

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE DESENHO INDUSTRIAL
CURSO DE BACHARELADO EM DESIGN

WELLITON OLIVEIRA SANTOS

**PROPOSTA DE UM *FRAMEWORK* ÁGIL APLICADO A PROJETOS
DE DESIGN**

TRABALHO DE CONCLUSÃO DE CURSO

CURITIBA
2015

WELLITON OLIVEIRA SANTOS

**PROPOSTA DE UM *FRAMEWORK* ÁGIL APLICADO A PROJETOS
DE DESIGN**

Trabalho de Conclusão de Curso apresentado como requisito parcial para aprovação na disciplina de TCC 2, do Curso de Bacharelado em Design, Departamento Acadêmico de Desenho Industrial, da Universidade Tecnológica Federal do Paraná, Campus Curitiba.

Orientador: Prof. MSC. Manoel Alexandre Schroeder.

CURITIBA
2015

DEDICATÓRIA

A dona Maria e o senhor Florentino por me criarem com essa ânsia em abraçar o mundo e também todas as pessoas que em algum momento passaram por mim nesses 26 anos. Um pouco de cada um de vocês está nesse trabalho.

AGRADECIMENTOS

O trabalho a seguir é produto de mais de dois anos de pesquisa sobre os mais variados assuntos. Ele foi construído, acima de tudo por várias horas de discussão, conversas em mesa de bar, conselhos e incentivos de um grande grupo de pessoas que, sem as quais nada disso seria possível.

Agradeço primeiramente ao meu pai Florentino Alves dos Santos, que fez a minha inscrição no vestibular para Design, mesmo não entendendo muito bem o que um designer faz (até hoje). A minha mãe, Maria Claudete de Oliveira Santos e as suas lembranças diárias de “já terminou esse TCC?” que foram primordiais para o meu desespero e conseqüentemente o termino desse trabalho. Sou grato a todos os meus colegas de faculdade e amigos que de alguma forma me apoiaram ao longo desse projeto. Seria impossível citar os nomes de todos aqui, mas se você está lendo isso, é muito provavelmente que você esteja incluso. Um agradecimento especial as pessoas que estiveram (sofreram) comigo até a véspera da entrega: Josiane Tochetto minha ex-dupla que me impediu várias vezes de dormir em cima do teclado, ao Paulo Abreu pelas muitas cervejas e horas de conversa, a Isadora Tonet minha fiel escudeira que praticamente me carregou nos últimos meses antes da entrega.

Aos membros CADUT Gestão 2013-2014: Andréia Damaceno, Ana Cintra, Stephanie Cleto, Ana Guimarães, e novamente Paulo Abreu e Isadora Tonet, que acreditaram na proposta de reerguer o centro acadêmico o que me ajudou a descobrir meu gosto por gestão (matéria prima desse trabalho) e também me ajudaram a atrasar ainda mais esse projeto e conseqüentemente estender meu tempo de curso.

Agradeço ao meu orientador Manoel Alexandre Schroeder, acredito que eu tenha sido um dos orientandos mais estranhos que ele já teve na vida. As nossas muitas reuniões nesses dois anos, em que passamos a maior parte do tempo conversando sobre qualquer coisa, menos sobre o TCC me ajudaram muito a encontrar os melhores caminhos não só para esse trabalho, mas para a minha carreira.

“Depois de escalar uma montanha muito alta, descobrimos que há muitas outras montanhas por escalar.”

Nelson Mandela

RESUMO

SANTOS, Welliton Oliveira. **PROPOSTA DE UM FRAMEWORK ÁGIL APLICADO A PROJETOS DE DESIGN**. 2015. Trabalho de Conclusão de Curso – Bacharelado em Design, Universidade Tecnológica Federal do Paraná. Curitiba, 2015.

O presente trabalho de conclusão de curso consiste na proposta de um *framework* ágil destinado ao desenvolvimento de projetos de design, levando como pré-requisitos indispensáveis a qualidade da entrega, a versatilidade de sua aplicação, economia de tempo e a redução de recursos financeiros. O processo de elaboração passou por uma revisão das principais metodologias de design lecionadas no meio acadêmico, considerando o contexto em que foram concebidas e os metodólogos que as elaboraram. Na sequência a pesquisa aborda as origens do movimento Ágil e suas metodologias, buscando com isso contextualizar o movimento e encontrar pontos comuns entre o desenvolvimento ágil e o desenvolvimento de projetos de design. Para tal, apresentamos um modelo de análise de complexidade em ambientes de desenvolvimento, denominado *Cynefin* e um resumo sobre um dos principais *frameworks* desenvolvidos pelos fundadores do movimento Ágil. Na etapa seguinte é elaborado de forma mais detalhada o *Scrum*, um dos mais conhecidos *frameworks* Ágil. Dentro da apresentação do *Scrum* são exemplificadas minuciosamente todas as suas etapas, nomenclaturas e funções, buscando assim transpor os pontos positivos do *framework* para a proposta do presente trabalho. Buscando complementar o modelo e corrigir faltas da metodologia de design e das metodologias ágeis fez-se necessário o estudo de rotinas e métodos dedicados a controle de qualidade e solução de problemas. Os dados coletados nessas pesquisas deram subsídios para a elaboração da proposta, que é apresentada com todas as suas etapas e é posteriormente exemplificada. A fim de aprimorar o trabalho, outros métodos foram adicionados ao *framework*, como conceitos de *brainstorming*, elaboração de planos de ação, definição e implantação de *roadmaps*, criação de *dashboards*, aplicação do *brandkey*: um modelo metodológico de desenvolvimento de marcas e a aplicação de indicadores de desempenho. Além disso, o projeto fica aberto à inclusão de módulos adicionais o deixando ainda mais flexível e ampliando sua aplicabilidade de forma eficaz. A conclusão do trabalho se da por meio de um exemplo didático das principais funcionalidades do *framework* aplicado em um projeto real.

Palavras-chave: Design. Metodologia. Ágil. Framework.

ABSTRACT

SANTOS, Welliton Oliveira. **AN AGILE FRAMEWORK PROPOSAL APPLIED TO DESIGN PROJECTS**. 2015. Final Year Research Project – Bachelor in Design, Federal University of Technology - Paraná. Curitiba, 2015.

This final year project is a proposal of a flexible framework for design projects development, taking as essential prerequisites: delivery quality, versatility of application, time saving and low cost resources. The drafting process has undergone a review of the main design methodologies taught in academic field, considering the context in which they were designed and methodologists that have developed. Following, the research deals with the Agile Movement origins and its methodologies, seeking to contextualize this movement and find common points between this one and the development of design projects. In order to do this, a analysis of complexity on development environments is presented (Cynefin), as well as a summary of the main frameworks made by founders of Movement Agile. The next step, Scrum is more detailed, one of the most popular Agile frameworks. Within the presentation of Scrum, all its stages are thoroughly exemplified, even as classifications and functions, seeking to transpose the good points of the framework for the proposal of this work. Complementing the model and correct design and agile methodologies shortages, it was necessary to study the routines and methods dedicated to quality control and problem solving. The data collected in these surveys gave support for the preparation of the proposal, which is presented in all its stages and is subsequently exemplified. In order to improve the work, other methods have been added to the framework, such as brainstorming concepts, preparation of action plans, defining and implementing roadmaps, creating dashboards, application of brandkey (a methodological model of brand development) and the application of performance indicators. Besides, the project is flexible to the inclusion of additional modules leaving it even more bendable and expanding its applicability effectively. This project completion is through a textbook example of the main features of the framework applied in a real project.

Keywords: Design. Methodology. Agile. Framework.

LISTA DE QUADROS

Quadro 1 - Comparativo das metodologias de design	25
Quadro 2 - <i>Brainstorming</i> categorizado.....	49

LISTA DE FIGURAS

FIGURA 1 – ETAPAS DA METODOLOGIA SEGUNDO MUNARI	19
FIGURA 2 – ETAPAS DA METODOLOGIA SEGUNDO LÖBACH	21
FIGURA 3 – ETAPAS DA METODOLOGIA SEGUNDO MESTRINER	25
FIGURA 4 – DIAGRAMA DE <i>CYNEFIN</i>	28
FIGURA 5 – EXEMPLO CICLO <i>SPRINT</i>	32
FIGURA 6 – EXEMPLO DE <i>SPRINT BACKLOG</i>	34
FIGURA 7 – EXEMPLO DE <i>BURNOUT CHART</i>	35
FIGURA 8 – FLUXO MASP.....	38
FIGURA 9 – ETAPAS.....	41
FIGURA 10 – ETAPAS 2.....	42
FIGURA 11 – ETAPAS 3.....	43
FIGURA 12 – ETAPAS 4.....	44
FIGURA 13 – DIAGRAMA DE CAUSA-RAIZ	48
FIGURA 14 – APLICAÇÃO DIAGRAMA DE CAUSA-RAIZ.....	50
FIGURA 15 – <i>VALUE DESIGN PROPOSITION</i>	51
FIGURA 16 – <i>VALUE DESIGN PROPOSITION</i>	53
FIGURA 17 – REPRESENTAÇÃO 5W2H.....	54
FIGURA 18 – MATRIZ DE PRIORIZAÇÃO.....	55
FIGURA 19 – <i>PRODUCT DASHBOARD</i>	57
FIGURA 20 – <i>PRODUCT DASHBOARD</i>	59
FIGURA 21 – <i>ROADMAP</i> DO PROJETO.....	60
FIGURA 22 – <i>BURNOUT CHART</i> DA <i>SPRINT</i>	63
FIGURA 23 – <i>DASHBOARD</i> DA REUNIÃO DE <i>SPRINT RETROSPECTIVE</i>	65
FIGURA 24 – EXEMPLO <i>BURNOUT CHART</i> DA <i>SPRINT</i>	67
FIGURA 25 – MODELO DE <i>BRANDKEY</i>	68
FIGURA 26 – <i>BRANDKEY</i> PREENCHIDO.	70

SUMÁRIO

1. INTRODUÇÃO	9
1.1 JUSTIFICATIVA	11
1.2 OBJETIVOS	13
1.2.1 Objetivo geral.....	13
1.2.2 Objetivos específicos	13
1.3 ESTRUTURA DO TRABALHO	13
2. PROJETO	14
2.1 PROCEDIMENTOS METODOLÓGICOS.....	14
2.2 PREPARAÇÃO	14
2.2.1 Pesquisa sobre metodologias de design	14
2.2.2 Principais nomes da metodologia	18
2.2.2.1 Bruno Munari.....	18
2.2.2.2 Bernd Löbach	20
2.2.2.3 Mike Baxter	22
2.2.2.4 Fábio Mestriner	24
2.2.3 Desenvolvimento ágil	25
2.2.3.1 Sobre o <i>scrum</i>	26
2.2.3.2 Conceitos de ambientes de desenvolvimento complexo	27
2.2.3.3 A estrutura do <i>scrum</i>	31
2.2.4 Ferramentas de qualidade.....	36
2.2.4.1 Utilizando os conceitos de qualidade para entender o problema	37
3. VALIDAÇÃO DO PROJETO	41
3.1 O MODELO PROPOSTO.....	41
3.2 APLICAÇÃO	45
3.2.1 Objetivo do projeto.	45
3.2.2 Coleta e análise de dados.	45
3.2.2.1 <i>Brainstorming</i>	46
3.2.2.2 Diagrama de ishikawa	47
3.2.2.3 <i>Value proposition design</i>	50
3.2.3 Planejamento.	53
3.2.3.2 Plano de ação.	53
3.2.3.2 <i>Product dashboard</i>	57
3.2.3.3 <i>Roadmap</i>	59
3.2.4 <i>Sprint</i>	60
3.2.4.1 Componentes de uma <i>sprint</i>	62
4. CONSIDERAÇÕES FINAIS	72
REFERÊNCIAS	74

1 INTRODUÇÃO

Durante aproximadamente quatro anos (tempo médio que dura uma graduação de design no Brasil) o estudante de design tem contato com uma série de conhecimentos úteis que constroem a visão necessária para que o estudante, quase de forma inconsciente, consiga efetuar atividades atribuídas a sua profissão, ser um designer.

Entre essa gama de conhecimentos, um em específico tem um valor especial, a metodologia. Seja qual for o foco do estudante: desenvolvimento de produtos, programação visual, design de serviços, entre outros. As metodologias trazem em si uma mesma função: apresentar um modelo de planejamento para realização de um projeto.

A definição de metodologia muitas vezes se confunde com a própria definição de o que é design, isso pode ser notado na forma com que os teóricos e renomados profissionais de design definem a profissão. "Design é projeto", diz Alexandre Wollner (2005 apud STOLARSKI, 2005), pioneiro do design gráfico no Brasil. Para Löbach (2001, p. 16) o design pode ser entendido como "a concretização de uma ideia em forma de projetos ou modelos, mediante a construção e configuração resultando em um produto industrial passível de produção em série". Em 1588 quando pela primeira vez o termo design apareceu descrito no "*Oxford Dictionary*" como "Um plano desenvolvido pelo homem ou um esquema que possa ser realizado" (BURDEK, 2006, p. 16).

Para o presente trabalho foi adotada a definição de metodologia como "a área do campo das ciências, relacionada à teoria do conhecimento, que se dedica ao estudo (criação, análise ou descrição) de qualquer método científico" (COELHO 2006 p.17) e método como "caminho para se chegar a um fim; conjunto de ações com as quais se pretende atingir um objetivo; programa que regula previamente e controla a execução de uma série de operações que devem ser realizadas de modo a evitar erros" (COELHO 2006 p.17).

No sentido acadêmico, as bases metodológicas do design são, em sua grande maioria proveniente da década de 60, época em que surgiram os primeiros cursos voltados para o ensino e prática do design. Durante os anos, ocorreram

várias revisões e adaptações das metodologias, algumas se destacando e tornando-se praticamente literatura base da maioria dos cursos de design até a atualidade.

Mesmo com revisões pontuais, a velocidade dessas atualizações, pesquisas e até o surgimento de novas metodologias é muito desproporcional se comparado com a velocidade com que se deram as mudanças nas vidas das pessoas, seja na democratização da informação ou do novo cenário econômico, essas mudanças influem em muito no ato de projetar produtos.

O presente trabalho se fundamentou nas principais metodologias de *design* lecionadas e propõe um modelo flexível de metodologia, buscando sanar os principais problemas encontrados no desenvolvimento de projetos de *design* atualmente no sentido tecnológico, metódico e econômico. Para tal, buscou-se aprimorar o modelo adaptando conceitos e práticas com o desempenho atestado em outras áreas do conhecimento como o desenvolvimento de softwares, controle de qualidade e marketing.

1.1 JUSTIFICATIVA

A revolução industrial foi um período de grande mudança, foi um marco, não só tecnologicamente, mas uma mudança de muitos paradigmas no cotidiano das pessoas. A vida se tornou mais ágil dentro das indústrias, como denunciava Chaplin, mas as mudanças ocorreram também fora delas. Com o decorrer dos anos a corrida pela economia de tempo tomou vários caminhos, um deles foi a redução e o aprimoramento dos processos como propôs Frederick W. Taylor, obtendo um impacto gigante nos processos produtivos do século passado (SANTOS, 2013). Nos dias de hoje o tema tomou tal importância que é possível encontrar centenas de publicações sobre gerenciamento do tempo para os mais diversos segmentos.

Nos últimos 50 anos pode-se observar mudanças drásticas no modo de vida das pessoas, com a popularização da *internet* e o aumento dos recursos tecnológicos. As distâncias diminuíram e a impressão é que os dias também. É necessário se adaptar a essa realidade de forma muito veloz, a mesma velocidade de mudança se tornou mais complicada em alguns campos como educação, cultura e no caso aqui abordado, na metodologia de Design. Os métodos conhecidos e mais utilizados além de terem sido pensados em outra época são onerosos de mais no que tange tempo e burocracia de documentações, e pouco falam ou não falam sobre gerenciamento de equipes trabalhando em um projeto de design, tornando o trabalho em grupo mais complicado.

O projeto a seguir busca referências em áreas do conhecimento, que assim como o design sofrem grandes pressões em relação ao tempo de execução e qualidade da entrega. Não buscou-se apenas reduzir tempo dos processos, mas sim utilizá-lo de forma mais inteligente, preenchendo as lacunas que a metodologia de design possui e propondo uma abordagem metodológica mais ágil. Para tal, o trabalho apresenta uma pesquisa sobre as metodologias de *Design*, um levantamento teórico sobre desenvolvimento ágil, alguns pontos pertinentes sobre qualidade e melhora de desempenho e um modelo de análise de público alvo mais atual e dinâmica.

Após a análise, parte-se para a estruturação do conteúdo a fim de se gerar um modelo metodológico a ser apresentado e exemplificado em um projeto real.

Nessa etapa, cada ciclo do processo será apresentado e aplicado para maior entendimento e para testar sua real eficácia.

1.2 OBJETIVOS

1.2.1 Objetivo geral

Esse projeto visa propor um *framework* ágil, adaptativo às necessidades e peculiaridades dos projetos de design, buscando a qualidade na entrega, versatilidade na aplicação, economia e controle do tempo.

1.2.2 Objetivos específicos

Para a realização do projeto, as seguintes etapas se fazem necessárias:

- Levantar metodologias de projeto de design de produto e suas etapas.
- Realizar uma pesquisa sobre desenvolvimento Ágil e suas aplicações.
- Pesquisar sobre sistemas de controle de qualidade.
- Desenvolvimento de uma metodologia baseada em processos de Design e desenvolvimento
- Avaliar o modelo desenvolvido e apresenta-lo de forma didática.

1.3 Estrutura do Trabalho

O presente trabalho foi subdividido em quatro capítulos. O primeiro apresenta os elementos de apresentação divididos em Introdução, Justificativa e Objetivos. O segundo capítulo é composto pelo levantamento teórico e análise de dados a cerca do problema, dando subsídios para a próxima fase. No terceiro capítulo é apresentado o modelo proposto, são justificadas as escolhas na elaboração do mesmo e realiza-se uma apresentação didática das etapas. O quarto e último capítulo é composto pelas Considerações Finais e as Referências Bibliográficas.

2. PROJETO

2.1 PROCEDIMENTOS METODOLÓGICOS

Os procedimentos metodológicos aplicados ao projeto foram escolhidos a partir da análise das etapas e a melhor da adaptação ao tipo de projeto realizado. A metodologia empregada foi baseada as fases do processo de design descritas por LÖBACH (2001, p. 142).

A pesquisa partirá do levantamento bibliográfico da origem e das principais metodologias de design e posteriormente de outra pesquisa bibliográfica, que será realizada sobre as metodologias Ágeis.

A etapa seguinte trata-se do início do projeto do *framework* e a elaboração de uma versão de teste, verificando assim o funcionamento do sistema e a necessidade de correções, possibilitando a implementação da ferramenta de forma completa.

Após a realização de todas as etapas anteriores serão compilados os resultados, finalizando-se a monografia e apresentação os resultados para a banca.

2.2 PREPARAÇÃO

2.2.1 Pesquisa sobre metodologias de Design

Antes de começar a discutir sobre a metodologia aplicada ao campo do design, é necessário entender o conceito global que a metodologia trata e sua importância para o desenvolvimento do design, não só como um campo do conhecimento, mas também como algo possível de se ensinar.

Realizando uma rápida pesquisa é possível notar que são muitas as definições sobre a metodologia do design e metodologia projetual. Para Bomfim (1995 apud VASCONCELOS, 2010, p.3) “a Metodologia é o estudo dos métodos,

técnicas e ferramentas e de suas aplicações à definição, organização e solução de problemas teóricos e práticos”.

Segundo Munari (1983, p.20) “O método projetual não é mais do que uma série de operações necessárias, dispostas em ordem lógica, ditada pela experiência. O seu objetivo é o de atingir o melhor resultado com o menor esforço”.

A partir dessas definições podemos concluir que a metodologia de design, consiste no estudo dos métodos com o objetivo de ordena-los de forma a atingir um determinado objetivo da melhor maneira possível.

O início da metodologia do design se entrelaça com o princípio do ensino acadêmico do design, esses dois acontecimentos possuem várias causas em comum, entre essas causas, talvez a mais relevante seja o contexto cultural e econômico em que a Europa se encontrava.

Após a segunda guerra mundial o cenário econômico europeu era marcado pela grande concorrência entre as indústrias em terrenos internacionais o que hoje habitualmente é conhecido como “globalização” (BURDEK, 2011).

A briga por espaço no mercado fez com que as indústrias buscassem um diferencial competitivo para os seus produtos, o que culminou na abertura do mercado industrial para os designers. Contudo, para se adaptar ao método racionalista das indústrias, o design precisou recorrer ao embasamento científico, para ser aceito como sério parceiro de dialogo frente às empresas (BURDEK, 2011).

A HfG Ulm (*Hochschule für Gestaltung Ulm* ou Escola de Design de Ulm) teve um papel extremamente importante nesse processo, foi um dos primeiros centros de formação de designers para o mercado. Um dos primeiros metodólogos, considerado um dos pais da metodologia de design, Christopher Alexander (1964 apud BURDEK, 2011, p. 251) apresentou quatro motivos para a metodologia ser aplicada na pratica projetual:

1. Os problemas de projeto se tornaram por demais complexos para que sejam tratados de forma intuitiva.
2. A quantidade de informações necessárias para a resolução de problemas de projeto elevou-se de tal forma que o designer por si só não consegue coletar nem manipular.
3. A quantidade de problemas de projeto aumentou rapidamente.

4. A espécie de problemas de projeto, comparada com épocas anteriores vem se modificando em um ritmo acelerado, de forma que se torna cada vez mais raro poder se valer de experiências anteriores.

As primeiras pesquisas sobre metodologia de projeto foram impulsionadas pelas necessidades de projeto no campo aeroespacial provindo do norte da Europa. Esses projetos possuíam problemas de solução extremamente complexa. Essas primeiras pesquisas sobre metodologia levaram o nome de Sistemas de Primeira Geração, que seguiam a idéia de que o projeto poderia ser dividido em etapas bem definidas, listadas a baixo como visto em (BURDEK, 2011, p. 252):

1. Compreenda o seu problema.
2. Colete informações sobre a situação atual, limitações técnicas, etc.
3. Analise as informações coletadas.
4. Desenvolva alternativas.
5. Avalie as alternativas, essa é a etapa onde acontece a construção de protótipos e as simulações.
6. Teste e implementação.

Vários outros autores trataram de pesquisar e elaboraram suas próprias metodologias, cada uma com suas particularidades e enfoques em pontos específicos entre eles estão Bruce Archer, Morris Assimov, Bernd Löbach, Christopher Jones, Bernhard Burdek entre outros que serão mencionados a seguir.

Um dos pontos mais marcantes da escola de Ulm no que tange o desenvolvimento de metodologias de design era o afastamento entre a prática projetual e a concepção artística, aproximando o design do cientificismo. Isso podia ser notado pela grade curricular dos cursos da escola de Ulm, que contemplava várias disciplinas voltas ao racionalismo e cientificismo e muitas outras eram testadas periodicamente e também pela ausência das matérias voltadas a práticas artísticas se diferenciando das *Werkkunstschule* (escolas profissionais de arte) o que para Bonsiepe (1983) fazia sentido visto que, na sua visão, desenho industrial possui ligações com o mundo da estética, mas não se pode dizer o mesmo sobre o mundo da arte. Em contrapartida, afirma também que o design não é e nunca será uma atividade científica, o que não impede que algumas iterações entre essas áreas do conhecimento sejam frutíferas (BONSIEPE, 1983).

A intenção era formar alunos capazes de seguir as práticas do mercado, os primeiros pesquisadores da metodologia projetual foram Tomás Maldonado e Gui

Bonsiepe (BURDEK, 2011) esse sendo um dos responsáveis pela disseminação do design na América Latina.

“Os partidários dessa nova disciplina constataram o relativo subdesenvolvimento e atraso da atividade projetual, quando comparadas a outras atividades humanas, sobretudo as técnico-científicas” (BONSIEPE, 1983, p. 52). Os primeiros metodólogos notaram que deveriam seguir uma estrutura racionalista para se compatibilizar com a visão das indústrias. A intenção era de criar uma ordem lógica de passos para formar a estrutura da atividade projetual. O objetivo da pesquisa era um método único para a prática projetual, porém não se levou em conta que problemas diferentes necessitam de métodos e abordagens diferentes.

A metodologia então possuía uma posição absolutista: a estrutura dos processos poderiam sofrer pequenas variações, mas em essência as etapas deveriam ser as mesmas não importa o projeto a ser realizado.

O problema é que a maioria das metodologias não levava em consideração pontos importantes, como a contaminação ambiental, o uso de recursos não renováveis, o consumo de energia durante o ciclo de produção e o descarte ao fim da vida útil (BONSIEPE, 1983, p. 52). Essas questões não eram levadas na época do surgimento das metodologias de design. Bonsiepe propôs algumas mudanças para as metodologias projetuais, levantando alguns pontos não contemplados como:

1. Complexidade do problema projetual: Os métodos devem ser ajustados, dependendo do nível de complexidade do projeto.
2. Disponibilidade de recursos tecnológicos: A metodologia deve se adaptar a tecnologia e aos processos que se tem disponíveis para não gerar uma solução inviável.
3. Os objetivos políticos-econômicos. Mercados com práticas diferentes precisam de métodos variados.
4. Tipo de problema projetual.

Outro ponto importante em que as metodologias se omitem é a parte de criação estética, um dos alicerces desta pesquisa. Para Bonsiepe (1983) as maiores falhas da metodologia eram sua omissão frente ao tratamento dos detalhes formais de um produto e o pouco aprofundamento em relação a criação visual do design.

Ao passar dos anos, os métodos e metodologias foram sendo desenvolvidas e contestadas. As novas tendências foram influenciando na maneira de se projetar as coisas e também na noção de conhecimento (BURDEK, 2011, p. 256). Na

década de noventa várias outras correntes influenciaram a maneira de fazer design, com a chegada do conceito de Gestão do Design a profissão foi levada para um âmbito estratégico mudando a forma de pensar o projeto. Essas mudanças fazem com que, segundo Burdek (2011) o projeto passe a ser pensado de “dentro para fora”, ou seja, a partir da observação do mercado e das necessidades de um público específico é possível concluir se um produto novo possui aderência de mercado e como ele deve ser para alcançar um determinado objetivo.

2.2.2 Principais nomes da metodologia

Quando se pesquisa metodologia de design, alguns nomes aparecem de forma recorrente. A seguir será apresentado um breve resumo das metodologias propostas pelos principais teóricos das graduações de design.

A escolha dos metodólogos se deu por meio da revisão da grade curricular do curso de bacharelado em design da Universidade Tecnológica Federal do Paraná e da observação dos livros que compõem a literatura básica do curso, principalmente nas matérias de projeto e metodologias.

2.2.2.1 Bruno Munari

Para Munari (1998, p. 10), “O método projetual não é nada mais do que uma série de operações necessárias dispostas em ordem lógica, ditada pela experiência. O seu objetivo é o de se atingir o melhor resultado com o menor esforço”. Ele acredita que o método projetual, diferente do que muitos pensam, não robotiza o projeto, mas, assim como uma receita culinária, dita os passos numa ordem lógica que se seguida, aumenta a possibilidade de um resultado satisfatório.

Ainda afirma que o método não limita a criatividade, pelo contrário, abre a possibilidade de entender o problema mais a fundo, abrindo um leque maior de soluções.

O método proposto por ele consiste em várias etapas ordenadas de forma linear. O mais peculiar em relação a outros métodos é a geração de alternativas,

onde se realizam os desenhos do projeto, só ocorre após várias etapas de análise de dados, sendo que o desenho construtivo se encontra no fim do processo, em contramão a outros métodos em que essa etapa se encontra mais próxima do início. O diagrama da figura 1 apresenta as etapas da metodologia proposta por Munari:

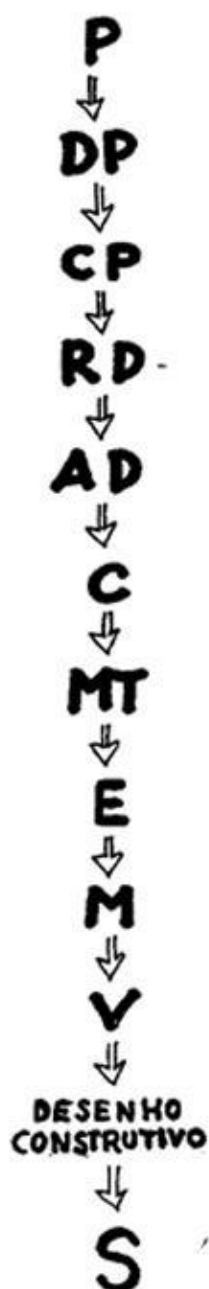


Figura 1 – Etapas da metodologia segundo Munari
Fonte: Munari (1998, p.55).

Onde:

P = Problema a ser resolvido.

DP = Definição do problema: qual(is) aspecto(s) do problema se deseja solucionar?

CP = Componentes do problema: estuda os detalhes do problema, assim como outros fatores que influenciam para a solução do problema.

RD = Coleta de dados: onde se identifica materiais que podem ser utilizados assim como tecnologias disponíveis para se projetar uma solução viável.

AD = Análise dos Dados, é identifica-se os componentes do problema, procurando soluções comuns para os muitos aspectos de um problema.

C = Criatividade: levando em conta todos os dados coletados, busca-se as alternativas para a solução proposta.

MT = Materiais e Técnicas: estudo dos materiais que se tem disponível, e do ferramental.

E = Experimentação: busca de outras possibilidades produtivas para baratear o processo ou possa trazer outra vantagem.

M = Modelo: desenvolvimento do mockup, estudo dimensional e de detalhes construtivos.

V = Verificação: a proposta resolve o problema a que se propõe? Se não, deve-se realizar as adaptações necessárias.

DC = Desenho construtivo: como devo montar o projeto? Trata também da configuração visual do projeto.

S = Solução: completas todas as etapas, o projeto foi completo e o problema solucionado.

2.2.2.2 Bernd Löbach

A metodologia aplicada por Löbach dá ênfase nas análises prévias a cerca do problema e também evidencia a parte criativa, embora não mencione muito a fundo os métodos de criação.

Ele divide o processo em quatro etapas (exemplificado na figura 2): Análise do Problema, Geração de Alternativas, Avaliação das Alternativas e Realização da Solução do projeto. As etapas são lineares, porém, como o próprio autor cita nada

impede que as etapas possam acontecer em paralelo, e nem que haja regressões durante o processo (LÖBACH, 2001, p. 141).



Figura 2 – Etapas da metodologia segundo Löbach
Fonte: Autoria própria (2014)

Cada uma das quatro etapas possui subetapas que se fazem necessárias ou não dependendo da complexidade do projeto, elas são:

1. Análise do problema.

- a) Conhecimento do problema. “A descoberta de um problema constitui o ponto de partida e motivação para o processo de design, que depois se define melhor no seu desenrolar dependendo do tipo de problema” (LÖBACH, 2011, p. 143).
- b) Coleta de informações. Devem-se coletar todos os dados possíveis a cerca de um problema.
- c) Análise das informações. Compreende um conjunto de análises que devem ser feitas durante o projeto. Entre elas estão: quantas pessoas estariam interessadas no produto, a relação social pessoa/projeto, relação entre produto e ambiente (durante e depois do ciclo de vida), desenvolvimento histórico do produto, análise de mercado, análise comparativa, análise estrutural, análise funcional e análise de configuração formal (aparência).
- d) Definição do problema. Analisando todos os dados recolhidos anteriormente pode se ter uma visão geral do problema.

2. Geração de alternativas.

É o momento em que se deve buscar o maior numero de soluções possíveis. Para LÖBACH existem dois caminhos para se elaborar soluções para um problema durante a etapa de geração de alternativas: aguardar a inspiração ou tentativa e erro

(LÖBACH, 2001, p. 151). Em seguida ele acusa que estes caminhos são usualmente seguidos por artistas, mas não são indicados para designers, que devem ir por um viés menos aleatório para solucionar um problema (LÖBACH, 2001, p. 153). Ele sugere que uma boa prática quando se trata de geração de alternativas é aguçar a criatividade por meio da livre associação de ideias que permite chegar a um leque maior de soluções, todavia ele não se aprofunda muito nos detalhes desse método.

3. Avaliação das alternativas.

Com base nos critérios estabelecidos a partir das análises da primeira etapa, comparam-se as alternativas geradas e então se escolhe a que melhor supra os componentes do problema.

4. Realização da solução do projeto.

A última etapa compete a finalização do projeto, sendo a criação do modelo, a revisão e correção de falhas e ao se alcançar o resultado desejado do protótipo elaboram-se os desenhos de produção e instruções de montagem.

2.2.2.3 Mike Baxter

A metodologia projetual proposta por Baxter possui uma grande estrutura extremamente detalhada, dividida em grandes etapas, cada uma possuindo uma lógica interna com vários métodos. Ele diz que a atividade projetual “não segue uma linha reta, é marcada por avanços e retornos” (BAXTER, 2000, p. 36). O desenrolar do projeto depende muito das informações encontradas. Cada decisão tomada inclui nos próximos passos novas informações, que podem fazer com que se retroceda no projeto para repensar algum aspecto a partir de um novo ponto de vista que se julgue importante ou a adição de uma nova etapa não planejada.

As grandes etapas são:

1. Oportunidade de negócio.
2. Especificação do projeto.
3. Projeto Conceitual.
4. Projeto da configuração.
5. Projeto Detalhado.

6. Projeto para Fabricação.

1. Oportunidade de negócio:

Antecede o desenvolvimento do projeto. A ideia não é apenas escolher entre desenvolver um produto ou outro, mas selecionar a melhor oportunidade possível de inovação. Isso é relacionado diretamente com o plano de inovação da empresa. Por exemplo, a empresa deseja incluir produtos mais econômicos no mercado? Deslocar-se para um mercado mais sofisticado? Redesenhar os produtos já existentes? (BAXTER, 2000, p. 29).

2. Especificação do Projeto.

Aqui são decididos os pré-quesitos do projeto, o objetivo que se deseja alcançar com ele. “Nessa etapa são fixadas as metas técnicas para o novo produto, abrangendo desde as suas funções básicas, sua aparência, até as embalagens e a forma de embarque para as distribuidoras” (BAXTER, 2000, p. 39).

O conteúdo dessa etapa deve ser fruto do consenso entre os diversos setores de uma empresa.

O autor confere grande importância para as etapas preparatórias no desenvolvimento do projeto. Ele as justifica pelo argumento de que “o produto com uma especificação clara e precisa, antes de começar o desenvolvimento, tem três vezes mais chance de sucesso do que aquele cujo desenvolvimento é iniciado sem esse tipo de cuidado” (BAXTER, 2000, p. 167).

3. Projeto conceitual.

É o início do desenvolvimento do produto em si, em linhas gerais, deve mostrar como o produto deve suprir as necessidades requisitadas. Baxter (2000) diz que para se obter sucesso nessa etapa existem dois segredos simples: 1. Esforce-se para gerar o maior número de conceitos possível; 2. Escolha o melhor deles.

Baxter justifica esse argumento afirmando que criatividade é 99% transpiração e 1% inspiração. (BAXTER, 2000, p. 231)

4. Projeto da Configuração.

Listadas as necessidades, é necessário estudá-las a fundo para gerar suas configurações. Dificilmente se consegue agregar todas as

características necessárias de um produto em um só esboço, por isso elabora-se as configurações separadamente e depois uni-se as alternativas em uma configuração de produto. Ela começa com o conceito do produto e termina com o protótipo completamente desenvolvido e testado (BAXTER, 2000, p. 305).

5. Projeto detalhado.

O projeto detalhado parte dos resultados da Configuração. Se determina como o produto será produzido, quais os materiais e técnicas serão empregados na fabricação, se haverá necessidade de se terceirizar algum processo. Nesta etapa também é realizada a construção dos protótipos para a realização de testes.

6. Projeto para fabricação.

Terminada todas as fases e todos os detalhes necessários para a fabricação, disponíveis no projeto que deve ser encaminhado para a produção.

2.2.2.4 Fábio Mestriner

A metodologia apresentada por Mestriner (2005) é voltada para o desenvolvimento de embalagens. Em sua estrutura projetual ele lista dez pontos para um design realmente efetivo, esquematizado na figura 3.

- Conhecer o produto.
- Conhecer o consumidor
- Conhecer o mercado.
- Conhecer a concorrência.
- Conhecer tecnicamente a embalagem a ser projetada.
- Conhecer os objetivos mercadológicos.
- Ter uma estratégia de design.
- Desenhar de forma consciente.
- Trabalho integrado com a indústria.
- Revisar o projeto.



Figura 3 – Etapas da metodologia segundo Mestriner
Fonte: Autoria própria (2014)

Após a pesquisa e definidas as estruturas e os componentes de cada uma das metodologias é possível esquematizar através de um quadro um comparativo entre as principais metodologias. Utilizou-se a o modelo de classificação desenvolvido por Gui Bonsiepe (1984), onde o mesmo enquadrava as metodologias em quatro grupos: Lineares, quando não há possibilidade de retroceder nas etapas durante o desenvolvimento; com *feedback*, quando as etapas se desenvolvem em paralelo e são interdependentes; circulares quando a metodologia se desenvolve em ciclos e por fim o flexível onde o projeto pode percorrer caminhos diferentes buscando gerar variedade de produtos.

Teórico	Bruno Munari	Bernd Löbach	Mike Baxter	Fábio Mestriner
Área de Aplicação	Design Gráfico	Design de Produto	Design de Produto	Design de Embalagens
Etapas	<ul style="list-style-type: none"> - Problema - Definição do Problema - Componentes do Problema - Coleta de dados - Análise dos dados - Criatividade - Materiais e técnicas - Experimentação - Modelo - Verificação - Desenho Construtivo - Solução 	<ul style="list-style-type: none"> - Análise do problema - Geração de alternativas - Avaliação das alternativas - Realização da solução do projeto 	<ul style="list-style-type: none"> - Oportunidade de negócio. - Especificação do projeto. - Projeto Conceitual. - Projeto da configuração. - Projeto Detalhado. - Projeto para Fabricação. 	<ul style="list-style-type: none"> - Briefing - Estudo de campo - Estratégia de design - Desenho - Implantação
Tipo	Linear	Feedback	Feedback	Feedback

Quadro 1 - Comparativo das metodologias de design
Fonte: Autoria própria (2015).

2.2.3 Desenvolvimento Ágil

Analisando as metodologias listadas na etapa anterior, uma das coisas que pode ser observada é o tempo utilizado no desenvolvimento de um projeto. A cada dia é necessário se fazer entregas com mais agilidade e com maior precisão. É possível observar isso de forma mais evidente na velocidade em que aplicativos e *smartphones* são colocados no mercado. Em uma primeira observação é possível notar uma diferença entre a realidade atual do mercado e da abordagem das metodologias de design. Indo mais a fundo nos métodos de gerenciamento e desenvolvimento de aplicativos, encontra-se uma estrutura muito peculiar de trabalho onde as entregas são feitas continuamente e atualizações são lançadas com frequência, a esse tipo de metodologias se denomina desenvolvimento Ágil.

O desenvolvimento Ágil é um conceito relativamente novo se comparado com outras metodologias de desenvolvimento. Surgiu em pequenas manifestações na década de 90, culminando na reunião de vários programadores de *software* que criticavam a metodologia tradicional e em 2001 se uniram buscando uma solução e escreveram o manifesto Ágil, definindo um marco para o movimento.

Manifesto Ágil

“Estamos descobrindo maneiras melhores de desenvolver software, fazendo-o nós mesmos e ajudando outros a fazerem o mesmo. Através deste trabalho, passamos a valorizar:
Indivíduos e interações mais que processos e ferramentas
Software em funcionamento mais que documentação abrangente
Colaboração com o cliente mais que negociação de contratos
Responder a mudanças mais que seguir um plano
Ou seja, mesmo havendo valor nos itens à direita, valorizamos mais os itens à esquerda”. (MANIFESTO PARA DESENVOLVIMENTO ÁGIL DE SOFTWARE, 2001).

As metodologias de desenvolvimento Ágil já existiam antes da criação do manifesto. Entre elas podemos destacar o *Extreme Programming* (XP) e o *Scrum*. Esta segunda será alvo de nossa pesquisa.

2.2.3.1 Sobre o Scrum

Oficialmente, o primeiro time de desenvolvimento *Scrum* foi criado por Jeff Sutherland em 1993, a sua primeira aplicação foi relacionada ao desenvolvimento de *softwares* na Easel Corporation. Alguns anos depois Jeff, em parceria com Ken Schwaber, que na época pesquisava metodologias de otimização de tempo em projetos desenvolveram a primeira definição clara de *Scrum* e apresentaram na OOPSLA (*Object-Oriented Programming, Systems, Languages & Applications*) no ano de 1995.

Jeff e Ken fizeram parte do time de desenvolvedores responsáveis por escrever o manifesto Ágil.

Uma definição clara de *Scrum*, adotada nessa pesquisa foi a definida por Rafael Sabbagh (2013, p. 17) que diz:

“*Scrum* é um *framework* Ágil, simples e leve, utilizado para a gestão do desenvolvimento de produtos complexos imersos em ambientes complexos. *Scrum* é embasado no empirismo e utiliza uma abordagem iterativa e incremental para entregar valor com frequência e, assim reduzir os riscos do projeto”.

A fim de entender melhor do que o *Scrum* trata, desmembrou-se os conceitos “*framework*” e “ágil” contidos nas seguintes definições:

Framework: Segundo SABBAGH (2013, p. 29), um “*framework* ou arcabouço é uma estrutura básica que pretende servir de suporte e guia para construção, a partir da expansão dessa própria estrutura, de algo com uso prático”. Disso, pode-se concluir que o *Scrum* é uma estrutura básica de desenvolvimento que sugere boas práticas com o objetivo de se alcançar algum resultado. Ainda, um *framework* não tem a função de apresentar etapas específicas, ele indica um programa onde os métodos devem ser adicionados ou retirados dependendo diretamente do projeto a ser desenvolvido.

Dentro desse contexto, Ágil refere-se ao movimento que surgiu na década de 90 e que veio a culminar no Manifesto para o desenvolvimento Ágil de *software*, escrito e assinado em 2001, por 17 programadores renomados coordenados por Kent Beck.

2.2.3.2 Conceitos de ambientes de desenvolvimento complexo

Para optar por uma metodologia ágil como base para o projeto do *framework* de design é necessário descobrir se existe semelhança no ambiente e contexto que os projetos de design e dos de programação se dão e se as necessidades que forçaram os metodólogos que criaram o *scrum* são parecidas com as que são enfrentadas nos projetos de design. Para tal, buscou-se entender o conceito de ambiente de desenvolvimento complexo e de produtos complexos. Produtos complexos imersos em ambientes complexos, segundo Holland (2006 apud SABBAGH, 2013) “são sistemas que envolvem um grande número de componentes, frequentemente chamados de “agentes” que se adaptam ou aprendem à medida que interagem”, algo frequentemente observado em projetos de design. Onde o projetista lida com várias informações e a partir destas combinações de dados concebe o projeto.

Para entender melhor o que é um ambiente de desenvolvimento complexo será utilizado outro *framework* chamado *Cynefin* desenvolvido por Dave Snowden que se caracteriza por um quadro esquemático para tomada de decisão em tarefas.

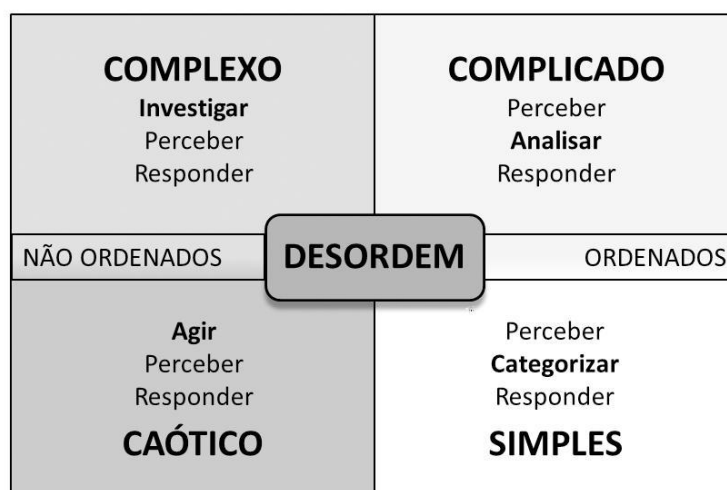


Figura 4 – Diagrama de *Cynefin*
Fonte: SABBAGH (2013, p. 32).

O *Cynefin* delimita os ambientes de desenvolvimento a partir da relação de causa e efeito e a possibilidade de tomada de decisão frente a uma ação. O modelo

apresenta cinco domínios: Ambiente Simples, Complicado, Complexo, Caótico e a Desordem.

Ambiente Simples:

Ambiente onde existe uma estabilidade entre as relações de causa e efeito. As situações são claras, previsíveis e repetíveis.

Nesses ambientes deve-se:

- Avaliar seus fatos (Perceber);
- Para definir que tipo de situação se trata (Categorizar);
- E então aplicar a melhora prática conhecida (Responder).

Um exemplo de ambiente simples é trocar uma tubulação entupida. Já existe um conhecimento prévio para o encanador, ele só precisa atestar e realizar o reparo.

Complicado:

Causa e efeito possuem relação, porém ela não pode ser entendida por qualquer pessoa. Existem várias respostas viáveis e é necessária a ajuda de um especialista.

Nesses ambientes deve-se:

- Avaliar seus fatos (Perceber);
- Para definir qual solução é mais adequada (Analisar);
- E então aplicá-la (Responder).

Exemplo: O concerto de um eletrodomestico, se pode até conhecer o defeito, mas só um especialista pode analisar as causas e aplicar uma das possíveis soluções.

Complexo:

Causa e efeito só são evidentes após testes, mediante isso é preciso uma abordagem empírica e a análise dos resultados alcançados e é necessário aceitar a possibilidade de falhas.

Nesses ambientes deve-se:

- Investigar por meio de experimentos (Investigar);
- Para entender seus impactos (Perceber);
- E então adotar a abordagem que traga resultados satisfatórios e descontinuar as que não trazem o retorno esperado (Responder).

Exemplo: Desenvolvimento de produtos e de *softwares*, onde é necessária a prototipagem e se realizam de alterações no projeto mediante as respostas obtidas e testes.

Caótico:

É o domínio das novas práticas, não é possível determinar causa e efeito uma vez que não existe conhecimento prévio ou padrões estabelecidos.

Nesses ambientes deve-se:

- Agir rapidamente buscando estabelecer uma ordem (AGIR);
- Perceber onde é possível estabelecer padrões estáveis (PERCEBER);
- Trabalhar os padrões para trazer a situação do caos para a complexidade (RESPONDER);

Exemplo: Situações de crise, sequestros, projetos de pesquisa.

Desordem:

Não existe qualquer conhecimento sobre causa e efeito, o problema deve ser fragmentado em partes menores e enquadrado em uma das quatro possibilidades de ambiente.

Seguindo esse modelo o domínio que mais se parece com as situações enfrentadas no desenvolvimento de projetos de design é o sistema complexo que por sua vez é o domínio em que o uso do *Scrum* é indicado. Segundo SABBAGH (2001, p. 33) “Nos contextos complexos, aplicam-se a experimentação, a inovação, a criatividade e a emergência de novas formas de se trabalhar”.

Por que o usar o *Scrum* como base para o desenvolvimento de projetos de Design? Como já dito anteriormente, as metodologias de Design ensinadas na academia, ainda remetem ou são quase integralmente as mesmas praticadas desde o início do ensino do Design na década de 60. Isso, somado com o atual cenário do mercado de Design, que em muito pouco se parece com o daquela época faz se necessária à adoção de uma metodologia de resposta mais rápida, que consiga ter um acompanhamento mais de perto pelo usuário permitido a prototipação frequente durante o projeto e que perca menos tempo no pré-projeto em análises e documentações, fato recorrente das principais metodologias de Design.

Entre os benefícios do *Scrum* observados por SABBAGH (2001) estão:

1. Redução dos riscos no projeto:

Os ciclos e as entregas frequentes geram ações corretivas mais rápidas, reforçados pela priorização das demandas onde o problema principal é resolvido já no início do projeto.

2. Maior qualidade no produto gerado:

Realização de testes é frequente e as reuniões de *feedback*, tanto em relação as entregas quanto ao processo compete em maior qualidade para o projeto.

3. Mudanças utilizadas como vantagem competitiva:

Mudanças são encaradas como oportunidades.

4. Visibilidade do progresso do projeto:

O *Scrum* possui várias ferramentas de acompanhamento do progresso e reuniões frequentes de alinhamento da equipe.

5. Redução do desperdício:

Método de trabalho orientando para o essencial, voltado para a simplicidade e funcionalidade.

6. Planejar apenas com o nível de detalhe possível:

Perde-se pouco tempo analisando quesitos que só serão utilizados ao termino do projeto. Quanto mais longe se estiver olhando, mais incorreta será a observação (SABBAGH, 2013).

7. Utilizar apenas os artefatos necessários e suficientes:

O *Scrum* utiliza de poucos artefatos não limitando a equipe com excesso de burocracia e buscando representar sempre que possível os monitoramentos em meio físico.

2.2.3.3 A estrutura do Scrum

A estrutura do *Scrum* é relativamente simples e a sua necessidade de operação, referindo-se a equipe se resume em apenas três papéis: o *Product Owner*, o *Scrum Master* e a Equipe de desenvolvimento.

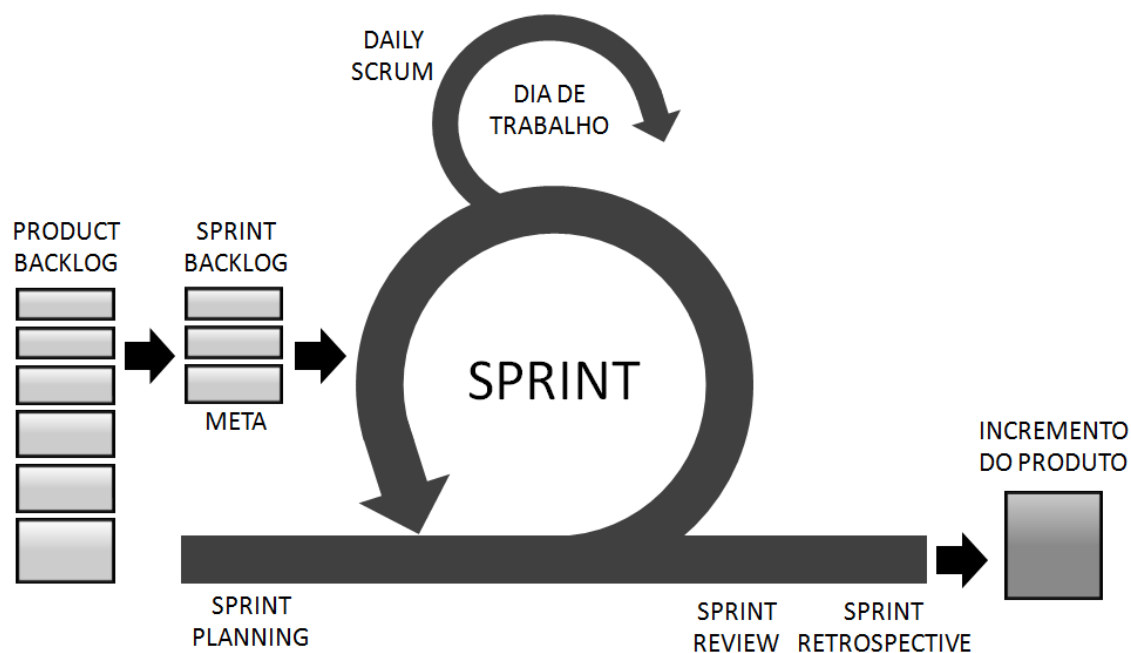


Figura 5 – Exemplo ciclo sprint
Fonte: Sabbagh (2013, p.42).

Product Owner. Responsável por fazer a interface entre o cliente, a equipe de desenvolvimento, os usuários e qualquer outra pessoa que possa influenciar no decorrer do projeto. O *Product Owner* ainda possui a salva-guarda da Visão do Produto, é ele quem entre outras atividades cria o plano de evolução do produto conhecido como *Roadmap*.

Scrum Master. Está presente no dia-a-dia do projeto e procura sempre facilitar o trabalho da equipe de desenvolvimento, ensinar a equipe sobre as práticas do *Scrum* e mediar às reuniões. É responsável pela gestão de riscos do projeto eliminando os empecilhos que venham a atrasar o projeto.

Equipe de Desenvolvimento: É a equipe responsável por transformar em realidade tudo que foi planejado. Os membros devem possuir todos os conhecimentos necessários para realizar as demandas. Um ponto importante sobre o *Scrum* é que a equipe de desenvolvimento tem plena autonomia de se organizar dentro do ciclo de trabalho para decidir como realizar as demandas.

Cronologicamente, o *Scrum* pode iniciar-se a partir de algum estímulo de natureza variada, como uma necessidade do cliente, oportunidade de negócio e etc.

Em seguida, o *Product Owner*, junto com o cliente elabora a Visão do Produto, que define, em linhas gerais, qual é o objetivo principal do projeto a ser seguido e constantemente visitado até o final do projeto.

Ainda o *Product Owner* é responsável por mais duas atividades de preparação que são de suma importância: a definição do *Product Backlog* e do *Roadmap*.

Product Backlog é uma lista ordenada dos pré-requisitos do projeto, nela estão os detalhes que o projeto deve atender, assim como as metas a serem alcançadas. O *Product Backlog* deve ser elaborado com a presença de todos os envolvidos, *Scrum Master*, Equipe de desenvolvimento, *Product Owner* e clientes. Ele é mutável e sofrerá inúmeras alterações durante o projeto, como a adição, subtração e repriorização de demandas.

Outro detalhe importante sobre o *Product Backlog* é a ordenação das demandas e o nível de detalhamento delas. Devido ao fato do *Scrum* tentar entregar a proposta de valor logo no primeiro ciclo de trabalho, as demandas são priorizadas a fim de entregar o sistema funcionando. Essas demandas essenciais ficam na parte de cima do *Product Backlog* e estão mais bem detalhadas. À medida que se desce no quadro de demandas, menos informação se tem sobre elas, pois serão executadas nos próximos ciclos, portanto não convém um detalhamento muito preciso, visto que elas possivelmente serão alteradas.

O *Roadmap* é o plano de evolução do produto. Ele é representado por meio de uma linha do tempo em que, metas e datas importantes estão representadas a fim de se ter um controle do tempo e das entregas pelo *Product Owner*, tornando mais fácil projetar prazos para os clientes e a própria equipe de desenvolvimento.

Feita essa preparação, o processo passa para a etapa de ciclos de desenvolvimento, que no *Scrum* é chamada de *Sprint*. Em linhas gerais, os *Sprints* são as sessões de trabalho que acontecessem sucessivamente até o término do projeto. Uma *Sprint* possui alguns eventos internos, todos eles têm seus tempos definidos pela equipe e devem ser rigorosamente respeitados. A duração máxima de um *Sprint* é usualmente chamada de *timebox*.

A estrutura básica de uma *Sprint* é definida pelos seguintes eventos: *Sprint Planning*, *Sprint*, *Daily Sprint*, *Sprint Review* e *Sprint Retrospective*.

Sprint Planning é a reunião de abertura de uma *Sprint*, onde é definido o conteúdo a ser desenvolvido, buscando priorização das demandas necessárias para

se alcançar a meta do *Sprint*. Orientado pelo *Product Backlog*, as demandas são ordenadas no *Sprint Backlog* (Figura 6), um painel de monitoramento de demandas da *Sprint*.

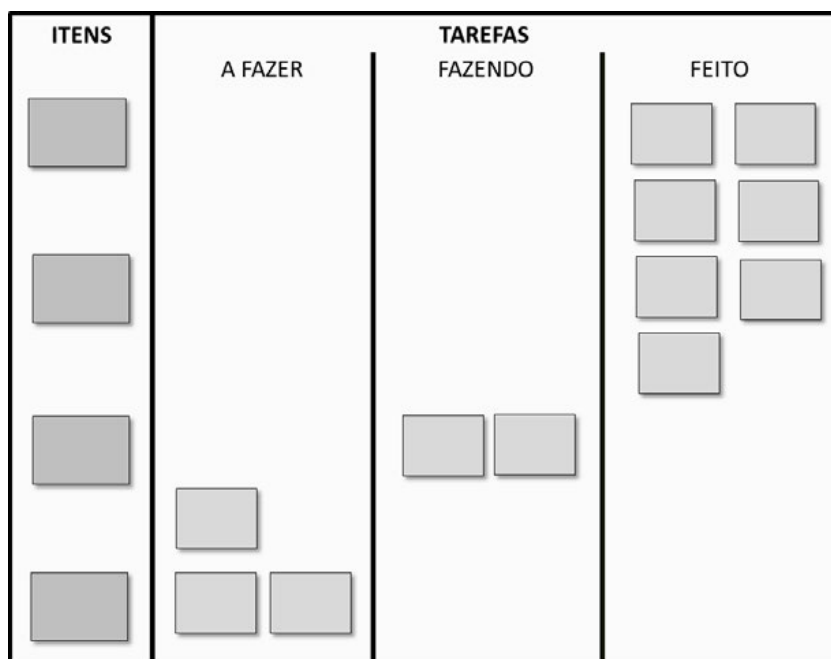


Figura 6 – Exemplo de *Sprint Backlog*
 Fonte: Sabbagh (2013, p. 44).

A *Daily Sprint* é a reunião diária da equipe de desenvolvimento. Nela são apresentadas de forma concisa o andamento das atividades da *Sprint*, os problemas encontrados e as próximas atividades. Essa reunião costuma acontecer no começo do dia e não deve ocupar muito tempo. Esse evento é uma oportunidade para a atualização do *Burnout Chart*, mais uma ferramenta do *Scrum* que ajuda no monitoramento do tempo das demandas. O *Burnout Chart* consiste em um diagrama onde em um eixo é representado o tempo de iteração definido para a *sprint* (decréscante em dias) e em outro eixo a quantidade de trabalho (soma do tempo previsto de conclusão das demandas) para a conclusão da *Sprint* (representado na figura 7). Essa ferramenta tem como função principal monitorar o rendimento da equipe e evitar eventuais atrasos nas entregas, permitindo localizar a causa do atraso e realizar a correção pelo *Scrum Master*.

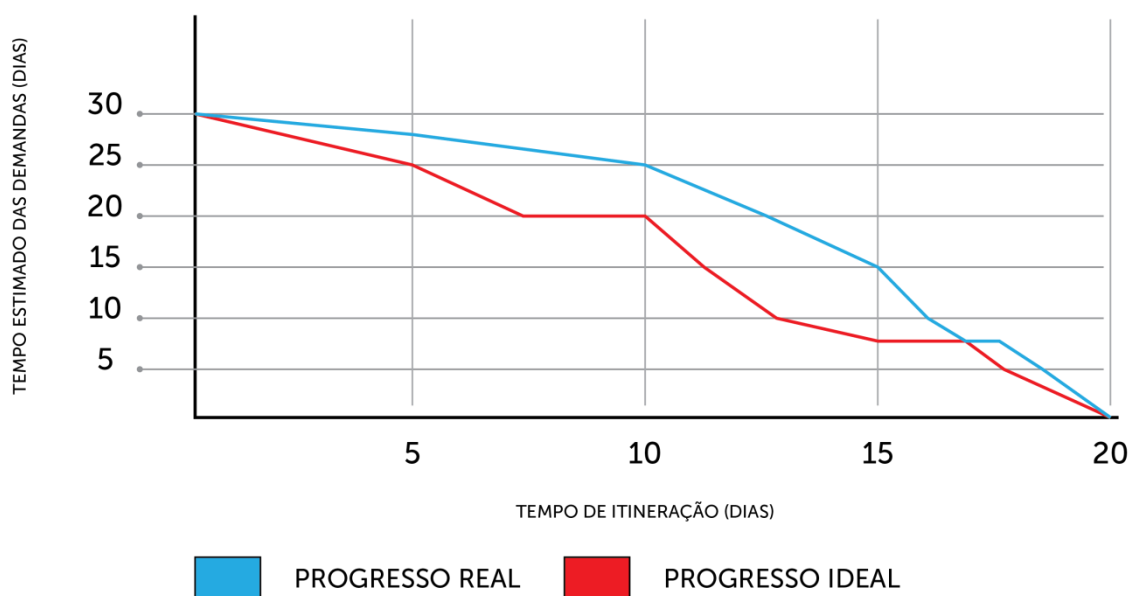


Figura 7 – Exemplo de *Burnout Chart*
 Fonte: Autoria própria (2015).

Ao término de cada *Sprint*, dois eventos acontecem: o *Sprint Review* e o *Sprint Retrospective*. No *Sprint Review* o *Product Owner* e a equipe apresentam ao cliente e aos usuários (se for o caso) o resultado obtido na *Sprint*. Esse é o momento de apresentar o projeto já com suas funções básicas (se possível) para receber os *feedbacks* e a partir deles definir como será e o que será desenvolvido na próxima *Sprint*, lembrando que isso possivelmente traga alterações ao *Product Backlog*. Já o *Sprint Retrospective* trata as boas práticas de trabalho, nessa reunião entre a equipe de desenvolvimento e o *Scrum Master* a equipe apresenta os pontos que deram certo na *Sprint* e os pontos onde eles tiveram problemas ou coisas que não surtiram efeito, esse conteúdo é analisado pela equipe e monta-se um plano de contingência para a próxima *Sprint*, buscando sempre melhorar o processo.

A *Sprint Retrospective* encerra a *Sprint*, nesse momento imediatamente outra *Sprint* deve ser iniciada, a menos que o projeto tenha sido concluído.

O *Scrum* se destaca por ser um sistema que entrega valor com frequência, isso minimiza a possibilidade de erros. As entregas são definidas como *Releases*, a data em que elas acontecem é definida em uma reunião de *Release Planning* entre

a equipe de desenvolvimento e o *Product Owner*. Outro ponto a ser definido é a quem o produto deve ser entregue, se a um grupo de usuários específico ou ao usuário final por exemplo. A ideia é que o produto chegue ao usuário final o mais rápido possível, mas quem deve definir isso é a equipe, o *Product Owner* e o cliente.

Ao término do projeto, espera-se que todos os pontos e metas tenham sido alcançados. Geralmente, os projetos de *Scrum* tem um tempo determinado para seu encerramento. Os projetos, normalmente necessitam de acompanhamento posterior ao lançamento, para pequenas melhorias e serviço de suporte. Para Sabbagh (2013), o *Scrum* não é uma ferramenta indicada para esse tipo de atividade, sendo mais indicada uma ferramenta de controle de demandas por *Tickets* como o *Kanban*¹ por exemplo.

O estudo da estrutura do *scrum* possibilitou o entendimento de como funciona um *framework*. O *scrum* é uma das metodologias de desenvolvimento mais utilizadas no mundo, devido a isso configura como uma base sólida para a elaboração da presente proposta desse trabalho. Porém o modelo necessita de algumas adaptações para que consiga ser utilizado em projetos de design com o maior aproveitamento possível. Buscando suprir essas lacunas, nas próximas etapas o projeto se aprofunda em outras áreas do conhecimento como gestão de qualidade e marketing.

2.2.4 Ferramentas de qualidade

Quando se trata de negócios, analisando o atual cenário de instabilidade é possível concluir que manter-se no mercado pode ser uma tarefa bem complicada. Um diferencial, buscado por grandes corporações e muitas vezes exibidos como troféus em forma de selos e certificações são as ferramentas de controle de qualidade. Cada vez mais difundidos, os conceitos de qualidade de produtos, serviços e processos ditam diretamente a sobrevivência ou não no mercado. Para

¹ *Kanban* em japonês significa “cartão”. Este nome surgiu em razão do sistema de controle visual dos estoques de materiais, pois frequentemente são utilizados cartões para representar os contentores cheios ou vazios, estes cartões são retirados ou colocados em um quadro à medida que o material é utilizado ou repostado (AGUIAR; PEINADO, 2007, p.137).

(CAMPOS, 2004, p. 9) “garantir a sobrevivência de uma empresa é cultivar uma equipe de pessoas que saiba montar e operar um sistema, que seja capaz de projetar um produto que conquiste a preferência do consumidor a um custo inferior ao de seu concorrente”. A isso que se propõem as ferramentas de controle de qualidade: criar condições que garantirão a empresa alcançar esses objetivos focando na qualidade.

Diante disso, não seria possível propor um sistema que visa melhorar a produtividade e qualidade dos processos e produtos sem citar e incorporar ferramentas utilizadas para esses fins, mas que são pouco difundidas no campo do design.

2.2.4.1 Utilizando os conceitos de Qualidade para entender o problema.

Geralmente, as metodologias de design se iniciam a partir da detecção de um problema e como lidar com ele. As metodologias falam com baixo nível de detalhamento sobre como definir bem um problema. Archer (1967 apud MUNARI, 1998, p.32) diz “É necessário, portanto, começar pela definição do problema, que servirá também para definir os limites dentro dos quais o projetista deverá trabalhar”. Uma ferramenta que pode ser extremamente útil quando incorporada a projetos de design é MASP, Metodologia de Análise e Solução de Problemas.

O MASP é um modelo de solução de problemas baseado na análise e controle de uma situação risco e é largamente utilizado em sistemas de gestão de qualidade.

Embora seja creditado ao Brasil, o MASP é originado dos sistemas de solução de problemas criados no Japão a partir da década de 50 baseados no *Qr Story* que, além do MASP deu origem a outros sistemas consagrados de qualidade, como o ciclo PDCA (GONÇALVES et al. 2011, p. 2).

Segundo Colengui (2007 apud ROVERI et al., 2012), “O MASP - Metodo de Análise e Solução de Problemas é um método prescritivo, racional, estruturado e sistemático para o desenvolvimento de um processo de melhoria num ambiente organizacional, visando solução de problemas e obtenção de resultados otimizados”.

Embora possa ser apresentado em outras configurações, o MASP geralmente é apresentado contendo sete etapas (figura 8), que são: Identificar, Observar, Analisar, Planejar, Verificar, Padronizar e Concluir.

PDCA	FLUXO-GRAMA	FASE	OBJETIVO
P	1 ↓	IDENTIFICAÇÃO	Definir claramente o problema e reconhecer sua importância.
	2 ↓	OBSERVAÇÃO	Investigar as características específicas do problema com uma visão ampla e sob vários pontos de vista.
	3 ↓	ANÁLISE	Descobrir, testar e confirmar a(s) causa(s) fundamental(is).
D	4 ↓	PLANO DE AÇÃO	Conceber um plano para bloquear a(s) causa(s) fundamental(is) eliminando seu(s) efeito(s).
C	5 OK? ↓	VERIFICAÇÃO	Verificar os resultados checando se o bloqueio foi efetivo.
A	6 ↓	PADRONIZAÇÃO	Prevenir contra o reaparecimento do problema.
	7	CONCLUSÃO	Recapitular todo o processo de solução do problema para trabalho futuro e descrever os resultados.

Figura 8 – Fluxo MASP.
Fonte: Rodrigues (2013).

1. Identificar:

Antes de tentar resolver um problema, ele deve ser o mais claro e bem definido possível. Essa etapa tem justamente essa função, por meio de *Brainstorming*, análise de dados e etc.

2. Observar:

Definido o problema, os dados coletados devem ser observados com cuidado buscando o maior número de informações possíveis sobre o problema. As

técnicas usualmente empregadas são o *Benchmarking*² e a Lista de verificação.

3. Analisar:

Essa etapa é o momento em que se realizam as análises do problema em questão, buscando as suas causas fundamentais. Uma das principais premissas do MASP é que, para a solução eficaz de um problema a causa raiz deve ser neutralizada. Para tal, utilizam-se as ferramentas como questionários e o diagrama de causa e efeito.

4. Planejar:

Momento da montagem do plano de ação, deve-se dar prioridade as ações que atinjam diretamente as principais causas do problema. Nessa etapa também se define as responsabilidades, prazos, custos, métodos de execução e indicadores de desempenho. Uma ferramenta útil nessa etapa é o 5W2H1S.

5. Agir:

Execução do plano.

6. Verificar:

É o momento em que se analisam os resultados obtidos, se foram satisfatórios ou não. Caso a ação não tenha alcançado o objetivo, retorna-se a primeira etapa para se tentar outra abordagem. Verifica-se o resultado através da análise dos dados do problema, da aplicação de pesquisas de satisfação e da verificação de indicadores.

7. Padronizar:

Se o objetivo foi alcançado, a fim de evitar a reincidência do problema é preciso buscar a padronização do processo. Ela se dá por meio da instrução de processo, pela elaboração de formulários, inclusão de rotinas e várias outras ferramentas de controle.

8. Concluir:

Ao término do processo de solução, os resultados obtidos devem ser documentados e o conhecimento gerado utilizado em novos projetos.

² *Benchmarking* é uma ferramenta de gestão que consiste na mensuração do desempenho de uma organização, permitindo que ela compare sua eficiência com a de outras organizações, frequentemente com a empresa líder do segmento ou outro concorrente muito relevante.

Baseando-se no MASP, é possível encontrar várias aplicações úteis com o propósito de definição de um problema. O principal objetivo das ferramentas de qualidade é encontrar a causa raiz de um problema e monitorá-la para que não haja reincidência.

3 VALIDAÇÃO DO PROJETO

3.1 O modelo proposto

Após todo o levantamento de dados, ferramentas de qualidade, estudo das metodologias de design e programação, chegou-se a apresentação da proposta de *framework*. Ele foi estruturado para ditar as etapas essenciais para alcançar a proposta do projeto: ser ágil, flexível e minimize falhas.

O modelo é dividido em três etapas principais: Análise, Planejamento e Desenvolvimento. Sendo que o Desenvolvimento comporta a produção, testes e correções.



Figura 9 – Etapas
Fonte: Autoria própria (2015).

Em Análise procura-se entender melhor o problema que se pretende solucionar e traçar os objetivos do projeto. O modelo tem um cuidado especial nessa parte, pois quanto mais bem definido o problema, melhor se desenrolam todas as outras etapas do projeto. Empregou-se os conceitos da gestão qualidade para analisar a causa raiz de um problema e o *brainstorming* para levantar a lista de influenciadores da situação problema que estamos tratando.

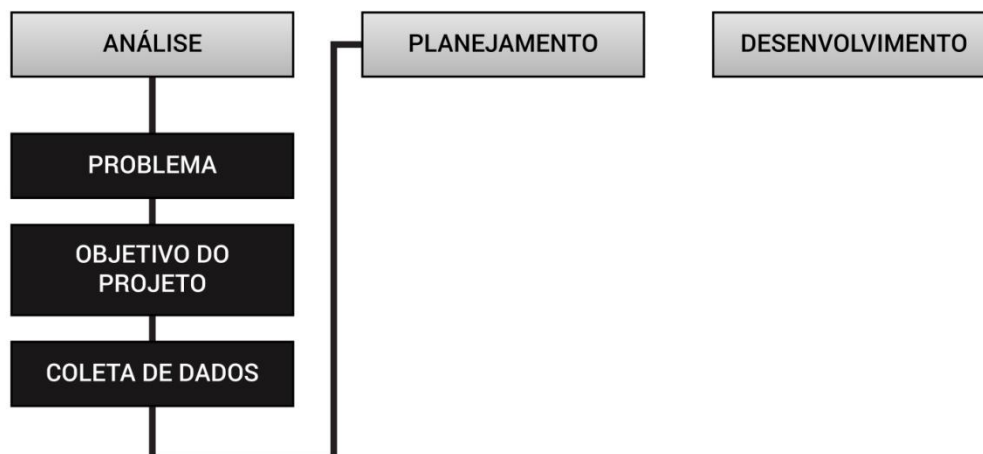


Figura 10 – Etapas 2
Fonte: Autoria própria (2015).

Munido das informações coletadas e analisadas, parte-se para a segunda etapa, o Planejamento. É aqui que serão utilizados os conhecimentos adquiridos para garantir o menor número de erros e retrabalhos possíveis e ainda sim garantir a participação do usuário e do cliente em toda a etapa de desenvolvimento, sem comprometer o prazo. Os pilares dessa etapa são a priorização da causa ou as causas do problema que se pretende solucionar. No exemplo foi utilizada uma matriz de “impacto X esforço”, mas a equipe pode escolher trabalhar com qualquer ferramenta de priorização que efetue de forma satisfatória essa etapa.

Priorizadas as demandas, segue-se para a criação do quadro de visualização e acompanhamento de demandas, que aqui chamado de *Product Dashboard*. Na escolha do nome buscou-se, manter os termos padronizados com os nomes populares no meio do gerenciamento de projeto, que são na sua grande maioria em inglês, e o termo *Dashboard* é usado justamente para esses fins: expor informações para uma equipe de trabalho de forma gráfica. Nesse caso, não só como um quadro de informação, o *Dashboard* é uma ferramenta de controle visual do andamento das atividades e de comunicação entre a equipe. Uma definição para *dashboard* é “é um painel de controle que fornece uma representação ilustrada do desempenho do negócio em todas as estruturas da organização” (PROJECTBUILDER, 2014).

Ainda tratando de planejamento, as próximas duas subetapas são primordiais para o controle dos ciclos de trabalho, que aqui manteremos nomeando-as como *Sprints*. As subetapas são a montagem do plano de ação e elaboração do *roadmap*, ambos estão diretamente ligados com a definição da duração dos ciclos de trabalho e do cronograma de entregas.

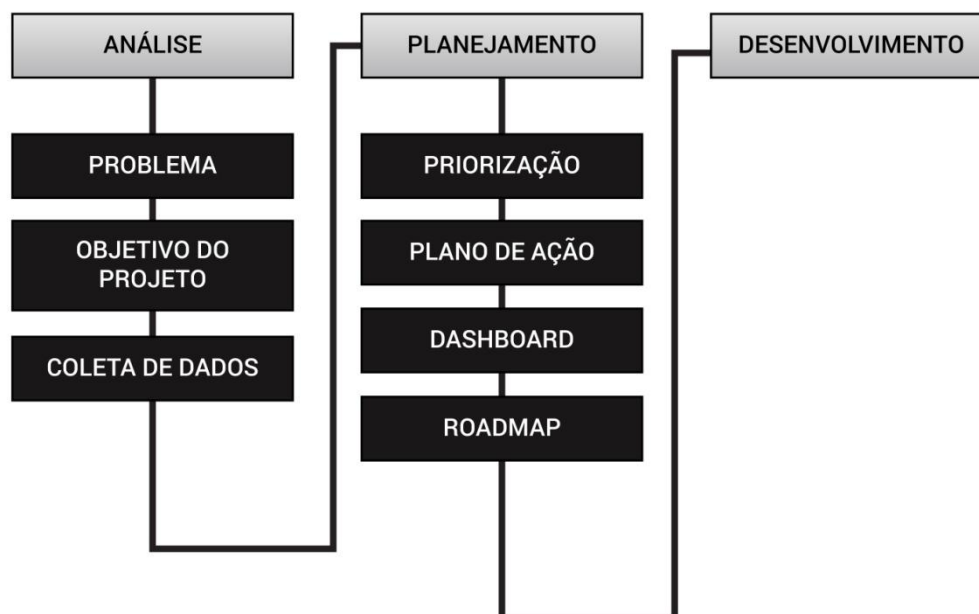


Figura 11 – Etapas 3
Fonte: Autoria própria (2015).

Por fim na última etapa é a execução do projeto, ou seja, as *Sprints*, testes e correções. Essa etapa, assim como no *Scrum* possui alguns eventos, mas que diferentes do *sprint* não são obrigatórias em caso de equipes reduzidas ou trabalhos sem equipe. As reuniões são: reuniões de planejamento, chamadas de *Planning Sprint* que tem a função de planejamento do ciclo de trabalho; reuniões diárias para acompanhar o andamento dos trabalhos e corrigir pequenas falhas, que será chamada de *Daily Sprint*; e as reuniões de encerramento de ciclo, *Sprint Review* e *Sprint Retrospective*. Serão mantidos os nomes das etapas do *sprint* semelhante à nomenclatura no *scrum*, pois elas já são empregadas com a mesma função em outras metodologias.

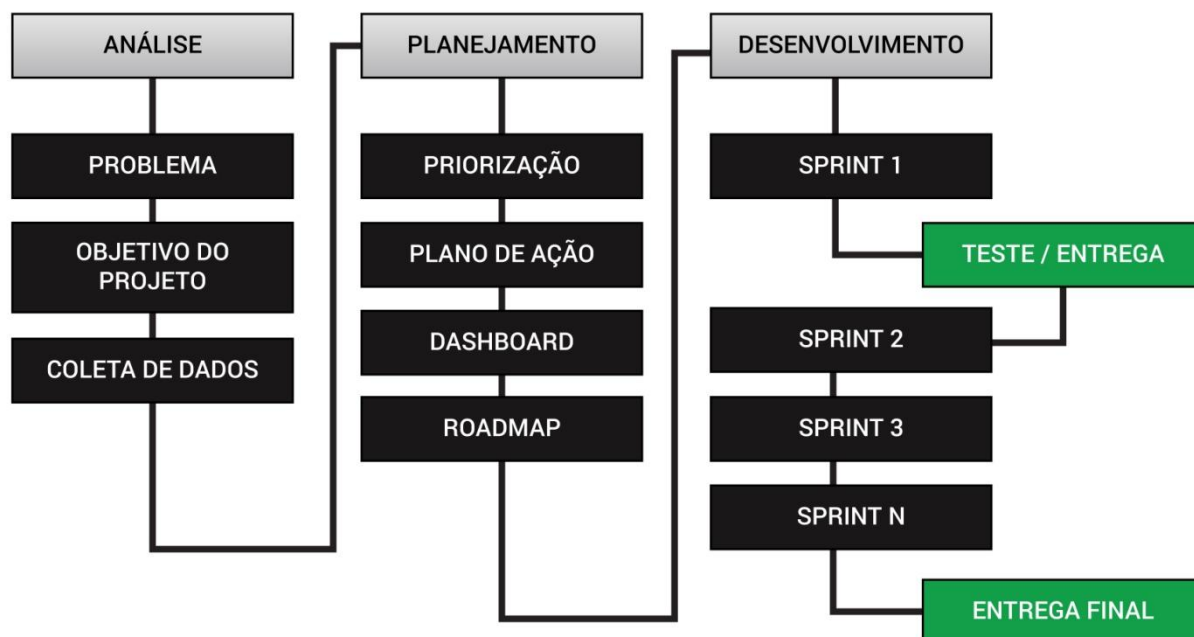


Figura 12 – Etapas 4
Fonte: Autoria própria (2015).

A grande vantagem desse modelo é que os ciclos de trabalho são realizados e acompanhados por toda equipe, pelos usuários e o cliente durante todo o processo por meio de pequenas entregas que ocorrem nessas reuniões, isso minimiza em muito a taxa de retrabalho, pois todos estão cientes de até onde o sistema funciona e o que precisa ser melhorado. Outra vantagem é que os ciclos de trabalho são itinerantes, o produto é constantemente testado e corrigido até chegar à forma ou desempenho desejado.

Em relação à necessidade de equipe, indica-se (em projetos mais extensos) que ela seja dividida em: um gerente de projeto que será responsável pela gestão das pessoas, facilitar o trabalho da equipe e mediar as reuniões; um gerente estratégico responsável pela gestão de cronogramas, acompanhamento das equipes e interface com os usuários e clientes; e por fim o time de desenvolvimento, responsável por desenvolver o produto e deve ter as aptidões necessárias para realizar o projeto.

A seguir, será apresentado o modelo de forma resumida, a fim de dar um panorama geral e facilitar esse primeiro contato com o *framework*. Posteriormente, as etapas do modelo serão exemplificadas em um projeto real.

3.2 Aplicação

A seguir será apresentado o modelo aplicado em um projeto real, as etapas da proposta modelo do *framework* serão aplicadas e explicadas com a intenção de servir como um manual do processo. Lembrando que a proposta é que o método seja mutável e adaptado à necessidade do projeto que se esteja desenvolvendo, não é pretendido aqui criar um modelo único e absoluto e sim apresentar uma abordagem com pontos-chave que devem ser considerados, mas que dependendo das circunstâncias podem e devem ser alterados.

3.2.1 Objetivo do Projeto.

O objetivo do projeto é caracterizado pela meta a ser alcançada. O objetivo deve ser claro e conciso. Quanto mais bem definido, melhor será ponto de partida para o desenvolvimento do projeto. Ele pode ser obtido de várias formas, por uma necessidade de um cliente, uma oportunidade observada pelo próprio desenvolvedor do projeto, uma ideia discutida pela equipe de desenvolvimento e etc.

O objetivo para o exemplo foi definido a partir de um projeto real e é “Desenvolver um portal de ensino de softwares e metodologias voltado para o design e gestão de projetos”.

Mais do que só o motivo pelo qual o projeto deve existir, o Objetivo do Projeto é uma missão que deve ser lembrada pela equipe durante todo o desenvolvimento como a meta a ser atingida.

3.2.2 Coleta e Análise de dados.

Com o objetivo de entender melhor o problema, é necessário coletar e analisar o maior número de dados possíveis sobre o ele. A intenção é que esses dados sejam coletados e analisados o mais rápido possível, para que não se perca muito tempo planejando o projeto e que se parta direto para o desenvolvimento e a

prototipação, por que muitas vezes os dados coletados não condizem com a realidade, até mesmo o que o próprio usuário diz não é exatamente o que ele necessita.

Os dois métodos sugeridos para essa etapa são a realização de um *Brainstorming* sobre o problema combinado com a criação de uma matriz de causa e efeito para traçar o perfil do público alvo usaremos a técnica denominada *Value Design Proposition*.

3.2.2.1 *Brainstorming*

O *Brainstorming* é uma técnica de exploração criativa que consiste em gerar o maior número possível de idéias.

Desenvolvida por Alex F. Osborn e utilizada pela primeira vez em 1938. O objetivo inicial dessa técnica foi coletar idéias do pessoal de sua agência de publicidade. Esse processo foi chamado inicialmente de “Sessões de *Brainstorming*” (BOMFIM, 1995, p. 23).

Atualmente a técnica é largamente conhecida e utilizada pelos mais diversos setores, como uma poderosa fórmula de estimular a criatividade dos colaboradores e propiciar o surgimento de ideias valiosas.

O *Brainstorming* é baseado no princípio da livre associação e tem como objetivo básico estimular um grupo de pessoas a detectar problemas ou produzir idéias e soluções para questões existentes de maneira rápida e direta.

O número de pessoas de um grupo de *brainstorming* pode variar em quantidade. Não é aconselhável que o grupo ultrapasse 12 pessoas por questão de ordem e nem menos de quatro pessoas (OSBORN, 1960). Usualmente uma pessoa deve ser a coordenadora do *brainstorming*. Esse coordenador é responsável por anotar as ideias, manter a sessão dentro do tema especificado, estimular a equipe, entre outras ações que mantenham o bom funcionamento da sessão. Outra figura importante durante o *brainstorming* é o assistente que tem a função de anotar as idéias.

“Não existe tempo ideal para a duração de uma sessão de brainstorming, não deve tomar uma manhã ou uma tarde. Sessenta minutos parecem ser a duração ideal” (KELLEY, 2001, p.74).

Mesmo se caracterizando por um tipo de reunião mais aberta e menos formal, segundo Osborn (1960) quatro regras são fundamentais para retirar o melhor resultado de uma sessão de *brainstorming*:

1. Deve-se banir as críticas, todas as ideias devem ser apresentadas sem a interferência negativa dos outros participantes, todas as considerações sobre as ideias devem ser apresentadas ao fim da sessão.

2. Ideias absurdas devem ser incentivadas: isso reforça ainda mais a possibilidade da livre associação de ideias gerarem *insights* valiosos.

3. Procurar por quantidade: uma sessão de *brainstorm* deve extrair o maior número de ideias possível, quanto mais ideias o grupo tiver, maior será o número de opções a serem escolhidas.

4. Incentive os participantes a criarem sobre a ideia dos outros: numa sessão de *brainstorming* não existe propriedade sobre as idéias, os participantes devem ser incentivados a construir e retomarem as ideias já apresentadas pelos outros participantes transformando-as em ideias melhores.

Para o projeto exemplo, o *brainstorming* foi realizado a partir da pergunta “O que você gostaria de encontrar em um site quando necessita estudar algo relativo a design na internet?”. Um grupo de colaboradores foi convidado para a atividade. A escolha dos convidados seguiu o seguinte critério: ser aluno de design e utilizar a internet para estudar sobre ferramentas de desenvolvimento de projetos. As respostas foram coletadas durante a sessão de *brainstorming* que aconteceu no dia 18/03/2015 na dependência das UTFPR e contou com quatro participantes.

E as respostas encontradas foram:

Área de usuários, Tutoriais, Simples, Vídeos, Apostilas, Arquivos em .pdf, Workshop, Área de fórum, Artigos, *Mobile*, Aplicativos, Cursos completos, Projetos exemplificados, Conteúdos para *download*, Tem que ser convidativo, Objetivo, Fácil de encontrar as informações, Ter conteúdo Grátis, Apresentações de *cases*, Ter um *mailing*, Fontes confiáveis, Conteúdo de qualidade.

3.2.2.2 Diagrama de Ishikawa, ou espinha de peixe:

O diagrama de Ishikawa, também conhecido como espinha de peixe ou diagrama de causa raiz é uma ferramenta de representação visual, organização e mapeamento das causas de um determinado problema.

O método leva o nome de seu criador, Kaoru Ishikawa, um dos pioneiros no campo da gestão de qualidade, largamente estudados no Japão na década de 60.

O diagrama consiste basicamente em um gráfico, semelhante a uma espinha de peixe onde possíveis causas levantadas sobre um problema são colocadas em cada uma das pontas da cadeia e a partir daí, deve-se preencher a “espinha” com fatores-causa e influenciadores de um determinado problema.

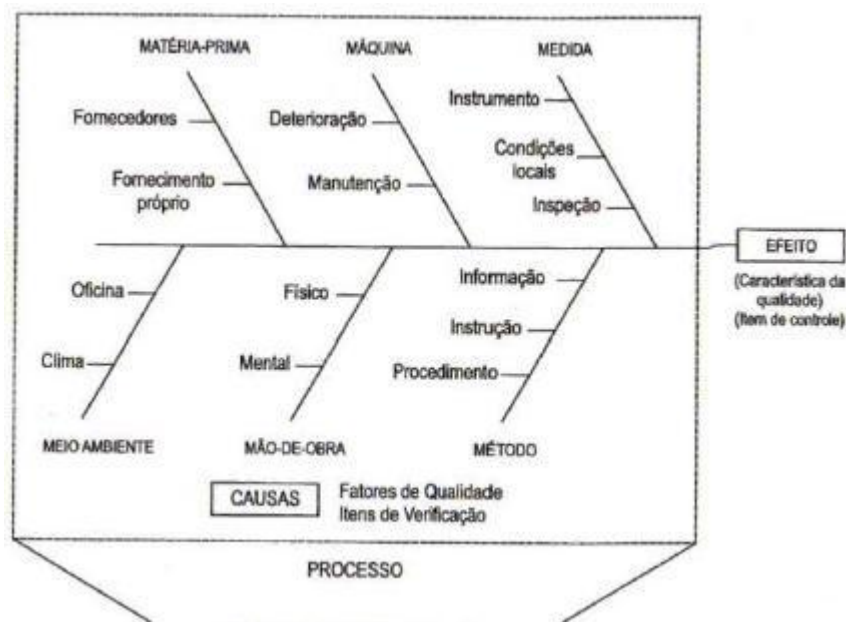


Figura 13 – Diagrama de causa-raiz
Fonte: Campos (2004, p. 20).

O processo de criação de um diagrama de Ishikawa segue os seguintes passos:

1. Identificar o problema ou definir o objetivo a ser atingido;
2. Colocar o problema a ser investigado em uma caixa à direita;
3. Realizar um brainstorming para encontrar todas as causas e influenciadores relativos ao problema;
4. Filtrar causa dos sintomas encontrados;

5. Analisar os dados encontrados para ter certeza que realmente são variáveis do processo;
6. Categorizar as variáveis e agrupa-las;
7. Transferir as variáveis para o diagrama já agrupadas;
8. Decompor todos os dados em que isso por possível, procurando a causa raiz;
9. Decompor até encontrar uma resposta que satisfaça a equipe.

Exemplo:

Nas etapas anteriores foi definido o objetivo do projeto (Desenvolver um portal de ensino de softwares e metodologias voltado para design e gestão de projetos) e realizado um *brainstorming* para levantar os influenciadores do projeto. O próximo passo foi categorizar as informações encontradas no *brainstorming* para definir onde eles ficarão situados nos eixos da espinha de peixe.

Funcionalidade	Conteúdo (formato)	Conteúdo (segmento)	Diferencial
Área de usuários	Tutoriais	<i>Branding</i>	Objetivo
Campo de fórum	Aulas em vídeo	<i>Webdesign</i>	Fácil de encontrar
<i>Mailing / Newsletter</i>	Apostilas para download	Design Gráfico	Fontes confiáveis
<i>Mobile</i>	<i>Workshops</i>	Programação	Simples
	Estudos de caso	Modelagem 3D	Conteúdo free
	Dicas de conteúdo	Gestão de projetos	Convidativo

Quadro 2 - *Brainstorming* categorizado

Fonte: Autoria própria (2015).

Feito isso, montou-se o diagrama, colocando as categorias nos extremos da espinha e preenchido com os itens encontrados, buscando identificar suas causas, até encontrar uma que não possa ser desmembrada. Esse processo ajuda muito a elaborar a próxima etapa: o planejamento da elaboração do projeto.

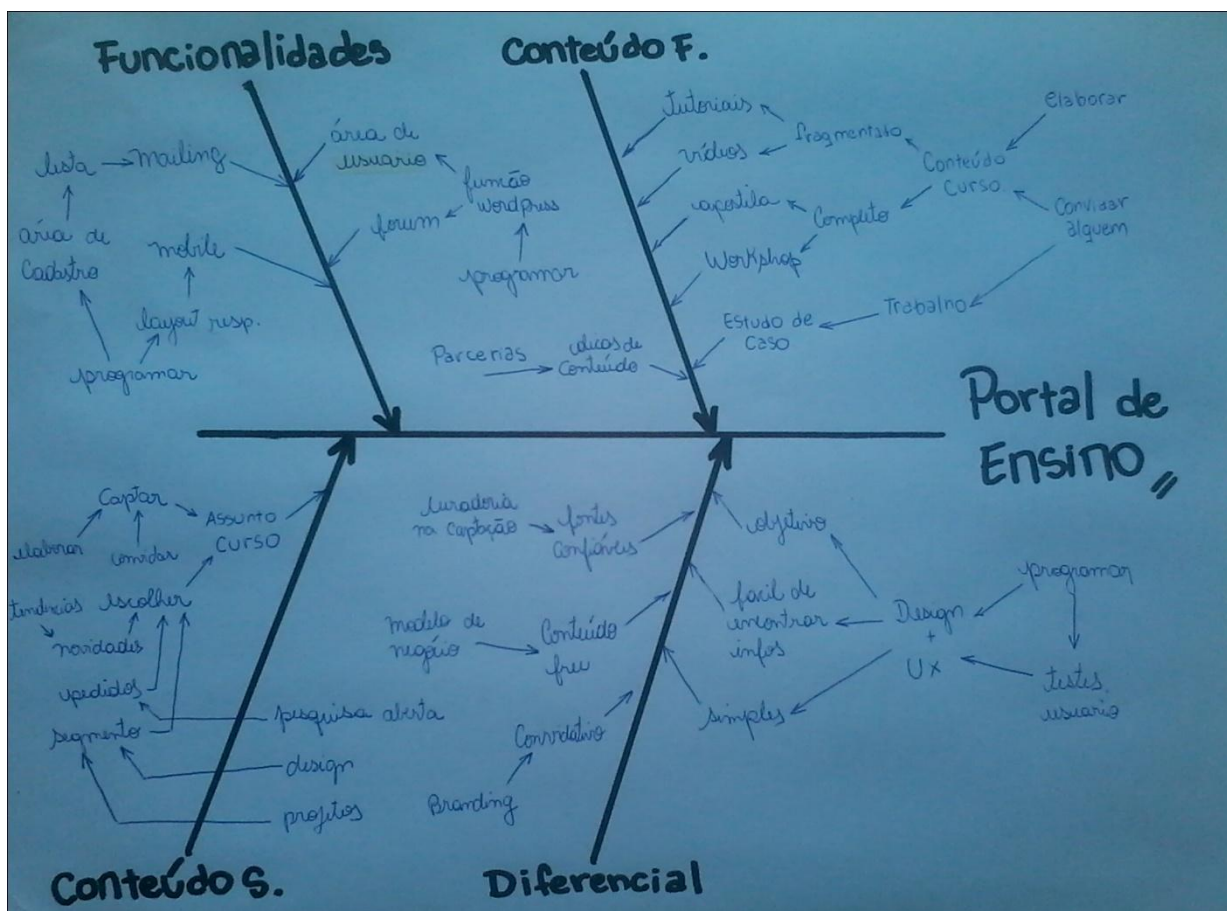


Figura 14 – Aplicação diagrama de causa-raiz
Fonte: Acervo pessoal (2015).

Acima está o resultado obtido no preenchimento da matriz.

3.2.2.3 Value design proposition

Para traçar de forma mais coerente possível o perfil do público alvo, que servirá de subsídio para decisões nas próximas etapas do projeto será utilizado o método desenvolvido por um grupo americano de designers liderados por Alexander Osterwalder, criador do *Business Model Canvas*, denominado *Value Design Proposition*.

O método é baseado em painéis visuais e tem como função a análise profunda do perfil do cliente para buscar o resultado alinhado a esses anseios. A relação entre o perfil do cliente e a proposta de valor foi chamada por eles de *Canvas de Proposta de Valor*.

No livro *Value Proposition Design*, Osterwalder (2014, p.6) define como proposta de valor "uma proposta de valor descreve os benefícios que os clientes podem esperar de determinados produtos e serviços".

Entrando na estrutura dos painéis e funções, ele é dividido em três partes cada uma com uma função específica.

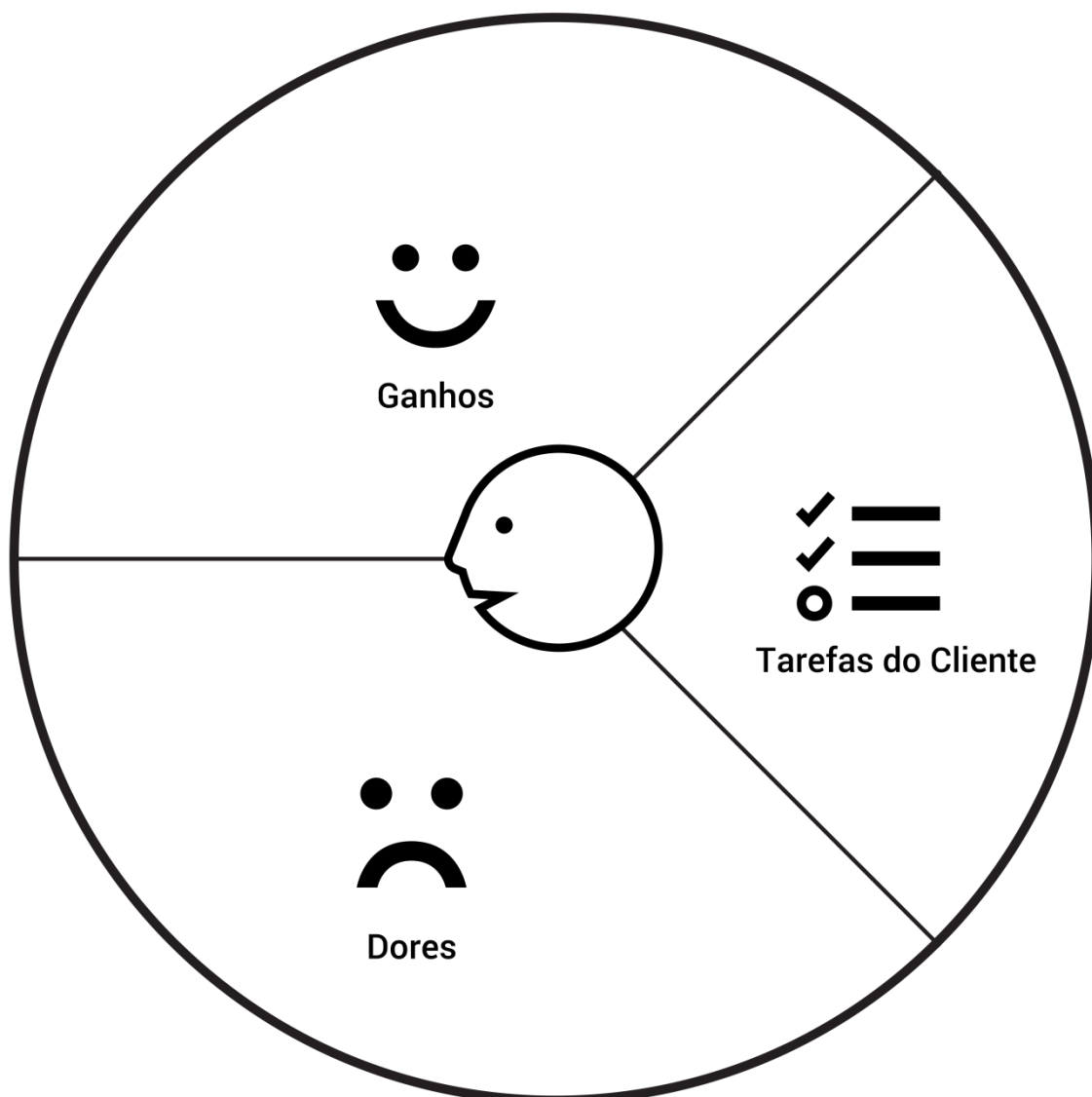


Figura 15 – *Value Design Proposition*
Fonte: Osterwalder (2014, p.9).

Tarefas do cliente: são as atividades que o público está tentando executar no trabalho e na vida pessoal. É de suma importância descrever as tarefas da perspectiva do cliente e atentar aos três tipos primários de tarefas:

A. Tarefas funcionais: quando o público tenta solucionar um problema específico.

B. Social: como o cliente quer ser percebido.

C. Emocional: quando o cliente busca um estado emocional específico.

Dores: as dores se resumem basicamente em três segmentos; a obtenção de resultados indesejados durante a realização de uma tarefa; obstáculos que impeçam a execução de algo e os riscos e as consequências de um resultado ruim ou um erro.

Ganhos: são todos os benefícios que o cliente deseja. Os ganhos são categorizados por: necessários, sem os quais a solução não teria serventia; esperados, são os ganhos básicos além dos funcionais; desejados, são as melhorias que o cliente gostaria que um produto tivesse e os inesperados quando o produto apresenta funções que superam as expectativas.

Sobre os ganhos, esses podem ser funcionais, emocionais, sociais e custo de compra.

Na aplicação foi utilizado como perfil ideal de público o estudante de design que busca aprender conteúdos por meio de sites da internet, na figura abaixo seguem os dados coletados em relação a “ensino de *softwares*” e “gestão de projetos”.



Figura 16 – Value Design Proposition
Fonte: Autoria própria (2015).

3.2.3 Planejamento

Ao término da etapa de análises do problema, agora se segue a elaboração da estratégia do projeto, por assim dizer. Essa etapa é composta da montagem do plano de ação, criação do *Product Dashboard* e do *Roadmap* do projeto.

3.2.3.2 Plano de ação

Levando em consideração os dados anteriores, agora deve-se pensar na ordem em que os pontos serão trabalhados e como será feito o desenvolvimento do projeto. Existem várias ferramentas utilizadas na elaboração de um plano de ação, a adotada por nós será a *5W2H*.

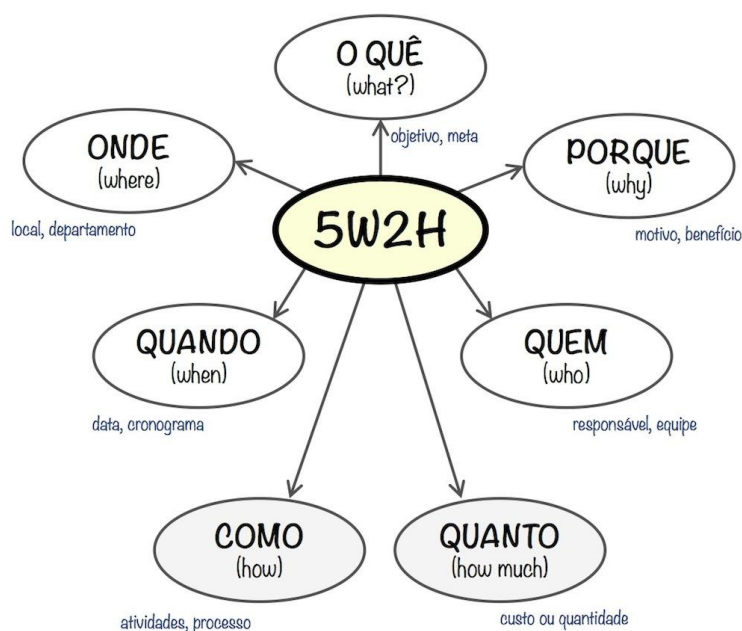


Figura 17 – Representação 5W2H
Fonte: Gomes (2014).

O *5W2H* consiste em um método de elaboração de *checklist* detalhado para a realização de uma atividade específica. Com ele, é possível definir a rota em que as atividades serão realizadas. Para tal, o método propõe que sejam respondidas algumas perguntas-chave (NAKAGAWA, 2012).

A escolha do *5W2H* se deve pelo fato de ser um método leve, ágil e fácil de ser elaborado. O método ajudará a pensar em uma ordem lógica para realizar o projeto. O nome *5W2H* provém das iniciais (em inglês) das etapas do processo: *what*, *why*, *where*, *when*, *who*, *how* e *how much*. Cada uma dessas palavras caracterizam uma etapa ou uma pergunta a ser respondida.

What? O que será feito?

Why? Por que será feito?

Where? Onde será feito?

When? Quando será feito?

Who? Quem irá fazer?

How? Como será feito?

How much? Quanto irá custar?

Exemplo:

No diagrama de causa raiz, foi identificado como ponto de partida de alguns processos: programar, elaborar o layout, elaborar e captar conteúdo, identificar tendências e pesquisar conteúdos, modelo de negócios e *branding*. Algumas causas foram agrupadas por terem o mesmo sentido. Agora, seguindo as premissas da metodologia ágil, elenca-se elencar os quesitos necessários para que o sistema funcione o mais rápido possível utilizando um diagrama impacto X esforço.

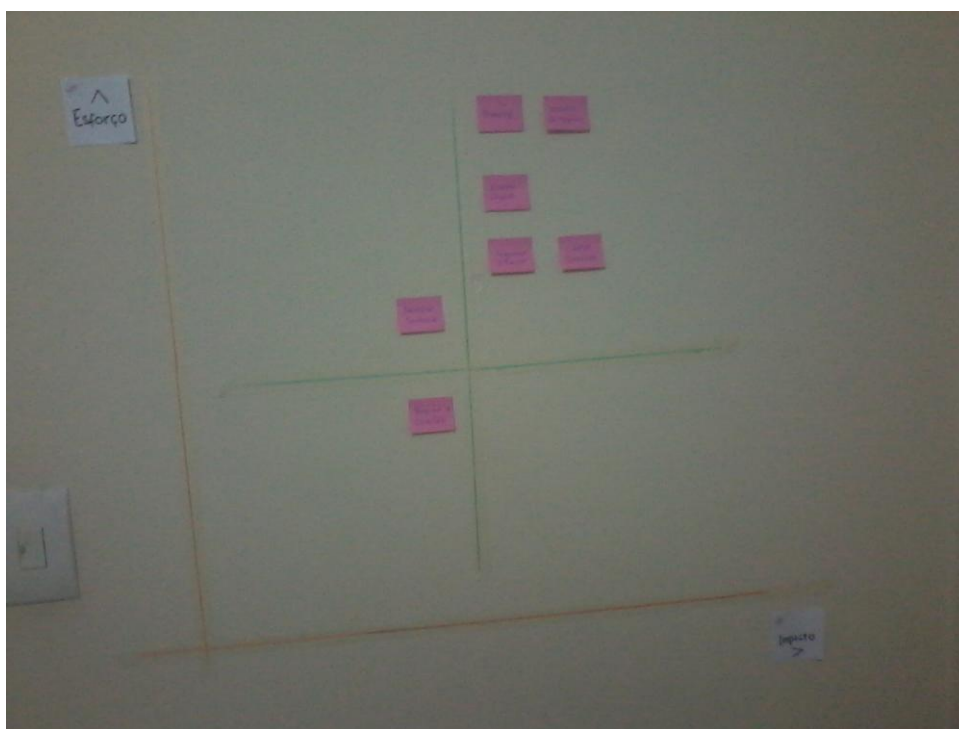


Figura 18 – Matriz de priorização.
Fonte: Autoria própria.

Nesse exemplo, por ser o início de um projeto complexo, grande parte das demandas se situa em “muito esforço e grande impacto”, pois são primordiais para o lançamento do projeto. As situadas em “grande impacto e pouco esforço”, “muito

esforço e pequeno impacto” e “pequeno impacto e pouco esforço” serão realizadas com menor prioridade.

Lembrando que se leva em consideração o primordial para colocar o sistema para funcionar o mais rápido possível. Então definidas as atividades monta-se o plano de ação seguindo essa ideia. A ordem definida das atividades foi:

1. *Branding*;
2. Modelo de negócio;
3. Elaborar layout;
4. Programar e publicar;
5. Captar conteúdos;
6. Pesquisar mais conteúdos;
7. Identificar tendências.

A seguir é apresentado um exemplo de plano de ação, usando como base a etapa de *branding* do projeto.

5w2h aplicada a etapa de *Branding*:

What? (O que será feito?): Uma estruturação do branding do negócio.

Why? (Por que será feito?): A estruturação do branding tem como objetivo definir a imagem que queremos dar ao negócio, a segmentação do mercado a ser atendido, o posicionamento da marca, entre outros fatores.

Where? (Onde será feito?): O local que será realizado é uma opção da equipe de desenvolvimento, nesse caso uma locação própria do desenvolvedor.

When? (Quando será feito?): A etapa deve ser iniciada até 04/11/2014 e tem o prazo de conclusão até 07/11/2014.

Who? (Quem irá fazer?): Os envolvidos no projeto serão o desenvolvedor e alguns convidados.

How? (Como será feito?): O método empregado será o *Brandkey*, com ajuda de alguns convidados para a elaboração.

How much? (Quanto irá custar?): Sem custo envolvido diretamente ao desenvolvimento, apenas as horas empregadas no processo.

Esse modelo deve ser aplicado nos demais itens, com isso tem-se projeção do projeto de forma mais detalhada, sendo possível estimar melhor a data da primeira entrega e facilita a ordenação das atividades no *Product Dashboard*.

3.2.3.2 Product Dashboard

O *Product Dashboard* é o quadro onde os quesitos do projeto serão organizados seguindo sua ordem de execução (figura 20).

Na etapa anterior, levantaram-se todas as causas possíveis de influência sobre o projeto a ser realizado. Em seguida, foi elaborado o plano de ação para elas e agora é a hora de ordenar as tarefas necessárias para alcançar nosso objetivo, exibi-las no *dashboard* para termos controle visual das atividades que se deve realizar e acompanhar o *status* do trabalho no decorrer do projeto.

Primeiramente, define-se a estrutura básica do *dashboard* que em muito se assemelha com um *kanban*. O quadro possui os seguintes campos: a realizar (*to do*), em processo (*in progress*), concluído (*done*) e melhorias (*improvement*).

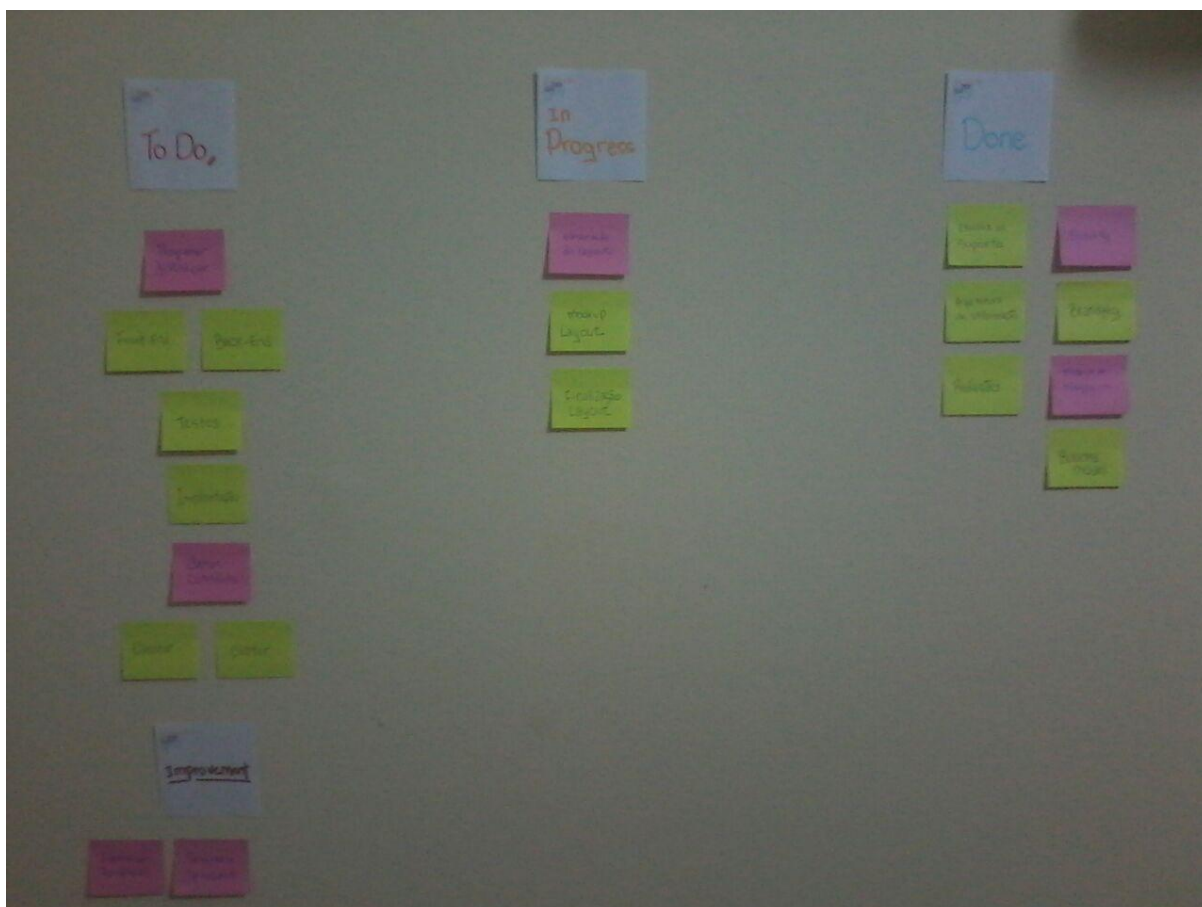


Figura 19 – Product Dashboard
Fonte: Autoria própria (2015).

Em “a realizar” ficam as demandas em ordem decrescente a serem trabalhadas. O nível de detalhamento das demandas deve ser maior dependendo de sua urgência a ser realizada. Assim como no *scrum*, não deve ater-se a muitos detalhes no planejamento de demandas que serão realizadas em longo prazo, pois é possível que elas não venham a ser realizadas, assim como novas demandas serão anexadas ao processo a medida que as entregas e testes forem realizados.

Na divisão "em processo" ficarão as tarefas que estão sendo executadas no momento presente. É indicado que nessa área não estejam muitas atividades simultaneamente, pois isso indica que elas estão sendo realizadas ao mesmo tempo e ao menos que se tenham uma grande equipe, isso pode trazer alguns problemas, principalmente relacionados a foco, prazo e qualidade das entregas.

Ao finalizar uma atividade ela passará para o campo “concluído”, local do painel onde todas as etapas já realizadas no projeto ficam.

No campo “melhorias”, aparecem além das atividades relacionadas a melhorias do produto também atividades de melhoria de processo e as demandas que só são concluídas ao fim do projeto (reuniões, por exemplo).

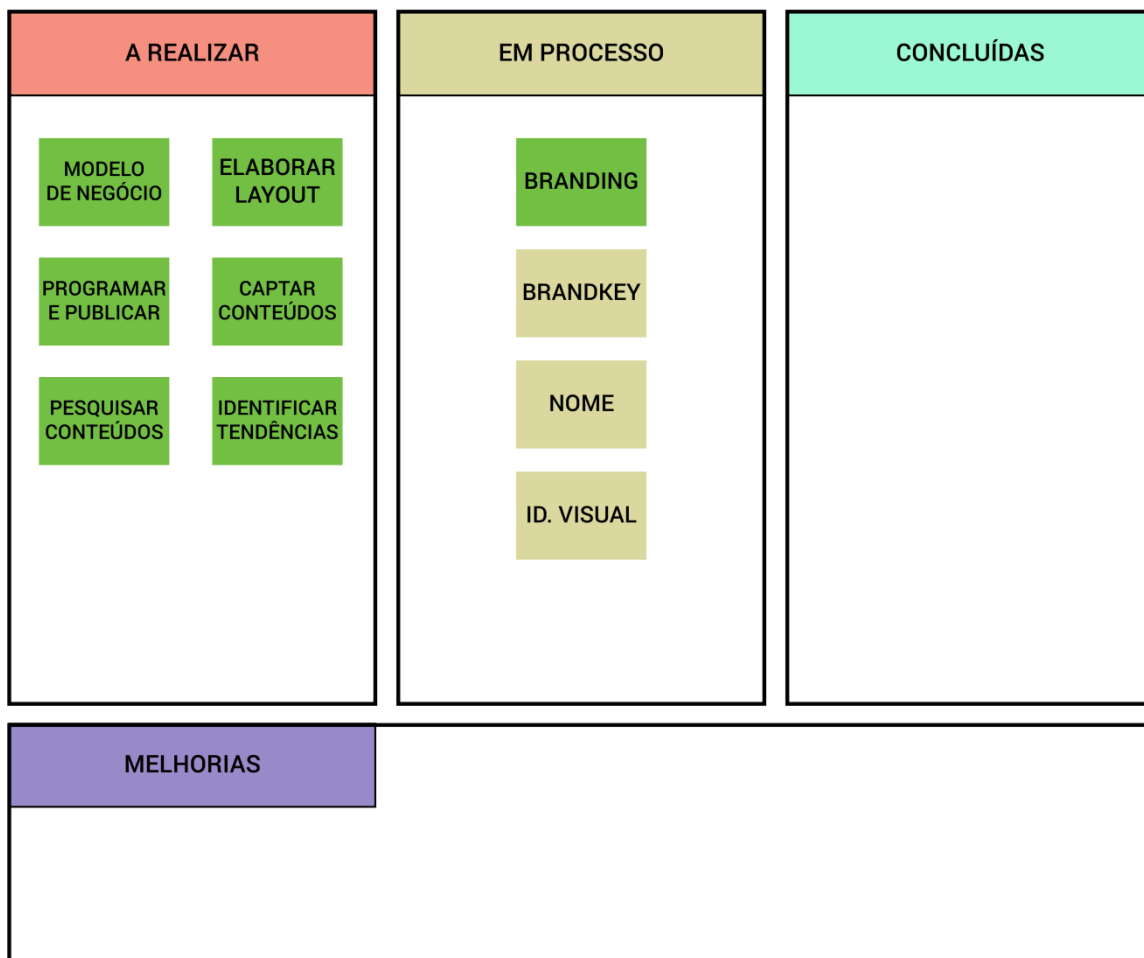


Figura 20 – Product Dashboard
 Fonte: Autoria própria (2015).

3.2.3.3 Roadmap

O *Roadmap* é a etapa final do planejamento e uma das mais importantes. Pode-se definir *Roadmap* como “o plano de liberação de recursos, macro-funcionalidade ou características-chave de um produto, através de prévias, ao longo do seu ciclo e vida” (ELABORANDO UM ROADMAP DO PRODUTO, 2011).

Pensando em sua aplicação o *Roadmap* é primordial para o bom andamento do projeto e suas entregas, é por meio dele que se planeja o conteúdo entregável de cada *sprint*, definem-se as datas das entregas e baseado nisso o cronograma é montado.

O ideal é que no final de cada *sprint* exista algo entregável, seja o produto com as suas funcionalidades básicas ou subsídios suficientes para a próxima etapa do projeto.

A utilidade do *roadmap* no projeto fica evidente quando se consideram os problemas enfrentados pela falta desse planejamento. Os mais comuns são o desrespeito aos prazos, ausência de entregas frequentes aos clientes, alto índice de retrabalho pela falta de *feedback*, má gestão das demandas e erro na previsão do tempo de conclusão do projeto.

Vale frisar é que o *Roadmap* não é elaborado de forma definitiva para todo o projeto. Durante as *sprints*, novas informações e necessidades podem ser identificadas, aumentando ou diminuindo o tempo do projeto. No momento em que fatos que influenciem no tempo de conclusão do projeto sejam identificados, deve-se atualizar o *Roadmap*. Lembrando que, além de uma ferramenta de gestão, o *Roadmap* é uma ferramenta de comunicação e deve estar exposta a toda a equipe.

Exemplo:

O *Roadmap* deve contemplar todas as entregas até a finalização do projeto. No exemplo, foi definido que as entregas aconteçam a cada quatro dias, independente do fim da *sprint*, sendo a duração dessa relativa ao conteúdo a ser entregue.

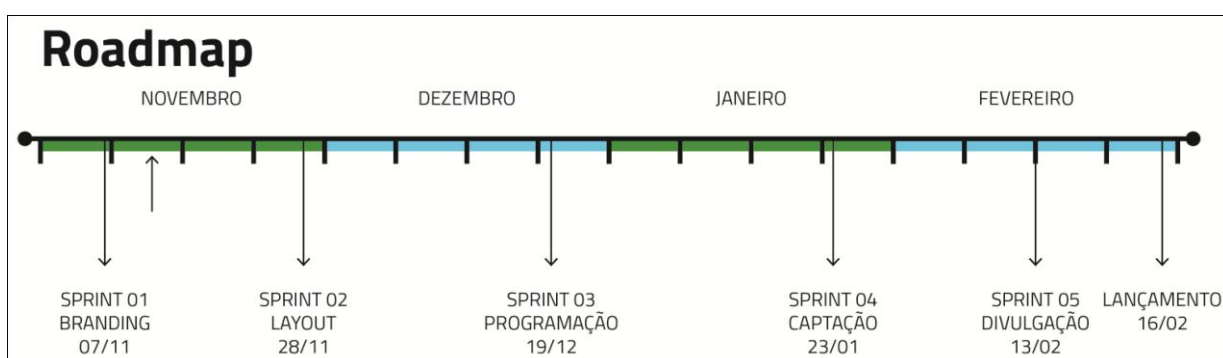


Figura 21 – Roadmap do projeto.
Fonte: Autoria própria (2015).

3.2.4 Sprint

Define-se como *sprint*, um ciclo de trabalho que visa entregar um incremento do produto pronto a cada vez que esse ciclo chega ao fim. Esse incremento é a meta que cada *sprint* deve alcançar, ela é definida seguindo a ordem das demandas representadas no *Product Dashboard*.

O *Sprint* possui estrutura própria de desenvolvimento e se assemelha a um miniprojeto. Durante seu ciclo acontecem alguns pequenos eventos, eles são: uma reunião de planejamento da *sprint* (*Sprint Planning*), reuniões diárias de acompanhamento (*daily sprint*), e reuniões de encerramento (*sprint review* e *sprint retrospective*). Esses eventos são semelhantes aos realizados no *Scrum*, salvo algumas modificações na aplicação, visto que no modelo proposto, a estruturação da equipe é diferente.

Quando um projeto chega à etapa de *sprint*, ele permanece em ciclos sucessivos até sua conclusão. Sendo assim, correções de planejamento, adição de novos pré-requisitos ou quaisquer outras mudanças de escopo devem acontecer dentro dos tempos das *Sprints*, não sendo necessário retornar a etapas de planejamento.

Como dito anteriormente, a meta do *Sprint* é selecionada seguindo a premissa de que se deve entregar o produto com suas funções básicas o mais rápido possível. Esse parâmetro já foi definido na elaboração do *Product Dashboard*, portanto a equipe deve discutir durante a *Sprint Planning* qual será a meta a ser atingida baseando-se na ordem do *dashboard*.

O tempo de duração de uma *Sprint* é fixo e deve ser respeitado sem alterações, dessa forma é estabelecido um ritmo para a equipe e as entregas são frequentes, garantindo um *feedback* mais preciso e rápido, ajudando a monitorar o rendimento da equipe pelo *Burnout Chart*, que será mantido no modelo por ser um importante indicador de performance da equipe. Segundo SABBAGH (2013, p. 198) devem-se levar em consideração alguns quesitos para definir o tempo de uma *Sprint*, eles são:

1. Frequência em que o projeto é alterado: Se as necessidades do projeto mudam constantemente é necessário que a *Sprint* tenha uma duração menor, assim a equipe de desenvolvimento consegue responder melhor a alterações.
2. Frequência de *feedback*: Quanto menores forem os *Sprints*, mais frequentes serão as entregas e conseqüentemente o *feedback* sobre a

entrega. Esse cenário de entregas rotineiras diminui muito as chances de grandes retrabalhos, pois em caso de algum problema detectado em uma das entregas a correção é imediata.

3. Sobrecarga: Em *sprints* muito curtos, as reuniões de *planning*, *daily*, *review* e *retrospective* podem ocupar mais tempo do que o próprio ciclo de trabalho, injustificando a *Sprint*.
4. Ritmo: *Sprints* menores ditam ritmo para as equipes, porém se o ciclo for muito curto pode tornar a rotina da equipe de desenvolvimento estressante.
5. Tamanho do projeto: Para projetos curtos é mais indicado que as *Sprints* também sejam curtas, dessa forma aumentam as possibilidades de erros serem detectados no decorrer do trabalho.
6. Valor do negócio: É interessante que ao término de cada *Sprint* a entrega tenha um valor significativo, portanto, *Sprints* muito curtas podem não ser suficientes para uma entrega de valor considerável.
7. Iniciantes: Times iniciantes devem optar por ciclos menores, assim obterão *feedbacks* internos e externos com mais frequência, aprimorando o processo.
8. Foco: *Sprints* muito longos tendem a sofrer perda no foco central.

Como dito anteriormente, ao fim de cada *sprint*, deve-se obter um produto entregável que possua valor, mas, a decisão de entregá-lo ao cliente final é do gerente do projeto em consenso com a equipe de desenvolvimento. Às vezes se torna necessário somar algumas *Sprints* para se obter um valor entregável palpável ao cliente, porém isso atrasa o *feedback* e aumenta a chance de retrabalho.

3.2.4.1 Componentes de uma *sprint*

Como dito a pouco, a *sprint* possui alguns eventos internos, que serão explicados detalhadamente.

Sprint Planning

É o planejamento inicial da *sprint*, nele define-se a meta a ser alcançada e a forma com que a equipe de desenvolvimento irá trabalhar para chegar a esse

resultado. A escolha da meta se dá pela ordem definida no *dashboard*. Seleccionada a meta, o time discute a forma que irá executar utilizando os dados levantados na *5W2H* e estima quanto tempo irá levar para realizar cada sub-tarefa.

Nessa etapa é definido o *Burnout Chart* da *Sprint*, ele é o gráfico de acompanhamento e controle do tempo empregado na *Sprint*. A partir dele é possível reconhecer atrasos nas entregas e estimar o tempo de produção da equipe. Para elaborar o *Burnout Chart*, a equipe, munida dos tempos para realizar cada sub-etapa constrói um gráfico “tempo restante x dias” da *Sprint*, onde tempo restante é a soma dos tempos das demandas da *Sprint* e dias restantes configuram o tempo restante para a conclusão da *Sprint*. Ambos indicadores devem decrescer no decorrer do projeto (exemplo na figura 22).

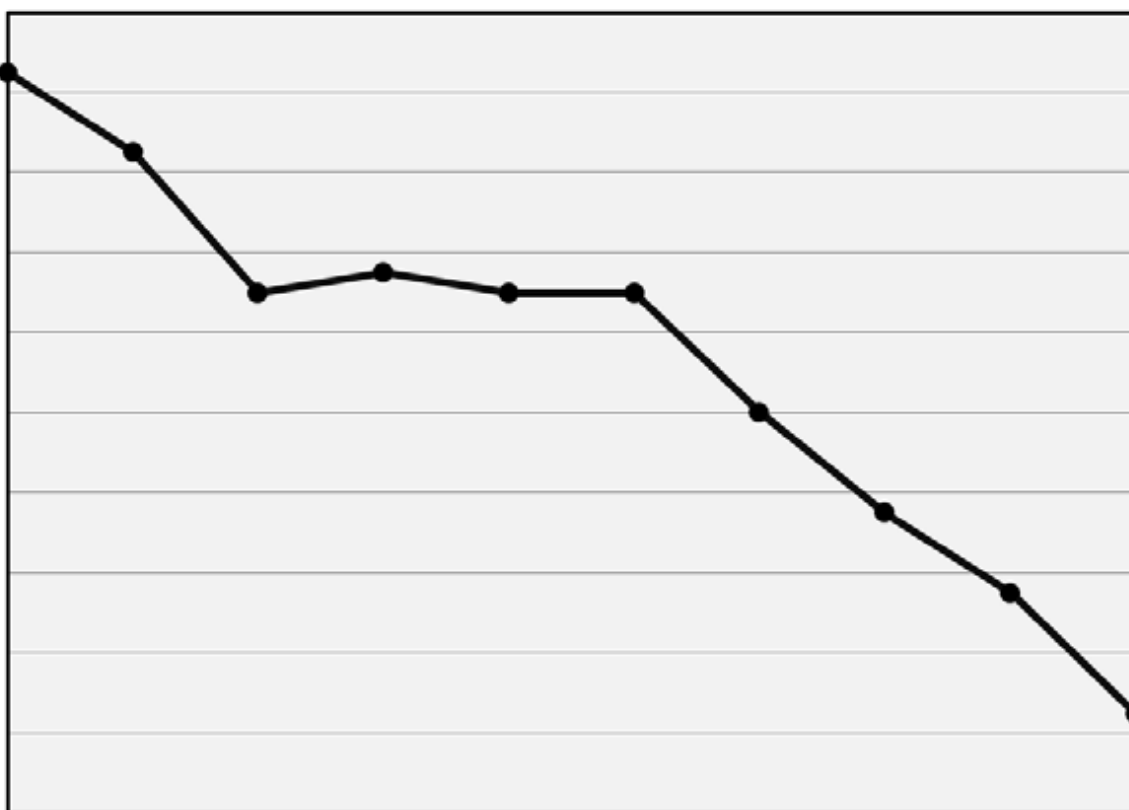


Figura 22 – Exemplo de Burnout Chart da Sprint.
Fonte: Sabbagh (2013, p. 45).

O *Burnout Chart* deve ser atualizado todos os dias na *Daily Sprint*.

Daily Sprint

Reunião diária de curta duração onde os integrantes da equipe apresentam no que estão trabalhando, alguma dificuldade que estejam enfrentando e o plano de

ação para o dia. É indicado que a *Daily Sprint* ocorra no início do dia de trabalho e tenha duração máxima de 15 minutos.

A *Daily Sprint* não tem a função de reunião de debate ou discussão de demandas e objetivos, essas questões se surgirem devem ser discutidas entre os interessados e fora do tempo de reunião para evitar desperdício de tempo dos membros da equipe que não estão envolvidos no assunto.

Sprint Review

É a reunião de apresentação do resultado da *Sprint* que ocorre no último dia do ciclo vigente. Na *Sprint Review* são convidados todos os interessados no resultado obtido, sejam clientes, usuários ou qualquer um que possa dar um *feedback* no sentido de aprimorar o resultado.

Feita a apresentação do resultado, a equipe deve coletar os dados do *feedback*, eles serão utilizados na próxima *sprint planning*, buscando aprimorar o produto.

Sprint Retrospective

Na *Sprint Retrospective* a equipe analisa a *Sprint* concluída a cerca dos métodos empregados e ao desempenho da equipe, buscando a melhoria do processo. A *Sprint Retrospective* não busca levantar melhorias relacionadas ao desenvolvimento do produto em si, mas sim fatos pertinentes ao desempenho da equipe.

A *Sprint Retrospective* é de suma importância, pois é possível aprimorar muito o trabalho da equipe por meio dela. As observações permitem encontrar pontos favoráveis e processos em que há chance de melhoria. Para facilitar a mediação da reunião, se sugere a elaboração de uma lista de duas colunas em que na primeira os membros da equipe listarão o “que deu certo” e na segunda coluna “o que pode ser melhorado”, facilitando assim a discussão e visualização de toda a equipe (exemplo na figura 23).



Figura 23 – Dashboard da reunião de sprint retrospectiva.
Fonte: Sabbagh (2013, p. 230).

Encerramento da *Sprint*

Uma *Sprint* pode chegar ao fim em três tipos de situação:

- A meta a ser alcançada foi atingida;
- O prazo da *Sprint* se excedeu;
- A *Sprint* foi cancelada pelo gerente do projeto.

No primeiro caso a *Sprint* se encerra da forma correta, todas as metas da foram alcançadas, então ela termina para dar lugar ao próximo ciclo de trabalho, caso esse não seja o último *Sprint* do projeto.

A segunda situação acontece geralmente quando a equipe não se planeja tão bem em relação aos prazos ou enfrenta alguma dificuldade que não pode ser sanada durante a *Sprint*. O gerente de projeto deve decidir como proceder nesses casos, se propõe uma *Sprint* adicional para terminar o trabalho ou abandona a meta da *Sprint* por algum motivo.

A terceira situação é mais rara, acontece em casos de cancelamento de projetos, falta de pessoal qualificado e etc. Essa é geralmente a pior saída para uma *Sprint*, só utilizada em casos extremos.

Alguns eventos são realizados no final da *Sprint*, eles são a entrega do trabalho realizado durante o ciclo na reunião de *Sprint review*. A reunião é

convocada pelo gerente de projeto e gerente de estratégia e nela são apresentados os resultados ao cliente e procura-se receber o *feedback* para iniciar a próxima etapa de desenvolvimento. A outra reunião é a *sprint retrospective* que busca mapear o ciclo de trabalho buscando reforçar os pontos fortes da equipe e melhorar pontos de dificuldade.

Exemplo de execução de uma *sprint*.

Sprint Planning

Inicia-se o *Sprint Planning* definindo a meta que se deseja alcançar no *Sprint*. Pela ordem do *dashboard* a primeira demanda a ser realizada é uma estrutura básica de *branding* do projeto. Lembrando que já foi definido na etapa da elaboração do plano de ação um escopo básico para realização da demanda que agora pode ser utilizado.

1. *What?* O que será feito?

Uma estruturação do *branding* do negócio.

2. *Why?* Por que será feito?

A estruturação do *branding* tem como objetivo definir a imagem que se quer dar ao negócio, a segmentação do mercado a ser atendido, o posicionamento da marca, entre outros fatores.

3. *Where?* Onde será feito?

O local que será realizado é uma opção da equipe de desenvolvimento, nesse caso uma locação própria do desenvolvedor.

4. *When?* Quando será feito?

A etapa deve ser iniciada até 04/11/2014 e tem o prazo de conclusão até 07/11/2014.

5. *Who?* Quem irá fazer?

Os envolvidos no projeto serão o desenvolvedor e alguns convidados.

6. *How?* Como será feito?

O método empregado será o *Brandkey*.

7. *How much?* Quanto irá custar?

Sem custo envolvido diretamente ao desenvolvimento, apenas as horas empregadas no processo.

Munido das etapas do plano de ação detalha-se melhor o plano antes de prosseguir.

a) Estudar a ferramenta *brandkey*;

- b) Elaborar o *brandkey*;
- c) Pesquisa com o público alvo;
- d) Realizar correções;
- e) Concluir.

Temos as etapas bem definidas, agora vamos atribuir tempos a elas.

- a) Estudar a ferramenta *brandkey*; (4 horas).
- b) Elaborar o *brandkey*; (16 horas).
- c) Pesquisa com o público alvo; (2 horas).
- d) Realizar correções; (1 hora).
- e) Concluir.

Com isso, tem-se um total de 23 horas de trabalho nessa *Sprint*, seguindo a recomendação do plano de ação segue-se o prazo de 04/11/2014 a 07/11/2014 uma vez que a equipe de desenvolvimento concorde (pensando em uma situação real) que é um prazo realizável. Com esses dados é possível elaborar o *Burnout Chart* dessa *Sprint*.

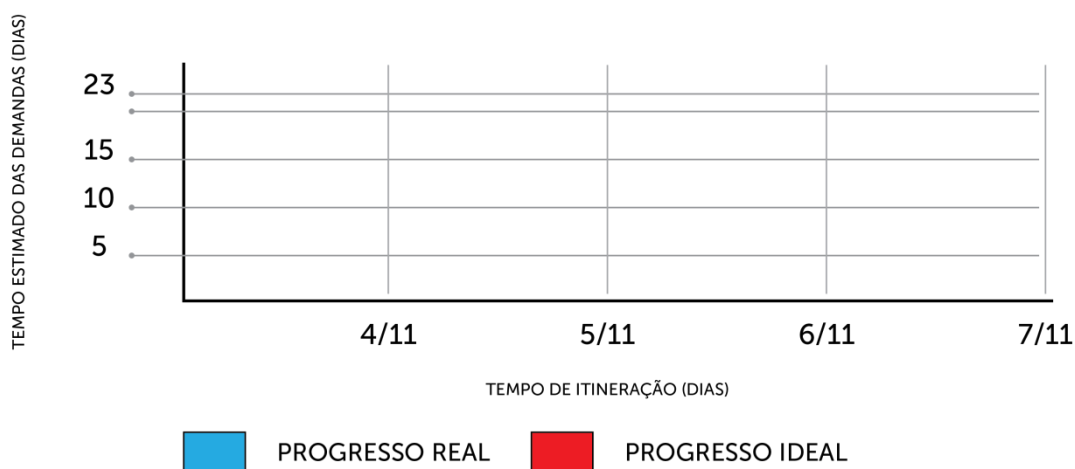


Figura 24 – Exemplo Burnout Chart da sprint
Fonte: Autoria própria.

Início do ciclo de trabalho

Na sequência inicia-se a *Sprint*, a primeira etapa foi definida como “Pesquisar sobre o *Brandkey*” que consistiu em um levantamento teórico sobre a ferramenta. Os materiais encontrados sobre o assunto foram o modelo da *European*

Institute for Brand Management (EIBM), e uma palestra do evento *InnovationExperts* chamada “*Brandkey*, a chave para gerenciar sua marca”.

O *brandkey* é uma ferramenta de gerenciamento que busca simplificar a montagem e a apresentação da estratégia de uma marca. O método é fundamentado em encontrar a “essência” da marca, a razão dela existir. É um método imersivo, ou seja, o diagrama é preenchido por meio de uma pequena coleta de dados para cada um dos campos. Feito todo o processo o usuário fica apto a criar uma rota para a elaboração das ações envolvidas a estratégica da marca (INNOVATION EXPERTS, 2014).

Um grande problema encontrado (relatado no *Daily Sprint*) no uso do *brandkey* é o fato de ser extremamente escassa a literatura técnica sobre o método, principalmente em português. Na aplicação, optou-se utilizar o modelo exibido na figura 25, por ser mais completo do que o modelo disponibilizado pela EIBM. Esse modelo é semelhante ao utilizado na palestra da *Innovation Experts*.

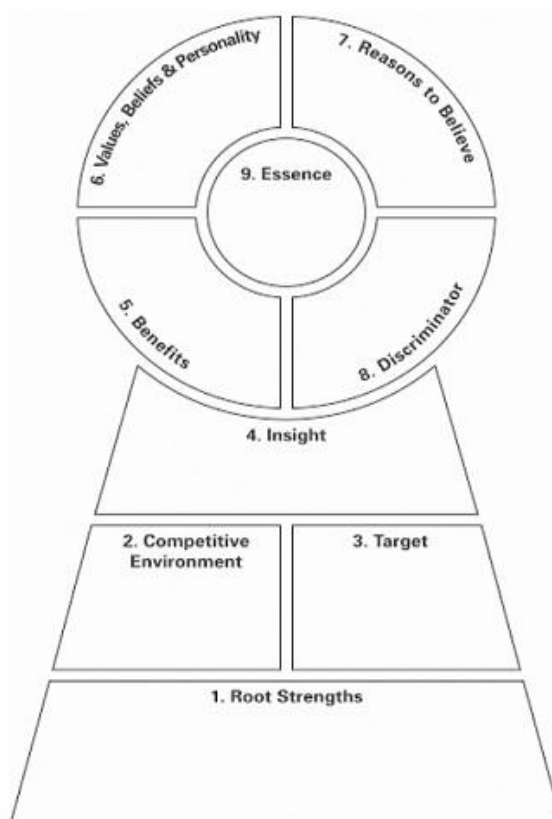


Figura 25 – Modelo de *brandkey*
Fonte: SPARK IDEAS (2011).

O diagrama possui nove campos, o preenchimento de cada um deles colabora para que, ao final do processo se tenha várias informações sobre a forma da marca a ser criada. Abaixo segue uma breve descrição de como preencher cada um dos campos.

1. Pontos fortes;

O produto, valores e/ou benefícios que tornam grande a marca que queremos construir.

2. Ambiente Competitivo;

O mercado e as alternativas como são vistos pelo consumidor e o valor relativo que a sua marca oferece nesse mercado.

3. Alvo;

A pessoa e a situação para a qual a marca é sempre será a melhor escolha. Suas atitudes e valores definidos em termos, e não apenas dados demográficos.

4. Introspecção;

Composto de tudo o que se sabe sobre o consumidor e suas necessidades (neste ambiente competitivo), no qual a marca está situada.

5. Benefícios;

A diferenciação funcional, os benefícios emocionais e sensoriais que motivam a compra.

6. Valores, crenças e personalidade;

Os valores da marca, o que a marca representa e acredita em e sua personalidade.

7. Razões para acreditar;

A prova que oferecemos para substanciar os benefícios e a experiência de marca.

8. Discriminador

A razão que o torna único, mais atraente para o consumidor-alvo e que o faz escolher a sua marca.

9. Essência

Destilação da marca em uma ideia núcleo ou promessa.

A elaboração do *brandkey* ocorreu no segundo e terceiro dias da *Sprint*, o resultado obtido pode ser observado na figura 26.

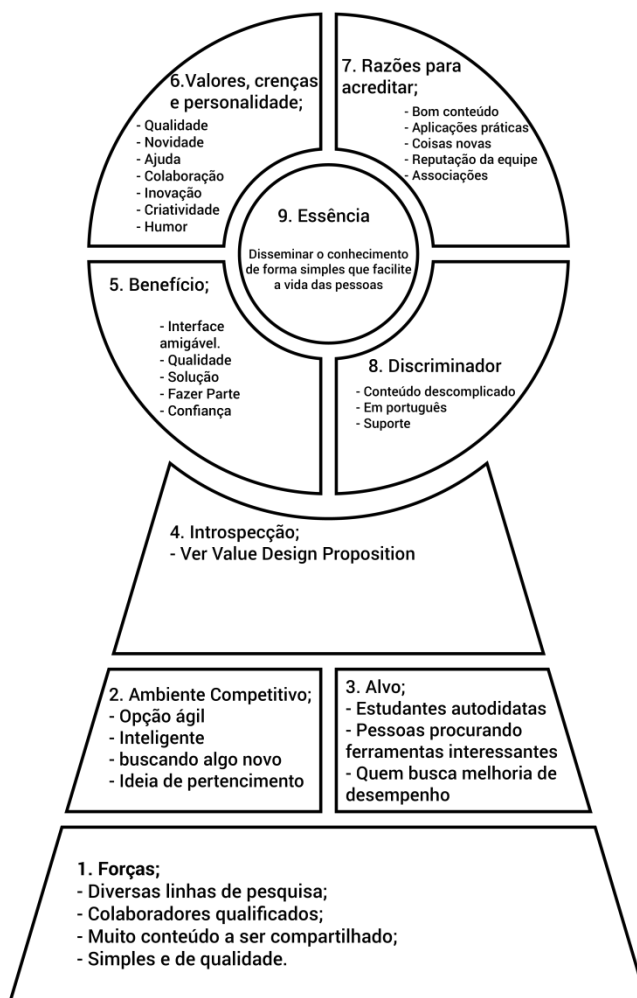


Figura 26 – Brandkey preenchido.
Fonte: Autoria própria.

Baseando-se na *sprint* exemplo, a reunião de *review* da mesma teria os seguintes tópicos:

- Apresentação do objetivo da *Sprint*;
- Apresentação do resultado obtido;
- Abertura para as considerações do usuário.

Na reunião de *retrospective*, que é realizada entre a equipe de desenvolvimento a pauta seria:

- Nível de satisfação com o resultado obtido?
- O que deu certo?
- O que pode ser melhorado?

Fechada a *sprint* um novo ciclo deve ser iniciado até que o *dashboard* esteja vazio e o projeto se encerre.

Encerramento do projeto

O projeto se dá por encerrado quando todas as metas foram alcançadas e não existe mais a necessidade de se realizar mais uma *Sprint*. O projeto já foi aprovado e deve ser entregue ao cliente com todos os direitos previstos no contrato de trabalho ou acordados no início do projeto.

4 CONSIDERAÇÕES FINAIS

Elaborar um trabalho de graduação antes de qualquer coisa é um exercício de desapego ao senso de previsibilidade das coisas. Esse projeto em específico passou por uma reformulação total no momento em que a pesquisa original chegou a um ponto de contradição entre a proposta e o resultado obtido.

O título original do trabalho era “Sistema de gerenciamento de referências como apoio a etapa da metodologia criativa dentro do processo de design”. Além de ter um nome bem extenso, o sentido que a proposta abrigava era de igual tamanho. Embora não esteja de forma explícita nesse projeto, a proposta original esteve presente em quase todo o desenvolvimento do trabalho atual.

O projeto original buscava entender o funcionamento da “mente criativa”. Para tal propunha-se mapear as pesquisas sobre os mecanismos da criatividade e então propor uma ferramenta que servisse de suporte para desenvolvedores na etapa de criação dentro do projeto de design.

A pesquisa envolveu conceitos de aprendizagem, programação neurolinguística, pensamento divergente, estudo de foco, estados de fluxo, conceitos de *hyperlink*, mapas mentais, algoritmos entre outros conceitos. Devido à complexidade dos assuntos, a pesquisa se tornou muito extensa e fugiu dos propósitos de uma graduação de design, inviabilizando o modelo que, nesse momento se tratava mais do estudo do comportamento do projetista do que do próprio projeto em si.

A escolha então foi mudar o viés da pesquisa, ao passo que essa tinha se tornado superficial e só seria mais um modelo propondo uma “receita de criatividade” de embasamento duvidoso.

A mudança na proposta surgiu de um misto de oportunidade, necessidade e embasamento teórico quase involuntário. O foco da pesquisa se tornou então, a realidade de muitos designers, principalmente quando se trata do desenvolvimento de projetos pessoais e/ou autônomos. O objetivo então se tornou buscar um modelo de possibilidade para quem não tem recursos financeiros suficientes, equipe e tempo e necessita mais do que ninguém ser extremamente efetivo.

À medida que as informações foram chegando ao projeto por meio das pesquisas, novamente o trabalho começou a ganhar volume. Tanto que foi

necessário suprimir algumas funcionalidades do *framework* por terem o conceito muito complexo, como os indicadores de desempenho, a matriz de gravidade / urgência / tendência, noções de mapas estratégicos, conceitos de *branding*, entre outros.

Contudo, uma das grandes vantagens, e na opinião do autor deste trabalho o ponto forte do projeto é o modelo baseado em um *framework* que permite a adição de todos os módulos que foram removidos da pesquisa por algum motivo que podem ser adicionados, dependendo da aplicação e da natureza do projeto que se está desenvolvendo.

Em relação aos resultados, o modelo apresentado alcançou os objetivos propostos de uma forma metódica. O acompanhamento frequente propõe o desenvolvimento ágil, sem perder de vista o planejamento e permitindo que o projeto seja constantemente aprimorado por meio das entregas regulares e por consequência dos *feedbacks* recebidos, apontando para novas possibilidades.

Mas, pela natureza do projeto, o modelo ainda é bem instável e requer testes e ajustes, assim como adição de novas ferramentas e a aplicação em diferentes contextos e segmentos. Algumas melhorias pontuais podem ser destacadas, por exemplo, na etapa de planejamento o processo ainda é meio confuso e parece se sobrepôr em alguns momentos.

O próximo passo é aprimorar o modelo a fim de transcrevê-lo para sua proposta original, um *software* de gerenciamento de projeto.

Outra linha possível é revisitar a pesquisa inicial, abordando agora a melhoria de rendimento da equipe a partir da programação neurolinguística e do estudo do comportamento biológico e neurológico do projetista e dos clientes, buscando melhores resultados baseando-se no estudo do foco, da redução de estresse e a estimulação da criatividade.

Todas essas linhas possibilitam boas oportunidades de continuidade em um mestrado, que é o possível futuro dessa presente pesquisa.

REFERÊNCIAS

AGUIAR, Giancarlo de França; PEINADO, Jurandir. **Revista da Vinci**. 3 ed. Curitiba: Universidade Positivo, 2007

BAXTER, Mike. **Projeto de Produto**. São Paulo: Edgard Blücher, 2000.

BOMFIM, G. A. **Metodologia para desenvolvimento de projetos**. João Pessoa: Editora Universitária / UFPB, 1995.

BONSIEPE, Gui. **A tecnologia da tecnologia**. São Paulo: Editora Edgard Blücher, 1983.

BROWN, Tim. **Design Thinking**. Rio de Janeiro: Campus, 2010.

BÜRDEK, Bernhard E. História, **Teoria e Prática do Design de Produtos**. São Paulo: Edgard Blücher, 2006.

CAMPOS, Vicente Falconi. **TQC: Controle da qualidade total**. Nova Lima: INDG Tecnologia e Serviços, 2004.

COELHO, Luiz Antonio L. (Org.). **Design Método**. Rio de Janeiro: PUC/Rio; Teresópolis: Novas Idéias, 2006.

European Institute for Brand Management. **Brandkey**. Disponível em: <http://www.eurib.org/fileadmin/user_upload/Documenten/PDF/Positionering_ENGELS/r_-_Brand_Key__EN_.pdf> Acesso em 05 dez. 2014.

GOMES, Luciano. **5W2H: Ferramenta para a elaboração de Planos de Ação**. Disponível em: <<http://blog.iprocess.com.br/2014/06/5w2h-ferramenta-para-a-elaboracao-de-planos-de-acao/>>. Acesso em 05 dez. 2014.

GONÇALVES, Carlos H. et al. **Aplicação do masp (qc story) no ciclo básico dos cursos de engenharia de uma instituição pública de ensino superior**. XXXIX Congresso Brasileiro de Educação em Engenharia, Blumenau, 2011.
INNOVATION EXPERTS. **Brandkey - A chave para gerenciar sua marca**. 2014.

KELLEY, Tom. **A arte da inovação**. São Paulo: Futura, 2001.

LÖBACH, Bernd. **Design Industrial: bases para configuração dos produtos industriais**. Rio de Janeiro: Edgard Blücher, 2001.

MANIFESTO PARA DESENVOLVIMENTO ÁGIL DE SOFTWARE. Manifesto para desenvolvimento ágil de software.

Disponível em: <<http://www.agilemanifesto.org/iso/ptbr/>>.

Acesso em 08 dez. 2014

MESTRINER, Fábio. **Design de embalagens**. São Paulo: Makron Books, 2005.

MUNARI, Bruno. **Das coisas nascem coisas**. São Paulo: Martin Fontes, 1998.

MUNARI, Bruno. **Das coisas nascem as coisas. Apontamentos por uma metodologia projetual**. Barcelona: Editorial Gustavo Gili, 1983.

NAKAGAWA, Marcelo. **5W2H – Plano de ação para empreendedores**. Disponível em: <http://cmsempreenda.s3.amazonaws.com/empreenda/files_static/arquivos/2014/07/01/5W2H.pdf>.

Acesso em 03 jun. 2014.

OSBORN, Alex F. **O poder criador da mente**. São Paulo: Ibrasa, 1960.

OSTERWALDER, Alexander. **Value Proposition Design**. São Paulo: HSM do Brasil, 2014.

PROJECT BUILDER, **7 segredos matadores para a criação de dashboards estratégicos**. Disponível em <<http://academia.projectbuilder.com.br/conteudo>>.

Acesso em 06 jun. 2015

RODRIGUES, Jorge Luiz B. **Metodologia de Análise de Solução de Problema – MASP – Como Impulsionador da Competitividade**. Disponível em:

<http://www.techoje.com.br/site/techoje/categoria/detalhe_artigo/1731>.

Acesso em 07 dez. 2014.

ROVERI, Vinicius. et al. **MASP – Método de Análise e Solução de Problemas – aplicado à gestão das trilhas da Estação Ecológica Juréia Itatins/Núcleo Arpoador –**

Município de Peruíbe – SP. **Encontro de Pós-Graduação Lato Sensu e Stricto Sensu da Unisanta**, Santos, 2012.

SABBAGH, Rafael. **SCRUM - Gestão ágil para projetos de sucesso**. Casa do Código, 2013.

SANTOS, Valério G. V. **A administração científica e sua colaboração para as organizações do século XXI**. Aracruz, 2013

SCRUMEX. **Elaborando um roadmap do produto**. Disponível em: <<http://scrumex.com.br/blog/?p=1019>>.

Acesso em 08 dez. 2014.

SPARK IDEAS. **Brandkey**. Disponível em: <<https://sparkideasnl.wordpress.com/2011/03/29/brand-key/>> Acesso 08 dez. 2014.

STOLARSKI, Andre. Alexandre **Wollner e a Formação do Design Moderno no Brasil**. COSAC & NAIFY EDICOES LTDA., 2005. 1 videodisco (85 min).

VASCONCELOS, Luis et al. Um Modelo de Classificação para Metodologias de Design. **9º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design**, São Paulo, 2010.