



IGOR GOMES MORAES

**PROCESSO DE DESIGN EM UMA EMPRESA
QUE ADOTA METODOLOGIA ÁGIL**

LAVRAS – MG

2019

IGOR GOMES MORAES

**PROCESSO DE DESIGN EM UMA EMPRESA QUE ADOTA
METODOLOGIA ÁGIL**

Relatório de estágio supervisionado apresentado à
Universidade Federal de Lavras, como parte das
exigências do Curso de Ciência da Computação,
para a obtenção do título de Bacharel



Prof. Dr. André Vital Saúde

Orientador

LAVRAS – MG

2019

IGOR GOMES MORAES

**PROCESSO DE DESIGN EM UMA EMPRESA QUE ADOTA
METODOLOGIA ÁGIL**

Relatório de estágio supervisionado apresentado à
Universidade Federal de Lavras, como parte das
exigências do Curso de Ciência da Computação,
para a obtenção do título de Bacharel

APROVADA em 04/10/2019.

Fernando Henrique dos Santos Rocha LEMAF
Juliana Galvani Gregghi UFLA

Prof. Dr. André Vital Saúde
Orientador

**LAVRAS – MG
2019**

Dedico à minha família, que é minha base, e sempre lutaram e estiveram ao meu lado nas etapas importantes da minha vida.

AGRADECIMENTOS

Agradeço a Deus, aos meus pais Michelle e Junior, minha irmã Yara, a Universidade Federal de Lavras, principalmente ao Departamento de Ciência da Computação, a todos os meus professores da Ufla e de Três Pontas, aos meus amigos, aos meus colegas de curso e de estágio, ao Lemaf pela oportunidade de estágio que deu origem a este trabalho. Obrigado a todos vocês, pois cada um em sua maneira pode me ajudar a realizar este trabalho e ser uma pessoa e um profissional melhor.

RESUMO

As metodologias ágeis estão cada dia mais presentes nas empresas de desenvolvimento de *software*. A partir de observações no processo de *design* utilizado no LEMAF, organização onde foi realizado o estágio relatado nesse documento, verificou-se a necessidade de utilizar uma metodologia atualizada para o desenvolvimento de soluções de *design*. Diante do contexto apresentado, o principal objetivo desse trabalho será mostrar como o *Design Sprint 2.0*, adaptado à realidade da organização, contribuiu para que as soluções de *design* de interface fossem desenvolvidas com maior rapidez e eficiência. O *Design Sprint* é uma metodologia ágil criada por Jake Knapp, *ex-designer* da Google, que utiliza conceitos do *Design Thinking* e visa em cinco dias construir algo tangível e testável. No LEMAF essa metodologia foi adaptada e foi criado o *Design Studio*, que também é realizado em quatro dias e é possível obter resultados semelhantes ao *Design Sprint 2.0* na concepção de soluções de *design* de interface. Como resultado pôde-se observar que após adotar o novo processo de *design* as soluções foram criadas com maior sucesso e chance de aprovação pelas partes interessadas. A partir desse trabalho foi possível concluir que o uso de uma metodologia ágil como o *Design Sprint 2.0* é imprescindível quando se trata de solucionar um problema de interface e usabilidade para o desenvolvimento de uma nova funcionalidade ou protótipo de alta fidelidade em uma organização.

Palavras-chave: *Design Sprint*. Metodologia ágil. *Design* de interface.

ABSTRACT

Agile methodologies are increasingly present in software development companies. From observations in the design process used in LEMAF, organization where the internship reported in this document was done, was verified the need to use an updated methodology to develop design solutions. Against the context presented, the main goal of this paper will be to show how Design Sprint 2.0, adapted to the organization reality, contributed to develop the design solutions more quickly and efficiently. Design Sprint is an agile methodology created by the former Google designer, Jake Knapp, who uses concepts from Design Thinking and aims at five days to build something tangible and testable. In LEMAF this methodology was adapted and the Design Studio was created, which is also done in four days and it is possible to obtain results similar to Design Sprint 2.0 in the conception of interface design solutions. As a result, it could be observed that after adopting the new design process the solutions were created with greater success and chance of approval by the stakeholders. From this work it was possible to conclude that the use of an agile methodology such as Design Sprint 2.0 is essential when it comes to solving an interface and usability problem for the development of a new functionality or high fidelity prototype in an organization.

Keywords: Design Sprint. Agile Methodology. Interface Design.

LISTA DE FIGURAS

Figura 2.1 – Organograma Lemaf.	18
Figura 3.1 – Duplo diamante.	27
Figura 3.2 – Divisão semanal do <i>Design Sprint</i>	30
Figura 3.3 – Divisão semanal do <i>Design Sprint 2.0</i>	32
Figura 3.4 – Protótipo de baixa fidelidade de uma tela de login.	34
Figura 3.5 – <i>Storyboard</i> do processo de login em uma rede social.	35
Figura 3.6 – Protótipo de alta fidelidade da uma tela de login de uma Rede Social.	35
Figura 5.1 – Divisão dos dias no <i>Design Studio</i>	39
Figura 6.1 – Protótipo de baixa fidelidade produzido no segundo dia do <i>Design Studio</i>	41
Figura 6.2 – Protótipo de alta fidelidade da tela inicial do CBPA.	42
Figura 6.3 – Protótipo de alta fidelidade da tela de fichas de inspeção de cada barragem.	42
Figura 6.4 – Protótipo de alta fidelidade da tela de barragem selecionada.	43
Figura 6.5 – Protótipo de alta fidelidade do cadastro de uma nova ficha de inspeção.	43
Figura 6.6 – Protótipo de alta fidelidade da tela de envio de imagens no cadastro.	44

SUMÁRIO

1	INTRODUÇÃO	15
2	ORGANIZAÇÃO ONDE FOI REALIZADO O ESTÁGIO . .	17
2.1	Lemaf	17
2.2	Estrutura e Organograma Institucional	17
2.3	Atividades Realizadas no Estágio	21
3	REFERENCIAL TEÓRICO	25
3.1	Metodologias Ágeis	25
3.2	Design Thinking	26
3.3	Design Sprint	27
3.4	Metodologias ágeis e <i>Design Sprint</i>	30
3.5	<i>Design Sprint 2.0</i>	31
3.6	Prototipação	33
3.6.1	Protótipo de baixa Fidelidade	33
3.6.1.1	<i>Storyboard</i>	33
3.6.2	Protótipo de alta fidelidade	34
4	METODOLOGIA	36
5	DESIGN STUDIO	37
5.1	Integrantes	37
5.2	Definição de algumas atividades realizadas no <i>Design Studio</i> .	37
5.2.1	<i>How Might We (HMW)</i>	37
5.2.2	<i>Crazy 8's</i>	38
5.3	Execução	38
6	RESULTADOS	40
6.1	Participação do autor	40
7	CONCLUSÃO	45
	REFERÊNCIAS	46

1 INTRODUÇÃO

Os métodos ágeis já se consolidaram como o novo estado de prática na indústria de software. A maioria das empresas de tecnologia da informação já adotam essa prática e isso também vem se expandindo por outras áreas. Os métodos ágeis trazem grandes benefícios para as organizações que os adotam, dentre eles: a habilidade para lidar com mudanças, melhor visibilidade dos projetos e aumento da produtividade do time. Também é possível citar que ao utilizar algum método ágil, as equipes se sentem mais motivadas e há uma melhora relevante na qualidade de vida no trabalho (BORGES, 2017).

Criado em 2009, baseado no *Design Thinking*, e posteriormente consolidado pelo *Google Ventures* em seus programas de impulsionamento de *startups*, o *Design Sprint* é uma metodologia que tem o objetivo de, ao final de seus ritos, criar um protótipo e testá-lo para descobrir se a ideia proposta é ou não viável. Esse processo de validação, que pode durar meses se feito nos métodos tradicionais, no *Design Sprint* dura apenas cinco dias (TEIXEIRA, 2015), ou quatro dias, se utilizado a versão 2.0, atualizada pela equipe da AJ&Smart.

Este trabalho relata a criação e aplicação em um projeto real do *Design Studio*, um método de solução de problemas de *design* de interface, baseado no *Design Sprint*, para o LEMAF, organização onde foi realizado o estágio.

O Capítulo 2 apresenta a organização onde o estágio foi realizado com as atividades que o estagiário exerceu. O referencial teórico aborda os conceitos utilizados para o relatório. O Capítulo 4 traz a metodologia para a criação do *Design Studio*, que será descrito e exemplificado através de um projeto no Capítulo 5. Os resultados serão apresentados no Capítulo 6. No Capítulo 7, são apresentadas as conclusões. Por fim, as referências bibliográficas, no Capítulo 8.

2 ORGANIZAÇÃO ONDE FOI REALIZADO O ESTÁGIO

Este capítulo tem o propósito de apresentar a empresa onde o estágio foi realizado, abordando sua fundação, serviços prestados e sua organização.

2.1 Lemaf

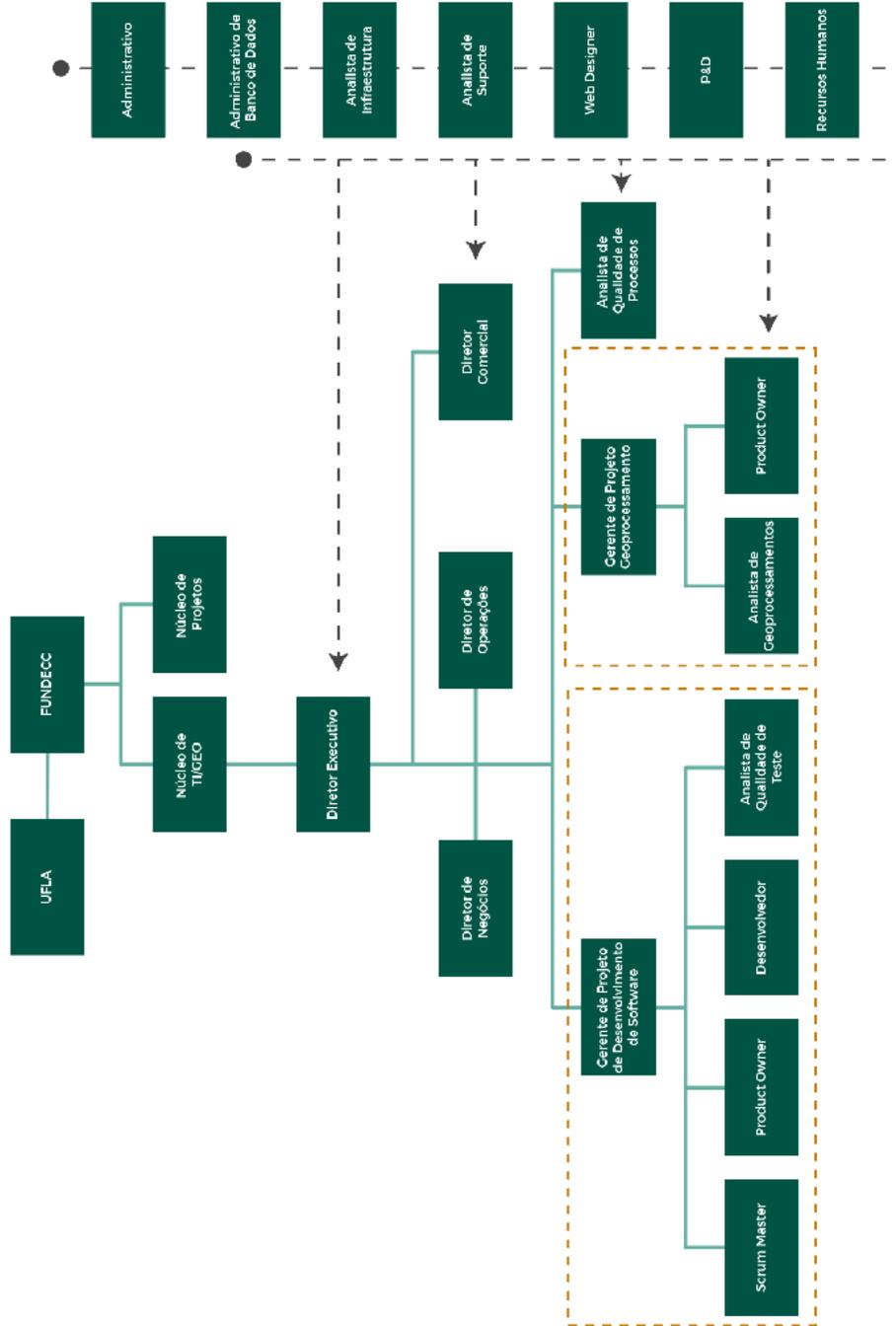
Inserido no Departamento de Ciências Florestais (DCF) da Universidade Federal de Lavras (Ufla), o Laboratório de Estudos e Projetos em Manejo Florestal (Lemaf) tem como objetivo efetuar pesquisa, ensino e extensão. Fundado em 2004, conduz diversos projetos em parceria e convênio com órgãos estaduais e federais, bem como a iniciativa privada. Estes estudos subsidiam o crescimento, desenvolvimento e aperfeiçoamento do ensino (LEMAF, 2018).

2.2 Estrutura e Organograma Institucional

O Lemaf utiliza como metodologia para gestão e planejamento do desenvolvimento dos seus softwares o *Scrum*. O *Scrum* é uma metodologia ágil, em que os projetos são divididos em ciclos chamados de *Sprints*. A *Sprint* representa um *Time Box* dentro do qual um conjunto de atividades deve ser executado. Metodologias ágeis de desenvolvimento de software, como o *Scrum*, são iterativas, ou seja, o trabalho é dividido em pequenas entregas contínuas. As funcionalidades a serem implementadas em um projeto são mantidas em uma lista que é conhecida como *Product Backlog*. No início de cada *Sprint*, faz-se uma *Sprint Planning*, ou seja, uma reunião de planejamento na qual o *Product Owner* prioriza os itens do *Product Backlog* e a equipe seleciona as atividades que ela será capaz de implementar durante a *Sprint* a ser iniciada. As tarefas alocadas em uma *Sprint* são transferidas do *Product Backlog* para a *Sprint Backlog* (LEMAF, 2018).

O Lemaf possui as seguintes funções definidas (LEMAF, 2018). Também possível visualizar no organograma na Figura 2.1.

Figura 2.1 – Organograma Lemaf.



- Equipe de gestão:
 - Diretor de tecnologia da informação (TI): É responsável por tudo que ocorre na equipe de tecnologia de informação. Esse Diretor também é o responsável por fechar novos contratos e estar em contato direto com clientes, antigos e novos.
 - Diretor Comercial: É responsável por mapear oportunidades de expansão de mercado, gerir o pós-venda e acompanhar relatórios financeiros. Está sempre viajando para articular e gerenciar o relacionamento com clientes.
 - Gerente de Projetos: É responsável pela gestão interna das equipes de desenvolvimento, acompanhamento de prazos, alocação de colaboradores, estar sempre em contato com as equipes a fim de alinhar os processos, prazos e status de cada projeto. O gerente deve garantir condições de ambiente, organização e prazos.
 - Analista de Recursos Humanos: Responsável pela mediação da relação entre colaboradores e entre colaborador/gerência. Trabalha no sentido de mediar conflitos e resolver diferenças. Entre suas atribuições está a aplicação das avaliações qualitativas, gerar *feedbacks*, desenvolver processos de seleção e acompanhar o plano de melhorias.
- Composição das equipes de desenvolvimento:
 - *Scrum Master*: É o guardião do processo. Ele é responsável por fazer o processo correr bem removendo os obstáculos que atrapalham a produtividade da equipe, organizando e facilitando as reuniões.
 - *Product Owner* (PO): É o membro da equipe que entra em contato com o cliente e deve entender as necessidades e prioridades de todos os envolvidos na empresa para agir como seu porta-voz. Neste sentido, ele atua como um gestor do produto, garantindo que a solução correta é

desenvolvida. Do outro lado, o *Product Owner* deve se comunicar com o time *scrum* para ajudar na ordem em que o produto será construído. O *Product Owner* também deve garantir que os critérios para aceitação do produto estão especificados e que os testes que verificam esses critérios foram executados para determinar que o produto (ou *release*) possa ser considerado como pronto ao final da *sprint*.

– Líder Técnico: Desenvolvedor mais experiente do time que fica responsável por auxiliar a equipe com relação a dificuldades e impedimentos técnicos. É de sua responsabilidade auditar os códigos fonte e garantir a qualidade do código desenvolvido.

– Desenvolvedor: Faz programação ou faz manutenção de *software* e códigos dos sistemas desenvolvidos pelo Lemaf. É de sua responsabilidade participar de todos os ritos do *scrum*.

– Analista de Qualidade de Software: Elabora testes de interface, usabilidade, funcionalidade e integração, gerencia e investiga *bugs*, cria documentos de testes, manuais e aplica treinamentos aos cliente.

- Equipe de apoio às equipes de desenvolvimento:

– Administrador de Banco de Dados: Responsável por gerenciar, instalar, configurar, atualizar e monitorar um banco de dados ou sistemas de bancos de dados.

– Analista de Infraestrutura: Presta suporte e manutenção de servidores e data center, planeja capacidade e desenvolve projetos de melhorias dos serviços corporativos. Realiza criação de usuários e manutenção de grupos para concessão de privilégios com permissão de acesso. Instala servidores, incluindo configuração de *clusters* para alta disponibilidade com acompanhamento e cumprimento das políticas dos re-

- cursos computacionais. Elabora documentação de procedimentos operacionais.
- Analista de Suporte: Presta suporte técnico aos usuários dos sistemas desenvolvidos pelo Lemaf. Gera relatos de erros/melhorias reportados pelo cliente no Mantis, sistema livre de controle de ocorrências utilizado pelo Lemaf para este fim. Escreve manuais e dá treinamentos para os clientes.
 - Pesquisa e Desenvolvimento: É o núcleo responsável por estar constantemente desenvolvendo pesquisas e testes com novas tecnologias ou abordagens para que esteja sempre alinhado com o que há de mais moderno e efetivo em seu ramo de atuação de forma a apresentar produtos de excelência.
 - *Web Designer*: Criar e adaptar propostas de identidade visual dos projetos, criar manual de identidade visual. Modelar interações e comportamentos do sistema. E auxiliar as equipes em decisões sobre o *design* do projeto.
 - Secretaria: É o setor responsável por organizar administrativamente o Lemaf, como viagens, orçamentos, compras, pagamento de diárias, gerenciamento de folha de ponto assim como pagamento da bolsa dos estagiários.

2.3 Atividades Realizadas no Estágio

O estagiário atuou como *Web Designer*, membro da equipe de apoio às equipes de desenvolvimento, conforme estrutura apresentada na Seção anterior. Teve suas atividades enfatizadas em projetos solicitados pelos governos dos estados do Pará, Tocantins, Amazonas, Acre, Minas Gerais, Paraná, Rondônia, Rio Grande do Sul, Santa Catarina e governo federal com foco em manejo florestal,

controle ambiental e prevenção de acidentes em barragens, etc. Dentre essas atividades, as que se destacam são:

- Criar identidades visuais para vários propósitos: Identidade visual é o conjunto de elementos como fonte, cor, logo e qualquer outro elemento visual a fim de evocar certos sentimentos e experiências com a marca.
- Definir guias de estilo e criação de manuais: Para que a identidade visual seja mantida dentro do sistema e também em reprodução gráfica da marca, é criada a guia de estilo e os manuais de utilização dos artefatos, fontes e cores.
- Definir os comportamentos e iterações no sistema: O *designer* tem a responsabilidade de definir como a interface do sistema apresentará em diferentes situações, animações dos componentes e telas.
- Participar de ritos do Scrum: O *designer* geralmente está uma *sprint* à frente da equipe de desenvolvimento, já que deve entregar o protótipo para ser desenvolvido, e é na *planning* onde verá qual tela deverá ter seu protótipo em alta fidelidade criado.
- Participar das avaliações heurísticas: junto ao time de *designers*, o estagiário avalia os sistemas desenvolvidos baseando-se nas heurísticas de Nelsen, como documentado por Rodrigues (2018) em seu relatório de estágio.
- Participar dos *Design Studios*: Quando o P.O. vê a necessidade de ser feita uma *sprint*, para o desenvolvimento de algum produto ou nova funcionalidade, o *designer* é alocado nas fases de entrevista, prototipagem da solução e validação com os *stakeholders*¹.

¹ O termo *Stakeholder* designa uma pessoa que se interessa e possui participação financeira em uma empresa ou algum negócio que, neste caso, trata-se do cliente.

Esta última atividade será descrita posteriormente neste trabalho com mais aprofundamento, tendo em vista que o estagiário não apenas participou dos *Design Studios* como também contribuiu para a definição do processo de *Design Studios* do Lemaf.

3 REFERENCIAL TEÓRICO

Neste capítulo serão apresentados os conceitos de metodologias ágeis, *Design Thinking*, *Design Sprint* e sua atualização, o *Design Sprint 2.0*, utilizado como base para a criação do *Design Studio*, e também definições de técnicas utilizadas na criação de soluções de *design* de interface que serão mencionadas em outros capítulos.

3.1 Metodologias Ágeis

De acordo com Sommerville (2011), a ideia de metodologia ágil foi apresentada no final da década de 1990, pois já se via a necessidade de desenvolver *software* rapidamente e com processos capazes de lidar com mudanças de requisitos. Os métodos ágeis se caracterizam por serem incrementais e cada incremento é, geralmente, uma pequena versão do sistema que é gerada e disponibilizada ao cliente. A base para os métodos ágeis é o manifesto ágil, que foi criado pelos principais desenvolvedores desses métodos. Os princípios desse manifesto são:

- Indivíduos e interações são mais importantes que processos e ferramentas
- Software em funcionamento é mais importantes que documentação abrangente
- Colaboração do cliente é mais importante que negociação de contrato
- Respostas a mudanças são mais importantes que seguir um plano

Um processo de desenvolvimento é considerado ágil quando ele é incremental – as entregas são rápidas e pequenas; cooperativo – partes interessadas do projeto estão sempre em comunicação, incluindo os clientes; simples – o processo é fácil de se aprender e documentar; finalmente, é adaptável – em qualquer estágio do processo é possível fazer alterações. (ABRAHAMSSON et al., 2017)

3.2 Design Thinking

Os conceitos do *Design Thinking* precisam ser definidos para que seja possível entender em que o *Design Sprint* foi baseado para ser desenvolvido.

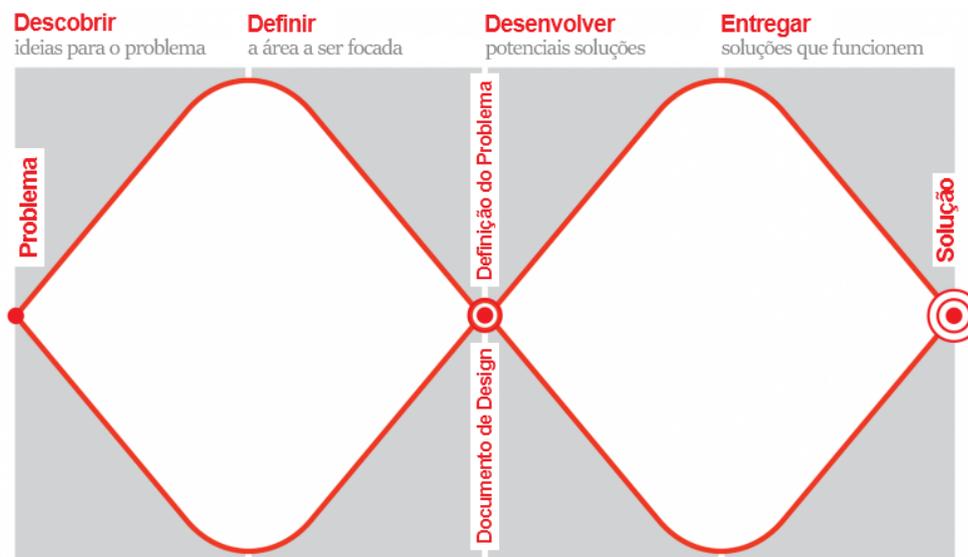
De acordo com IDEO (2019), *Design Thinking* é um modelo mental centrado no ser humano, o que significa que os produtos, processos internos de empresas, serviços, etc, são criados pensando nas pessoas que os utilizarão. O processo começa com a compreensão e sobre fazer as perguntas certas sobre o problema a ser resolvido.

Os três pilares do *Design Thinking* são:

- Empatia - Compreender as necessidades daqueles que utilizarão o que será projetado.
- Ideação - Gerar muitas ideias. *Brainstorming* é uma técnica, mas existem muitas outras.
- Experimentação - Utilizar protótipos para testar as ideias.

O *Design Thinking* é um processo em quatro etapas e uma maneira simples de compreendê-lo é usando o Duplo Diamante ilustrado na Figura 3.1. No processo há pontos de divergência (primeira e terceira fase), para criar várias possibilidades de ideias, e pontos de convergência (segunda e quarta fase), onde as ideias são refinadas para que sejam escolhidas as melhores. O Duplo Diamante é definido da seguinte forma: primeiro tem-se a fase de descoberta, onde deve-se entender a fundo o problema e olhar o mundo de uma maneira nova, afim de entender as principais necessidades do usuário. Na próxima fase, os objetivos finais são alinhados com as descobertas e então é definido um ponto de ação a ser focado. Na terceira fase começa um período de desenvolvimento, onde as soluções são prototipadas e testadas, isso pode ser repetido até que tenha um resultado satisfatório. E então, após todos os processos anteriores, há a entrega, onde tudo que foi levantado se converge em uma solução (COUNCIL, 2015).

Figura 3.1 – Duplo diamante.



Fonte: Adaptado de (COUNCIL, 2015)

3.3 Design Sprint

Segundo Knapp, Zeratsky e Kowitz (2016), o *Design Sprint* é um processo de cinco dias para resolver questões críticas de negócios por meio de design, prototipagem e teste de ideias com os clientes. Ele combina as ideias de estratégia de negócios, inovação, *design*, psicologia, entre outras, que são combinadas completamente em um passo a passo pronto, que pode ser usado por qualquer equipe, independentemente do problema a ser resolvido.

Sua primeira versão foi criada em 2009 por Jake Knapp, com o objetivo de melhorar os processos e os resultados gerados pelos *brainstorms*¹ convencionais realizados pelas equipes. Posteriormente foi evoluído no *Google Ventures*, um programa da Google para impulsionar *startups*, no qual as *sprints* pudessem ajudar as jovens empresas a testar suas ideias antes de lançar seus produtos.

¹ Brainstorm é uma dinâmica de grupo para de geração de ideias, solucionar um problema específico, estimular o pensamento criativo e juntar informações.

Três coisas precisam estar bem definidas antes de se começar um *Design Sprint* (KNAPP; ZERATSKY; KOWITZ, 2016). Em primeiro lugar, o problema a ser resolvido deve ser reconhecido e compreendido pela equipe. Em segundo lugar, uma equipe para executar a *sprint* deve ser formada. Recomenda-se que a equipe não tenha mais que sete integrantes, pois um número maior de pessoas dificulta a eficiência e foco nas tarefas. Além disso, as equipes devem ser multifuncionais e mistas, e deve haver um especialista em cada tópicos específico presente durante as *sprints*, isso porque pessoas com diferentes habilidades e características na mesma equipe podem fornecer informações importantes e ajudar a analisar o problema de diferentes perspectivas. Finalmente, o tempo e o espaço devem ser organizados. A equipe a realizar o *Design Sprint* deve ter suas tarefas rotineiras concluídas para ficarem focadas durante uma semana de trabalho exclusivo da *sprint*. Existem três papéis-chave que alguns integrantes da equipe devem assumir:

- Decisor – ele será responsável por tomar as decisões mais importantes durante a *sprint*.
- Facilitador – irá orientar toda a equipe durante a *sprint*, gerenciar o tempo, as discussões e resumir as conversas para que o cronograma da *sprint* siga como planejado.
- Guardiã da metodologia – ele é importante, principalmente, nas primeiras *sprints*, pois será responsável por assegurar que todos os ritos e hábitos sejam seguidos para que não haja atraso durante as atividades.

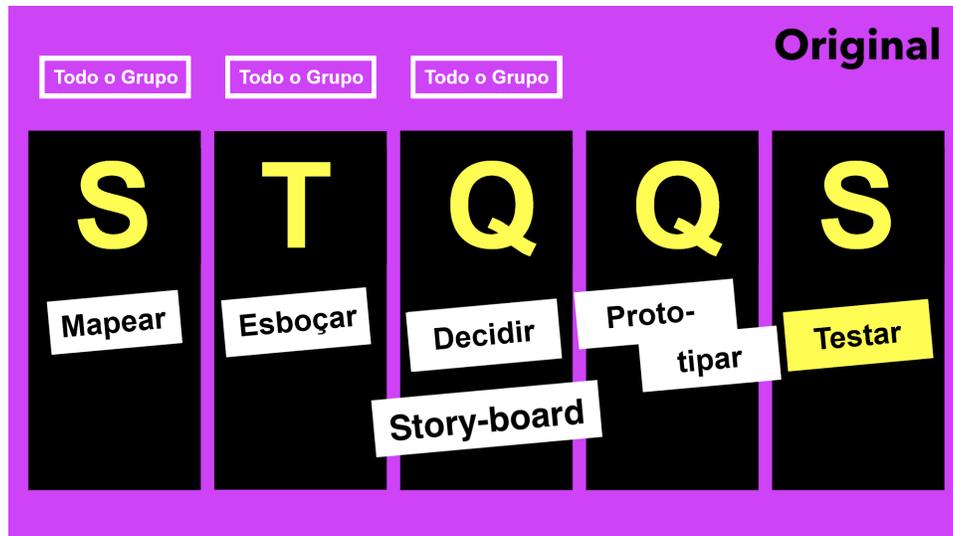
Os detalhes e tarefas de cada dia serão descritos abaixo e também na Figura 3.2:

1. Segunda-feira - a questão a ser resolvida é escolhida, e um mapa de todo o desafio sobre ela é criado para que a equipe toda tenha visão detalhada sobre ela. Os integrantes compartilham seus conhecimentos e opiniões,

baseados em suas especialidades, e o objetivo do *sprint* é escolhido – o que exatamente precisa ser resolvido nesses cinco dias e quais são os desafios envolvidos. Todo o processo é documentado utilizando um quadro e *post-its*.

2. Terça-feira - o objetivo desse dia é criar diferentes soluções para o problema. Para isso a equipe deve realizar uma pesquisa sobre ideias, inspirações e possíveis estudos de caso. Cada integrante expõe sua própria ideia e depois fazem esboços que serão coletados para serem utilizados na quarta-feira.
3. Quarta-feira - todos fazem críticas e dão suas opiniões em cada esboço coletado no dia anterior, em seguida eles são votados e a solução mais pertinente para ser trabalhada é escolhida pelo Decisor. Baseado em todas as ideias é criado um *storyboard*, definido na seção 3.6.1.1. O protótipo a ser criado terá como base a solução escolhida pelo Decisor e o *storyboard*, mas as outras ideias apresentadas também poderão ser aproveitadas.
4. Quinta-feira - este dia é dedicado a desenvolver o protótipo, que pode ser feito em computador, em papel, impresso em 3D, etc. Varia de acordo com o contexto do problema a ser resolvido.
5. Sexta-feira - finalmente o protótipo é testado por pelo menos cinco pessoas, que não devem ser pessoas próximas, possíveis clientes, por exemplo, são os mais indicados. Eles são observados durante o teste e entrevistados, isso permite que a equipe possa analisar problemas no protótipo, erros que deverão ser corrigidos ou se a ideia proposta não deve ser desenvolvida.

Segundo Teixeira (2015), "*Design Sprint* é uma ótima ideia para agilizar o trabalho de times de design e conseguir validar ideias antes que muito tempo ou

Figura 3.2 – Divisão semanal do *Design Sprint*.

Fonte: Adaptado de (COURTNEY, 2018)

dinheiro sejam investidos nela". Outro ponto citado por ele é a limitação de tempo nas atividades, apesar de em um primeiro momento isso parecer sofrido, é positivo pois deixa os membros envolvidos na *sprint* mais motivados, por terem sempre a sensação de estarem movendo rapidamente.

3.4 Metodologias ágeis e *Design Sprint*

Como mencionado anteriormente, as principais características das metodologias ágeis é que o desenvolvimento do produto acontece rapidamente, cooperativamente e é fácil documentá-lo, ser eficiente e produzir bons resultados o mais rápido possível. Isso corresponde com a estrutura do *Design Sprint*, onde a ideia principal é produzir um protótipo viável em um período de tempo limitado, cinco dias, e muitas das atividades durante cada um dos dias também são colocadas sob certos limites de tempo. Além disso, o *Design Sprint*, também é colaborativo, já que o trabalho é realizado por uma equipe e também não enfatiza muito a documentação, já que todas as ideias são coletadas em um quadro branco, *post-it* e

folhas de papel. Finalmente outra semelhança pode ser novamente observada, ambas as metodologias ágeis quanto o *Design Sprint*, fogem das ideias tradicionais de desenvolvimento (POLIAKOVA, 2017).

3.5 *Design Sprint 2.0*

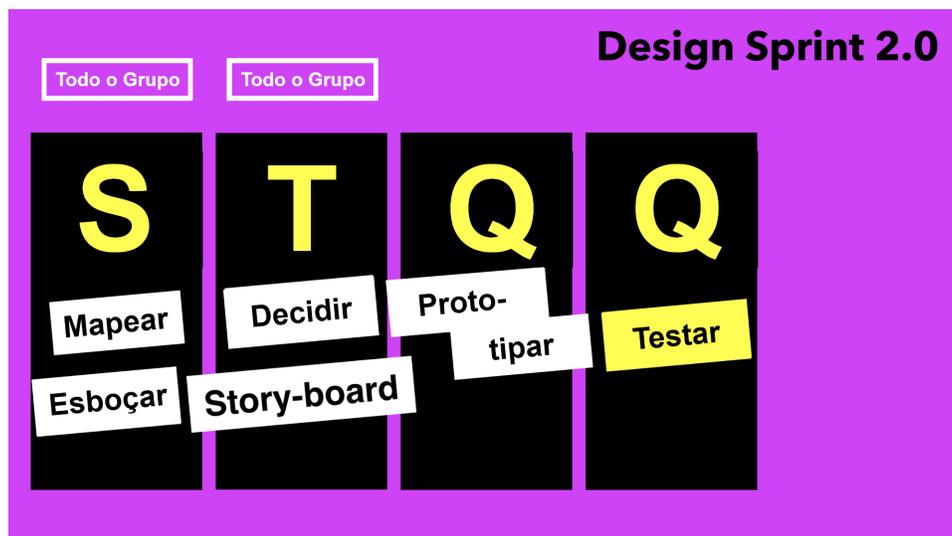
O *Design Sprint 2.0* é a versão mais recente do *Design Sprint* criada pela AJ&Smart, que após executar centenas de *sprints*, viu que se feitas algumas modificações, a metodologia poderia ser ainda mais eficiente (COURTNEY, 2018). Uma das maiores diferenças da versão 2.0, é que ela é otimizada para funcionar não apenas para impulsionar *startups*, como é o foco na versão original, mas também em projetos de organizações já estabelecidas, podendo ser utilizado em times de desenvolvimento de produtos e adição de recursos novos, *marketing* e inovação. Outra diferença que deixa o *Design Sprint 2.0* mais flexível é que o time a realizar o *sprint* só precisa estar completo nos dois primeiros dias, o que é um grande ganho, pois nem sempre nas organizações se tem disponibilidade de desalocar funcionários de suas atividades durante uma semana de trabalho. As atividades também sofreram alteração em suas ordens, o que deixou o fluxo mais lógico, fazendo com que nessa versão a *sprint* dure somente quatro dias (COURTNEY, 2018), descritos a seguir e ilustrados na Figura 3.3:

1. Segunda-feira - No primeiro dia da *sprint* são realizadas as atividades de segunda-feira e parte da terça-feira do *Design Sprint* em um dia só. São realizadas as entrevistas e mapeamento do problema na parte da manhã. À tarde são realizados os esboços das primeiras ideias de solução do problema principal.
2. Terça-feira - Assim como na segunda-feira, neste dias serão realizadas as atividades de dois dias em um. No começo do dia é decidido o que será prototipado baseado em votações da equipe dos esboços realizados

no dia anterior. Após as votações o Decisor votará das ideias que acha mais promissoras. A tarde é realizado o Storyboard que servirá de base para o protótipo.

3. Quarta-feira - Este dia é dedicado somente a prototipar, assim como na quinta-feira do *Design Sprint*
4. Quinta-feira - Novamente assim como na sexta-feira do *Design Sprint*, este dia é somente para testes.

Figura 3.3 – Divisão semanal do *Design Sprint 2.0*.



Fonte: Adaptado de (COURTNEY, 2018)

Segundo o co-fundador da AJ&Smart, Courtney (2018), os únicos dias que devem ser seguidos fielmente e suas atividades imutáveis são os de prototipar e de testar, pois gastam um dia inteiro cada. Nos outros, as atividades podem ser divididas e suas ordens alteradas para que faça mais sentido na *sprint*.

3.6 Prototipação

Um protótipo pode ser qualquer representação que simule o produto final a ser desenvolvido, podendo ser feito de qualquer material, seja de papel ou até mesmo uma parte complexa de um programa de computador. O protótipo permite que os *stakeholders* interajam com o produto visionado para que possam ganhar experiência e explorar o uso. Portanto, um protótipo é uma representação limitada de um *design* que permite aos usuários interagir com ele e verificar se a ideia é suficiente para o que se deseja alcançar (PREECE; ROGERS; SHARP, 2002).

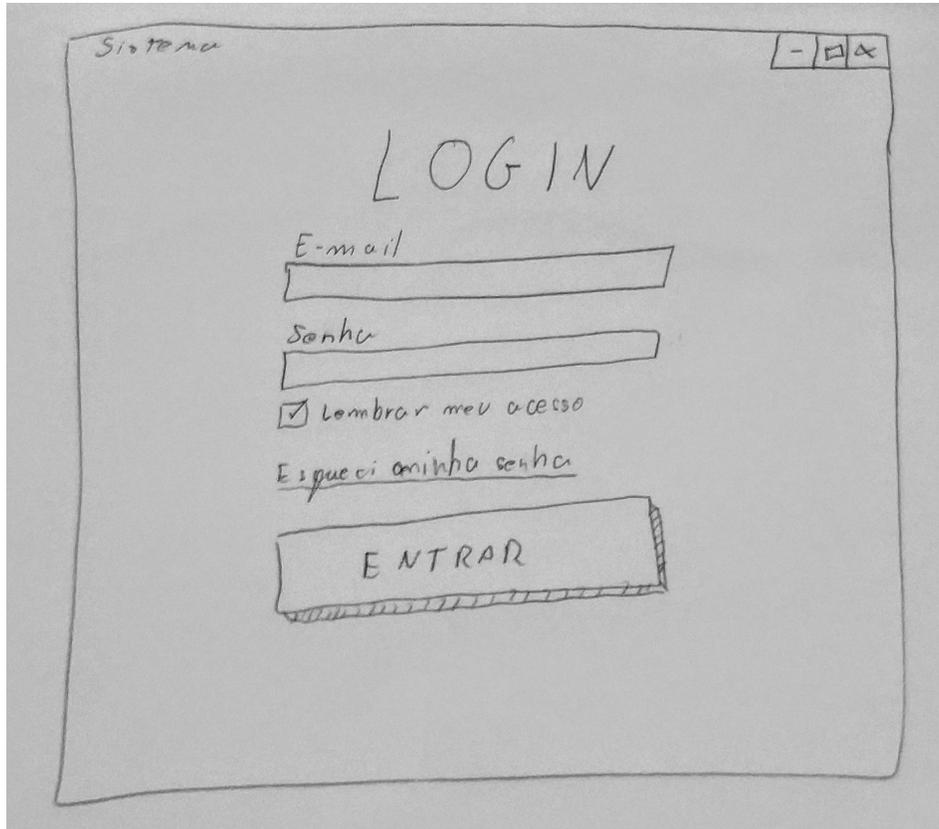
3.6.1 Protótipo de baixa Fidelidade

Um protótipo de baixa fidelidade não deve ser fiel ao produto final e são geralmente feitos de papel ou papelão. São úteis no início do desenvolvimento pois têm baixo custo e tendem a ser rápidos para produzir e modificar, já que sua principal função é testar e explorar novas ideias. Eles nunca fazem parte da versão final do produto (PREECE; ROGERS; SHARP, 2002). A Figura 3.4 mostra um exemplo de protótipo de baixa fidelidade feito em papel.

3.6.1.1 *Storyboard*

Um *storyboard* é um exemplo de protótipo de baixa fidelidade, ele consiste em uma série de esboços mostrando como um usuário pode progredir através de uma tarefa usando o produto que está sendo desenvolvido. Pode ser uma série de telas esboçadas para um sistema ou uma série de esboços de cenas mostrando o usuário executando as tarefas (PREECE; ROGERS; SHARP, 2002). A Figura 3.5 mostra um exemplo de um *storyboard*.

Figura 3.4 – Protótipo de baixa fidelidade de uma tela de login.

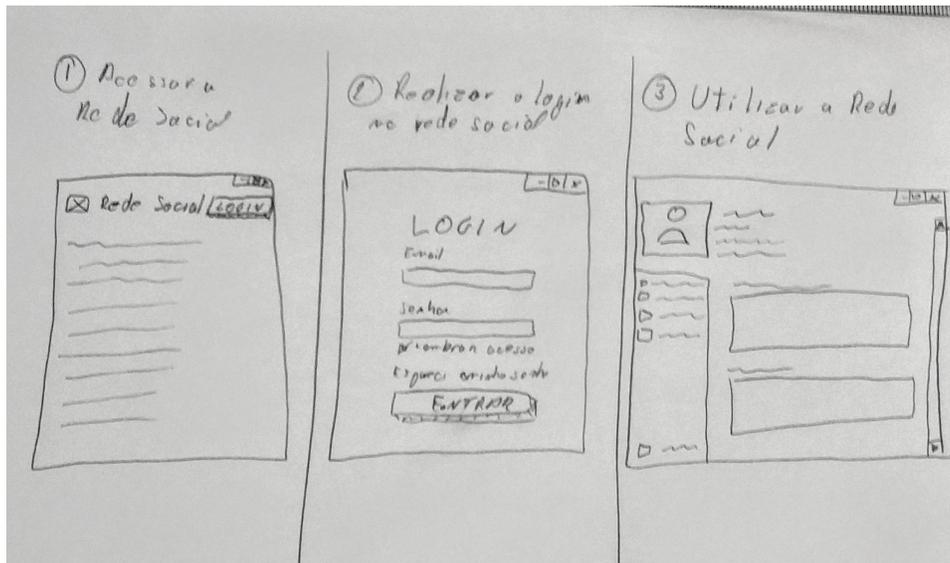


Fonte: Do autor

3.6.2 Protótipo de alta fidelidade

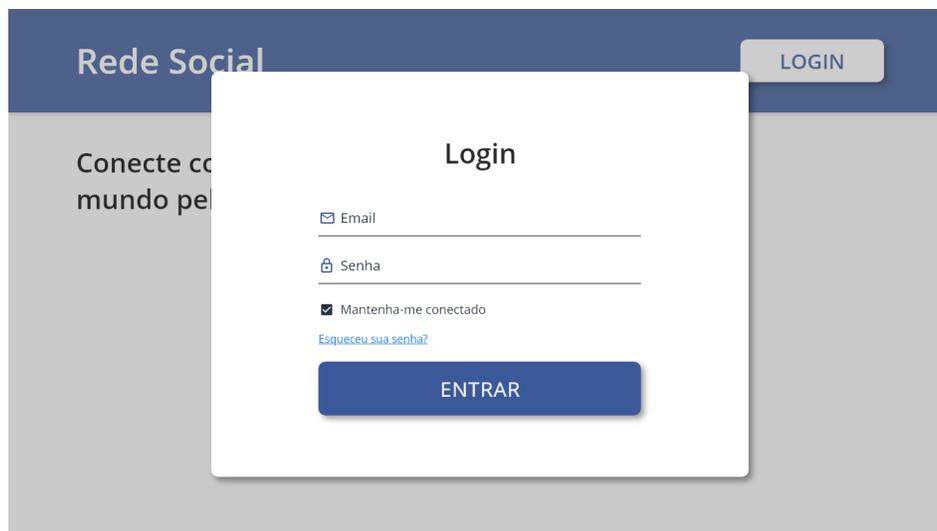
Um protótipo de alta fidelidade, como o nome já diz, deve ser o mais fiel possível ao resultado final. Geralmente é feito do mesmo material do produto, por exemplo, um protótipo de um sistema feito no Adobe XD (ADOBE INC, 2019) tem seu visual e comportamento idêntico ao do sistema. A prototipagem de alta fidelidade é útil para vender ideias a pessoas e testar problemas técnicos (PREECE; ROGERS; SHARP, 2002). A Figura 3.6 mostra um exemplo de um protótipo de alta fidelidade feito no Adobe XD.

Figura 3.5 – Storyboard do processo de login em uma rede social.



Fonte: Do autor

Figura 3.6 – Protótipo de alta fidelidade da uma tela de login de uma Rede Social.



Fonte: Do autor

4 METODOLOGIA

A concepção do *Design Studio* foi dada a partir de um *workshop* de *Design Sprint* realizado na Universidade Federal de Lavras, com início no dia 21 de janeiro de 2019, onde duas equipes do LEMAF participaram, cada uma com um problema diferente a ser resolvido. Um desses problemas era o retrabalho de prototipar e programar componentes básicos que eram sempre utilizados em todos os projetos. Ao final do *sprint* foi obtida a solução de criação do *Design System*, prática que já está sendo adotada e vem trazendo bons resultados nos projetos. O outro problema era a demora no mapeamentos e classificação de imagens de satélite. A solução foi automatizar algumas das etapas do mapeamento utilizando inteligência artificial. O trabalho já está sendo realizado na organização e a expectativa é que o novo processo entre em atividade em breve.

Visto o sucesso nas duas *sprints* executados no *workshop*, o time de *design*, juntamente aos POs, levaram a ideia de reproduzir o método nos projetos de alguns sistemas no LEMAF. A princípio surgiram alguns problemas: (i) a equipe não poderia ficar alocada cinco dias consecutivos para se dedicar ao *sprint* (ii) e alguns membros possuem tarefas diárias obrigatórias, (iii) como na maioria das vezes o problema a ser resolvido é uma nova funcionalidade de um sistema já existente ou que já está em desenvolvimento, cinco dias apenas para a prototipagem foi julgado muito tempo pelos diretores e gerentes. Após uma análise no *Design Sprint*, observação em outras organizações e o conhecimento do sucesso do *Design Sprint 2.0*, os envolvidos optaram por adaptar a metodologia e criar o *Design Studio*, onde em quatro dias o protótipo de alta fidelidade é desenvolvido e validado com o cliente.

O método criado será descrito em detalhes no próximo capítulo.

5 DESIGN STUDIO

Como dito no capítulo anterior, o *Design Studio* é um método baseado no *Design Sprint*, ele prega a colaboração, iteração e criação de ideias. No *Le-maf*, seu propósito é que em quatro dias um novo sistema ou nova funcionalidade tenha seu protótipo de alta fidelidade produzido e validado com os *stakeholders* para começar a ser desenvolvido. Algumas vantagens no *Design Studio* é sua rapidez de execução: a colaboração dos integrantes permite que os problemas sejam compreendidos com maior facilidade, e que haja o compartilhamento de conhecimento, visto que a equipe é formada de pessoas de diferentes áreas e que possuem diferentes pontos de vista.

5.1 Integrantes

Assim como no *Design Sprint*, o *Design Studio* conta com um facilitador, que geralmente é um membro do time de *design*, e um decisor, o PO do projeto, pois é quem está mais inteirado com as regras de negócio. A equipe conta também com integrantes que entendem do assunto a ser abordado, desenvolvedores e outros *designers*.

5.2 Definição de algumas atividades realizadas no *Design Studio*

5.2.1 *How Might We (HMW)*

How Might We, ou em português, Como Nós Podemos, é uma prática que consiste em escrever em *post-its* perguntas que irão fazer com que problemas soem como oportunidades não como algo sem solução. "Como" guia o time a pensar que a solução é possível, "Nós" é para lembrar que o *Design Studio* é realizado em equipe, "Podemos" sugere que não há apenas uma resposta para o problema, o que expande as possibilidades. As perguntas não devem ser muito abrangentes, nem muito específicas. Por exemplo, "Como nós podemos fazer melhores

protótipos?" é uma pergunta muito abrangente, e "Como nos podemos adaptar o *Design Sprint* para que possamos criar protótipos de forma mais eficiente?" é uma pergunta muito específica. "Como nós podemos melhorar o processo de design?" seria uma pergunta mais apropriada.

5.2.2 *Crazy 8's*

O *Crazy 8's* é um exercício que desafia os participantes, individualmente, a fazerem oito esboços simples e diretos de soluções em oito minutos. O objetivo é que uma ideia específica se expanda em 8 diferentes. O tempo curto para se fazer os desenhos faz com que os membros fiquem mais inspirados e a criatividade aflore.

5.3 Execução

O *Design Studio* é realizado da seguinte forma:

- Dia 1 - O primeiro dia é a fase de descoberta: o PO e o *designer* entrevistam o cliente, que passará todos os requisitos e funcionalidades do sistema. Algumas das perguntas essenciais que o PO e *designer* devem fazer são: "O que é o produto ou funcionalidade? O que ele deve fazer?", "Quem irá utilizar a aplicação?", "Qual é o problema que sua aplicação irá solucionar?". Essas perguntas simples darão informações valiosas para o time do *Design Studio*. O PO anota tudo que foi falado na entrevista e leva para segundo dia.
- Dia 2 - Este dia é dedicado a convergir as informações obtidas na entrevista e todos os membros participam. Primeiramente o facilitador deve organizar a sala, e garantir os materiais para as atividades: *post-it*, canetas, pincéis para quadro branco e folhas.

Na primeira atividade os participantes, individualmente, devem anotar os *HMW's* e colá-los na parede ou quadro para serem votados por todos com o intuito de selecionar as melhores ideias.

Em seguida, baseando nos *HMW's* mais votados, é realizado o *Crazy 8's*. Após concluído cada integrante explica seus desenhos e então começa outra rodada de votação que selecionará os melhores esboços para servirem como inspiração para o *storyboard*, definido na seção 3.6.1.1.

No final do dia é feito o protótipo de baixa fidelidade baseado nas atividades realizadas anteriormente.

- Dia 3 - No terceiro dia somente o *designer* irá participar. Após ter entendido bem o problema e ter obtido ideias de pessoas com habilidades diferentes, o *designer* é capaz de criar um protótipo de alta fidelidade mais assertivo. Para a criação do protótipo no LEMAF é utilizado o Adobe XD.
- Dia 4 - O último dia é o de validação. O PO e *designer* apresentam o protótipo de alta fidelidade para o cliente que irá aprovar ou fazer algumas considerações que julgar necessário.

Na Figura 5.1 abaixo é possível visualizar a organização dos dias.

Figura 5.1 – Divisão dos dias no *Design Studio*.

Dia 1	Dia 2	Dia 3	Dia 4
Designer e PO	Equipe toda	Apenas o designer	Designer e PO
Entrevista com cliente 	Desenhar  Decidir 	Prototipar 	Validação com o cliente 

Fonte: Do autor

6 RESULTADOS

Os resultados do *Design Studio* foram de grande importância para os projetos em que se utilizou a metodologia. Alguns dos principais comentários dos participantes foram que o entendimento do problema e criação dos protótipos foram muito mais eficientes e claros quando comparado com o processo antigo e que foi economizado cerca de um mês de trabalho – alguns projetos que poderiam ser difíceis de serem compreendidos e levarem semanas até serem prototipados, em 4 dias são feitos os protótipos e já são validados com cliente utilizando o *Design Studio*.

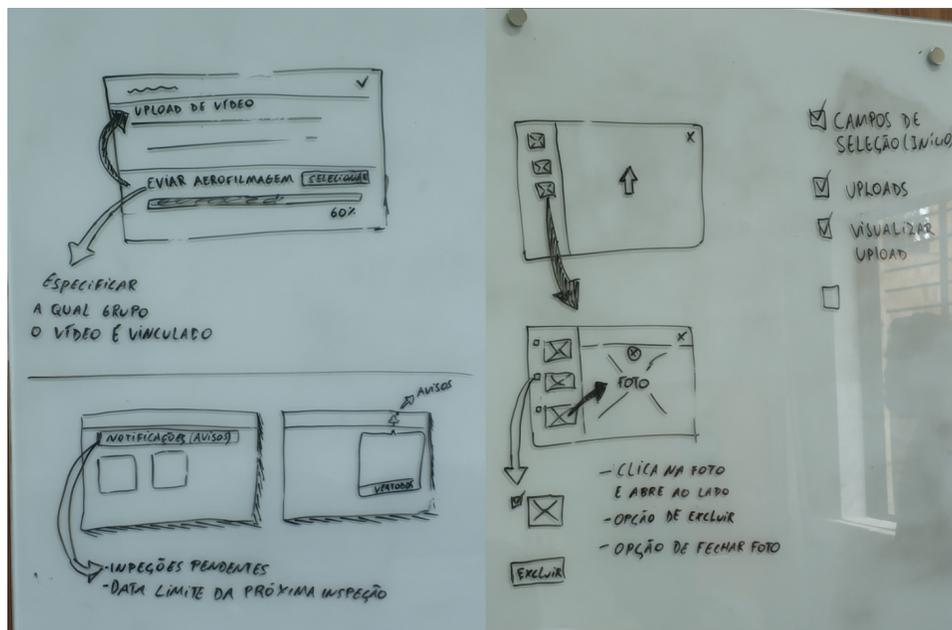
Um dos projetos que foi utilizado o *Design Studio* foi o CBPA - Cadastro de Barragens do Estado do Pará, que já encontra-se em produção em <<http://sistemas.semas.pa.gov.br/cbpa>>. Nele o cliente solicitou algumas novas funcionalidades, dentre elas: notificações na tela principal do sistema e envio de fotos dentro de um formulário. Foram quatro integrantes que participaram da *sprint*, um facilitador (*designer*), um decisor (PO), um *designer* e um *tester*. No segundo dia foram feitos os protótipos de baixa fidelidade, que podem ser vistos na Figura 6.1 e no terceiro dia, o *designer* responsável pelo projeto produziu os protótipos de alta fidelidade.

6.1 Participação do autor

As atividades do estagiário já foram elencadas na Seção 2.3. Especificamente para o desenvolvimento da metodologia *Design Studio*, o autor teve participação no projeto nas fases do segundo dia gerando ideias através dos *HMWs*, fazendo esboços na atividade *Crazy 8's*, foi o responsável por criar o protótipo de alta fidelidade no terceiro dia.

As Figuras 6.2, 6.3, 6.4, 6.5 e 6.6 mostram cinco protótipos de alta fidelidade. A Figura 6.2 é a tela de início do sistema, contendo as notificações pertinentes ao usuário e dois botões principais para gerenciamento de barragens

Figura 6.1 – Protótipo de baixa fidelidade produzido no segundo dia do *Design Studio*.

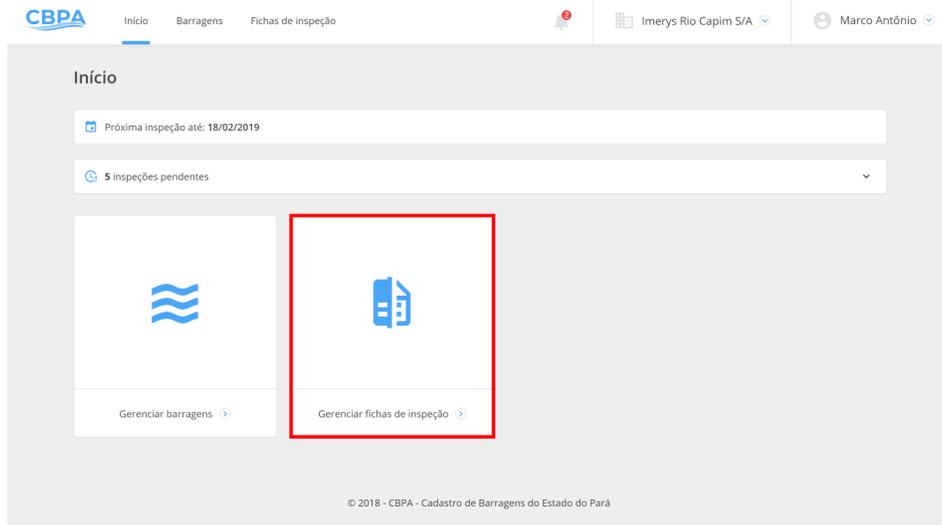


Fonte: Do autor

ou de fichas de inspeção, um clique sobre "Gerenciar fichas de inspeção" (área destacada) leva o usuário à tela com a listagem de barragens, que aparece na Figura 6.3. Um clique sobre "Entrar" (área destacada) da tela de Fichas de Inspeção (Figura 6.3) leva o usuário à tela da barragem selecionada, ilustrada na Figura 6.4, onde ao clicar em "Nova ficha de inspeção" (área destacada) leva o usuário à tela de cadastro de uma nova ficha de inspeção, conforme a Figura 6.5. No cadastro há um campo no qual o usuário deverá realizar o envio de de imagens aéreas da barragem (área destacada na Figura 6.5), ao clicar no botão de envio o usuário será levado à tela de envio das imagens, ilustrada na Figura 6.6.

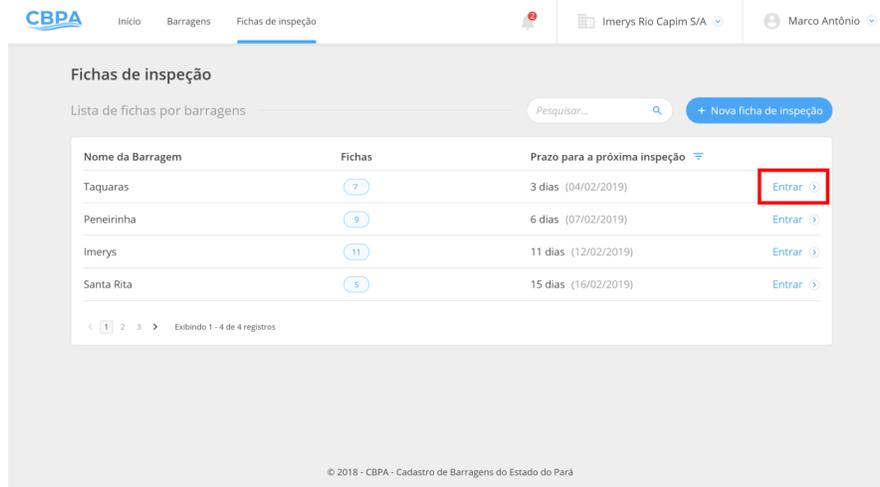
O protótipo de alta fidelidade produzido no Adobe XD implementa, inclusive essa a navegação descrita. Além dessas três telas de exemplos, o autor participou da produção de 6 outras telas para o mesmo sistema de Cadastro de Barragens do Estado do Pará, executando o *Design Studio* desenvolvido pela equipe.

Figura 6.2 – Protótipo de alta fidelidade da tela inicial do CBPA.



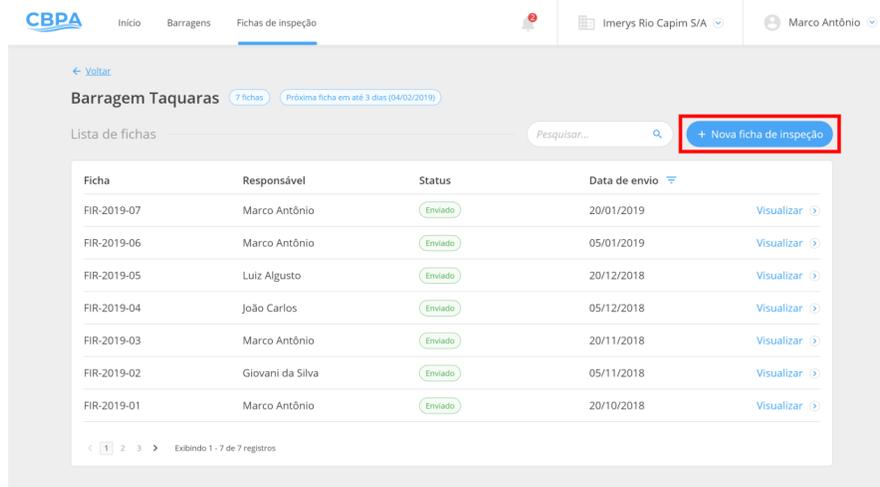
Fonte: Do autor

Figura 6.3 – Protótipo de alta fidelidade da tela de fichas de inspeção de cada barragem.



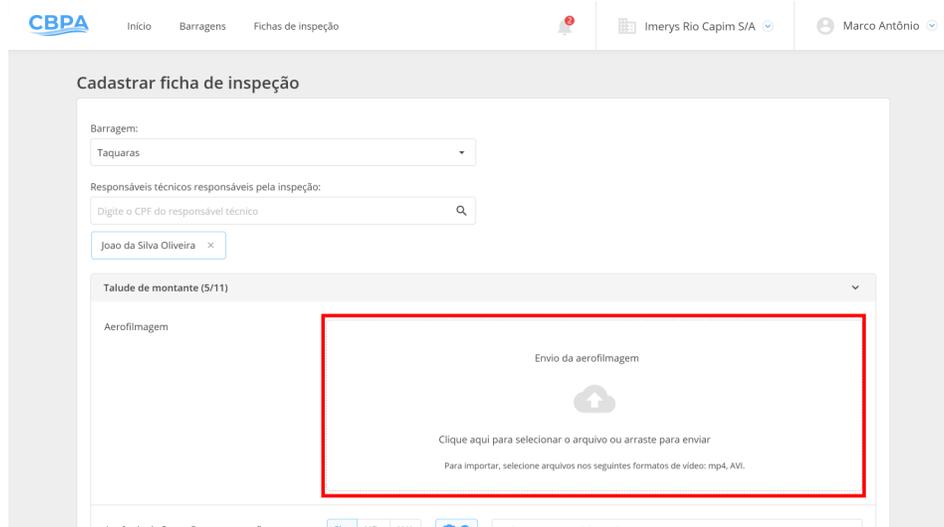
Fonte: Do autor

Figura 6.4 – Protótipo de alta fidelidade da tela de barragem selecionada.



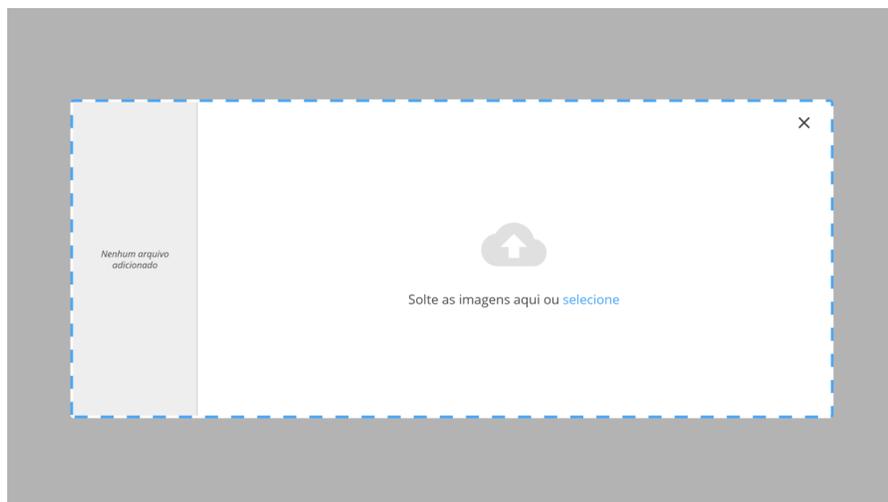
Fonte: Do autor

Figura 6.5 – Protótipo de alta fidelidade do cadastro de uma nova ficha de inspeção.



Fonte: Do autor

Figura 6.6 – Protótipo de alta fidelidade da tela de envio de imagens no cadastro.



Fonte: Do autor

7 CONCLUSÃO

O estágio é fundamental para o acadêmico que almeja ingressar no mercado de trabalho. Os conhecimentos adquiridos nas disciplinas cursadas durante a graduação foram essenciais para as atividades realizadas no estágio, principalmente em Interação Humano-Computador, onde pôde-se ver os conceitos teóricos de *design* de interação focado no usuário, acessibilidade, processo de *design*, entre outros assuntos de suma importância, e Engenharia de *Software*, disciplina que nos apresenta, além de outros temas, os conceitos das metodologias ágeis e levantamento de requisitos. Estes conceitos teóricos, somados a vivência e aprendizado prático, prepararam o estagiário para a vida profissional.

O objetivo deste trabalho foi mostrar como o *Design Studio* contribuiu para que as soluções de *design* de interface fossem desenvolvidas com maior rapidez e eficiência, o que pode ser visto a partir dos comentários dos integrantes que participaram das *sprints*, e pelo sucesso do resultado do projeto em que utilizou-se a metodologia.

Como sugestão de trabalhos futuros, recomenda-se aplicar o *Design Studio* em novos projetos para que se possa aperfeiçoar a metodologia ainda mais, adotar os testes com usuário, ferramenta fundamental em uma organização que desenvolve produtos, serviços e aplicativos utilizados por pessoas, e, mesmo com os limites da organização, como alocação de um time, horários, entre outros aspectos do tipo, se possa otimizar o processo de criação de soluções de *design* de interface.

Finalmente, conclui-se que este trabalho foi de grande importância para o estagiário pois a metodologia desenvolvida durante o estágio agregou muitos conhecimentos, e também para a organização, que agora conta com um processo de *design* de interface eficiente, onde poupam-se custos e evita-se retrabalho tanto de *design* quando de desenvolvimento.

REFERÊNCIAS

- ABRAHAMSSON, P. et al. Agile software development methods: Review and analysis. set. 2017. Disponível em: <<https://www.vtt.fi/inf/pdf/publications/2002/P478.pdf>>.
- ADOBE INC. **Adobe XD**. San Jose, CA, EUA, 2019. Disponível em: <<https://www.adobe.com/br/products/xd.html>>.
- BORGES, E. P. **Desenvolvimento ágil em números, no Brasil e no Mundo**. 2017. Disponível em: <<http://igti.com.br/blog/desenvolvimento-agil-em-numeros/>>. Acesso em: 07 jun. 2019.
- COUNCIL, D. **The Design Process: What is the Double Diamond?** 2015. Disponível em: <<https://www.designcouncil.org.uk/news-opinion/design-process-what-double-diamond>>. Acesso em: 14 jul. 2019.
- COURTNEY, J. **The Design Sprint 2.0: What is it and what does it look like?** 2018. Disponível em: <<https://www.invisionapp.com/inside-design/design-sprint-2/>>. Acesso em: 04 jun. 2019.
- IDEO. **What is Design Thinking?** 2019. Disponível em: <<https://www.ideo.com/blogs/inspiration/what-is-design-thinking>>. Acesso em: 05 jun. 2019.
- KNAPP, J.; ZERATSKY, J.; KOWITZ, B. **Sprint: How to Solve Big Problems and Test New Ideas in Just Five Days**. 1. ed. New York, NY, EUA: Simon Schuster, 2016.
- LEMAF. Sobre o lemaf. out. 2018. Disponível em: <<http://wiki.ti.lemaf.ufla.br/index.php/Sobre>>.
- POLIAKOVA, V. **Using Google Ventures Design Sprint Framework for Software Product Development in Startups**. 2017. Disponível em: <<http://urn.fi/URN:NBN:fi:amk-2017121922067>>.
- PREECE, J.; ROGERS, Y.; SHARP, H. **Interaction Design: Beyond Human-Computer Interaction**. Hoboken, Nova Jersey, EUA: Wiley, 2002.
- RODRIGUES, I. **Avaliação heurística de usabilidade no sistema de Cadastro de Barragens do Pará - CBPA**. Lavras: UFLA, 2018.
- SOMMERVILLE, I. **Engenharia de Software**. 8. ed. Campinas, SP: Pearson, 2011.
- TEIXEIRA, F. **Google Design Sprint: como funciona e como aplicar no seu projeto**. 2015. Disponível em: <<https://brasil.uxdesignn.cc/google-design-sprint-como-funciona-e-como-aplicar-no-seu-projeto-279107363659>>. Acesso em: 04 jun. 2019.