento e Organização

Este trabalho foi planejado a partir da pesquisa bibliográfica realizada, em especial de trabalhos publicados no exterior, com destaque para o mapeamento sistemático relacionado ao desenvolvimento de software em startups realizado por Parternoster et al. (2014) e ao estudo empírico sobre padrões em startups por Dande et al. (2014). A partir destes trabalhos foi traçado um plano de execução e definidos os componentes estruturais do presente trabalho.

Buscou-se manter um alinhamento conceitual e prático com os demais estudos realizados, a fim de permitir uma comparação mais fluida dos resultados obtidos em trabalhos futuros. O estudo empírico realizado por Dande et al (2014) tem especial influência neste trabalho por apresentar uma visão holística das startups de software a

partir da perspectiva interna, de seus fundadores e colaboradores, com destaque para aspectos ligados a engenharia de software.

O planejamento original para a execução deste trabalho objetivou um levantamento do universo de startups de software brasileiras, buscando a abrangência total do território nacional. No entanto optou-se por ampliar o universo pesquisado em função da baixa aderência que foi observada na obtenção de formulários respondidos.

3.1.1 Estrutura organizacional do trabalho

Com o objetivo de fornecer uma estrutura uniforme ao resultado do trabalho foi necessário estabelecer um modelo padronizado o qual comportasse as descobertas realizadas apresentando-as de forma clara e objetiva. Para tanto se partiu do modelo encontrado em Dande et al (2014), ali denominado simplesmente “*pattern*” o qual teve origem conceitual nos “*organizational patterns*”.

Neste trabalho adotou-se o termo “*padrão organizacional*” como forma de identificar e definir práticas de trabalho e suas estruturas de relacionamento, desta forma catalogando e organizando a memória organizacional. Assim, a partir do estudo empírico, da observação atenta e do questionamento ativo é possível realizar uma inspeção analítica destas experiências e práticas e de forma estruturada disponibilizar para consumo amplo um conhecimento essencialmente tácito e pulverizado.

Um modelo simples foi utilizado durante a etapa de coleta de informações com base nas entrevistas. Um formato sintético, baseado no formulário de Portland (2014). Neste formato, o padrão primariamente identificado é apresentado como um par problema-solução, que neste trabalho foi definido como “situação-reação”.

A “situação” visa expressar as ocorrências relatadas pelo entrevistado e percebidas pelo pesquisador. A “reação” indica as atitudes, técnicas, ferramentas, e outros, utilizadas ou planejadas pelos entrevistados de forma a atuar diante da “situação” exposta.

Os padrões apresentados neste trabalho têm a preocupação central nos comportamentos dos seres humanos, em vez dos componentes de software. O padrão organizacional emerge a partir da pesquisa com distintos agentes e praticantes dentro de um contexto específico.

Muitos formatos diferentes foram propostos para a apresentação de padrões. Adotou-se aqui uma versão ligeiramente modificada do formato Coplien (1996), incluindo novos itens e desconsiderando os itens “*Related patterns*” e “*References*”, tratados como opcionais no modelo original não foram considerados neste trabalho.

O primeiro item adicionado foi "Exemplo real", onde uma ocorrência real do padrão em alguma startup é apresentada de forma abstrata, conforme capturado durante o processo de entrevista. O segundo item adicionado foi “Categoria” que busca agrupar os padrões, refletindo a preocupação central na qual o padrão está contextualizado.

O item “*Related patterns*” foi excluído pois não é escopo deste trabalho fazer uma avaliação cruzada dos padrões ou estabelecer hierarquia ou fluxo entre os mesmos. O item “*References*” foi excluído pois não é objetivo deste trabalho uma análise bibliográfica individualizada bem como um aprofundamento acadêmico para cada um dos padrões identificados.

O modelo de padrão organizacional definido para este trabalho pode ser visualizado na tabela 3-1. Optou-se por manter a compatibilidade com o modelo de Dante et al (2014) permitindo análises cruzadas e viabilizando futuras trocas de informação com outros pesquisadores.

Tabela 3-1 Padrão organizacional (definição)

Dimensão	Descrição
Nome	Nome descritivo do padrão, expressando a ideia central
Categoria	Categoria do padrão, isto é, macropreocupação ou macroprocesso
Contexto	O contexto onde este problema ocorre e a solução pode ser aplicada (por exemplo, uma fase específica da startup)
Problema	O problema endereçado pela solução
Forças	Os fatos que atuam causando conflitos para gerar o problema
Solução	Descrição da solução adotada
Consequências	Benefícios e passivos da solução
Exemplo real	Ocorrência real ilustrando o acontecimento do padrão na prática

3.2 Metodologia de pesquisa

A pesquisa pode ser entendida como uma busca por padrões organizacionais baseada em entrevistas com os praticantes em startups de software. O principal resultado gerado é a lista com os padrões organizacionais candidatos, expressos de

forma sucinta como uma relação problema-solução (situação-reação neste trabalho), assim como em Dande et al (2014).

O conjunto destes padrões candidatos é então submetido a uma análise mais apurada com base no aprendizado consolidado obtido durante o processo de execução das entrevistas ponderada pela quantidade de ocorrências, experiência dos agentes envolvidos no campo de startups de software e relação com engenharia de software. Esta abordagem é similar ao método utilizado na descoberta de serviços em projetos de arquitetura de software orientados a serviços (Ganapathy et al, 2010).

A partir desta avaliação foram selecionados cinco padrões para uma descrição mais detalhada, adequando-os ao modelo apresentado na seção 3.1.2. Os demais foram mantidos no modelo mais simples sugerido em Portland (2014). A metodologia aplicada neste trabalho foi fortemente influenciada pela abordagem e embasamento pelo estudo de Dande *et al* (2014), visando propiciar futuras análises comparativas.

3.2.1 Detalhamento do processo de pesquisa

A pesquisa foi realizada conforma o fluxograma ilustrado na Figura 3-1. Cada etapa é detalhada a seguir.



Figura 3-1 Fluxograma do processo de pesquisa

3.2.1.1 Revisão da literatura

Mapeados os estudos existentes sobre startups de software. O resultado deste mapeamento pode ser consultado no Capítulo 2. Além da pesquisa literária, foram realizadas entrevistas não estruturadas com empreendedores de startups de software no Brasil, Estados Unidos, Reino Unido, Alemanha, Suíça, Malásia, Sérvia, Líbano e Chile, a fim de identificar citações bibliográficas espontâneas.

A citação espontânea mais recorrente foi ao livro “*The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*” de Eric Ries publicado inicialmente em 2011 pela Crown Business, apresentando um

modelo simples de operacionalização para startups baseado em ciclos de Deming, também conhecidos com PDCA (*plan-do-check-act*). Outra citação literária recorrente foi ao livro "*The Four Steps to the Epiphany: Successful Strategies for Products that Win*," de Steve Blank publicado pela K&S Ranch Press em 2005. Nenhum trabalho acadêmico foi citado espontaneamente.

3.2.1.2 Definição dos critérios de seleção das Startups

O primeiro critério foi definir “startup”, pois na literatura existem definições diversas. Neste trabalho foram consideradas duas definições amplamente reconhecidas pela comunidade:

- “*A startup is a human institution designed to deliver a new product or service under conditions of extreme uncertainty.*” Ries (2011)
- “*...a startup is an organization formed to search for a repeatable and scalable business model.*” Blank (2010).

Definiu-se assim startup como sendo uma *organização formada por pessoas com o objetivo de entregar um novo produto ou serviço, a partir de um modelo de negócios escalável e repetível, sob circunstâncias de extrema incerteza.*

O segundo critério adotado foi ter como objetivo de negócio principal o desenvolvimento funcional contínuo de software, de forma que o mesmo seja parte integrante e indissociável do modelo de negócios perseguido. Ressalta-se que neste critério *sites de e-commerce* tradicionais não foram considerados. Embora o software seja indissociável do modelo de negócios, tipicamente não apresentam desenvolvimento funcional contínuo.

A seleção da amostra foi realizada de forma contínua entre setembro de 2014 e junho de 2015 face o dinamismo do ecossistema de startups e ausência de bases de dados públicas que abordassem os critérios determinados e suas definições.

Foi realizada pesquisa contínua em sites Internet e eventos em busca de startups que atendessem aos critérios estabelecidos. Toda startup que se enquadrava nos critérios de seleção foi convidada a participar da pesquisa. Ao todo foram convidadas 93 startups de software, de diferentes países, conforme Tabela 3-2.

Tabela 3-2 Relação de startups convidadas por país

País	Startups convidadas
Brasil	59
Estados Unidos	18
Continente Africano	5
Suíça	3
Reino Unido	2
Alemanha	2
Servia	1
Malásia	1
Líbano	1
Austrália	1

3.2.1.3 Planejamento das entrevistas

Face à dispersão geográfica da amostra, foi definida utilização de um sistema eletrônico para fazer a coleta inicial de informações. Um formulário padronizado foi criado (disponível no Apêndice 1) a partir do modelo em Dande *et al* (2014). Os formulários foram enviados eletronicamente para os pontos de contato das startups convidadas.

Daquelas que responderam, foi realizada uma tentativa de entrevista com os colaboradores disponíveis, sempre buscando trazer para a entrevista alguém diretamente ligado às atividades de desenvolvimento de software, além do principal executivo (tipicamente um fundador ocupando o cargo de presidente ou *CEO*).

3.2.1.4 Execução das entrevistas

As entrevistas foram executadas entre setembro de 2014 e junho de 2015. Técnicas e ferramentas distintas foram utilizadas neste processo, tais como vídeo conferência, ligações telefônicas, reuniões formais, participação em eventos, entre outras.

Obteve-se uma taxa de aproximadamente 17% de retorno dos formulários preenchidos e cerca de 10% de entrevistas realizadas, ante o total de startups convidadas. Não foi possível validar estatisticamente a representatividade desta amostra face a dificuldade em estabelecer o conjunto universo dentro dos critérios estabelecidos.

A Tabela 3-3 apresenta os dados quantitativos da amostra e retorno por país participante. Além dos resultados do volume de respostas e entrevistas terem sido quantitativamente baixos, de acordo com a expectativa de retornos na ordem de 50%, a qualidade do preenchimento dos questionários também não atendeu o esperado.

Tabela 3-3 Questionários respondidos e entrevistas realizadas

País	Startups convidadas	Questionários respondidos	Entrevistas realizadas
Brasil	59	8	4
Estados Unidos	18	4	2
Continente Africano	5	1	1
Suíça	3	-	-
Reino Unido	2	1	-
Alemanha	2	1	1
Servia	1	1	1
Malásia	1	-	-
Líbano	1	-	-
Austrália	1	-	-
Total	93	16	9

3.2.1.5 Identificação dos padrões

Os dados obtidos na etapa anterior foram consolidados usando principalmente diagramas de Ishikawa (espinha de peixe) como ferramenta auxiliar. Os padrões candidatos foram identificados a partir do extrato dos diagramas gerados. No total 32 padrões candidatos foram identificados e estruturados utilizando a abordagem do formulário Portland (2014) citada.

Cada padrão candidato foi observado em pelo menos duas empresas distintas. Embora o foco tenha sido em startups de software, é possível que os padrões encontrados sejam observados em empresas que não se enquadrem nos critérios adotados neste trabalho, sejam já maduras ou startups. O fato de um padrão candidato poder ser encontrado em outros tipos de empresa não foi considerada uma razão para rejeita-lo neste momento, racional similar a Paternoster et al (2014).

Depois de identificar os padrões, os mesmos foram agrupados conforme a área de interesse ao qual estavam associados. Estas categorias representam essencialmente uma área de atenção e preocupação pertinentes a uma startup de software, e foram reconhecidas a partir da observação dos dados obtidos em si .

Assim, é possível que as categorias encontradas não abranjam necessariamente todos os aspectos de uma startup de software. Optou-se por utilizar as mesmas categorias identificadas por Dande et al (2014), quais sejam: Metas, Cultura, Financiamento, Cliente, Organização, Competência, Time, Processo, Projeto, Qualidade e Tecnologia. Durante esta etapa, alguns padrões foram mesclados, e outros foram removidos, resultando em um conjunto final de 27 padrões candidatos.

3.2.1.6 Seleção dos padrões

Tomando por base os padrões candidatos, uma nova rodada de consolidação conceitual foi realizada. O resultado obtido apresentou padrões demasiados generalistas em função da consolidação de conceitos dos distintos padrões em elemento mais amplos que comportassem todos os padrões candidatos vinculados.

Descartada a seleção dos padrões a serem detalhados com base na aglutinação, optou-se por definir um critério de seleção. O critério foi concentrar-se em padrões que são particularmente relevantes para startups e/ou que lidam particularmente com os desafios de Engenharia de Software. Considerando esta definição cinco padrões candidatos foram selecionados para detalhamento.

Cabe salientar, no entanto, que o processo de seleção foi superficial e não pretende ser cientificamente inequívoco. Um processo de seleção mais estruturado pode levar a um conjunto diferente tanto dos padrões organizacionais detalhados como dos próprios agrupamentos e classificações.

Outro aspecto relevante, que pode ser abordado em trabalhos futuros, foi a ausência de uma seção de retorno junto ao entrevistados ou mesmo uma seção conjunta com diferentes praticantes, a fim de debater os resultados. Em Dande et al (2014) foram realizadas seções conjuntas, bem como a equipe de pesquisa debateu internamente os resultados resultando em maior conhecimento agregado ao trabalho e validação das descobertas junto aos entrevistados.

3.2.1.7 Detalhamento dos padrões

Os padrões candidatos escolhidos para detalhamento foram analisados em maior profundidade e elaborados de forma mais completa usando a versão adaptada do formato Coplien (1996) conforme ilustrado na Tabela 3-1. No estudo europeu, cada

padrão foi objeto de um seminário, liderado por um especialista da equipe de pesquisadores e debatido com os demais participantes (Dande et al, 2014).

O trabalho de detalhamento retomou a análise dos dados obtidos preliminarmente na identificação dos padrões candidatos. Nesta análise, também foram agregados conhecimentos obtidos na revisão bibliográfica, bem como novas consultas a referências foram realizadas com o objetivo de enriquecer o detalhamento.

Ao final desta etapa os cinco padrões organizacionais selecionados inicialmente como pares de situação-reação foram estendidos do ponto de vista de conteúdo nas dimensões sugeridas no modelo apresentado, conforme Tabela 3-1.

3.2.1.8 Formatação e apresentação dos padrões

Os padrões organizacionais são normatizados e documentados de acordo com o modelo proposto e são apresentados no Capítulo 4. Os padrões candidatos, incluindo os que foram objeto de detalhamento, são apresentados na Seção 3.3, formatados conforme o modelo de Portland (2014), em pares situação-reação.

3.3 Padrões candidatos identificados

Nesta seção estão listados os padrões candidatos que foram identificados a partir da análise das pesquisas realizadas. Os padrões organizacionais submetidos a um maior detalhamento encontram-se relacionados no Capítulo 4. Os padrões candidatos foram organizados levando em consideração as categorias definidas em Dande et al (2014), já citadas anteriormente.

3.3.1 Metas

3.3.1.1 Foco do produto

- Situação: O foco do produto não é claro quanto existem vários segmentos de mercado e/ou clientes aos quais o produto pode ser direcionado. Alcançar todos os públicos é dispendioso.
- Reação: Identificar o segmento de maior potencial de atração de clientes utilizando grupos de foco, pesquisas automatizadas e técnicas de abordagem de campo. Manter demais segmentos em observação para abordagem em momento

futuro. Estar preparado para mudar o foco rapidamente usando métodos ágeis, conforme mais informações são coletadas. Manter o foco definido ao avaliar solicitações de mudança, internas ou externa, de acordo com a contribuição à visão principal.

3.3.1.2 Proposta de valor única

- Situação: A aquisição de um bem ou serviço pelo cliente é um processo de tomada de decisão de investimento governado pela percepção de valor. A proposta de valor não é clara e/ou não é bem comunicada. Ela não alcança as pessoas "certas" na organização do cliente.
- Reação: Criar uma proposta de valor única que seja percebida pelo cliente. Engajar profissionais dos níveis estratégicos e operacionais no cliente. Introduzir a proposição de valor para as pessoas que realmente podem criticá-la e tomar decisões sobre o investimento.

3.3.1.3 Evitar customizações

- Situação: Desenvolvedores são escassos. Variantes do produto em clientes distintos resulta em maior necessidade de manutenção e reduz a capacidade de escalar o produto. Startup cede aos anseios do cliente para não perder o negócio, mas este normalmente não se torna rentável.
- Reação: O produto deve ser único mas modular e altamente configurável/flexível. Um produto que permite diferentes configurações e modularizações por cliente ainda é um único produto. Desistir de clientes que gerem customizações muito específicas, pois reduz o potencial de rentabilizar a solução em outros clientes.

3.3.2 Cultura

3.3.2.1 Liberação controlada de atualizações

- Situação: Liberação de novas versões contendo erros ou funcionalidades incompletas. Indisponibilidade de todo o produto em função de parâmetros de produção configurados equivocadamente. Ausência de revisão de parâmetros obrigatórios para publicação do software nas lojas de aplicativos oficiais, tais

como AppStore (Apple) ou Google Play (Android), geram atrasos imprevisíveis na liberação.

- Reação: Estabelecer um processo rígido de liberação, que embora possa atrasar o feedback de clientes, eleva a disponibilidade. A contrapartida desta decisão leva a um ciclo mais longo de *feedback* junto aos clientes.

3.3.2.2 Criar a cultura de desenvolvimento

- Situação: Preocupação grande no “como”. Processos mudam radicalmente com a evolução do negócio.
- Reação: Conscientemente desenvolver desde o início uma cultura de desenvolvimento, de “fazer”, de “criar”, com menor preocupação no processo.

3.3.3 Financiamento

3.3.3.1 Autofinanciamento

- Situação: Agentes de financiamento estão relutantes em investir em uma prova de conceito do produto. Acesso a capital de terceiros demanda tempo e burocracia.
- Reação: Financiamento primeiro com o próprio capital e tempo. Convocar parentes e amigos para participar do negócio e fazer pequenos investimentos ou empréstimos.

3.3.3.2 Busca por investidores

- Situação: Os fundadores já assumiram o risco em deixar carreiras corporativas ou mesmo os estudos de lato para empreender. Existe a necessidade de mais recursos humanos e tecnológicos que demandam recursos financeiros.
- Reação: Identificar investidores iniciais (anjos) que possam se atrair pelo negócio. Avaliar a participação em editais de financiamento público voltados para inovação e desenvolvimento econômico.

3.3.3.3 Identificar compradores

- Situação: Incerteza sobre a viabilidade comercial do produto. Dificuldade de formular preço. Falta de experiência com vendas.

- Reação: Apenas começar a construir algo que é certo que venda. Usar "tempo livre" para buscar clientes ao redor e validar o produto e o preço. Obter clientes para financiar o seu produto inicial oferecendo vantagens.

3.3.4 Cliente

3.3.4.1 Focar em um nicho específico de clientes

- Situação: Os recursos disponíveis não são suficientes para construir um produto que satisfaça uma audiência global. Ações de divulgação exigem capital de forma proporcional ao público alcançado.
- Reação: Identificar um nicho de clientes no qual o produto tem grande aderência. Tornar o produto parte da fundamental da rotina deste segmento. A seleção de processos, métodos e tecnologias deve considerar o crescimento global.

3.3.4.2 Comunicação com o cliente

- Situação: Startups não contam com áreas de comunicação e relacionamento com o cliente. Fundadores não têm experiência no atendimento ao cliente.
- Reação: Forneça aos clientes maneiras naturais para dar *feedback*. Incentivar os contatos diretos por e-mail. Visitar frequentemente os escritórios dos clientes.

3.3.5 Organização

3.3.5.1 Organização horizontal

- Situação: Comunicação através de intermediários consome tempo adicional. Decisões são tomadas apenas por uma determinada pessoa.
- Reação: Não criar a cultura de hierarquia. Todas as pessoas estão empenhadas no bem comum, confiar que as suas decisões são acertadas. Desenvolver confiança e dar autonomia aos colaboradores. A pessoa que está enfrentando uma decisão importante tem geralmente a maioria da informação contextual.

3.3.5.2 Não crescer em pessoal

- Situação: Propensão a seguir as tradições e costumes do meio organizacional tradicional. Associação de crescimento do negócio com aumento de equipe.
- Reação: Uma equipe pequena pode dar conta do desenvolvimento não importa quantos clientes utilizem o software. Se não há necessidade de obter mais recursos ou competências, não se deve crescer a equipe.

3.3.6 Competência

3.3.6.1 Formar parcerias

- Situação: Necessidade de determinado tipo de serviços ou de especialização.
- Reação: Utilizar parceiros e outras startups com foco na permuta. O produto de um parceiro pode atender as necessidade de forma rápida e fácil. Identificar outras startups com produtos complementares.

3.3.6.2 Manter a vanguarda

- Situação: Dificuldade em manter o nível de atualização sobre novas tecnologias e oportunidade de aplicação. Falta de recursos para financiar participação em eventos ou cursos de aperfeiçoamento.
- Reação: Engajar em comunidades técnicas *online* e utilizar plataforma de ensino a distância especializadas (ex. Udemy).

3.3.6.3 Compartilhar ideias

- Situação: Ideias não comunicadas e elaboradas acabam não sendo desenvolvidas.
- Reação: Compartilhar ideias com todos que estiverem dispostos a ouvir. Não ser paranoico. É mais provável que se obtenha *feedback* valioso do que se tenha a ideia “roubada”.

3.3.7 Equipe

3.3.7.1 Desenvolvedores polivalentes

- Situação: Sempre há coisas diferentes para fazer. Não é possível ter um especialista para toda tarefa. Desenvolvedores se cansam rapidamente de fazer a mesma coisa.
- Reação: Buscar desenvolvedores com qualificações diversas e auto-motivados. Identificar a atitude em relação ao desenvolvimento de software e procurar aqueles que demonstrem paixão por criar. A vontade e determinação são muitas vezes mais importantes do que as habilidades específicas.

3.3.7.2 Começar com uma equipe pequena

- Situação: Construir uma solução completa em software requer habilidades distintas. No estágio inicial não se tem trabalho suficiente para determinadas especialidades.
- Reação: Ao invés de antecipar todas as habilidades necessárias, começar com uma pequena, mas experiente equipe, com uma comunicação eficiente e ampla escala de conhecimento e experiência. Terceirizar mão de obra para atividades que não façam parte do negócio principal.

3.3.8 Processo

3.3.8.1 Melhorias de processos no momento certo

- Situação: Em algum momento, preparando para o crescimento, a startup precisa planejar melhores processos. Mudar processos exige o foco e esforço das melhores pessoas.
- Reação: Retardar mudanças nos processos que não tenham uma clara justificativa de retorno do investimento. Utilizar ferramentas alinhadas ao momento da empresa e as melhores práticas. Conhecer as deficiências não críticas, mas ainda não fazer nada sobre elas.

3.3.8.2 Customize práticas ágeis

- Situação: práticas de livros didáticos podem não ser o melhor para a empresa como parecem ser de forma genérica. Uma startup não é genérica, e deve encontrar as práticas que refletem essa singularidade.

- Reação: Adaptar práticas ágeis comuns para a cultura e necessidades da startup. O “espírito de inovação” pode ser comprometido com rigidez de processos e práticas “*by the book*” (tal qual no livro, em tradução livre).

3.3.8.3 Falhar rápido

- Situação: fazer coisas erradas por muito tempo pode levar a grandes problemas, especialmente para startups, onde todos os recursos são bastante escassos.
- Reação: os colaboradores devem ter a liberdade de fazer avaliações do produto e do negócio rapidamente, parar imediatamente o que estão fazendo quando as coisas dão errado, e corrigir a abordagem. Este é um aspecto cultural, não uma característica do processo.

3.3.9 Projeto

3.3.9.1 Desenvolver apenas o que é necessário

- Situação: mesmo um produto inovador comporta inúmeras funcionalidades tradicionais.
- Reação: apenas desenvolver novas funcionalidades que contribuam diretamente com o foco definido. Não generalizar projetos. Não implementar para o próximo projeto.

3.3.9.2 Usar arquitetura do produto extensível

- Situação: o produto começa focado em pequeno grupo de funcionalidades especializadas. Novas demandas e clientes pressionam por mais funcionalidades.
- Reação: criar uma arquitetura modular que permita fácil extensão do projeto e reduza o refatoramento.

3.3.10 Testes

3.3.10.1 Buscar por erros

- Situação: a qualidade geral do produto diminui à medida que o desenvolvimento de novas funcionalidades é acelerado.

- Reação: organizar dias de testes intensivos onde todos devem atender, incluindo colaboradores não técnicos. Criar um fluxo de comunicação, a fim de evitar conflitos de código e redundância na correção de erros.

3.3.10.2 Automatizar testes

- Situação: os desenvolvedores são pressionados para concluir novas entregas. Boa parte dos desenvolvedores não gosta de fazer e executar planos de testes.
- Reação: Use as ferramentas técnicas estabelecidas (JUnit, Jenkins etc.) para testes automatizados. Estabelecer planos de teste juntamente com a especificação visando aprimorar o desenvolvimento. Para interfaces do usuário usar teste manual. Reservar um dia antes da implantação e concentrar toda a equipe nesses testes.

3.3.11 Tecnologia

3.3.11.1 Uso de tecnologias genéricas não proprietárias

- Situação: incerteza sobre exatamente quem vai utilizar o software e em quais dispositivos e plataformas. Necessidade de investimento para uso de plataformas e tecnologias proprietárias.
- Reação: Valorizar tecnologias independentes de plataforma, de código aberto e plataformas de desenvolvimento que permitam a publicação do software em dispositivos distintos.

3.3.11.2 Tecnologias escaláveis

- Situação: o universo de atuação inicial da startup é pequeno, mas o objetivo final é global.
- Reação: escolher técnicas de produtos, implantação e desenvolvimento, que sejam escalares com a mínima necessidade de manipulação de código, mas de forma que não comprometam os objetivos de curto prazo.

3.3.11.3 Tecnologias familiares

- Situação: Aprender novas tecnologias ou processos consome tempo e pode causar atrasos.

- Reação: Usar tecnologias e processos nos quais os membros da equipe tenham experiência.

3.4 Considerações finais

Diante da observação dos padrões candidatos identificados foi possível encontrar indícios de sobreposições e contradições aparentes. Isto decorre da essência empírica deste trabalho, pois as empresas entrevistadas têm formas distintas de lidar com as situações que encontram ao longo da jornada empreendedora.

Outro fato que pode contribuir para esta percepção é a dispersão geográfica dos respondentes, que pode implicar em divergências culturais significativas no que tange a forma de tomar decisões e lidar com o ambiente externo e interno.

Finalmente, cinco padrões foram selecionados para um maior detalhamento. Esta seleção foi tanto influenciada pela riqueza de dados disponíveis como pela relevância do assunto conforme observado na revisão bibliográfica e entrevistas realizadas.

4 Análise dos dados

Este capítulo apresenta os cinco padrões empresariais identificados nas startups de software como de maior relevância de acordo com o modelo definido e apresentado na Seção 3.1.2.

4.1.1 Criar a cultura de desenvolvimento

- Categoria: Cultura
- Contexto: Muitas vezes, quando uma nova empresa é formada, nenhum dos elementos da atividade organizacional clássica está presente - só há uma equipe de pessoas e as suas competências, um objetivo e algumas visões vagas de como proceder. Todo o resto deve ser construído.
- Problema: No início do ciclo de vida da empresa, os esforços conscientes da empresa são dispersos pois tudo é novo e precisa ser criado, desde o sistema de gestão até o software em si. Muito em breve a equipe deve ser capaz de produzir sistemas de software que oferecem um bom valor para os primeiros clientes. As decisões devem ser feitas baseadas no que é para desenvolver em primeiro lugar? E se os processos forem a prioridade?
- Forças: Logo no início, as coisas são simples. Não há clientes cuja situação exija que os desenvolvedores parem de se concentrar no desenvolvimento em detrimento da operação. Mas as coisas vão mudar com o sucesso dos primeiros desenvolvimentos e tudo o que auxilia nas competências da empresa deve ser pensado para o futuro. Os processos são necessários quando as coisas ficam mais complicadas, o produto cresce e há novos clientes. Mas como os processos devem ser ainda é desconhecido, eles vão apenas apoiar as necessidades futuras de desenvolvimento e de negócios. Há uma necessidade de construir os pilares do sucesso que vão levar a empresa através de todas as mudanças, ajudar a ser flexível e a trabalhar em equipe e com os clientes. Startups são geralmente

baseados em pessoas, metas compartilhadas e uma forma de canalizar a motivação dos indivíduos em resultados, no caso em um *software*. Essa visão e organização voltada ao sucesso se contradiz com os pontos de vista da engenharia de software tradicional.

- Solução: Desde o início propagar a cultura organizacional que reflita o sucesso que se pretende atingir. Este sucesso está vinculado a um produto ou serviço de software. Assim, a cultura deve refletir o foco no desenvolvimento, para manter a evolução contínua e aspectos inovadores. Desenvolver processos apenas quando chegar a fase de estabilização dos negócios.
- Consequências
 - A cultura fornece recursos que são essenciais para os primeiros passos de um empresa e permite a construção de práticas em uma base sólida.
 - Foco demais na cultura leva a dívidas técnicas. As pessoas chave devem estar cientes da trajetória da empresa, às fases e necessidades de cada etapa de seu ciclo de vida, isso pode ser difícil de ver se toda a energia vai para as tarefas diárias.
 - Processos fracos fazem com que equipe e competências pessoais sejam críticos. A colaboração deve funcionar e os desenvolvedores devem ser muito competentes.
- História real: A cultura é um diferenciador chave que oferece suporte a todos os aspectos do negócio. Uma determinada startup tem apenas um processo documentado em apenas uma folha de papel. Todo novo colaborador recebe este “manual” ainda no processo de contratação. O manual aborda um único tópico: como se comunicar, além de trazer o lema da empresa – “desenvolvendo o amanhã, hoje.”. Isso não significa que as práticas padronizadas ou fluxos de trabalho não são desenvolvidos. Eles são, mas de uma forma flexível, contando com as equipes para escolher organicamente a forma como eles funcionam. A mensagem passada pela empresa com o seu manual é exatamente o foco no desenvolvimento. Os fundadores quiseram deixar claro que toda a energia deve estar focada na criação e desenvolvimento do produto, e não em atividades orbitais, que no entendimento deles podem sempre ser rapidamente resolvidas com uma comunicação eficiente.

4.1.2 Autofinanciamento

- Categoria: Financiamento
- Contexto: A empresa é na verdade uma ideia ainda e precisa de recursos para alcançar o próximo estágio. O negócio e seu estágio atual são passíveis de serem autofinanciados. Este padrão ocorre no momento de formação da empresa efetivamente; quando o conceito é abstrato demais para o momento e difícil de vender a investidores; ou quando os fundadores querem liberdade plena na tomada de decisão.
- Problema: O problema principal é a falta de recursos financeiros para sustentar a criação de um novo produto. Em paralelo existe a questão do controle acionário da empresa que implica na distribuição de poder e tomadas de decisão. E por fim a questão de autoconfiança, em dedicar-se a algo que pode ou não ser bem sucedido.
- Forças: As empresas geralmente precisam de financiamento para se construir e crescer. As pessoas que trabalham na empresa precisa de dinheiro para viver. Obter financiamento externo não é simples e exige garantias, elevando o risco sistêmico. Investidores externos exercem influência no negócio e consomem parte da energia que deveria ir para o desenvolvimento. Extrema limitação de recursos financeiros gera necessidade de criar soluções viáveis economicamente com mais agilidade.
- Solução: Não buscar financiamento externo, financiar o negócio com recursos próprios dos fundadores, seja com reservas ou por meio de trabalho externo (como consultorias pontuais ou *freelancers* por exemplo). Buscar financiamento junto a clientes, antecipando receitas.
- Consequências: Não depender de financiamento externo traz liberdade na condução da startup. Obriga a encontrar maneiras de se ganhar dinheiro. Autofinanciamento por muito tempo pode dificultar o crescimento e deixar que concorrentes com acesso a mais recursos possam alcançar o produto e se equiparar em termos de inovação. No pior cenário, a startup não consegue alcançar o mercado com um bom produto ou antes da concorrência e os fundadores vão a falência.
- História real: Os fundadores de uma startup que idealizou um aplicativo para celular financiaram o projeto por cerca de oito meses antes que um investidor anjo investisse capital próprio. Após a entrada do investidor a startup durou mais

cinco meses e então encerrou as atividades. Segundo um dos fundadores, o investidor não estava preparado para a situação e acabou por consumir energia demais da equipe, levando a desmotivação e consequente impacto na qualidade e agilidade em lançar o produto.

4.1.3 Organização Horizontal

- Categoria: Organização
- Contexto: a startup está crescendo e já avançou a caminho do sucesso. A organização cresce e já tem em torno de 20 pessoas. A cultura de desenvolvimento está disseminada, e assim não existe muito formalismo e processos. A comunicação torna-se lenta à medida que mais e mais pessoas são contratadas. Além disso, alguém precisa tomar decisões de negócios, e são cada vez mais decisões importantes. Os fundadores não têm mais tempo para serem envolvidos em todas as decisões.
- Problema: À medida que a organização cresce, uma ou algumas pessoas podem não serem capazes de fazer “tudo” ou tomar todas as decisões, pois passa a não ser mais possível dominar todas as coisas que acontecem na organização.
- Forças: Comunicação através de intermediários impacta no tempo de resposta e qualidade da comunicação. Se todas as decisões têm de ser abordadas a uma determinada pessoa, a tomada de decisão na organização torna-se lenta. Como a organização é uma startup a tomada de decisão deve ser rápida, sob risco de perder a vantagem competitiva. Concentrar todas as decisões em poucas pessoas cria um gargalo conforme a empresa cresce. O melhor tomador de decisão geralmente é a pessoa preocupada e afetada pela decisão a ser tomada. Assim, os desenvolvedores devem ter a autoridade para tomar decisões que dizem respeito à sua forma de trabalhar ou a parte do produto que estão trabalhando. Os tempos de resposta aumentam à medida que deve haver passagem da comunicação entre os membros. O último momento propício pode já ter passado quando a decisão é tomada. A autonomia é um bom motivador para os trabalhadores do conhecimento. Como é interesse da startup manter as pessoas motivadas, autonomia suficiente deve ser dada para os desenvolvedores.
- Solução: A fim de manter a comunicação rápida e permitir que as pessoas tenham autonomia no seu trabalho, manter a estrutura de organização horizontal (sem hierarquias). Como as pessoas são comprometidas com o bem comum,

confiar que as suas decisões são acertadas e elas podem fazê-las de forma autônoma. A pedra angular da organização horizontal está fundida em uma Comunidade de Confiança, conforme Coplien & Harrison (2004). Desenvolvimento cultural deve vir antes do processo. A cultura da organização orienta os indivíduos para tomar decisões corretas. Deixar pessoas se auto-organizarem em equipes ou com uma sutil orientação. Se a equipe precisa de um líder, ela auto-seleciona na própria equipe. Há apenas alguns papéis coadjuvantes na empresa e os desenvolvedores são todos pares. Conforme a empresa cresce pode haver necessidade de alguns papéis de apoio que cuidam de rotinas diárias da empresa, permitindo que as pessoas executem o seu melhor. Eles não devem tentar conduzir as equipes ou tomar decisões por eles. A pessoa que está enfrentando uma decisão importante pode precisar de ajuda e consultoria da visão estratégica, mas ela não precisa que as decisões sejam tomadas por ela. Uma organização horizontal dá a oportunidade para equipes decidirem sobre questões importantes para si mesmos.

- Consequências: As decisões são tomadas rapidamente pelas pessoas que têm o contexto de informações necessário. Isso permite à organização reagir rapidamente às mudanças no ambiente de negócios. Sem burocracia ou estranhos envolvidos na tomada de decisões. Por exemplo, se alguém precisa de um novo equipamento, o compra diretamente. Tal abordagem requer confiança e os funcionários precisam ser confiáveis. Os novos funcionários podem se sentir desorientados ao serem contratados. Não existem gestores para orientar o novo desenvolvedor para seu trabalho. Ele precisa descobrir sua própria posição na equipe. Este problema pode ser reduzido com um manual para o novo funcionário explicando como a organização funciona. Se houver membros de outras organizações no esforço inicial, pode ser difícil de alcançar a estrutura de organização horizontal para a startup. Como existem diferentes fatores em jogo, com a quantidade de financiamento, experiência anterior, penetração no mercado, entre outros, pode levar à situação em que uma parte pode querer mais poder de tomar decisões e, assim, impedir o uso do modelo de organização horizontal. A autonomia das pessoas também traz maior responsabilidade aos indivíduos. Deve-se dar crédito para boas decisões, e cautela para não punir as pessoas por causa de más decisões. Cada indivíduo deve assumir a responsabilidade de más decisões e cuidar do consequências. Como as pessoas

podem decidir por si próprios, o processo de tomada de decisão pode ser influenciado mais fortemente pelos aspectos locais. Assim, deve-se certificar de que a visão de futuro e a cultura estão bem comunicados para cada funcionário, de forma que possam tomar decisões orientadas também pela estratégia e objetivos de negócio.

- **História real:** Uma startup de software pequena tinha cerca de 8 funcionários. Uma vez que a empresa começou a crescer e alcançou cerca de 20 funcionários, perceberam que precisavam parar e decidir sobre a estrutura de organização. Eles decidiram usar o padrão de organização horizontal para formar a estrutura. A diretriz foi não forçar quaisquer funções a qualquer um dos funcionários, permitindo a qualquer pessoa decidir sobre assuntos diferentes. Se fosse necessário um gerente de projeto, alguém se tornaria então gerente de projeto para o projeto específico. Os desenvolvedores podem decidir por conta própria como organizar o trabalho ou quando comprar novos computadores. Cada decisão tinha de ser racionalizada. A startup tinha papéis coadjuvantes, como um gerente de RH que foi responsável pela contratação de novos desenvolvedores. Como a legislação ordenou, a empresa também teve que nomear um CEO. Além disso, a startup contratou um gerente de vendas dedicado, cujo trabalho era vender novos projetos. Por fim, eles tiveram que criar um papel de "zelador", cujo trabalho era cuidar de todos os outros assuntos que ninguém iria cuidar.

4.1.4 Proposta de valor única

- **Categoria:** Competência
- **Contexto:** O mundo já está cheio de boas ideias as quais diversas pessoas e organizações visam executar. A Internet coloca vários mercados a disposição de soluções de software locais. Nesta configuração, uma proposta de valor única combinando valores técnicos e sociais é necessária para garantir a diferenciação e sustentabilidade do negócio a longo prazo.
- **Problema:** Startups tem que se destacar na competição com os agentes de mercado estabelecidos. Adotar uma abordagem viável ao problema endereçado não é simples, é preciso conhecimento do mercado, expertise técnica e acesso a potenciais clientes.
- **Forças:** Competência distintiva é necessário para lidar com o dinamismo imposto pelo mercado. Adoção à evolução das necessidades do mercado é

essencial para lidar com concorrentes. Os agentes estabelecidos tem muita inércia e geram muita gravidade.

- Solução: Encontrar uma abordagem única que diferencie a startup no mercado, juntamente com uma forma eficaz de comunicar esta abordagem. Tornar a abordagem visível como uma parte da marca, e cultivá-la diariamente na empresa. Encontrar uma forma consciente de ação/competência para praticar a abordagem adotada, destacando a startup de forma eficaz dos concorrentes. Enfatizar as habilidades de comunicação a fim de compreender melhor os clientes e suas necessidades. Comercializar a competência como um serviço para o cliente. Desenvolver gradualmente competências, práticas e ferramentas dentro do cotidiano da startup.
- Consequências: a startup destaca-se no mercado e encontra clientes de maneira original e natural. Adotar uma forma única de “fazer as coisas” garantem uma maior motivação e satisfação para os funcionários e clientes. A construção da proposta de valor efetivamente única pode exigir recursos adicionais.
- Exemplo história: Uma empresa que desenvolve uma solução de inteligência para varejistas do mundo real adotou uma proposta de valor única, simples e eficiente. Em um mercado saturado de novidades, a empresa encontrou seu valor único na simplicidade de contratação, instalação e uso de sua solução. A proposta de valor é simples de ser percebida por seus clientes alvo, ela simplesmente o ajuda o varejista a vender mais. Ou seja, o que ela entrega ao cliente não é um belo portal de inteligência de negócios, mas sim mais vendas.

4.1.5 Desenvolver apenas o que é necessário

- Categoria: Projeto
- Contexto: startups frequentemente tem que lidar com a extensibilidade dos seus produtos. É claro desde o início que as versões iniciais serão estendidas mais tarde para determinados clientes ou ao público em geral. Esta é uma questão presente em todas as fases do negócio, pois sempre há algo a desenvolver, seja oriundo da criatividade dos colaboradores ou por demanda do mercado.
- Problema: Como resolver a questão extensibilidade? Como encontrar a melhor abordagem para desenvolvimento que faz com que a extensão e alterações sejam tão fáceis e ágeis quanto possíveis?

- Forças: o futuro é desconhecido, e é difícil planejar para o que não se conhece. Planejar muito adiante pode levar a desenvolver algo que não é necessário. Tradicionalmente generalizar os projetos, pensando que as novas implementações, extensões e alternativas poderiam ser deduzidas a partir da generalização e então pela *refatoração* da implementação atual. No entanto, os recursos e especialmente o tempo disponíveis são apenas suficientes para criar o que é necessário no momento atual.
- Solução: extensibilidade eficiente é desenvolver apenas o que que for necessário (será utilizado) logo, ou seja, pelo primeiro cliente ou próximos clientes. A próxima versão deve incluir o que as histórias dos usuários atuais exigem. Não generalizar projetos, ou implementar para o próximo projeto. Planejar somente para o que é conhecido.
- Consequências: melhora a eficiência do desenvolvimento e aumenta a motivação dos desenvolvedores. O desenvolvimento é rápido, já que menos tempo é gasto em arquitetura e design - o que poderia revelar-se mais tarde ter sido gasto em vão. Esta abordagem pode gerar déficits técnicos quando levada muito longe, especialmente se a comunicação dos desenvolvedores não for contínua, impactando normalmente na performance e capacidade de integração do software. A ênfase colocada sobre arquitetura deve prever a complexidade para adicionar e remover componentes e a escalabilidade das tecnologias selecionadas.
- História real: Uma startup que desenvolve sistemas de informação para indústrias enfrentou o problema de como evoluir seus sistemas depois de entregues e com isso alavancar novas receitas. A dúvida foi se deveriam abstrair o processo de negócio sendo abordado pela aplicação baseando a primeira entrega já neste modelo “escalável” e abstrato, ou se focavam efetivamente nas necessidades concretas identificadas. A empresa encontrou na segunda opção uma tática bem sucedida, ao contrário dos primeiros pensamentos dos fundadores. A decisão foi bem sucedida graças ao bom nível dos desenvolvedores, que mesmo estando preocupados em resolver a necessidade pontual, souberam preparar a aplicação para ser extensível e escalável desde o início.

5 Conclusão

As práticas de trabalho nas startups de *software* foram empiricamente exploradas e representadas em 27 padrões organizacionais sintéticos, sendo cinco destes analisados mais detidamente. Este deve ser visto apenas como um primeiro passo na identificação de um conjunto de padrões organizacionais efetivos nas startups de software.

Observou-se uma extensa preocupação em alinhar os objetivos do negócio, e consequentemente as funcionalidades e experiência de uso do *software*, aos anseios do mercado, com o objetivo de criar uma solução economicamente viável. Em outras palavras: “que venda”.

A preocupação em criar com agilidade e qualidade um *software* que possa ser entregue ao cliente para que se inicie um processo de faturamento o quanto antes foi notada extensivamente. Este fenômeno está diretamente associado à natureza autofinanciada da maior partes dos negócios analisados. Quando os recursos dos fundadores estão financiando diretamente o negócio a estratégia de *go to market* (saída ao mercado, em tradução livre) é acelerada.

Do ponto de vista da engenharia de software e seus processos, foi percebido um grande desprendimento formal em relação ao que é praticado efetivamente neste campo. Embora o perfil de ao menos um dos fundadores tipicamente indicar formação técnica formal, se observou que dentro da amostra avaliada as práticas de engenharia de software não tem rigidez em relação aos processos e métodos. O que importa ao empreendedor é agilidade e praticidade.

Por fim, o trabalho apresenta de forma objetiva as práticas adotadas por outros empreendedores, formando um conjunto de padrões organizacionais para startups de software. O conhecimento produzido até aqui tanto pode contribuir com outros trabalhos acadêmicos como pode ser consumido por empreendedores visando ampliar seu leque de ferramentas no apoio a tomada de decisão.

5.1 Considerações finais

Considera-se satisfatório o resultado final deste trabalho, em especial por contribuir a uma área já estudada e em franca ascensão no exterior e ainda com poucos estudos publicados nacionalmente.

Os empreendedores brasileiros que vislumbrem se aventurar em negócios focados no desenvolvimento de software podem contar com o presente trabalho como uma base de conhecimento a qual podem consultar para tomar decisões com maior embasamento e referência.

5.2 Limitações do projeto

A quantidade de startups que contribuíram para o trabalho não foi satisfatória, e desta forma, os resultados podem apresentar um viés a priori imperceptível, dado o desvio estatístico. Adiciona-se a este risco o fato que não foi realizada comparação aos resultados obtidos em outros trabalhos congêneres. Desta forma, é importante salientar que os padrões organizacionais constantes neste trabalho refletem tão somente a prática das empresas pesquisadas e seus colaboradores.

5.3 Trabalhos futuros

Os padrões escolhidos para uma apresentação mais refinada ainda precisam de uma análise mais aprofundada, e muitos dos padrões candidatos ainda são apenas ideias áspers. A extensão da amostra e seleção somente de empresas nacionais podem trazer resultados mais aderentes a realidade nacional.

Alguns trabalhos que podem ser considerados incluem: análise comparativa com os resultados de outros estudos e publicações; análise dos padrões organizacionais em startups bem sucedidas e fracassadas, visando identificar comportamentos e práticas que possam preponderar para o sucesso ou falha; análise dos padrões identificados em startups e em empresas já estabelecidas; correlação entre os padrões identificados com o objetivo de verificar a existência de fluxos e/ou hierarquias relevantes.

Referências Bibliográficas

GAGE, Deborah. The Venture Capital Secret: 3 Out of 4 Start-Ups Fail. Disponível em: <<http://www.wsj.com/articles/SB10000872396390443720204578004980476429190>>. Acessado em: 20 nov. 2014.

LEE, Aileen. Welcome to the unicorn club: laerning from billion-dollar startups. Disponível em: <<http://techcrunch.com/2013/11/02/welcome-to-the-unicorn-club/>>. Acessado em: 20 nov. 2014.

SILVA, Francisco Aguiar de Melo e. Fatores que contribuem para o insucesso das startups: o reverso da "medalha". Braga, 2013. 210p. Dissertação (mestrado em gestão) - Universidade do Minho - Escola de Gestão e Economia, Portugal, Braga, 2013. [Orientadores Porfessor Doutor José Carlos Pinto e Dra. Elisabete Sampaio de Sá]. Disponível em: <<http://hdl.handle.net/1822/25153>>. Acessado em: 22 nov. 2014.

HIRAI, Akira. What kills startups? Disponível em: <<http://www.caycon.com/downloads/What-Kills-Startups.pdf>>. Acessado em: 22 nov. 2014.

RIES, Eric. Lean Startup - How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses. Crown Business 2011.

BLANK, Steve; DORF, Bob. The Startup Owner's Manual: The Step-By-Step Guide for Building a Great Company. K&S Ranch 2012.

COLEMAN, Gerry; O'CONNOR, Rory V. An investigation into software development process formation in software start-ups. Journal of Enterprise Information Management, Vol. 21 Iss: 6, p 633 - 648. 2008. Disponível em: <<http://dx.doi.org/10.1108/17410390810911221>>. Acesso em: 22 nov. 2014.

PATERNOSTER, NICOLÓ et al, Software development in startup companies: A systematic mapping study. Information and Software Technology v. 56 p. 1200–1218. 2014. Disponível em <<http://www.sciencedirect.com/science/article/pii/S0950584914000950>>. Acesso em: 22 nov. 2014.

DANDE, Anuradha et al. Software Startup Patterns – An Empirical Study. Tampere University of Technology. Department of Pervasive Computing. Report 4. 2014. Disponível em: <<https://dspace.cc.tut.fi/dpub/handle/123456789/22067>> Acesso em: 22 nov. 2014.

(Alexander et al, 1977) Alexander, C., Ishikawa, S., Silverstein, M., Jacobson, M., Fiksdahl-King, M. and Angel, S.: A Pattern Language: Towns, Buildings, Construction. Oxford University Press, New York, 1977

Portland Repository, Disponível em <<http://c2.com/ppr/about/portland.html>>, Acessado em: 30 jun. 2014.

IDC, “Mercado Brasileiro de Software - Panorama e Tendências / edição 2015. Prévia dos dados de 2014”, Disponível em: <<http://www.abesssoftware.com.br/dados-do-setor/dados-2014>>, Acessado em: 30 jun. 2015.

Coplien, J.O.: Software Patterns. SIGS Books, New York, 1996

Coplien J.O., Harrison, N.B.: Organizational Patterns of Agile Software Development. Lucent Technologies, Pearson Prentice Hall 2005

G. Ganapathy, C. Surianarayanan, "An approach to identify candidate services for semantic web service discovery", IEEE international conference on service-oriented computing and applications, Perth, Austrália, 2010.

MEIRA, S. Novos negócios inovadores de crescimento empreendedor no Brasil. Rio de Janeiro: Casa da Palavra. 2013.

BARBIERI, J. C.; ANDREASSI, T.; SIMANTOB, M.; ÁLVARES, A. C. T.; NOBRE FILHO, W. Organizações inovadoras: estudos e casos brasileiros. (2 ed.). Rio de Janeiro: FGV, 2004

DRUCKER, P. A era da descontinuidade. Rio de Janeiro, Zahar 1974.

KIM, W. C., & MAUBORGNE, R. A Estratégia do oceano azul- como criar novos mercados e tornar a concorrência irrelevante. Rio de Janeiro: Editora Campus, 2005

Organização para a Cooperação e Desenvolvimento Econômica - OCDE. Manual de Oslo: proposta de diretrizes para coleta e interpretação de dados sobre inovação tecnológica. 3. ed. Rio de Janeiro: FINEP, 2004.

OSTERWALDER, A.; PIGNEUR, Y. Business Model Generation. London: Wiley John&Sons. 2010.

SCHUMPETER, J.A. Teoria do desenvolvimento econômico. São Paulo: Nova Cultural,1985.

ZACHARY, G. P. A geografia Global da Inovação. Disponível em: <http://www.embaixadaamericana.org.br/HTML/ijse1109p/zachary.htm>, E-Journal Embaixada Americana – Publicada em Wall Street Journal, 2012.

J. Bosch, H.H. Olsson, J. Björk, J. Ljungblad, The early stage software startup development model: a framework for operationalizing lean principles in software startups, in: Lean Enterprise Software and Systems, Springer, 2013.

May B.: Applying Lean Startup: An Experience Report. In: Proc. Agile Conference '12, IEEE CS, 2012.

Apêndice 1: Formulário de entrevista

Este formulário foi baseado no trabalho de Dande *et al* (2014), e buscaou manter compatibilidade das questões formuladas, permitindo futuros estudos comparativos de forma mais direta e integrada.

Introdução

O objetivo deste estudo é identificar e analisar as práticas de desenvolvimento de software e gestão praticados pela startup. A hipótese é que essas práticas diferem pelo menos até certo ponto, das práticas na indústria de software mais estabelecida. Para analisar a lógica das práticas, a entrevista irá também abranger questões relacionadas à composição da empresa.

Os resultados deste estudo serão distribuídos para as empresas participantes e, possivelmente, mais tarde publicados em fóruns científicos, mas toda informação específica da empresa ou que permita sua identificação permanecerá exclusivamente com a equipe de pesquisa.

1 Informações básicas da entrevista

Companhia

Data e hora

Os entrevistados e sua papéis na empresa

Entrevistador

2 Antecedentes da Startup

2.1 Dados empresa

Dê um breve histórico da empresa (como você começou), cobrindo:

Idade da empresa

Visão

Área de negócio

Modelo de negócio (atual e desejado)

Se atual e o desejado diferem, perguntar depois sobre práticas de engenharia de software que ajudam a empresa a atingir o objetivo desejado

Qual é o seu caminho para alcançar seu objetivo futuro desejado?

Você está planejando para atingir novos mercados, novo tipo de clientes? O que e como?

Fase atual (inicialização, estabilização, o crescimento) A fase de arranque real significa que você está apenas a formação da empresa em todo o fundador da equipe, em fase de estabilização de criar infra-estruturas, processos e tem um casal de clientes, em fase de crescimento que você está crescendo.

2.2 Pessoal

Número de funcionários e suas funções (papéis possivelmente primários e papéis secundários)

Como experientes são os seus empregados (baseado em idades profissional dos trabalhadores?)

Os empregados têm experiência em engenharia de software?

2.3 Clientes

Número de clientes

Qual é o segmento de clientes? Quem você criar produtos para?

Como você está única e se destacar na competição? Qual é a sua proposição de valor?

Você tem parceiros de negócios?

2.4 Produto

Qual é o tipo de seu produto de software (web, alguns, móvel, tradicional)

Se você pode tirar uma foto de você produto ecossistema e esclarece alguma coisa, fazê-lo.

O que é o domínio do seu produto (empresa, entretenimento, ...)

2,5 desafios de negócios

Que desafios o seu negócio presente ao desenvolvimento de software e como você abordá-los?

A curto prazo (o que é essencial para sobreviver a vinda mês (s)) e longo prazo (o que é essencial para a estratégia de longo prazo, o crescimento, etc.).

possíveis áreas de desafio: desafios financeiros (financiamento, orçamento), os desafios de pessoal, os desafios relacionados ao usuário, os desafios da concorrência, desafios técnicos

O que significa, especialmente s / w desenvolvimento que você está usando para enfrentar esses desafios?

3 Desenvolvimento de Software

Que aspectos da engenharia de software que você encontra o mais importante para a sua empresa:

A: Metas

B: Projeto

C: Desenvolvimento

D: Qualidade

E: Versionamento e gerenciamento de código

Vamos passar por definir a questão na ordem implícita pelas marcas da empresa, as primeiras questões relacionadas às áreas temáticas que a empresa considera importantes.

E as perguntas no tópico geral vamos discutir último com todas as empresas.

3.1 Metas

Como é que você venha com o seu conceito de produto e escopo? (Por quem, onde, dinâmica de criação, pessoas-chave envolvidas ...)

Como você testar / avaliar as idéias preliminares? (Proto, demonstração, MVP, grupos focais, principais clientes ...)?

Como você responder a novas solicitações de recursos?

Como você manter-se com a evolução das necessidades dos clientes?

O que você faz se você vê que algum recurso não é usado ou realmente popular?

Como você priorizar desejos e necessidades do cliente?

Como você coletar feedback dos clientes? Que tipo de feedback (são pessoas usando o recurso, como você controlar isso).

O que você está fazendo em termos de engenharia de software para chegar ao seu modelo de negócio desejado?

3.2 Projeto

O que é a plataforma para o produto? Por que você escolheu? (por exemplo, Supercell focada em iOS como plataforma primária porque tem o maior mercado)

Você tem múltiplas plataformas para o seu produto? Como você lida com isso?

O que é e por que você escolheu as tecnologias-chave para o produto?

Como você mantém o produto simples o suficiente para garantir curto espaço de tempo para o mercado? O que está no produto que permite um tempo mais curto para o mercado.

Como você mantém o produto suficientemente flexível para permitir novos recursos?

Descreva as suas práticas para a concepção de produtos desejáveis. Como você sabe satisfação e experiências dos usuários em relação ao seu produto?

Que tipos de conceitos de solução (algoritmos, arquiteturas, bibliotecas adquiridos, serviços, etc.) que você usa no produto para satisfazer as metas?

3.3 Desenvolvimento

Você pode descrever seu processo de desenvolvimento? Por que você escolheu esse processo? Quais são as características do processo que, especialmente servem os objetivos da empresa?

Você está seguindo algum processo padrão (por exemplo, Scrum), e em caso afirmativo, como você tem que adaptar e por quê?

Como adaptável é a sua maneira de desenvolver produtos para alterar situações (por exemplo, o rápido crescimento do mercado)? O que o torna adaptável?

Como você gerencia a dar seus desenvolvedores a possibilidade de concentrar-se no desenvolvimento de produtos e não para agradar a clientes individuais?

Como você manter o conhecimento fundamental e competência sobre o produto e processos na empresa? (Por exemplo, se uma pessoa chave folhas)

3.4 Qualidade

Que métodos para medir a qualidade do produto que você está usando?

Como você testar o seu produto? Você usa o teste automático?

Como você impede dívida técnica? Como você lida com qualquer dívida técnico acumulado?

Como você mantém a qualidade do produto no longo prazo?

3.5 Implantação

O que provoca a liberação de seu produto? Você tem entrega contínua?

Como você implantar novas versões para os clientes?

Como você minimizar o tempo de inatividade durante a implantação?

3.6 Versionamento e gerenciamento de código

Como você gerenciar diferentes versões do produto?

Você remover o código de recursos que não estão mais disponíveis para os usuários?

Você tem práticas para organizar código para que ele apóie o contínuo melhoramento, agilidade e crescimento rápido?

Como você minimizar o "código legado infestação"? (= Manter a base de código fresca e livre de código que ninguém quer tocar)

3.7 Geral

Será que você tente algumas práticas de processos que não funcionam?

Você está usando algumas outras práticas características de sua empresa, não tão longe discutido?

Como você gerenciar o crescimento da equipe? (Funções, tarefas de transferência de competência, a equipa dinâmica ...)

3.8 Outras questões

Qualquer outra coisa que você acha que é essencial?