

SÁVIO DA SILVA ARAÚJO

DESIGN SPRINT: ESTUDO DA APLICAÇÃO NA SENSEDIA

BACHARELADO

EM

CIÊNCIA DA COMPUTAÇÃO

FIC – MINAS GERAIS
2016

SÁVIO DA SILVA ARAÚJO

DESIGN SPRINT: ESTUDO DA APLICAÇÃO NA SENSEDIA

Monografia apresentada à banca examinadora da Faculdade de Ciência da Computação das Faculdades Integradas de Caratinga como exigência parcial para obtenção do grau de bacharel em Ciência da Computação, sob orientação da professora Msc. Fabrícia Pires Souza.

SÁVIO DA SILVA ARAÚJO

DESIGN SPRINT: ESTUDO DA APLICAÇÃO NA SENSEDIA

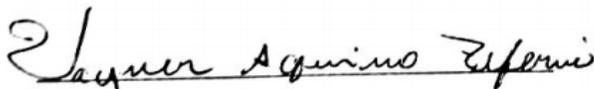
Monografia submetida à Comissão examinadora designada pelo Curso de Graduação em Ciência como requisito para obtenção do grau de Bacharel.



Prof^ª. Msc. Fabrícia Pires Souza Tiola
Faculdades Integradas de Caratinga



Prof^º Maicon Vinicius Ribeiro
Faculdades Integradas de Caratinga



Prof^º Wagner Aquino Zeferino
Faculdades integradas de Caratinga

AGRADECIMENTOS

Agradeço primeiramente a Deus pela minha existência. A minha mãe Efigênia que sempre acreditou em mim e na minha força de vontade, me incentivando sempre. Ao meu pai Milton pela compreensão de minha ausência no trabalho e por todo apoio que me deu durante todo esse percurso.

Agradeço ao pessoal da Sensedia, que acolheram meu trabalho, me dando a oportunidade de desenvolver este estudo.

Agradeço também a todos os professores e colegas de turma que estiveram comigo, em especial a minha professora e orientadora que sempre caminhou do meu lado durante todo o projeto.

RESUMO

Tentar compreender as necessidades do cliente para um sistema ser desenvolvido não é uma tarefa simples, as empresas de tecnologia da informação enfrentam alguns problemas para que tenham êxito nesse objetivo. Falhas podem ser originadas e conseqüentemente afetar o *software* de modo geral colocando em risco a integridade do produto de *software*. Tais situações geralmente decorrem no momento da análise e especificação dos requisitos, onde, os usuários podem não saber o que realmente eles querem, ou podem encontrar dificuldades para se expressarem. Mudanças de requisitos, pouca objetividade na especificação dos mesmos e modificações nas configurações do projeto conforme surgirem novos requisitos são alguns dos fatores preponderantes para que aconteçam falhas no projeto. Com base no cenário descrito acima, um estudo de caso foi realizado na Sensedia onde é abordada uma situação encontrada na empresa a partir da aplicação do *design sprint*, sob a ótica da dinâmica de sistemas, com o intuito de criar modelagens que podem auxiliar no entendimento do processo no que se refere à aplicação do *design sprint*, e também de criar conceitos que possam ajudar a empresa no entendimento do episódio estudado.

Para realização deste estudo, além do cenário encontrado na empresa, também foram modelados pressupostos elaborado por membros da organização considerando o problema onde existe uma pressão do cliente para entrega de novas versões do *software* e a passagem do tempo na qual o projeto pode sofrer atrasos, desobedecendo a um cronograma. Para que no fim seja possível analisar também outras vertentes gerando um aprendizado que pode servir de estratégia de ação.

Palavras-chave: design sprint, requisitos, processo, Sensedia.

LISTA DE ILUSTRAÇÕES

Figura 1: Exemplo de relações de mesmo sentido e sentido oposto.	19
Figura 2: Exemplo relação com <i>delay</i>	20
Figura 3: Exemplo de diagrama de causalidade.	20
Figura 4: Representação da estrutura do arquétipo concertos que estragam.....	24
Figura 5: Representação da estrutura do arquétipo processo de equilíbrio com defasagem.	25
Figura 6: Estrutura do arquétipo sucesso para os bem sucedidos.	26
Figura 7: Linha de tempo da Sensedia.	27
Figura 8: Plataformas de API'S Sensedia. Desenho de solução.	28
Figura 9: Clientes da Sensedia	29
Figura 10: Mapa resultante do fim do primeiro dia da aplicação do <i>design Sprint</i>	34
Figura 11: <i>storyboard's</i> produzidos no segundo dia da aplicação do design sprint.....	35
Figura 12: o mural de <i>storyborads</i> durante o processo de votação.	36
Figura 13: Exemplo de prototipação.	36
Figura 14: Ambiente onde acontecem os testes.	37
Figura 15: Diagrama de causalidade concertos que estragam.....	39
Figura 16: diagrama de causalidade concertos que estragam.....	40
Figura 17: estrutura concertos que estragam.	41
Figura 18: Diagrama de causalidade processo de equilíbrio com defasagem.	43
Figura 19: Estrutura do arquétipo processo de equilíbrio com defasagem.	43
Figura 20: Diagrama de causalidade do arquétipo sucesso para os bens sucedidos.....	45
Figura 21: Diagrama de causalidade do arquétipo sucesso para os bens sucedidos.....	45
Figura 22: Diagrama de causalidade do arquétipo sucesso para os bens sucedidos.....	46
Figura 23: Diagrama de causalidade do arquétipo sucesso para os bem sucedidos.	47
Figura 24: Estrutura do arquétipo sucesso para os bem sucedidos.	47

LISTA DE TABELAS

Tabela 1. Quadro comparativo das modelagens <i>hard e soft</i>	22
Tabela 2. Cronograma de atividades.....	30

LISTA DE SIGLAS

API - *Application programming interface*

APIX - *Application programming interface experience*

EUA – Estados Unidos da América

MIT - *Massachusetts institute of technology*

MVP - *Minimum viable product*

SOA - *Service oriented architecture*

SP - São Paulo

SUMÁRIO

INTRODUÇÃO	12
1. REFERENCIAL TEÓRICO	14
1.1 Gerência de requisitos	14
1.2 Mvp	15
1.3 Design sprint	16
1.3.1 Dinâmica de sistemas	17
1.3.2 Conceitos e elementos da dinâmica de sistemas	19
1.3.3 Modelagens da dinâmica de sistemas	21
1.4.1 Arquétipos sistêmicos	23
1.4.2 Concertos que estragam	24
1.4.3 Processo de equilíbrio com defasagem	25
1.4.4 Sucesso para os bem sucedidos	26
1.5 Sensedia.....	27
1.5.1 Foco de mercado	29
2. Metodologia	30
3. Resultados	33
3.1 Problemática analisada.....	37
3.2 Modelagem concertos que estragam	38
3.3 Modelagem processo de equilíbrio com defasagem.....	42
3.4 Modelagem sucesso para os bem sucedidos.....	44
3.5 Análise das modelagens	48
3.6 Análise concertos que estragam	48
3.7 Análise processo de equilíbrio com defasagem	49
3.8 Análise sucesso para os bem sucedidos	49
3.9 Discussão dos resultados	51
4. Conclusão	52
5. Trabalhos futuros	54
REFERÊNCIAS.....	55
ANEXO 1 – AUTORIZAÇÃO PARA REDAÇÃO DE ESTUDO DE CASO	60

INTRODUÇÃO

Nos dias atuais, as organizações de desenvolvimento de software ainda enfrentam problemas na comunicação entre cliente e empresa, pouca sinergia ou a entropia entre essas partes podem fazer surgir uma série de problemas colocando em risco a integridade do projeto, tais problemas podem ocorrer a qualquer momento e podem ser originados por falhas do próprio profissional de software.

Os profissionais de softwares são seres humanos e, portanto, algumas vezes, cometem erros ao desenvolver o produto. Conseqüentemente existirá uma falha no software. Se esse erro for cometido na fase de levantamento de necessidades, a falha resultante provavelmente também irá aparecer nas especificações, no projeto e no código (Schach 2010, p.13).

Outros fatores que podem corromper o sucesso do projeto são estimativas de prazo de entrega e custos irreais, falhas nas entregas das funcionalidades. Em seu livro, Bartié (2002), cita estudos realizados nos EUA, onde estes apontam que mais de 30% dos projetos são cancelados antes de serem finalizados, os prazos extrapolam mais de 200% os cronogramas originais, os custos excedem em mais de 180% os valores originalmente previstos e mais de 70% dos projetos contêm falhas nas entregas das funcionalidades esperadas.

Pensando no cenário descrito por Bartié, um estudo foi feito com o objetivo de compreender melhor a aplicação do *design sprint* na Empresa, tentando identificar situações que possam ser modeladas nos conceitos da dinâmica de sistemas com o intuito de criar modelos que poderão gerar conhecimentos em torno da situação abordada.

Na atualidade a plataforma de API's é o principal produto da Sensedia, como é um produto inovador, demanda muita evolução, o *design sprint* surge como um método mais avançado para a construção e evolução de produtos.

A partir do entendimento da aplicação da metodologia, uma situação proveniente do *design sprint* de acordo com alguns profissionais que atuam na empresa foi abordada conforme os parâmetros da dinâmica de sistemas. A dinâmica de sistemas nesse contexto surge como uma ferramenta útil para entender o comportamento de sistemas que sofrem alterações ao longo do tempo, no princípio a dinâmica de sistemas era usada apenas para entender as falhas nos processos administrativos, mas com o passar dos anos foi se disseminando a outras, onde no presente trabalho a dinâmica de sistemas se relaciona com a engenharia de requisitos.

Com o objetivo de constatar se a dinâmica de sistemas pode colaborar no ambiente de uma empresa de desenvolvimento de *software*, a presente pesquisa integra o emprego de arquétipos sistemicos para melhorar o entendimento dos processos de desenvolvimento.

1. REFERENCIAL TEÓRICO

Nos tópicos a seguir, serão apresentados todos os principais conceitos utilizados na elaboração do presente trabalho.

1.1 Gerência de requisitos

Basicamente o gerenciamento de requisitos pode ser um conjunto de atividades que auxilia a equipe a identificar e controlar os requisitos. De acordo com HIRAMA (2012) em um projeto de desenvolvimento de *software* é esperado que o mesmo atenda as necessidades do cliente com restrições de custo e prazo, para isso é necessário que haja um gerenciamento dessas informações desde a especificação de requisitos até a entrega. Ainda segundo ele, o gerenciamento de requisitos trata não somente dessa entrega mais como também possíveis mudanças que podem ocorrer nos requisitos.

Para MARTINS (2007) O projeto precisa ser elaborado de modo a incorporar facilmente as mudanças, identificando os requisitos que tem maior influência no custo e nos aspectos técnicos, segundo ele o desafio do gerenciamento de requisitos está no fato de que os requisitos são dinâmicos e mudam por vários motivos, tais como:

- Usuário muda de ideia;
- O problema muda;
- Mudanças técnicas;
- Mudanças de mercado;

Segundo REZENDE (2005) os requisitos são controlados através de uma gestão de forma a documentar e estabelecer um perfil mínimo a ser utilizado pela engenharia de *software* e pela administração. Para AUDY e PRIKLADNICKI (2007) no gerenciamento de requisitos é conduzida a manutenção dos artefatos de requisito, assegurando que estão constantemente alinhados com os objetivos de negócios e atualizados de acordo com as modificações do ambiente. Ainda de acordo com AUDY e PRIKLADNICKI (2007) existem dos tipos de artefatos de requisitos, os artefatos de entrada e os artefatos de saída.

Audy e Prikładnicki (2007) afirmam que os artefatos de entrada são especificações dos requisitos do *software* inspecionado, a solicitação de alteração nos requisitos,

1.2 Mvp

O MVP (Produto mínimo viável) é basicamente uma pequena parte do produto, uma versão fragmentada com menor investimento, segundo Massari (2016) o MVP corresponde a um conjunto mínimo de funcionalidades de um produto. De acordo com Ries (2011) O MVP é apenas o primeiro passo numa jornada de aprendizagem onde, após diversas alterações você pode ter algum elemento do seu produto ou estratégia, e decidir que o momento de fazer uma mudança para alcançar um modo distinto para atingir sua visão.

Para Meira (2013) MVP é um teste de viabilidade, que possibilita maior aprendizado possível junto a clientes reais, deste modo reduzindo o tempo que a ideia vai levar para ser testada no mercado pela primeira vez e ao mesmo tempo serve como uma forma de prevenir o aumento da complexidade, tempo e custo. Soler (2015) cita que com o domínio total ou parcial do problema a ser desenvolvido a equipe deve conduzir o cliente ao MVP para que seja possível realizar ciclos de iteração, desta maneira o cliente será forçado a ter uma alta participação elevada no projeto.

Segundo Telles e Matos (2013) o conceito de MVP foca no produto onde uma série de funcionalidades é distribuída mantendo apenas aquelas que tornam o produto mais atraente para a parcela de formadores de opinião para o público alvo. É exatamente esse público que fará os primeiros testes com o produto ou serviço, e é neles que o empreendedor deve manter o foco nessa fase inicial na qual o desenvolvimento de adequações deve estar extremamente alerta para satisfazer esse público tão exigente.

Em outras palavras o MVP é uma versão mínima do produto contendo apenas o necessário para cumprir o objetivo planejado, contudo o processo de MVP demanda certo tempo para o desenvolvimento com o objetivo de entregar essas funcionalidades para que sejam validadas.

1.3 Design sprint

Criada pela *Google ventures* seu objetivo principal é acelerar e validar ideias. Basicamente esse processo que tem duração de cinco dias onde um grupo de pessoas se reúne, e ao final desse período é esperado pronunciar-se sobre uma funcionalidade através de respostas de questões críticas de negócios através de design, prototipagem e teste das ideias com usuários. Em cada um desses cinco dias ocorre uma tarefa específica. Em cada dia de trabalho no *design sprint* ocorre uma tarefa diferente, para Banfield, Lombardo e Wax (2016) cada dia consiste em:

- Na segunda-feira temos a tarefa que é denominada *unpack*, basicamente nesse dia o objetivo é desempacotar ideias, onde todos os funcionários dos departamentos envolvidos no projeto expõe suas ideias e conhecimento de maneira individual.
- Na terça-feira o objetivo é denominado como *sketch*, ou seja, um esboço. Nesta etapa serão propostas de possíveis soluções de maneira individual para atingir a meta.
- Na quarta-feira acontece a etapa de *decide*, essa etapa consiste em decidir a(s) melhor(e)s solução(e)s e refina-la(s).
- A quinta-feira é designada para *prototype*, basicamente nesse dia acontece a prototipação da(s) ideia(s).
- Na sexta-feira ocorre a fase de *test*, nessa fase acontece os testes junto ao usuário.

Segundo um artigo publicado pela CIO Magazine em uma entrevista com Jeancarlo Cerasoli UX *Expert* da C&T, para conceber o *design Sprint* a *google ventures* procurou adicionar maior objetividade ao processo para atender às necessidades de foco, eficiência e agilidade das *startups* para as quais presta serviços. Ainda segundo Jeancarlo o *design sprint* se trata de uma metodologia que enfatiza justamente a construção colaborativa de soluções através de ciclos de convergência e divergência com a prototipação rápida, sendo uma ferramenta poderosa para construir consensos, testar hipóteses, alinhar expectativas e direcionar projetos que estejam em dúvidas em um curto espaço de tempo.

De acordo com Jeancarlo, é notório que realizar em uma semana o que levaria um mês para construir significa abrir mão de certos aspectos, como por exemplo um entendimento

mais profundo de um contexto de uso ou até mesmo uma conceituação menos detalhada, afinal de contas o *design sprint* é um teste rápido de conceito, por outro lado essa metodologia pode ser um instrumento para quebrar a inércia de grandes corporações que não estão acostumadas com metodologias rápidas, centradas nas necessidades das pessoas. Para Jeancarlo após um trabalho prévio de preparação levantando as expectativas do negocio, as possibilidades tecnológicas e as necessidades dos consumidores é possível construir um poderoso consenso que leve a inovação num período de tempo surpreendentemente pequeno para a empresa desta natureza.

1.3.1 Dinâmica de sistemas

A dinâmica de sistemas foi construída a partir dos estudos do professor Jay Wright Forrester do MIT (Instituto tecnológico de Massachusetts) durante o século vinte, e apresentada inicialmente em 1961 com a publicação de seu livro *Industrial Dynamics*, nos quais Forrester aplicou os conceitos da dinâmica de sistemas com o intuito de averiguar as razões dos fracassos das corporações que faziam uso de sistemas de realimentação em processos administrativos.

De acordo com Sales Franco e Penaforte (2009) alguns anos mais tarde, mais precisamente em 1968 J.W. Forrester usou os conceitos da dinâmica de sistemas nas problemáticas sociais através de contatos com o professor James Collins que visitou o MIT para tratar de questões urbanas, originando em 1969 o livro *Urban Dynamics*, em 1973 Forrester lança mais um livro intitulado *World dynamics* em que basicamente ele apresenta modelos de simulação que mostram como o crescimento exponencial da população relacionado com os recursos naturais conduzem os problemas de poluição e fome. A partir de então, a utilização dos conceitos em torno da dinâmica de sistemas tem se desenvolvido de forma considerável oferecendo estrutura para análise de problemas em diferentes áreas tais como: Ecologia, gestão empresarial, economia e educação. Segundo Chwif e Medina (2014) a dinâmica de sistemas foi originalmente desenvolvida para trabalhar com problemas industriais, porém diante de sua natureza geral é possível dilata-la em outras classes de sistemas. A dinâmica de sistemas surge como uma metodologia para entender e modelar algumas classes de sistemas complexos que utilizam os mecanismos de *feedback*.

Já na década de 70, de acordo com Vilela (2005) Peter Senge trabalhou na realização de alguns seminários com executivos incorporando os conceitos da dinâmica de sistemas nas grandes organizações. Nos dias atuais, a obra de P. Senge está consolidada como uma metodologia de administração de empresas que utiliza basicamente o ferramental da dinâmica de sistemas e é conhecida como organizações que aprendem (*Learning Organizations*) e pensamentos sistêmicos (*System Thinking*). Durante os anos 80, segundo Guerrini, Escrivão Filho, Cazarini e Pádua (2014) a proposta de Forrester ficou estagnada, mais logo nos anos 90 com o avanço da informática, a dinâmica de sistemas ganhou ênfase novamente. Em 1990 Peter Senge lançou seu *Best seller* intitulado “a quinta disciplina” que faz uso de praticamente todas as ferramentas metodológicas desenvolvidas por Jay Forrester na década de 50 e estruturado no início da década de 60 no livro *Industrial Dynamics*. Ainda segundo Vilela, a aplicação da dinâmica de sistemas vem alcançando um sucesso inquestionável em todo o mundo, desde os modelos globais e urbanos de Forrester e Collins aos simuladores de vãos gerencias que vem sendo usados nas grandes corporações ao redor do mundo, a disciplina de dinâmica de sistemas vem provando seu potencial como ferramenta auxiliar em varias áreas do conhecimento.

Senge (1990) se refere ao pensamento sistêmico como uma disciplina para ver o todo, é um conjunto de disciplinas abrangendo diversos campos, Ainda seguindo as ideias de Senge (1990) o pensamento sistêmico é cada vez mais necessário, pois nos deparamos com situações complexas, isso porque, a humanidade está produzindo mais informações do que o homem pode absorver, gerando uma interdependência maior do que as pessoas podem administrar e de acelerar mudanças com uma velocidade muito maior do que o ser humano pode acompanhar.

Certamente a escala de complexidade é sem precedentes. Tudo a nossa volta é exemplo de “colapsos sistêmicos” problemas como o aquecimento global, a diminuição da camada de ozônio, o tráfico internacional de drogas e o déficit comercial e orçamentário norte americano, problemas que não possuem simples causa local. Da mesma forma as organizações também entram em colapso, apesar da inteligência individual e dos produtos inovadores, pois elas são incapazes de reunir suas diversas funções e talentos para criar um todo produtivo (Senge 1990, p.99).

Deste modo a dinâmica de sistemas objetiva investigar situações através da elaboração de modelos que irão refletir as questões analisadas através do pensamento sistêmico, onde a complexidade da dinâmica aparece quando uma ação provoca efeitos drasticamente a pequenos e grandes prazos.

1.3.2 Conceitos e elementos da dinâmica de sistemas

De acordo com Chwif e Medina (2014) a dinâmica de sistemas trabalha com relações de causa e efeito, acumulação de fluxos e atrasos de tempo, onde uma situação pode ser representada por um diagrama de laços causais. De acordo com Sterman (2000) o papel de tais diagramas na modelagem dos sistemas é proporcionar um esboço das possibilidades causais e permitir que os pressupostos estruturais sejam explicitamente comunicados, segundo ele a dinâmica de sistemas é uma ferramenta ideal para a modelagem de sistemas complexos.

De acordo com Valença (1999) para que esses diagramas sejam construídos é necessário o entendimento dos seguintes conceitos.

- Variáveis do sistema: são as entidades do sistema, e podem sofrer mudanças ao longo do tempo.
- Relações: são setas que indicam o relacionamento e a direção da variável. Graficamente a letra “m” ou “+” (mesmo sentido) indica que a variação no fator causador, gerando uma variação no mesmo sentido ao fator que recebe o efeito, em contrapartida os sinais de “o” e “-” (sentido oposto) indicam que a variação no fator causador gera uma variação no sentido contrário no fator que recebe o efeito.

Em resumo segundo Cazarini, Escrivão Filho, Guerrini e Pádua (2014) a letra “m” ou o sinal “+” contidos na seta indicam que os dois conceitos se deslocam no mesmo sentido e a letra “o” ou o sinal de “-” contidos na seta indicam basicamente que os dois conceitos se deslocam em sentidos opostos.

Na figura 1 a seguir pode-se observar relações de mesmo sentido entre a variável “descoberta de erros” e “retrabalho”, isso por que quanto mais erros descobertos no projeto, mais retrabalho devera ser realizado. Em contrapartida, encontramos uma relação de sentido oposto entre as variáveis “retrabalho” e “rendimento e produção” devido ao fato de quanto maior o retrabalho menor será o rendimento e produção no projeto.



Figura 1: Exemplo de relações de mesmo sentido e sentido oposto.
Fonte: Próprio autor.

- *Delay*: Basicamente são efeitos que serão percebidos apenas com um tempo de espera, são representados por duas barras cortando uma seta. Na figura 2 localizada abaixo pode - se encontrar um *delay*, onde a má alimentação pode refletir em graves problemas de saúde com o passar do tempo.

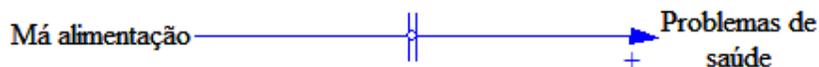


Figura 2: Exemplo relação com *delay*.
Fonte: Próprio autor.

- Diagramas de causalidade: De acordo com Marujo (2007), as relações de causalidade relacionam duas variáveis de cada vez desconsiderando a influência de outras variáveis entre elas, e uma realimentação pode conter uma ou mais relações de causalidade. Veja a seguir na figura 3 um exemplo diagrama de causalidade com mais de uma relação entre as variáveis:

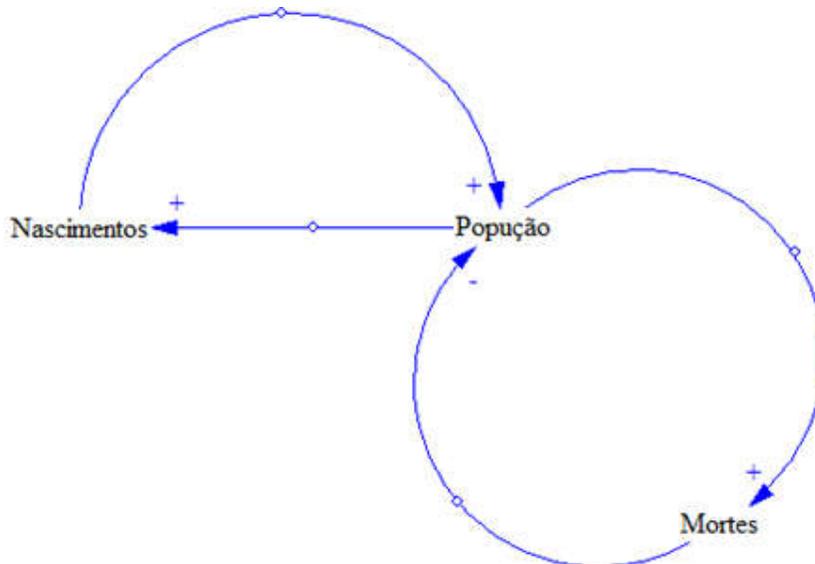


Figura 3: Exemplo de diagrama de causalidade.
Fonte: Próprio autor.

Na figura 3 pode - se observar um diagrama de causalidade que representa o tamanho da população, onde quanto maior o número de nascimento de pessoas maior também será o tamanho da população, assim pode - se notar uma relação de mesmo sentido.

É possível também identificar uma relação de mesmo sentido entre população e mortes onde quanto maior a população maior será a quantidade de mortes e como consequência menor a população assim resultando numa relação de sentido oposto.

Para Guerrini, Escrivão Filho, Cazarini e Pádua(2014) o diagrama de enlace causal é usado para entender e compreender a estrutura geral do sistema, Segundo Chwif e Medina (2014) um sistema pode ser representado pelo seu diagrama de laços causais

1.3.3 Modelagens da dinâmica de sistemas

Segundo o conceito de Radzicki (1997) um modelo pode ser visto como uma representação de parte da realidade que é percebida pela pessoa que deseja entender e gerenciar a questão percebida. Felício (2010) afirma que é praticamente impossível realizar a descrição de todos os aspectos de determinado processo ou situação do mundo real, devido a isso é preciso decidir quais características irão ser consideradas e quais serão desconsideradas. Neste contexto Felício (2010, p.2) cita que “A arte de modelar é saber selecionar somente as características, dentre muitas disponíveis, que são necessárias e suficientes para descrever o processo com precisão satisfatória”.

A dinâmica de sistemas possui alguns conceitos e elementos que basicamente são primordiais e necessários à modelagem de sistemas complexos. Segundo referem - se Oliveira e Ferreira (2014) a modelação da dinâmica de sistemas possibilita entender as variáveis de um sistema ao longo do tempo e também nos permite analisar de maneira eficiente os estímulos dos impactos que ainda se pretende estudar.

Segundo Vilela (2005) A disciplina da dinâmica de sistemas é antes de tudo uma nova linguagem que permite expressar mais adequadamente as cadeias de eventos circulares (*loops*) existentes na natureza. Através de vários tipos de diagramas (causais; estoque e fluxo) é possível expressar graficamente um sistema (um pedaço delimitado da natureza) possibilitando ver mais claramente, a complexidade dinâmica da natureza, isto é, o processo envolvido desta ao longo do tempo.

Para Chaim (2011) um modelo pode ser visto como uma representação da realidade. Em outras palavras, um modelo é basicamente uma representação externa de um sistema onde seu objeto é entender como o mesmo trabalha. Na modelagem da dinâmica de sistemas existem duas abordagens que são empregadas para a construção de um modelo. A *soft* ou mental e a *hard* ou formal. Que de acordo com Chaim (2011) se resumem em:

- *Hard* ou formal: Parte do princípio que os modelos são uma representação do mundo real e que os resultados da simulação podem ser confrontados diretamente com a realidade.
- *Soft* ou mental: Defende a ideia de que os modelos serão construídos para permitir às pessoas pensarem por meio de suas próprias posições e engajarem-se em debates com outras pessoas para determinar o curso das possíveis ações.

Veja na tabela 1 as características das duas modelagens:

	Abordagens hard	Abordagens soft
Definição do problema	Vista como direta, unitária.	Vista como problemática, pluralística.
Objetivo	Orientado à busca de metas. Supõe que a realidade possui sistemas que podem ser planejados	Orientado ao aprendizado Supõe que a realidade é complexa, mas pode ser explorada por modelos de sistemas.
A organização	Assumida tacitamente.	Requer negociação.
O modelo	Uma representação do mundo real (ontologias). Fala a linguagem de problemas e soluções.	Uma forma que gera debate e insight a respeito do mundo real. Podem ser construções intelectuais (epistemologias). Fala a linguagem de questões e acomodações.
Resultado	Um produto ou recomendação.	Progresso através da aprendizagem.
Vantagens	Permite o uso de poderosas técnicas. Mais fáceis de serem validadas.	É utilizável tanto por profissionais especializados como pelos detentores do problema; busca considerar o conteúdo humano das situações problemáticas.
Desvantagens	Necessita de profissionais especializados em técnicas específicas; ignora participação dos detentores do problema. Deixa de considerar aspectos que transcendem a simples lógica da situação.	Não produz respostas definitivas. Aceita a ideia de que o processo de questionamento é infundável. Problemas para a validação do modelo.

Tabela 1: quadro comparativo das modelagens *hard* e *soft*
Fonte: Rossoni (2006).

De acordo com Rossoni (2006) problemas bem definidos podem ser desenvolvidos por técnicas *hard* enquanto um contexto mais amplo deve ser questionado seguindo a modelagem *soft*. Em ambos os casos deve ser observado a natureza do problema em questão.

De acordo com Chaim (2011) em contraste com a modelagem *hard* onde o trabalho tem início uma vez que é definida uma necessidade e que é necessário supri-la, a metodologia *soft* parte do princípio de que as percepções do mundo das pessoas terão variações e de que as preferências poderão também diferir. Em resumo a modelagem *soft* parte de um princípio onde modelo é criado como uma forma de gerar debate a respeito do mundo real, realizando construções intelectuais. Já a modelagem *hard* parte do ponto de vista onde a construção do modelo como uma representação do mundo real com o objetivo de encontrar soluções e otimizações para a problemática modelada.

Em outras palavras o que distingue basicamente as duas técnicas é que em quanto à *hard* busca soluções e otimizações através da observação do mundo real, a *soft* apenas busca criar entendimento através da questão abordada.

1.4.1 Arquétipos sistêmicos

Segundo Ocaña (2006) a palavra arquétipo vem do grego *arkhetypos* que significa o primeiro de sua espécie. Ainda seguindo sua ideia, Ocaña cita que os arquétipos são ferramentas acessíveis que permitem construir hipóteses e conhecimentos sobre as forças que operam os sistemas, eles constituem um veículo natural para verificar modelos mentais sobre estes sistemas. De acordo com Boyett e Boyett (1998) um arquétipo é por definição, um modelo a partir do qual se faz todas as outras coisas do mesmo tipo. A equipe de Senge documentou uma série destes arquétipos para serem usados como ferramentas de diagnóstico e resolução de problemas. Ainda de acordo com Boyett e Boyett (1998) os arquétipos são ferramentas que ajudam a pensar da qualidade interconectadora do mundo. Para Sales Franco e Penaforte (2009), os arquétipos são padrões que estão associados a um comportamento que exhibe as características de um sistema.

Para Ocaña (2006) algumas empresas ou pessoas ao usufruírem do conceito da dinâmica de sistemas para entenderem melhor alguns processos, acabam por selecionar os arquétipos de maneira errônea. Segundo ele, os arquétipos devem ser escolhidos a partir de

um estudo prévio da situação, onde são elaborados os diagramas de causalidade, e apenas depois escolher o arquétipo que melhor se enquadra no estudo.

Ao longo de 50 anos de estudos, foram identificados 11 arquétipos de sistema, dentre os quais destacam-se três deles que serão abordados no presente estudo, tais arquétipos foram identificados de modo intuitivo a partir dos diagramas de causalidade presentes na metodologia deste trabalho, e também pelo princípio gerencial de cada um deles descrito por Senge (1990). Esses arquétipos e seus princípios serão apresentados nos tópicos a seguir.

1.4.2 Consertos que estragam

Este arquétipo vai de encontro onde uma decisão é tomada para uma para uma resposta em curto prazo, mas que, ao longo do tempo podem render atrasos no sistema. De acordo com Senge (1990) o princípio gerencial desse arquétipo se caracteriza a partir de uma solução em curto prazo, que tem consequências imprevistas ao longo do tempo, que talvez acarrete na utilização ainda maior do mesmo recurso. Veja o esboço do arquétipo na figura 4:

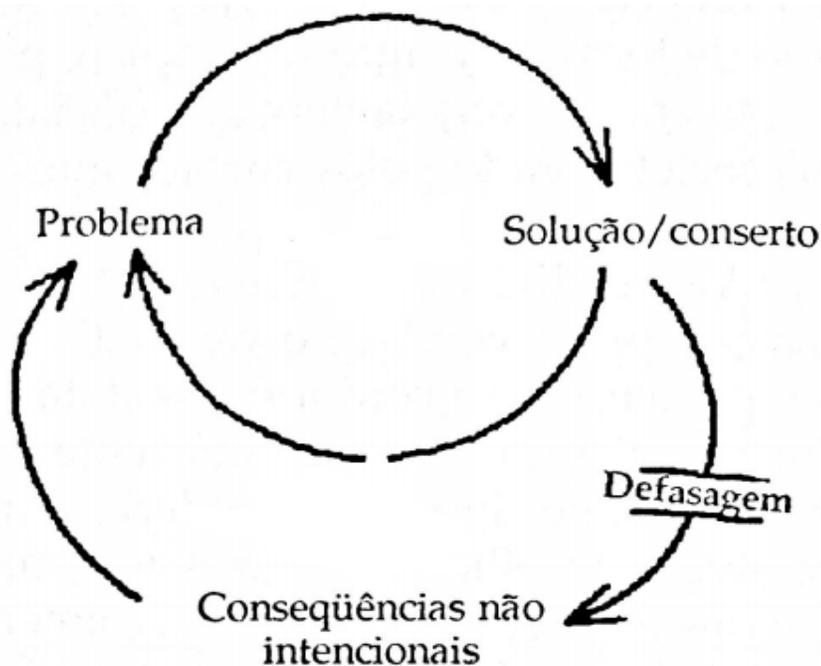


Figura 4: Representação da estrutura do arquétipo concertos que estragam
Fonte: Senge (1990).

Em outras palavras, este arquétipo representa um cenário onde uma decisão em curto prazo é tomada para resolver um problema, onde, ao longo do tempo podem surgir consequências não intencionais que irão piorar o desempenho ao a situação que estão tentando corrigir.

1.4.3 Processo de equilíbrio com defasagem

O arquétipo processo de equilíbrio com defasagem, onde uma decisão é tomada para cumprir um objetivo, de acordo com um *feedback* com atraso.

De acordo com Senge (1990) o princípio gerencial deste arquétipo se baseia em uma pessoa, grupo ou organização agindo para concretização de uma meta, ajustando seu comportamento a um *feedback* com defasagem. Ainda de acordo com as ideias do eminente autor, caso não estiverem conscientes de defasagem acabam realizando mais ações corretivas que o necessário ou simplesmente desistem, pois não enxergam progresso algum sendo realizado. Na figura 5 que se apresenta a seguir pode - se ver a estrutura desse arquétipo, vejamos:

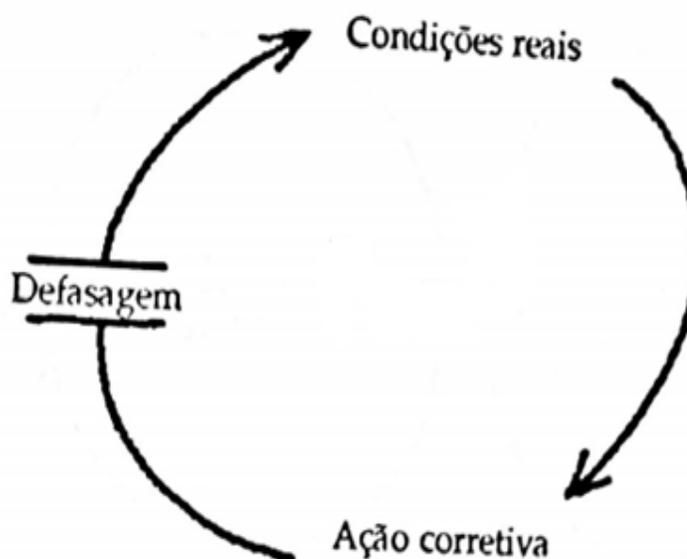


Figura 5: Representação da estrutura do arquétipo processo de equilíbrio com defasagem. Fonte: Senge (1990).

Em outras palavras, esse arquétipo representa uma ação tomada no sentido de alcançar alguma meta gera um *feedback* com atraso, onde, a falta de discernimento desse atraso pode levar a realizações de ações desnecessárias e o tempo de resposta influencia no tempo de oscilação do sistema.

1.4.4 Sucesso para os bem sucedidos

De acordo com Senge (1990) o princípio gerencial deste arquétipo se resume em duas atividades que competem por apoio ou recursos limitados, ao passo que, quanto mais apoio uma delas ganha, mais bem sucedida se torna, de maneira que, a outra entra em declínio. A variável que recebe mais recursos ou apoio, automaticamente tem vantagens sobre a outra de alcançar o sucesso e como consequência receber mais recursos ainda, em contrapartida a variável oposta passa a escassez dos mesmos recursos.

Na figura 6 pode - se visualizar a estrutura desse arquétipo:

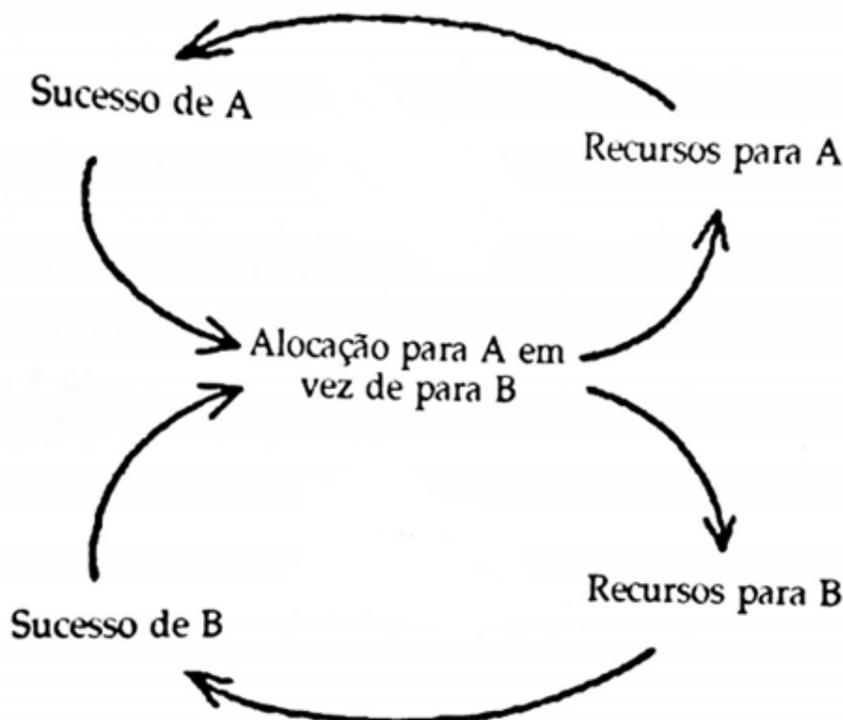


Figura 6: Estrutura do arquétipo sucesso para os bem sucedidos.
Fonte: Senge (1990).

Este arquétipo se associa bem a um cenário onde existem duas variáveis, ambas competindo pelo mesmo recurso seja ele, intelectual, ou pela mesma dedicação de indivíduos onde apenas uma delas recebe os recursos.

Com o intuito de descobrir se a dinâmica de sistemas pode favorecer na gestão de requisitos e gerenciamento de desenvolvimento de *software* o presente estudo engloba o emprego de arquétipos sistêmicos para melhorar o entendimento das metodologias de processos de *software*.

1.5 Sensedia

Para a aplicação do estudo proposto o ambiente de desenvolvimento escolhido foi a Sensedia. De acordo com Marcilio Oliveira diretor executivo da empresa, a Sensedia é uma empresa nacional, sediada em Campinas/SP, com operação em diversos estados. Especializada em soluções para integração de aplicações, a Sensedia tem como seu principal produto e solução, uma plataforma de *API Management*, chamada de “Sensedia API Suite”. A seguir na figura 7, alguns pontos históricos da empresa:

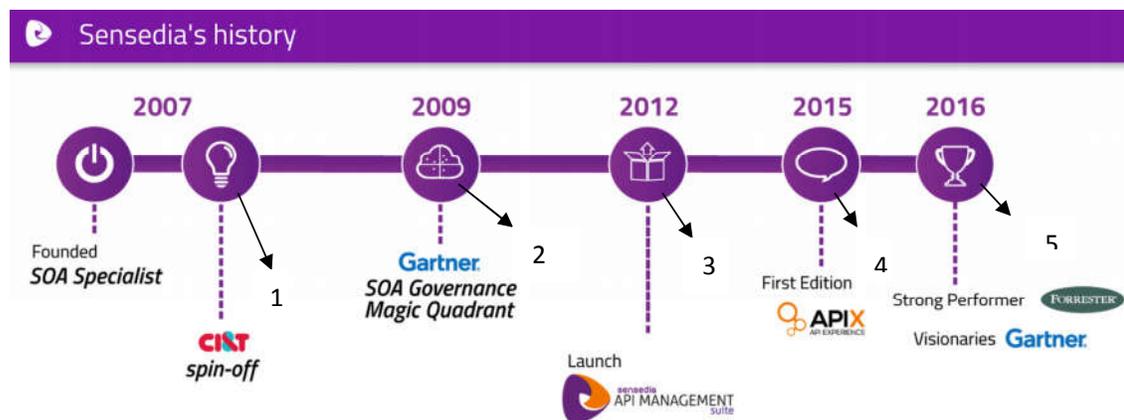


Figura 7: Linha de tempo da Sensedia.
Fonte: Sensedia.

1. Fundada em 2007, como uma *spin-off* da CI&T com a Unicamp, a Sensedia teve seu início como empresa especializada em SOA (*Service Oriented Architecture*).

2. Em 2009, foi reconhecida pelo *Gartner group* como visionário através do *Gartner Magic Quadrant*. A Sensedia é a única empresa brasileira com esta classificação pelo *Gartner group*, que é uma empresa de consultoria sediada em *Stanford*.
3. De acordo com a evolução do mercado de tecnologia e a necessidade de transformação digital dos negócios nas grandes empresas, em 2012 a Sensedia evoluiu sua oferta criando a plataforma de *API Management*.
4. O mercado de APIs vem chamado muito a atenção de todos os grandes consumidores de software, e a Sensedia lançou, em 2015 o evento APIX (*API Experience*), que é o maior evento nacional focado em solução de APIs, e o segundo maior evento do mundo neste setor.
5. Recentemente, em 2016, a Sensedia foi novamente reconhecida pelos dois maiores grupos de referência tecnológica do mundo: *Gartner*, através do *Gartner Magic Quadrant* para soluções de APIs, e do *Forrester*, através do *Forrester Wave*¹ de soluções de APIs.

A figura 8 apresenta a plataforma de API's da empresa:

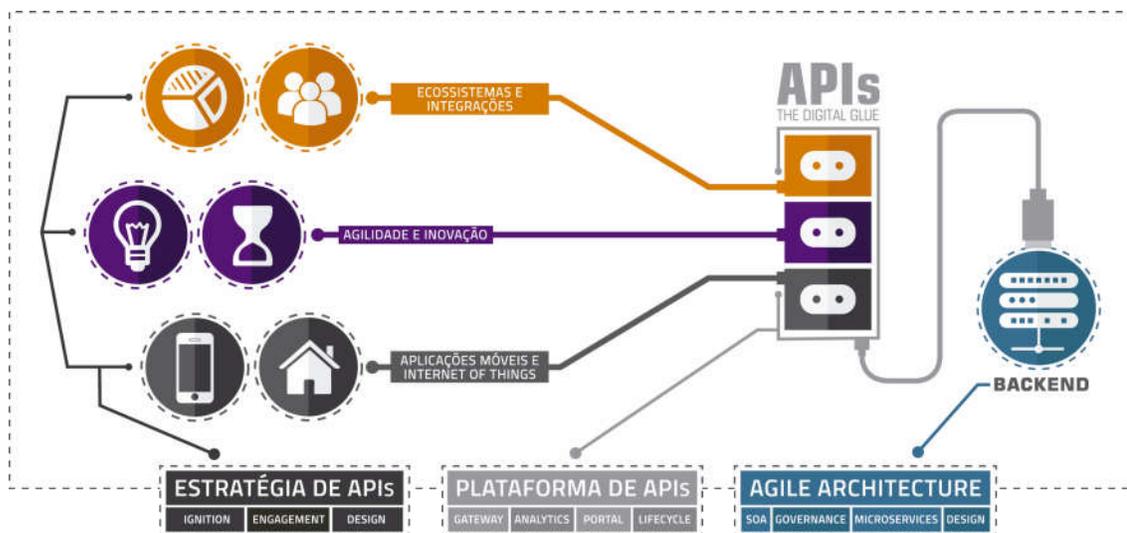


Figura 8: Plataformas de API'S Sensedia. Desenho de solução.

Fonte: Sensedia

Para Marcilio oliveira, a plataforma de APIs da Sensedia, seu principal produto, se posiciona como o *building-block* central para a plataforma digital de empresas: gerenciamento de APIs. Além disso, a empresa tem serviços de consultoria ajudando no desenho da sua

arquitetura de TI através de especialistas em *Microservices*, SOA, APIs, *Enterprise Architecture* e Governança SOA.

1.5.1 Foco de mercado

Soluções complexas de arquitetura e, sobretudo, o desafio de transformação digital atinge os grandes mercados. A base de clientes e mercado alvo da Sensedia atualmente contém grandes empresas, com clientes nos setores de:

- Financeiro (Bancos, seguradoras e meios de pagamentos).
- *E-commerce* (lojas online, comparadores de preços e *marketplaces*).
- Serviços (Educação, tecnologia, aviação).

A figura 9 mostra alguns dos clientes da Sensedia.



Figura 9: Clientes da Sensedia
Fonte: Sensedia

Por conter este ambiente de desenvolvimento complexo esta empresa se apresentou como objeto suficiente para a aplicação da presente pesquisa. No próximo capítulo dessa pesquisa, será apresentada a metodologia.

2. Metodologia

O objetivo desta pesquisa foi analisar se a dinâmica de sistemas pode ajudar nas práticas dos profissionais de uma empresa de desenvolvimento de *software*. Para isso foram realizadas diversas pesquisas com os colaboradores da Sensedia, o cronograma de atividades segue na tabela 2:

JANEIRO 2016	JULHO 2016	OUTUBRO 2016
Entender a engenharia de requisitos.	Entender o desenvolvimento de <i>software</i> .	Levantamento dos pressupostos.
Entender o desenvolvimento de <i>software</i> .	Levantamento bibliográfico (MVP, <i>design sprint</i> , arquétipos de sistemas).	Estudos dos arquétipos apropriados para cada pressuposto.
Conhecer o ambiente de trabalho da Sensedia.	Entender como a Sensedia aplica o <i>design sprint</i> .	Aplicação da modelagem mental nos arquétipos.
Reunião com colaboradores.	Reunião com os colaboradores.	Teste dos arquétipos modelados com os colaboradores.

Tabela 2: Cronograma de atividades.

Fonte: Próprio Autor

A dinâmica de sistemas foi escolhida para auxiliar o presente trabalho, pois ela permite a construção de gráficos de relações causais nos quais pode se delinear as relações de causa e efeito existentes entre os elementos de um sistema, nos quais seus modelos ajudam a estudar os tempos de respostas diante de uma decisão inerente a qualquer sistema.

Um modelo é basicamente uma representação externa de um sistema, nos quais seu objetivo é entender como o mesmo trabalha. Na modelagem da dinâmica de sistemas existem duas abordagens que são empregadas para a construção de um modelo: A *soft* ou mental e a *hard* ou formal. As principais diferenças entre os dois conceitos de modelagem é que na primeira tem-se uma percepção de uma problemática na qual o modelo irá proporcionar uma forma de gerar debates a respeito do mundo real, onde o objetivo é gerar aprendizagem em torno da problemática abordada. Já no segundo modelo, se supõe que o problema pode ser definido gerando uma representação do mundo real, e tem como objetivo gerar soluções ou otimizações. Na presente pesquisa será abordada a modelagem *soft* ou mental.

Para desenvolver uma modelagem *soft* de acordo com Senge (2001), é preciso seguir alguns passos:

- Definir uma situação de interesse da organização.

- Identificar os eventos relacionados com a situação.
- Reconhecer as variáveis chaves para compreensão do sistema.
- Realizar a observação dos padrões de comportamento e identificar as relações causais.
- Elaborar os modelos mentais, identificando os arquétipos que reflitam a realidade do sistema envolvido.

Com intuito de fortalecer a presente pesquisa, as informações descritas diante de algumas dessas etapas descritas por Senge (1990) foram confrontadas com referências bibliográficas. A seguir, foram descritas as definições de cada passo para construção do modelo *soft*:

Em um primeiro momento é definida uma situação de interesse da organização, ou seja, junto a colaboradores foi discutida e definida uma situação relevante de interesse da organização para ser estudada.

Depois foi identificado os eventos relacionados com a situação, para isso, foram confrontadas referências bibliográficas junto com entrevistas aos colaboradores, uma vez que o presente estudo analisa a situação real da empresa como também alguns pressupostos levantados por colaboradores.

O terceiro passo consiste em identificar as variáveis chaves para compreensão do sistema, para isso usaram-se todas as informações que foram discutidas através de colaboradores e referências bibliográficas, tais como: Mendes, Pressman, Amaral, Horstmann, Audy e Prikladnicki.

No quarto passo foram observados os padrões de comportamento para identificar as relações causais, ou seja, basicamente analisar o que foi estudado para criar os diagramas de causalidade identificando as relações entre uma variável e outra.

O último passo consiste na elaboração dos modelos mentais, modelando todas as informações, relações e diagramas de causalidade na estrutura dos arquétipos que reflitam a realidade do sistema analisado.

A seguir os resultados da aplicação da pesquisa, que conta com entrevistas com alguns colaboradores da empresa, são eles:

- Marcílio Oliveira: Diretor executivo.
- Fábio Rosato: Gerente de consultoria.
- José Cláudio Valh Júnior: Gerente de produtos de tecnologia.

O *software* usado para elaborar as modelagens e os diagramas de causalidade foi o *Vensim*, ele foi o escolhido para auxiliar na confecção do presente trabalho, pois possui uma interface simples e amigável que fornece um modo simples de construir modelos de simulação a partir de diagramas causais, além de facilitar o aprendizado da dinâmica de sistemas. Criado pela *ventana systems*, *vensim* é uma ferramenta para a modelagem de sistemas dinâmicos que possui configurações para estudantes e profissionais.

3. Resultados

Este capítulo apresenta os resultados obtidos com a aplicação da pesquisa. Serão apresentados o *design sprint*, a problemática em torno de requisitos apresenta pelos colaboradores da Sensedia, modelagens e diagramas estabelecidos como viáveis para o estudo de caso.

Por meio da pesquisa pode-se descobrir que a Sensedia em seu cenário atual adota a metodologia *design sprint*, segundo José Cláudio Valh gerente de produtos de tecnologia, justamente para tentar identificar as necessidades dos consumidores o quanto antes. Ao invés de levar meses para conseguir validar uma proposta de solução, através da metodologia da *Google*, é proposto que a validação leve apenas cinco dias.

José Cláudio Valh ainda ressalta que foi principalmente em função da plataforma de APIs que surgiu a necessidade do *design sprint* na Sensedia, porém antes de aplicar o *design sprint* na empresa, existem alguns procedimentos que são executados pela equipe responsável, esses procedimentos consistem em:

- Escolher o desafio correto: O desafio escolhido não pode ser muito grande e nem muito pequeno. Além disso, o desafio deve ser relevante, conciso, inspirador e focado na audiência e seguimento em questão.
- Escalar um time de campeões: São envolvidas no *design sprint* pessoas com habilidades diferentes.
- Bloquear a agenda de todos os envolvidos por cinco dias: Geralmente é uma tarefa árdua que exige muita renegociação, onde a empresa usa como argumento a importância do resultado que estão buscando, por isso a escolha do desafio é muito importante.
- Garantir uma *war room*: uma *war room* torna a experiência bem mais imersiva, levando em conta que a memória humana é de curta duração e não é boa, mais a espacial é. A Sensedia usa esquemas onde as ideias são descritas e coladas em lousas, deste modo qualquer funcionário tem acesso as ideias de colegas de trabalho.
- *Briefing*: Um email que apresenta qual será o desafio e com o que a empresa espera sair do outro lado.

José Cláudio Valh ainda explica como é o decorrer dos cinco dias do *design Sprint* na Sensedia:

Na segunda-feira alguns questionamentos são levantados, tais como: o que nós queremos com esse projeto? Onde queremos estar daqui a meses ou um ano? Se estivéssemos no final do projeto, quais questões estariam respondidas? Como e onde pode - se falhar? Quais os pontos que podem dar errado?

Em seguida a esses questionamentos, o foco é listar os principais clientes e atores a esquerda e o objetivo da empresa a direita. Depois disso, acontece um questionamento voltado para especialistas e usuários onde são abordados, e por fim listar ideias em forma de perguntas, transformando problemas em oportunidades e organizando os *post-its* e selecionar os mais votados, ao fim do dia será criado um caminho para toda a semana, um mapa.

A figura 10 representa o mapa resultante do fim do primeiro dia de *design sprint* que é basicamente um diagrama de como as coisas acontecem do início ao fim do produto ou serviço.

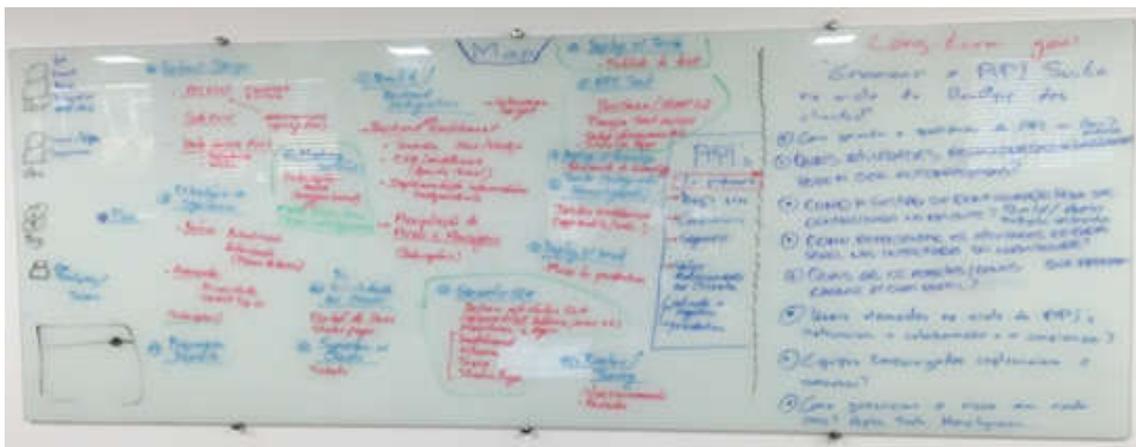


Figura 10: Mapa resultante do fim do primeiro dia da aplicação do *design Sprint*.
Fonte: Próprio autor.

No segundo dia são revisadas todas as questões do dia anterior. Posteriormente acontece o que chamamos de *sketch* ou esboço onde são selecionadas as notas mais promissoras, logo depois essas notas são transformadas em ideias grosseiras e depois as principais ideias são colocadas na parede na forma de um *storyboard*. *Storyboard's* precisam ser explicativos e anônimos, os *storubards* são basicamente os desenhos das ideias.

A figura 11 a seguir apresenta um mural com os *storyboard's* produzidos na Sensedia no fim do segundo dia de *design sprint*.



Figura 11: *storyboard's* produzidos no segundo dia da aplicação do design sprint.
Fonte: Próprio autor.

No terceiro dia de *design sprint* ocorre o que é denominado *decide*, ou seja, é o dia de decisão. Nesse dia todos os desenhos e ideias colados nas paredes da empresa (*storyboard's*) são analisados. Silenciosamente os funcionários envolvidos no *design sprint* andam colando votos nas partes mais interessantes. Naturalmente cada pessoa teria a chance de apresentar sua ideia, mas no ambiente da Sensedia isso não acontece, pois nesse caso o mais carismático venceria. Basicamente uma ideia boa tem que se sustentar no ambiente de trabalho da empresa, se elas estão confusas no *design sprint* então estarão também confusas para o cliente.

Cada pessoa inserida no *design sprint* recebe vinte adesivos podendo avaliar cada ideia de um a três adesivos. Em contra partida se não houver um consenso sobre as melhores ideias, será prevalecida à opinião dos decisores que são as pessoas que mais conhecem sobre a questão abordada no *design sprint*, os decisores recebem três adesivos cada para decidirem o que realmente será prototipado e assim criar um *storyboard* da solução só que dessa vez mais detalhado.

Na figura 12 pode - se ver o mural de *storyboards* durante o processo de votação. Os funcionários silenciosamente analisam as ideias e em seguida votam nas ideias que acharam mais interessantes.



Figura 12: o mural de *storyborads* durante o processo de votação.
Fonte: Próprio autor.

No quarto dia é realizada a fase de *prototype*, em outras palavras acontece a prototipação da ideia que foi selecionada no dia anterior. Nessa fase, a Sensedia trabalha na construção apenas do que é suficiente para validar, o que não significa que não tenha que parecer real. Nesse processo de prototipação todos podem contribuir, seja com um ícone, um texto ou até mesmo uma pergunta. Na figura 13 pode - se visualizar um exemplo de prototipação feita durante o quarto dia de trabalho do *design sprint* na sensedia:

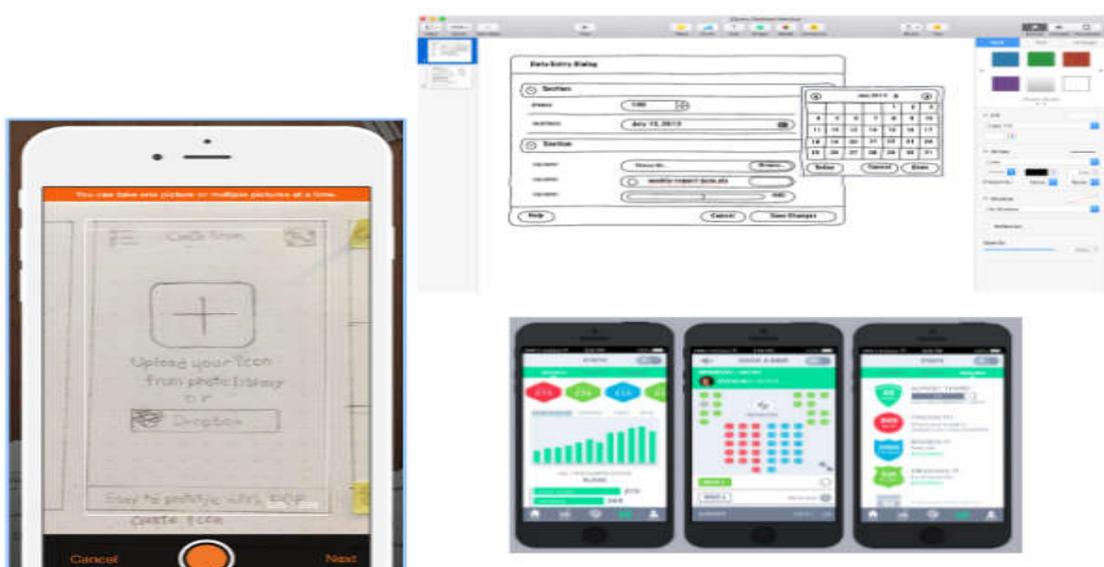


Figura 13: Exemplo de prototipação.
Fonte: Sensedia

Já no último dia do *design sprint* é realizada a etapa conhecida como *test*. Nesta etapa são realizados testes com a participação de *stakeholders* e clientes. A figura 14 Mostra a sala onde são realizados esses testes.

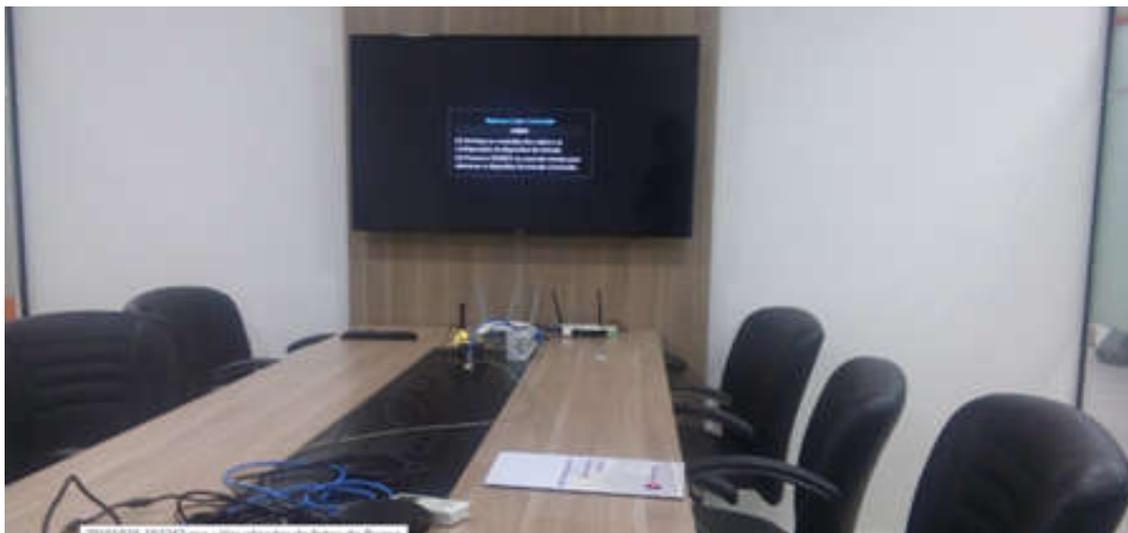


Figura 14: Ambiente onde acontecem os testes.
Fonte: Próprio autor

Nesse ambiente a Sensedia filma as reações do cliente ao ver o protótipo produzido no dia anterior e enquanto isso o cliente vai dando os *feedbacks*. A partir disso a sensedia consegue elaborar um resumo do *sprint* conseguindo elaborar uma estratégia de lançamento. Segundo afirma José Cláudio Vahl gerente de produtos da sensedia é uma boa ideia manter os participantes sobre a evolução do que foi discutido caso eles não estejam tão próximos.

3.1 Problemática analisada

Seguindo o primeiro passo proposto por Senge (2001), a situação de interesse a ser definida pela organização foi a seguinte:

A partir do *design sprint* aplicado pela empresa o resultado pode ser um MVP (*Minimum Viable Product*). De acordo com José Cláudio Valh gerente de produtos da Sensedia o *design sprint* é aplicado na empresa para entender os problemas e desenvolver

soluções, onde no final desse *design sprint* pode ser gerado um MVP que vai ser evoluído posteriormente independente do *design sprint*.

Ainda segundo José Claudio Valh, nos produtos desenvolvidos pela Sensedia o *design sprint* gera um MVP, esse MVP é implantado e entra em produção e posteriormente, os clientes começam a utilizar a *feature*, e deste modo, contribuem com *feedbacks*, em consequência, a empresa evolui o produto. José Cláudio Valh ainda ressalta que os *feedbacks* dos clientes não geram novos *design sprints*, a não ser que o *feedback* seja tão relevante que justifique desenvolver um *design sprint* com base no *feedback* recebido para a evolução de determinada funcionalidade, ou um *feedback* solicitando o acréscimo de uma funcionalidade não prevista no escopo do produto.

Diante do que foi descrito, em uma decisão dos colaboradores foi especificado que a situação analisada seria cenário onde o *feedback* do cliente é o acréscimo de uma nova funcionalidade no produto, na qual pode acarretar em um atraso na entrega do produto final.

De acordo com o que foi determinado, foi feito um estudo relacionado com os arquétipos abordados neste trabalho. Nessa pesquisa lista-se alguns possíveis pressupostos de tomadas de decisão:

- Abandonar um processo de gerência de requisitos.
- Contratação de novos profissionais.

3.2 Modelagem consertos que estragam

Agora com que temos a situação já definida pela organização, o segundo passo, de acordo com Senge (2001) é Identificar os eventos relacionados com a situação. Basicamente essa etapa consiste em levantar os eventos acerca da situação analisada. Tais eventos foram levantados de acordo com Fábio Rosato, Marcilio Oliveira e José Valh.

Como pontapé inicial para a realização deste estudo, o primeiro evento abordado pelos três colaboradores consiste em um cenário onde existe uma pressão exercida pelo cliente para que a entrega do produto seja realizada dentro do prazo, com a funcionalidade desenvolvida. Uma hipótese levantada para resolução deste problema seria abrir mão de uma Gerência de requisitos efetiva, deste modo, a economia de tempo seria maior, afetando diretamente em uma possível entrega dentro do cronograma previsto.

Para Pressman (2006) os gerentes com responsabilidade sobre o *software* estão frequentemente pressionados para obedecer a um cronograma no prazo e melhorar a qualidade. Tal pressão pode levar a promessas de entregas no prazo combinado, porém segundo Horstmann (2009) é notório o fato de que *softwares* comerciais nunca são distribuídos na data estabelecida. Com intuito de diminuir a pressão sofrida são feitas promessas de entrega do produto em curtos prazos para obter uma solução ainda que momentânea, com isso podem surgir consequências imprevistas em longo prazo que talvez exijam ações ainda mais impactantes. Diante do que foi descrito, pode-se identificar duas variáveis:

- Pressão do cliente na entrega de novas versões dentro do cronograma;
- Ausência de uma gerência de requisitos efetiva;

Devido a essa pressão exercida para entregar novas versões do produto obedecendo a um prazo e a um cronograma, são feitas promessas de entregas num curto espaço de tempo. “A partir disso, pode-se identificar uma relação entre as duas variáveis citadas acima, onde a variável “Pressão do cliente na entrega de novas versões dentro do cronograma” exerce uma relação de mesmo sentido sobre a variável” Ausência de análise de requisitos efetiva”, ou seja, a pressão do cliente gera a necessidade de alavancar o desenvolvimento, e como estratégia de economia de tempo foi abandonado o processo de análise de requisitos mais efetiva e detalhado. Por outro lado identificam-se outra relação entre as duas variáveis na qual quanto mais pleno for o processo de análise de requisitos, maior será a demora pela entrega de entrega de novas versões do produto, podendo até extrapolar o tempo previsto no cronograma. Sendo assim, pode - se identificar uma relação de sentido oposto. Observe a figura 15:



Figura 15: Diagrama de causalidade concertos que estragam.

Fonte Próprio autor

Com base no que foi descrito, levando em conta o cenário onde existe uma pressão do cliente para que novas versões do *software* sejam entregues dentro de um prazo previsto no cronograma, foi levantada inicialmente, para resolução do problema retratado a hipótese de que adotar um processo de análise mal definido como uma possível solução, uma vez que a economia de tempo seria maior, afetando no prazo de entrega.

Uma gerência de requisitos efetiva inicialmente não foi adotada com intuito de alavancar o projeto e entrega novas versões do produto dentro de um prazo combinado ao cliente, criando assim uma relação de mesmo sentido entre as variáveis “entrega de novas versões dentro do cronograma” e “ausência de uma gerencia de requisitos efetiva”. Porém, segundo Amaral (2015) o retrabalho gerado sobre uma implementação de um requisito é o principal indício de má qualidade de um produto, esse trabalho é dificilmente enxergado sem uma efetiva gerência de requisitos.

Com base no que foi descrito por Amaral (2015) pode - se identificar outra variável:

- *Software* de má qualidade.

Partindo do ponto de vista descrito, pode - se então entender que existe uma relação de mesmo sentido entre as variáveis “Ausência de uma análise de requisitos efetiva” e “*Software* de má qualidade”, onde a primeira variável pode culminar na segunda. Para Amaral (2015) o *software* de má qualidade culminará em uma pressão do cliente por consequência da qualidade do produto. Para Audy, Prikladnicki (2007) Uma má gerência de requisitos pode causar problemas, como a necessidade de um novo ciclo de especificação, projeto, codificação e teste afetando diretamente nos custos e prazos envolvidos, logo pode - se identificar uma relação de mesmo sentido entre as variáveis “*Software* de má qualidade” e Pressão do cliente por novas versões dentro do cronograma. Observe o diagrama na figura 16:

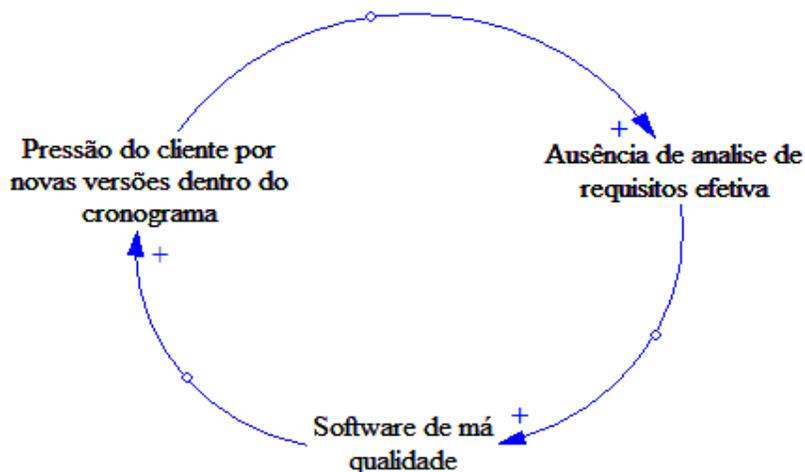


Figura 16: diagrama de causalidade concertos que estragam
Fonte:Próprio autor.

Deste modo pode-se depois de, elaborado os conhecimentos sobre as variáveis dessa problemática, criando os diagramas de causalidade. Pode-se então aplicar o arquétipo concertos que estragam, de acordo com o princípio gerencial descrito por Senge (1990). Na figura 17 a seguir pode - se observar as variáveis da problemática já modelada sob a visão do arquétipo proposto.

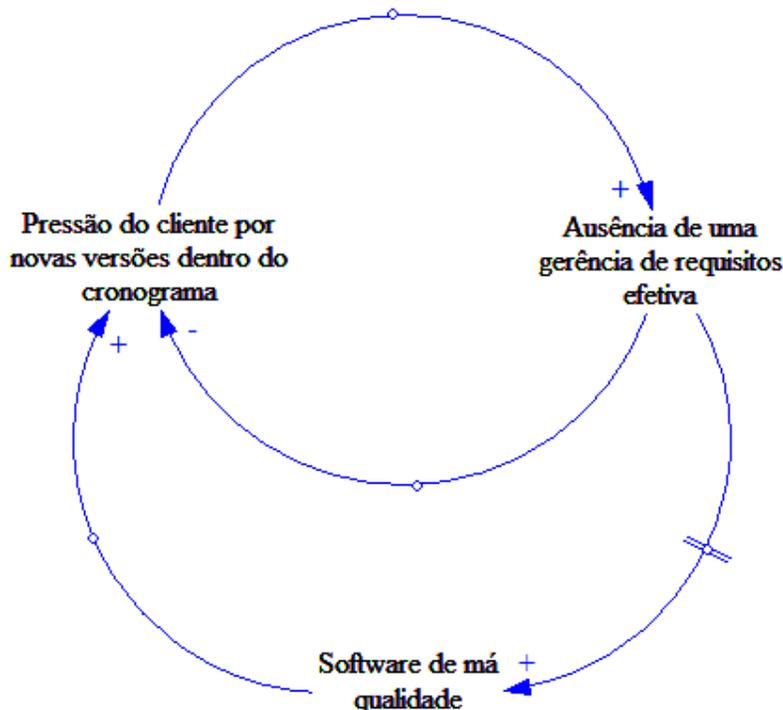


Figura 17: estrutura concertos que estragam.
Fonte: Próprio autor.

Diante da situação proveniente de um MVP que foi gerado a partir de um *design sprint*, onde, foi especificado que a situação analisada seria um *feedback* solicitando o acréscimo de uma nova funcionalidade no produto, considerando que a pressão do cliente é o centro do problema, onde essa pressão vai de encontro a entrega de novas versões do produto dentro do cronograma, para uma possível solução, foi adotada a hipótese de que a ausência de uma gerência de requisitos efetiva resolveria o problema visto que, isso iria implicar em um menor tempo de execução e conseqüentemente antecipar a entrega. Porém a partir do resultado e análise do arquétipo concertos que estragam, pode-se concluir que diante da pressão feita pelos clientes para entrega de novas versões do *software*, a hipótese levantada como possível solução apresenta uma defasagem ao longo do tempo resultando assim em conseqüências não intencionais.

3.3 Modelagem processo de equilíbrio com defasagem

Segundo Mendes (2013) sempre que uma nova funcionalidade for desejada, torna-se necessário adicionar ou modificar instruções já existentes no *software* e, portanto, defeitos podem ser introduzidos, aumentando o número de falhas e deteriorando a qualidade do software. Com isso retorna-se ao problema inicial que é a pressão do cliente para entrega do produto, porém dessa vez essa pressão vem com mais ênfase devido à qualidade do *software*.

Ainda segundo Mendes (2013) o tempo de desenvolvimento depende da equipe, onde a produtividade final do projeto depende da produtividade de cada uma das pessoas que trabalharam em equipe. Com base nas informações citadas por Mendes, tendo em vista a pressão do cliente pela entrega de novas versões, o pressuposto levantado dessa vez como possível solução se resume na contratação de novos profissionais,

Logo esta poderia ser uma solução em potencial ao problema estudado. Novos profissionais aumentariam a produtividade da equipe. Com base no que foi descrito acima, pode-se identificar duas variáveis:

- Pressão do cliente;
- Contratação de novos profissionais;

Diante do problema inicial que é a pressão exercida pelo cliente, a contratação de novos profissionais iria acarretar em uma produtividade maior e o objetivo seria alcançado com sucesso de uma forma mais rápida, estabelecendo uma relação de sentido oposto com a pressão do cliente, na qual quanto mais profissionais envolvidos no projeto maior será o rendimento.

Contratar novos profissionais poderia também ser uma solução em potencial para o problema da pressão do cliente por novas versões do produto com a nova funcionalidade, pois esse remanejamento de pessoas alavancaria o processo de desenvolvimento, deste modo entregando o produto dentro do cronograma.

Na figura 18 pode-se observar o comportamento dessas variáveis através do diagrama de causalidade.



Figura 18: Diagrama de causalidade processo de equilíbrio com defasagem.
Fonte: Próprio autor

Com base nas informações levantadas acima, pode - se modelar essa hipótese ao arquétipo processo de equilíbrio com defasagem, Observe na figura 19 a estrutura modelada

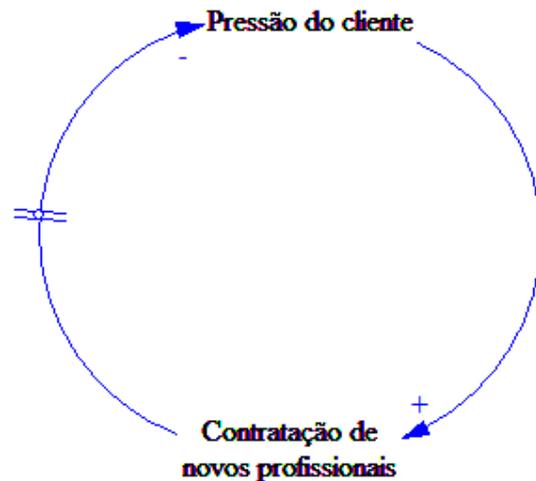


Figura 19: Estrutura do arquétipo processo de equilíbrio com defasagem.
Fonte: Próprio autor.

No entanto, em meio a essa relação existe uma defasagem indicada pelo arquétipo, esse *delay* está ligado diretamente ao tempo que os novos profissionais irão precisar para se situarem no projeto. Tal defasagem é mais bem explicada por Pressman (2011) onde ele afirma que acrescentar mais programadores para alavancar um projeto com o cronograma atrasado só irá o tornar mais atrasado ainda, quando novas pessoas entram, as que já estavam terão de gastar tempo situando os recém-chegados, reduzindo, conseqüentemente o tempo

destinado ao desenvolvimento produtivo. Pode se adicionar pessoas, mais somente de forma planejada e bem coordenada.

Pode-se entender que a hipótese de que o problema seria resolvido com a contratação de novos profissionais, visto que o tempo de desenvolvimento do *software* depende do desempenho de cada profissional, porém foi identificado através da análise do arquétipo que existe uma defasagem nessa hipótese. Tal defasagem está diretamente ligada a um mito gerencial, esse mito explicita que basicamente a contratação de novos profissionais para alavancar um projeto que já está em atraso, não seria uma boa solução por que os novos profissionais devem passar por um treinamento antes de integrar a equipe.

3.4 Modelagem sucesso para os bem sucedidos

O terceiro e último pressuposto levantado é de acordo com o modo com que a empresa trata realmente o problema em questão. José Cláudio Valh gerente de produtos da Sensedia cita que o *backlog* de produtos é dividido em *features* prioritárias e *features* menos importantes. A partir disso, os recursos são direcionados para o desenvolvimento das funcionalidades relevantes, em contrapartida são renunciados os recursos para as funcionalidades inexpressivas.

Porém de acordo com José Cláudio Valh conforme as funcionalidades prioritárias vão sendo concluídas, não necessariamente a equipe vai começar a atacar as menos importantes por que é necessário levar em consideração a passagem do tempo e conseqüentemente a entrada de novas potenciais *features* naquele *backlog*. Analisando-se no que foi descrito acima, nas quais a empresa visa uma evolução do produto diante de uma nova funcionalidade que foi acrescentada naquele *backlog*, pode-se identificar algumas variáveis para a construção do arquétipo:

- Recursos direcionados.
- Desenvolvimento de funcionalidades prioritárias.
- Desenvolvimento de funcionalidades inexpressivas.
- Conclusão do desenvolvimento das funcionalidades prioritárias.
- Conclusão do desenvolvimento das funcionalidades inexpressivas.

De acordo com tudo que foi descrito, pode-se verificar uma relação de mesmo sentido entre “recursos direcionados” e “desenvolvimento de funcionalidades prioritárias”, essa relação é recíproca entre essas variáveis, uma vez que, o desenvolvimento dessas funcionalidades depende diretamente da alocação de recursos. Na figura 20 pode-se observar o *loop* de causalidade entre essas duas variáveis.

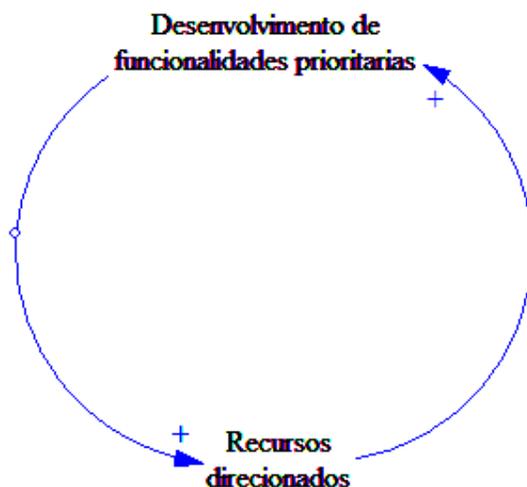


Figura 20: Diagrama de causalidade do arquétipo sucesso para os bens sucedidos.
Fonte: Próprio autor.

Diante deste raciocínio pode-se avaliar outra relação de mesmo sentido envolvendo o *loop* de causalidade acima e a variável “conclusão do desenvolvimento das funcionalidades prioritárias”, na qual, o desenvolvimento vai de encontro à conclusão e a conclusão depende diretamente dos recursos alocados. A figura 21 apresenta o diagrama de causalidade.



Figura 21: Diagrama de causalidade do arquétipo sucesso para os bens sucedidos
Fonte: Próprio autor

Por do outro lado, não pode-se fazer a mesma argumentação no que se refere às variáveis “recursos direcionados” e “desenvolvimento de funcionalidades inexpressivas”. Entre essas duas variáveis existem uma relação de sentido oposto, já que o desenvolvimento dessas funcionalidades depende dos recursos, recursos estes que não são direcionados a estas funcionalidades, essa relação é mútua entre “desenvolvimento de funcionalidades inexpressivas” e “recursos direcionados”, pois o desenvolvimento dessas funcionalidades depende dos recursos, o que inicialmente não existe para elas. Veja na figura 22 o *loop* de causalidade entre elas:



Figura 22: Diagrama de causalidade do arquétipo sucesso para os bens sucedidos
Fonte: Próprio autor

Proseguindo com a coerência do *loop* acima, pode-se discernir uma relação de mesmo sentido com a variável “conclusão do desenvolvimento das funcionalidades inexpressivas”, na qual essa relação se caracteriza a partir do momento que o desenvolvimento vai de encontro à conclusão, porém, a conclusão depende dos recursos, que não existem para essas funcionalidades, uma vez que os mesmos estão direcionados para o desenvolvimento das funcionalidades prioritárias, deste modo, caracteriza uma relação de sentido oposto.

A figura 23 ilustra o diagrama de causalidade, observe a imagem:



Figura 23: Diagrama de causalidade do arquétipo sucesso para os bem sucedidos.
Fonte: Próprio autor.

Após a elaboração dos diagramas de causalidade, pode-se modelar a situação de acordo com o arquétipo sucesso para os bem sucedidos, onde os mesmos e os relatos de José Valh se assemelham ao princípio gerencial descrito por Senge (1990). Observe a imagem:

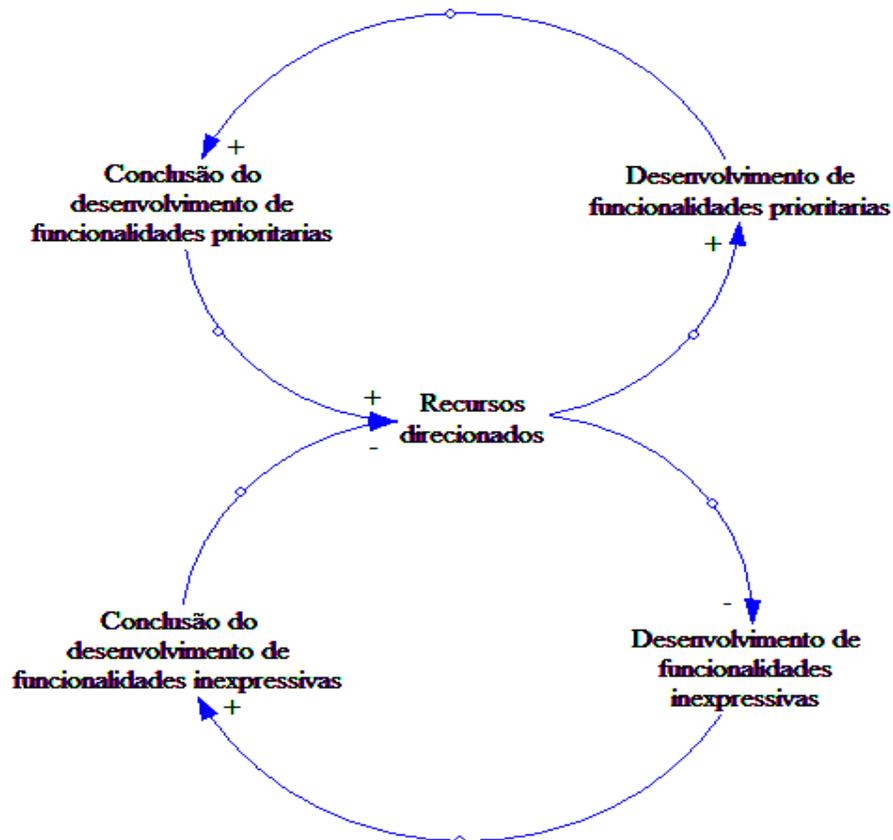


Figura 24: Estrutura do arquétipo sucesso para os bem sucedidos.
Fonte: Próprio Autor.

Pode-se perceber através da análise deste arquétipo que não existe defasagem alguma neste processo, percebe-se também que os recursos são voltados às funcionalidades prioritárias como já descrito antes e que funcionalidades inexpressivas não são implementadas conforme as funcionalidades mais importantes estão em desenvolvimento.

3.5 Análise das modelagens

Nesta seção serão analisadas pelos colaboradores Marcílio Oliveira, Fábio Rosato e José Cláudio, as estruturas dos arquétipos modelados.

3.6 Análise consertos que estragam

Através da análise do primeiro arquétipo, pode-se entender que a hipótese de que a ausência de uma gerência de requisitos efetiva resolveria o problema visto que, isso iria implicar em um menor tempo de execução e conseqüentemente antecipar a entrega. Porém, esse arquétipo nos mostra uma defasagem ao longo do tempo, onde essa defasagem fica caracterizada em conseqüências não intencionais que, de acordo com os profissionais da Sensedia, conseqüências que podem afetar diretamente na integridade do produto podendo acarretar em problemas ainda piores como, por exemplo, a perda da credibilidade e confiança na equipe.

Fábio Rosato gerente de consultoria da Sensedia afirma que quase sempre o pagamento do projeto está atrelado a entregas. Se um novo requisito ou funcionalidade é solicitado, essa é uma boa oportunidade para replanejar o projeto e diminuir a pressão existente junto ao cliente, para José Valh atrasos podem existir, se a empresa falha nas entregas essa defasagem indicada no arquétipo pode ser também a perda de um cliente para uma empresa concorrente.

3.7 Análise processo de equilíbrio com defasagem

Levando em consideração que os dados e resultados da análise da modelagem anterior, a hipótese da vez foi contratar novos profissionais para solucionar o problema, visto que o seria necessário alavancar o desenvolvimento para sanar o problema da pressão do cliente, porém de acordo com a modelagem processo de equilíbrio com defasagem, existe uma *delay* em meio a esse processo de remanejamento de pessoas. Para Fábio Rosato adicionar novas pessoas realmente tende a atrapalhar, mas segundo ele se a complexidade do projeto for baixa e o nível de senioridade for alto, colocar mais pessoas no projeto pode ajudar na alavancagem, nesse contexto pessoas seniores tendem a atuar com autonomia e a complexidade baixa ajuda a baixar a curva de aprendizagem, Fora a hipótese levantada por Fabio Rosato, foi considerada pelos profissionais da Sensedia que participaram desta análise que a contratação de novos profissionais não é uma tomada de decisão adequada para alavancar o desenvolvimento quando é levada em consideração a pressão do cliente para entrega da evolução do produto, pois o gasto de tempo com recursos intelectuais para o treinamento dos recém-contratados seria um potencial atraso. Ao invés de contratar novos profissionais, fazer horas extras também surge como solução alternativa, porém segundo Fábio Rosato afirma que este tende a ser o último recurso, pois envolve custos adicionais e podem também desmotivar a equipe.

3.8 Análise sucesso para os bem sucedidos

Pode-se perceber então que as duas hipóteses de possíveis tomadas de decisão não se mostraram eficientes, uma vez que a análise dos arquétipos deixam bem claro que existem defasagens que vão influenciar diretamente na entrega do produto final. Porém, a análise do arquétipo sucesso para os bem sucedidos não nos mostra defasagem alguma. Esse arquétipo ilustra bem como é o modo que a empresa trata o problema de possíveis funcionalidades não previstas no escopo.

A opinião dos três profissionais atuantes na empresa que auxiliaram nessa pesquisa em torno da análise desse arquétipo é basicamente a seguinte: Pensando num período de

desenvolvimento, nem sempre a conclusão das funcionalidades prioritárias implicará no início do desenvolvimento das funcionalidades menos importantes, durante esse tempo acontece à entrada de novas funcionalidades, logo ao invés de atacar as menos importantes é gerado um novo *backlog* com as funcionalidades menos importantes, mas também com todas as funcionalidades que entraram e é realizado um novo processo para organizar o *backlog*. Em meio esse processo pode ser que as funcionalidades menos importantes continuem menos importantes e os recursos novamente vão ser direcionados para implementar as funcionalidades mais importantes.

De acordo com José Valh nesse processo pode ser que as menos importantes se tornem obsoletas, tais funcionalidades são retiradas do escopo do projeto em um processo que envolve o cliente. Deste modo ocorre a troca de funcionalidades, onde as novas que entraram para o escopo substituem as que se tornaram obsoletas, com isso é possível uma economia no tempo de entrega e da mesma forma o cliente fica ciente do desenvolvimento do projeto através do *backlog* de produto.

De acordo com Marcilio Oliveira diretor executivo da Empresa, o cliente tem acesso ao *backlog* e participa do processo onde são priorizadas as funcionalidades no escopo, com isso ele consegue perceber quais funcionalidades serão desenvolvidas na próxima versão ou em outra versão posteriormente, caso ele fique insatisfeito com a ordem pela qual as *features* estão sendo desenvolvidas ele pode entrar em contato solicitando que a funcionalidade que ele gostaria que saísse mais rápido fosse feita sob demanda.

Pode-se então, através da reflexão imposta pelas modelagens da dinâmica de sistemas, perceber que o método como a Sensedia trata o problema se faz melhor que as hipóteses levantadas, uma vez que, nem a ausência de uma boa gerência de requisitos e tão quanto a contratação de novos profissionais não são boas saídas para o problema em questão, pois ambas não iriam sanar o problema imposto, pelo contrario, ao longo do tempo iriam proporcionar problemas mais impactantes. Contudo, pode - se também perceber que o método como a Sensedia trata o problema é sem duvida alguma, melhor que as hipóteses levantadas, pois, a economia de tempo com as eliminações de funcionalidades que se tornam obsoletas, e o processo no qual o cliente faz parte, com o intuito de organizar as prioridades no *backlog* iria diminuir a expectativa para entrega, uma vez que o cliente ficaria por dentro da ordem de desenvolvimento especificada. De acordo com Mendes (2013), é possível congelar o escopo do projeto, mais não a expectativa do cliente. O processo que ocorre na Sensedia contradiz essa ideia, onde o escopo pode ser alterado e a expectativa do cliente não é necessariamente

congelada, mais sim acompanha o desenvolvimento do projeto através de cada *release*, onde ele fica ciente do desenvolvimento de todo o projeto.

3.9 Discussão dos resultados

Através das análises feitas através dos arquétipos, considerando alguns fatores como a possível entrada de novas funcionalidades no escopo e um projeto em atraso, considerando a pressão do cliente por novas entregas de versões do *software*, o arquétipo consertos que estragam expõe que existe uma defasagem quando se abandona uma gerência de requisitos efetiva com o intuito de economizar tempo de execução e por consequência antecipar a entrega. Essa defasagem pode afetar diretamente na integridade do *software*.

Contudo, considerando os mesmos fatores, o arquétipo processo de equilíbrio com defasagem também evidencia uma defasagem a partir do momento que é feita a contratação de novos profissionais para alavancar o projeto em atraso, devido ao fato de que, os novos profissionais que irão precisar de tempo para se situarem no projeto e devem passar por treinamento.

O arquétipo sucesso para os bem sucedidos retrata a realidade da Sensedia, através de sua análise podemos perceber que os recursos são voltados para o desenvolvimento das funcionalidades mais importantes, enquanto as funcionalidades menos relevantes são desprovidas desses recursos. Ao passo que novas funcionalidades são adicionadas ao escopo, um novo *backlog* é e feito contendo as funcionalidades menos importantes e as que entraram no escopo, pode ser que nesse *backlog* feito as menos importantes continuaram irrelevantes e não serão desenvolvidas naquele momento.

4. Conclusão

O presente trabalho procurou analisar o cenário atual de desenvolvimento de *software* e gestão de requisitos da Sensedia e relacioná-lo com a dinâmica de sistemas, identificando fatores associados a este cenário a partir da construção dos diagramas de causalidade, pode-se assim, eleger os arquétipos que mais enquadram-se ao cenário da situação proposta pela problemática. Para que isso fosse feito, buscou-se conhecimentos da situação bem como fatores que podem influenciar na questão analisada dentro da empresa, e também foi buscado conhecimento em referências bibliográfico com o intuito de fortalecer o trabalho de informações.

A partir da realização deste trabalho procurou-se entender mais sobre o modo como a empresa aplica o *design sprint*, tentando buscar entendimento partindo da problemática onde o cliente solicita a inserção de uma nova funcionalidade não prevista no escopo do projeto.

Além disso, foi possível mostrar que a dinâmica de sistemas, com o uso de seus conceitos (variáveis, *loops*, relações, modelos) pode ser uma ferramenta poderosa na busca de entendimento de situações ligadas ao ambiente de uma empresa de desenvolvimento de *software* podendo impactar positivamente na gerência de seus projetos.

Mostrar que a partir da leitura dos arquétipos do modelo *soft* é possível refletir sobre o comportamento da situação analisada na Sensedia gerando aprendizado em torno do cenário abordado, onde esse aprendizado pode servir de estratégia de ação.

Para gestores da empresa que contribuíram com essa pesquisa, levando em consideração alguns fatores como a possível entrada de novas funcionalidades no escopo e um projeto em atraso, considerando a pressão do cliente por novas entregas de versões do *software*, a defasagem exposta pelo arquétipo concertos que estragam podem ser falhas no processo de desenvolvimento, a perda de credibilidade e confiança na equipe ou até mesmo a perda de um cliente para uma empresa concorrente.

Em contrapartida, considerando os mesmos fatores, o arquétipo processo de equilíbrio com defasagem também evidencia uma defasagem a partir do momento que é feita a contratação de novos profissionais para alavancar o projeto em atraso. Para os gestores essa defasagem está atrelada ao tempo gasto com recursos intelectuais para o treinamento dos recém-chegados.

Por fim, o arquétipo sucesso para os bem sucedidos retrata a realidade da Sensedia, onde através de sua análise pode-se notar que a conclusão do desenvolvimento das

funcionalidades mais importantes não implicará no desenvolvimento das funcionalidades menos relevantes, levando-se em conta a passagem do tempo e a possível entrada de novas funcionalidades, é gerado um novo *backlog* com as funcionalidades menos importantes, mas também, com as funcionalidades que entraram no escopo. Em meio esse processo pode ser que as funcionalidades menos importantes continuem inexpressivas e os recursos vão ser direcionados novamente para as funcionalidades mais importantes.

Nesse processo, pode ser que as menos importantes se tornem obsoletas, tais funcionalidades são retiradas do escopo em um processo que envolve o cliente, deste modo ocorre a troca de funcionalidades onde as novas que surgiram substituem as que foram excluídas, com isso é possível uma economia no tempo de entrega e o cliente fica ciente do desenvolvimento do projeto.

Pode-se concluir através das modelagens dos arquétipos da dinâmica de sistemas, que o método como a sensedia trata o problema se faz melhor que os pressupostos levantados, uma vez que, a ausência de uma boa gerência de requisitos nem a contratação de novos profissionais não se fazem boas saídas para o problema em questão. Contudo, pode-se perceber que o método como a sensedia trata o problema se faz melhor, pois a economia de tempo com as funcionalidades que se tornam obsoletas e o processo no qual o cliente faz parte com o objetivo de organizar o *backlog* iria diminuir a expectativa para entrega.

5.Trabalhos futuros

Devido este trabalho se tratar de um estudo de caso que apresenta as observações realizadas pelo autor com o auxílio de alguns profissionais atuantes da empresa durante a realização do *design sprint*, é possível estender este estudo através de novos pressupostos acerca do assunto, ou até mesmo um estudo mais detalhado sobre o *design sprint* ou o processo de MVP gerado a partir dele. Pela natureza geral da dinâmica de sistemas é possível emprega-la a outros sistemas, podendo assim analisar outras formas de desenvolvimento de *software*, bem como realizar um estudo sistêmico acerca de algumas metodologias ágeis (*scrum*, *kanbam*, *xp*) podendo propor análises em outras empresas de desenvolvimento de *software*.

REFERÊNCIAS

AGUILAR, Luis Joyanes. **Fundamentos de programação: Algoritmos, estrutura de dados e objetos**. 3ª edição AMGH 2011, 654.

AMARAL, Maxwell Lelpo do. **Implantação de melhoria de processos de software com CMMI – DEV NÍVEL 2: Planejamento baseado em exemplos**. IFPB 2015, 202.

AUDY, PRIKLADNICKI. Jorge Luis, Rafael. **Desenvolvimento distribuído de software: Desenvolvimento de software com equipes distribuídas**: Desenvolvimento de software com equipes distribuídas. Elsevier 2007, 210.

BANFIELD, LOMBARDO, WAX. Richard, C. Tood, trace. *Design sprint: A practical guidebook for building great digital products*. Copyright 2016, 232.

BARTIÉ, Alexandre. **Garantia de qualidade de software: As melhores praticas de engenharia de software aplicadas a sua empresa**. Campus 2002, 291.

BEZERRA, Eduardo. **Princípios de análise e projeto de sistemas com uml**. 3ª edição Elsevier 2015, 320.

BOYETT, BOYETT. Joseph, Jimmie. **Hablan los gúrus: las mejores ideias de los máximos pensadores de La administracion**. Norma S.A 1998, 387.

BROOKSHEAR, J Glenn. **Ciência da computação: Uma visão abrangente**. 11ª edição Bookman 2013, 502.

CHAIM, Ricardo Matos. **Modelagem, simulação e dinâmica de sistemas**. 2011, 48.

CHWIF, MEDINA. **Modelagens e simulações de eventos discretos: Teoria e aplicações**. 4ª edição Elsevier 2014, 320.

CRUZ, Fábio. **PMO Ágil: Escritório ágil de gerenciamento de projetos**. Brasport 2016, 453.

CERASOLI, Jeancarlo. **Design sprint: O remédio definitivo?** CIO Magazine. Disponível em < <http://cio.com.br/opinioao/2016/09/15/design-sprint-o-remedio-definitivo/>> Acessado em 03 nov. 2016.

FELICIO, Luis Carlos. **Modelagem da dinâmica de sistemas e estudo da resposta**. 2ª edição Rima 2010,551.

FRANCO, Raoni de Oliveira Sales; Penaforte, Daniel de Andrade. **Uma análise sistêmica dos problemas enfrentados por empresas de software durante o processo de engenharia de requisitos**. Universidade Federal de Pernambuco - UFPE, 53.

GUERRINI, ESCRIVÃO FILHO, CAZARINI, PÁDUA. Fábio Müller, Edmundo, Edson, Walmir, Silvia Inês Dallavalle. **Modelagem da organização: Uma visão integrada**. Bookman 2014, 128.

HIRAMA, Kechi. **Engenharia de software: Qualidade e produtividade com tecnologia**. Elsevier 2012, 209.

HORSTMANN, Cay. **Conceitos de computação com Java**. 5ª edição Bookman 2009, 718.

IIBA, *international institute of business analytics*. **O guia para o corpo de conhecimento para análise de negócios. (GUIA BABOK)** versão 2.0

KRUTCHEN,Philippe. **Utilizando UML e padrões. Uma introdução a análise e ao projeto orientados a objetos e ao desenvolvimento iterativo**. 3ª edição Bookman 2007, 694.

MARUJO, Lino Guimarães. **Um modelo dinâmico de gerenciamento de projetos baseado em recursos**. CEFET– RJ ENEGEP 2007, 10.

MARTINS, José Carlos Cordeiro. **Técnicas de gerenciamento de projetos de software.** Brasport, 2007. 465.

MASSARI, Vitor L. **AGILE Scrum Master no gerenciamento Avançado de projetos:** Base para certificação EXIN Agile Scrum Master. Brasport 2016, 295.

MEIRA, Silvio. **Novos negócios inovadores de crescimento empreendedor no Brasil.** Casa da palavra 2013,

MENDES, Antônio. **Custo do Software:** Planejamento e gestão. 1º edição Elsevier 2013, 184. 416.

OCAÑA, José Andrés. **Pienso, luego mi empresa existe.** Editorial club universitario 2006, 379.

OLIVEIRA, Rui Carvalho; FERREIRA, José Soeiro. **Investigação operacional em ação: Casos de aplicação.** Apdio 2014, 741.

PRESSMAN, Roger S. **Engenharia de software.** 6ª edição Mc Graw – Hill 2006, 720.

PRIKLADNICKI, WILLI, MILANI. Rafael, Renato, Fabiano. **Métodos ágeis para desenvolvimento de software.** Bookman 2014, 288.

RADZICKI, J Michael; ROBERT A Taylor. **Introduction to system dynamics: A system approach to understanding Complex Policy Issues,** US Department of Energy, 1997.

Disponível em < <http://www.systemdynamics.org/DL-IntroSysDyn/index.html> > Acessado em 12 jul.2016.

REZENDE, Denis Alcides. **Engenharia de software e sistemas de informação.** 3ª edição Brasport, 2005. 297.

RIES, Eric. **A startup enxuta.** Como os empreendedores atuais utilizam a inovação contínua para criar empresas extremamente bem sucedidas. leYA 2011.

ROSSONI, Luciano. **Modelagem e simulação *soft* em estratégia soft modelling and simulation and in strategy.** Universidade Federal do Paraná – UFPR 2006.

SALES FRANCO, PENAFORTE. Raoni de Oliveira, Daniel de Andrade. **Uma análise sistêmica dos problemas enfrentados por empresas de software durante o processo de engenharia de requisitos.** Universidade federal de Pernambuco – UFPE 2009, 53.

SCHACH, Stephen.R. Engenharia de software: Os paradigmas clássico orientado a objetos. 7ª edição Bookman 2010, 618.

SENGE, Peter M. **A quinta disciplina: Arte e pratica da organização que aprende.** Beste Seller 2001, 466.

SENGE, Peter M. **A quinta disciplina: Arte e pratica da organização que aprende.** Uma nova e revolucionaria concepção de liderança e gerenciamento empresarial 24ª edição Beste Seller 1990,443

_____ Sensedia. Disponível em < <http://sensedia.com> > Acessado em 19 Jul 2016.

SOLER, Manzini Alonso. **Gerenciamento de projetos em tirinhas:** Especialistas comentam a vida de trabalho de Rosalina, a gerente de projetos. Brasport 2015, 319.

STERMAN, John, D. **Business dynamics: Systems thinking and modeling for a complex word.** McGraw-Hill 2000, 982.

TELLES, MATOS. André; Carlos. **Empreendedor viável:** Uma mentoria para as empresas na era da cultura startup. Leya 2013, 160.

VALENÇA, Antonio Carlos; GRATULIANO, João. **Pensamento sistêmico: 25 aplicações práticas.** Bargaço, 1999.

VILELA, Paulo. **Introdução a dinâmica de sistemas.** Universidade Federal de Juiz de Fora
– UFJF 2005, 59

ANEXO 1 – AUTORIZAÇÃO PARA REDAÇÃO DE ESTUDO DE CASO

